ITS Physical Architecture

Prepared by:

Lockheed Martin Federal Systems
Odetics Intelligent Transportation Systems Division

Prepared for:

Federal Highway Administration
US Department of Transportation
Washington, D. C. 20590

September 1998

Table of Contents

Executive Summary

The National ITS Architecture is a framework of physical elements on which ITS deployment, standards, and evaluation can be built. The framework consists of three layers: a transportation layer including functions required to implement ITS user services, a communication layer including identification of communication technologies and systems which will be used to exchange data required by the transportation layer, and an institutional layer which provides structure to the forces specifying requirements and deploying the architecture over time. The communications analysis is expanded in the *Communications Document.* The institutional layer is defined in the *Implementation Strategy*.

The transportation layer contains subsystems, terminators, and physical data flows between these entities. There are 19 subsystems representing groupings of functions defined in the logical architecture that may be operated by single entities. These 19 subsystems include Traffic Management, Transit Management, Commercial Vehicle Administration, Roadside, Toll Collection, Remote Traveler Support and Vehicles and others. The interfaces between these subsystems represent not only physical interfaces between equipment and computers but between operating agencies in the real world. Careful definition of these interfaces provides developers with an understanding of how to build components which will reliably intemperate with other components in future ITS deployments. While specifying the groupings of functionality and the interfaces between entities, the architecture development team has been careful to not over specify an ITS design. It is impossible to foresee what technology will be forthcoming or what roles agencies wish to play in ITS. Therefore, the architecture remains flexible indicating top level types of data that is exchanged and basic functions which are performed. It leaves the specific system design up to implementers and the interface standards development up to standards development bodies and stakeholders with specific domain knowledge and vested interest in the outcome.

Similarly, the communication layer specifies general requirements to make ITS functions able to communicate with each other. Only 4 types of traditional communication are called for with the assumption that users will adopt existing and emerging technologies as they develop. Specific recommendations are that beacon technologies are ideally suited to several types of ITS communication requirements where it is desirable to communicate with a vehicle within the immediate proximity in a very short period of time. More general communication requirements between vehicles and the infrastructure are accommodated with existing deployed wide area wireless technology. Evaluation of the requirements of the architecture can find no justification for significant expenditures for development of additional infrastructure for wide area communication for ITS over and above that currently emerging. Other communications involve advanced vehicle-vehicle and traditional wireline communications. Specific requirements for each type of communication are identified in the physical architecture in support of data loading and communication performance analysis in other documents.

Deployment of ITS will be by the operating agencies currently in existence and by entrepreneurs who see a chance to better support travel needs with new technology. The User Service requirements are re-distributed into deployment elements called Market Packages. Each Market Package represents a service which the architecture team feels will be deployed as an integrated capability. Looking at the architecture structure, one can decompose each of these services into elements which will be performed by each of the defined subsystems. These elements called Equipment Packages are the basic functions which implementers will develop or buy. The architecture specifies the general capabilities of each of these functions and the interface of the function with other relevant functions.

All elements of the architecture presented in this document are maintained in a database providing a tool for easy retrieval of information, consistency checking, and ready documentation. The databases can be shared with other groups developing more detailed versions of special applications such as CVO, Transit, Traffic Management, and ATIS. In this way, it is possible to maintain a coherent, integrated ITS database of functions, data flows, standards, and performance.

1. INTRODUCTION

The Physical Architecture identifies the physical subsystems and, architecture flows between subsystems that will implement the processes and support the data flows of the ITS Logical Architecture. The Physical Architecture further identifies the system terminator inputs (sources) and system terminator outputs (destinations) for architecture flows into and out of the system (see the Physical Architecture Context Diagram,).

1.1 System Overview

The Architecture is structured in 3 layers as shown in (Figure 1.1-1). A transportation layer performs transportation functions such as traffic management and traveler information provision. Functions (process specifications in the Logical Architecture) are assigned to subsystems such that the interfaces between subsystems represent candidate interfaces in the physical world. Subsystems were selected based on a limited set of criteria in order to:

- represent the functions of each major stakeholder group at a high level on in the physical architecture (e.g. Transit agencies, Emergency Services, Commercial Vehicle Operators all see both infrastructure and vehicle components)

- explicitly specify those functions that are performed in centers and those functions performed in vehicles (e.g. Route Planning could be performed either in the vehicle or by a service provider, route selection and turn-by-turn guidance is always done in the vehicle)

- collect functions currently performed by a single agency together (e.g. transit center functions are all performed in the Transit Management Subsystem, traffic management is performed in the Traffic Management Subsystem)

- separate functions that may in the future be split out to third party providers (e.g. provision of information in the Information Service Provider Subsystem, and data brokering in the Commercial Vehicle Administration Subsystem)

In order to provide for enough flexibility to stimulate market growth, the subsystems may be combined during system design within agencies, buildings, vehicles and so on. That means that a Traffic Management Center may include in addition to the Traffic Management Subsystem, an Information Service Provider Subsystem, and possibly an Emergency Management Subsystem.

The Communication layer represents the technology which will support the interfaces between transportation functions. Each data flow required by the transportation functions is evaluated with respect to the type of communication service which will be needed. The communication analysis document draws some conclusions regarding the current technical capabilities to support the requirements.

The Institutional layer represents the policy makers, planners and other users of the ITS services. These agencies and organizations are further addressed in Implementation Strategy Document.

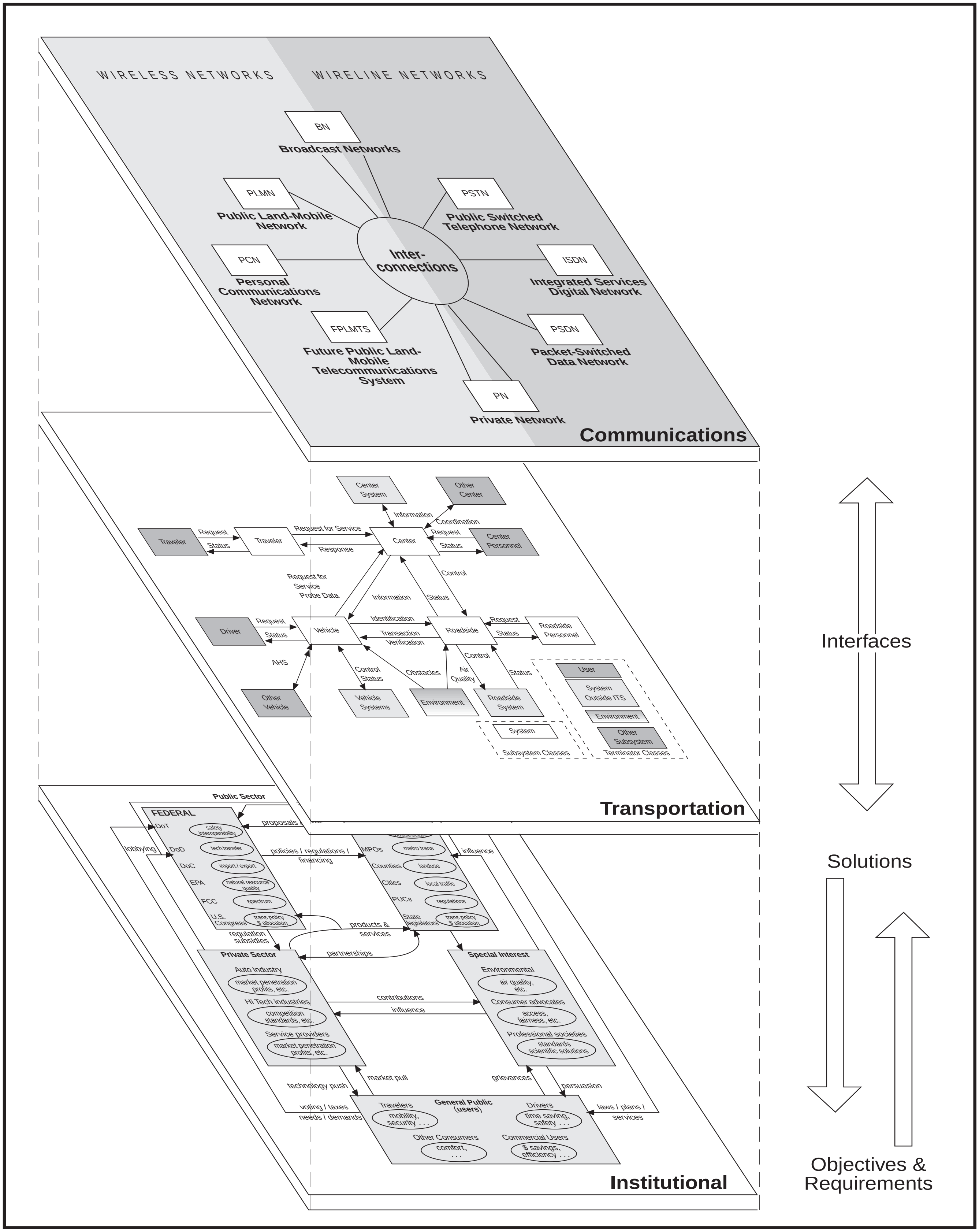


Figure 1.1-1 3 Layer Architecture

1.2 Strategies and Principles

The following are the strategies and principles that the Architecture team has followed in developing its ITS logical and physical architecture so as to best achieve the goals of ITS and the requirements of the User Services.

A summary of the strategies and principles is given below.

1. Low Entry Cost

Our architecture provides immediate service to all users, regardless of the degree of special instrumentation available to them. It does so by using all the information available from all terminators in order to devise the management strategy, and then disseminate the necessary information to the users through the information channels available to them. The fact that users having access to advanced channels can receive improved service will provide incentives for deployment of ITS instrumentation, but will not totally deny service to those users who do not own such channels. Basic service will be provided to the latter users through publicly available channels such as VMS, RDS and HAR.
The Architecture team is concerned that ITS benefits should be available to large numbers of commercial and private travelers at no cost or a small cost. Some examples of how we have designed this into the architecture are:

a. CVO with an ID tag. The architecture allows that a commercial vehicle have only an inexpensive electronic ID-tag in order to participate in the electronic clearance at roadside stations. The architecture further allows for enhanced tag technologies which may store detailed cargo and safety information for special purposes

b. Low/no cost traveler information services.

   Travelers will benefit from better regional travel information broadcast by commercial AM/FM/Cable operators if these operators use the travel information that is available from local TMCs and ISPs via ITS media interfaces. Also, extensive Highway Advisory Radio (HAR) can be deployed for local advisory information based on real-time TMC surveillance. Information will also be available to travelers from publicly provided kiosks.

c. Traditional Toll-tag services- the Architecture support the current deployment of toll-tags.

• 2. Provide Choices (in price/ performance) for travelers to receive user services

The ITS architecture provides not just a single implementation of each user service, but in many cases supports a multiplicity of implementations with varying performance and associated costs to the user.

For example, in the area of route guidance the architecture supports three distinct modes of operation:

a. Traveler-based route selection, where all route selection processing equipment as well as the navigable database that route selection is based on is included in equipment located with the traveler (either in their vehicle in their portable device).

b. Traveler-based route selection coupled to infrastructure-based and provided link/queue-times. In this method, the traveler-based route selection system is augmented by data from the infrastructure about current and possibly estimated future (predicted) link transit times and intersection queue delays. Using this type of data, the traveler will be able to use his equipment to compute better routes since his navigable road database will be augmented with information about current and future congestion conditions.

c. In-vehicle route guidance coupled to infrastructure based route selection. In this approach the infrastructure (in our architecture the ISP) selects the route based upon the traveler route request. Because of this approach the traveler equipment is simplified since it no longer requires a navigable map database, or the processing power to calculate a best route (only the processing power to display the route guidance).

3. Provide travelers with Privacy

In the area of privacy the ITS architecture takes into account that travelers have many distinct needs or desires with respect to privacy and the architecture provides the capability for these needs to be met: The route selection choices above offer a spectrum of options with respect to privacy for the ITS using traveler. The traveler can select routes totally independently of any infrastructure based entity, or they can choose a higher level of service that requires allowing the infrastructure to provide personalized service which requires sending personalized messages to their traveler based equipment.

4. Accommodate increasing levels of system integration

The ITS architecture is designed not only to support the introduction of new technologies, but has been designed in order to facility the ability of advances in technology to provide ever higher levels of system integration and hence ever higher levels of system performance. Such advanced concepts as Dynamic Traffic Assignment can be supported by the architecture. Through the coupling of Traffic Control and infrastructure based Route Selection the architecture can, when the technology permits approach optimum performance.

5. Assure equity

Providing an equitable division of benefits and costs is a key design strategy for the Architecture team. By splitting the key ITS infrastructure elements between private and public entities, the Architecture is able to assure equity in expenditures/payments. Public funds are used by public agencies (for example by running TMCs and Roadside facilities) to benefit all travelers equally, and private funds (and fees) are used to supply additional ”value added” services to those individuals willing to pay for those services.

6. Detailed, Open Standardization to maximize Interoperability, reduce market entry risk.

One of the most important requirements on the ITS architecture is interoperability- the ability for the user to obtain the user services nationwide with a single set of equipment. To do this the architecture effort must choose limited areas of detailed national standardization to maximize interoperability and market breadth for travelers and manufacturers. In particular, the traveler to infrastructure interface should be specified.

7. Leverage the existing and emerging open, infrastructures.

Communications is key to providing ITS user services to the wide range of users. The Architecture strategy is to maximize use of the existing (and planned future) communications infrastructure. This approach has the following benefits:

a. Minimize capital investment.

A national communications infrastructure to support ITS would be costly to deploy if it were only to be used for ITS. The capital formation necessary for this hypothetical ITS only infrastructure could substantially limit or slow the deployment of ITS.

b. Leverage existing infrastructure.

The Architecture ITS architecture makes use of existing commercially available wired and wireless data communications services.

c. Limit dependence on new spectrum allocations. New spectrum requirements which require FCC approval could delay and add considerable risk to the ITS evolutionary deployment.

8. Facilitate profitability for Private Industry to speed early deployment.

New travel information technologies will require capital investments to deploy. The private sector is best prepared to rapidly form capital and efficiently deploy advanced technologies.

Several subsystems in the architecture including the Commercial Vehicle Administration Subsystem and Information Service Provider Subsystem allow for a multiplicity of vendors to compete for the traveler and commercial vehicle business. Users will benefit from a competitive environment that will supply choices of service levels, privacy levels and cost levels.

9. Architecture is open and not biased towards any particular products.

Use of open standards is a priority of the architecture. All subsystems in the architecture will support a range of existing or anticipated product offerings from an unlimited range of hardware or service providers.

10. Encourage public-private infrastructure cooperation.

By carefully allocating processes to public and private subsystems, the Architecture has been designed to encourage mutually beneficial cooperation between public and private institutions through transportation surveillance and predictive model data exchange.

Examples of this are TMCs providing surveillance data to ISPs which will allow the ISPs to compute better routes for their clients, and the ISPs providing (anonymous) probe data to the TMCs so that their surveillance of non-instrumented roadways is enhanced and thus their ability to manage traffic for all travelers is enhanced. Similarly, TMCs can access traffic data stored at neighboring TMCs over the data network.

11. Enhance traveler safety.

The Architecture team believes that enhancing traveler safety is a key requirement for the architecture. There are many ways in which the Architecture will enhance safety. Some of these are:

a. Reduce emergency response time.

Our early safety analysis leads us to believe that any reduction in time between the occurrence of an injury accident and the arrival of medical help has a substantial impact on survivability. In the high-end state architecture, emergency vehicles will have their routes selected by the infrastructure, and those routes will be communicated to the TMC Traffic Management service package for priority signal service for the emergency vehicles (with minimal disruption to the rest of the transportation network). In addition, rapid data based deployment of emergency response vehicles via the Emergency Management Subsystem will get help to incidents faster, and will enhance traveler safety as a direct consequence.

b. Reduce congestion.

By using standardized interfaces that the architecture identifies, demand management strategies and policies can be effectively implemented, congestion can be reduced, thus reducing the number of transitions from free-flow to stop-and-go traffic conditions. These transitions have been identified in our preliminary safety analysis as a cause of traffic accidents.

c. Fail-safe infrastructure architecture.

The Architecture has not allocated any new life threatening functions to the infrastructure. Vehicle control (for collision avoidance) remains entirely within the vehicle subsystem (and in the case of platooning and AHS related functions, is based on communication directly between adjacent vehicles). In the event of a total infrastructure failure, signals would fall back to local sensor based signal control or fixed time plans, exactly as they do today.

12. Provide Locally determined Management Capabilities

For ITS to be desirable in some areas and in some time frames it must be able to address demand management (in addition to supply management). The architecture gives local agencies (and ultimately elected officials) enormous latitude to decide how a limited transportation resources are to be allocated.

The Architecture team has designed the architecture so that any particular form of demand management is optional, and a local decision to deploy. Examples include congestion pricing, vehicle class preferences, and extensions of ramp metering, HOV management, lane management.

1.3 Development Methodology

The Physical architecture contains the elements on which the evaluations, standards, and deployment and implementation strategies are built. It defines the framework for the whole architecture.

Architecture Definition

- **Subsystems -** Subsystems are the primary structural components of the Physical Architecture. Focus group concerns, institutional issues, and technology constraints and capabilities are used to determine subsystems which can most likely be supported by single institution, which perform functions which "belong" together, and whose interfaces may require standards to promote interoperability and compatibility.

- **Physical Architecture Flows -** Processes from the logical architecture are assigned to each of the subsystems according to stakeholder inputs. Architecture flows between subsystems are determined based on the data exchange implied by the process specification assignments and the data flows in the logical architecture.

- **Physical Architecture Interconnections -** Each type of data flowing between subsystems requires a specific type of interconnect. The collection of interconnects which support all data flows are defined in the communications layer of the architecture. The data loading analysis compares the capabilities of these interconnects with expected requirements under various deployments. This information is reported in the communication analysis documentation.

Iterative Architecture Refinement

Our approach to iterative architecture refinement exploits a technical methodology that rapidly and easily enables changes to the logical and physical architectures and easily regenerates data loading and other analysis. We have demonstrated that we can use this methodology to rapidly adapt our architecture based on consensus feedback and other architecture design analysis. We accept feedback in a number of different ways:

Customer inputs are collected based on the following:

1. Formally documented customer requirements. E.g. the *User Services.* These inputs were used as a starting basis for designing the architecture (as well as our own initial analysis).

2. Written feedback from the customer on the Architecture.

3. Direct dialog at Program Reviews.

Stakeholder feedback is included as well from:

1. *Customer initiated interactions.* E.g. Stakeholder Consensus Forum meetings sponsored and facilitated by the customer.

2. *Team initiated informal interactions.* Our own team meetings with stakeholders and stakeholder groups and visits to ITS demonstration sites.

3. *Team initiated Stakeholder Focus Groups.* The Team has held a number of formal focus groups with selected stakeholders at the University of Michigan. The purpose of these focus groups has been to gauge stakeholder perceptions and reactions to

specific ITS benefits, costs, and solutions.

Other Team Analysis

We held periodic team review meetings where the current status of the architecture is reviewed with all team members. Discussion at these meetings often focused on identifying features and detractors of the architecture based on various analyses that are proceeding as well as identifying changes to the architecture that can enhance its overall value.

1.4 Document Organization

The Physical Architecture is made up of many real components. These are presented in three layers: a Transportation Layer (chapter 2), Communications Layer (chapter 3), and an Institutional Layer (Implementation Strategy Document).

2. TRANSPORTATION LAYER

This layer of the architecture provides for the transportation related functions. Section 2.1 describes the architecture context, that is, what functions are defined in the architecture and what is considered outside the scope of the architecture. The functions defined in the architecture are contained within subsystems. The functions outside the scope of the architecture are represented by terminators. These two types of architectural elements (subsystems and terminators) are called entities. Classes bundle entities together into different categories such as humans, computers, and the physical world. Interfaces between these entities are sometimes very clearly data flows which can be carried by communication media. Some interfaces are fuzzier representing physical observation, contact, or human interaction.

Section 2.2 provides a Top Level representation of the architecture in terms of the entity classes. This representation indicates the different types of entities and the kinds of relations that the architecture include.

Section 2.3 defines the principal Physical Architecture components Subsystem terminators, flows, and some top level architecture flow diagrams.

The sections following section 2.3 contain descriptions of each subsystem (functions in the scope of the architecture). They begin with a brief summary of how to deploy the subsystem with other subsystems. It describes how the subsystem may logically be combined with other subsystems within one jurisdiction. Alternative implementations of the subsystem itself within one operating agency are also described.

Next, the equipment packages assigned to each subsystem are described. Equipment packages are a functional capability that may be deployed at some specific time. Each equipment packages is composed of a collection of process specifications from the logical architecture. These assigned process specifications are listed with each equipment package. A brief overview of each process specification is included to provide the reader with one stop access to subsystem information.

Logical data flows between subsystems are assigned by identifying the connections between process specifications assigned to different subsystems. Assignment of logical data flows between subsystems leads to the description of physical interconnections between subsystems. For each architecture flow identified in the physical architecture, the list of associated logical architecture data flows is provided. The architecture flows indicate the type of information that is expected to be exchanged between subsystems. The logical data flows provide the type of data required to execute the process specifications assigned to each subsystem.

2.1 Physical Architecture Context

The scope of "Context" of the physical architecture is identical to the scope of the logical architecture. The identical set of terminators establish exactly the same boundary. The physical architecture adds a broad structural overlay to the structured analysis model documented in the logical architecture. Figure 2.1-1 contains the physical architecture context diagram.

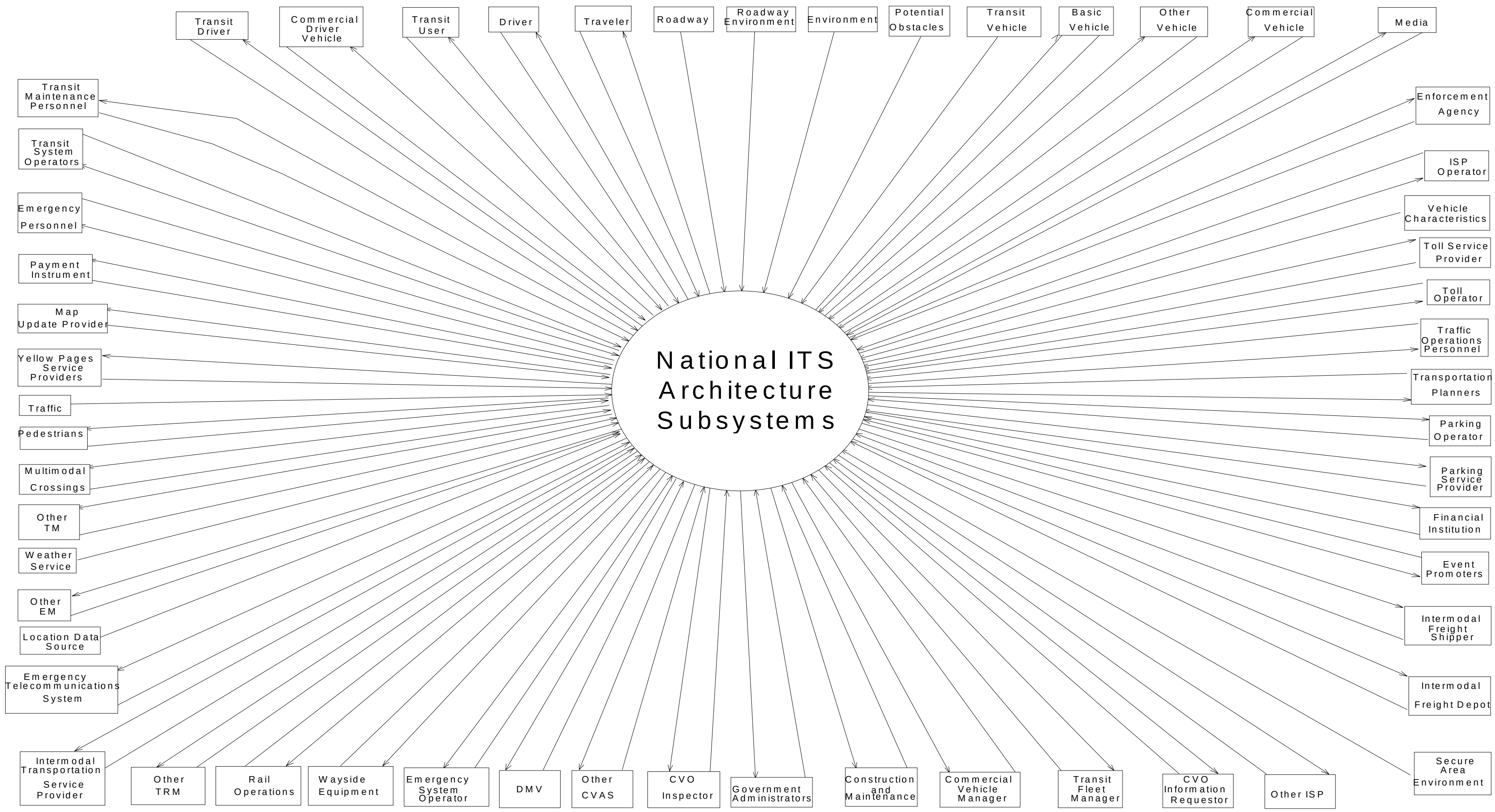


Figure 2.1-1 Architecture Context Diagram

2.2 Physical Architecture Decomposition

The architecture can be viewed at several levels of detail. The architecture team has defined collections of subsystems sufficient to express interfaces and processes that are essential to the development of interface standards that are necessary for nationwide interoperability. Multi-disciplinary considerations and the contributions of many different agencies all of whom have a stake in ITS have been incorporated. Efforts by other organizations (e.g., APL, ORNL Sandia) have further defined processes and interfaces within these subsystems or sets of subsystems to help support the detailed standards definition process within a particular discipline. Even though this National Architecture does not go into extreme detail for each User Service, the overall architecture is still very complex therefore it is presented in several levels of decomposition.

2.2.1 Top Level Architecture

Initial diagrams show the architecture at only the top level. Additional detail is added as it pertains to specific topics (e.g., stakeholders, technologies, Market Packages).

Figure 2.2-1. The box in the center represents the architecture subsystems while the outside boxes represent collections of terminators with which the subsystems interact. The lines between boxes represent at a high level the interfaces to the ITS system. Five classes of physical entities are defined:

- Subsystems - These perform transportation functions (e.g., collect data from the roadside, perform route planning, etc.). All of the functions are defined in the logical architecture as process specifications. Processes that are likely to be collected together under one physical agency, jurisdiction, physical unit are grouped together into a subsystem. This grouping is done to optimize the overall expected performance of the resulting ITS deployments taking into consideration anticipated communication technologies, performance, risk, deployment, etc. Significant detail is included for each of these subsystems and its interfaces.

- Users - These are people who interact with the architecture implementation. The people could either be travelers who use ITS to achieve travel goals, or operators of ITS who use features to streamline their operations, improve service, or make money. Each interface to a user involves human interaction with the system.

- Other Systems outside ITS - These are organizations or agencies that will likely interact with ITS through computer interfaces. These interfaces are similar to internal architecture interfaces.

- Environment - This is the physical world of pavement, air, obstacles and so-on.

- Other Subsystems within the Architecture - There may be a multiplicity of instantiations of each of the Architecture subsystems. To adequately model the interaction between these multiple implementations, one representative of each subsystem is explicitly included in the diagrams while those which it communicates with are represented as Other Subsystems.

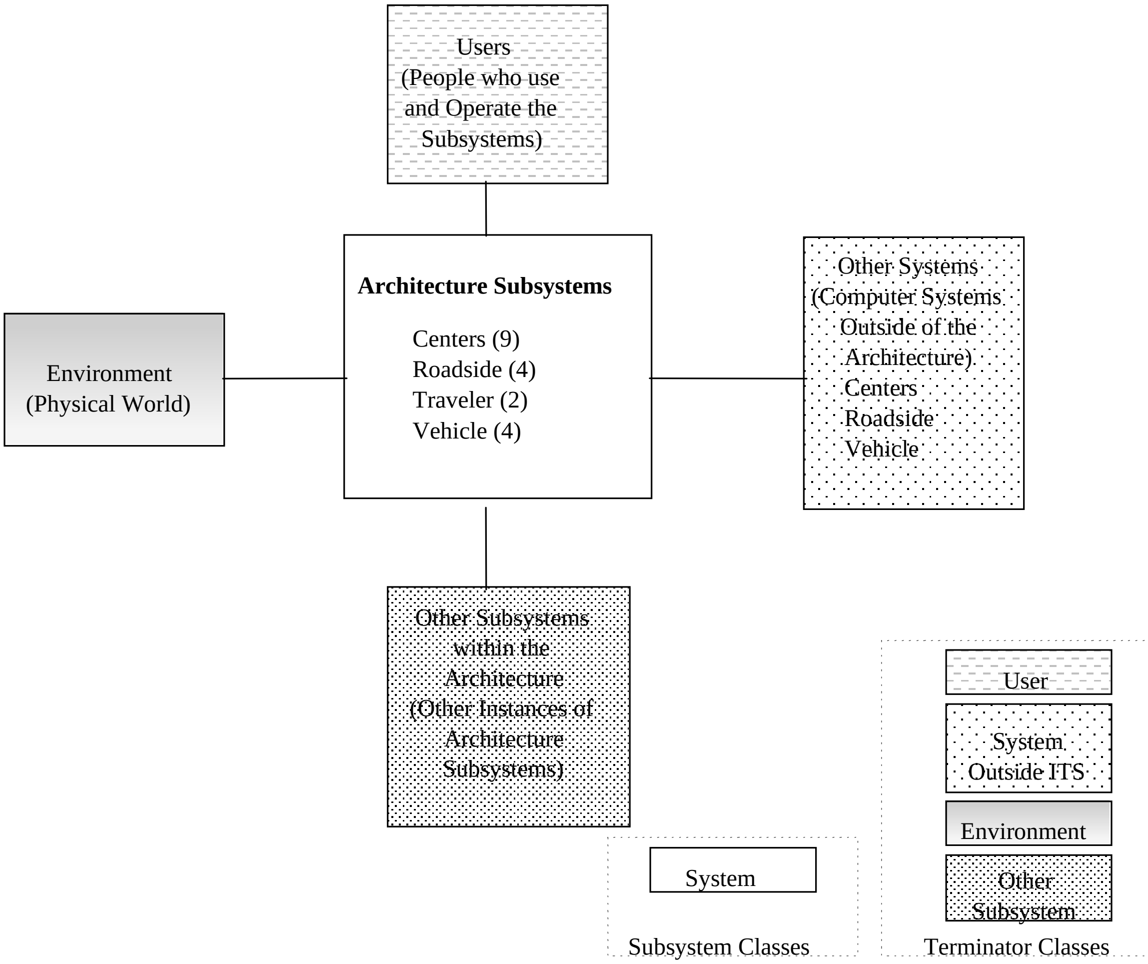


Figure 2.2-1 Simple View of ITS Architecture Structure

In ITS, there are four subclasses of subsystems, terminators, and users:

- Centers which collect and store information within the infrastructure
- Roadside which is deployed along the side of the road at many locations
- Vehicles
- Travelers representing ITS users with transportation needs

For example, other center systems may be a weather reporting agency or a law enforcement agency. Other subsystems within the Architecture could be peer Emergency Management subsystems or TMC's that reside in an adjacent jurisdiction.

Figure 2.2-2 presents a top level Architecture Flow Diagram. The diagram represents the four classes of subsystems, the terminators associated with each of the classes and the type of information that is exchanged between the classes. A definition of each of the entity classes in the figure is provided in Table 2.2-1.

The information types indicated in Figure 2.2-2 are exchanged between entity classes using different types of communication media. A very simplified view of this communications interface is provided in the Top Level simplified Architecture Interconnect Diagram in Figure 2.2-3. The details of each of the interconnections are further explained in the communications layer of the architecture.

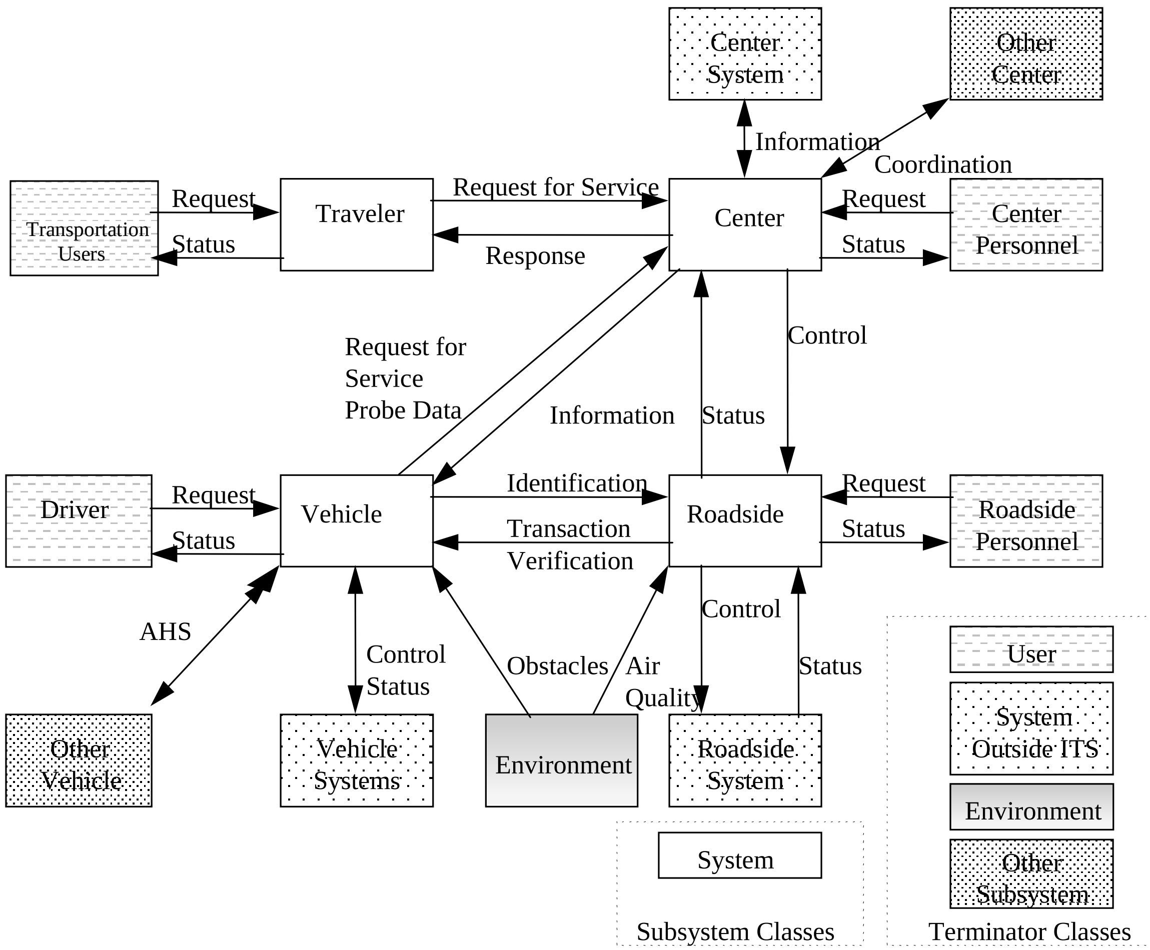


Figure 2.2-2 Top Level Architecture Flow Diagram

Table 2.2-1 Contents of Entity Classes

Entity	Entity Name	Entity Type	Entity Class
CVAS	Commercial Vehicle Administration	Subsystem	center
CVCS	Commercial Vehicle Check	Subsystem	roadside
CVS	Commercial Vehicle Subsystem	Subsystem	vehicle
EM	Emergency Management	Subsystem	center
EMMS	Emissions Management	Subsystem	center
EVS	Emergency Vehicle Subsystem	Subsystem	vehicle
FMS	Fleet and Freight Management	Subsystem	center
ISP	Information Service Provider	Subsystem	center
PIAS	Personal Information Access	Subsystem	traveler
PMS	Parking Management	Subsystem	roadside
PS	Planning Subsystem	Subsystem	center
RS	Roadway Subsystem	Subsystem	roadside
RTS	Remote Traveler Support	Subsystem	traveler
TAS	Toll Administration	Subsystem	center
TCS	Toll Collection	Subsystem	roadside
TMS	Traffic Management	Subsystem	center
TRMS	Transit Management	Subsystem	center
TRVS	Transit Vehicle Subsystem	Subsystem	vehicle
VS	Vehicle	Subsystem	vehicle
X01	Intermodal Freight Shipper	Terminator	center system
X02	Intermodal Transportation Service Provider	Terminator	center system
X03	Basic Vehicle	Terminator	vehicle system
X06	Commercial Vehicle Driver	Terminator	driver
X07	Commercial Vehicle Manager	Terminator	center personnel
X08	Commercial Vehicle	Terminator	vehicle system
X09	Construction and Maintenance	Terminator	center system
X10	CVO Inspector	Terminator	roadside personnel
X12	Driver	Terminator	driver
X13	Emergency Telecommunications System	Terminator	center system
X14	Emergency System Operator	Terminator	center personnel
X15	Emergency Personnel	Terminator	driver
X18	Environment	Terminator	environment
X19	Event Promoters	Terminator	center personnel
X21	Financial Institution	Terminator	center system
X22	Government Administrators	Terminator	center system
X23	Map Update Provider	Terminator	center system
X24	Yellow Pages Service Providers	Terminator	center personnel
X25	Transportation Planners	Terminator	center system
X26	Location Data Source	Terminator	environment
X27	Media	Terminator	center system
X29	Multimodal Crossings	Terminator	roadside system
X30	Other EM	Terminator	other subsystem
X31	Other ISP	Terminator	other subsystem
X33	Other TRM	Terminator	other subsystem

Table 2.2-1. Contents of Entity Classes (continued)

Entity	Entity Name	Entity Type	Entity Class
X34	Other Vehicle	Terminator	other subsystem
X35	Other TM	Terminator	other subsystem
X36	Parking Operator	Terminator	roadside personnel
X37	Parking Service Provider	Terminator	center system
X38	Pedestrians	Terminator	environment
X39	Potential Obstacles	Terminator	environment
X40	Roadway	Terminator	environment
X41	Roadway Environment	Terminator	environment
X42	Secure Area Environment	Terminator	environment
X43	Toll Operator	Terminator	roadside personnel
X44	Toll Service Provider	Terminator	center personnel
X45	Traffic	Terminator	environment
X46	Traffic Operations Personnel	Terminator	center personnel
X47	Transit Fleet Manager	Terminator	center personnel
X49	Transit System Operators	Terminator	center personnel
X50	Transit User	Terminator	traveler
X51	Transit Vehicle	Terminator	vehicle system
X52	Transit Driver	Terminator	driver
X53	Transit Maintenance Personnel	Terminator	center personnel
X56	Traveler	Terminator	traveler
X57	Vehicle Characteristics	Terminator	environment
X58	Weather Service	Terminator	center system
X59	Other CVAS	Terminator	other subsystem
X60	Intermodal Freight Depot	Terminator	roadside system
X61	Payment Instrument	Terminator	traveler
X62	Enforcement Agency	Terminator	Center System
X63	ISP Operator	Terminator	Center Personnel
X64	DMV	Terminator	Center System
X65	CVO Information Requestor	Terminator	Center System
X66	Wayside Equipment	Terminator	roadside system
X67	Rail Operations	Terminator	Center System

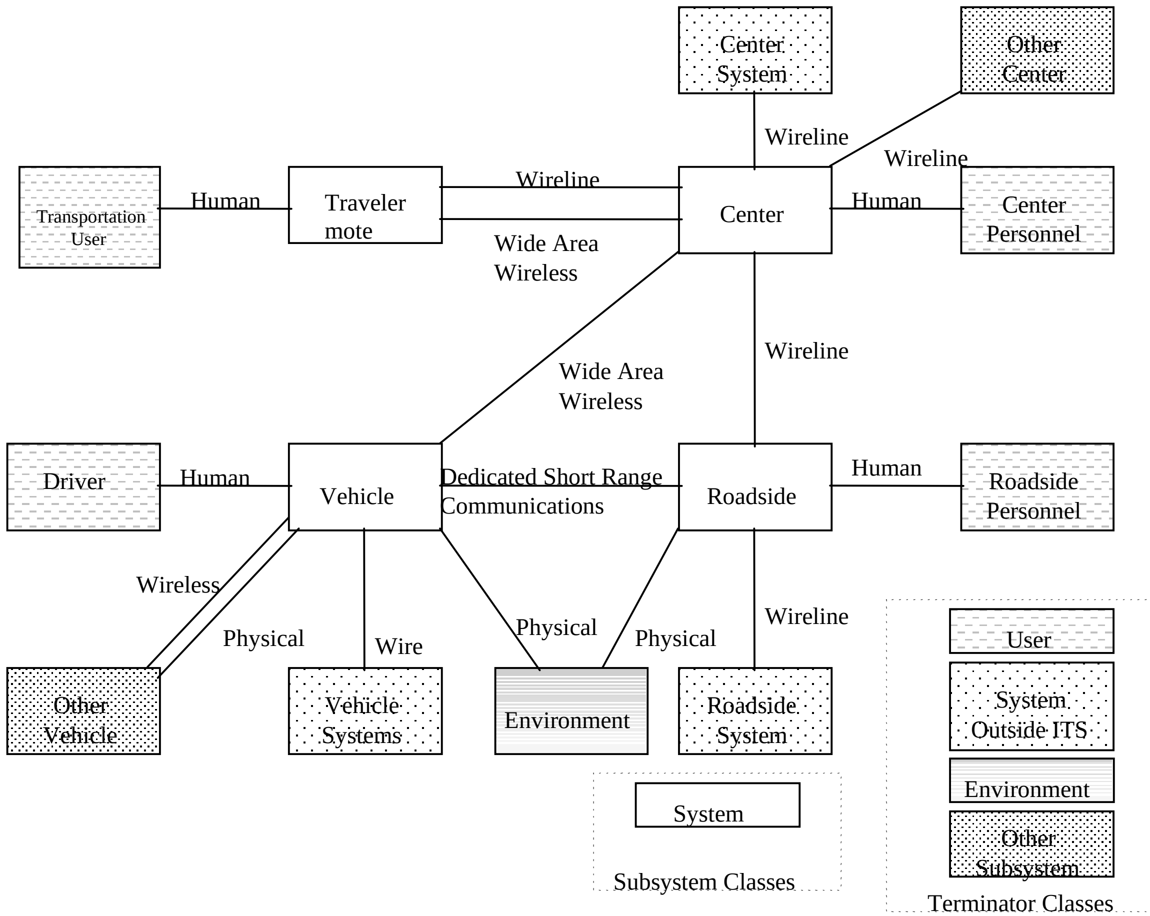


Figure 2.2-3. Top Level Architecture Interconnect Diagram

Expanding each of the entity classes represented in Table 2-1 results in a detailed level 0 Architecture Flow Diagram. Such a diagram is presented in Figures 3-x for each of the entities indicated in Table 2-1.

2.3 Physical Architecture Dictionary

2.3.1 Internal Entity Definitions

The ITS architecture subsystems may be grouped into four distinct subsystem classes that share basic functional, deployment, and institutional characteristics. These classes are used to frame top level descriptions for each of the subsystems in the following material.

Center Subsystems

Table 2.3-1, provide management, administration, and support functions for the transportation system. The center subsystems each communicate with other centers to enable coordination between modes and across jurisdictions within a region. The center subsystems also communicate with roadside and vehicle subsystems to gather information and provide information and control that is coordinated by the center subsystems.

Table 2.3-1 Center Subsystem Descriptions

Entity Name	Entity Description
Commercial Vehicle Administration	The Commercial Vehicle Administration Subsystem will operate at one or more fixed locations within a region. This subsystem performs administrative functions supporting credentials, tax, and safety regulations. It issues credentials, collects fees and taxes, and supports enforcement of credential requirements. This subsystem communicates with the Fleet Management Subsystems associated with the motor carriers to process credentials applications and collect fuel taxes, weight/distance taxes, and other taxes and fees associated with commercial vehicle operations. The subsystem also receives applications for, and issues special Oversize/Overweight and HAZMAT permits in coordination with other cognizant authorities. The subsystem coordinates with other Commercial Vehicle Administration Subsystems (in other states/regions) to support nationwide access to credentials and safety information for administration and enforcement functions. This subsystem supports communications with Commercial Vehicle Check Subsystems operating at the roadside to enable credential checking and safety information collection. The collected safety information is processed, stored, and made available to qualified stakeholders to identify carriers and drivers that operate unsafely.
Emergency Management	The Emergency Management Subsystem operates in various emergency centers supporting public safety including police and fire stations, search and rescue special detachments, and HAZMAT response teams. This subsystem interfaces with other Emergency Management Subsystems to support coordinated emergency response involving multiple agencies. The subsystem creates, stores, and utilizes emergency response plans to facilitate coordinated response. The subsystem tracks and manages emergency vehicle fleets using automated vehicle location technology and two-way communications with the vehicle fleet. Real-time traffic information received from the other center subsystems is used to further aide the emergency dispatcher in selecting the emergency vehicle(s) and routes that will provide the most timely response. Interface with the Traffic Management Subsystem allows strategic coordination in tailoring traffic control to support en-route emergency vehicles. Interface with the Transit Management Subsystem allows coordinated use of transit vehicles to facilitate response to major emergencies.

Table 2.3-1 Center Subsystem Descriptions (continued)

Entity Name	Entity Description
Emissions Management	This subsystem operates at a fixed location and may co-reside with the Traffic Management Subsystem or may operate in its own distinct location depending on regional preferences and priorities. This subsystem provides the capabilities for air quality managers to monitor and manage air quality. These capabilities include collecting emissions data from distributed emissions sensors within the roadway subsystem. These sensors monitor general air quality within each sector of the area and also monitor the emissions of individual vehicles on the roadway. The sector emissions measures are collected, processed, and used to identify sectors exceeding safe pollution levels. This information is provided to toll administration, traffic management, and transit management systems and used to implement strategies intended to reduce emissions in and around the problem areas. Emissions data associated with individual vehicles, supplied by the Roadway Subsystem, is also processed and monitored to identify vehicles that exceed standards. This subsystem provides any functions necessary to inform the violators and otherwise ensure timely compliance with the emissions standards.
Fleet and Freight Management	The Fleet and Freight Management Subsystem provides the capability for commercial drivers and dispatchers to receive real-time routing information and access databases containing vehicle and cargo locations as well as carrier, vehicle, cargo, and driver information. In addition, the capability to purchase credentials electronically shall be provided, with automated and efficient connections to financial institutions and regulatory agencies, along with post-trip automated mileage and fuel usage reporting. The Fleet Management Subsystem also provides the capability for Fleet Managers to monitor the safety of their commercial vehicle drivers and fleet. The subsystem also supports application for Hazmat credentials and makes information about Hazmat cargo available to agencies as required.

Table 2.3-1 Center Subsystem Descriptions (continued)

Entity Name	Entity Description
Information Service Provider	This subsystem collects, processes, stores, and disseminates transportation information to system operators and the traveling public. The subsystem can play several different roles in an integrated ITS. In one role, the ISP provides a general data warehousing function, collecting information from transportation system operators and redistributing this information to other system operators in the region and other ISPs. In this information redistribution role, the ISP provides a bridge between the various transportation systems that produce the information and the other ISPs and their subscribers that use the information. The second role of an ISP is focused on delivery of traveler information to subscribers and the public at large. Information provided includes basic advisories, real time traffic condition and transit schedule information, yellow pages information, ridematching information, and parking information. The subsystem also provides the capability to provide specific directions to travelers by receiving origin and destination requests from travelers, generating route plans, and returning the calculated plans to the users. In addition to general route planning for travelers, the ISP also supports specialized route planning for vehicle fleets. In this third role, the ISP function may be dedicated to, or even embedded within, the dispatch system. Reservation services are also provided in advanced implementations. The information is provided to the traveler through the Personal Information Access Subsystem, Remote Traveler Support Subsystem, and various Vehicle Subsystems through available communications links. Both basic one-way (broadcast) and personalized two-way information provision is supported. The subsystem provides the capability for an informational infrastructure to connect providers and consumers, and gather that market information needed to assist in the planning of service improvements and in maintenance of operations.
Planning Subsystem	The Planning Subsystem provides a data archiving and analysis function for the National ITS Architecture. It collects historical, current, and predicted transportation information from the other center subsystems. The collected information is used in analysis and evaluation of current transportation system performance and in planning for future transportation improvements. The broad data interfaces supported by this subsystem make transportation data available to researchers and planners to facilitate the deployment and operation of ITS services.

Table 2.3-1 Center Subsystem Descriptions (continued)

Entity Name	Entity Description
Toll Administration	The Toll Administration Subsystem provides general payment administration capabilities and supports the electronic transfer of authenticated funds from the customer to the transportation system operator. This subsystem supports traveler enrollment and collection of both pre-payment and post-payment transportation fees in coordination with the existing and evolving financial infrastructure supporting electronic payment transactions. The system may establish and administer escrow accounts depending on the clearinghouse scheme and the type of payments involved. This subsystem posts a transaction to the customer account and generates a bill (for post-payment accounts), debits an escrow account, or interfaces to the financial infrastructure to debit a customer designated account. It supports communications with the Toll Collection Subsystem to support fee collection operations. The subsystem also sets and administers the pricing structures and includes the capability to implement road pricing policies in coordination with the Traffic Management Subsystem. The electronic financial transactions in which this subsystem is an intermediary between the customer and the financial infrastructure shall be cryptographically protected and authenticated to preserve privacy and ensure authenticity and audibility.
Traffic Management	The Traffic Management Subsystem operates within a traffic management center or other fixed location. This subsystem communicates with the Roadway Subsystem to monitor and manage traffic flow. Incidents are detected and verified and incident information is provided to the Emergency Management Subsystem, travelers (through Roadway Subsystem Highway Advisory Radio and Dynamic Message Signs), and to third party providers. The subsystem supports HOV lane management and coordination, road pricing, and other demand management policies that can alleviate congestion and influence mode selection. The subsystem monitors and manages maintenance work and disseminates maintenance work schedules and road closures. The subsystem also manages reversible lane facilities, and processes probe vehicle information. The subsystem communicates with other Traffic Management Subsystems to coordinate traffic information and control strategies in neighboring jurisdictions. It also coordinates with rail operations to support safer and more efficient highway traffic management at highway-rail intersections. Finally, the Traffic Management Subsystem provides the capabilities to exercise control over those devices utilized for AHS traffic and vehicle control.

Table 2.3-1 Center Subsystem Descriptions (continued)

Entity Name	Entity Description
Transit Management	The transit management subsystem manages transit vehicle fleets and coordinates with other modes and transportation services. It provides operations, maintenance, customer information, and planning and management functions for the transit property. It spans distinct central dispatch and garage management systems and supports the spectrum of fixed route, flexible route, and paratransit services. The subsystem's interfaces allow for communication between transit departments and with other operating entities such as emergency response services and traffic management systems. This subsystem receives special event and real-time incident data from the traffic management subsystem. It provides current transit operations data to other center subsystems. The Transit Management Subsystem collects and stores accurate readership levels and implements corresponding fare structures. It collects operational and maintenance data from transit vehicles, manages vehicle service histories, and assigns drivers and maintenance personnel to vehicles and routes. The Transit Management Subsystem also provides the capability for automated planning and scheduling of public transit operations. It furnishes travelers with real-time travel information, continuously updated schedules, schedule adherence information, transfer options, and transit routes and fares. In addition, the monitoring of key transit locations with both video and audio systems is provided with automatic alerting of operators and police of potential incidents including support for traveler activated alarms.

Roadside Subsystems

These infrastructure subsystems presented in Table 2.3-2 provide the direct interface to the roadway network, vehicles traveling on the roadway network, and travelers in transit. Each of the roadway subsystems includes functions that require distribution to the roadside to support direct surveillance, information provision, and control plan execution. All roadside subsystems interface to one or more of the center subsystems that govern overall operation of the roadside subsystems. The roadside subsystems also generally include direct user interfaces to drivers and transit users and short-range interfaces to the Vehicle Subsystems to support operations.

Table 2.3-2 Roadside Subsystem Descriptions

Entity Name	Entity Description
Commercial Vehicle Check	The Commercial Vehicle Check Subsystem supports automated vehicle identification at mainline speeds for credential checking, roadside safety inspections, and weigh-in-motion using two-way data exchange. These capabilities include providing warnings to the commercial vehicle drivers, their fleet managers, and proper authorities of any safety problems that have been identified, accessing and examining historical safety data, and automatically deciding whether to allow the vehicle to pass or require it to stop with operator manual override. The Commercial Vehicle Check Subsystem also provides supplemental inspection services to current capabilities by supporting expedited brake inspections, the use of operator hand-held devices, on-board safety database access, and the enrollment of vehicles and carriers in the electronic clearance program.
Parking Management	The Parking Subsystem provides the capability to provide parking availability and parking fee information, allow for parking payment without the use of cash with a multiple use medium, and support the detection, classification, and control of vehicles seeking parking.
Roadway Subsystem	This subsystem includes the equipment distributed on and along the roadway which monitors and controls traffic. Equipment includes highway advisory radios, variable message signs, cellular call boxes, CCTV cameras and video image processing systems for incident detection and verification, vehicle detectors, traffic signals, grade crossing warning systems, and freeway ramp metering systems. This subsystem also provides the capability for emissions and environmental condition monitoring including weather sensors, pavement icing sensors, fog etc. HOV lane management and reversible lane management functions are also available. In advanced implementations, this subsystem supports automated vehicle safety systems by safely controlling access to and egress from an Automated Highway System through monitoring of, and communications with, AHS vehicles. Intersection collision avoidance functions are provided by determining the probability of a collision in the intersection and sending appropriate warnings and/or control actions to the approaching vehicles.
Toll Collection	The Toll Collection Subsystem provides the capability for vehicle operators to pay tolls without stopping their vehicles using locally determined pricing structures and including the capability to implement various variable road pricing policies. Each transaction is accompanied by feedback to the customer which indicates the general status of the customer account. A record of the transactions is provided to the Toll Administration subsystem for reconciliation and so that the customer can periodically receive a detailed record of the transactions.

Vehicle Subsystems

These subsystems Table 2.3-3 are all vehicle-based and share many general driver information, vehicle navigation, and advanced safety systems functions. The vehicle subsystems communicate with the roadside subsystems and center subsystems for provision of information to the driver. In the following descriptions, the Personal Vehicle Subsystem description includes general traveler information and vehicle safety functions that are also applicable to the three fleet vehicle subsystems (Commercial Vehicle Subsystem, Emergency Vehicle Subsystem, and Transit Vehicle Subsystem). The fleet vehicle subsystems all include vehicle location and two-way communications functions that support efficient fleet operations. Each of the three fleet vehicle subsystems also include functions that support their specific service area.

Table 2.3-3 Vehicle Subsystem Descriptions

Entity Name	Entity Description
Commercial Vehicle Subsystem	This subsystem resides in a commercial vehicle and provides the sensory, processing, storage, and communications functions necessary to support safe and efficient commercial vehicle operations. The Commercial Vehicle Subsystem provides two-way communications between the commercial vehicle drivers, their fleet managers, and roadside officials, and provides HAZMAT response teams with timely and accurate cargo contents information after a vehicle incident. This subsystem provides the capability to collect and process vehicle, cargo, and driver safety data and status and alert the driver whenever there is a potential safety problem. Basic identification and safety status data are supplied to inspection facilities at mainline speeds. In addition, the subsystem will automatically collect and record mileage, fuel usage, and borders crossings.
Emergency Vehicle Subsystem	This subsystem resides in an emergency vehicle and provides the sensory, processing, storage, and communications functions necessary to support safe and efficient emergency response. The Emergency Vehicle Subsystem includes two-way communications to support coordinated response to emergencies in accordance with an associated Emergency Management Subsystem. Emergency vehicles are equipped with automated vehicle location capability for monitoring by vehicle tracking and fleet management functions in the Emergency Management Subsystem. Using these capabilities, the appropriate emergency vehicle to respond to each emergency is determined. Route guidance capabilities within the vehicle enable safe and efficient routing to the emergency. In addition, the emergency vehicle may be equipped to support signal preemption through communications with the roadside subsystem.
Transit Vehicle Subsystem	This subsystem resides in a transit vehicle and provides the sensory, processing, storage, and communications functions necessary to support safe and efficient movement of passengers. The Transit Vehicle Subsystem collects accurate ridership levels and supports electronic fare collection. An optional traffic signal prioritization function communicates with the roadside subsystem to improve on-schedule performance. Automated vehicle location functions enhance the information available to the Transit Management Subsystem enabling more efficient operations. On-board sensors support transit vehicle maintenance. The Transit Vehicle Subsystem also furnishes travelers with real-time travel information, continuously updated schedules, transfer options, routes, and fares.

Table 2.3-3 Vehicle Subsystem Description (continued)

Entity Name	Entity Description

Table 2.3-3 Vehicle Subsystem Description (continued)

Entity Name	Entity Description
Vehicle	This subsystem resides in an automobile and provides the sensory, processing, storage, and communications functions necessary to support efficient, safe, and convenient travel by personal automobile. Information services provide the driver with current travel conditions and the availability of services along the route and at the destination. Both one-way and two-way communications options support a spectrum of information services from low-cost broadcast services to advanced, pay for use personalized information services. Route guidance capabilities assist in formulation of an optimal route and step by step guidance along the travel route. Advanced sensors, processors, enhanced driver interfaces, and actuators complement the driver information services so that, in addition to making informed mode and route selections, the driver travels these routes in a safer and more consistent manner. Initial collision avoidance functions provide “vigilant co-pilot” driver warning capabilities. More advanced functions assume limited control of the vehicle to maintain safe headway. Ultimately, this subsystem supports completely automated vehicle operation through advanced communications with other vehicles in the vicinity and in coordination with supporting infrastructure subsystems. Pre-crash safety systems are deployed and emergency notification messages are issued when unavoidable collisions do occur.

Table **2.3-4** include the equipment that is typically owned and operated by the traveler. Though this equipment is often general purpose in nature and used for a variety of tasks, this equipment is specifically used for gaining access to traveler information within the scope of the ITS architecture. These subsystems interface to the information provider (one of the center subsystems, most commonly the Information Service Provider Subsystem) to access the traveler information. A range of service options and levels of equipment sophistication are considered and supported. Specific equipment included in this subsystem class include personal computers, telephones, personal digital assistants (PDAs), televisions, and any other communications-capable consumer products that can be used to supply information to the traveler.

Table 2.3-4 Traveler Subsystem Descriptions

Entity Name	Entity Description
Personal Information Access	This subsystem provides the capability for travelers to receive formatted traffic advisories from their homes, place of work, major trip generation sites, personal portable devices, and over multiple types of electronic media. These capabilities shall also provide basic routing information and allow users to select those transportation modes that allow them to avoid congestion, or more advanced capabilities to allow users to specify those transportation parameters that are unique to their individual needs and receive travel information. This subsystem shall provide capabilities to receive route planning from the infrastructure at fixed locations such as in their homes, their place of work, and at mobile locations such as from personal portable devices and in the vehicle or perform the route planning process at a mobile information access location. This subsystem shall also provide the capability to initiate a distress signal and cancel a prior issued manual request for help.
Remote Traveler Support	This subsystem provides access to traveler information at transit stations, transit stops, other fixed sites along travel routes, and at major trip generation locations such as special event centers, hotels, office complexes, amusement parks, and theatres. Traveler information access points include kiosks and informational displays supporting varied levels of interaction and information access. At transit stops, simple displays providing schedule information and imminent arrival signals can be provided. This basic information may be extended to include multi-modal information including traffic conditions and transit schedules along with yellow pages information to support mode and route selection at major trip generation sites. Personalized route planning and route guidance information can also be provided based on criteria supplied by the traveler. In addition to traveler information provision, this subsystem also supports public safety monitoring using CCTV cameras or other surveillance equipment and emergency notification within these public areas. Fare card maintenance, and other features, which enhance traveler convenience, may also be provided at the discretion of the deploying agency.

2.3.2 Terminator Descriptions

Entities that establish the boundary of the architecture are defined in the Logical Architecture as terminators. These terminators are replicated here in Table 2.3-5.

Table 2.3-5 Terminator Descriptions

Entity Name	Entity Description
Basic Vehicle	This terminator represents the basic vehicle platform that interfaces with and hosts ITS electronics. The Basic Vehicle terminator provides an interface to drive train, driver convenience and entertainment systems, and other non-ITS electronics on-board the vehicle. This interface allows general vehicle systems (e.g., the stereo speaker system) to be shared by ITS and non-ITS systems. It also allows monitoring and control of the vehicle platform for advanced vehicle control system applications.

Table 2.3-5 Terminator Descriptions (continued)

Entity Name	Entity Description
Commercial Vehicle	The actual commercial vehicle along with the special aspects of large commercial vehicles and vehicles designed to carry cargo that extend beyond the characteristics defined for the Basic Vehicle. This terminator thus represents a special type of Basic Vehicle that is used to transport goods or services which are operated by professional drivers, typically administered as part of a larger fleet, and regulated by a Commercial Vehicle Manager. This classification applies to all such vehicles ranging from small panel vans used in local pick-up and delivery services to large, multi-axle tractor trailer rigs operating on long haul routes.
Commercial Vehicle Driver	This terminator represents the human entity that operates vehicles transporting goods including both long haul trucks and local pick up and delivery vans. This terminator is complementary to the Driver terminator in that it represents those interactions which are unique to Commercial Vehicle Operations. In general, a "real world" commercial vehicle driver will interact as both a Driver and a CVO Driver. Data flowing from the Commercial Vehicle Driver terminator will include those system inputs specific to Commercial Vehicle Operations, such as information back to the Commercial Vehicle Manager. Data flowing to the Commercial Vehicle Driver may include system outputs such as commands to pull into a roadside safety inspection facility. Showing the Driver as the external interface includes the user interface devices within the ITS architecture boundary. The CVO Driver will be expected to interact with the ITS with interface devices designed to provide support for their usage.
Commercial Vehicle Manager	This terminator represents the human entities that are responsible for the dispatching and management of Commercial Vehicle fleets (e.g. traditional Fleet Managers). It may be many people in a large tracking organization but it can also be a single person (owner driver) in the case of single vehicle fleets. The Commercial Vehicle Manager provides instructions and coordination for Commercial Vehicles, including electronic clearance and tax filing, and receives the status of the Vehicles in the fleet that they manage. The Commercial Vehicle Manager is expected to interface with ITS on a regular basis to enhance productivity. Many interfaces with the system are also provided through normal user interfaces This interface is specific to CVO and is intended to complement these other interfaces.
Construction and Maintenance	This terminator represents the information systems that are used to manage and track construction and maintenance of the roadway infrastructure. These Construction and Maintenance systems are used by roadway maintenance personnel, roadway construction personnel, or other work crew personnel assigned to highway construction and maintenance. Coordination with these systems allows the ITS Architecture to rapidly correct deficiencies noted through its advanced surveillance capabilities and also improves the quality and accuracy of information available to Travelers regarding closures and other roadway construction and maintenance activities.

Table 2.3-5 Terminator Descriptions (continued)

Entity Name	Entity Description
CVO Information Requestor	This terminator represents any organization requesting information from the CVO Information Exchange network. It typically represents insurance companies requesting safety information on carriers etc.
CVO Inspector	This terminator represents the human entities who perform regulatory inspection of Commercial Vehicles in the field. CVO Inspectors support the roadside inspection, weighing, and checking of credentials either through automated preclearance or manual methods. The CVO Inspector is an inspection and enforcement arm of the regulatory agencies with frequent direct interface with the Commercial Vehicles and their Drivers.
DMV	This terminator represents a specific (state) public organization responsible for registering vehicles, e.g., the Department of Motor Vehicles. The DMV terminator is a special case of the Government Administrators terminator but in some areas is identified separately to emphasize the specific nature of the data being exchanged, i.e. vehicle identification.
Driver	This terminator represents the human entity that operates a licensed vehicle on the roadway. Included are operators of private, Transit, Commercial, and Emergency vehicles where the data being sent or received is not particular to the type of vehicle. Thus this external originates driver requests and receives driver information that reflects the interactions which might be useful to all drivers, regardless of vehicle classification. The Driver terminator is the operator of the Basic Vehicle terminator. Information and interactions which are unique to drivers of a specific vehicle type (e.g., fleet interactions with transit, commercial, or emergency vehicle drivers) are covered separately.
Emergency Personnel	This terminator represents personnel that are responsible for police, fire, emergency medical services, towing, and other special response team (e.g., hazardous material clean-up) activities at an incident site. These personnel are associated with the Emergency Vehicle Subsystem during dispatch to the incident site, but often work independently of the Emergency Vehicle Subsystem while providing their incident response services. Emergency personnel may include an Officer in Charge (OIC) and a crew. When managing an incident following standard Incident Command System practices, the on-site emergency personnel form an organizational structure under the auspices of an Incident Commander.
Emergency System Operator	This terminator represents the human entity that monitors all ITS emergency requests, (including those from the E911 Operator) and sets up pre-defined responses to be executed by an emergency management system. The operator may also override predefined responses where it is observed that they are not achieving the desired result. This terminator includes dispatchers who manage an emergency fleet (police, fire, ambulance, HAZMAT, etc.) or higher order emergency managers who provide response coordination during emergencies.

Table 2.3-5 Terminator Descriptions (continued)

Entity Name	Entity Description
Emergency Telecommunications System	This terminator represents the telecommunications systems that connect a caller with a Public Safety Answering Point (PSAP). These systems transparently support priority wireline and wireless caller access to the PSAP through 9-1-1 and other access mechanisms like 7 digit local access numbers, and motorist aid call boxes. The calls are routed to the appropriate PSAP, based on caller location when this information is available. When available, the caller's location and call-back number are also provided to the PSAP by this interface.
Enforcement Agency	This terminator represents an external entity which receives reports of violations detected by various ITS facilities, e.g. individual vehicle emissions, toll violations, CVO violations, etc.
Environment	This terminator is the operational setting in which the ITS interfaces and operates. This setting consists of weather effects such as snow, rain, fog, pollution, dust, temperature, humidity, solar radiation, and man made electromagnetic (RF) effects. Environmental conditions must be monitored by the ITS Architecture so that Travelers may be informed and control strategies can reflect adverse environmental conditions in a timely fashion.
Event Promoters	This terminator represents external Special Event Sponsors that have knowledge of events that may impact travel on roadways or other modal means. Examples of special event sponsors include sporting events, conventions, motorcades/parades, and public/political events. These promoters interface to the ITS to provide event information such as date, time, estimated duration, location, and any other information pertinent to traffic movement in the surrounding area.
Financial Institution	This terminator represents the organization that handles all electronic fund transfer requests to enable the transfer of funds from the user of the service to the provider of the service. The functions and activities of financial clearinghouses are subsumed by this entity.
Government Administrators	This terminator represents those public organizations responsible for regulating commercial vehicle operations, e.g., the Interstate Commerce Commission, state commerce offices, state Department of Motor Vehicles, state Department of Revenue, and Department of Transportation. Regulatory Agencies are envisioned to be an integral part of the ITS Commercial Vehicle Operations (CVO) as they will be directly involved with issuance of licenses, permits and other credentials for preclearance, provide database information to support most CVO services, and will receive, distribute, and audit CVO related taxes.
Intermodal Freight Depot	A Depot operated either by an ITS Freight manager or an alternate mode freight shipper which is capable of tracking cargo as it is moved from one mode to another.
Intermodal Freight Shipper	This terminator represents specialist organizations that engage in the shipment of freight by means other than road. They enable ITS to move goods as opposed to people on routes that require the use of other modes of transportation such as heavy rail, air, sea, etc. An example is a shipping agency which interfaces with Freight and Fleet Managers to transfer cargo from one mode to another.

Table 2.3-5 Terminator Descriptions (continued)

Entity Name	Entity Description
Intermodal Transportation Service Provider	This terminator provides the interface through which Transportation Service Providers can exchange data with ITS. They are the operators of non-roadway transportation systems (e.g. airlines, ferry services, passenger carrying heavy rail) . This two-way interface enables coordination for efficient movement of people and goods across multiple transportation modes. It also enables the traveler and shipper to efficiently plan itineraries which include segments using modes not directly included in the ITS User Services.
ISP Operator	This terminator is the human entity that may be physically present at the ISP to monitor the operational status of the facility and provide human interface capabilities to travelers and other ISP subsystems.
Location Data Source	This terminator represents an external entity which provides accurate position information. External systems which use GPS, terrestrial trilateration, or driver inputs are potential examples. This terminator contains sensors such as radio position receivers (e.g. GPS) and/or dead reckoning sensors (e.g. odometer, differential odometer, magnetic compass, gyro, etc.). This external implies that some additional functionality associated with developing an absolute position is outside the system and will not be directly modeled by the logical or physical architecture representations of the system.
Map Update Provider	This terminator represents a third-party developer and provider of digitized map databases used to support ITS services. It supports the provision of the databases that are required exclusively for route guidance (navigable_map) as well as those that are used exclusively for display by operators and at traveler information points, e.g. kiosks (display_map).
Media	This terminator represents the information systems that provide traffic reports, travel conditions, and other transportation-related news services to the traveling public through radio, TV, and other media. Traffic and travel advisory information that are collected by ITS are provided to this terminator. It is also a source for traffic flow information, incident and special event information, and other events which may have implications for the transportation system.
Multimodal Crossings	This terminator represents the control equipment that interfaces to a non-road based transportation system at an interference crossing with the roadway. The majority of these crossings are railroad grade crossings that are more specifically addressed by the "Wayside Equipment" terminator. This terminator addresses similar interface requirements, but for other specialized intersections like draw bridges at rivers and canals. Like highway-rail intersections, these other multimodal crossings carry traffic that may take priority over the road traffic at the intersection. The data provided will in its basic form be a simple "stop road traffic" indication. However more complex data flows may be provided that give the time at which right-of-way will be required and the duration of that right-of-way requirement.

Table 2.3-5 Terminator Descriptions (continued)

Entity Name	Entity Description
Other CVAS	This terminator is intended to provide a source and destination for ITS data flows between peer (e.g. inter-regional) commercial vehicle administration functions. It enables commercial vehicle administration activities to be coordinated across different jurisdictional areas. In the Physical Architecture, this terminator is a reciprocal Commercial Vehicle Administration Subsystem (CVAS).
Other EM	Representing other Emergency Management centers, systems or subsystems, this terminator provides a source and destination for ITS data flows between various communications centers operated by public safety agencies as well as centers operated by other allied agencies and private companies that participate in coordinated management of highway-related incidents. The interface represented by this terminator enables emergency management activities to be coordinated across jurisdictional boundaries and between functional areas. In the Physical Architecture this terminator is a reciprocal Emergency Management Subsystem (EM) implying the requirements for general networks connecting many allied agencies. The interface between this terminator and the EM supports coordination of incident management information between many different centers providing Public Safety Answering Point (both public or private sector implementations), Public Safety Dispatch, Emergency Operations, and other functions that participate in the detection, verification, response, and clearance of highway incidents. This terminator also supports interface to other allied agencies like utility companies that also participate in the coordinated response to selected highway-related incidents.
Other ISP	Representing other distinct Information Service Providers, this terminator is intended to provide a source and destination for ITS data flows between peer information and service provider functions. It enables cooperative information sharing between providers as conditions warrant. In the Physical Architecture this terminator is a reciprocal Information Service Provider (ISP) Subsystem.
Other TM	Representing another Traffic Management center, system or subsystem, this terminator is intended to provide a source and destination for ITS data flows between peer (e.g. inter-regional) traffic management functions. It enables traffic management activities to be coordinated across different jurisdictional areas. In the Physical Architecture this terminator is a reciprocal Traffic Management Subsystem (TMS).
Other TRM	Representing another Transit Management center, system or subsystem, this terminator is intended to provide a source and destination for ITS data flows between peer (e.g. inter-regional) transit management functions. It enables traffic management activities to be coordinated across geographic boundaries or different jurisdictional areas. In the Physical Architecture this terminator represents a reciprocal Transit Management Subsystem (TRMS).

Table 2.3-5 Terminator Descriptions (continued)

Entity Name	Entity Description
Other Vehicle	This terminator represents a vehicle (of any 4 vehicle types) that is neighboring the Basic Vehicle, where the Basic Vehicle is equipped to support vehicle-to-vehicle communication and coordination. These features are associated with advanced vehicle safety User Service implementations. These high-end vehicle control services may involve vehicles coordinating their activities.
Parking Operator	This terminator is the human entity that may be physically present at the parking lot facility to monitor the operational status of the facility.
Parking Service Provider	This terminator represents private or public organizations that provide parking lot facilities and that determine the parking lot pricing structures. This provider interfaces to the ITS to coordinate parking information, such as facility location, parking availability status, applicable rates, hours of operation, and other information (security services, valet services, shuttle availability, etc.).
Payment Instrument	This terminator represents the entity that enables the actual transfer of funds from the user of a service to the provider of the service. This terminator can be as abstract as an account number in the Logical Architecture, or as real as the electronic tag in the Physical architecture.
Pedestrians	This terminator provides input (e.g. a request for right of way at an intersection) from a specialized from of the Traveler, who is not using any type of vehicle (including bicycles) as a form of transport. Pedestrians may comprise those on foot and those in wheelchairs.
Potential Obstacles	Any object that possesses the potential of being sensed and struck and thus also possesses physical attributes. Potential Obstacles include roadside obstructions, other vehicles, pedestrians, infrastructure elements or any other element which is in a potential path of the vehicle. This external represents the physical obstacles which possess properties which enable detection using sensory functions included as part of the ITS architecture. These physical attributes are represented as a data input to the system.
Rail Operations	This is roughly the railroad equivalent to a highway Traffic Management Center. It is (usually) a centralized control point for a substantial segment of a railroad's operations. It is the source and destination of information that can be used to coordinate rail and highway traffic management. This terminator would also represent a railroad's management information system, if that system is the source or destination for this information. The use of a single terminator for multiple sources and destination for information exchange with the railroad entity is meant to imply the need for a single, consistent interface between a given railroad's operations and ITS traffic management. In any given implementation of ITS there may be multiple instantiations of this interface. For example, a city like Chicago may have interfaces to 5 or more Rail Operations Centers (e.g. BNSF, CSX, NS, UP. CR, etc.)

Table 2.3-5 Terminator Descriptions (continued)

Entity Name	Entity Description
Roadway	This terminator represents the physical conditions and geometry of the surface on which vehicles travel from an origin to a destination. Roadways can vary in type, such as surface streets, arterials, multi-lane highways, 2-lane rural roads, expressways, tollways, freeways, or any other vehicle travel surface. The condition of the roadway must be monitored by the architecture to enable corrective action and information dissemination regarding roadway conditions which may adversely affect travel. Roadways can also depict travel networks, such as surface street networks, arterial networks, or freeway networks. The roadway interface to the system carries the physical condition and geometry attributes which must be sensed, interpreted, and processed by functions internal to the system to achieve ITS User Service functionality.
Roadway Environment	This terminator represents the physical conditions surrounding the roadway itself. These may include emissions, fog, ice, snow, rain, etc. which will influence the way in which a vehicle can be safely operated on the roadway.
Secure Area Environment	This terminator comprises public access areas that transit users frequent during trips. Areas include bus stops, park and ride (PAR) facilities, at kiosks, and other transit transfer locations. These environments are monitored as part of the ITS Architecture functions to promote transit safety.
Toll Operator	The Toll Operator is the human entity that may be physically present at the toll plaza to monitor the operational status of the plaza.
Toll Service Provider	The Toll Service Provider represents organizations that operate private toll plazas that have provider determined toll pricing structures that are not necessarily under the control of the ITS Traffic Management function. However the ITS Traffic Management function determines the toll pricing structures of public toll plazas.
Traffic	The Traffic terminator represents the collective body of vehicles that travel on surface streets, arterials, highways, expressways, tollways, freeways, or any other vehicle travel surface. Traffic depicts the vehicle population from which traffic flow surveillance information is collected (average occupancy, average speed, total volume, average delay, etc.), and to which traffic control indicators are applied (intersection signals, stop signs, ramp meters, lane control barriers, variable speed limit indicators, etc.). All sensory and control elements that interface to this vehicle population are internal to ITS.
Traffic Operations Personnel	This terminator represents the human entity that directly interfaces with vehicle traffic operations. These personnel interact with traffic control systems, traffic surveillance systems, incident management systems, work zone management systems, and travel demand management systems to accomplish ITS services. They provide operator data and command inputs to direct systems' operations to varying degrees depending on the type of system and the deployment scenario. All functionality associated with these services that might be automated in the course of ITS deployment is modeled as internal to the architecture.

Table 2.3-5 Terminator Descriptions (continued)

Entity Name	Entity Description
Transit Driver	This terminator represents the human entity that is a special form of the Driver terminator that receives and provides additional information that is specific to Transit (including demand responsive transit) operations. This information will not be received by other types of Driver. The Transit Driver terminator operates the Transit Vehicle terminator and represents random route drivers, flexible fixed route drivers and fixed route drivers. The fixed route drivers require minimal information such as run times and passenger loading. The flex fixed and random route drivers require additional information such as dynamically changing routes.
Transit Fleet Manager	This terminator represents the human entity that is responsible for planning the operation of transit fleets, including monitoring and controlling the transit fleet route schedules and the transit fleet maintenance schedules. This comprises planning routes and schedules for either daily use or for special occasions as distinct from making day to day variations to schedules and routes.
Transit Maintenance Personnel	The terminator represents the human entity that is actively responsible for monitoring, controlling, and planning the schedules for the maintenance of transit fleets.
Transit System Operators	This terminator represents the human entities that are responsible for all aspects of the Transit subsystem operation including planning and management. They actively monitor, control, and modify the transit fleet routes and schedules on a day to day basis. The modifications will be to take account of abnormal situations such as vehicle breakdown, vehicle delay, etc. These personnel may also be responsible for demand responsive transit operation and for managing emergency situations within the transit network.
Transit User	This terminator represents the human entities using Public Transit vehicles. They may be in the act of embarking or debarking the vehicles and are thus sensed for the purpose of determining passenger loading and fares, or on the vehicles and able to request and receive information.
Transit Vehicle	This terminator represents a specialized form of the Basic Vehicle used by transit service providers. It supports equipment to collect fares, monitor activities, request priority at signals, and provide information to travelers. It may be a bus, LRT vehicle, or other vehicle specially designed for the carriage of passengers, such as those used by demand responsive transit operators. The monitoring of the Transit Vehicle mechanical condition and mileage provides the major inputs for transit vehicle maintenance scheduling.
Transportation Planners	This terminator represents the human entities responsible for planning, maintaining and changing the transportation network managed by the ITS. It includes organizations such as MPOs who are responsible for long term planning. These organizations will benefit from data collected by ITS subsystems. They may have a direct impact on the road network but can only monitor and request changes to other areas of transportation such as transit operations, toll prices and parking lot charges, capacities, etc.

Table 2.3-5 Terminator Descriptions (continued)

Entity Name	Entity Description
Traveler	This terminator represents any individual (human) who uses transportation services. At the time that data is passed to or from the terminator the individual is neither a driver, pedestrian, or transit user. This means that the data provided is that for pre-trip planning or multi-modal personal guidance and includes their requests for assistance in an emergency. Subsequent to receipt of pre-trip information, a Traveler may become a vehicle driver, passenger, transit user, or pedestrian.
Vehicle Characteristics	This terminator represents the external view of an individual vehicle. It includes vehicle characteristics such as height, width, length, weight, and other properties (e.g., magnetic properties, number of axles) that allow an individual vehicle to be detected and measured or classified. This external view of an individual vehicle is also used as a source of visible data that supports individual vehicle imaging requirements in the architecture. ITS subsystems at the roadside sense these characteristics and generate ITS data flows. These individual vehicle characteristics are important for toll collection, parking management, and other applications that identify and measure individual vehicles. See also the related "Traffic" terminator which represents physical characteristics of many vehicles in the aggregate that is measured for general traffic applications.
Wayside Equipment	This terminator represents train interface equipment (usually) maintained and operated by the railroad and (usually) physically located at or near a grade crossing. This terminator is the source and destination for HRI information for, or about, approaching trains and their crews (e.g. the time at which the train will arrive and the time it will take to clear a crossing, crossing status or warnings, etc.). Generally one wayside equipment interface would be associated with one highway rail intersection. However, multiple crossings may be controlled using information based on data from one wayside equipment interface.
Weather Service	An external source of current and forecast weather conditions. This externally derived weather data is integrated with the other information collected and disseminated by the IVHS architecture to support travel planning.
Yellow Pages Service Providers	This terminator represents organizations that provide any service oriented towards the Traveler. Example services that could be included are gas, food, lodging, vehicle repair, points of interest, and recreation areas. The interface with the Service Provider is necessary so that accurate, up-to-date service information can be provided to the traveler and to support electronic reservation capabilities included in the ITS User Services.

Entities are connected in the physical architecture flow diagram with data flows. Table 2.3-6 contains the descriptions of all physical architecture flows. More detailed definition of the message sizes and contents can be obtained from the appropriate logical data dictionary entries. Detailed timing information is implementation dependent and is provided in the evaluatory design.

Also included are architecture flow diagrams containing the subsystems associated with each set of stakeholder areas. Each of these diagrams is further broken down in subsequent sections.

Table 2.3-6 Architecture Flow Descriptions

Architecture Flow	*Flow Description*
activity reports	Activity reports containing records of citations, accidents, inspections, etc.
aggregate travel data	Aggregated transportation infrastructure data and associated traveler transaction data for planning purposes.
AHS control data	Information required for vehicles to operate on AHS lanes.
AHS control information	Control data to AHS roadway equipment
AHS status	Status of AHS equipment, lane controls etc.
AHS vehicle data	AHS route and vehicle condition data
alerts, messages	Specific alerts and messages related to Commercial Vehicles (e.g. trucks not advised, trucks over 10 tons not allowed on bridge, route details)
arriving train information	Information for a train approaching a highway-rail intersection that may include direction and allow calculation of approximate arrival time and closure duration.
bad tag list	List of invalid transit user tags which may have previously failed a fare payment transaction.
border clearance event record	Results of border clearance check.
broadcast advisories	General broadcast advisories that are provided over wide-area wireless communications direct to the vehicle radio. These analog advisory messages may provide similar content to ITS broadcast information flows, but include no digital data component. Existing Highway-Advisory Radio (HAR) advisory messages are a prime example of this flow.
broadcast information	General broadcast information that contains link travel times, incidents, advisories, transit services and a myriad of other traveler information.
citation data	Safety problems related to the carrier, driver and vehcile that may lead to a citation.
clearance event record	Results of vehicle clearance activity.
closure coordination	Coordination between subsystems regarding construction and maintenance closure times and durations.
commercial vehicle data	Information about the commercial vehicles cargo, credentials, and payments.
commercial vehicle data request	Request for commercial vehicle information (cargo, driver's credit, vehicle location).
compliance review report	Report containing data from facility activity logs from various roadside facilities.
credential application	Application for commercial vehicle credentials for a particular route/trip.
credentials and safety information request	Request for additional credentials and safety information.
credentials and safety information response	Instructions to commercial vehicle managing and/or information systems indicating which vehicles are to be allowed to pass and which are out of service or have not been credentialed.
credentials information	Response containing credentials information.

Table 2.3-6 Architecture Flow Descriptions (continued)

Architecture Flow	*Flow Description*
credentials information request	Request for credential information.
crossing call	Request for pedestrian crossing.
crossing permission	Signal to pedestrians indicating permission to cross roadway.
current network conditions	Current traffic information, road conditions, and camera images that can be used to locate and verify reported incidents, and plan and implement an appropriate response.
CVAS information exchange	Tax and credential fee information exchanged between cooperating commercial vehicle administration offices (e.g. regional or inter-state preclearance data).
CVC override mode	Manual override by the commercial vehicle roadside facility inspector of automated pass/pull-in signage information.
CVO database update	Credential information and safety problem list updates.
CVO driver initialization	Commercial vehicle driver and vehicle information and requests to the commercial vehicle managing system.
CVO inspector information	Credential, safety, and preclearance information and instructions to the commercial vehicle inspector.
CVO inspector input	Requests from the commercial vehicle inspector to operate the commercial vehicle inspection station.
CVO Pull in Message	Message sent to commercial vehicle driver requesting pull in to inspection/verification stop along with inspection results.
CVO weight and presence	Weigh-In-Motion message to indicate presence of commercial vehicle and its weight.
demand management request	Request to change the demand for road facility use through pricing or other mechanisms.
demand management response	Response to various demand management change requests indicating level of compliance with request.
demand responsive transit plan	Plan regarding overall demand responsive transit schedules and deployment.
demand responsive transit request	Request for paratransit support.
dispatch information	Dispatch information and command to emergency personnel.
driver and vehicle information	Requests from the driver and vehicle for routing, payment, and enrollment information.
driver information	General advisory and traffic control information provided to the driver while en-route.
driver inputs	Driver commands to the vehicle.
driver instructions	Transit service instructions for both transit and paratransit drivers.
driver updates	Information displayed or otherwise conveyed by the vehicle to the driver.
electronic clearance data	Information required for electronic clearance (toll, safety, customs, etc.).
electronic clearance request	Request for electronic clearance data (Toll, safety, customs, etc.).
electronic credentials	Authenticated credentials including route enrollment and payment confirmation.

Table 2.3-6 Architecture Flow Descriptions (continued)

Architecture Flow	*Flow Description*
emergency acknowledge	Acknowledge request for emergency assistance and provide additional details regarding actions and verification requirements.
emergency dispatch requests	Emergency vehicle dispatch instructions including incident location and available information concerning the incident.
emergency dispatch response	Request for additional emergency dispatch information (e.g., a suggested route) and provision of en-route status.
emergency notification	An emergency request for assistance originated by a traveler using an in-vehicle, public access, or personal device. Sufficient information is provided so that the recipient can determine the location of the emergency as a minimum. Additional information identifying the requestor and requesting device and the nature and severity of the emergency may also be provided (and required) by some systems.
emergency operations request	Emergency operator inputs supporting call taking, dispatch, and other operations and communications center operator functions.
emergency operations status	Emergency operations data supporting a range of emergency operating positions including call taker, dispatch, and various other operations and communications center operator positions.
emergency personnel inputs	Current incident status information and requests from emergency personnel in the field for information and/or resources.
emergency traffic control request	Special request to preempt the current traffic control strategy in effect at one or more signalized intersections or highway segments. For example, this flow can request all signals to red-flash, request a green wave for an emergency vehicle, or request another special traffic control plan.
emergency traffic control response	Status of the green wave or other special traffic signal control strategy implemented in response to the emergency traffic control request.
emergency vehicle route	Routing for emergency vehicle including greenwave paths.
emergency vehicle route request	Special routing instructions and signal priority for emergency vehicles.
emergency vehicle tracking data	The current location and operating status of the emergency vehicle.
emissions data	Emissions data and associated imagery collected by roadside equipment.
environmental conditions	Current environment conditions (e.g., air temperature, wind speed, surface temperature) as measured by environmental sensors and communicated by supporting field equipment.
equipment maintenance status	Current status of field equipment maintenance actions.
event confirmation	Confirmation that special event details have been received and processed.
event plans	Plans for major events possibly impacting traffic.
external reports	Traffic and incident information that is collected by the media through a variety of mechanisms (e.g., radio station call-in programs, air surveillance).
fare and payment status	Current fare collection information including the operational status of the fare collection equipment and financial payment transaction data.
fare and price information	Current transit, parking, and toll fee schedule information.

Table 2.3-6 Architecture Flow Descriptions (continued)

Architecture Flow	*Flow Description*
fare management information	Transit fare information and transaction data used to manage transit fare processing on the transit vehicle.
fault reports	Reports from field equipment (sensors, signals, signs, controllers, etc.) which indicate current operational status.
fleet manager inquiry	Inquiry from fleet manager requesting data from commercial vehicle management system.
fleet status	Fleet status information including enrollment status, routing information, current vehicle information, and emergency information.
fleet to driver update	Updated instructions to the driver including dispatch, routing, and special instructions
freeway control data	Control commands and operating parameters for ramp meters, dynamic message signs, mainline metering/lane controls and other systems associated with freeway operations.
freeway control status	Current operational status and operating parameters for ramp meters, dynamic message signs, mainline metering/lane controls and other control equipment associated with freeway operations.
Hazmat information	Information about a particular Hazmat load including nature of the load and unloading instructions. May also include Hazmat vehicle route and route update information
Hazmat information request	Request for information about a particular Hazmat load.
highway control status	Current traffic control equipment status that indicates operational status and right-of-way availability to the non-highway transportation mode at a multimodal crossing.
hov data	Current HOV lane information including both standard traffic flow measures and information regarding vehicle occupancy in HOV lanes.
hri advisories	Notification of Highway-Rail Intersection equipment failure, intersection blockage, or other condition requiring attention, and maintenance actives at or near highway rail intersections.
hri control data	Data required for HRI information transmitted at railroad grade crossings and within railroad operations.
hri request	A request for highway-rail intersection status or a specific control request intended to modify HRI operation.
hri status	Status of the highway-rail intersection equipment including both the current state or mode of operation and the current equipment condition.
incident command information	Information that supports local management of an incident. It includes resource deployment status, hazardous material information, traffic, road, and weather conditions, evacuation advice, and other information that enables emergency personnel in the field to implement an effective, safe incident response.
incident command information presentation	Presentation of information to emergency personnel in the field that supports local tactical decision-making within an incident command system structure.
incident command request	Request for resources, commands for relay to other allied response agencies, and other requests that reflect local command of an evolving incident response.

Table 2.3-6 Architecture Flow Descriptions (continued)

Architecture Flow	Flow Description
incident data	Data and imagery from the roadside supporting incident detection and verification.
incident information	Notification of existence of incident and expected severity, location, time and nature of incident.
incident information for media	Report of current desensitized incident information prepared for public dissemination through the media.
incident information request	Request for incident information, clearing time, severity. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.
incident notification	The notification of an incident including its nature, severity, and location.
incident notification response	Interactive acknowledgement and verification of the incident information received, requests for additional information, and general information on incident response status.
incident report	Report of an identified incident including incident location, type, severity and other information necessary to initiate an appropriate incident response.
incident response coordination	Incident response procedures, resource coordination, and current incident response status that are shared between allied response agencies to support a coordinated response to incidents. This flow also coordinates a positive hand off of responsibility for all or part of an incident response between agencies.
incident response status	Status of the current incident response including traffic management strategies implemented at the site (e.g., closures, diversions, traffic signal control overrides).
incident status	Information gathered at the incident site that more completely characterizes the incident and provides current incident response status.
information on violators	Response from law enforcement agency to violations notification request.
information request	General purpose information request for data stored within the commercial vehicle operations information exchange network.
intermod CVO coord	Cargo movement logs and cargo ID’s exchanged between freight shippers.
intermodal information	Schedule information for alternate mode transportation providers such as train, ferry, air and bus.
international border crossing data	Cleared commercial vehicle data to allow pass-thru international border crossings.
international border crossing data update	Update from commercial vehcile check stations of international border crossing events.
intersection blockage notification	Notification that a highway-rail intersection is obstructed and supporting information.
intersection status	Status of intersection congestion, approaching vehicles, etc.
ISP coordination	Coordination and exchange of transportation information between centers. This flow allows a broad range of transportation information collected by one ISP to be redistributed to many other ISPs and their clients.

Table 2.3-6 Architecture Flow Descriptions (continued)

Architecture Flow	*Flow Description*
ISP operating parameter updates	Tuning and performance enhancement parameters to ISP algorithms
ISP operating parameters	Parameters provided to the ISP Operator by the ISP including broadcast information settings, route selection controls, and travel optimization algorithms.
license request	Request supporting registration data based on license plate read during violation.
local signal preemption request	Direct control signal or message to a signalized intersection that results in preemption of the current control plan and grants right-of-way to the requesting vehicle.
local signal priority request	Request from a vehicle to a signalized intersection for priority at that intersection.
lock tag data	Tag information on cargo lock.
lock tag data request	Request to supply lock information on cargo lock for retransmission to international border crossing station.
log information	Request information to be entered into the driver log.
logged special vehicle route	Anticipated route information for special vehicles (e.g., oversize vehicles) or groups of vehicles (e.g., governer's motorcade) that may require changes in traffic control strategy.
maintenance resource request	Request for road maintenance resources that can be used in the diversion of traffic (cones, portable signs), clearance of an incident, and repair of ancillary damage.
maintenance resource response	Current status of maintenance resources included availability and deployment status.
maintenance status	Current maintenance status of vehicle.
map update request	Request for a map update which could include a new underlying map or map layer updates.
map updates	Map update which could include a new underlying static or real-time map or map layer(s) update.
media information request	Request from the media for current transportation information.
multimodal crossing status	Indication of operational status and pending requests for right-of-way from equipment supporting the non-highway mode at multimodal crossings.
on board safety data	Vehicle safety data measured by vehicle sensors and sent to inspection stations
on board vehicle data	Condition of the commercial vehicle sent to commercial vehicle manager primarily for maintenance purposes.
on-board safety request	Request for onboard vehicle safety data.
operational data	Statistical data used for planning purposes.
parking availability	Parking lot occupancy, availability and payment information.
parking instructions	Instructions from traffic manager or parking operator regarding operation strategy of a parking facility.
parking lot data request	Request for parking lot occupancy, fares, and availability. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Table 2.3-6 Architecture Flow Descriptions (continued)

Architecture Flow	*Flow Description*
parking lot reservation confirmation	Confirmation for parking lot reservation.
parking reservations request	Reservation request for parking lot.
parking status	Parking lot operational status.
pass/pull-in	Command to commercial vehicle to pull into inspection station.
payment	Payment of some kind (e.g., toll, parking, fare) by traveler which in most cases can be related to a credit account.
payment request	Request for payment from financial institution.
personal transit information	General and personalized transit information for a particular fixed route, flexible route, or paratransit system.
physical presence	Detection of an obstacle by a vehicle. Obstacle could include animals, other vehicles, pedestrians, rocks in roadway etc.
planning data	Data and commands from Transportation Planners.
pollution data	Measured emissions data comprised of various atmospheric pollutants.
pollution data display	Both reference and current pollution status details for a given geographic area.
pollution data parameters	Nominal pollution data compliance (reference) levels for each sector of an urban area.
pollution state data request	Aggregated emissions data information request.
position fix	Information which provides a traveler or vehicles geographical position.
probe data	Aggregate data from probe vehicles including location, speed for a given link or collection of links.
provider profile confirm	Confirmation of profile information received by a service provider (e.g. for a hotel or restaurant).
provider profile data	Information supplied by a service provider (e.g., a hotel or restaurant) that identifies the service provider and provides details of the service offering. This flow covers initial registration of a service provider and subsequent submittal of new information and status updates so that data currency is maintained.
railroad advisories	Real-time notification of railway-related incident or advisory.
railroad schedules	Train schedules, maintenance schedules, and other information from the railroad that supports forecast of HRI closures.
registration	Registered owner of vehicle and associated vehicle information.
regulations	Regulations imposed on Commercial Vehicle Administration agencies including safety ratings, facility locations and credential fee structure.
remote surveillance control	The control commands used to remotely operate another center's sensors or surveillance equipment so that roadside surveillance assets can be shared by more than one agency.
request fare and price information	Requests for current fare and price information from a service provider that can be used to augment the traffic manager's overall view of current transportation network status.
request for bad tag list	Request for list of bad vehicle tag Ids.

Table 2.3-6 Architecture Flow Descriptions (continued)

Architecture Flow	*Flow Description*
request for information on violators	Request for law enforcement information on vehicles and drivers suspected of violations.
request for payment	Request to deduct cost of service from user's payment account.
request for performance data	Request issued by a service provider for current parking service performance data.
request for right-of-way	Forwarded request from signal prioritization, signal preemption, pedestrian call, multi-modal crossing activation, or other source for right-of-way.
request for service	A traveler service request initiated by a driver or traveler. The request may result in a financial transaction, summon an emergency response, or initiate another service at the behest of the driver.
request for traffic information	Request for traffic information that specifies the region/route of interest, the desired effective time period, and other parameters that allow preparation of a tailored response. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.
request for vehicle measures	Request for vehicle performance and maintenance data collected by onboard sensors.
request tag data	Request for tag information including credit identity, stored value card cash, etc.
request transit information	Request for transit service information and current transit status.
resource deployment status	Status of traffic management center resource deployment identifying the resources available and their current deployment status.
resource request	A request for traffic management resources to implement special traffic control measures, assist in clean up, verify an incident, etc.
reversible lane status	Current reversible lane status including traffic sensor and surveillance data and the operational status and mode of the reversible lane control equipment.
road network use	Aggregated route usage and associated travel data from clients for planning and analysis.
roadside log update	Update of activities at commercial vehicle check stations including clearance events and inspection reports.
roadway characteristics	Detectable or measurable road characteristics such as friction coefficient and general surface conditions, road geometry and markings, etc. These characteristics are monitored or measured by vehicle ITS components and used to support advanced vehicle safety and control capabilities.
roadway information system data	Information used to initialize, configure, and control roadside systems that provide driver information (e.g., dynamic message signs, highway advisory radio, beacon systems). This flow can provide message content and delivery attributes, local message store maintenance requests, control mode commands, status queries, and all other commands and associated parameters that support remote management of these systems.
roadway information system status	Current operating status of dynamic message signs, highway advisory radios, beacon systems, or other configurable field equipment that provides dynamic information to the driver.

Table 2.3-6 Architecture Flow Descriptions (continued)

Architecture Flow	*Flow Description*
route assignment	Route assignment information for transit driver.
route plan	Tailored route provided by ISP in response to a specific request.
route request	Request for a tailored route based on given constraints.
safety information	Response containing commercial vehicle safety information.
safety information request	Request for commercial vehicle safety information.
safety inspection record	Record containing results of commercial vehicle safety inspection.
screening data	Data stored in vehicle's tag allowing electronic clearance at border crossings, debits at toll plazas, and clearance at safety inspections.
screening request	Request for screening data based on vehicle and possibly cargo's tags.
secure area characteristics	Characteristics (visual, audible, other) that are monitored by surveillance security systems via sensors.
secure area monitoring support	Commands that control surveillance equipment and security sensors that monitor secure public transportation areas. Also includes information for general advisories and alerts intended for general dissemination in these same public areas.
secure area surveillance data	Data collected from surveillance systems used to monitor secure areas. Includes video, audio, and other security sensor outputs.
selected routes	Routes selected based on route request criteria.
sensor and surveillance control	Information used to configure and control sensor and surveillance systems at the roadside.
signal control data	Information used to configure and control traffic signal control systems.
signal control status	Status of surface street signal controls.
suggested route	Suggested route for a dispatched emergency vehicle that may reflect current network conditions and the additional routing options available to en-route emergency vehicles that are not available to the general public.
tag data	Unique tag ID and related vehicle information for the purposes of payment for services.
tag update	Update data held in tag which can be read at another screening.
tax filing, audit data	Commercial vehcile tax filing and audit data.
tax-credentials-fees request	Request to government agency for tax, credential and/or fee data.
toll data	Current toll schedules for different types of vehicles as well as advanced toll payment information.
toll data request	Request made to obtain toll schedule information or pay a toll in advance. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.
toll fees	Instructions indicating toll fees which should be charged.
toll instructions	Demand management toll pricing information based on current congestion.
toll operator requests	Request for information from toll operator at toll collection site.
toll revenues and summary reports	Summary of toll revenues and toll-related reports to toll service provider.
toll transaction reports	Summary report sent to toll collection point operator containing previous toll transactions.
Toll Transactions	Detailed list of transactions from a toll station.

Table 2.3-6 Architecture Flow Descriptions (continued)

Architecture Flow	Flow Description
track status	Current status of the wayside equipment and notification of an arriving train.
traffic characteristics	Physical traffic characteristics which are monitored and translated into macroscopic measures like occupancy, volume, density, and average speed. Point measures support presence detection and individual vehicle measures like speed.
traffic control coordination	Information transfers that enable remote monitoring and control of traffic management devices. This flow is intended to allow cooperative access to, and control of, field equipment during incidents and special events and during day-to-day operations. This flow also allows 24-hour centers to monitor and control assets of other centers during off-hours, allows system redundancies and fail-over capabilities to be established, and otherwise enables integrated traffic control strategies in a region.
traffic control priority request	Request for signal priority at one or more intersections along a particular route.
traffic control priority status	Status of signal priority request functions at the roadside (e.g. enabled or disabled).
traffic equipment status	Identification of field equipment requiring repair and known information about the associated faults.
traffic flow	Raw and/or processed traffic detector information which allows derivation of traffic flow, speed, and density measures.
traffic images	High fidelity, real-time traffic images suitable for surveillance monitoring by the operator or for use in machine vision applications.
traffic information	Current and predicted traffic information, road and weather conditions, incident information, and pricing data. Either raw data, processed data, or some combination of both may be provided by this architecture flow.
traffic information coordination	Traffic information exchanged between TMC's. Normally would include incidents, congestion data, traffic data, signal timing plans, and real-time signal control information.
traffic information for media	Report of current traffic conditions, incidents, maintenance activities and other traffic-related information prepared for public dissemination through the media.
traffic information for transit	Current and predicted traffic information and incident information.
traffic operations data	Presentation of traffic operations data to the operator including traffic conditions, current operating status of traffic control equipment, maintenance activity status, incident status, and other information. This data keeps the operator appraised of current road network status, provides feedback to the operator as traffic control actions are implemented, and supports review of historical data and preparation for future traffic operations activities.
traffic operations requests	Traffic operations requests for information, commands to adjust current traffic control strategies (e.g., adjust signal timing plans, change VMS messages).
transaction status	Response to transaction request. Normally dealing with a request for payment.

Table 2.3-6 Architecture Flow Descriptions (continued)

Architecture Flow	*Flow Description*
transit and fare schedules	Specific transit and fare schedule information including schedule adherence.
transit driver availability	Transit driver availability data that can be used to develop driver assignments and detailed operations schedules.
transit driver display	Display (either video or audio) to transit driver containing status of various ITS services.
transit driver inputs	Transit driver emergency request as well as fare transaction data.
transit emergency coordination data	Data exchanged between centers dealing with a transit-related incident.
transit emergency data	Initial notification of transit emergency at a transit stop or on transit vehicles and further coordination as additional details become available and the response is coordinated.
transit fare payment requests	Information provided from the transit user location that supports fare payments and associated record keeping.
transit fare payment responses	Information provided by transit management that supports a fare payment transaction
transit fleet manager inputs	Instructions governing service availability, schedules, emergency response plans, transit personnel assignments, transit maintenance requirements, and other inputs that establish general system operating requirements and procedures.
transit incident information	Information on transit incidents that impact transit services for public dissemination.
transit incidents for media	Report of an incident impacting transit operations for public dissemination through the media.
transit information for media	Report of transit schedule deviations for public dissemination through the media.
transit information request	Request for transit operations information including schedule and fare information. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.
transit information user request	Request for special transit routing, real-time schedule information, and availability information.
transit operations planning data	Accumulated schedule and fare information, emergency response plans, transit personnel information, maintenance records, and other information intended to support overall planning and management of a transit property.
transit operator display	Display for transit operations personnel regarding performance of the transit fleet, current ridership and on-time performance.
transit operator management data	Information and control provided by transit system operators involving many aspects of managing transit operations.
transit parking coordination	Request for coordinated fare payment and parking lot price data.
transit parking lot response	Response to transit occupancy inquiries and coordination with parking lots.
transit request confirmation	Confirmation of a request for transit information or service.

Table 2.3-6 Architecture Flow Descriptions (continued)

Architecture Flow	*Flow Description*
transit schedule information	Current and projected transit schedule adherence.
transit system data	Current transit system operations information indicating current transit routes, the level of service on each route, and the progress of individual vehicles along their routes for use in forecasting demand and estimating current transportation network performance.
transit traveler information	Transit information prepared to support transit users and other travelers. It contains transit schedules, real-time arrival information, fare schedules, and general transit service information.
transit user fare status	Status of fare transaction for transit user.
transit user inputs	Requests from transit user through either an on-board or fixed location traveler information station.
transit user outputs	Information for traveler from either an on-board or fixed location traveler information station.
transit vehicle conditions	Operating conditions of transit vehicle (e.g., mileage).
transit vehicle location data	Current transit vehicle location and related operational conditions data provided by a transit vehicle.
transit vehicle passenger and use data	Data collected on board the transit vehicle pertaining to availability and/or passenger count.
transit vehicle schedule performance	Estimated times of arrival and anticipated schedule deviations reported by a transit vehicle.
transit work schedule	Orders for maintenance of transit vehicle or other transit system equipment.
travel service info	Reservation information or yellow pages data.
travel service request	Request for reservation or other service (e.g., yellow pages).
traveler advisory request	In vehicle communication between transit and vehicle systems includes advisories and advance payment deductions.
traveler information	Traveler information comprised of traffic status, advisories, incidents, responses to traveler requests (e.g., traveler routing, yellow pages), payment information and many other travel-related data updates and confirmations.
traveler information for media	General traveler information regarding incidents, unusual traffic conditions, transit issues, or other advisory information that has been desensitized and provided to the media.
traveler interface updates	Visual or audio information (e.g., routes, messages, guidance) to the traveler.
traveler profile	Information about a traveler including equipment capabilities, personal preferences and recurring trip characteristics.
traveler request	Request by a traveler to summon assistance, request information, make a reservation, or initiate any other traveler service.
trip confirmation	Acknowledgement by the driver/traveler of acceptance of a route.
trip plan	A sequence of links and special instructions comprising a trip plan indicating efficient routes for navigating the links. Normally coordinated with traffic conditions, other incidents, preemption and prioritization plans.
trip request	Request by a driver/traveler for special routing.

Table 2.3-6 Architecture Flow Descriptions (continued)

Architecture Flow	*Flow Description*
TRMS coord	Coordination information between local/regional transit organizations including schedule, on-time information and ridership.
vehicle characteristics	The physical or visible characteristics of an individual vehicle that can be measured to classify a vehicle and imaged to uniquely identify a vehicle.
vehicle control	Vehicular control commands
vehicle location	Location of vehicle and other vehcile characteristics which are exchanged between vehicle subsystems.
vehicle measures	Sensing information from vehicle sensors.
vehicle pollution criteria	Vehicular pollution acceptance criteria.
vehicle probe data	Vehicle probe data indicating identity, route segment identity, link time and location.
vehicle signage data	In-vehicle signage data generated by the roadway infrastructure indicating either road conditions, street names, or special information which will be useful for a vehicle passing a specific point on the roadway.
vehicle to vehicle coordination	Any type of advanced vehicle to vehicle communication.
violation notification	Notification to enforcement agency of violation or regulations.
weather conditions	Collected weather condition data from sensors.
weather information	Accumulated predicted and current weather data (e.g., temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.).
wide area statistical pollution information	Aggregated region-wide measured emissions data and possible pollution incident information.
work zone status	Status of maintenance work zone.
yellow pages request	Request for information through a yellow pages type service.

Figure 2.3-1 through Figure 2.3-9 contain top level views of the architecture for some combinations of subsystems. Later sections describe each subsystem and its interfaces in detail.

.

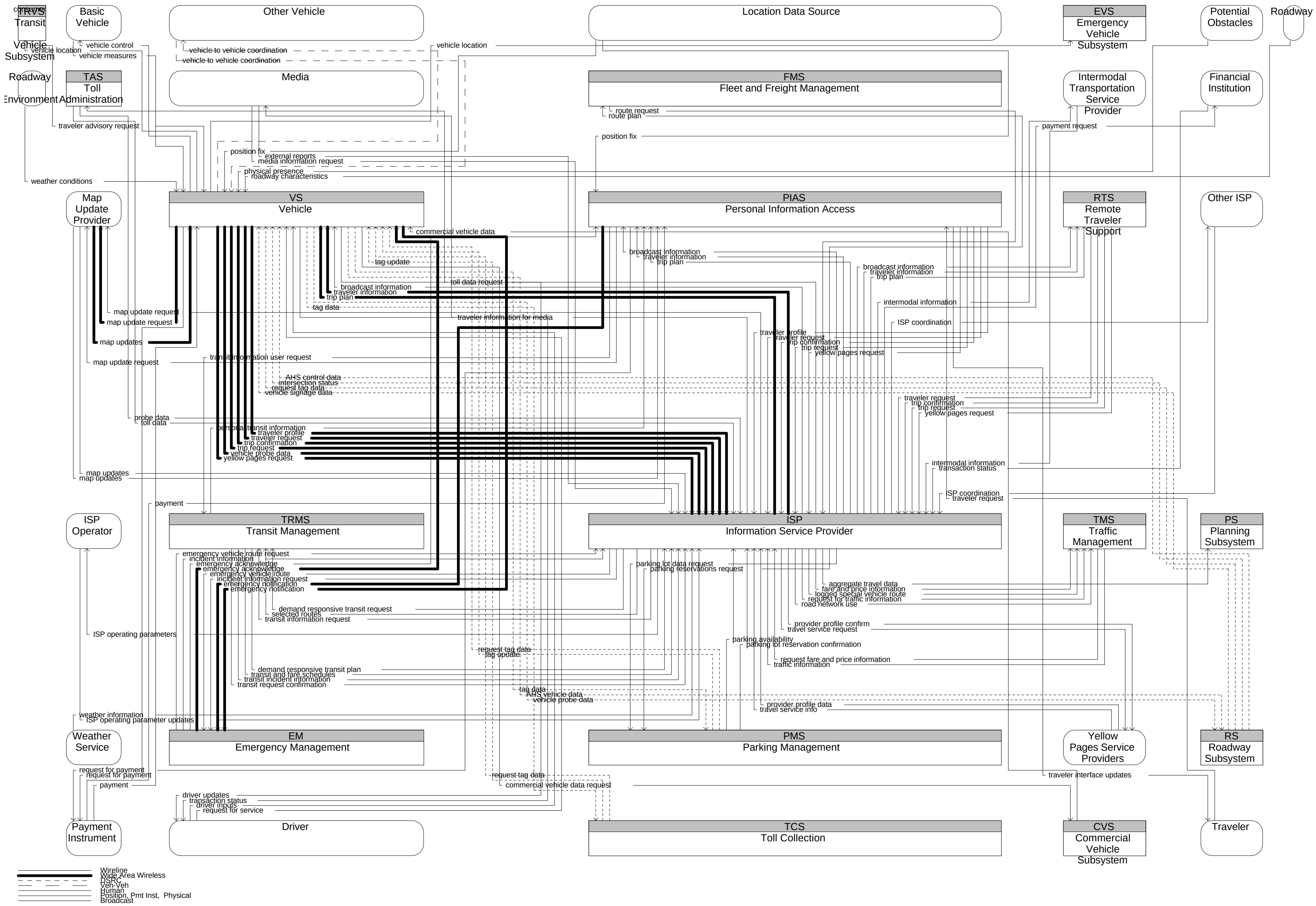


Figure 2.3-1 Architecture Flow Diagram for the Vehicle, Personal Information, Remote Traveler Access, and ISP

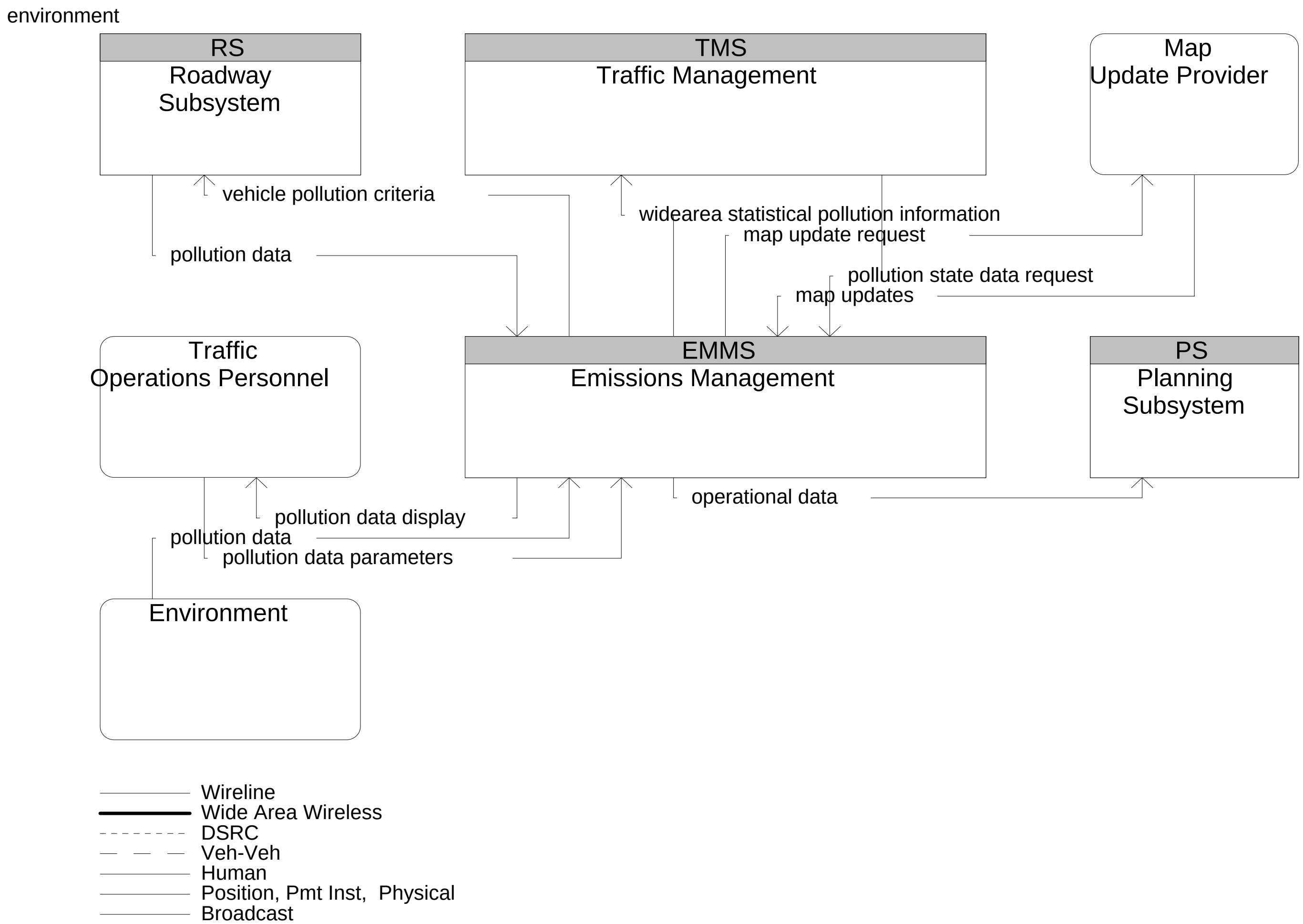


Figure 2.3-2 Architecture Flow Diagram for Emissions Management

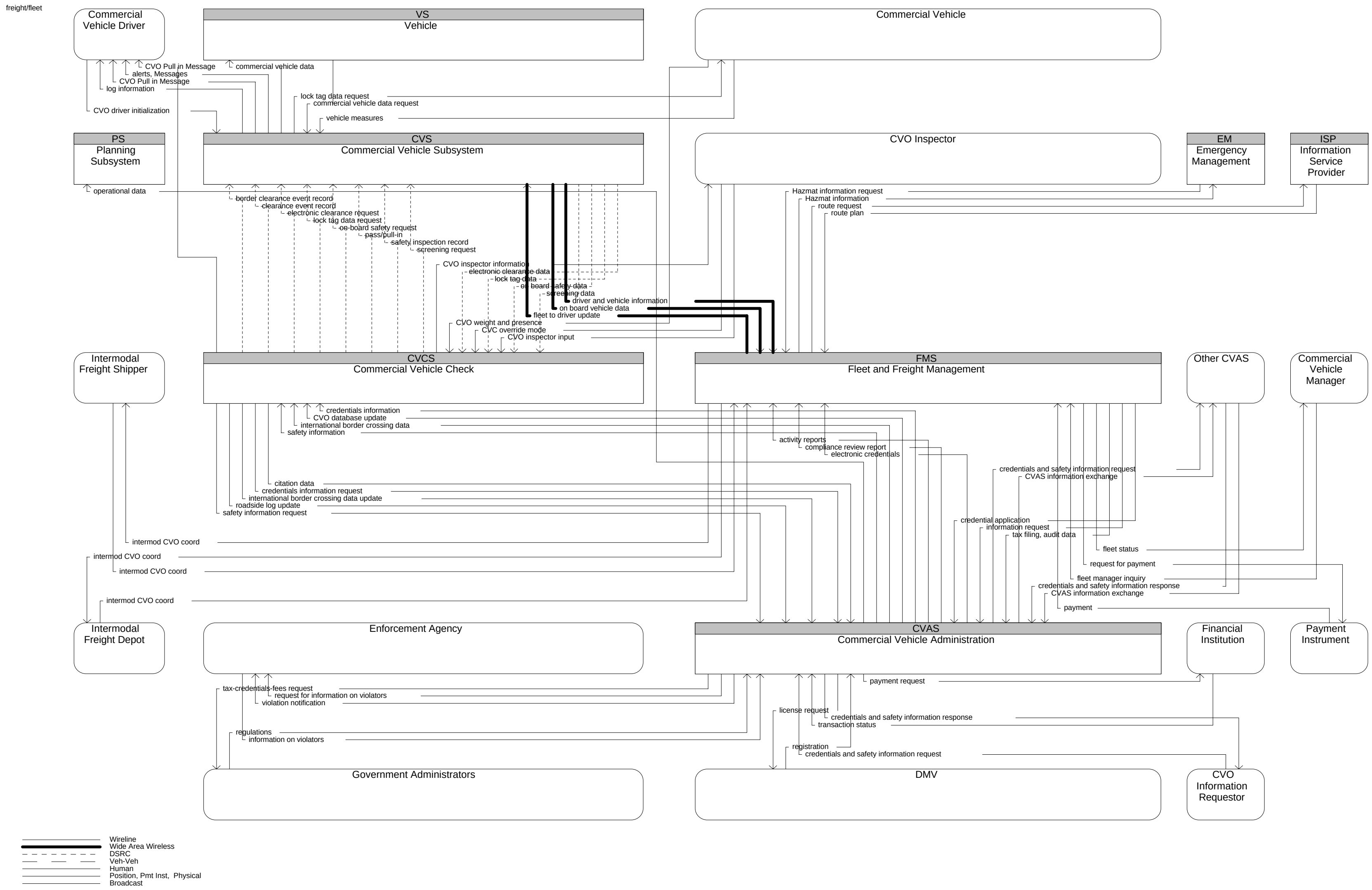


Figure 2.3-3 Architecture Flow Diagram for Commercial Admin, Check, Vehicle and Fleet and Freight Management

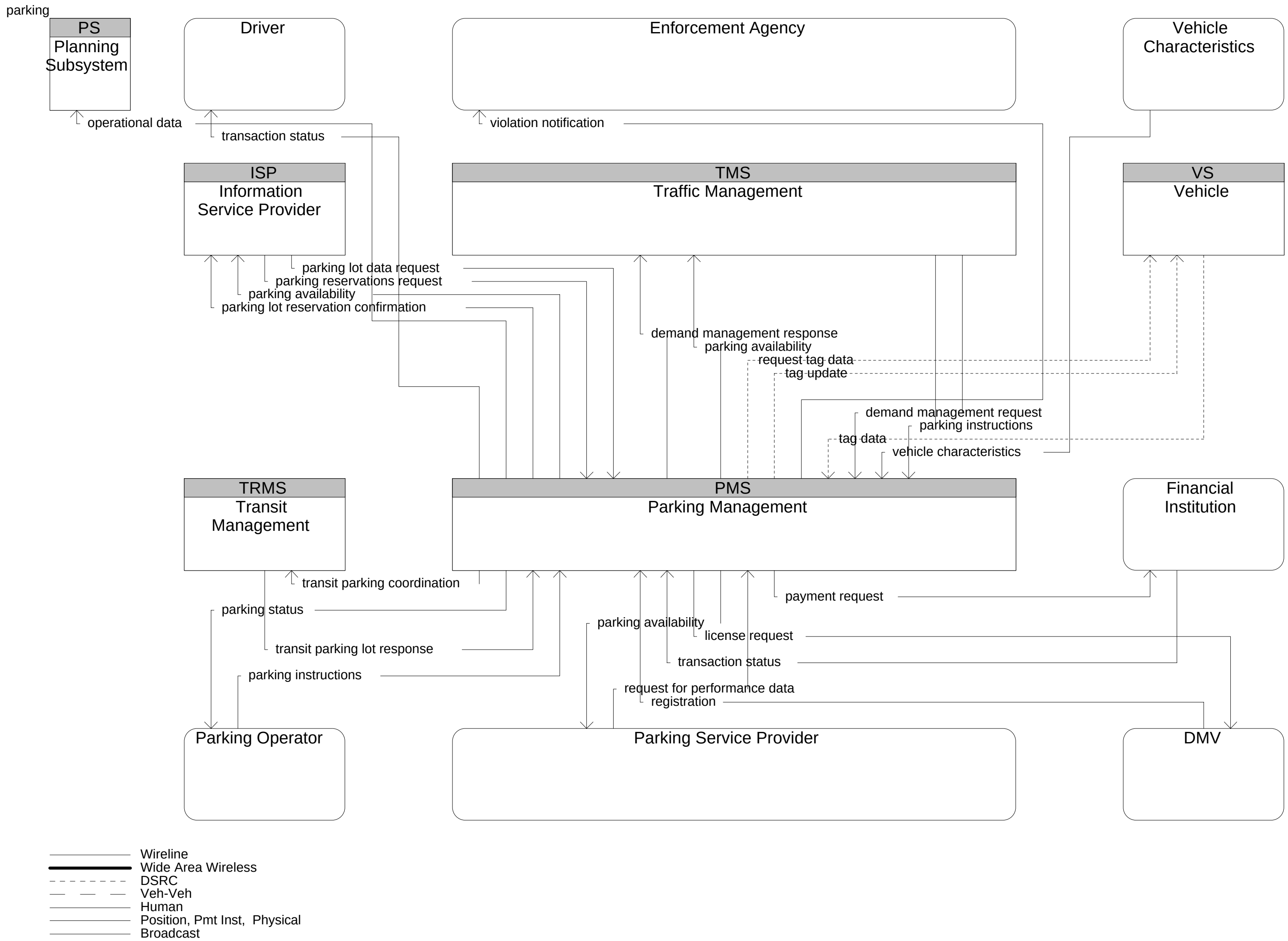


Figure 2.3-4 Architecture Flow Diagram for Parking Management

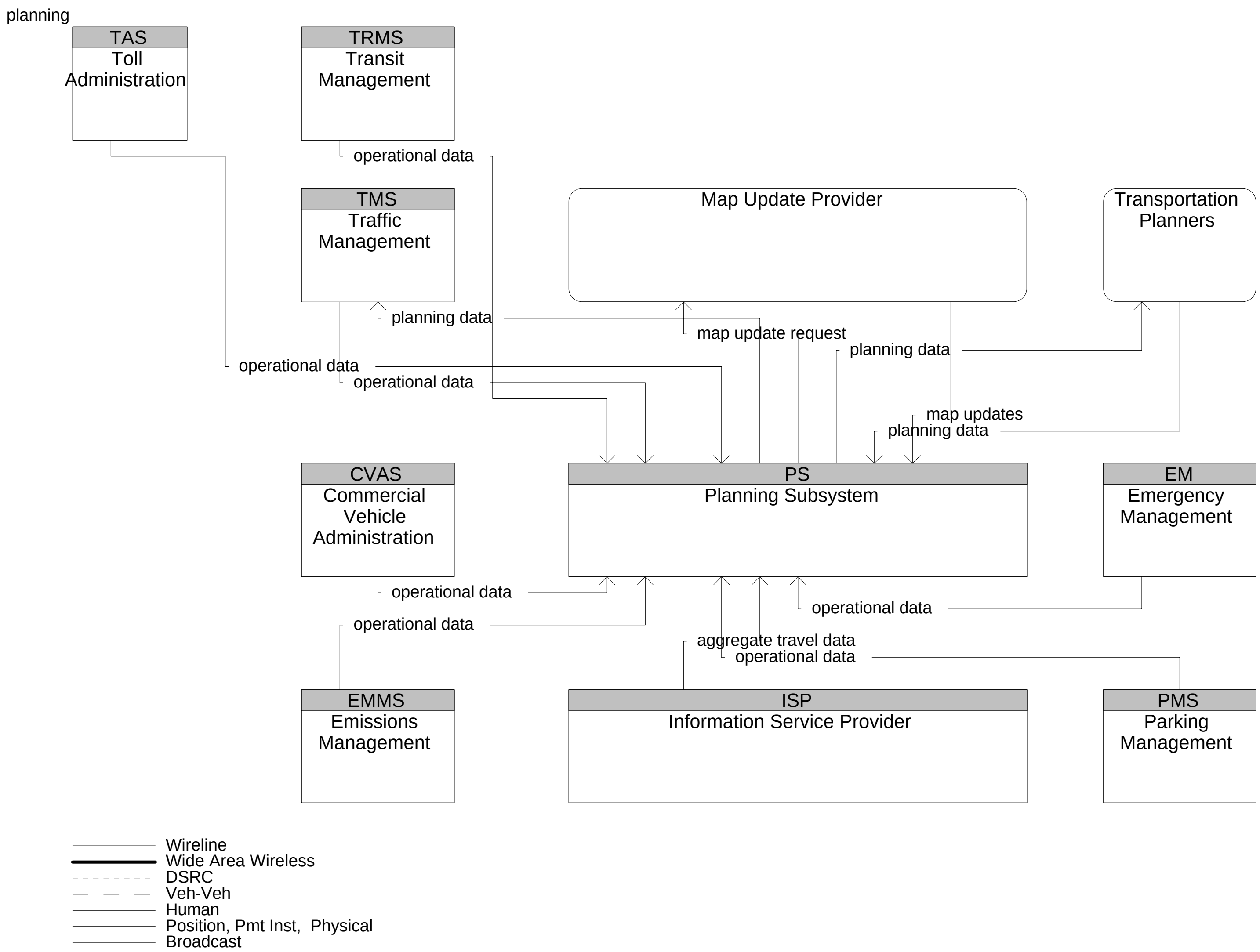


Figure 2.3-5 Architecture Flow Diagram for the Planning Subsystem

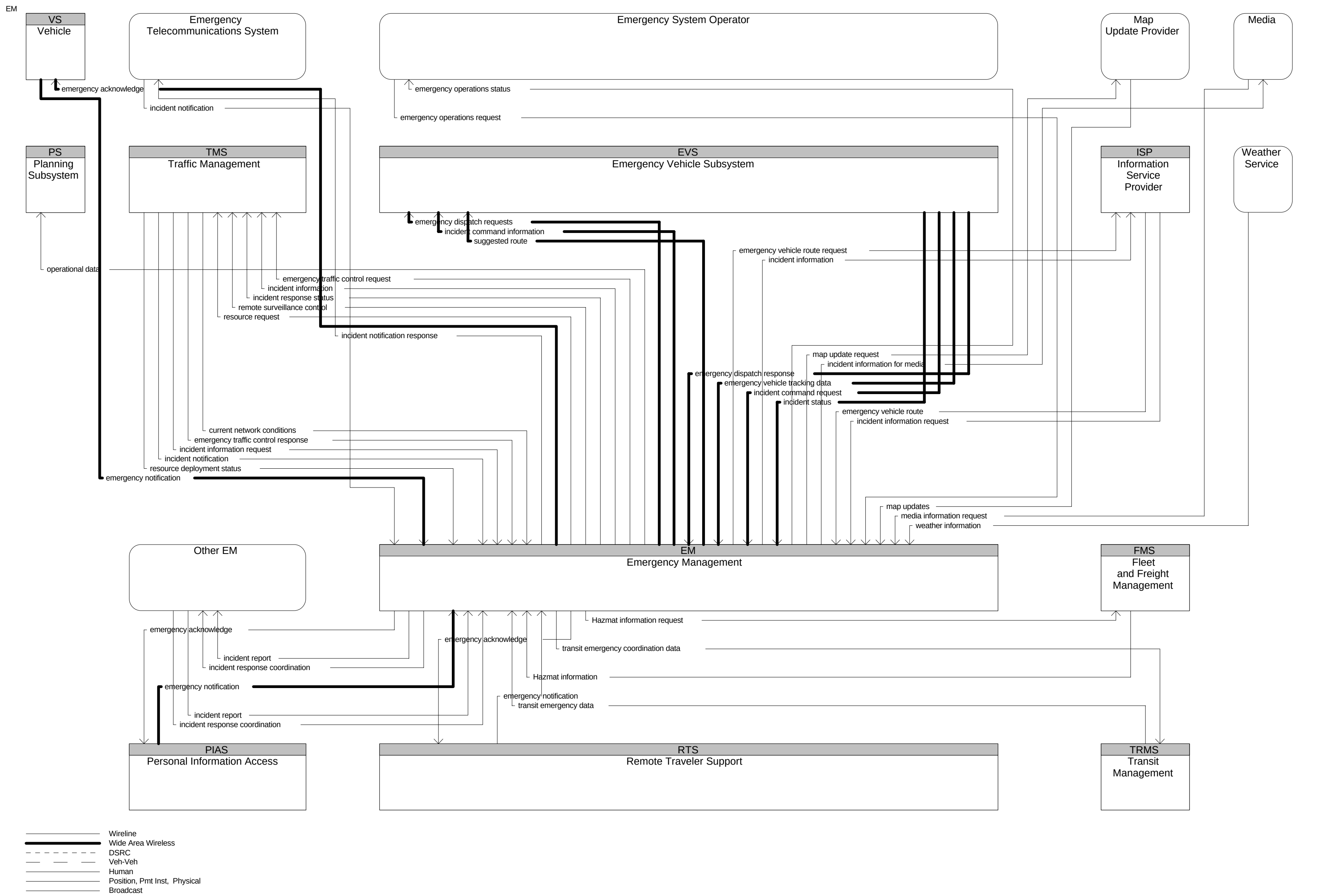


Figure 2.3-6 Architecture Flow Diagram for Emergency Vehicles and Management

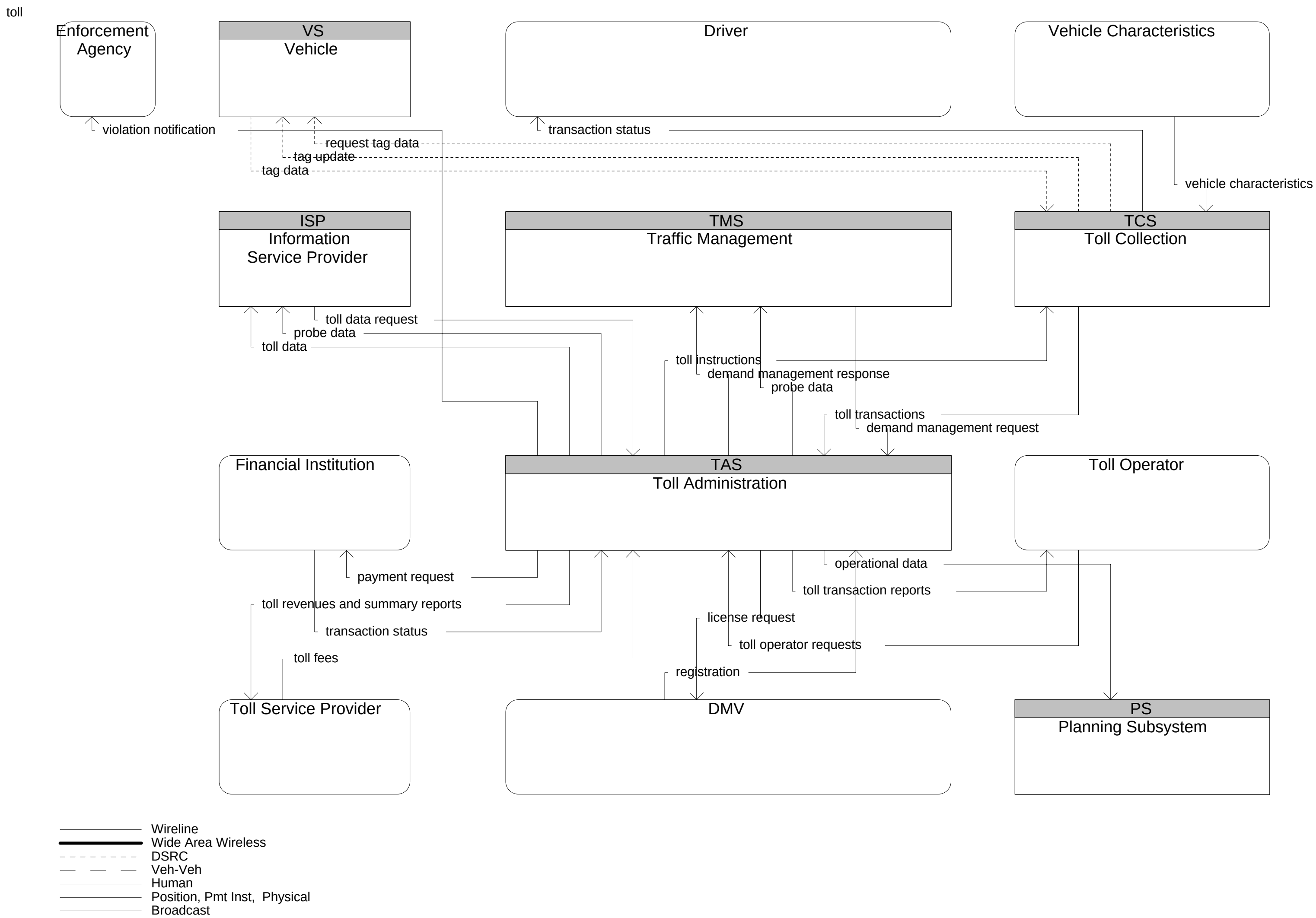


Figure 2.3-7 Architecture Flow Diagrams for Toll Administration and Toll Collection

TMS

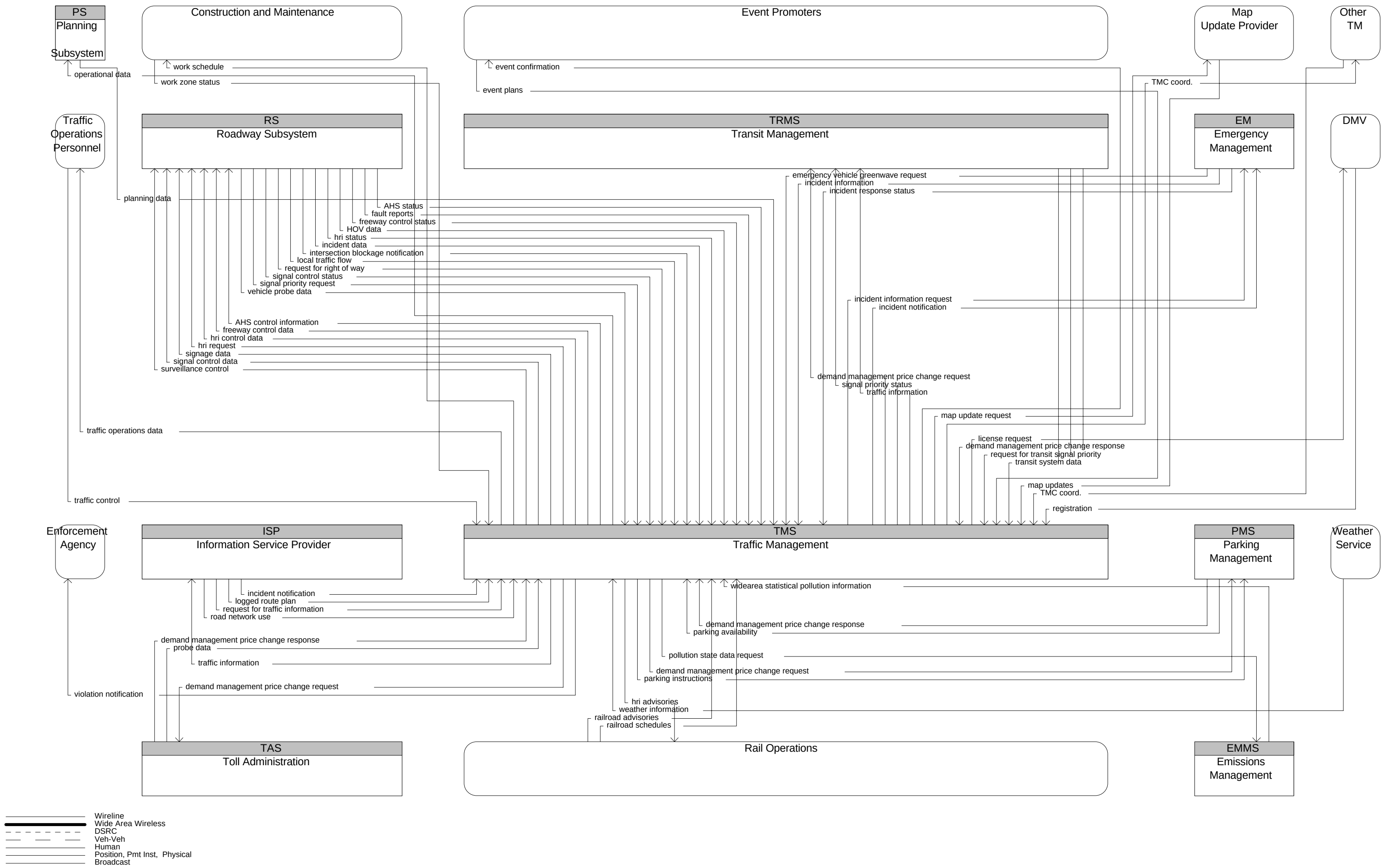


Figure 2.3-8 Architecture Flow Diagram for Traffic Management and Roadside

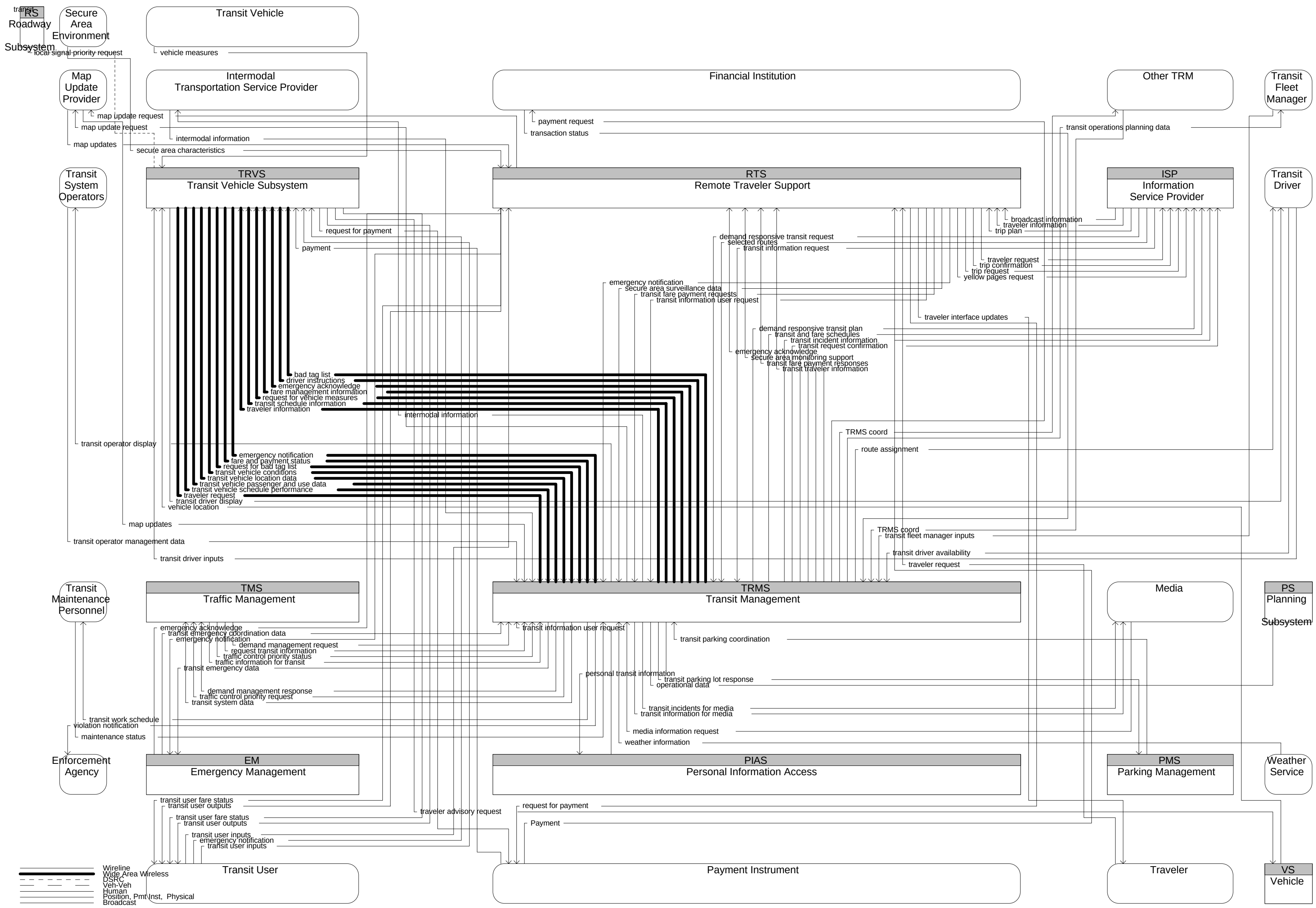


Figure 2.3-9 Architecture Flow Diagram for Transit Management

2.4 Commercial Vehicle Administration

The Commercial Vehicle Administration Subsystem will operate at one or more fixed locations within a region. This subsystem performs administrative functions supporting credentials, tax, and safety regulations. It issues credentials, collects fees and taxes, and supports enforcement of credential requirements. This subsystem communicates with the Fleet Management Subsystems associated with the motor carriers to process credentials applications and collect fuel taxes, weight/distance taxes, and other taxes and fees associated with commercial vehicle operations. The subsystem also receives applications for, and issues special Oversize/Overweight and HAZMAT permits in coordination with other cognizant authorities. The subsystem coordinates with other Commercial Vehicle Administration Subsystems (in other states/regions) to support nationwide access to credentials and safety information for administration and enforcement functions. This subsystem supports communications with Commercial Vehicle Check Subsystems operating at the roadside to enable credential checking and safety information collection. The collected safety information is processed, stored, and made available to qualified stakeholders to identify carriers and drivers that operate unsafely.

2.4.1 Alternative Configurations

The Commercial Vehicle Administration Subsystem can be configured in a number of ways to perform commercial vehicle clearance and safety administration functions. Figure 2.4-1 shows how subsystems can be deployed and operated separately in each state, as a regional center, or international center. Typically, the Commercial Vehicle Administration subsystem is connected to a number of other such subsystems of these subsystems has connections to the Commercial Vehicle Check locations to verify compliance with local requirements. To facilitate exchange of information between the subsystems, there is a special purpose version of the CVAS which maintains common databases from which each of the states may make inquiries and supply data. The content of the data is implementation dependent. In this figure, a separate CVAS (in this case privately operated) has been identified which handles international clearance functions and exchange of data with neighboring countries. Alternative configuration functionality is represented in the following section by the list of suggested equipment packages.

Highlighted configurations include:

- A state agency with interfaces to other states and regional centers, interfaces to in-state private CVO operations, and interfaces to the roadside check facilities
- A regional clearinghouse which maintains appropriate snapshot databases of specific numbers (e.g. carrier, driver, and vehicle numbers)
- An international agency with interfaces to adjacent countries.

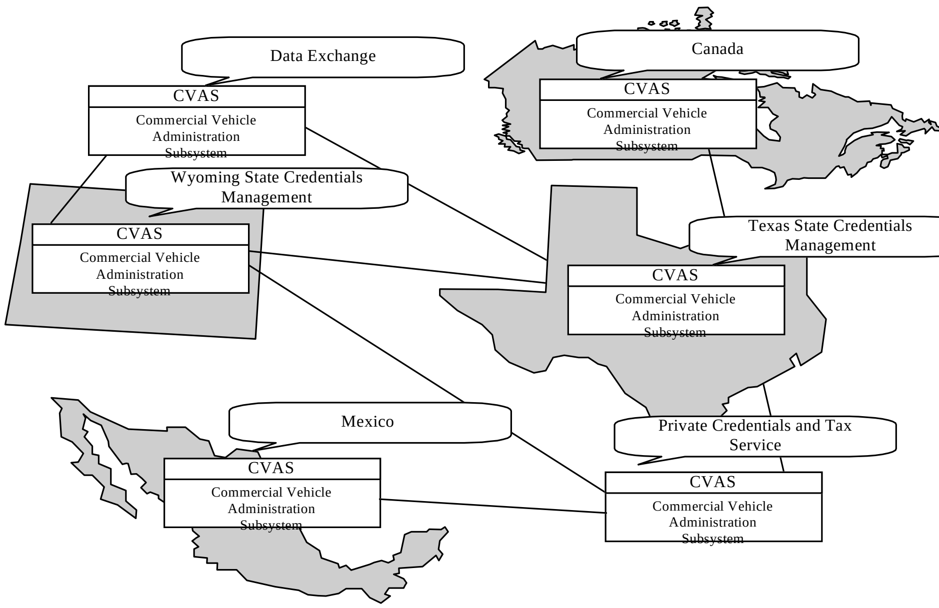


Figure 2.4-1 Alternative Configurations for CVAS

2.4.2 Subsystem Equipment Packages and Process Specifications for CVAS

Credentials and Taxes Administration
This Equipment package provides administrative capabilities for commercial vehicle operations including database management and administrator-to-roadside and administrator-to-administrator interfaces. For example, this Equipment package would manage the electronic credentials database for a state, perform reconciliation of mileage and fuel taxes (possibly post trip), and interface with roadsides performing credential checks. This equipment package communicates with similar packages in other CVAS locations to exchange credentials database information. Example locations would be state agency or regional offices that are involved with commercial vehicle operations.

Process Specifications
2.5.1 Manage Commercial Vehicle Trips and Clearances

Overview: This process shall be responsible for the advance acquisition of electronic credentials and tax filing for commercial vehicles. The process will support the payment of the necessary taxes and duties that will enable a vehicle to be cleared through the credentials checks at the roadside checkstation facilities along its route, including those at border crossings. For this activity the process uses information about the vehicle's route provided by the commercial vehicle manager, or by the driver acting in that role when the vehicle is owned and operated by the driver. The actual payment activity and the subsequent notification of the roadside facilities along the route is carried out by other processes. Where the roadside facilities are outside the area served by the local ITS functions, the process requests that the necessary vehicle data is passed to the similar processes serving the appropriate areas.

2.5.2 Obtain Electronic Credential and Tax Filing Payment

Overview: This process shall be responsible for making payment for electronic credential and tax filing. The data on which the payment is based shall be that for a commercial vehicle's route as provided by the commercial vehicle manager or the commercial vehicle driver who is also the owner of the vehicle. The actual payment activity will be carried out by another process in the Provide Electronic Payment Services function.

2.5.3 Update Permits and Duties Store

Overview: This process shall be responsible for receiving data from Government Administrators. This data comprises updates to the list of electronic credentials and tax filing required for a commercial vehicle to pass each roadside checkstation facility, plus carrier safety ratings for use in roadside safety inspections. These updates are both loaded into a store used by other process in the commercial vehicle administration facility.

2.5.4 Communicate with Other Commercial Vehicle Administration System

Overview: This process shall be responsible for communicating with commercial vehicle administration facilities in ITS functions that serve areas outside that which is served by the local function. The communications supported by the process shall enable the local function to enroll commercial vehicles in other areas, and for those other areas to enroll their commercial vehicles in the local area. The process shall thus support the coordination and the determination of electronic credentials and tax filing across geographic and jurisdictional boundaries.

2.5.5 Manage Commercial Vehicle Credentials and Enrollment

Overview: This process shall be responsible for enabling commercial vehicle managers and drivers (who are owners) to enroll the electronic credentials for their vehicles. This enrollment shall be achieved by loading the credentials data into a data store from which such data shall be downloaded to the commercial vehicle roadside checkstation and border crossing facilities by another process. When the roadside facility is located in the area not served by the local Manage Commercial Vehicles function, the process sends the data to another process that is responsible for communicating with a similar function in other geographic and/or jurisdictional areas. The process shall also be able to accept commercial vehicle enrollment requests from similar functions in other areas, query enforcement agency databases for outstanding prosecutions, and shall be able to respond to requests for information from authorized entities, such as insurance underwriters.

2.5.6 Output Commercial Vehicle Enrollment Data to Roadside Facilities

Overview: This process shall be responsible for providing credentials, safety and border crossing data to commercial vehicle roadside checkstation and border crossing facilities. This data shall be output by the process periodically (e.g. daily) from an interrogation of the stores of safety history and credentials, and sent to the roadside facilities served by the local Manage Commercial Vehicles function. The process shall also provide selected credentials and safety data on request from the commercial vehicle inspectors at particular roadside checkstation facilities.

2.5.7 Process Commercial Vehicle Violations

Overview: This process shall be responsible for sending details of commercial vehicle carriers and drivers that require prosecution to a process in the Manage Emergency Services function. The receiving process in that function will be responsible for sending the data to the appropriate law enforcement agency. This process shall obtain the data by periodically (e.g. daily) scanning the data in the log obtained from the commercial vehicle roadside checkstation facilities.

2.5.8 Process Data Received from Roadside Facilities

Overview: This process shall be responsible for the examination of the daily logs received periodically from the commercial vehicle checkstation and border crossing facilities. It shall also be responsible for the receipt in real time of data about commercial vehicles that have failed their safety inspections. The examination of the received data shall lead the process to update the local stores containing the facility logs and vehicle safety history. This process shall also issue quarterly reports for use by government administrators, send details of the activity at the roadside facility to the Plan System Deployment function. It shall also provide responses to requests from the commercial vehicle manager for reports of fleet activity through roadside facilities, either on-demand or as periodic summaries.

5.4.6 Process CV Violations

Overview: This process shall manage the details of violations committed by commercial vehicles, their drivers and/or operators, reported by the Manage Commercial Vehicles function. The process shall use the parameters in the store of commercial vehicle violation (enforcement) data to obtain the vehicle registration data from the appropriate State Department of Motor Vehicles (DMV) office, before sending all of the received data to the correct law enforcement agency. This process shall also maintain the commercial vehicle violation (enforcement) data store.

7.4.1.1 Process Commercial Vehicle Payments

Overview: This process shall be responsible for transacting payments for electronic credential and tax filing by processes in the Manage Commercial Vehicles function. The payment transaction shall be initiated by processes in the Administer Commercial Vehicles facility which may accept inputs from both the commercial vehicle fleet manager and the commercial vehicle driver acting in the role of fleet manager, i.e. the owner driver. The process shall send the transaction data to the financial institution and report the response back to the requesting process.

CV Information Exchange

This equipment package supports the exchange of safety and credentials data among jurisdiction. The package also supports the exchange of safety and credentials data between agencies (for example, an administrative center and the roadside check facilities) within a single jurisdiction. Data are collected from multiple authoritative sources and packaged into snapshots (top-level summary and critical status information) and profiles (detailed and historical data).

Process Specifications

2.5.4 Communicate with Other Commercial Vehicle Administration System

Overview: This process shall be responsible for communicating with commercial vehicle administration facilities in ITS functions that serve areas outside that which is served by the local function. The communications supported by the process shall enable the local function to enroll commercial vehicles in other areas, and for those other areas to enroll their commercial vehicles in the local area. The process shall thus support the coordination and the determination of electronic credentials and tax filing across geographic and jurisdictional boundaries.

2.5.6 Output Commercial Vehicle Enrollment Data to Roadside Facilities

Overview: This process shall be responsible for providing credentials, safety and border crossing data to commercial vehicle roadside checkstation and border crossing facilities. This data shall be output by the process periodically (e.g. daily) from an interrogation of the stores of safety history and credentials, and sent to the roadside facilities served by the local Manage Commercial Vehicles function. The process shall also provide selected credentials and safety data on request from the commercial vehicle inspectors at particular roadside checkstation facilities.

CV Safety Administration

This Equipment package augments the Credentials and Taxes Administration Equipment package with safety data. This package ensures that safety criteria are available for automated roadside safety checks. Supports the collection and review of carrier safety data and determines the carrier safety rating based on criteria supplied by Government Administration

Process Specifications

2.5.5 Manage Commercial Vehicle Credentials and Enrollment

Overview: This process shall be responsible for enabling commercial vehicle managers and drivers (who are owners) to enroll the electronic credentials for their vehicles. This enrollment shall be achieved by loading the credentials data into a data store from which such data shall be downloaded to the commercial vehicle roadside checkstation and border crossing facilities by another process. When the roadside facility is located in the area not served by the local Manage Commercial Vehicles function, the process sends the data to another process that is responsible for communicating with a similar function in other geographic and/or jurisdictional areas. The process shall also be able to accept commercial vehicle enrollment requests from similar functions in other areas, query enforcement agency databases for outstanding prosecutions, and shall be able to respond to requests for information from authorized entities, such as insurance underwriters.

International CV Administration
This Equipment package is used by government agencies such as customs and immigration, carriers, and service providers (e.g., brokers) to generate and process the entry documentation necessary to obtain release of vehicle, cargo, and driver across and international border, report the results of the crossing event, and handle duty fee processing.

Process Specifications
2.5.1 Manage Commercial Vehicle Trips and Clearances

Overview: This process shall be responsible for the advance acquisition of electronic credentials and tax filing for commercial vehicles. The process will support the payment of the necessary taxes and duties that will enable a vehicle to be cleared through the credentials checks at the roadside checkstation facilities along its route, including those at border crossings. For this activity the process uses information about the vehicle's route provided by the commercial vehicle manager, or by the driver acting in that role when the vehicle is owned and operated by the driver. The actual payment activity and the subsequent notification of the roadside facilities along the route is carried out by other processes. Where the roadside facilities are outside the area served by the local ITS functions, the process requests that the necessary vehicle data is passed to the similar processes serving the appropriate areas.

2.5.6 Output Commercial Vehicle Enrollment Data to Roadside Facilities

Overview: This process shall be responsible for providing credentials, safety and border crossing data to commercial vehicle roadside checkstation and border crossing facilities. This data shall be output by the process periodically (e.g. daily) from an interrogation of the stores of safety history and credentials, and sent to the roadside facilities served by the local Manage Commercial Vehicles function. The process shall also provide selected credentials and safety data on request from the commercial vehicle inspectors at particular roadside checkstation facilities.

2.5.8 Process Data Received from Roadside Facilities

Overview: This process shall be responsible for the examination of the daily logs received periodically from the commercial vehicle checkstation and border crossing facilities. It shall also be responsible for the receipt in real time of data about commercial vehicles that have failed their safety inspections. The examination of the received data shall lead the process to update the local stores containing the facility logs and vehicle safety history. This process shall also issue quarterly reports for use by government administrators, send details of the activity at the roadside facility to the Plan System Deployment function. It shall also provide responses to requests from the commercial vehicle manager for reports of fleet activity through roadside facilities, either on-demand or as periodic summaries.

2.4.3 Subsystem Interfaces for CVAS

Commercial Vehicle Administration => Commercial Vehicle Check

Physical Architecture Flow Name: credentials information

Response containing credentials information.

Logical Architecture Reference Flow: cv_credentials_information_response
This data flow is used within the Manage Commercial Vehicles function and contains the data resulting from a request for some commercial vehicle credentials data to be down loaded to the database maintained by the commercial vehicles roadside checkstation facility. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_credentials_status_code.

Physical Architecture Flow Name: CVO database update

Credential information and safety problem list updates.

Logical Architecture Reference Flow: cv_credentials_database_update
This data flow is used within the Manage Commercial Vehicles function. It contains the list of enrolled commercial vehicle credentials maintained by the commercial vehicle administrative processes and is used to periodically update the credentials database at the roadside checkstation facilities served by the function. The databases provide the facilities with an up to date list of which vehicles have been cleared (enrolled) to potentially pass through without stopping. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_credentials_status_code + cv_trip_classification_data.

Logical Architecture Reference Flow: cv_safety_database_update
This data flow is used within the Manage Commercial Vehicles function and contains data to update the data store containing the safety problem list on a periodic basis (i.e. daily). It contains the following data items each of which is defined in its own DDE: cv_credentials_details + cv_roadside_safety_data.

Physical Architecture Flow Name: international border crossing data

Cleared commercial vehicle data to allow pass-thru international border crossings.

Logical Architecture Reference Flow: cv_border_database_update
This data flow is used within the Manage Commercial Vehicles function. It contains the list of enrolled commercial vehicle credentials maintained by the commercial vehicle administrative processes and is used to periodically update the database at the roadside border crossing facilities served by the function. The databases provide the facilities with an up to date list of which vehicles have been cleared (enrolled) to potentially pass through without stopping. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_trip_identity.

Physical Architecture Flow Name: safety information

Response containing commercial vehicle safety information.

Logical Architecture Reference Flow: cv_safety_information_response
This data flow is used within the Manage Commercial Vehicles function and contains the output resulting from a request by the commercial vehicle roadside checkstation inspector for output of some data from the facility safety database. The data to be output will have been specified by the inspector in terms of a carrier, driver and/or vehicle number. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_roadside_safety_data.

Commercial Vehicle Administration => **CVO Information Requestor**

Physical Architecture Flow Name: credentials and safety information response

Instructions to commercial vehicle managing and/or information systems indicating which vehicles are to be allowed to pass and which are out of service or have not been credentialed.

Logical Architecture Reference Flow: tcvoir_carrier_or_vehicle_information
This data flow is sent to the commercial vehicle information requester from the Manage Commercial Vehicles function. It contains information about a commercial vehicle or carrier and is based on data generated within the function in response to a previous request. The size estimate below is based on a coded response to a standardized query.

Commercial Vehicle Administration => **DMV**

Physical Architecture Flow Name: license request

Request supporting registration data based on license plate read during violation.

Logical Architecture Reference Flow: tdmv_cv_violation_identity_code
This data flow is sent to the department of motor vehicles from the Manage Emergency Services function and contains the identity code of the ITS that is requesting the vehicle registration data so that a commercial vehicle credential filing or tax payment violation can be processed.

Logical Architecture Reference Flow: tdmv_cv_violation_vehicle_license
This data flow is sent to the department of motor vehicles from the Manage Emergency Services function and contains the vehicle license for which the corresponding registration data is required so that a commercial vehicle credential filing or tax payment violation can be processed.

Commercial Vehicle Administration => Enforcement Agency

Physical Architecture Flow Name: request for information on violators

Request for law enforcement information on vehicles and drivers suspected of violations.

Logical Architecture Reference Flow: tea_cv_request_for_information
This data flow is sent to the enforcement agency from the Manage Commercial Vehicles function. It contains a request for information from an enforcement agency about a commercial carrier, vehicle or driver that may be the subject of prosecution for previous violations. The size estimate is based on a coded request (16 bytes) and one subject identity.

Physical Architecture Flow Name: violation notification

Notification to enforcement agency of violation or regulations.

Logical Architecture Reference Flow: tea_cv_violation_data
This data flow is sent from the Manage Emergency Services function to the enforcement agency and contains information about violations of commercial vehicle electronic credential and tax filing payment procedures etc. that have been detected by the Manage Commercial Vehicles function. The data in the flow will enable the notified enforcement agency to take the appropriate action against those committing the violation.

Commercial Vehicle Administration => Financial Institution

Physical Architecture Flow Name: payment request

Request for payment from financial institution.

Logical Architecture Reference Flow: tfi_cv_payment_request
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains a request from a commercial fleet manager or commercial vehicle driver (acting in the role of fleet manager) for payment of electronic credentials and tax filing.

Commercial Vehicle Administration => Fleet and Freight Management

Physical Architecture Flow Name: activity reports

Activity reports containing records of citations, accidents, inspections, etc.

Logical Architecture Reference Flow: cf_periodic_activity_report

This data flow is used within the Manage Commercial Vehicles function. It consists of the data from the commercial vehicle roadside checkstation facility logs from which a report on activities will be issued. This data flow will have been produced as a result of a request from a commercial vehicle manager for periodic reports. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{cv_roadside_facility_identity + cv_roadside_periodic_activity_data}.

Physical Architecture Flow Name: compliance review report

Report containing data from facility activity logs from various roadside facilities.

Logical Architecture Reference Flow: cf_roadside_activity_report
This data flow is used within the Manage Commercial Vehicles function. It contains data from the commercial vehicle roadside checkstation facility logs from which a report on activities will be issued. This data flow will have been produced as a result of a specific request from a commercial vehicle manager and consists of the following data items each of which is defined in its own DDE: list_size + list_size{cv_roadside_facility_identity + cv_roadside_single_activity_data}.

Physical Architecture Flow Name: electronic credentials

Authenticated credentials including route enrollment and payment confirmation.

Logical Architecture Reference Flow: cf_enrollment_information
This data flow is used within the Manage Commercial Vehicles function and contains the data for enrollment on a particular route produced from data supplied by the commercial fleet manager. It contains the following data items each of which is defined in its own DDE: cv_route_number + cv_taxes_and_duties + route + route_type + cv_border_enrollments.

Logical Architecture Reference Flow: cf_enrollment_payment_confirmation
This data flow is used within the Manage Commercial Vehicles function to confirm that a payment of the taxes and duties for the enrollment of a particular commercial vehicle cargo, weight and type on a particular route from the commercial fleet manager has been accepted. It consists of the following data items each of which is defined in its own DDE: cv_route_number + cv_account_number + cv_amount_billed.

Logical Architecture Reference Flow: cv_enrollment_information
This data flow is used within the Manage Commercial Vehicles function and contains the data for enrollment on a particular route produced from data supplied by the commercial vehicle driver. It contains the following data items each of which is defined in its own DDE: cv_route_number + cv_taxes_and_duties + route + route_type + cv_border_enrollments.

Logical Architecture Reference Flow: cv_enrollment_payment_confirmation
This data flow is used within the Manage Commercial Vehicles function to confirm that a payment of the taxes and duties for the enrollment of a particular commercial vehicle cargo, weight and type on a particular route from the commercial vehicle driver has been accepted. It consists of the following data items each of which is defined in its own DDE: cv_account_number + cv_amount_billed + cv_driver_credit_identity + cv_route_number.

Commercial Vehicle Administration => Government Administrators

Physical Architecture Flow Name: tax-credentials-fees request

Request to government agency for tax, credential and/or fee data.

Logical Architecture Reference Flow: tga_quarterly_reports
This data flow is sent to the government agencies from the Manage Commercial Vehicles function and contains the quarterly reports of the data in the commercial vehicle roadside facility log.

Logical Architecture Reference Flow: tga_request_fees_updates
This data flow is sent to the government agencies from the Manage Commercial Vehicles function and contains a request for an update to the store of taxes and credential fees payable for operating commercial vehicles across states and through borders.

Commercial Vehicle Administration => **Other CVAS**

Physical Architecture Flow Name: credentials and safety information request

Request for additional credentials and safety information.

Logical Architecture Reference Flow: tocvas_commit_remote_enrollment
This data flow is sent from to other commercial vehicle administration system by the Manage Commercial Vehicles function and contains a request for the commitment of the enrollment of the carrier, vehicle and driver that has been previously enrolled. This means that the local and any remote enrollment transactions were successful.

Logical Architecture Reference Flow: tocvas_enrollment_confirmation
This data flow is sent to the other commercial vehicle administration system from the Manage Commercial Vehicles function and contains confirmation that enrollment of the carrier, vehicle and driver has been accepted.

Logical Architecture Reference Flow: tocvas_enrollment_request
This data flow is sent to the other commercial vehicle administration system from the Manage Commercial Vehicles function and contains a request for enrollment of the carrier, vehicle and driver.

Physical Architecture Flow Name: CVAS information exchange

Tax and credential fee information exchanged between cooperating commercial vehicle administration offices (e.g. regional or inter-state preclearance data).

Logical Architecture Reference Flow: tocvas_data_table
This data flow is sent to the other commercial vehicle administration system from the Manage Commercial Vehicles function and contains data about required taxes and credential fees.

Logical Architecture Reference Flow: tocvas_provide_data
This data flow is sent to the other commercial vehicle administration system from the Manage Commercial Vehicles function and contains a request for data about required taxes and credential fees to be provided.

Commercial Vehicle Administration => **Planning Subsystem**

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: cv_operational_data
This data is sent from the Manage Commercial Vehicle function to the Plan System Deployment function. It contains data about the number of commercial vehicles passing each roadside checking facility and how many are passing or failing their checks. The data is obtained from the roadside facility log, with details of vehicle identities is removed by the Manage Commercial Vehicles function before the data is sent.

Commercial Vehicle Check => **Commercial Vehicle Administration**

Physical Architecture Flow Name: citation data

Safety problems related to the carrier, driver and vehcile that may lead to a citation.

Logical Architecture Reference Flow: cv_update_safety_problems_list
This data flow is used within the Manage Commercial Vehicles function and contains details of the carrier, driver and vehicle for which safety problems have been found during a inspection at the commercial vehicle roadside checkstation facility. It consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_roadside_facility_identity + cv_roadside_safety_data.

Physical Architecture Flow Name: credentials information request

Request for credential information.

Logical Architecture Reference Flow: cv_credentials_information_request
This data flow is used within the Manage Commercial Vehicles function and contains the request for some commercial vehicle credentials data to be down loaded to the database maintained by the commercial vehicles roadside checkstation facility. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_credentials_request_type + cv_credentials_request_identity + cv_roadside_facility_identity.

Physical Architecture Flow Name: international border crossing data update

Update from commercial vehcile check stations of international border crossing events.

Logical Architecture Reference Flow: cv_border_daily_log
This data flow is used within the Manage Commercial Vehicles function. It contains a copy of the daily log of activities that have taken place at a commercial vehicle border crossing facility. The data flow consists of the following data items each of which is defined in its own DDE: cv_roadside_facility_identity + cv_border_record + date.

Physical Architecture Flow Name: roadside log update

Update of activities at commercial vehicle check stations including clearance events and inspection reports.

Logical Architecture Reference Flow: cv_roadside_daily_log
This data flow is used within the Manage Commercial Vehicles function. It contains a copy of the daily log of all the activities that have taken place at a commercial vehicle roadside checkstation facility. This data is analyzed to determine problem vehicles, drivers and carriers for future pull-in requests and reports to the government administrators. The data flow consists of the following data items each of which is defined in its own DDE: cv_roadside_facility_identity + cv_roadside_record + date.

Physical Architecture Flow Name: safety information request

Request for commercial vehicle safety information.

Logical Architecture Reference Flow: cv_safety_information_request
This data flow is used within the Manage Commercial Vehicles function and contains a request from the commercial vehicle roadside checkstation inspector for output of data held in the facility safety database. The data to be output is specified in terms of a carrier, driver and/or vehicle number. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_safety_information_request_identity + cv_safety_information_request_type + cv_roadside_facility_identity.

CVO Information Requestor => **Commercial Vehicle Administration**

Physical Architecture Flow Name: credentials and safety information request

Request for additional credentials and safety information.

Logical Architecture Reference Flow: fcvoir_request_for_information
This data flow is sent from the commercial vehicle information requester to the Manage Commercial Vehicles function. It contains a request for commercial vehicle operations information such as carrier or vehicle information for use by an organization such as an insurance underwriter. The size estimate below is based on a coded request to execute a standardized query.

DMV => **Commercial Vehicle Administration**

Physical Architecture Flow Name: registration

Registered owner of vehicle and associated vehicle information.

Logical Architecture Reference Flow: fdmv_cv_violation_state_identity
This data flow is sent from the department of motor vehicles to the Manage Emergency Services function and contains the identity of the state that is supplying the requested vehicle registration data to enable a commercial vehicle credential filing or tax payment violation to be processed.

Logical Architecture Reference Flow: fdmv_cv_violation_vehicle_registration
This data flow is sent from the department of motor vehicles to the Manage Emergency Services function and contains the requested vehicle registration data to enable a commercial vehicle credential filing or tax payment violation to be processed.

Enforcement Agency => **Commercial Vehicle Administration**

Physical Architecture Flow Name: information on violators

Response from law enforcement agency to violations notification request.

Logical Architecture Reference Flow: fea_cv_enforcement_agency_response
This data flow is sent from the enforcement agency to the Manage Commercial Vehicles function. It contains the response from an enforcement agency to the previous request for data from the commercial vehicle administration facility. The size estimate below is based on a coded response to a standardized query.

Financial Institution => **Commercial Vehicle Administration**

Physical Architecture Flow Name: transaction status

Response to transaction request. Normally dealing with a request for payment.

Logical Architecture Reference Flow: ffi_cv_payment_confirm
This data flow is sent from the financial institution to the Provide Electronic Payment Services function. It contains confirmation that a previously submitted request from a commercial fleet manager or commercial vehicle driver (acting in the role of fleet manager) for payment of electronic credentials and tax filing has been accepted and made.

Fleet and Freight Management => **Commercial Vehicle Administration**

Physical Architecture Flow Name: credential application

Application for commercial vehicle credentials for a particular route/trip.

Logical Architecture Reference Flow: cf_enrollment_request

This data flow is used within the Manage Commercial Vehicles function and contains the data needed to obtain enrollment information for a particular commercial vehicle cargo, type and weight on a particular route as provided by the commercial fleet manager. It consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_route_data + cv_route_number + cv_trip_classification_data + cv_trip_identity + route_type + border_crossing_request.

Logical Architecture Reference Flow: cv_enrollment_request

This data flow is used within the Manage Commercial Vehicles function and contains the data needed to obtain enrollment information for a particular commercial vehicle cargo, type and weight on a particular route as provided by the commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_route_data + cv_route_number + cv_trip_classification_data + cv_trip_identity + route_type + border_crossing_request.

Physical Architecture Flow Name: information request

General purpose information request for data stored within the commercial vehicle operations information exchange network.

Logical Architecture Reference Flow: cf_request_activity_report

This data flow is used within the Manage Commercial Vehicles function and contains a request from the commercial vehicle manager for one of two types of activity report to be provided. This may be either a specific one time report of clearance safety activity at roadside facilities, or a request that periodic summary reports to be sent on a regular basis. The reports will only relate to the carrier, driver, vehicle combination specified in the request, although it will be possible for all the vehicles and drivers related to a specific carrier to be requested. The data flow contains the following data items each of which is defined in its own DDE: cv_credentials_details + cv_roadside_activity_report_frequency + date + list_size + list_size{cv_roadside_facility_identity}.

Physical Architecture Flow Name: tax filing, audit data

Commercial vehcile tax filing and audit data.

Logical Architecture Reference Flow: cf_enrollment_payment_request

This data flow is used within the Manage Commercial Vehicles function to request payment for the enrollment of a particular commercial vehicle cargo, weight and type on a particular route by the commercial fleet manager. The number of sets of taxes and duties has been set at fourteen (14) for the definition, but has been set to one and a half (1.5) for the size expression as this is a more typical value. The data flow consists of the following items each of which is defined in its own DDE: cf_manager_credit_identity + cv_account_number + cv_route_number + 1{cv_taxes_and_duties}14.

Logical Architecture Reference Flow: cf_tax_audit_data

This data flow is used within the Manage Commercial Vehicles function. It contains tax data and audit filings not related to specific credentials application and is generated as a result of input from the commercial vehicle manager.

Logical Architecture Reference Flow: cv_enrollment_payment_request

This data flow is used within the Manage Commercial Vehicles function to request payment for the enrollment of a particular commercial vehicle cargo, weight and type on a particular route by the commercial vehicle driver acting in the role of the commercial fleet manager. The number of sets of taxes and duties has been set at fourteen (14) for the definition, but has been set to one and a half (1.5) for the size expression as this is a more typical value. The data flow consists of the following items each of which is defined in its own DDE: cv_account_number + cv_driver_credit_identity + cv_route_number + 1{cv_taxes_and_duties}14.

Government Administrators => Commercial Vehicle Administration

Physical Architecture Flow Name: regulations

Regulations imposed on Commercial Vehicle Administration agencies including safety ratings, facility locations and credential fee structure.

Logical Architecture Reference Flow: fga_carrier_safety_ratings
This data flow is sent by the government administrator to the Manage Commercial Vehicles function. It contains the current safety rating for a carrier determined as a result of previous roadside safety inspections.

Logical Architecture Reference Flow: fga_roadside_facility_locations
This data flow is sent by the government administrator to the Manage Commercial Vehicles function. It contains the location of a commercial vehicle roadside checkstation facility. roadside_facility_list_FB.

Logical Architecture Reference Flow: fga_tax_and_credential_fees
This data flow is sent from the government agencies to the Manage Commercial Vehicles function and contains data about the taxes and fees payable by commercial vehicle operators for the movement of vehicles through states and across borders.

Other CVAS => Commercial Vehicle Administration

Physical Architecture Flow Name: credentials and safety information response

Instructions to commercial vehicle managing and/or information systems indicating which vehicles are to be allowed to pass and which are out of service or have not been credentialed.

Logical Architecture Reference Flow: focvas_commit_local_enrollment
This data flow is sent from the other commercial vehicle administration system to the Manage Commercial Vehicles function and contains a request for the commitment of the enrollment of the carrier, vehicle and driver that has been previously enrolled. This means that the other remote enrollment transactions were successful.

Logical Architecture Reference Flow: focvas_enrollment_confirmation
This data flow is sent from the other commercial vehicle administration system to the Manage Commercial Vehicles function and contains confirmation that enrollment of the carrier, vehicle and driver has been accepted.

Logical Architecture Reference Flow: focvas_enrollment_request
This data flow is sent from the other commercial vehicle administration system to the Manage Commercial Vehicles function and contains a request for enrollment of the carrier, vehicle and driver.

Physical Architecture Flow Name: CVAS information exchange

Tax and credential fee information exchanged between cooperating commercial vehicle administration offices (e.g. regional or inter-state preclearance data).

Logical Architecture Reference Flow: focvas_data_table
This data flow is sent from the other commercial vehicle administration system to the Manage Commercial Vehicles function and contains data about required taxes and credential fees.

Logical Architecture Reference Flow: focvas_provide_data
This data flow is sent from the other commercial vehicle administration system to the Manage Commercial Vehicles function and contains a request for data about required taxes and credential fees to be provided.

2.4.4 Subsystem Architecture Flow Diagram

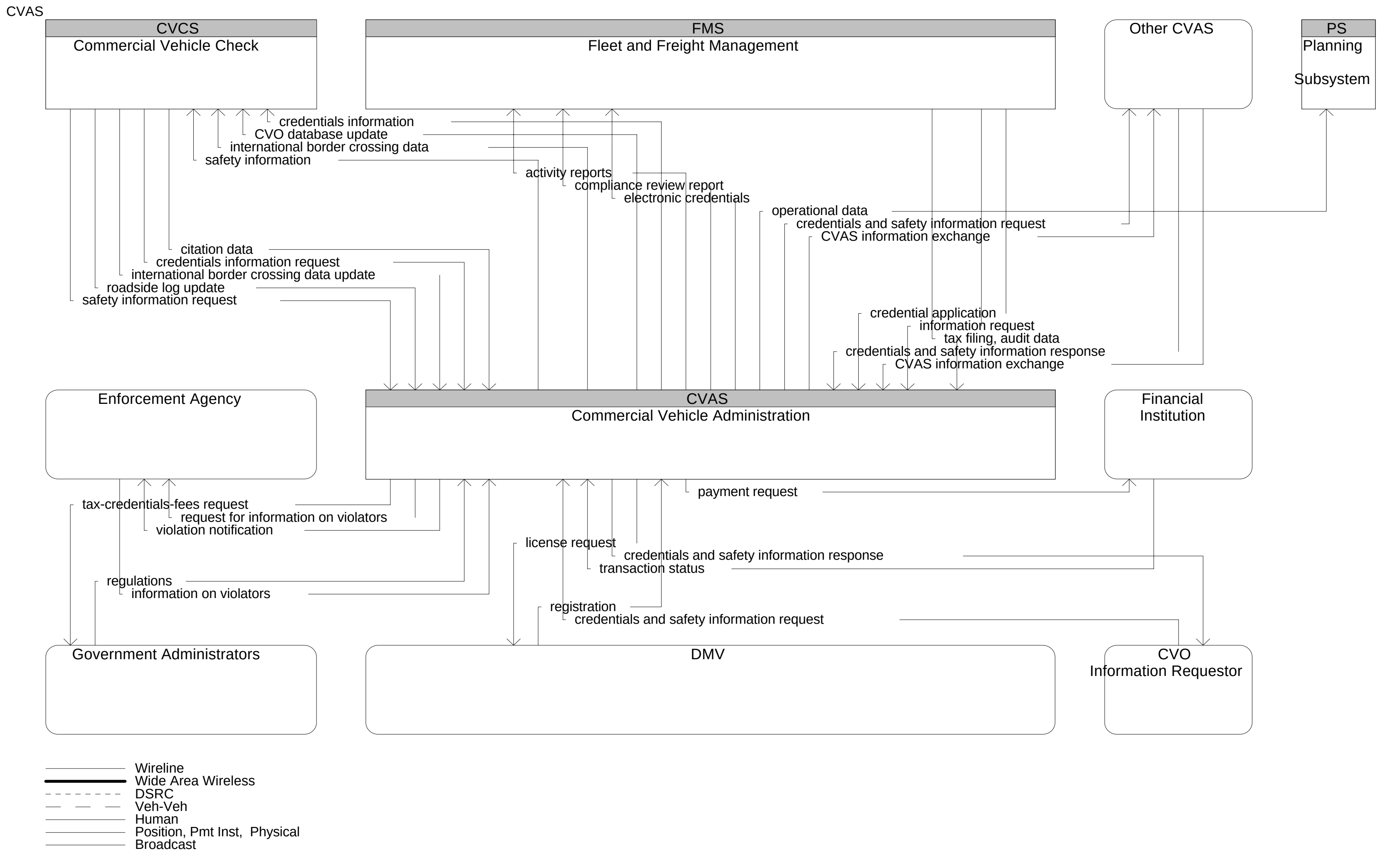


Figure 2.4-2 Architecture Flow Diagram for CVAS

2.5 Commercial Vehicle Check Subsystem

The Commercial Vehicle Check Subsystem supports automated vehicle identification at mainline speeds for credential checking, roadside safety inspections, and weigh-in-motion using two-way data exchange. These capabilities include providing warnings to the commercial vehicle drivers, their fleet managers, and proper authorities of any safety problems that have been identified, accessing and examining historical safety data, and automatically deciding whether to allow the vehicle to pass or require it to stop with operator manual override. The Commercial Vehicle Check Subsystem also provides supplemental inspection services to current capabilities by supporting expedited brake inspections, the use of operator hand-held devices, on-board safety database access, and the enrollment of vehicles and carriers in the electronic clearance program.

2.5.1 Alternative Configurations

The Commercial Vehicle Check Subsystem is the asset which is placed by the roadside to check compliance of vehicles with regulations and to spot check commercial vehicles for highway safety Figure 2.5-1. These stations will be equipped with beacon technology to read tags from properly equipped vehicles and to indicate to the vehicles whether the check was successful. Stations may also be equipped with Weigh-in-motion sensors to check vehicle weight at roadway speed. The Check Station will also have a database against which to check the information on each commercial vehicle. This database will be supplied by the Commercial Vehicle Administration subsystem. The Check Station may be a fixed facility communicating with the CVAS over normal wireline dialup or leased lines. The Check Stations may also be a mobile station which obtains its databases at the beginning of the day and communicates violations over a wide area wireless medium, or may be a mobile station which saves database updates until the end of some period and then updates the CVAS in a batch mode.

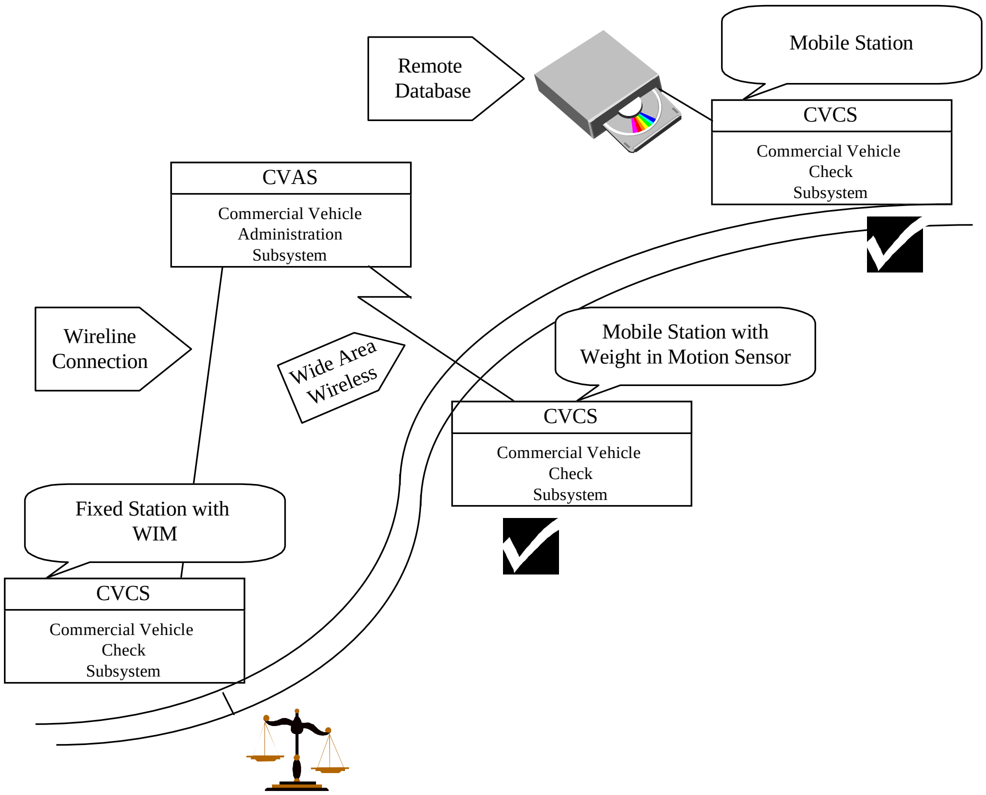


Figure 2.5-1 Alternative Configurations for CVCS

2.5.2 Subsystem Equipment Packages and Supporting Process Specifications for CVCS

Citation and Accident Electronic Recording
The equipment package documents violations and forwards the information to the Commercial vehicle if available and to the CVAS for processing as part of the normal credentials processing package

Process Specifications
2.3.3.4 Carry-out Commercial Vehicle Roadside Safety Screening

Overview: This process shall be responsible for checking commercial vehicle credentials against the list of those known to have safety problems held in a store maintained by another process in the roadside checkstation facility. The process shall send the result of each check to the roadside inspector interface process so that an override input can be generated if required. The process shall send a request for the commercial vehicle to pull-in if the vehicle's credentials are in the list of those with safety problems, and shall also send a record of each decision to the process that maintains the commercial vehicle roadside checkstation facility log.

International Border Crossing
This Equipment package is used by government agencies such as customs and immigration to check compliance with import/export and immigration regulations to allow release of cargo, vehicle, and driver across an international border.

Process Specifications

2.3.8 Provide Commercial Vehicle Border Screening

Overview: This process shall be responsible for checking a commercial vehicle and its cargo through a border crossing point. The checks carried out by the process shall comprise a comparison of the trip identity already provided by the commercial vehicle administration processes, and held in a local data store. A check shall also be made by the process to see if the lock tag attached to the vehicle's cargo has been changed. If either of these two checks produce negative results then the process shall request the vehicle to pull-in, otherwise the vehicle shall be allowed to pass. The process shall send its decision to the process that provides the roadside inspectors' interface, to enable an override to be applied if required. The decision of the process (with the override if it is applied) shall be sent to the message output process and be written back to the vehicle's on-board tag.

Roadside Electronic Screening

This Equipment package provides the Commercial Vehicle Check Subsystem the capabilities for two-way communication with approaching properly equipped commercial vehicles at mainline speeds, reading tags for automated vehicle identification and credential checking. There will be a capability to appropriately screen all vehicles, not just those that are equipped. This Equipment package shall be able to process the data from the commercial vehicles along with accessed database information to determine whether a pull-in message is needed or to generate random pull-in messages with provisions for facility operators and enforcement officials to have manual override capabilities. Support shall be provided to both interstate and intrastate carriers.

Process Specifications

2.3.1 Produce Commercial Vehicle Driver Message at Roadside

Overview: This process shall be responsible for the output of pull-in or pass messages to commercial vehicle drivers as they approach the commercial vehicle roadside checkstation or border crossing facilities. The process shall support the use of roadside equipment such as variable message signs (vms), or simple red-green lights, flashing orange lights, etc. to provide the output. These output messages shall be received by the process from other processes responsible for roadside facilities within the Manage Commercial Vehicles function. The process shall support pull-in messages that are the result of checks on a commercial vehicle's electronic credentials, safety and border crossing data, the result of the vehicle's tag not being properly read, or the result of a general pull-in decision for all vehicles being issued by inspectors at the roadside facility. The process shall also generate a message to be sent to the vehicle so that an indication can be output directly to the driver at the same time as it appears on the roadside equipment.

2.3.2.1 Administer Commercial Vehicle Roadside Credentials Database

Overview: This process shall be responsible for receiving the electronic credentials sent to the roadside checkstation facility as part of a commercial vehicle's enrollment process. The process shall store the data for use by another process and shall also enable the inspector in the roadside facility to obtain a copy of the data in the store. If the requested data is not in the store, the process shall request it from another process in the commercial vehicle administration facility.

2.3.2.2 Process Screening Transactions

Overview: This process shall be responsible for checking commercial vehicle credentials against those held in a store maintained by another process in the roadside checkstation facility. The process shall send the result of each check to the roadside inspector interface process so that an override input can be generated if required. The process shall send a request for the commercial vehicle to pull-in if the vehicle's credentials do not match those in the store, and shall also send a record of each decision to the process that maintains the commercial vehicle roadside checkstation facility log.

2.3.3.4 Carry-out Commercial Vehicle Roadside Safety Screening

Overview: This process shall be responsible for checking commercial vehicle credentials against the list of those known to have safety problems held in a store maintained by another process in the roadside checkstation facility. The process shall send the result of each check to the roadside inspector interface process so that an override input can be generated if required. The process shall send a request for the commercial vehicle to pull-in if the vehicle's credentials are in the list of those with safety problems, and shall also send a record of each decision to the process that maintains the commercial vehicle roadside checkstation facility log.

2.3.4 Detect Commercial Vehicle

Overview: This process shall be responsible for detecting the presence of commercial vehicles through the use of sensors that can differentiate between the different types of vehicle. The process shall use the sensors to determine the number of axles, gross vehicle weight and weight per axle data for use by inspectors at the roadside checkstation facilities. When a commercial vehicle is detected, the process shall transmit a request for its on-board tag data, which when received shall be passed to other processes within the roadside facility. If no tag data is received, or the data cannot be interpreted correctly, the process shall send a request for the vehicle to pull-in to be output by another process in the roadside checkstation facility.

2.3.5 Provide Commercial Vehicle Roadside Operator Interface

Overview: This process shall be responsible for providing the commercial vehicle inspector interface at the roadside checkstation facility. The process shall provide an interface which enables the inspector to monitor and if necessary override the pull-in decisions made by those of the border crossing, credentials and safety data checking processes that are present in the facility. The process shall also make it possible for the inspector to issue a manual general pull-in request for all commercial vehicles to pull into the roadside checkstation facility, to have access the contents of the facility's log, and to obtain credentials or safety data on a selected combination of carrier, driver, and vehicle. The process shall support inputs from the traffic operations personnel in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be in hardcopy, or as a display.

2.3.6 Provide Commercial Vehicle Reports

Overview: This process shall be responsible for collecting data from those of the border crossing, credential and safety checking processes that are present in a commercial vehicle roadside checkstation facility. The data shall be stored by the process in a roadside facility log, to which the roadside inspector interface process shall have access. Once a day the process shall make a copy of the roadside facility log and send it to the commercial vehicle administration facility for further processing.

Roadside Safety Inspection

This Equipment package provides the Commercial Vehicle Check Subsystem the capabilities for operators to automate the roadside safety inspection process including the support of use of hand held devices to rapidly inspect the vehicle and driver. In addition this Equipment package provides the Roadside Check Subsystem the capabilities for operators to automate the roadside safety inspection process including the support of automated mainline speed reading of on-board safety data to rapidly screen the vehicle and driver. This Equipment package shall also provide the capabilities to collect, store, maintain, and provide safety data and access historical safety data after receiving identification from vehicles at mainline speeds or while stopped at the roadside. Results of screening and summary safety inspection can be written back onto the tag. The capabilities to process safety data and issue pull-in messages or provide warnings to the driver, carrier, and enforcement agencies shall be provided. These capabilities have a prerequisite of the Roadside Electronic Screening Equipment package and shall be provided primarily through the utilization of an additional safety database.

Since a vehicle may cross jurisdiction boundaries during a trip, this equipment package supports the concept of a last clearance event record (aka trip ticket) carried on the vehicle s tag. The last clearance event record reflects the results of the roadside verification action. For example, if the vehicle is pulled over in State A and undergoes credential, weight, and safety checks, the results of the clearance process are written to the vehicle s tag. If the vehicle continues the trip and passes a roadside station in State B, the State B station has access to the results of the previous pull-in because it can read the last clearance event record written by the State A roadside station.

Process Specifications

2.3.3.1 Provide Commercial Vehicle Checkstation Communications

Overview: This process shall be responsible for providing an interface through which a commercial vehicles roadside checkstation facility can communicate with a passing commercial vehicle. When a request for on-board data is received from another process within the facility, the process shall issue a data request to the identified commercial vehicle. The data received by the process from the vehicle shall be stored in the store of collected data for use by the roadside inspection process.

2.3.3.2 Provide Commercial Vehicle Inspector Handheld Terminal Interface

Overview: This process shall be responsible for providing an interface for a hand held terminal which can be used by a commercial vehicle inspector. The process shall enable the inspector to start a commercial vehicle roadside inspection, to review the results, and to add comments to the results data. The process shall support inputs from the inspectors in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the form of the visual output to be in hardcopy, or as a display.

2.3.3.3 Administer Commercial Vehicle Roadside Safety Database

Overview: This process shall be responsible for maintaining in the commercial vehicle roadside checkstation facility a database of credentials for commercial vehicles with safety problems. This process shall store the data about these vehicles received from the commercial vehicle administration facility. It shall enable this data to be used by another process and shall also enable the inspector in the roadside facility to obtain a copy of the data in the store. If the requested data is not in the store, the process shall request it from another process in the commercial vehicle administration facility.

2.3.3.4 Carry-out Commercial Vehicle Roadside Safety Screening

Overview: This process shall be responsible for checking commercial vehicle credentials against the list of those known to have safety problems held in a store maintained by another process in the roadside checkstation facility. The process shall send the result of each check to the roadside inspector interface process so that an override input can be generated if required. The process shall send a request for the commercial vehicle to pull-in if the vehicle's credentials are in the list of those with safety problems, and shall also send a record of each decision to the process that maintains the commercial vehicle roadside checkstation facility log.

2.3.3.5 Carry-out Commercial Vehicle Roadside Inspection

Overview: This process shall be responsible for carrying out roadside safety inspections at the request of the roadside facility inspector. The result of the inspection shall be sent by the process to the inspector, the commercial vehicle driver, the roadside checkstation facility log, and the commercial vehicle itself. The process shall enable the inspector to add comments to the result of the inspection before it is sent to the above outputs. These comments shall be received by the process in the form of data input from the inspector's hand held terminal interface.

2.3.5 Provide Commercial Vehicle Roadside Operator Interface

Overview: This process shall be responsible for providing the commercial vehicle inspector interface at the roadside checkstation facility. The process shall provide an interface which enables the inspector to monitor and if necessary override the pull-in decisions made by those of the border crossing, credentials and safety data checking processes that are present in the facility. The process shall also make it possible for the inspector to issue a manual general pull-in request for all commercial vehicles to pull into the roadside checkstation facility, to have access the contents of the facility's log, and to obtain credentials or safety data on a selected combination of carrier, driver, and vehicle. The process shall support inputs from the traffic operations personnel in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be in hardcopy, or as a display.

Roadside WIM

This Equipment package allows for roadside high speed weigh in motion. This package can be fixed to a location or mobile. It can include an interface to the credential check package and augment electronic credentials check with electronic weight check or it can be a stand alone package with display.

Process Specifications

2.3.4 Detect Commercial Vehicle

Overview: This process shall be responsible for detecting the presence of commercial vehicles through the use of sensors that can differentiate between the different types of vehicle. The process shall use the sensors to determine the number of axles, gross vehicle weight and weight per axle data for use by inspectors at the roadside checkstation facilities. When a commercial vehicle is detected, the process shall transmit a request for its on-board tag data, which when received shall be passed to other processes within the roadside facility. If no tag data is received, or the data cannot be interpreted correctly, the process shall send a request for the vehicle to pull-in to be output by another process in the roadside checkstation facility.

2.5.3 Subsystem Interfaces for CVCS

Commercial Vehicle => **Commercial Vehicle Check**

Physical Architecture Flow Name: CVO weight and presence

Weigh-In-Motion message to indicate presence of commercial vehicle and its weight.

Logical Architecture Reference Flow: fcv_vehicle_characteristics
This data flow is used within the Manage Commercial Vehicles function and represents vehicle characteristics that are sensed at a commercial vehicle roadside checking facility. This data may be acquired by sensors from the vehicle either electronically, optically, or manually and will include data such as weight, size, number of axles, use of trailer, etc. The size assumption stated below is based on an equivalent digital size for loading estimation purposes.

Commercial Vehicle Administration => **Commercial Vehicle Check**

Physical Architecture Flow Name: credentials information

Response containing credentials information.

Logical Architecture Reference Flow: cv_credentials_information_response
This data flow is used within the Manage Commercial Vehicles function and contains the data resulting from a request for some commercial vehicle credentials data to be down loaded to the database maintained by the commercial vehicles roadside checkstation facility. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_credentials_status_code.

Physical Architecture Flow Name: CVO database update

Credential information and safety problem list updates.

Logical Architecture Reference Flow: cv_credentials_database_update
This data flow is used within the Manage Commercial Vehicles function. It contains the list of enrolled commercial vehicle credentials maintained by the commercial vehicle administrative processes and is used to periodically update the credentials database at the roadside checkstation facilities served by the function. The databases provide the facilities with an up to date list of which vehicles have been cleared (enrolled) to potentially pass through without stopping. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_credentials_status_code + cv_trip_classification_data.

Logical Architecture Reference Flow: cv_safety_database_update
This data flow is used within the Manage Commercial Vehicles function and contains data to update the data store containing the safety problem list on a periodic basis (i.e. daily). It contains the following data items each of which is defined in its own DDE: cv_credentials_details + cv_roadside_safety_data.

Physical Architecture Flow Name: international border crossing data

Cleared commercial vehicle data to allow pass-thru international border crossings.

Logical Architecture Reference Flow: cv_border_database_update

This data flow is used within the Manage Commercial Vehicles function. It contains the list of enrolled commercial vehicle credentials maintained by the commercial vehicle administrative processes and is used to periodically update the database at the roadside border crossing facilities served by the function. The databases provide the facilities with an up to date list of which vehicles have been cleared (enrolled) to potentially pass through without stopping. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_trip_identity.

Physical Architecture Flow Name: safety information

Response containing commercial vehicle safety information.

Logical Architecture Reference Flow: cv_safety_information_response

This data flow is used within the Manage Commercial Vehicles function and contains the output resulting from a request by the commercial vehicle roadside checkstation inspector for output of some data from the facility safety database. The data to be output will have been specified by the inspector in terms of a carrier, driver and/or vehicle number. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_roadside_safety_data.

Commercial Vehicle Check => **Commercial Vehicle Administration**

Physical Architecture Flow Name: citation data

Safety problems related to the carrier, driver and vehcile that may lead to a citation.

Logical Architecture Reference Flow: cv_update_safety_problems_list

This data flow is used within the Manage Commercial Vehicles function and contains details of the carrier, driver and vehicle for which safety problems have been found during a inspection at the commercial vehicle roadside checkstation facility. It consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_roadside_facility_identity + cv_roadside_safety_data.

Physical Architecture Flow Name: credentials information request

Request for credential information.

Logical Architecture Reference Flow: cv_credentials_information_request

This data flow is used within the Manage Commercial Vehicles function and contains the request for some commercial vehicle credentials data to be down loaded to the database maintained by the commercial vehicles roadside checkstation facility. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_credentials_request_type + cv_credentials_request_identity + cv_roadside_facility_identity.

Physical Architecture Flow Name: international border crossing data update

Update from commercial vehcile check stations of international border crossing events.

Logical Architecture Reference Flow: cv_border_daily_log

This data flow is used within the Manage Commercial Vehicles function. It contains a copy of the daily log of activities that have taken place at a commercial vehicle border crossing facility. The data flow consists of the following data items each of which is defined in its own DDE: cv_roadside_facility_identity + cv_border_record + date.

Physical Architecture Flow Name: roadside log update

Update of activities at commercial vehicle check stations including clearance events and inspection reports.

Logical Architecture Reference Flow: cv_roadside_daily_log

This data flow is used within the Manage Commercial Vehicles function. It contains a copy of the daily log of all the activities that have taken place at a commercial vehicle roadside checkstation facility. This data is analyzed to determine problem vehicles, drivers and carriers for future pull-in requests and reports to the government administrators. The data flow consists of the following data items each of which is defined in its own DDE: cv_roadside_facility_identity + cv_roadside_record + date.

Physical Architecture Flow Name: safety information request

Request for commercial vehicle safety information.

Logical Architecture Reference Flow: cv_safety_information_request

This data flow is used within the Manage Commercial Vehicles function and contains a request from the commercial vehicle roadside checkstation inspector for output of data held in the facility safety database. The data to be output is specified in terms of a carrier, driver and/or vehicle number. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_safety_information_request_identity + cv_safety_information_request_type + cv_roadside_facility_identity.

Commercial Vehicle Check => Commercial Vehicle Driver

Physical Architecture Flow Name: CVO Pull in message

Message sent to commercial vehicle driver requesting pull in to inspection/verification stop along with inspection results.

Logical Architecture Reference Flow: tcvd_border_pull_in_output

This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicle function and represents the actual message to be conveyed to the driver of a commercial vehicle that is being pulled in to a roadside facility for border clearance (permits, duties, trip number, etc.) reasons.

Logical Architecture Reference Flow: tcvd_clearance_pull_in_output

This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicle function and represents the actual message to be conveyed to the driver of a commercial vehicle that is being pulled in to a roadside facility for clearance (permits duties, etc.) reasons.

Logical Architecture Reference Flow: tcvd_general_pull_in_output

This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicle function. It represents the actual message to be conveyed to the driver of a vehicle that is being pulled in to a commercial vehicle roadside checking facility for general reasons using roadside displays, e.g. variable message signs (vms), etc.

Logical Architecture Reference Flow: tcvd_inspection_results

This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicles function and represents a message to the driver of a commercial vehicle that provides the results of an inspection at a commercial vehicle roadside check facility.

Logical Architecture Reference Flow: tcvd_safety_pull_in_output

This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicles function and represents the actual message to be conveyed to the driver of a commercial vehicle that is being pulled in to a roadside facility for safety reasons.

Commercial Vehicle Check => Commercial Vehicle Subsystem

Physical Architecture Flow Name: border clearance event record

Results of border clearance check.

Logical Architecture Reference Flow: cv_on_board_border_record
This data flow is used within the Manage Commercial Vehicles function. It contains the results of the of the border clearance checks at a commercial vehicle roadside border crossing facility. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_on_board_border_details.

Physical Architecture Flow Name: clearance event record

Results of vehicle clearance activity.

Logical Architecture Reference Flow: cv_on_board_screening_record
This data flow is used within the Manage Commercial Vehicles function and contains the results of the screening of a commercial vehicle at a particular commercial vehicle roadside checking facility. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_on_board_screening_details.

Physical Architecture Flow Name: electronic clearance request

Request for electronic clearance data (Toll, safety, customs, etc.).

Logical Architecture Reference Flow: cv_request_electronic_clearance_data
This data flow is used within the Manage Commercial Vehicles function. It contains a request from a commercial vehicle roadside checkstation or border crossing facility for the output of the current contents of a commercial vehicle's tag, plus the cargo lock tag, if one is being carried by the vehicle.

Physical Architecture Flow Name: lock tag data request

Request to supply lock information on cargo lock for retransmission to international border crossing station.

Logical Architecture Reference Flow: cv_request_electronic_clearance_data
This data flow is used within the Manage Commercial Vehicles function. It contains a request from a commercial vehicle roadside checkstation or border crossing facility for the output of the current contents of a commercial vehicle's tag, plus the cargo lock tag, if one is being carried by the vehicle.

Physical Architecture Flow Name: on-board safety request

Request for onboard vehicle safety data.

Logical Architecture Reference Flow: cv_request_on_board_data
This data flow is used within the Manage Commercial Vehicles function by the commercial vehicle roadside processing to request on-board data from a commercial vehicle. It contains the following data items each of which is defined in its own DDE: cv_roadside_facility_identity + cv_roadside_facility_address.

Physical Architecture Flow Name: pass/pull-in

Command to commercial vehicle to pull into inspection station.

Logical Architecture Reference Flow: cv_on_board_pull_in_output
This data flow is used in the Manage Commercial Vehicles function and contains the output result of the commercial vehicle safety or screening processes, in terms of a pull-in or pass decision for the vehicle, or a general pull-in pass decision for all vehicles, or a pull-in decision based on a problem with reading the vehicle's tag. It consists of the following data items, each of which is defined in its own DDE: [cv_general_pull-in | cv_screening_pull_in_output | cv_safety_pull_in_output | cv_border_pull_in_output].

Physical Architecture Flow Name: safety inspection record

Record containing results of commercial vehicle safety inspection.

Logical Architecture Reference Flow: cv_inspection_data_output
This data flow is used within the Manage Commercial Vehicle function and contains the results of the commercial vehicle roadside inspection. These are down loaded for storage on-board the vehicle.

Physical Architecture Flow Name: screening request

Request for screening data based on vehicle and possibly cargo's tags.

Logical Architecture Reference Flow: cv_request_electronic_clearance_data
This data flow is used within the Manage Commercial Vehicles function. It contains a request from a commercial vehicle roadside checkstation or border crossing facility for the output of the current contents of a commercial vehicle's tag, plus the cargo lock tag, if one is being carried by the vehicle.

Commercial Vehicle Check => **CVO Inspector**

Physical Architecture Flow Name: CVO inspector information

Credential, safety, and preclearance information and instructions to the commercial vehicle inspector.

Logical Architecture Reference Flow: tci_credentials_data_output
This data flow is sent to the commercial vehicle inspector from the Manage Commercial Vehicles function and contains the formatted output of the previously requested credentials for a particular combination of carrier, driver and vehicle.

Logical Architecture Reference Flow: tci_inspection_report
This data flow is sent to the commercial vehicle inspector from the Manage Commercial Vehicles function and contains the formatted output of the results of the commercial vehicle roadside inspection previously initiated by the inspector.

Logical Architecture Reference Flow: tci_output_log_report
This data flow is sent to the commercial vehicle inspector from the Manage Commercial Vehicles function and contains the commercial vehicle roadside checking facility log showing which vehicles have been stopped, passed, or pulled-in by the inspectors.

Logical Architecture Reference Flow: tci_pull_in_information
This data flow is sent to the commercial vehicle inspector from the Manage Commercial Vehicles function and contains details of the pull-in or pass decision made as a result of the safety or preclearance processing for a commercial vehicle. The inspector can override this decision if needed.

Logical Architecture Reference Flow: tci_safety_data_output
This data flow is sent to the commercial vehicle inspector from the Manage Commercial Vehicles function and contains the formatted output of the safety data for a particular carrier, driver or vehicle. This data will have been previously requested by the commercial vehicle roadside inspector.

Commercial Vehicle Subsystem => Commercial Vehicle Check

Physical Architecture Flow Name: electronic clearance data

Information required for electronic clearance (toll, safety, customs, etc.).

Logical Architecture Reference Flow: cv_electronic_clearance_data
This data flow is used within the Manage Commercial Vehicles and Provide Electronic Payment Services functions. It contains data that has been stored on a commercial vehicle tag to enable its identification at commercial vehicle roadside checkstation facilities for the purposes of electronic clearance, and/or safety inspection, and/or border clearance, as well as at toll plazas for toll payment collection. The data flow consists of the following data items each of which is defined in its own DDE: cv_on_board_tag_data + cv_lock_tag_data.

Physical Architecture Flow Name: lock tag data

Tag information on cargo lock.

Logical Architecture Reference Flow: cv_electronic_clearance_data
This data flow is used within the Manage Commercial Vehicles and Provide Electronic Payment Services functions. It contains data that has been stored on a commercial vehicle tag to enable its identification at commercial vehicle roadside checkstation facilities for the purposes of electronic clearance, and/or safety inspection, and/or border clearance, as well as at toll plazas for toll payment collection. The data flow consists of the following data items each of which is defined in its own DDE: cv_on_board_tag_data + cv_lock_tag_data.

Physical Architecture Flow Name: on board safety data

Vehicle safety data measured by vehicle sensors and sent to inspection stations

Logical Architecture Reference Flow: cv_on_board_data
This data flow is used within the Manage Commercial Vehicles function to send on-board commercial vehicle data from the vehicle to a commercial vehicle roadside facility. It contains the following data items each of which is defined in its own DDE: cv_identity_details + cv_credentials + cv_driver_credentials + cv_driver_license_citations + cv_repairs_and_service_records + cv_inspection_data + cv_inspection_activities_data + cv_fuel_purchase_data + cv_safety_systems_diagnostics_results + cv_vehicle_log + cv_log_data + cv_not_pulled_in.

Physical Architecture Flow Name: screening data

Data stored in vehicle's tag allowing electronic clearance at border crossings, debits at toll plazas, and clearance at safety inspections.

Logical Architecture Reference Flow: cv_electronic_clearance_data
This data flow is used within the Manage Commercial Vehicles and Provide Electronic Payment Services functions. It contains data that has been stored on a commercial vehicle tag to enable its identification at commercial vehicle roadside checkstation facilities for the purposes of electronic clearance, and/or safety inspection, and/or border clearance, as well as at toll plazas for toll payment collection. The data flow consists of the following data items each of which is defined in its own DDE: cv_on_board_tag_data + cv_lock_tag_data.

CVO Inspector => Commercial Vehicle Check

Physical Architecture Flow Name: CVC override mode

Manual override by the commercial vehicle roadside facility inspector of automated pass/pull-in signage information.

Logical Architecture Reference Flow: fci_pull_in_action
This data flow is sent from the commercial vehicle roadside facility inspector to the Manage Commercial Vehicles function. It contains an override of the pull-in or pass decision made as a result of the safety or preclearance processing for a commercial vehicle.

Physical Architecture Flow Name: CVO inspector input

Requests from the commercial vehicle inspector to operate the commercial vehicle inspection station.

Logical Architecture Reference Flow: fci_credentials_data_request
This data flow is sent from the commercial vehicle roadside facility inspector to the Manage Commercial Vehicles function. It contains a request for the output of the credentials data for a particular combination of carrier, vehicle and driver.

Logical Architecture Reference Flow: fci_inspection_data_input
This data flow is sent from the commercial vehicle roadside facility inspector to the Manage Commercial Vehicles function. It contains data about a commercial vehicle inspection that can only be supplied by the inspector.

Logical Architecture Reference Flow: fci_request_log_report
This data flow is sent from the commercial vehicle roadside facility inspector to the Manage Commercial Vehicles function. It contains a request for a part of the commercial vehicle roadside facility log to be output to its operator.

Logical Architecture Reference Flow: fci_safety_data_request
This data flow is sent from the commercial vehicle roadside facility inspector to the Manage Commercial Vehicles function. It contains a request for output of the safety data for a particular carrier, driver and vehicle being held in the commercial vehicle roadside facility safety database.

Logical Architecture Reference Flow: fci_start_inspection
This data flow is sent from the commercial vehicle inspector to the Manage Commercial Vehicles function and represents the roadside commercial vehicle inspectors' initiation of a vehicle inspection. The size assumption is based on some form of electronic signature of an authorized inspector.

2.5.4 Subsystem Architecture Flow Diagram

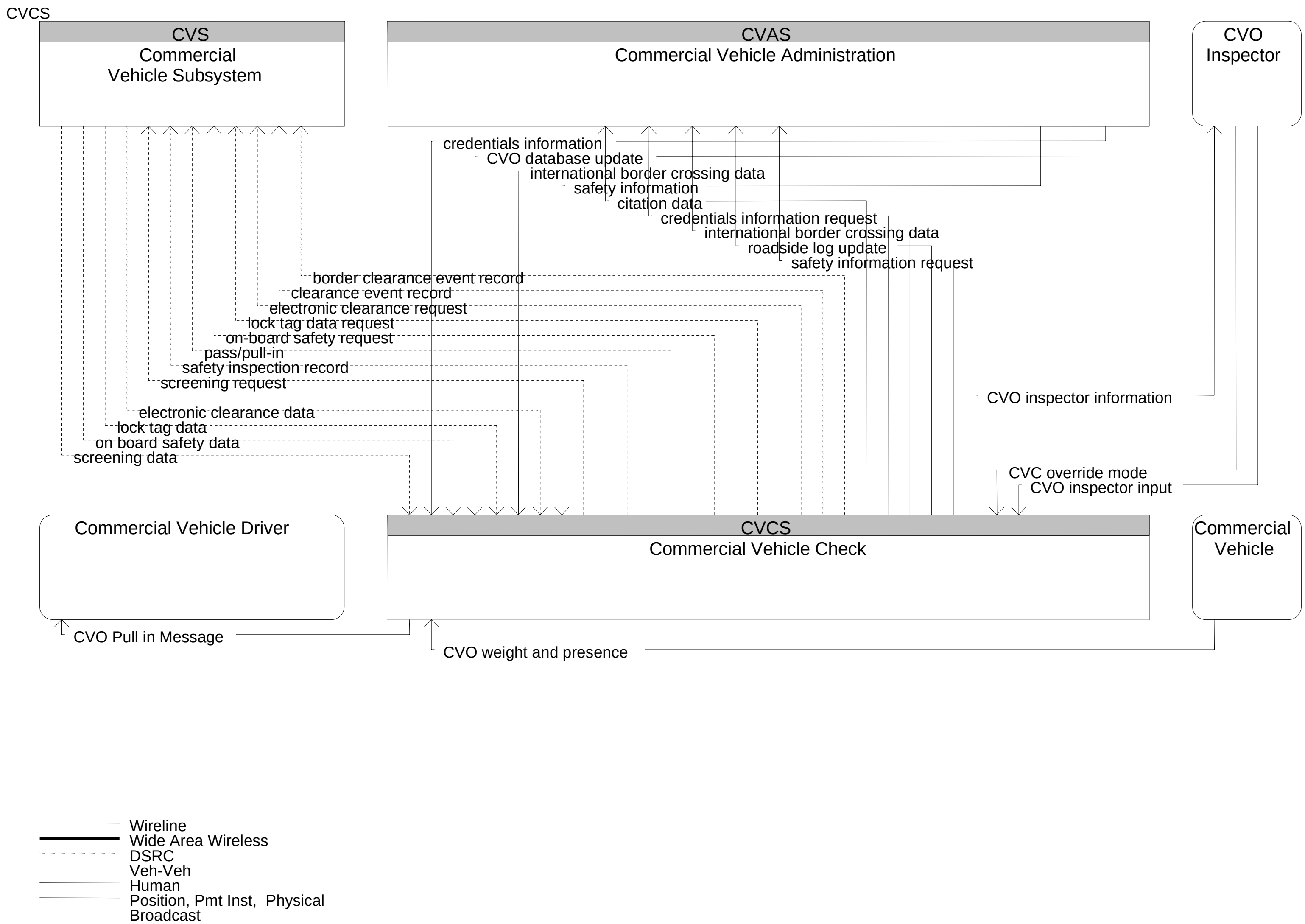


Figure 2.5-2 Architecture Flow Diagram for CVCS

2.6 Commercial Vehicle Subsystem

This subsystem resides in a commercial vehicle and provides the sensory, processing, storage, and communications functions necessary to support safe and efficient commercial vehicle operations. The Commercial Vehicle Subsystem provides two-way communications between the commercial vehicle drivers, their fleet managers, and roadside officials, and provides HAZMAT response teams with timely and accurate cargo contents information after a vehicle incident. This subsystem provides the capability to collect and process vehicle, cargo, and driver safety data and status and alert the driver whenever there is a potential safety problem. Basic identification and safety status data are supplied to inspection facilities at mainline speeds. In addition, the subsystem will automatically collect and record mileage, fuel usage, and border crossings.

2.6.1 Alternative Configurations

The Commercial Vehicle Subsystem contains functionality specifically associated with a commercial vehicle Figure 2.6-1. It is anticipated that the Commercial Vehicle will take advantage of all functions available in the Vehicle Subsystem in addition to the special CV applications. These common features include the ability to pay for tolls and communicate with the infrastructure to obtain up to date traffic information and route planning. In the future, these functions will include advanced vehicle safety systems. CV applications include AVLS functions for both primarily urban (short haul) and cross country (long haul). Specific capabilities are installed in the CV subsystem to support at speed checking of credentials and on-board safety information by a roadside check station. Special on-board safety checking of the commercial vehicle operation (e.g. tire pressure, load balance, break lining...) if available could be also made available to the roadside stations.

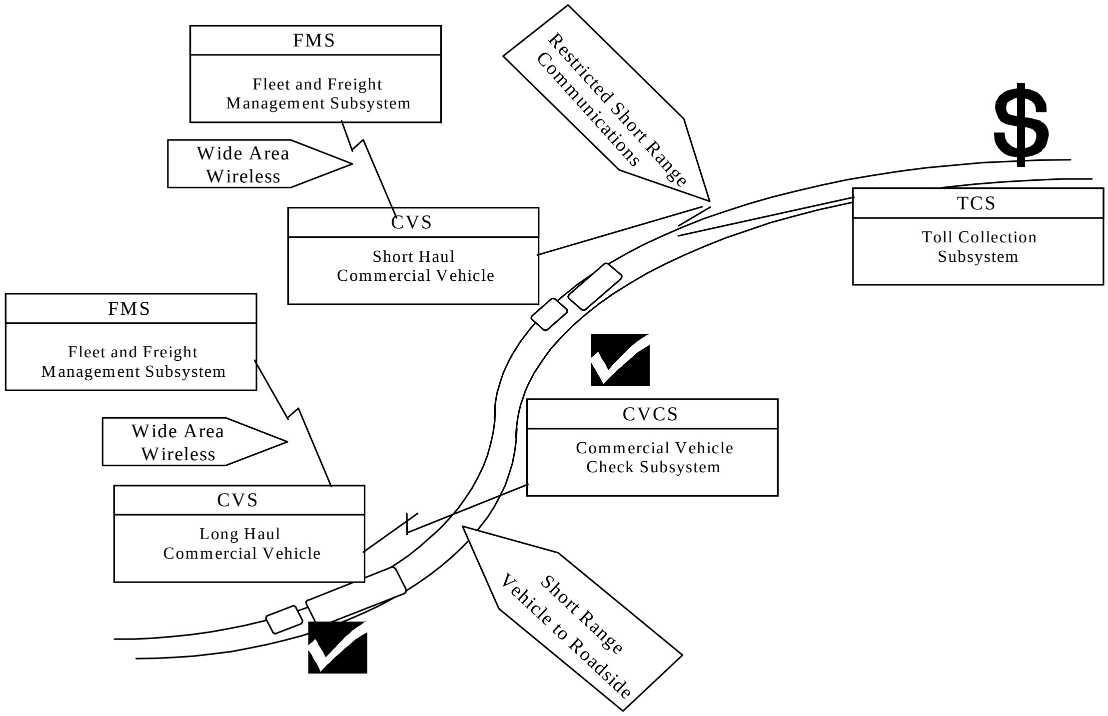


Figure 2.6-1 Alternative Configurations for CVS

2.6.2 Subsystem Equipment Packages and Supporting Process Specifications for CVS

On-board Cargo Monitoring

This Equipment package provides the Commercial Vehicle Subsystem the capability to monitor both interstate and intrastate cargo safety such that enforcement and HAZMAT response teams can be provided with timely and accurate information. This includes only the equipment on board the cargo container such as a communication device, possibly the addition of a cell-based radio, and equipment for the processing and storage of cargo material. This can also include optional sensors for temperature, pressure, load leveling, or acceleration depending upon the items monitored. It is already expected that the cargo location devices such as GPS equipment and an integration processor already exist. These items are presented as part of the On-board Trip Monitoring Equipment package.

Process Specifications
3.3.1 Provide Cargo Data for Incident Notification

Overview: This process shall be responsible for providing data about a commercial vehicle's cargo in the event that the vehicle is involved in some type of emergency. The process shall produce the output on request from another process in the vehicle regardless of whether the cargo has itself suffered from any damage. The cargo data being provided by the process shall cover all types, regardless of whether or not they are classified as HAZMAT cargoes.

On-board CV Electronic Data

This Equipment package provides the Commercial Vehicle Subsystem the capability for two-way data exchange between the vehicle and the roadside facility with the transmission of information such as status of driver, vehicle, and carrier IDs and cargo information. The driver, vehicle and carrier are identified via the tag so that actual weight from roadside mainline weigh-in-motion may be checked. This includes only the equipment on the commercial vehicle including a processor/tag for identification, especially a HAZMAT identification. The actual reading and processing required for the credential checking and weigh-in-motion will be performed by the roadside.

Process Specifications

2.2.3 Provide CV Driver Electronic Credential and Tax Filing Interface

Overview: This process shall be responsible for providing an interface for the commercial vehicle fleet manager. In this instance the driver is assumed to be acting in the role of a commercial vehicle manager, and is therefore likely to be the owner/driver of the vehicle. The process shall enable this interface to provide the driver with facilities for the input of data used to set up commercial vehicle routes, to pay all the necessary taxes and duties so that a commercial vehicle can be enrolled for a particular route, and to obtain a copy of the data collected by processes on-board the vehicle. The enrollment activity supported by the process shall enable a commercial vehicle to pass through the roadside checkstations along its route without stopping, unless safety checks are required. The process shall support inputs from the commercial vehicle driver in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be in hardcopy, or as a display.

2.3.7 Produce Commercial Vehicle Driver Message on Vehicle

Overview: This process shall be responsible for the output of the pull-in or pass messages to commercial vehicle drivers directly in their vehicles as they approach a commercial vehicle roadside checkstation facility. These messages shall be generated by other processes within the facility that are responsible for checking the commercial vehicle's credentials (including those for border crossing) and safety, or may be the result of the vehicle's tag not being properly read, or may be the result of a general pull-in decision for all vehicles being issued by inspectors at the roadside checkstation facility.

2.4.2 Collect On-board Commercial Vehicle Sensor Data

Overview: This process shall be responsible for continuously monitoring the conditions on-board a commercial vehicle. These inputs shall be processed by sensors, and if required converted from analog into a digital form. The process shall load all collected into an on-board vehicle data store for use by other processes in the vehicle.

2.4.3 Analyze Commercial Vehicle On-board Data

Overview: This process shall be responsible for analyzing the data collected on-board a commercial vehicle, and sending it to another process for loading into a store on-board the vehicle. If the analysis of the data carried out by the process shows that there is a critical safety problem, the process shall send data to the driver's interface process for output to the driver. The process shall also accept input of data from the commercial vehicle driver via the interface process and load it into the same store.

2.4.4 Provide Commercial Vehicle Driver Interface

Overview: This process shall be responsible for providing the interface between the commercial vehicle driver and processes on-board the commercial vehicle. The process shall provide interfaces to the processes responsible for collecting, analyzing and storing data about the vehicle, its cargo, the driver, etc., and for the exchange of data with the commercial vehicle manager. The process shall support inputs from the driver in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be in hardcopy, or as a display.

2.4.6 Provide Commercial Vehicle On-board Data Store Interface

Overview: This process shall be responsible for providing the interface through which data can be written to and read from the store of data that is held on-board a commercial vehicle. The data shall be provided by and on request from, other processes within the Manage Commercial Vehicles function that are on-board the vehicle.

2.6.2 Transmit Commercial Vehicle Tag Data

Overview: This process shall be responsible for providing the output of the data that has been previously stored on-board a commercial vehicle's tag on request from a commercial vehicle roadside checkstation facility. The process shall also provide the current status of the lock tag, if one is attached to the vehicle's cargo. The data shall only be sent by the process to the commercial vehicle roadside checkstation or border crossing facility that made the request. The output mechanism used by the process shall be an implementation issue, but it could be by radio, beacon, or a visual mechanism, such as a bar code.

2.6.3 Provide Commercial Driver Tag Data Interface

Overview: This process shall be responsible for providing the interface through which the commercial vehicle driver can set up the data in an on-board vehicle unit (e.g. an electronic tag). In this instance the driver is assumed to be acting in the role of a commercial vehicle manager, and is thus likely to be the owner of the vehicle. The data the process enables the manager to write to the tag will be that which identifies the carrier, driver and vehicle. The process shall also enable the read this data from the tag, but shall not enable the manager to read any other data from the tag. The process shall support inputs from the commercial vehicle driver in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be in hardcopy, or as a display.

2.6.4 Provide Lock Tag Data Interface

Overview: This process shall be responsible for producing an output of the current status of a lock tag that is being carried by the cargo of a commercial vehicle. The process shall only produce the output in response to a request for data that is received from the other process on-board the vehicle that is responsible for communication with commercial vehicle roadside checkstation facilities. The actual output mechanism used by the process shall be an implementation issue, but it could be by radio or beacon.

2.6.5 Manage Commercial Vehicle Tag Data Store

Overview: This process shall be responsible for managing the store of data that is held by a commercial vehicle's on-board tag. It shall manage all of the transactions that either write data to the store and read data from it, to ensure that the data retains its consistency. The process shall ensure that the commercial vehicle manager or driver can only read the data that they are enabled to write to the store, and that the store only contains data from the last two roadside checkstation facilities passed by the commercial vehicle.

On-board CV Safety

This Equipment package provides the Commercial Vehicle Subsystem the capability to collect and process on board vehicle and driver safety information to monitor the safety status and supply this information to the roadside facilities both at mainline speeds and while stopped for inspections. The capability to alert the commercial vehicle driver whenever there is a critical safety problem or potential emergency shall also be provided. These capabilities include only the equipment on the commercial vehicle including the sensors and processors to monitor the vehicle and driver with the information stored on the vehicle. When the information is transmitted to the roadside facility or after the trip, it will utilize the communication devices already in place. The package will also support onboard driver safety log maintenance and checking.

Process Specifications

2.4.1 Communicate Commercial Vehicle On-board Data to Roadside

Overview: This process shall be responsible for providing the commercial vehicle end of the communications link between itself and a commercial vehicle roadside checkstation facility. The process shall enable an inspector at the facility (or elsewhere, but with a suitably equipped hand held terminal) to have access to the data accumulated on-board the vehicle for use in a vehicle inspection. It shall also enable the inspector to send back data about the result of the inspection for storage on-board the vehicle.

2.4.2 Collect On-board Commercial Vehicle Sensor Data

Overview: This process shall be responsible for continuously monitoring the conditions on-board a commercial vehicle. These inputs shall be processed by sensors, and if required converted from analog into a digital form. The process shall load all collected into an on-board vehicle data store for use by other processes in the vehicle.

2.4.3 Analyze Commercial Vehicle On-board Data

Overview: This process shall be responsible for analyzing the data collected on-board a commercial vehicle, and sending it to another process for loading into a store on-board the vehicle. If the analysis of the data carried out by the process shows that there is a critical safety problem, the process shall send data to the driver's interface process for output to the driver. The process shall also accept input of data from the commercial vehicle driver via the interface process and load it into the same store.

2.4.4 Provide Commercial Vehicle Driver Interface

Overview: This process shall be responsible for providing the interface between the commercial vehicle driver and processes on-board the commercial vehicle. The process shall provide interfaces to the processes responsible for collecting, analyzing and storing data about the vehicle, its cargo, the driver, etc., and for the exchange of data with the commercial vehicle manager. The process shall support inputs from the driver in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be in hardcopy, or as a display.

On-board Trip Monitoring

This Equipment package provides the capabilities to support fleet management with automatic vehicle location and automated mileage and fuel reporting and auditing. This package may also record other special events resulting from communication with roadside equipment. This includes only the equipment on board the vehicle to support this function including the vehicle location devices such as GPS equipment, communication interfaces, a processor to record trip length, and the sensors/actuators/interfaces necessary to record mileage and fuel usage.

Process Specifications

2.1.5 Provide Commercial Vehicle Driver Routing Interface

Overview: This process shall be responsible for providing the communications interface through which a commercial vehicle driver can obtain details of the vehicle route that has been provided by the commercial vehicle manager. The process shall enable the output of the route instructions in audio and/or visual form. It shall be possible for the visual form to be either hardcopy output, or in the form of a display. The process shall retain the data for a particular route internally, so that successive requests for details of the same route do not require use of the communications network.

2.2.2 Provide Vehicle Static Route

Overview: This process shall be responsible for providing a static commercial vehicle route using data provided by the commercial vehicle driver. A static route is one which is based on geographic data and therefore takes no account of current or predicted traffic conditions, incidents, etc. The process shall provide the route using its own route generation algorithms and data from its own store of digitized map information. In this instance the driver is assumed to be acting in the role of a commercial vehicle manager, and is therefore likely to be the owner/driver of the vehicle.

2.2.4 Provide Commercial Vehicle Driver Communications

Overview: This process shall be responsible for providing communications between the commercial vehicle driver and the commercial vehicle. In this instance the driver is acting in the role of vehicle manager, and is therefore likely to be the owner/driver of the vehicle. The process shall support the receipt of data from the vehicle consisting of that processed from input received by sensors on board the vehicle The process shall enable access to the store of received data by the driver through the driver's interface process.

2.4.1 Communicate Commercial Vehicle On-board Data to Roadside

Overview: This process shall be responsible for providing the commercial vehicle end of the communications link between itself and a commercial vehicle roadside checkstation facility. The process shall enable an inspector at the facility (or elsewhere, but with a suitably equipped hand held terminal) to have access to the data accumulated on-board the vehicle for use in a vehicle inspection. It shall also enable the inspector to send back data about the result of the inspection for storage on-board the vehicle.

2.4.2 Collect On-board Commercial Vehicle Sensor Data

Overview: This process shall be responsible for continuously monitoring the conditions on-board a commercial vehicle. These inputs shall be processed by sensors, and if required converted from analog into a digital form. The process shall load all collected into an on-board vehicle data store for use by other processes in the vehicle.

2.4.3 Analyze Commercial Vehicle On-board Data

Overview: This process shall be responsible for analyzing the data collected on-board a commercial vehicle, and sending it to another process for loading into a store on-board the vehicle. If the analysis of the data carried out by the process shows that there is a critical safety problem, the process shall send data to the driver's interface process for output to the driver. The process shall also accept input of data from the commercial vehicle driver via the interface process and load it into the same store.

2.4.4 Provide Commercial Vehicle Driver Interface

Overview: This process shall be responsible for providing the interface between the commercial vehicle driver and processes on-board the commercial vehicle. The process shall provide interfaces to the processes responsible for collecting, analyzing and storing data about the vehicle, its cargo, the driver, etc., and for the exchange of data with the commercial vehicle manager. The process shall support inputs from the driver in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be in hardcopy, or as a display.

2.4.5 Communicate Commercial Vehicle On-board Data to Vehicle Manager

Overview: This process shall be responsible for providing the communications interface through which the commercial vehicle manager (or commercial vehicle driver acting in the role of the manager) can access the data stored on-board a commercial vehicle. The process shall also support the exchange of unformatted messages between the commercial vehicle manager and driver, and the ability of the driver to send the on-board data to the manager as an unsolicited data flow.

2.6.3 Subsystem Interfaces for CVS

Commercial Vehicle => **Commercial Vehicle Subsystem**

Physical Architecture Flow Name: vehicle measures

Sensing information from vehicle sensors.

Logical Architecture Reference Flow: fcv_brake_condition
This data flow is used within the Manage Commercial Vehicle function. It contains analog data from on-board commercial vehicle sensors from which the current condition of the vehicle's brakes can be determined.

Logical Architecture Reference Flow: fcv_cargo_data
This data flow is sent from the basic vehicle to the Provide Vehicle Monitoring and Control function. It contains sensor input from which the following can be determined: cargo_type - solid, liquid, powder, flammable liquid, livestock, etc., cargo_hazmat - details of the HAZMAT designation (if any) for the cargo, cargo_weight - the actual weight of the cargo including its packaging, i.e. what weight the vehicle is actually carrying, cargo_packaging - the type of container in which the cargo is held, e.g. closed ISO container, tank, open container, refrigerated container, etc. Size is 4 bytes per cargo element.

Logical Architecture Reference Flow: fcv_cargo_safety_status
This data flow is used within the Manage Commercial Vehicle function. It contains analog data from on-board commercial vehicle sensors from which the current condition of the vehicle's cargo can be determined. The data flow has been sized at 4 bytes per sensor with 4 sensors, giving a total of sixteen (16) bytes.

Logical Architecture Reference Flow: fcv_distance_travelled
This data flow is used within the Manage Commercial Vehicle function. It contains data from on-board commercial

vehicle sensors from which the total distance traveled by the vehicle (miles) can be determined.

Logical Architecture Reference Flow: fcv_driver_safety_status
This data flow is used within the Manage Commercial Vehicle function. It contains data which on-board commercial vehicle sensors can use to determine the following: driver_state - the ability of the driver to control the vehicle, negative factors being such things as alcohol on the breath, too many mistakes, etc. injuries - any detectable problems with the vehicle occupants, e.g. sudden change in heart rate, pulse, breathing, etc. The data flow has been sized at 4 bytes per sensor with 8 sensors, giving a total of thirty two (32) bytes.

Logical Architecture Reference Flow: fcv_driver_status
This data flow is used within the Manage Commercial Vehicle function. It contains analog data from on-board commercial vehicle sensors from which the current driver status can be determined.

Logical Architecture Reference Flow: fcv_lock_tag_data
This data flow is sent from a commercial vehicle to the Manage Commercial Vehicles function. It contains the current status of the lock tag that is used to control access to commercial vehicle cargoes that are being taken across borders, and is sent in response to an earlier request.

Logical Architecture Reference Flow: fcv_vehicle_safety_status
This data flow is used within the Manage Commercial Vehicle function. It contains analog data from on-board commercial vehicle sensors from which the extent of a vehicle's collision damage can be determined.

Logical Architecture Reference Flow: fcv_weight
This data flow is used within the Manage Commercial Vehicle function. It contains analog data from on-board commercial vehicle sensors from which the vehicle's gross weight can be determined in tons (US). Size is 4 bytes per sensor, with two sensors.

Commercial Vehicle Check => Commercial Vehicle Subsystem

Physical Architecture Flow Name: border clearance event record

Results of border clearance check.

Logical Architecture Reference Flow: cv_on_board_border_record
This data flow is used within the Manage Commercial Vehicles function. It contains the results of the of the border clearance checks at a commercial vehicle roadside border crossing facility. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_on_board_border_details.

Physical Architecture Flow Name: clearance event record

Results of vehicle clearance activity.

Logical Architecture Reference Flow: cv_on_board_screening_record
This data flow is used within the Manage Commercial Vehicles function and contains the results of the screening of a commercial vehicle at a particular commercial vehicle roadside checking facility. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_on_board_screening_details.

Physical Architecture Flow Name: electronic clearance request

Request for electronic clearance data (Toll, safety, customs, etc.).

Logical Architecture Reference Flow: cv_request_electronic_clearance_data
This data flow is used within the Manage Commercial Vehicles function. It contains a request from a commercial

vehicle roadside checkstation or border crossing facility for the output of the current contents of a commercial vehicle's tag, plus the cargo lock tag, if one is being carried by the vehicle.

Physical Architecture Flow Name: lock tag data request

Request to supply lock information on cargo lock for retransmission to international border crossing station.

Logical Architecture Reference Flow: cv_request_electronic_clearance_data
This data flow is used within the Manage Commercial Vehicles function. It contains a request from a commercial vehicle roadside checkstation or border crossing facility for the output of the current contents of a commercial vehicle's tag, plus the cargo lock tag, if one is being carried by the vehicle.

Physical Architecture Flow Name: on-board safety request

Request for onboard vehicle safety data.

Logical Architecture Reference Flow: cv_request_on_board_data
This data flow is used within the Manage Commercial Vehicles function by the commercial vehicle roadside processing to request on-board data from a commercial vehicle. It contains the following data items each of which is defined in its own DDE: cv_roadside_facility_identity + cv_roadside_facility_address.

Physical Architecture Flow Name: pass/pull-in

Command to commercial vehicle to pull into inspection station.

Logical Architecture Reference Flow: cv_on_board_pull_in_output
This data flow is used in the Manage Commercial Vehicles function and contains the output result of the commercial vehicle safety or screening processes, in terms of a pull-in or pass decision for the vehicle, or a general pull-in pass decision for all vehicles, or a pull-in decision based on a problem with reading the vehicle's tag. It consists of the following data items, each of which is defined in its own DDE: [cv_general_pull-in | cv_screening_pull_in_output | cv_safety_pull_in_output | cv_border_pull_in_output].

Physical Architecture Flow Name: safety inspection record

Record containing results of commercial vehicle safety inspection.

Logical Architecture Reference Flow: cv_inspection_data_output
This data flow is used within the Manage Commercial Vehicle function and contains the results of the commercial vehicle roadside inspection. These are down loaded for storage on-board the vehicle.

Physical Architecture Flow Name: screening request

Request for screening data based on vehicle and possibly cargo's tags.

Logical Architecture Reference Flow: cv_request_electronic_clearance_data
This data flow is used within the Manage Commercial Vehicles function. It contains a request from a commercial vehicle roadside checkstation or border crossing facility for the output of the current contents of a commercial vehicle's tag, plus the cargo lock tag, if one is being carried by the vehicle.

Commercial Vehicle Driver => **Commercial Vehicle Subsystem**

Physical Architecture Flow Name: CVO driver initialization

Commercial vehicle driver and vehicle information and requests to the commercial vehicle managing system.

Logical Architecture Reference Flow: fcvd_activity_request
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains an activity number which may be one of the following: 1 - request route, 2 - request preclearance, 3 - store route, 4 - provide list of stored routes 5 - display route details 6 - delete route. Values of three (3) and four (4) must be accompanied by a route number.

Logical Architecture Reference Flow: fcvd_carrier_number
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains the commercial vehicle carrier identification number to be entered into the commercial vehicle's tag data store.

Logical Architecture Reference Flow: fcvd_driver_data_input
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and includes the driver's log data as required by state and federal agencies as well as requested data for the previously input data type. This data is to be used by the on-board vehicle data collection system.

Logical Architecture Reference Flow: fcvd_driver_general_message
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains a general message for output to the commercial vehicle fleet manager as part of the on-board vehicle data. Its size will be restricted in a way that will depend upon the mechanism by which the data is input, e.g. time (1 minute?) for voice input, number of characters (512?) for typed input.

Logical Architecture Reference Flow: fcvd_driver_input_type
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains an indication of the type of data that the driver wants to input to the on-board vehicle data collection system. This type may be one of the following: 1 - cargo_data, 2 - repairs and service records, 3 - inspection and maintenance data, 4 - fuel purchase data, 5 - driver identity, 6 - driver_credentials, 7 - driver_license_citations, 8 - carrier_identity, 9 - data_store_contents.

Logical Architecture Reference Flow: fcvd_driver_number
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains the commercial vehicle driver identification number to be entered into the commercial vehicle's tag data store.

Logical Architecture Reference Flow: fcvd_enrollment_payment_request
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains a request for payment of the taxes and duties needed to cover the enrollment of a particular class of vehicle and cargo at a particular weight on a particular route. The data will include the account number from which the costs are to be deducted (12 bytes) and the route number (4 bytes) to which they apply. Drivers will make this request because they are acting on the role of their own fleet managers, i.e. they will be owner drivers.

Logical Architecture Reference Flow: fcvd_enrollment_request
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains a request for the enrollment of a particular class of vehicle (1 byte) and cargo (2 bytes) at a particular weight (6 bytes) on a particular route (4 bytes). Drivers will make this request because they are acting on the role of their own fleet managers, i.e. they will be owner/operators.

Logical Architecture Reference Flow: fcvd_other_data_input
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains the response to an earlier request for input of other data. Drivers will make this request because they are acting on the role of their own fleet managers, i.e. they will be owner drivers.

Logical Architecture Reference Flow: fcvd_request_routing_instructions
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains a request for instructions on which route to take and any cargo that must be picked up or dropped off at the origin, destination and/or intermediate points along the route. This data does not include any route guidance instructions as the

driver will be able to obtain this data through the Provide Driver and Traveler Services function when the route origin, destination and intermediate points are known.

Logical Architecture Reference Flow: fcvd_request_tag_data_output
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function. It contains a request for the output of the current data that is stored on a commercial vehicle's two tag. Only the data that can be written by the manager is output.

Logical Architecture Reference Flow: fcvd_route_data
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains data input in response to a previous request for the input of route data. Drivers will make this request because they are acting on the role of their own fleet managers, i.e. they will be owner drivers.

Logical Architecture Reference Flow: fcvd_route_request
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains a request for a route to be provided. Drivers will make this request because they are acting on the role of their own fleet managers, i.e. they will be owner drivers.

Logical Architecture Reference Flow: fcvd_trip_identity
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains the commercial vehicle trip identification number to be entered into the vehicle's tag data store.

Logical Architecture Reference Flow: fcvd_vehicle_number
This data flow is sent from the commercial vehicle driver to the Manage Commercial Vehicles function and contains the commercial vehicle identification number to be entered into the commercial vehicle's tag data store.

Commercial Vehicle Subsystem => Commercial Vehicle

Physical Architecture Flow Name: lock tag data request

Request to supply lock information on cargo lock for retransmission to international border crossing station.

Logical Architecture Reference Flow: tcv_lock_tag_data_request
This data flow is sent to a commercial vehicle from the Manage Commercial Vehicles function. It contains a request for the provision of the current status of the lock tag that is used to control access to commercial vehicle cargoes that are being taken across borders.

Commercial Vehicle Subsystem => Commercial Vehicle Check

Physical Architecture Flow Name: electronic clearance data

Information required for electronic clearance (toll, safety, customs, etc.).

Logical Architecture Reference Flow: cv_electronic_clearance_data
This data flow is used within the Manage Commercial Vehicles and Provide Electronic Payment Services functions. It contains data that has been stored on a commercial vehicle tag to enable its identification at commercial vehicle roadside checkstation facilities for the purposes of electronic clearance, and/or safety inspection, and/or border clearance, as well as at toll plazas for toll payment collection. The data flow consists of the following data items each of which is defined in its own DDE: cv_on_board_tag_data + cv_lock_tag_data.

Physical Architecture Flow Name: lock tag data

Tag information on cargo lock.

Logical Architecture Reference Flow: cv_electronic_clearance_data
This data flow is used within the Manage Commercial Vehicles and Provide Electronic Payment Services functions. It contains data that has been stored on a commercial vehicle tag to enable its identification at commercial vehicle roadside checkstation facilities for the purposes of electronic clearance, and/or safety inspection, and/or border clearance, as well as at toll plazas for toll payment collection. The data flow consists of the following data items each of which is defined in its own DDE: cv_on_board_tag_data + cv_lock_tag_data.

Physical Architecture Flow Name: on board safety data

Vehicle safety data measured by vehicle sensors and sent to inspection stations

Logical Architecture Reference Flow: cv_on_board_data
This data flow is used within the Manage Commercial Vehicles function to send on-board commercial vehicle data from the vehicle to a commercial vehicle roadside facility. It contains the following data items each of which is defined in its own DDE: cv_identity_details + cv_credentials + cv_driver_credentials + cv_driver_license_citations + cv_repairs_and_service_records + cv_inspection_data + cv_inspection_activities_data + cv_fuel_purchase_data + cv_safety_systems_diagnostics_results + cv_vehicle_log + cv_log_data + cv_not_pulled_in.

Physical Architecture Flow Name: screening data

Data stored in vehicle's tag allowing electronic clearance at border crossings, debits at toll plazas, and clearance at safety inspections.

Logical Architecture Reference Flow: cv_electronic_clearance_data
This data flow is used within the Manage Commercial Vehicles and Provide Electronic Payment Services functions. It contains data that has been stored on a commercial vehicle tag to enable its identification at commercial vehicle roadside checkstation facilities for the purposes of electronic clearance, and/or safety inspection, and/or border clearance, as well as at toll plazas for toll payment collection. The data flow consists of the following data items each of which is defined in its own DDE: cv_on_board_tag_data + cv_lock_tag_data.

Commercial Vehicle Subsystem => **Commercial Vehicle Driver**

Physical Architecture Flow Name: alerts, messages

Specific alerts and messages related to Commercial Vehicles (e.g. trucks not advised, trucks over 10 tons not allowed on bridge, route details)

Logical Architecture Reference Flow: tcvd_critical_safety_problem
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicle function and contains details of any commercial vehicle on-board safety problems which have been detected by processes within the function.

Logical Architecture Reference Flow: tcvd_data_request
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicles function and contains a request for the input of additional data to allow a previously requested action to be implemented. Drivers will receive this data in response to previous data input because they are acting on the role of their own fleet managers, i.e. they will be owner drivers.

Logical Architecture Reference Flow: tcvd_on_board_pull_in_output
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicle function. It represents the message to be sent to the driver of a vehicle that is being pulled in to a commercial vehicle roadside checking facility. This may be because problems with credentials, safety, or reading data from the tag of a particular vehicle, or a general pull-in request for all vehicles. It may also be output by a commercial vehicle roadside border crossing facility due to a problem with clearing a vehicle through a border crossing check point. The data will be output directly to the driver in

the vehicle cab and not using any mechanism external to the vehicle.

Logical Architecture Reference Flow: tcvd_route_data
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicles function and contains details of a route for a commercial vehicle, together with the location and type of each roadside facility along the route. Drivers will receive this data in response to previous data input because they are acting on the role of their own fleet managers, i.e. they will be owner drivers.

Physical Architecture Flow Name: CVO Pull in message

Message sent to commercial vehicle driver requesting pull in to inspection/verification stop along with inspection results.

Logical Architecture Reference Flow: tcvd_type_input_request
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicle function and contains the request for the type of data to be input to be provided by the driver.

Physical Architecture Flow Name: log information

Request information to be entered into the driver log.

Logical Architecture Reference Flow: tcvd_confirm_data_stored
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicles function and contains confirmation that the previously entered data has been stored in the on-board vehicle unit.

Logical Architecture Reference Flow: tcvd_data_input_request
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicles function and contains the request for the driver to input data for the type of data previously requested. It is part of a data input dialogue that the driver has with the vehicle's on-board data collection system.

Logical Architecture Reference Flow: tcvd_enrollment_confirmation
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicles function and contains confirmation that a request for the enrollment of a particular class of vehicle and cargo at a particular weight on a particular route has been accepted and includes a list of the required taxes and duties together with their costs. Drivers will receive this data in response to previous data input because they are acting on the role of their own fleet managers, i.e. they will be owner drivers.

Logical Architecture Reference Flow: tcvd_enrollment_payment_confirmation
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicles function and contains confirmation that a payment for the enrollment of a particular class of vehicle and cargo at a particular weight on a particular route has been accepted. Drivers will receive this data in response to previous data input because they are acting on the role of their own fleet managers, i.e. they will be owner drivers.

Logical Architecture Reference Flow: tcvd_other_data_request
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicles function and contains the identities of data items needed to complete all the data needed for a route to be stored in the data store, of commercial vehicle routes, but which have yet to be provided by the driver. Drivers will receive this data in response to previous data input because they are acting on the role of their own fleet managers, i.e. they will be owner drivers.

Logical Architecture Reference Flow: tcvd_output_data
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicle function and contains the data output previously requested by a commercial vehicle driver.

Logical Architecture Reference Flow: tcvd_output_tag_data
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicles function. It contains the

output of the current contents of a commercial vehicle's type two tag, produced in response to a previous request from the driver.

Logical Architecture Reference Flow: tcvd_routing_instructions
This data flow is sent to the commercial vehicle driver from the Manage Commercial Vehicles function and contains details of a route for a commercial vehicle, together with instructions about cargo that is to be picked up and/or dropped off at the origin, destination and/or intermediate points along the route.

Commercial Vehicle Subsystem => Fleet and Freight Management

Physical Architecture Flow Name: driver and vehicle information

Requests from the driver and vehicle for routing, payment, and enrollment information.

Logical Architecture Reference Flow: cf_driver_route_instructions_request
This data flow is used within the Manage Commercial Vehicles function and contains a request from the commercial vehicle driver for output of the driver route instructions. It consists of the following data items each of which is defined in its own DDE: cv_driver_number + cv_route_number.

Logical Architecture Reference Flow: cv_driver_enrollment_payment_request
This data flow is used within the Manage Commercial Vehicles function and contains data required to enable payment for enrollment of a commercial vehicle for the use of a particular route as provided by the commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cv_account_number + cv_driver_credit_identity + cv_route_number.

Logical Architecture Reference Flow: cv_driver_enrollment_request
This data flow is used within the Manage Commercial Vehicles function and contains data required for the enrollment of a commercial vehicle on a particular route as provided by a commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cv_cargo_class + cv_route_number + cv_trip_identity + cv_vehicle_class + cv_weight_class.

Logical Architecture Reference Flow: cv_driver_route_request
This data flow is used within the Manage Commercial Vehicles function by the commercial vehicle driver to request a commercial vehicle route. It contains the following data items each of which is defined in its own DDE: trip_request + route_type.

Logical Architecture Reference Flow: cv_driver_storage_request
This data flow is used within the Manage Commercial Vehicles function to manage the store of routes used by the commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cv_storage_action_flag + cv_route_number.

Logical Architecture Reference Flow: cv_static_route_data
This data flow is used within the Manage Commercial Vehicles function. It contains the data for a static based route provided as a result of data provided by the commercial vehicle driver acting as a fleet manager.

Physical Architecture Flow Name: on board vehicle data

Condition of the commercial vehicle sent to commercial vehicle manager primarily for maintenance purposes.

Logical Architecture Reference Flow: cf_on_board_vehicle_data
This data flow is used within the Manage Commercial Vehicle function and contains data collected on-board a commercial vehicle output of which has been requested by the commercial vehicle manager. It consists of the following data items each of which is defined in its own DDE: cv_on_board_data + cv_general_output_message + vehicle_location_for_cv.

Logical Architecture Reference Flow: cf_tag_data_store_output
This data flow is used within the Manage Commercial Vehicles function. It contains the output of the data currently being held by a type two commercial vehicle tag as previously requested by the commercial vehicle manager. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_trip_identity.

Commercial Vehicle Subsystem => Vehicle

Physical Architecture Flow Name: commercial vehicle data

Information about the commercial vehicles cargo, credentials, and payments.

Logical Architecture Reference Flow: cv_driver_enrollment_cost
This data flow is sent from the Manage Commercial Vehicles function to the Provide Electronic Payment Services function. It contains the cost of the electronic credential filing and taxes, payment of which was previously requested by the commercial vehicle driver acting in the role of fleet manager, and is only sent when the cost is to be deducted from the credit stored on the payment instrument being used by the driver. The data flow consists of the following data items each of which is defined in its own DDE: cv_amount_billed + stored_credit.

Logical Architecture Reference Flow: processed_cargo_data
This data flow is used within the Provide Vehicle Monitoring and Control function. It contains data obtained from the processing by sensors of analog data received on-board the vehicle about the composition and state of its cargo.

Fleet and Freight Management => Commercial Vehicle Subsystem

Physical Architecture Flow Name: fleet to driver update

Updated instructions to the driver including dispatch, routing, and special instructions

Logical Architecture Reference Flow: cf_driver_route_instructions
This data flow is used within the Manage Commercial Vehicles function and contains the driver route and loading instructions related to a particular route and driver for output to the commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cf_driver_route + cf_driver_load_details.

Logical Architecture Reference Flow: cf_request_on_board_vehicle_data
This data flow is used within the Manage Commercial Vehicles function. It contains a request from the commercial vehicle manager for the commercial vehicle to output the on-board data it has collected, plus any general message data from the driver. The data flow consists of the following data items each of which is defined in its own DDE: cv_on_board_data_required + cv_general_input_message.

Logical Architecture Reference Flow: cf_tag_data_store_request
This data flow is used within the Manage Commercial Vehicles function. It contains a request for the current on-board commercial vehicle tag data to be sent to the process that provides the interface with the commercial vehicle manager. This data flow is sent in response to a request by the manager for the output of the current tag data.

Logical Architecture Reference Flow: cf_tag_data_store_write
This data flow is used within the Manage Commercial Vehicles function. It contains on-board commercial vehicle tag data that is loaded by the commercial vehicle manager and is used by other processes in the function. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_trip_identity + tag_identity.

Logical Architecture Reference Flow: cv_driver_enrollment_information
This data flow is used within the Manage Commercial Vehicles function and contains data about the taxes and duties

required for a commercial vehicle to be enrolled for a particular route as provided by the commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cv_route_number + cv_taxes_and_duties.

Logical Architecture Reference Flow: cv_driver_enrollment_payment_confirmation

This data flow is used within the Manage Commercial Vehicles function to confirm that a payment of the taxes and duties for the enrollment of a particular commercial vehicle cargo, weight and type on a particular route from the commercial vehicle driver has been accepted. It consists of the following data items each of which is defined in its own DDE: cv_account_number + cv_amount_billed + cv_driver_credit_identity + cv_route_number.

Logical Architecture Reference Flow: cv_driver_route_data

This data flow is used within the Manage Commercial Vehicle function and contains data about a vehicle route requested by a commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cv_route_data + route_type + cv_route_number.

Logical Architecture Reference Flow: cv_static_route_request

This data flow is used within the Manage Commercial Vehicles function. It contains the data from which a static route can be determined for a commercial vehicle and is supplied by the commercial vehicle driver acting as a fleet manager. The data flow consists of the following data items each of which is defined in its own DDE: constraint_on_acceptable_travel_time + constraint_on_ahs_lanes + constraint_on_eta_change + constraint_on_interstate + constraint_on_load_classification + constraint_on_urban + constraint_on_vehicle_type + destination + departure_time + desired_arrival_time + modes + origin + preferred_alternate_routes + preferred_route_segments + preferred_routes + preferred_weather_conditions.

Vehicle => **Commercial Vehicle Subsystem**

Physical Architecture Flow Name: commercial vehicle data request

Request for commercial vehicle information (cargo, driver's credit, vehicle location).

Logical Architecture Reference Flow: cargo_data_request

This data flow is used within the Provide Vehicle Monitoring and Control function. It contains a request for data about a vehicle's cargo including any damage report to be sent back in reply.

Logical Architecture Reference Flow: cv_driver_credit_identity

This data flow is sent from the Provide Electronic Payments Services function to the Manage Commercial Vehicles function. It contains the credit identity of a commercial vehicle driver or the amount of stored credit obtained from the payment instrument terminator and consists of the following data items each of which is defined in its own DDE: credit_identity + stored_credit.

Logical Architecture Reference Flow: vehicle_location_for_cv

This data flow is sent from the Provide Driver and Traveler Services function to the Manage Commercial Vehicles function. It contains the vehicle location data for use in reporting on-board data to the commercial fleet manager, or the driver acting in this role. The data is computed from data input to sensors controlled by the processes that determines vehicle location. The data flow consists of the following data item which is defined in its own DDE: location_identity.

2.6.4 Subsystem Architecture Flow Diagram

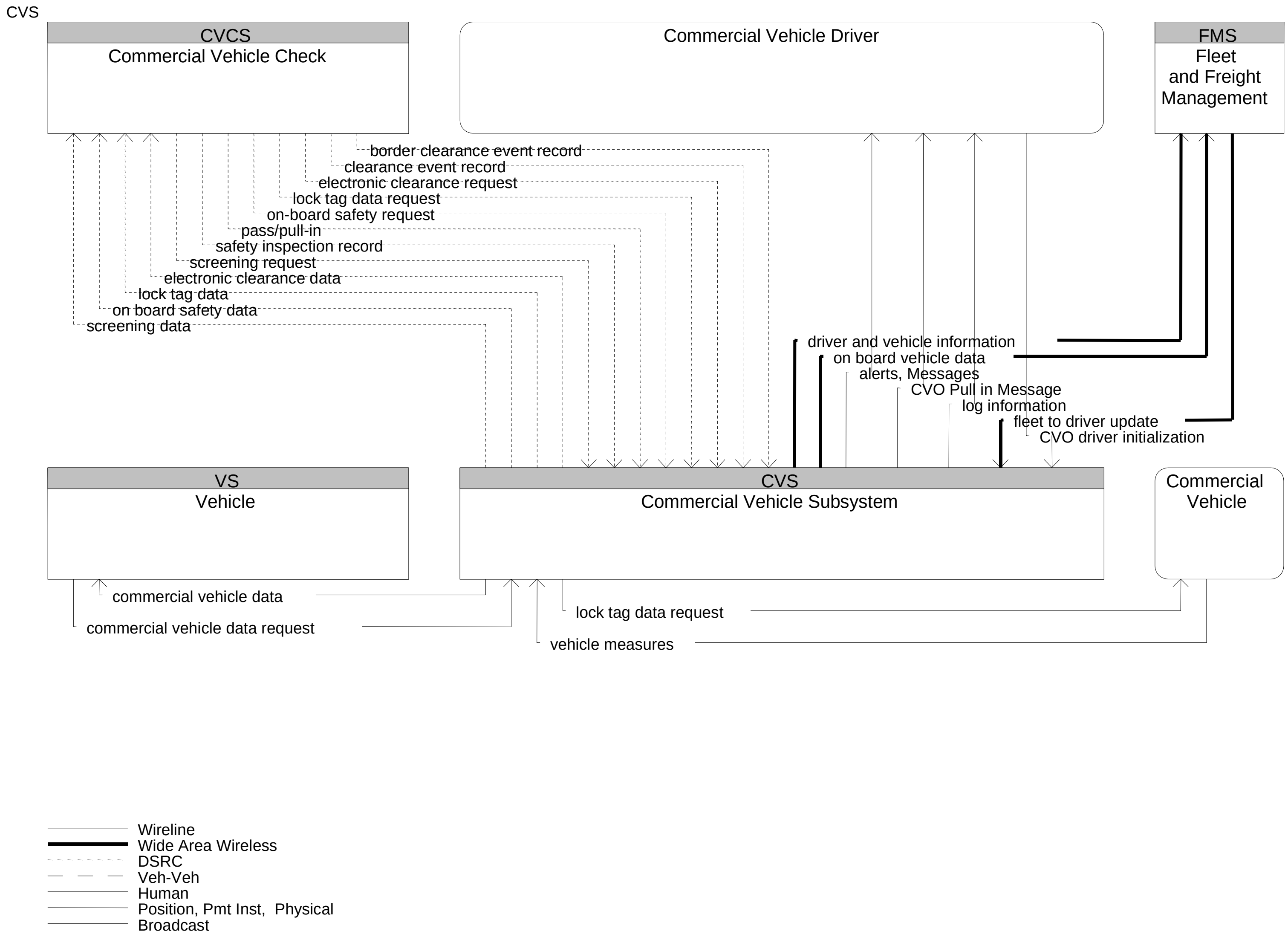


Figure 2.6-2 Architecture Flow Diagram for CVS

2.7 Emergency Management

This subsystem resides in a commercial vehicle and provides the sensory, processing, storage, and communications functions necessary to support safe and efficient commercial vehicle operations. The Commercial Vehicle Subsystem provides two-way communications between the commercial vehicle drivers, their fleet managers, and roadside officials, and provides HAZMAT response teams with timely and accurate cargo contents information after a vehicle incident. This subsystem provides the capability to collect and process vehicle, cargo, and driver safety data and status and alert the driver whenever there is a potential safety problem. Basic identification and safety status data are supplied to inspection facilities at mainline speeds. In addition, the subsystem will automatically collect and record mileage, fuel usage, and border crossings.

2.7.1 Alternative Configurations

The Emergency Management Subsystem may represent any collection of Public Safety Agencies or Private concerns dealing with Public Safety. As illustrated in Figure 2.7-1, the Emergency Management Subsystem provides the capability to manage emergency vehicles and exchange information with other such agencies. It also provides an interface between existing emergency telephone support and E911 centers. The emergency management subsystem may also be combined in the same location as a traffic management subsystem to provide integrated incident management.

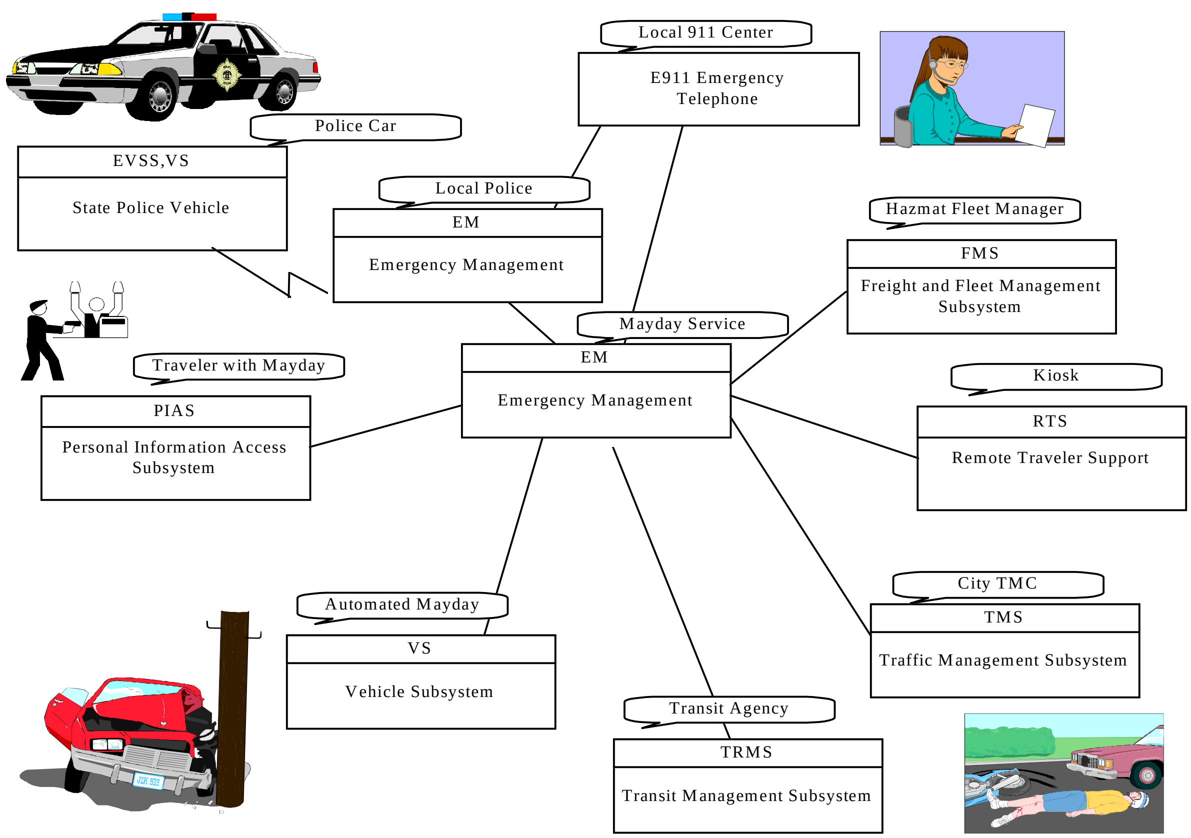


Figure 2.7-1 Alternative Configurations of the Emergency Management Subsystem

2.7.2 Subsystem Equipment Packages and Supporting Process Specifications for EM

Emergency Call-Taking

This Equipment package supports the emergency call-taker, collecting available information about the caller and the reported emergency, and forwarding this information to other equipment packages that formulate and manage the emergency response. This equipment package receives 9-1-1, 7-digit local access, and motorist call-box calls and interfaces to other agencies to assist in the verification and assessment of the emergency and to forward the emergency information to the appropriate response agency.

Process Specifications

5.1.1 Identify Emergencies from Inputs

Overview: This process shall enable existing emergency centers to receive the calls, determine response requirements (enough to determine what responding agencies to notify), and route distress calls to those predesignated responding agencies. This process shall provide the identified emergency information in a standard format as required. Every set of emergency data received shall be assigned a level of confidence by the process depending on its source, so that the subsequent processes can assess the level of response to be provided. This process shall include verification, in that it shall determine if a number of inputs might all be referring to the same incident, then designate that incident in its notifications to responding agencies. By reconciling numerous reports and other collaborative information from the field (e.g., CCTV images, reports from field staff), the verification function confirms the existence, location, and nature of a reported emergency.

5.1.3 Communicate Emergency Status

Overview: This process shall receive the emergency service response plans and the status of their implementation for dissemination to other ITS functions. That dissemination shall be subject to sanitation according to pre-arranged rules, implemented in this process. The process shall also read data about emergency responses from the emergency services action log. All data shall be communicated by the process in standard formats to travelers, drivers, and other ITS functions. In the case of in-vehicle, personal traveler, and transit emergencies, after each emergency becomes a verified incident, the data shall be sent as soon as new status or plan data is received. Dissemination shall be controlled according to rules determined in this process to limit the information transmitted to that information useful to the receiver. Emergency information that is received from the emergency telephone system or E911 operators, shall be disseminated only when the response plan data is first received. That has the effect of only disseminating data on incidents that have been verified, since only verified incidents will have response plans. The process shall also extract data from the emergency service action log on request from processes in other ITS functions, and from the emergency services operator.

5.2 Provide Operator Interface for Emergency Data

Overview: This process shall provide the emergency services operator with an interface to the other processes in the Manage Emergency Services function. The process shall enable the operator to review and update the data used to allocate emergency services to incidents, applying temporary overrides to current emergency service allocations to suit the special needs of a current incident, and requesting output of the log of emergency service actions. It shall also enable the output of a message showing the failure of an emergency vehicle dispatched in response to an incident. This output shall override all other outputs. The process shall support inputs from the emergency services operator in both manual and audio form, and shall provide its outputs in audible and visual forms. The visual output may appear in either hardcopy or as a display, or both, and an audible output shall accompany the emergency vehicle dispatch failure message.

Emergency Dispatch

This Equipment package supports efficient dispatch of emergency vehicles. It tracks emergency vehicles, dispatches these vehicles to an incident, and provides safe and efficient routes based on real-time traffic information.

Process Specifications

5.2 Provide Operator Interface for Emergency Data

Overview: This process shall provide the emergency services operator with an interface to the other processes in the Manage Emergency Services function. The process shall enable the operator to review and update the data used to allocate emergency services to incidents, applying temporary overrides to current emergency service allocations to suit the special needs of a current incident, and requesting output of the log of emergency service actions. It shall also enable the output of a message showing the failure of an emergency vehicle dispatched in response to an incident. This output shall override all other outputs. The process shall support inputs from the emergency services operator in both manual and audio form, and shall provide its outputs in audible and visual forms. The visual output may appear in either hardcopy or as a display, or both, and an audible output shall accompany the emergency vehicle dispatch failure message.

5.3.2 Dispatch Vehicle

Overview: This process shall direct selected emergency vehicles and drivers to respond to an incident, receive acknowledgement that they will in fact respond, and provide them with the location and details of the incident. The data about the incident shall be received from the process in the Manage Emergency Services function responsible for selecting the vehicles for the response. If called for, the process shall also request emergency vehicle routing from the Provide Driver and Traveler Services function and send details to the Manage Traffic function to request a (green wave) be provided for the vehicle(s) if that mode of priority is available and chosen. The data for the emergency vehicle driver shall be sent to the driver interface process.

5.3.6 Maintain Vehicle Status

Overview: This process shall maintain a data store of the current status of all emergency vehicles available for dispatch and that have been dispatched. It shall provide data from the store on request from other processes and shall update the contents of the store with new data received from other processes. The process shall output the status of a vehicle to the process responsible for vehicle tracking for as long as it is on its way to an incident, to update ETA estimates and enable local vehicle priority to be given at intersections, if that mode of priority is chosen and granted.

5.5 Update Emergency Display Map Data

Overview: This process shall provide updates to the store of digitized map data used as the background for displays of incidents and emergencies produced by processes in the Manage Emergency Services function. The process shall obtain the new data from a specialist data supplier or some other appropriate data source, on receiving an update request from the emergency system operator interface process within the Manage Emergency Services function.

Emergency Response Management

This Equipment package develops and stores emergency response plans and manages overall coordinated response to emergencies. It tracks the availability of resources and assists in the appropriate allocation of these resources for a particular emergency response. This Equipment package provides coordination between multiple allied agencies before and during emergencies to implement emergency response plans and track progress through the incident. It provides vital communications linkages which provide real-time information to emergency response personnel in the field.

Process Specifications

5.1.2 Determine Coordinated Response Plan

Overview: This process shall determine the appropriate response for a verified emergency. The process shall classify and prioritize the emergency and use this assessment to determine an appropriate response plan. A detailed description of the emergency and the suggested response plan shall be sent to other processes for implementation. The same information shall also be forwarded to other emergency centers (other EM) for information and possible action.

5.1.3 Communicate Emergency Status

Overview: This process shall receive the emergency service response plans and the status of their implementation for dissemination to other ITS functions. That dissemination shall be subject to sanitation according to pre-arranged rules, implemented in this process. The process shall also read data about emergency responses from the emergency services action log. All data shall be communicated by the process in standard formats to travelers, drivers, and other ITS functions. In the case of in-vehicle, personal traveler, and transit emergencies, after each emergency becomes a verified incident, the data shall be sent as soon as new status or plan data is received. Dissemination shall be controlled according to rules determined in this process to limit the information transmitted to that information useful to the receiver. Emergency information that is received from the emergency telephone system or E911 operators, shall be disseminated only when the response plan data is first received. That has the effect of only disseminating data on incidents that have been verified, since only verified incidents will have response plans. The process shall also extract data from the emergency service action log on request from processes in other ITS functions, and from the emergency services operator.

5.1.4 Manage Emergency Response

Overview: This process shall enable existing emergency centers to receive emergency calls, determine response requirements to the extent necessary to route the information, and route distress calls and emergency information to predesignated responding agencies and vehicles. All identified emergency information shall be provided by the process in a standard format as required. The process shall also communicate with commercial fleet managers to obtain details of cargo and other vehicle data where this will affect the response of the emergency services, e.g., in the case of a vehicle carrying a HAZMAT load. The current status of all emergency service responses shall be stored by the process in an action log, for access by the communications process.

5.1.5 Manage Emergency Service Allocation Store

Overview: This process shall manage the store of data that defines the way in which the emergency service resources shall be deployed in response to emergencies. Deployment shall vary by certain criteria, such as, type of emergency, source of information, time of day, location, etc. Parameters to define this allocation shall be loaded into the data store following receipt from the process that provides the emergency services operator interface.

5.2 Provide Operator Interface for Emergency Data

Overview: This process shall provide the emergency services operator with an interface to the other processes in the Manage Emergency Services function. The process shall enable the operator to review and update the data used to allocate emergency services to incidents, applying temporary overrides to current emergency service allocations to suit the special needs of a current incident, and requesting output of the log of emergency service actions. It shall also enable the output of a message showing the failure of an emergency vehicle dispatched in response to an incident. This output shall override all other outputs. The process shall support inputs from the emergency services operator in both manual and audio form, and shall provide its outputs in audible and visual forms. The visual output may appear in either hardcopy or as a display, or both, and an audible output shall accompany the emergency vehicle dispatch failure message.

5.3.1 Select Response Mode

Overview: This process shall select the appropriate emergency services and their vehicle(s) to respond to incidents. The process shall determine the type and number of vehicles to be dispatched, and provide the vehicle(s) with information on the type and location of the incident. It shall request data about the vehicles that are available from the interface process to the data store of emergency vehicle status. Once the vehicle determination has been made, the status data shall be changed by the process, and incident data sent to the process responsible for the actual dispatch of the vehicle(s).

5.3.4 Assess Response Status

Overview: This process shall assess the status of emergency vehicles that are responding to an incident. In making its assessment, the process shall use data from the process managing a store of vehicle status, plus data from the emergency vehicle driver interface process. The process shall send the results of the assessment to the process responsible for managing emergency and emergency response information and update the store of vehicle status.

5.5 Update Emergency Display Map Data

Overview: This process shall provide updates to the store of digitized map data used as the background for displays of incidents and emergencies produced by processes in the Manage Emergency Services function. The process shall obtain the new data from a specialist data supplier or some other appropriate data source, on receiving an update request from the emergency system operator interface process within the Manage Emergency Services function.

Mayday Support
This Equipment package receives Mayday messages, determines an appropriate response, and either uses internal resources or contacts a local agency to provide that response. The nature of the emergency is determined based on the information in the mayday message as well as other inputs. This package effectively serves as an interface between automated mobile mayday systems and the local public safety answering point for messages which require a public safety response.

Process Specifications

5.1.1 Identify Emergencies from Inputs

Overview: This process shall enable existing emergency centers to receive the calls, determine response requirements (enough to determine what responding agencies to notify), and route distress calls to those predesignated responding agencies. This process shall provide the identified emergency information in a standard format as required. Every set of emergency data received shall be assigned a level of confidence by the process depending on its source, so that the subsequent processes can assess the level of response to be provided. This process shall include verification, in that it shall determine if a number of inputs might all be referring to the same incident, then designate that incident in its notifications to responding agencies. By reconciling numerous reports and other collaborative information from the field (e.g., CCTV images, reports from field staff), the verification function confirms the existence, location, and nature of a reported emergency.

5.1.2 Determine Coordinated Response Plan

Overview: This process shall determine the appropriate response for a verified emergency. The process shall classify and prioritize the emergency and use this assessment to determine an appropriate response plan. A detailed description of the emergency and the suggested response plan shall be sent to other processes for implementation. The same information shall also be forwarded to other emergency centers (other EM) for information and possible action.

5.1.3 Communicate Emergency Status

Overview: This process shall receive the emergency service response plans and the status of their implementation for dissemination to other ITS functions. That dissemination shall be subject to sanitization according to pre-arranged rules, implemented in this process. The process shall also read data about emergency responses from the emergency services action log. All data shall be communicated by the process in standard formats to travelers, drivers, and other ITS functions. In the case of in-vehicle, personal traveler, and transit emergencies, after each emergency becomes a verified incident, the data shall be sent as soon as new status or plan data is received. Dissemination shall be controlled according to rules determined in this process to limit the information transmitted to that information useful to the receiver. Emergency information that is received from the emergency telephone system or E911 operators, shall be disseminated only when the response plan data is first received. That has the effect of only disseminating data on incidents that have been verified, since only verified incidents will have response plans. The process shall also extract data from the emergency service action log on request from processes in other ITS functions, and from the emergency services operator.

5.2 Provide Operator Interface for Emergency Data

Overview: This process shall provide the emergency services operator with an interface to the other processes in the Manage Emergency Services function. The process shall enable the operator to review and update the data used to allocate emergency services to incidents, applying temporary overrides to current emergency service allocations to suit the special needs of a current incident, and requesting output of the log of emergency service actions. It shall also enable the output of a message showing the failure of an emergency vehicle dispatched in response to an incident. This output shall override all other outputs. The process shall support inputs from the emergency services operator in both manual and audio form, and shall provide its outputs in audible and visual forms. The visual output may appear in either hardcopy or as a display, or both, and an audible output shall accompany the emergency vehicle dispatch failure message.

2.7.3 Subsystem Interfaces for EM

Emergency Management => **Emergency System Operator**

Physical Architecture Flow Name: emergency operations status

Emergency operations data supporting a range of emergency operating positions including call taker, dispatch, and various other operations and communications center operator positions.

Logical Architecture Reference Flow: teso_emergency_action_log_output
This data flow is sent to the emergency system operator by the Manage Emergency Services function and contains the response to an operator command for output of the contents of the emergency services action log. The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Logical Architecture Reference Flow: teso_emergency_data_output
This data flow is sent to the emergency system operator by the Manage Emergency Services function and contains the response to an operator command for output of data about emergency service allocations. The data that is output may be details of a recently completed emergency services allocation, or the contents of the allocation criteria store. The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Logical Architecture Reference Flow: teso_emergency_vehicle_dispatch_failure
This data flow is sent to the emergency system operator by the Manage Emergency Services function and contains details of an emergency services vehicle dispatch that has failed. Details of the type(s) of vehicle requested and the number that has been dispatched (if any) are provided. The output may be in audio and visual forms, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Emergency Management => **Emergency Telecommunications System**

Physical Architecture Flow Name: incident notification response

Interactive acknowledgement and verification of the incident information received, requests for additional information, and general information on incident response status.

Logical Architecture Reference Flow: tets_incident_acknowledge
This data flow acknowledges the receipt of incident information, requests additional information, and provides general information on response status.

Emergency Management => **Emergency Vehicle Subsystem**

Physical Architecture Flow Name: emergency dispatch requests

Emergency vehicle dispatch instructions including incident location and available information concerning the incident.

Logical Architecture Reference Flow: emergency_vehicle_dispatch_request
This data flow is used within the Manage Emergency Services function and contains data for the request for an emergency vehicle dispatch. The data flow specifies the information needed to respond to a particular incident. It consists of the following data items which are defined in their own DDEs. Up to eight (8) may be used to enable up to eight different types of emergency vehicles to be specified for a single incident. 8{emergency_center_identity +

emergency_vehicle_identity + emergency_vehicle_status_code + time + date + emergency_vehicle_incident_details}.

Physical Architecture Flow Name: incident command information

Information that supports local management of an incident. It includes resource deployment status, hazardous material information, traffic, road, and weather conditions, evacuation advice, and other information that enables emergency personnel in the field to implement an effective, safe incident response.

Logical Architecture Reference Flow: local_decision_support

This data flow provides data that enables emergency personnel in the field to implement an effective incident response. It includes local traffic, road, and weather conditions, hazardous material information, and the current status of resources that have been allocated to an incident.

Physical Architecture Flow Name: suggested route

Suggested route for a dispatched emergency vehicle that may reflect current network conditions and the additional routing options available to en-route emergency vehicles that are not available to the general public.

Logical Architecture Reference Flow: emergency_vehicle_suggested_route

This data flow provides a suggested route for a dispatched vehicle that takes into account current traffic conditions, the current location and routes of other responding vehicles, and any special traffic control measures that are currently in effect to speed the response and increase the safety of emergency personnel. This data flow identifies the emergency vehicle and provides turn-by-turn route information .

Emergency Management => **Fleet and Freight Management**

Physical Architecture Flow Name: Hazmat information request

Request for information about a particular hazmat load.

Logical Architecture Reference Flow: cf_hazmat_request

This data flow is sent from the Manage Emergency Services function to the Manage Commercial Vehicles function and contains a request for information about hazardous materials that are being or about to be carried by commercial vehicles.

Emergency Management => **Information Service Provider**

Physical Architecture Flow Name: emergency vehicle route request

Special routing instructions and signal priority for emergency vehicles.

Logical Architecture Reference Flow: emergency_vehicle_route_request

This data flow is used by the Manage Emergency Services function to request a dynamic route for an emergency vehicle. It contains the following data items each of which is defined in its own DDE: trip_request + vehicle_identity + emergency_request.

Physical Architecture Flow Name: incident information

Notification of existence of incident and expected severity, location, time and nature of incident.

Logical Architecture Reference Flow: incident_information

This data flow is sent from the Manage Emergency Services function to the Provide Driver and Traveler Services function and contains information that has been requested about incidents. It consists of the following items each of which is defined in its own DDE: incident_number + incident_location + incident_start_time + incident_duration + incident_type + incident_severity + incident_traffic_impact.

Emergency Management => **Map Update Provider**

Physical Architecture Flow Name: map update request

Request for a map update which could include a new underlying map or map layer updates.

Logical Architecture Reference Flow: tmup_request_emergency_display_update
This data flow is sent to the map update provider from the Manage Emergency Services function. It contains a request for an update to the digitized map data for displays that can be used as the background for the output of data incidents and emergencies to the emergency system operator.

Emergency Management => **Media**

Physical Architecture Flow Name: incident information for media

Report of current desensitized incident information prepared for public dissemination through the media.

Logical Architecture Reference Flow: tm_emergency_information
This data flow provides information about current incidents. It contains the following data item which is defined in its own DDE: incident_details.

Emergency Management => **Other EM**

Physical Architecture Flow Name: incident report

Report of an identified incident including incident location, type, severity and other information necessary to initiate an appropriate incident response.

Logical Architecture Reference Flow: toec_emergency_center_identity
This data flow is sent to the other emergency centers from the Manage Emergency Services function and contains the identity of the center that is providing the local function. The details of this incident are contained in a parallel data flow. This data flow consists of the following data item which is defined in its own DDE: emergency_center_identity.

Logical Architecture Reference Flow: toec_incident_details
This data flow is sent to the other emergency centers from the Manage Emergency Services function and contains data about an incident that has been reported in the area served by the local function and therefore outside their area of operation. It consists of the following data items each of which is defined in its own DDE: incident_location + incident_number + incident_description + incident_start_time + incident_duration + incident_severity + incident_type.

Physical Architecture Flow Name: incident response coordination

Incident response procedures, resource coordination, and current incident response status that are shared between allied response agencies to support a coordinated response to incidents. This flow also coordinates a positive hand off of responsibility for all or part of an incident response between agencies.

Logical Architecture Reference Flow: toec_incident_response_coordination
This data flow supports coordination of an incident response between allied response agencies. It supports the coordination of response procedures, status and resources between agencies. It consists of the following data items which are defined in their own DDEs: incident_response_status + agency_incident_response_procedures +

incident_resource_coordination + hand_off_coordination.

Emergency Management => **Personal Information Access**

Physical Architecture Flow Name: emergency acknowledge

Acknowledge request for emergency assistance and provide additional details regarding actions and verification requirements.

Logical Architecture Reference Flow: emergency_request_personal_traveler_acknowledge
This data flow is used by the Manage Emergency Services function to confirm that the request for emergency services previously sent by the traveler has been received from a personal device and is therefore sent to the Provide Driver and Traveler Services function for output. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Emergency Management => **Planning Subsystem**

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: emergency_vehicle_operational_data
This data flow is sent from the manage Emergency Services function to the Plan System Deployment function. It contains information about the movements of emergency vehicles whilst attending incidents, and therefore shows the usage of green wave routes, the times at which the vehicles passed various points in the road and highway network etc.

Emergency Management => **Remote Traveler Support**

Physical Architecture Flow Name: emergency acknowledge

Acknowledge request for emergency assistance and provide additional details regarding actions and verification requirements.

Logical Architecture Reference Flow: emergency_request_traveler_acknowledge
This data flow is used by the Manage Emergency Services function to confirm that the request for emergency services previously sent by the traveler has been received from a kiosk or other device and is therefore sent to the Provide Driver and Traveler Services function for output. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Emergency Management => **Traffic Management**

Physical Architecture Flow Name: emergency traffic control request

Special request to preempt the current traffic control strategy in effect at one or more signalized intersections or highway segments. For example, this flow can request all signals to red-flash, request a green wave for an emergency vehicle, or request another special traffic control plan.

Logical Architecture Reference Flow: emergency_traffic_control_request
This data flow is sent from the Manage Emergency Services function to the Manage Traffic function. It contains a list of the route segments that have been provided for use by an emergency vehicle, together with the arrival time at each segment. The data will be used by the Manage Traffic function to generate changes to the current traffic management

strategy to give the emergency vehicle priority. The data flow consists of the following data items each of which is defined in its own DDE: date + list_size + list_size{route_segment_identity + route_segment_estimated_arrival_time} + time.

Physical Architecture Flow Name: incident information

Notification of existence of incident and expected severity, location, time and nature of incident.

Logical Architecture Reference Flow: incident_details
This data flow is sent from the Manage Emergency Services function to the Manage Traffic function and provides information about current incidents. It contains the following data items each of which is defined in its own DDE: incident_number + incident_location + incident_start_time + incident_duration + incident_type + incident_severity.

Physical Architecture Flow Name: incident response status

Status of the current incident response including traffic management strategies implemented at the site (e.g., closures, diversions, traffic signal control overrides).

Logical Architecture Reference Flow: incident_response_status
This data flow provides the current status of an incident response indicating site management strategies in effect, incident clearance status, the incident command structure that is in place, and points of contact.

Physical Architecture Flow Name: remote surveillance control

The control commands used to remotely operate another center's sensors or surveillance equipment so that roadside surveillance assets can be shared by more than one agency.

Logical Architecture Reference Flow: remote_video_image_control
This data flow is used within the Manage Traffic function. It is a request from the Emergency Management function to control closed circuit televisions(cctv) images of incidents that occurred on roadways.

Physical Architecture Flow Name: resource request

A request for traffic management resources to implement special traffic control measures, assist in clean up, verify an incident, etc.

Logical Architecture Reference Flow: resource_request
This data flow is used within the Manage Emergency Services function and contains data for the request for traffic management resources to implement special traffic control measures, assist in clean up, etc. It consists of the following data item which is defined in its own DDE. traffic_resource_request.

Emergency Management => Transit Management

Physical Architecture Flow Name: transit emergency coordination data

Data exchanged between centers dealing with a transit-related incident.

Logical Architecture Reference Flow: transit_incident_coordination_data
This data flow is sent from the Manage Emergency Services function to the Manage Transit function and contains information needed to deal with a transit related incident. It contains the following data items each of which is defined in its own DDE: transit_coordination_information + transit_fleet_operation_request + transit_response_to_incident.

Emergency Management => Vehicle

Physical Architecture Flow Name: emergency acknowledge

Acknowledge request for emergency assistance and provide additional details regarding actions and verification requirements.

Logical Architecture Reference Flow: emergency_request_driver_acknowledge
This data flow is used by the Manage Emergency Services function to acknowledge that the request for emergency services previously sent by the driver has been received and is therefore sent to the Provide Driver and Traveler Services function for output.

Logical Architecture Reference Flow: emergency_request_vehicle_acknowledge
This data flow is used by the Manage Emergency Services function to acknowledge that the request for emergency services previously sent automatically by the vehicle through processes in the Provide Vehicle Control and Monitoring function has been received. It is sent to the Provide Driver and Traveler Services function for output.

Emergency System Operator => **Emergency Management**

Physical Architecture Flow Name: emergency operations request

Emergency operator inputs supporting call taking, dispatch, and other operations and communications center operator functions.

Logical Architecture Reference Flow: feso_emergency_action_log_request
This data flow is sent by the emergency system operator to the Manage Emergency Services function. It contains a request for output of the emergency services action log. This records responses to all incidents that have been reported through this function from the other Emergency Management Centers, the E911 or Emergency Telephone Services, or by other ITS functions.

Logical Architecture Reference Flow: feso_emergency_allocation_override
This data flow is sent by the emergency system operator to the Manage Emergency Services function. It contains an override of the current pre-defined allocation of emergency services to respond to a particular current incident.

Logical Architecture Reference Flow: feso_emergency_data_input
This data flow is sent by the emergency system operator to the Manage Emergency Services function. It contains the operator's request for one of the following: 0 - no action, 1 - override the emergency services allocation for an incident (requires the number and type), 2 - data for the allocation criteria store, 3 - a request for output from the log, 4 - a request for output of the criteria store.

Logical Architecture Reference Flow: feso_emergency_data_output_request
This data flow is sent by the emergency system operator to the Manage Emergency Services function. It contains a request for output of the data that defines the emergency services that must be allocated for all the different types of incident that can be detected by processes within ITS functions.

Logical Architecture Reference Flow: feso_emergency_display_update_request
This data flow is sent by the emergency system operator to the Manage Emergency Services function. It contains the operator's request for an update to be obtained of the digitized map data that is used as the background for the output of incident and emergency data to the operator.

Emergency Telecommunications System => **Emergency Management**

Physical Architecture Flow Name: incident notification

The notification of an incident including its nature, severity, and location.

Logical Architecture Reference Flow: fets_caller_information

This data flow contains information about the caller including a call-back number and the caller location (as a street address, latitude/ longitude, or other reference) when available. The details of this incident are contained in a parallel data flow. The caller information is sized as a string of up to sixty four (64) alphanumeric characters (covering the case where the location is a street address).

Logical Architecture Reference Flow: fets_incident_information

This data flow contains information about an incident as reported by a caller or caller device. Thought specific data items are defined, in many cases the information will be a verbal report with various inaccuracies and omissions. It consists of the following data items each of which is defined in its own DDE: incident_location + incident_description + incident_severity + incident_type.

Emergency Vehicle Subsystem => Emergency Management

Physical Architecture Flow Name: emergency dispatch response

Request for additional emergency dispatch information (e.g., a suggested route) and provision of en-route status.

Logical Architecture Reference Flow: emergency_vehicle_dispatch_response

This data flow provides current enroute status and requests any additional current incident response status .

Physical Architecture Flow Name: emergency vehicle tracking data

The current location and operating status of the emergency vehicle.

Logical Architecture Reference Flow: emergency_vehicle_tracking_data

This data flow is within the Manage Emergency function and contains the current location of an emergency vehicle, together with the time and date to which the location applies. In addition this data flow contains operational data for the emergency vehicle. The data flow consists of the following data items each of which is defined in its own DDE: date + time + vehicle_location_for_emergency_services + emergency_vehicle_operational_data.

Physical Architecture Flow Name: incident command request

Request for resources, commands for relay to other allied response agencies, and other requests that reflect local command of an evolving incident response.

Logical Architecture Reference Flow: incident_command_request

This data flow requests resources, provides commands for relay to other allied response agencies, and includes other requests that reflect local command of an evolving incident response .

Physical Architecture Flow Name: incident status

Information gathered at the incident site that more completely characterizes the incident and provides current incident response status.

Logical Architecture Reference Flow: incident_status_update

This data flow provides detailed incident information gathered by emergency personnel at the incident site. Information could include the number and extent of injuries, identification of vehicles and people involved, specification of hazardous material, and any other information required to completely and accurately determine the scope and severity of the incident and the required response. Current response status is also provided which includes identification of the resources on site, site management strategies in effect, and current clearance status. The data flow consists of the following data items each of which is defined in its own DDE: date + emergency_vehicle_identity +

emergency_vehicle_status_code + incident_number + time + incident_status.

Fleet and Freight Management => Emergency Management

Physical Architecture Flow Name: Hazmat information

Information about a particular Hazmat load including nature of the load and unloading instructions. May also include Hazmat vehicle route and route update information

Logical Architecture Reference Flow: cf_hazmat_route_information

This data flow is sent from the Manage Commercial Vehicles function to the Manage Emergency Services function and contains information about the route about to be used or planned for a commercial vehicle that will carry hazardous materials. This information may cause the Emergency Services to raise an incident for all or part of the vehicle's route. The data flow consists of the following data items each of which is define in its own DDE: cv_route_number + route.

Logical Architecture Reference Flow: cf_hazmat_vehicle_information

This data flow is sent from the Manage Commercial Vehicles function to the Manage Emergency Services function and contains information about hazardous materials that are on-board the vehicle and details of the vehicle itself. The data flow consists of the following data items each of which is defined in its own DDE: hazmat_load_data + hazmat_vehicle_data.

Information Service Provider => Emergency Management

Physical Architecture Flow Name: emergency vehicle route

Routing for emergency vehicle including greenwave paths.

Logical Architecture Reference Flow: emergency_vehicle_route

This data flow is sent from the Manage Emergency Services function to the Manage Traffic function. It contains details of the emergency vehicle's route and is used to trigger a special green wave route for the emergency vehicle. The data flow consists of the following items each of which is described in its own DDE: route + emergency_request + vehicle_identity.

Physical Architecture Flow Name: incident information request

Request for incident information, clearing time, severity. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Logical Architecture Reference Flow: incident_information_request

This data flow is sent from the Provide Driver and Traveler Services function to the Manage Emergency Services function and is a request for information about incidents It contains the following data items each of which is defined in its own DDE: incident_type + incident_oldest_time.

Map Update Provider => Emergency Management

Physical Architecture Flow Name: map updates

Map update which could include a new underlying static or real-time map or map layer(s) update.

Logical Architecture Reference Flow: fmup_emergency_display_update

This data flow is sent from the map update provider to the Manage Emergency Services function. It contains the digitized map data for displays that can be used as the background for the output of data on incidents and emergencies to the emergency system operator.

Media => **Emergency Management**

Physical Architecture Flow Name: media information request

Request from the media for current transportation information.

Logical Architecture Reference Flow: fm_emergency_information_request

This data flow contains a request for data on emergencies to be sent to the Media. The request must specify the type and severity of emergency desired to be reported on, and the geographic area(s) to which it should relate.

Other EM => **Emergency Management**

Physical Architecture Flow Name: incident report

Report of an identified incident including incident location, type, severity and other information necessary to initiate an appropriate incident response.

Logical Architecture Reference Flow: foec_emergency_center_identity

This data flow is sent from the other emergency centers to the Manage Emergency Services function and contains the identity of an emergency center that has reported an incident. The details of this incident are contained in a parallel data flow. This data flow consists of the following data item which is defined in its own DDE: emergency_center_identity.

Logical Architecture Reference Flow: foec_incident_details

This data flow is sent by the other emergency centers to the Manage Emergency Services function and contains data about an incident that is taking place outside the area of operation serviced by the local function. It consists of the following data items each of which is defined in its own DDE: incident_location + incident_number + incident_description + incident_start_time + incident_duration + incident_severity + incident_type.

Physical Architecture Flow Name: incident response coordination

Incident response procedures, resource coordination, and current incident response status that are shared between allied response agencies to support a coordinated response to incidents. This flow also coordinates a positive hand off of responsibility for all or part of an incident response between agencies.

Logical Architecture Reference Flow: foec_incident_response_coordination

This data flow supports coordination of an incident response between allied response agencies. It supports the coordination of response procedures, status and resources between agencies. It also supports the coordinated hand-off of responsibility for all or part of an emergency response. It consists of the following data items which are defined in their own DDEs: incident_response_status + agency_incident_response_procedures + incident_resource_coordination + hand_off_coordination.

Personal Information Access => **Emergency Management**

Physical Architecture Flow Name: emergency notification

An emergency request for assistance originated by a traveler using an in-vehicle, public access, or personal device. Sufficient information is provided so that the recipient can determine the location of the emergency as a minimum. Additional information identifying the requestor and requesting device and the nature and severity of the emergency may also be provided (and required) by some systems.

Logical Architecture Reference Flow: emergency_request_personal_traveler_details
This data flow is used by the Provide Driver and Traveler Services function to send data about an emergency declared by a traveler using a personal device to the Manage Emergency Services function. It contains the following data items each of which is defined in its own DDE: date + time + traveler_personal_emergency_request.

Remote Traveler Support => **Emergency Management**

Physical Architecture Flow Name: emergency notification

An emergency request for assistance originated by a traveler using an in-vehicle, public access, or personal device. Sufficient information is provided so that the recipient can determine the location of the emergency as a minimum. Additional information identifying the requestor and requesting device and the nature and severity of the emergency may also be provided (and required) by some systems.

Logical Architecture Reference Flow: emergency_request_traveler_details
This data flow is used by the Provide Driver and Traveler Services function to send data about an emergency declared by a traveler using a kiosk or other device to the Manage Emergency Services function. It contains the following data items each of which is defined in its own DDE: date + time + traveler_personal_emergency_request.

Traffic Management => **Emergency Management**

Physical Architecture Flow Name: current network conditions

Current traffic information, road conditions, and camera images that can be used to locate and verify reported incidents, and plan and implement an appropriate response.

Logical Architecture Reference Flow: incident_video_for_emergency_services
This data flow is used within the Manage Traffic function. It contains current video images of incidents requested by the Manage Emergency Services facility. It consists of the following data item which is defined in its own DDE: incident_video_image.

Logical Architecture Reference Flow: traffic_data_for_emergency_services
This data flow is used within the Manage Traffic function and contains current traffic information and roadway environmental conditions for the emergency management system. It consists of the following items each of which is defined in it own DDE: roadway_environment_conditions + link_state_data.

Physical Architecture Flow Name: emergency traffic control response

Status of the green wave or other special traffic signal control strategy implemented in response to the emergency traffic control request.

Logical Architecture Reference Flow: emergency_traffic_control_response
This data flow is sent from the Manage Traffic function to the Manage Emergency Services function. It contains a list of the route segments that have been provided for use by an emergency vehicle and indicates the traffic management strategy that is in effect for these route segments. date + list_size + list_size{route_segment_identity + route_segment_estimated_arrival_time} + time + selected_emergency_vehicle_strategy.

Physical Architecture Flow Name: incident information request

Request for incident information, clearing time, severity. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Logical Architecture Reference Flow: incident_details_request
This data flow is used by the Manage Traffic function to request details of incidents from the Manage Emergency Services function. It contains the following data items each of which is defined in its own DDE: incident_type + incident_oldest_time.

Physical Architecture Flow Name: incident notification

The notification of an incident including its nature, severity, and location.

Logical Architecture Reference Flow: incident_alert
This data flow is used to send details of an incident from the Manage Traffic function to the Manage Emergency Services function. It contains the following data items each of which is defined in its own DDE: incident_location + incident_start_time + incident_duration + incident_severity + incident_type + incident_traffic_impact.

Logical Architecture Reference Flow: incident_response_clear
This data flow is sent from the Manage Traffic function to the Manage Emergency Services function and is an indication that the Manage Traffic function has data indicating that an incident has been cleared. It consists of the following items of data each of which is defined in its own DDE: incident_location + incident_type.

Logical Architecture Reference Flow: wrong_way_vehicle_detection
This data flow is sent by the Manage Traffic function to the Manage Emergency Services function and contains data about wrong-way vehicles detected in reversible lanes. It consists of the following data items each of which is defined in its own DDE: vehicle_identity + vehicle_license + incident_video_image + traffic_video_image + vehicle_detection_data.

Physical Architecture Flow Name: resource deployment status

Status of traffic management center resource deployment identifying the resources available and their current deployment status.

Logical Architecture Reference Flow: resource_deployment_status
This data flow is sent to the Manage Traffic function indicating the availability of the requested traffic management resources and provides current status of their deployment.

Transit Management => **Emergency Management**

Physical Architecture Flow Name: transit emergency data

Initial notification of transit emergency at a transit stop or on transit vehicles and further coordination as additional details become available and the response is coordinated.

Logical Architecture Reference Flow: transit_coordination_data
This data flow is sent from the Manage Transit function to the Manage Emergency Services function. It is used to provide data on the way in which the response to a transit incident should be coordinated. The data flow consists of the following data items each of which is defined in its own DDE: transit_coordination_information + transit_fleet_operation_acknowledge + transit_response_to_incident.

Logical Architecture Reference Flow: transit_emergency_data
This data flow is sent from the Manage Transit function to the Manage Emergency Services function and contains details of an emergency on-board a transit vehicle. It consists of the following data items each of which is defined in its own DDE: incident_location + incident_start_time + incident_duration + incident_severity.

Logical Architecture Reference Flow: transit_incident_details
This data flow is sent from the Manage Transit function to the Manage Emergency Services function and contains

details of an incident in the transit operations network. It consists of the following data items each of which is defined in its own DDE: incident_location + incident_start_time + incident_duration + incident_severity.

Vehicle => **Emergency Management**

Physical Architecture Flow Name: emergency notification

An emergency request for assistance originated by a traveler using an in-vehicle, public access, or personal device. Sufficient information is provided so that the recipient can determine the location of the emergency as a minimum. Additional information identifying the requestor and requesting device and the nature and severity of the emergency may also be provided (and required) by some systems.

Logical Architecture Reference Flow: emergency_request_driver_details
This data flow is used by the Provide Driver and Traveler Services function to send data about an emergency declared by a driver to the Manage Emergency Services function. It contains the following data items each of which is defined in its own DDE: date + driver_personal_emergency_request + time.

Logical Architecture Reference Flow: emergency_request_vehicle_details
This data flow is used by the Provide Vehicle Control and Monitoring function to send data about an emergency automatically declared by a vehicle to the Manage Emergency Services function. It contains the following data items each of which is defined in its own DDE: date + time + vehicle_emergency_request.

Weather Service => **Emergency Management**

Physical Architecture Flow Name: weather information

Accumulated predicted and current weather data (e.g., temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.).

Logical Architecture Reference Flow: fws_current_weather
This data flow is sent to the Manage Traffic function and the Provide Driver and Traveler Services functions. It contains details of the current weather conditions, e.g. temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.

Logical Architecture Reference Flow: fws_predicted_weather
This data flow is sent to the Manage Traffic and Provide Driver and Traveler Services functions. It contains details of the predicted weather conditions, e.g. temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.

2.7.4 Subsystem Architecture Flow Diagrams

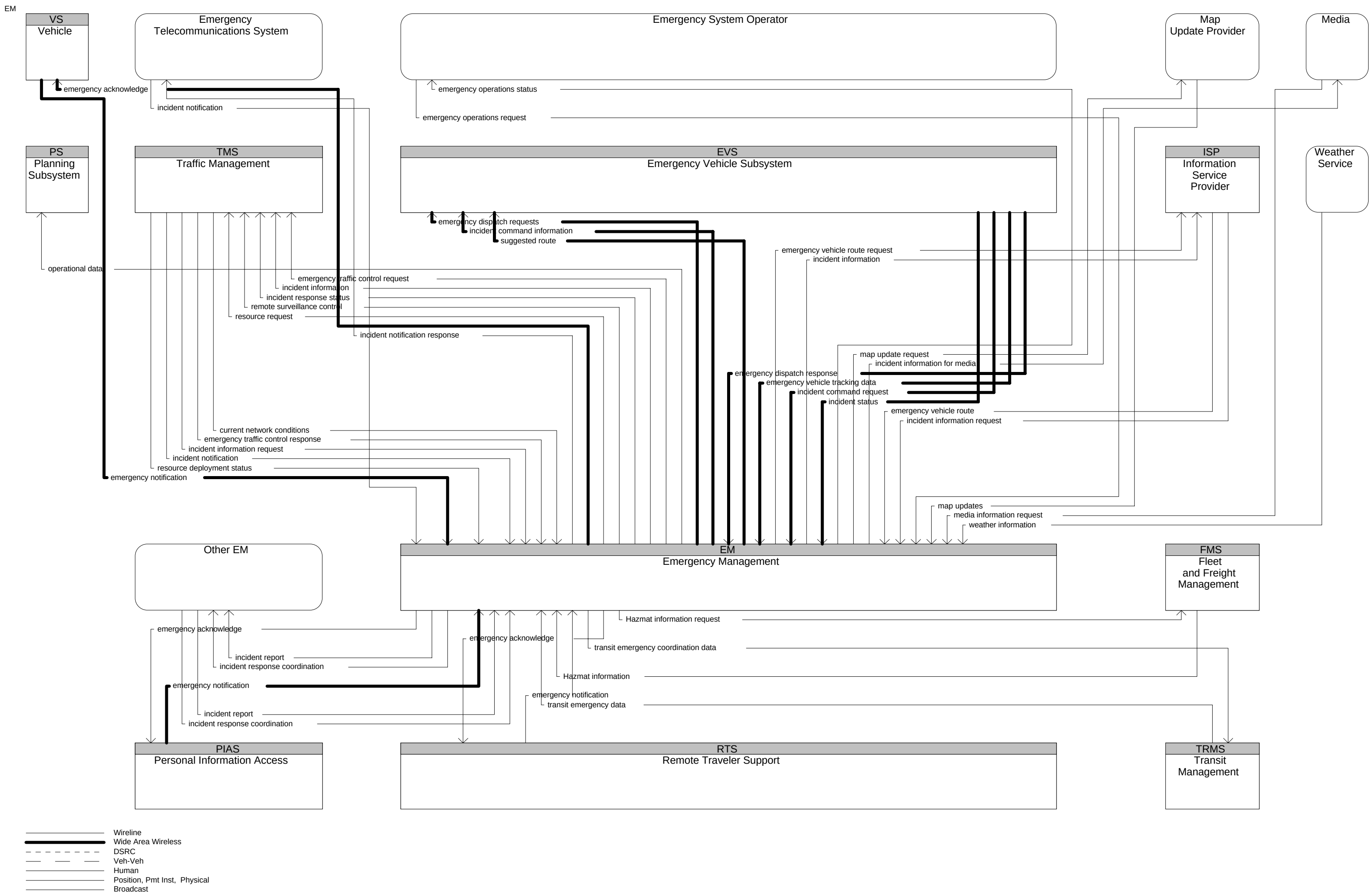


Figure 2.7-2 Architecture Flow Diagram for Emergency Management

2.8 Emergency Vehicle Subsystem

This subsystem resides in an emergency vehicle and provides the sensory, processing, storage, and communications functions necessary to support safe and efficient emergency response. The Emergency Vehicle Subsystem includes two-way communications to support coordinated response to emergencies in accordance with an associated Emergency Management Subsystem. Emergency vehicles are equipped with automated vehicle location capability for monitoring by vehicle tracking and fleet management functions in the Emergency Management Subsystem. Using these capabilities, the appropriate emergency vehicle to respond to each emergency is determined. Route guidance capabilities within the vehicle enable safe and efficient routing to the emergency. In addition, the emergency vehicle may be equipped to support signal preemption through communications with the roadside subsystem.

2.8.1 Alternative Configurations

The Emergency Vehicle Subsystem as with the other specialty vehicle subsystems is combined with the Vehicle Subsystem (VS) to form a complete package for emergency vehicles. As is indicated in the Figure 2.8-1, typically vehicles digital communicate links with their dispatch center although if vehicle based communication standards are adopted, vehicles may communicate with any dispatch center. Emergency vehicles which may use the Emergency Vehicle Subsystem include police, fire, ambulance, and tow trucks. It may be possible to take advantage of the in-vehicle display technology deployed enmass in the Vehicle Subsystem to display situation information for emergency vehicles including special routing information, proximity of other vehicles, and so on. Emergency Vehicles may also be equipped to supply signal preemption commands to the roadway although special routing through the Traffic Management Subsystem in which the TMS provides a “green wave” to the vehicle is preferable.

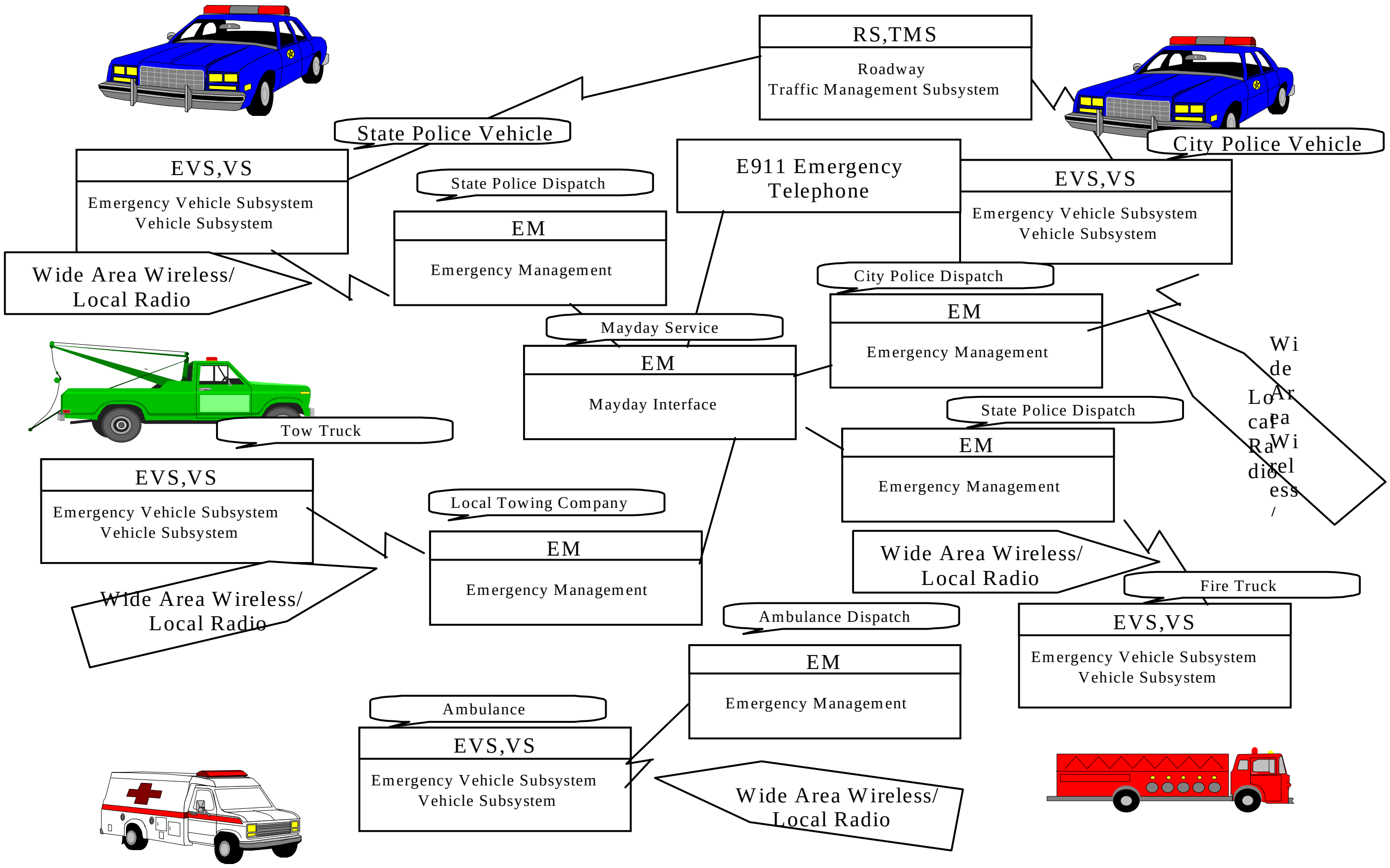


Figure 2.8-1 Alternative Configurations for EVS

2.8.2 Subsystem Equipment Packages and Supporting Process Specifications for EVS

On-board EV En Route Support

This Equipment package provides capabilities that support safe and expedient arrival on the incident scene. This package provides dispatch and routing information, tracks the vehicle, and preempt signals via short range communication directly with traffic control equipment at the roadside.

Process Specifications

5.3.3 Track Vehicle

Overview: This process shall manage information about the location of all emergency vehicles available for dispatch and that have been dispatched, and the ETA for vehicles en route. The process shall send this data to the store of emergency vehicle status data. If the vehicle is on its way to an emergency, as indicated by the received vehicle status, the process shall also send data to processes in the Manage Traffic function that will enable the vehicle to have whatever level and mode of priority is available and granted at traffic signals.

5.3.5 Provide Emergency Personnel Interface

Overview: This process shall provide an interface for emergency personnel, through which data can be exchanged with other processes in the Manage Emergency Services function. It shall support the exchange of incident data to which responses are being made by emergency personnel. The process shall support inputs from emergency personnel in both audible and manual forms, with outputs being available in both audio or visual forms. The visual form may include display and hardcopy formats. Both inputs and outputs shall be provided in such a way that while alerting the driver to the information they contain, they shall in no way impair the driver's ability to operate the vehicle in a safe manner.

On-board EV Incident Management Communication
This Equipment package provides a direct interface between the emergency vehicle and incident management personnel.

Process Specifications
5.3.5 Provide Emergency Personnel Interface

Overview: This process shall provide an interface for emergency personnel, through which data can be exchanged with other processes in the Manage Emergency Services function. It shall support the exchange of incident data to which responses are being made by emergency personnel. The process shall support inputs from emergency personnel in both audible and manual forms, with outputs being available in both audio or visual forms. The visual form may include display and hardcopy formats. Both inputs and outputs shall be provided in such a way that while alerting the driver to the information they contain, they shall in no way impair the driver's ability to operate the vehicle in a safe manner.

2.8.3 Subsystem Interfaces for EVS

Emergency Management => Emergency Vehicle Subsystem

Physical Architecture Flow Name: emergency dispatch requests

Emergency vehicle dispatch instructions including incident location and available information concerning the incident.

Logical Architecture Reference Flow: emergency_vehicle_dispatch_request
This data flow is used within the Manage Emergency Services function and contains data for the request for an emergency vehicle dispatch. The data flow specifies the information needed to respond to a particular incident. It consists of the following data items which are defined in their own DDEs. Up to eight (8) may be used to enable up to eight different types of emergency vehicles to be specified for a single incident. 8{emergency_center_identity + emergency_vehicle_identity + emergency_vehicle_status_code + time + date + emergency_vehicle_incident_details}.

Physical Architecture Flow Name: incident command information

Information that supports local management of an incident. It includes resource deployment status, hazardous material information, traffic, road, and weather conditions, evacuation advice, and other information that enables emergency personnel in the field to implement an effective, safe incident response.

Logical Architecture Reference Flow: local_decision_support
This data flow provides data that enables emergency personnel in the field to implement an effective incident response. It includes local traffic, road, and weather conditions, hazardous material information, and the current status of resources that have been allocated to an incident.

Physical Architecture Flow Name: suggested route

Suggested route for a dispatched emergency vehicle that may reflect current network conditions and the additional routing options available to en-route emergency vehicles that are not available to the general public.

Logical Architecture Reference Flow: emergency_vehicle_suggested_route
This data flow provides a suggested route for a dispatched vehicle that takes into account current traffic conditions, the current location and routes of other responding vehicles, and any special traffic control measures that are currently in effect to speed the response and increase the safety of emergency personnel. This data flow identifies the emergency

vehicle and provides turn-by-turn route information .

Emergency Personnel => **Emergency Vehicle Subsystem**

Physical Architecture Flow Name: emergency personnel inputs

Current incident status information and requests from emergency personnel in the field for information and/or resources.

Logical Architecture Reference Flow: fep_emergency_dispatch_acknowledge
This data flow is sent from the emergency personnel to the Manage Emergency Services function to acknowledge that the emergency vehicle has been dispatched and is on its way to the incident identified in the dispatch request.

Logical Architecture Reference Flow: fep_incident_command_request
This data flow identifies commands and resource requests associated with local management of an evolving incident response by emergency personnel in the field.

Logical Architecture Reference Flow: fep_incident_status
This data flow is sent from the emergency personnel to report the current status of an incident, e.g. length of time to clear site, length of time involved in work at site.

Emergency Vehicle Subsystem => **Emergency Management**

Physical Architecture Flow Name: emergency dispatch response

Request for additional emergency dispatch information (e.g., a suggested route) and provision of en-route status.

Logical Architecture Reference Flow: emergency_vehicle_dispatch_response
This data flow provides current enroute status and requests any additional current incident response status .

Physical Architecture Flow Name: emergency vehicle tracking data

The current location and operating status of the emergency vehicle.

Logical Architecture Reference Flow: emergency_vehicle_tracking_data
This data flow is within the Manage Emergency function and contains the current location of an emergency vehicle, together with the time and date to which the location applies. In addition this data flow contains operational data for the emergency vehicle. The data flow consists of the following data items each of which is defined in its own DDE: date + time + vehicle_location_for_emergency_services + emergency_vehicle_operational_data.

Physical Architecture Flow Name: incident command request

Request for resources, commands for relay to other allied response agencies, and other requests that reflect local command of an evolving incident response.

Logical Architecture Reference Flow: incident_command_request
This data flow requests resources, provides commands for relay to other allied response agencies, and includes other requests that reflect local command of an evolving incident response .

Physical Architecture Flow Name: incident status

Information gathered at the incident site that more completely characterizes the incident and provides current incident response status.

Logical Architecture Reference Flow: incident_status_update
This data flow provides detailed incident information gathered by emergency personnel at the incident site. Information could include the number and extent of injuries, identification of vehicles and people involved, specification of hazardous material, and any other information required to completely and accurately determine the scope and severity of the incident and the required response. Current response status is also provided which includes identification of the resources on site, site management strategies in effect, and current clearance status. The data flow consists of the following data items each of which is defined in its own DDE: date + emergency_vehicle_identity + emergency_vehicle_status_code + incident_number + time + incident_status.

Emergency Vehicle Subsystem => Emergency Personnel

Physical Architecture Flow Name: dispatch information

Dispatch information and command to emergency personnel.

Logical Architecture Reference Flow: tep_emergency_dispatch_order
This data flow is sent to the emergency personnel from the Manage Emergency Services function and is the order for the emergency personnel to proceed. It includes data on the emergency vehicle identity, the incident type and its location.

Physical Architecture Flow Name: incident command information presentation

Presentation of information to emergency personnel in the field that supports local tactical decision-making within an incident command system structure.

Logical Architecture Reference Flow: tep_decision_support
This data flow presents information to emergency personnel in the field that is necessary to support an effective incident response. It includes local traffic, road, and weather conditions, hazardous material information, and the current status of resources that have been allocated to the incident .

Emergency Vehicle Subsystem => Roadway Subsystem

Physical Architecture Flow Name: local signal preemption request

Direct control signal or message to a signalized intersection that results in preemption of the current control plan and grants right-of-way to the requesting vehicle.

Logical Architecture Reference Flow: emergency_vehicle_preemptions
This data flow is sent by the Manage Emergency Services function to the Manage Traffic function. It contains the data necessary for an individual emergency services vehicle and the stream of traffic in which the vehicle is traveling to be given preemption (priority) at an indicator controller. This will be at the controller for a particular road junction, pedestrian crossing, or highway entrance ramp. The data is sent directly from the emergency vehicle to the next controller along its route and therefore is not subject to any centralized coordination. Local coordination may be provided if there are links between adjacent controllers. The data flow consists of the following data items each of which is defined in its own DDE: emergency_vehicle_junction_preemption + emergency_vehicle_pedestrian_preemption + emergency_vehicle_ramp_preemption + emergency_vehicle_sign_preemption.

Vehicle => Emergency Vehicle Subsystem

Physical Architecture Flow Name: vehicle location

Location of vehicle and other vehcile characteristics which are exchanged between vehicle subsystems.

Logical Architecture Reference Flow: vehicle_location_for_emergency_services

This data flow is sent from the Provide Driver and Traveler Services function to the Manage Emergency Services function and contains a vehicle's location as computed from data input to sensors controlled by the processes that determines vehicle location. This is a high precision data flow that enables the location of an emergency vehicle to be pin-pointed to a high degree of accuracy by the Manage Emergency Services function. It consists of the following data item which is defined in its own DDE: location_identity.

2.8.4 Subsystem Architecture Flow Diagram

EVS

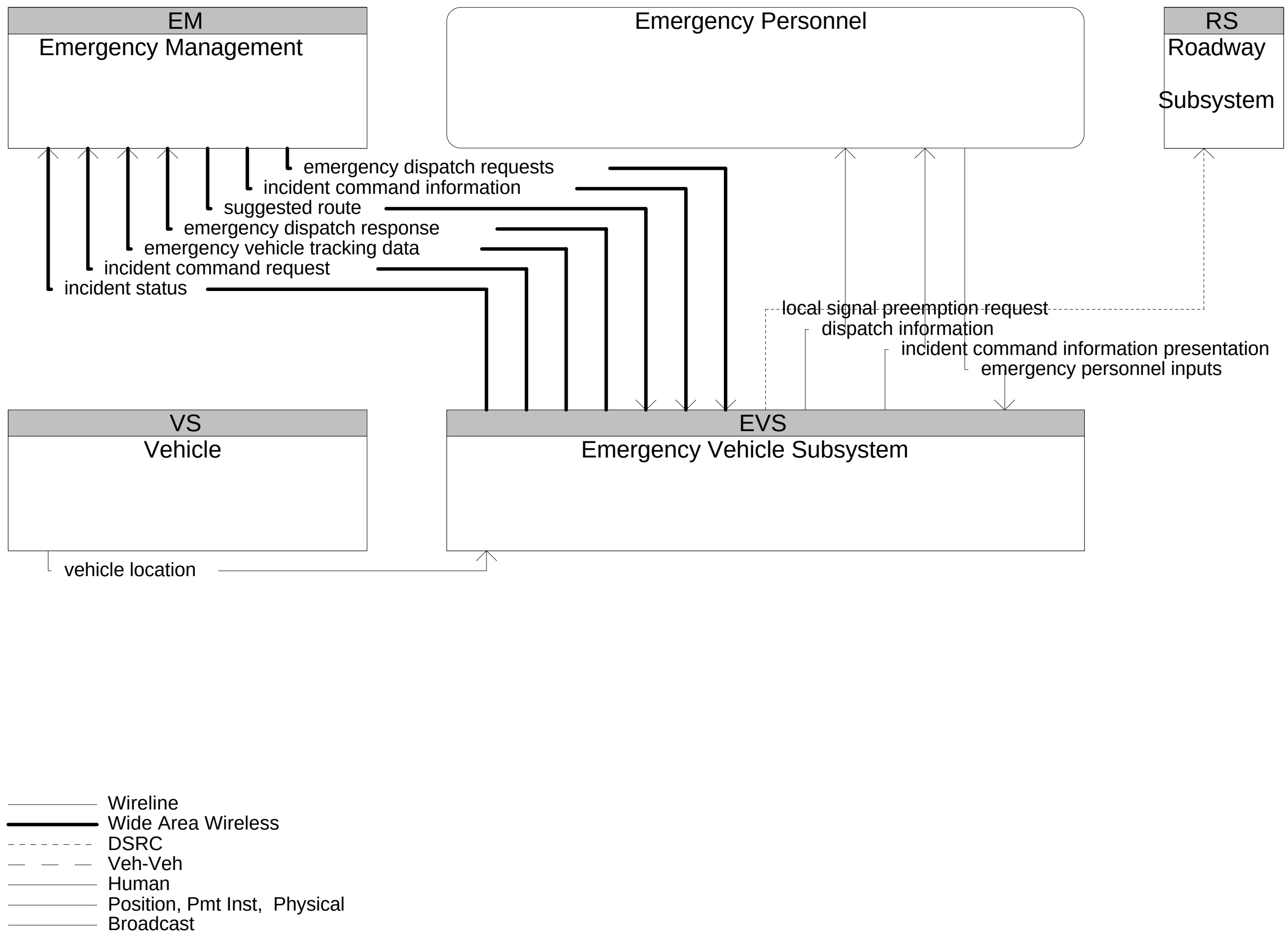


Figure 2.8-2 Architecture Flow Diagram for EVS

2.9 Emissions Management Subsystem

This subsystem operates at a fixed location and may co-reside with the Traffic Management Subsystem or may operate in its own distinct location depending on regional preferences and priorities. This subsystem provides the capabilities for air quality managers to monitor and manage air quality. These capabilities include collecting emissions data from distributed emissions sensors within the roadway subsystem. These sensors monitor general air quality within each sector of the area and also monitor the emissions of individual vehicles on the roadway. The sector emissions measures are collected, processed, and used to identify sectors exceeding safe pollution levels. This information is provided to toll administration, traffic management, and transit management systems and used to implement strategies intended to reduce emissions in and around the problem areas. Emissions data associated with individual vehicles, supplied by the Roadway Subsystem, is also processed and monitored to identify vehicles that exceed standards. This subsystem provides any functions necessary to inform the violators and otherwise ensure timely compliance with the emissions standards.

2.9.1 Alternative Configurations

The Emissions Management Subsystem represents environmental agencies such as an Air Quality Management District Figure 2.9-1. The subsystem works closely with the traffic management subsystem and receives pollution data from both traffic management and its own sensors.

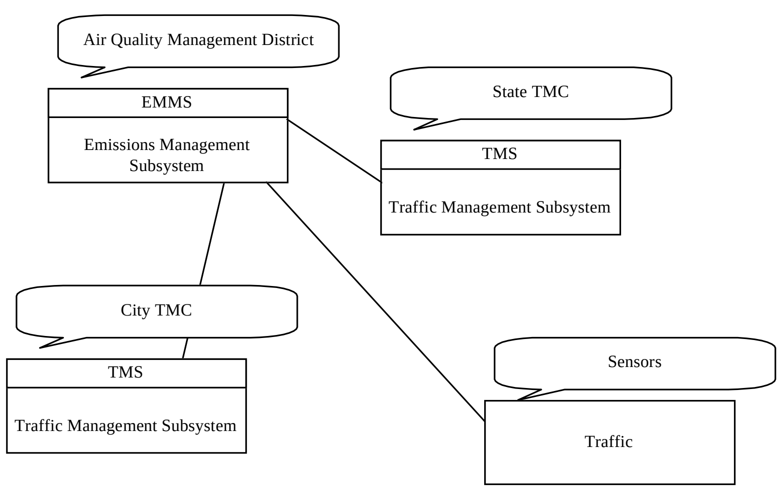


Figure 2.9-1 Configurations for Emissions Management

2.9.2 Subsystem Equipment Packages and Supporting Process Specifications for EMMS

Emissions Data Management
This Equipment package assimilates and stores air quality measures and roadside collected emissions data. General air quality measures are distributed as general traveler information and also may be used for in demand management programs. Collected roadside emissions are analyzed and used to detect, identify, and notify concerned parties regarding vehicles that exceed emissions standards.

Process Specifications
1.5.1 Provide Traffic Operations Personnel Pollution Data Interface

Overview: This process shall provide the interface between the traffic operations personnel and the processes and data stores used within the Manage Emissions facility of the Manage Traffic function. The process shall enable the personnel to access and update the pollution reference data used by other processes within the facility, and to access the pollution state data provided by those processes. The process shall support inputs from the traffic operations personnel. Where appropriate and/or requested by the traffic operations personnel, the process shall incorporate map data of the relevant part(s) of the road and freeway network served by the Manage Traffic function. The process shall obtain the map from a local data store, which it shall request to be updated by another process as and when required.

1.5.2 Process Pollution Data

Overview: This process shall process the pollution data being collected from sensors in the geographic area being served by the Manage Traffic function. The process shall integrate data from distributed roadside sensors (provided by another process) with that obtained directly from sensors looking at the general (wide area) environment. The data shall be checked by the process against the pollution levels that have been set up as reference points. If the process finds that the detected levels of pollution exceed the reference levels it shall generate pollution warnings. The process shall send these warnings to other processes in the Manage Traffic function for output to drivers and travelers.

1.5.3 Update Pollution Display Map Data

Overview: This process shall provide updates to the map data used in displays of pollution data produced by processes in the Manage Emissions facility of the Manage Traffic function. The process shall obtain the map data from a specialist map data supplier or some other appropriate data source, on receiving an update request from the traffic operations personnel interface process within the Manage Emissions facility.

1.5.4 Manage Pollution State Data Store

Overview: This process shall manage the store of pollution state data in the Manage Emissions facility of the Manage Traffic function. The data in the store shall be that which has been received by the process from other processes within the facility. The process shall manage the data in the store to enable its contents to be available to other processes within the Manage Traffic function, and to traffic operations personnel, via an interface process within the Manage Emissions facility.

1.5.7 Manage Pollution Data Log

Overview: This process shall manage the log of pollution data within the Manage Emissions facility of the Manage Traffic function. The process shall receive data for entry into the log from other processes within the facility. It shall also send the contents of the log to the Plan System Deployment function for use in planning future modifications to the ITS network.

1.5.8 Manage Pollution Reference Data Store

Overview: This process shall manage the store of pollution reference data within the Manage Emissions facility of the Manage Traffic function. It shall make the contents of the store available to other processes within the facility that are responsible for emissions management, and on request to the traffic operations personnel interface process. The process shall accept updates to the stored data from the traffic operations personnel interface process.

2.9.3 Subsystem Interfaces for EMMS

Emissions Management => **Map Update Provider**

Physical Architecture Flow Name: map update request

Request for a map update which could include a new underlying map or map layer updates.

Logical Architecture Reference Flow: tmup_request_pollution_display_update

This data flow is sent to the map update provider from the Manage Emissions facility within the Manage Traffic function. It contains a request for an update to the digitized map data for displays that can be used as background for the output of data on the levels of various atmospheric pollutants.

Emissions Management => **Planning Subsystem**

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: pollution_operational_data

This data flow is sent from the Manage Traffic function to the Plan System Deployment function. It contains data about the levels of pollution detected by roadside, wide area, and individual vehicle sensors located in the geographic area served by the function. The data will be that which has been collected since the last set of data was sent and will therefore have a time stamp record. The data flow consists of the following data items each of which is defined in its own DDE: date + list_size + list_size{pollution_state_roadside_collection + pollution_state_vehicle_collection} + pollution_state_area_collection + time.

Emissions Management => **Roadway Subsystem**

Physical Architecture Flow Name: vehicle pollution criteria

Vehicular pollution acceptance criteria.

Logical Architecture Reference Flow: pollution_state_vehicle_acceptance_criteria

This data flow is used within the Manage Traffic function and contains data on the pollution levels produced by different types of vehicle under various operating conditions. It consists of the following items of data each of which is defined in its own DDE: pollution_vehicle_acceptance_data + pollution_vehicle_acceptance_conditions.

Emissions Management => **Traffic Management**

Physical Architecture Flow Name: wide-area statistical pollution information

Aggregated region-wide measured emissions data and possible pollution incident information.

Logical Architecture Reference Flow: pollution_incident

This data flow is used within the Manage Traffic function and contains details of a current or predicted pollution incident. The incident type will be set to the three character code for a pollution incident and will depend on the type of pollutant that is involved. The data flow consists of the following data items each of which is defined in its own DDE: incident_start_time + incident_duration + incident_location + incident_severity + incident_type.

Logical Architecture Reference Flow: pollution_state_data
This data flow is used within the Manage Traffic function as a means of transferring current pollution data from the Manage Emissions facility to the Manage Demand facility. It contains data about the current levels of pollution obtained from the store of pollution data and consists of the following data items each of which is defined in its own DDE: current_ozone_pollution + current_nitrous_oxide_pollution + current_sulfur_dioxide_pollution + current_hydrocarbon_pollution + current_carbon_monoxide_pollution + current_particulate_pollution + current_pollution_location + vehicle_type.

Logical Architecture Reference Flow: wide_area_pollution_data
This data flow is used within the Manage Traffic function as a means of transferring current pollution data from the Manage Emissions facility to the Provide Traffic Surveillance facility. It contains data about the current levels of pollution obtained from the store of pollution data in the area covered by the Traffic Management Center (TMC) and consists of the following data items each of which is defined in its own DDE: pollution_state_area_collection + list_size + list_size{pollution_state_roadside_collection}.

Emissions Management => Traffic Operations Personnel

Physical Architecture Flow Name: pollution data display

Both reference and current pollution status details for a given geographic area.

Logical Architecture Reference Flow: ttop_pollution_data_display
This data flow is sent by the Manage Traffic function to the traffic operations personnel. It contains details of both the pollution reference data and the current pollution state in the geographic area served by the ITS functions.

Environment => Emissions Management

Physical Architecture Flow Name: pollution data

Measured emissions data comprised of various atmospheric pollutants.

Logical Architecture Reference Flow: fe_area_pollutant_levels
This data flow is used within the Manage Traffic function. It contains analog data from which sensors within ITS can determine the actual levels of various atmospheric pollutants, such as nitrous oxide, sulfur dioxide, hydrocarbons, carbon monoxide and ozone, that are generally present within a particular geographic area. They are therefore really background levels of pollutants that are not due to any particular sources such as road traffic. The sizing assumption below is based on an estimate of the digitized equivalent information.

Map Update Provider => Emissions Management

Physical Architecture Flow Name: map updates

Map update which could include a new underlying static or real-time map or map layer(s) update.

Logical Architecture Reference Flow: fmup_pollution_display_update
This data flow is sent from the map update provider to the Manage Emissions facility within the Manage Traffic function. It contains the digitized map data for displays that can be used as background for the output of data on the

levels of various atmospheric pollutants.

Roadway Subsystem => **Emissions Management**

Physical Architecture Flow Name: pollution data

Measured emissions data comprised of various atmospheric pollutants.

Logical Architecture Reference Flow: pollution_state_roadside_collection
This data flow is used within the Manage Traffic function and contains the digitized values of pollution levels obtained from roadside sensors in the geographic area served by the function. It consists of the following data items each of which is defined in its own DDE: current_ozone_pollution + current_nitrous_oxide_pollution + current_sulfur_dioxide_pollution + current_hydrocarbon_pollution + current_carbon_monoxide_pollution + current_particulate_pollution + current_roadside_pollution_location.

Logical Architecture Reference Flow: pollution_state_vehicle_collection
This data flow is used within the Manage Traffic function and contains the average levels of the various types of pollution that were being output by a particular type of violating vehicle. It consists of the following data items each of which is defined in its own DDE: current_ozone_pollution + current_nitrous_oxide_pollution + current_sulfur_dioxide_pollution + current_hydrocarbon_pollution + current_carbon_monoxide_pollution + current_particulate_pollution + vehicle_type.

Logical Architecture Reference Flow: pollution_state_vehicle_log_data
This data flow is used within the Manage Traffic function and contains a periodic average of the pollution data measured by sensors from actual vehicles. It consists of the following data item which is defined in its own DDE: pollution_state_vehicle_collection.

Traffic Management => **Emissions Management**

Physical Architecture Flow Name: pollution state data request

Aggregated emissions data information request.

Logical Architecture Reference Flow: pollution_state_data_request
This data flow is used within the Manage Traffic function as a means of requesting current pollution data to be sent from the Manage Emissions facility to the Manage Demand facility. It contains request for data about current levels of pollution. This data can be requested for a roadside or wide area location.

Traffic Operations Personnel => **Emissions Management**

Physical Architecture Flow Name: pollution data parameters

Nominal pollution data compliance (reference) levels for each sector of an urban area.

Logical Architecture Reference Flow: ftop_pollution_data_information_request
This data flow is sent from the traffic operations personnel to the Manage Traffic function and contains a request for a display of pollution reference or current pollution state data.

Logical Architecture Reference Flow: ftop_pollution_parameter_updates
This data flow is sent from the traffic operations personnel to the Manage Traffic function and contains a request for specified pollution reference data values to be updated to those provided.

2.9.4 Subsystem Architecture Flow Diagram

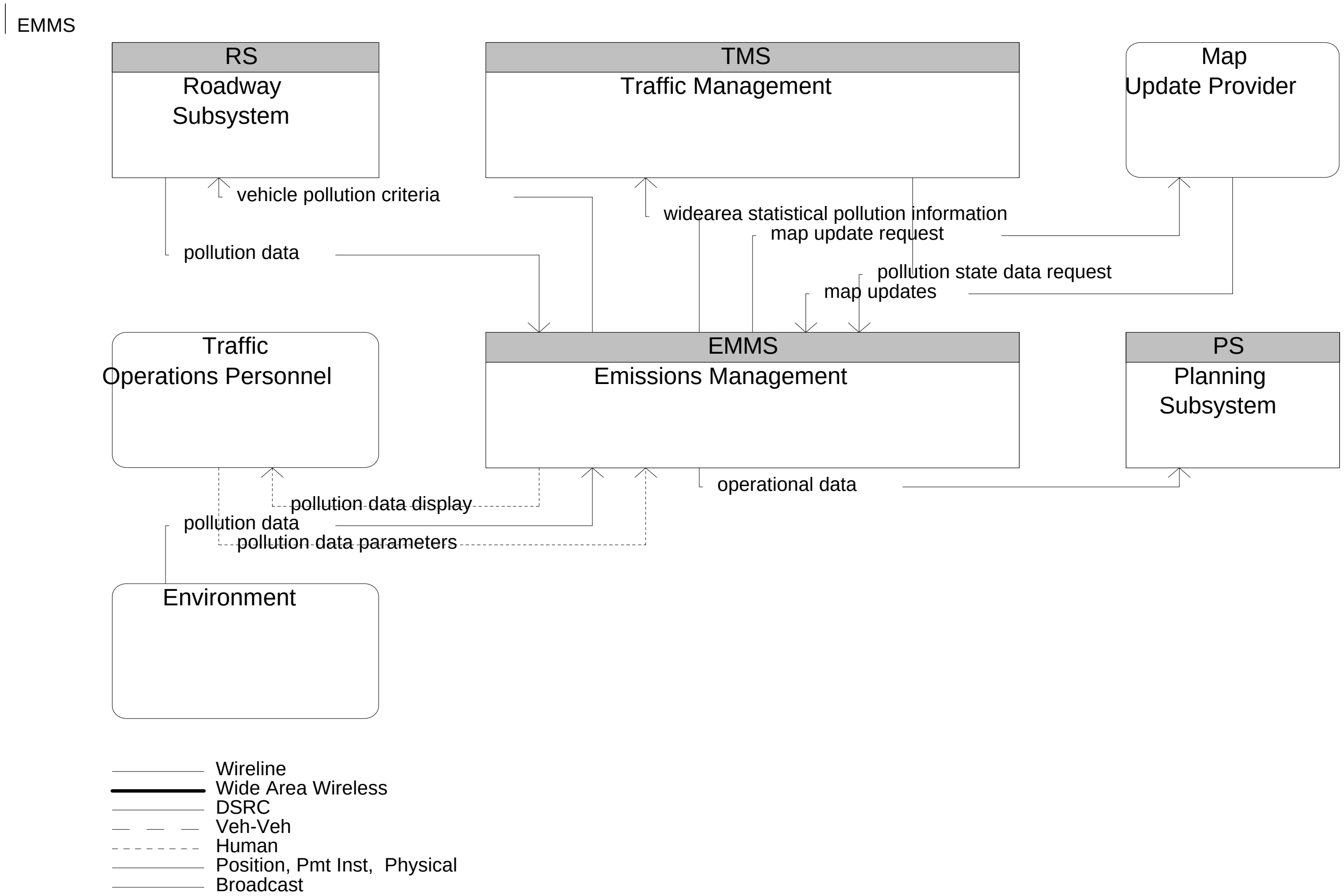


Figure 2.9-2 Architecture Flow Diagram for Emissions Management

2.10 Fleet and Freight Management

The Fleet and Freight Management Subsystem provides the capability for commercial drivers and dispatchers to receive real-time routing information and access databases containing vehicle and cargo locations as well as carrier, vehicle, cargo, and driver information. In addition, the capability to purchase credentials electronically shall be provided, with automated and efficient connections to financial institutions and regulatory agencies, along with post-trip automated mileage and fuel usage reporting. The Fleet Management Subsystem also provides the capability for Fleet Managers to monitor the safety of their commercial vehicle drivers and fleet. The subsystem also supports application for Hazmat credentials and makes information about Hazmat cargo available to agencies as required.

2.10.1 Alternative Configurations

The Fleet and Freight Management Subsystem manages fleets of CVO vehicles. The subsystem may be a large private trucking firm or public agency with a fleet of vehicles. It also supports single owner/operator concerns. A primary function of the FMS is to provide the capability for purchasing electronic credentials in an efficient manner. As illustrated in

Figure 2.10-1 the FMS may operate either in a firm's home office through dedicated facilities, or at a truck or rest stop through a public kiosk. These sites communicate with authorities via standard wireline communications. Alternatively, FMS functions could be loaded on a portable computer and managed from the cab of the vehicle using wide area wireless technology.

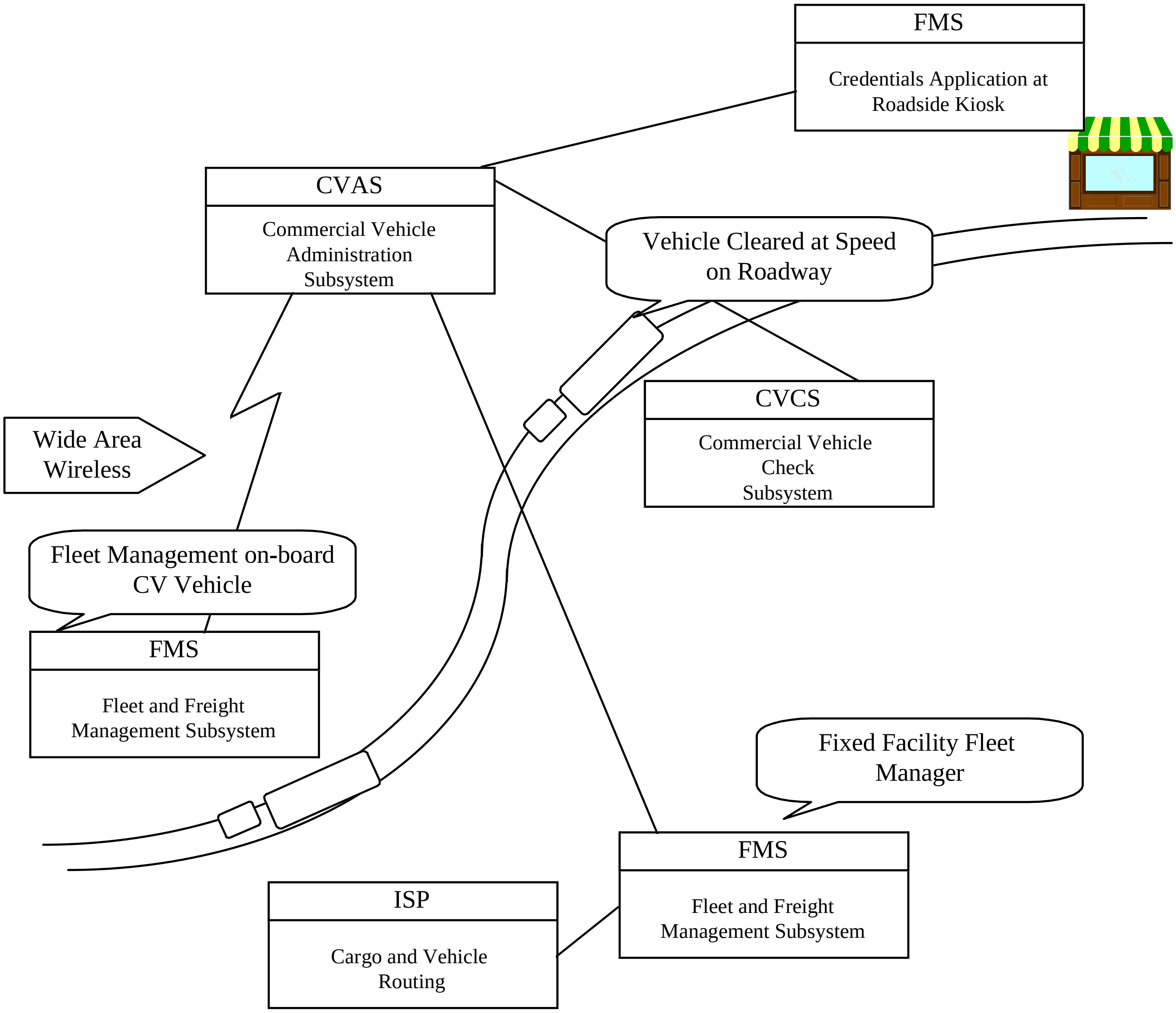


Figure 2.10-1 Alternative Configurations for Freight and Fleet Management

2.10.2 Subsystem Equipment Packages and Supporting Process Specifications for FMS

Fleet Administration
This Equipment package provides route plan information from the FMS to the TMS for network performance evaluation. It also provides for vehicle tracking, dispatch, and reporting to the fleet management center personnel

Process Specifications
2.1.2 Provide Commercial Fleet Static Route

Overview: This process shall be responsible for providing a static commercial vehicle route using data provided by the commercial vehicle manager. A static route is one which is based on geographic data and therefore takes no account of current or predicted traffic conditions, incidents, etc. The process shall provide the route using its own route generation algorithms and data from its own store of digitized map information.

2.1.4 Provide Fleet Manager Commercial Vehicle Communications

Overview: This process shall be responsible for providing the communications interface and data storage facility for data that is exchanged between the commercial vehicle manager and commercial vehicle drivers in their vehicles. The process shall support the receipt of data from the vehicle consisting of that processed from input received by sensors on board the vehicle and text data used to exchange general information with the driver. Only the output to the vehicle of the data that contains the general text message shall be supported by the process. The process shall enable access to the store of received data by the manager through the manager's interface process.

2.1.6 Manage Driver Instruction Store

Overview: This process shall be responsible for managing the store of driver route instructions so that they can be loaded with data for retrieval by the commercial vehicle driver. The data for loading into the store shall be sent to the process from other processes in the Manage Commercial Vehicle Fleet Operations facility of the Manage Commercial Vehicles function. The process shall enable the data to comprise vehicle route data and vehicle load information, including the points along the route at which identified cargo is to be picked up and/or dropped off. The process shall support the retrieval of this data by the commercial vehicle driver through the driver's interface process.

2.6.1 Provide Commercial Vehicle Manager Tag Data Interface

Overview: This process shall be responsible for providing an interface through which the commercial vehicle manager can set up the data in the tag on-board a commercial vehicle. The data that the process enables the manager to write to the tag will be that which identifies the carrier, driver and vehicle. The process shall also enable the manager to read this data from the tag, but shall not enable the reading of any other data from the tag. Data provided by the manager shall also be sent by the process to the tag the process that manages electronic credentials and tax filing for use by the manager in future enrollments. The process shall support inputs from the commercial vehicle manager in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be in hardcopy, or as a display.

Fleet Credentials and Taxes Management and Reporting

This Equipment package provides the Fleet and Freight Management Subsystem the capabilities to purchase credentials and file trip reports electronically by the fleet managers, to perform automated enrollment at the roadside facilities, and electronically manage the credentials checking by the roadside commercial vehicle inspectors. The electronic purchase shall be performed in accordance with developing standards such that a single integrated system for electronic payments might develop ensuring that deployment across multiple agency political boundaries is performed without degradation. Inherent to credential management shall be the management of the vehicles, with a prerequisite of the vehicle tracking software from the Fleet Administration Equipment package.

Process Specifications

2.1.1 Manage Commercial Fleet Electronic Credentials and Tax Filing

Overview: This process shall be responsible for providing the commercial vehicle manager with the ability to manage the activities of commercial vehicles. The process shall provide the capability for the manager to obtain commercial vehicle routes. When a route has been confirmed, the process shall enable the manager to enroll commercial vehicles for electronic clearance at roadside check station facilities, to process and pay for electronic credential and tax filing, to send tag data to the Provide Commercial Vehicle On-board Data facility, and to provide vehicle route instructions for use by the commercial vehicle driver. Periodically it shall also send reports about taxes that have been paid to the Administer Commercial Vehicles facility. The process shall also enable the manager to obtain commercial vehicle activity reports from the logs provided by roadside checkstation facilities. It shall be possible to obtain these reports either on request or at periodic intervals.

2.1.3 Provide Flt Mgr Electronic Credentials and Tax Filing Interface

Overview: This process shall be responsible for providing an interface for the commercial vehicle manager. The process shall enable this interface to provide the manager with facilities for the input of data used to set up commercial vehicle routes, to pay the necessary taxes and duties so that a commercial vehicle can be enrolled for a particular route, to exchange general information messages with a driver in a vehicle, and to set up instructions for a driver to take a vehicle on a particular route. It shall be possible for the driver's route instructions input by the manager to include details of the cargo to be picked up and/or dropped off at each point along the route. The enrollment activity supported by the process shall enable a commercial vehicle to pass through the roadside checkstations along its route without stopping, unless safety checks are required. The process shall support inputs from the commercial vehicle manager in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be in hardcopy, or as a display.

2.2.1 Manage CV Electronic Credential and Tax Filing Interface

Overview: This process shall be responsible for providing the commercial vehicle driver with the ability to manage the activities of a commercial vehicle. In this instance the driver is assumed to be acting in the role of a commercial vehicle manager, and is therefore probably the owner/driver of the vehicle. The process shall provide the capability for the driver to obtain commercial vehicle routes, to enroll commercial vehicles for electronic clearance at roadside check station facilities, and to process and pay for electronic credential and tax filing.

7.5.4 Provide Commercial Fleet Payment Instrument Interface

Overview: This process shall be responsible for providing the interface through which credit identity or stored credit values may be collected from the tags used by commercial fleet managers. The process shall support the use of the tag data to complete payment for the filing of electronic credentials and tax information that will enable a commercial vehicle to be cleared to travel within geographic area served by a particular jurisdictional authority. This process shall also enable the stored credit value to be debited as an alternative method of payment.

Fleet HAZMAT Management

This Equipment package provides the Fleet and Freight Management Subsystem the capabilities to enhance the Fleet Administration Equipment package functions by adding HAZMAT tracking. The additional requirements to perform this function include enhanced processing and enhanced fleet management software. In order to effectively track HAZMAT cargo, communication interfaces to Information Service Providers, and Emergency Management Subsystems shall be provided, including additional communication software.

Process Specifications

2.1.1 Manage Commercial Fleet Electronic Credentials and Tax Filing

Overview: This process shall be responsible for providing the commercial vehicle manager with the ability to manage the activities of commercial vehicles. The process shall provide the capability for the manager to obtain commercial vehicle routes. When a route has been confirmed, the process shall enable the manager to enroll commercial vehicles for electronic clearance at roadside check station facilities, to process and pay for electronic credential and tax filing, to send tag data to the Provide Commercial Vehicle On-board Data facility, and to provide vehicle route instructions for use by the commercial vehicle driver. Periodically it shall also send reports about taxes that have been paid to the Administer Commercial Vehicles facility. The process shall also enable the manager to obtain commercial vehicle activity reports from the logs provided by roadside checkstation facilities. It shall be possible to obtain these reports either on request or at periodic intervals.

Fleet Maintenance Management

This Equipment package provides the capability to use vehicle mileage data to automatically generate preventative maintenance schedules for each specific vehicle by utilizing vehicle tracking data from the prerequisite tracking Equipment package. In addition, capability to automatically ensure that proper service personnel are provided information for maintenance activities and to record and verify that maintenance work was performed shall be provided.

Process Specifications

2.1.6 Manage Driver Instruction Store

Overview: This process shall be responsible for managing the store of driver route instructions so that they can be loaded with data for retrieval by the commercial vehicle driver. The data for loading into the store shall be sent to the process from other processes in the Manage Commercial Vehicle Fleet Operations facility of the Manage Commercial Vehicles function. The process shall enable the data to comprise vehicle route data and vehicle load information, including the points along the route at which identified cargo is to be picked up and/or dropped off. The process shall support the retrieval of this data by the commercial vehicle driver through the driver's interface process.

Freight Administration and Management
This equipment package provides the communication necessary to track cargo from source to destination via links to intermodal freight shippers and depots. There are also communication links to cargo routing services.

Process Specifications
2.7 Manage Cargo

Overview: This process shall be responsible for providing facilities for the management of cargo shipments. The process shall enable these shipments to be routed via intermodal shippers and depots and may not need the services of a commercial vehicle manager or driver.

2.10.3 Subsystem Interfaces for FMS

Commercial Vehicle Administration => Fleet and Freight Management

Physical Architecture Flow Name: activity reports

Activity reports containing records of citations, accidents, inspections, etc.

Logical Architecture Reference Flow: cf_periodic_activity_report
This data flow is used within the Manage Commercial Vehicles function. It consists of the data from the commercial vehicle roadside checkstation facility logs from which a report on activities will be issued. This data flow will have been produced as a result of a request from a commercial vehicle manager for periodic reports. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{cv_roadside_facility_identity + cv_roadside_periodic_activity_data}.

Physical Architecture Flow Name: compliance review report

Report containing data from facility activity logs from various roadside facilities.

Logical Architecture Reference Flow: cf_roadside_activity_report
This data flow is used within the Manage Commercial Vehicles function. It contains data from the commercial vehicle roadside checkstation facility logs from which a report on activities will be issued. This data flow will have been produced as a result of a specific request from a commercial vehicle manager and consists of the following data items each of which is defined in its own DDE: list_size + list_size{cv_roadside_facility_identity + cv_roadside_single_activity_data}.

Physical Architecture Flow Name: electronic credentials

Authenticated credentials including route enrollment and payment confirmation.

Logical Architecture Reference Flow: cf_enrollment_information
This data flow is used within the Manage Commercial Vehicles function and contains the data for enrollment on a particular route produced from data supplied by the commercial fleet manager. It contains the following data items each of which is defined in its own DDE: cv_route_number + cv_taxes_and_duties + route + route_type + cv_border_enrollments.

Logical Architecture Reference Flow: cf_enrollment_payment_confirmation
This data flow is used within the Manage Commercial Vehicles function to confirm that a payment of the taxes and duties for the enrollment of a particular commercial vehicle cargo, weight and type on a particular route from the commercial fleet manager has been accepted. It consists of the following data items each of which is defined in its own DDE: cv_route_number + cv_account_number + cv_amount_billed.

Logical Architecture Reference Flow: cv_enrollment_information
This data flow is used within the Manage Commercial Vehicles function and contains the data for enrollment on a

particular route produced from data supplied by the commercial vehicle driver. It contains the following data items each of which is defined in its own DDE: cv_route_number + cv_taxes_and_duties + route + route_type + cv_border_enrollments.

Logical Architecture Reference Flow: cv_enrollment_payment_confirmation
This data flow is used within the Manage Commercial Vehicles function to confirm that a payment of the taxes and duties for the enrollment of a particular commercial vehicle cargo, weight and type on a particular route from the commercial vehicle driver has been accepted. It consists of the following data items each of which is defined in its own DDE: cv_account_number + cv_amount_billed + cv_driver_credit_identity + cv_route_number.

Commercial Vehicle Manager => Fleet and Freight Management

Physical Architecture Flow Name: fleet manager inquiry

Inquiry from fleet manager requesting data from commercial vehicle management system.

Logical Architecture Reference Flow: fcvm_carrier_number
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function and contains the commercial vehicle carrier identification number to be entered into the vehicle's tag data store.

Logical Architecture Reference Flow: fcvm_driver_number
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function and contains the commercial vehicle driver identification number to be entered into the vehicle's tag data store.

Logical Architecture Reference Flow: fcvm_enrollment_payment_request
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function and contains a request for payment of the taxes and duties needed to cover the enrollment of a particular class of vehicle and cargo at a particular weight on a particular route. The data will include the account number (12 bytes) from which the cost of the taxes and duties are to be deducted and the route number (4 bytes) to which they apply. Commercial vehicle drivers will make this request because they are acting on the role of their own fleet managers, i.e. they will be owner drivers.

Logical Architecture Reference Flow: fcvm_enrollment_request
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function and contains a request for the enrollment of a particular class of vehicle and cargo at a particular weight on a particular route.

Logical Architecture Reference Flow: fcvm_other_data_input
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function and contains the response to a previous request for other data to be input for use within the function. Commercial vehicle drivers will make this request because they are acting in the role of their own fleet managers, i.e. they will be owner drivers.

Logical Architecture Reference Flow: fcvm_preclearance_data
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function and contains input data defining the route (4 bytes) and vehicle identity (16 bytes) for which preclearance is required.

Logical Architecture Reference Flow: fcvm_request_driver_route_instructions
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function and contains a request for the data that defines the route that the commercial vehicle driver is to follow and/or details of cargo to be picked up and/or dropped at the origin, destination or intermediate points.

Logical Architecture Reference Flow: fcvm_request_on_board_vehicle_data
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function. It contains a request for the data that has been obtained from the store of safety and trip related data on-board a commercial vehicle.

Logical Architecture Reference Flow: fcvm_request_tag_data_output
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function. It contains a request for the output of the current data that is stored on a commercial vehicle's tag. Only the data that can be written by the manager is output.

Logical Architecture Reference Flow: fcvm_roadside_activity_report_request
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function. It contains a request for output of the commercial vehicle roadside checkstation facility logs, showing the activities of a particular carrier, driver and vehicle combination. The request may be for a one time report or for the report to be produced periodically.

Logical Architecture Reference Flow: fcvm_route_data
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function and contains input defining the requested commercial vehicle route data, e.g. origin, destination, preferences, constraints, etc.

Logical Architecture Reference Flow: fcvm_route_function_request
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function. It contains an activity number for facilities within the function, which may be one of the following: 1 - request route, 2 - request preclearance, 3 - store route, 4 - provide list of stored routes 5 - display route details 6 - delete route. Values of three (3) and four (4) must be accompanied by a route number.

Logical Architecture Reference Flow: fcvm_trip_identity
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function and contains the commercial vehicle trip identification number to be entered into the vehicle's tag data store.

Logical Architecture Reference Flow: fcvm_update_driver_route_instructions
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function and contains changes or additions to the data defining the route that the driver is to follow and/or details of cargo to be picked up and/or dropped at the origin, destination or intermediate points.

Logical Architecture Reference Flow: fcvm_vehicle_number
This data flow is sent from the commercial vehicle manager to the Manage Commercial Vehicles function and contains the commercial vehicle's identification number to be entered into the vehicle's tag data store.

Commercial Vehicle Subsystem => Fleet and Freight Management

Physical Architecture Flow Name: driver and vehicle information

Requests from the driver and vehicle for routing, payment, and enrollment information.

Logical Architecture Reference Flow: cf_driver_route_instructions_request
This data flow is used within the Manage Commercial Vehicles function and contains a request from the commercial vehicle driver for output of the driver route instructions. It consists of the following data items each of which is defined in its own DDE: cv_driver_number + cv_route_number.

Logical Architecture Reference Flow: cv_driver_enrollment_payment_request
This data flow is used within the Manage Commercial Vehicles function and contains data required to enable payment for enrollment of a commercial vehicle for the use of a particular route as provided by the commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cv_account_number + cv_driver_credit_identity + cv_route_number.

Logical Architecture Reference Flow: cv_driver_enrollment_request
This data flow is used within the Manage Commercial Vehicles function and contains data required for the enrollment of a commercial vehicle on a particular route as provided by a commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cv_cargo_class + cv_route_number + cv_trip_identity + cv_vehicle_class + cv_weight_class.

Logical Architecture Reference Flow: cv_driver_route_request
This data flow is used within the Manage Commercial Vehicles function by the commercial vehicle driver to request a commercial vehicle route. It contains the following data items each of which is defined in its own DDE: trip_request + route_type.

Logical Architecture Reference Flow: cv_driver_storage_request
This data flow is used within the Manage Commercial Vehicles function to manage the store of routes used by the commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cv_storage_action_flag + cv_route_number.

Logical Architecture Reference Flow: cv_static_route_data
This data flow is used within the Manage Commercial Vehicles function. It contains the data for a static based route provided as a result of data provided by the commercial vehicle driver acting as a fleet manager.

Physical Architecture Flow Name: on board vehicle data

Condition of the commercial vehicle sent to commercial vehicle manager primarily for maintenance purposes.

Logical Architecture Reference Flow: cf_on_board_vehicle_data
This data flow is used within the Manage Commercial Vehicle function and contains data collected on-board a commercial vehicle output of which has been requested by the commercial vehicle manager. It consists of the following data items each of which is defined in its own DDE: cv_on_board_data + cv_general_output_message + vehicle_location_for_cv.

Logical Architecture Reference Flow: cf_tag_data_store_output
This data flow is used within the Manage Commercial Vehicles function. It contains the output of the data currently being held by a type two commercial vehicle tag as previously requested by the commercial vehicle manager. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_trip_identity.

Emergency Management => Fleet and Freight Management

Physical Architecture Flow Name: Hazmat information request

Request for information about a particular Hazmat load.

Logical Architecture Reference Flow: cf_hazmat_request
This data flow is sent from the Manage Emergency Services function to the Manage Commercial Vehicles function and contains a request for information about hazardous materials that are being or about to be carried by commercial vehicles.

Fleet and Freight Management => Commercial Vehicle Administration

Physical Architecture Flow Name: credential application

Application for commercial vehicle credentials for a particular route/trip.

Logical Architecture Reference Flow: cf_enrollment_request
This data flow is used within the Manage Commercial Vehicles function and contains the data needed to obtain enrollment information for a particular commercial vehicle cargo, type and weight on a particular route as provided by the commercial fleet manager. It consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_route_data + cv_route_number + cv_trip_classification_data + cv_trip_identity + route_type + border_crossing_request.

Logical Architecture Reference Flow: cv_enrollment_request
This data flow is used within the Manage Commercial Vehicles function and contains the data needed to obtain enrollment information for a particular commercial vehicle cargo, type and weight on a particular route as provided by the commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_route_data + cv_route_number + cv_trip_classification_data + cv_trip_identity + route_type + border_crossing_request.

Physical Architecture Flow Name: information request

General purpose information request for data stored within the commercial vehicle operations information exchange network.

Logical Architecture Reference Flow: cf_request_activity_report
This data flow is used within the Manage Commercial Vehicles function and contains a request from the commercial vehicle manager for one of two types of activity report to be provided. This may be either a specific one time report of clearance safety activity at roadside facilities, or a request that periodic summary reports to be sent on a regular basis. The reports will only relate to the carrier, driver, vehicle combination specified in the request, although it will be possible for all the vehicles and drivers related to a specific carrier to be requested. The data flow contains the following data items each of which is defined in its own DDE: cv_credentials_details + cv_roadside_activity_report_frequency + date + list_size + list_size{cv_roadside_facility_identity}.

Physical Architecture Flow Name: tax filing, audit data

Commercial vehcile tax filing and audit data.

Logical Architecture Reference Flow: cf_enrollment_payment_request
This data flow is used within the Manage Commercial Vehicles function to request payment for the enrollment of a particular commercial vehicle cargo, weight and type on a particular route by the commercial fleet manager. The number of sets of taxes and duties has been set at fourteen (14) for the definition, but has been set to one and a half (1.5) for the size expression as this is a more typical value. The data flow consists of the following items each of which is defined in its own DDE: cf_manager_credit_identity + cv_account_number + cv_route_number + 1{cv_taxes_and_duties}14.

Logical Architecture Reference Flow: cf_tax_audit_data
This data flow is used within the Manage Commercial Vehicles function. It contains tax data and audit filings not related to specific credentials application and is generated as a result of input from the commercial vehicle manager.

Logical Architecture Reference Flow: cv_enrollment_payment_request
This data flow is used within the Manage Commercial Vehicles function to request payment for the enrollment of a particular commercial vehicle cargo, weight and type on a particular route by the commercial vehicle driver acting in the role of the commercial fleet manager. The number of sets of taxes and duties has been set at fourteen (14) for the definition, but has been set to one and a half (1.5) for the size expression as this is a more typical value. The data flow consists of the following items each of which is defined in its own DDE: cv_account_number + cv_driver_credit_identity + cv_route_number + 1{cv_taxes_and_duties}14.

Fleet and Freight Management => Commercial Vehicle Manager

Physical Architecture Flow Name: fleet status

Fleet status information including enrollment status, routing information, current vehicle information, and emergency information.

Logical Architecture Reference Flow: tcvm_confirm_enrollment_data_stored
This data flow is sent to the commercial vehicle manager from the Manage Commercial Vehicles function and contains confirmation that the previously entered enrollment data has been stored in the on-board vehicle unit.

Logical Architecture Reference Flow: tcvm_data_input_request
This data flow is sent to the commercial vehicle manager from the Manage Commercial Vehicles function and contains the identities of data items needed for a route request to be made, but which have not been provided by the manager.

Logical Architecture Reference Flow: tcvm_driver_route_instructions
This data flow is sent to the commercial vehicle manager from the Manage Commercial Vehicles function. It contains the output of the current instructions that have been loaded into a store for use by a commercial vehicle driver. They will enable the driver to follow a specified route picking up and dropping off cargo along the way.

Logical Architecture Reference Flow: tcvm_enrollment_confirmation
This data flow is sent to the commercial vehicle manager from the Manage Commercial Vehicles function and contains confirmation that a request for the enrollment of a particular class of vehicle and cargo at a particular weight on a particular route has been accepted and includes a list of the required taxes and duties together with their costs.

Logical Architecture Reference Flow: tcvm_enrollment_payment_confirmation
This data flow is sent to the commercial vehicle manager from the Manage Commercial Vehicles function and contains confirmation that a payment for the enrollment of a particular class of vehicle and cargo at a particular weight on a particular route has been accepted.

Logical Architecture Reference Flow: tcvm_other_data_request
This data flow is sent to the commercial vehicle manager from the Manage Commercial Vehicles function and contains the identities of data items needed to complete all the data needed for a route to be stored in the data store of commercial fleet routes, but which have yet to be provided by the manager.

Logical Architecture Reference Flow: tcvm_output_tag_data
This data flow is sent to the commercial vehicle manager from the Manage Commercial Vehicles function It contains the output of the current contents of a commercial vehicle's type two tag, produced in response to a previous request from the manager.

Logical Architecture Reference Flow: tcvm_preclearance_results
This data flow is sent to the commercial vehicle manager from the Manage Commercial Vehicles function and contains the route number, route details and a list of roadside facilities for which preclearance has been obtained.

Logical Architecture Reference Flow: tcvm_roadside_activity_report
This data flow is sent to the commercial vehicle manager from the Manage Commercial Vehicles function. It contains the output of the commercial vehicle roadside checkstation facility logs showing the activities of a particular carrier, driver and vehicle combination.

Logical Architecture Reference Flow: tcvm_route_data
This data flow is sent to the commercial vehicle manager from the Manage Commercial Vehicles function and contains details of a route for a commercial vehicle, together with the location and type of each roadside facility along the route.

Fleet and Freight Management => Commercial Vehicle Subsystem

Physical Architecture Flow Name: fleet to driver update

Updated instructions to the driver including dispatch, routing, and special instructions

Logical Architecture Reference Flow: cf_driver_route_instructions
This data flow is used within the Manage Commercial Vehicles function and contains the driver route and loading instructions related to a particular route and driver for output to the commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cf_driver_route + cf_driver_load_details.

Logical Architecture Reference Flow: cf_request_on_board_vehicle_data
This data flow is used within the Manage Commercial Vehicles function. It contains a request from the commercial

vehicle manager for the commercial vehicle to output the on-board data it has collected, plus any general message data from the driver. The data flow consists of the following data items each of which is defined in its own DDE: cv_on_board_data_required + cv_general_input_message.

Logical Architecture Reference Flow: cf_tag_data_store_request
This data flow is used within the Manage Commercial Vehicles function. It contains a request for the current on-board commercial vehicle tag data to be sent to the process that provides the interface with the commercial vehicle manager. This data flow is sent in response to a request by the manager for the output of the current tag data.

Logical Architecture Reference Flow: cf_tag_data_store_write
This data flow is used within the Manage Commercial Vehicles function. It contains on-board commercial vehicle tag data that is loaded by the commercial vehicle manager and is used by other processes in the function. The data flow consists of the following data items each of which is defined in its own DDE: cv_credentials_details + cv_trip_identity + tag_identity.

Logical Architecture Reference Flow: cv_driver_enrollment_information
This data flow is used within the Manage Commercial Vehicles function and contains data about the taxes and duties required for a commercial vehicle to be enrolled for a particular route as provided by the commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cv_route_number + cv_taxes_and_duties.

Logical Architecture Reference Flow: cv_driver_enrollment_payment_confirmation
This data flow is used within the Manage Commercial Vehicles function to confirm that a payment of the taxes and duties for the enrollment of a particular commercial vehicle cargo, weight and type on a particular route from the commercial vehicle driver has been accepted. It consists of the following data items each of which is defined in its own DDE: cv_account_number + cv_amount_billed + cv_driver_credit_identity + cv_route_number.

Logical Architecture Reference Flow: cv_driver_route_data
This data flow is used within the Manage Commercial Vehicle function and contains data about a vehicle route requested by a commercial vehicle driver. It consists of the following data items each of which is defined in its own DDE: cv_route_data + route_type + cv_route_number.

Logical Architecture Reference Flow: cv_static_route_request
This data flow is used within the Manage Commercial Vehicles function. It contains the data from which a static route can be determined for a commercial vehicle and is supplied by the commercial vehicle driver acting as a fleet manager. The data flow consists of the following data items each of which is defined in its own DDE: constraint_on_acceptable_travel_time + constraint_on_ahs_lanes + constraint_on_eta_change + constraint_on_interstate + constraint_on_load_classification + constraint_on_urban + constraint_on_vehicle_type + destination + departure_time + desired_arrival_time + modes + origin + preferred_alternate_routes + preferred_route_segments + preferred_routes + preferred_weather_conditions.

Fleet and Freight Management => Emergency Management

Physical Architecture Flow Name: Hazmat information

Information about a particular Hazmat load including nature of the load and unloading instructions. May also include Hazmat vehicle route and route update information

Logical Architecture Reference Flow: cf_hazmat_route_information
This data flow is sent from the Manage Commercial Vehicles function to the Manage Emergency Services function and contains information about the route about to be used or planned for a commercial vehicle that will carry hazardous materials. This information may cause the Emergency Services to raise an incident for all or part of the vehicle's route. The data flow consists of the following data items each of which is define in its own DDE: cv_route_number + route.

Logical Architecture Reference Flow: cf_hazmat_vehicle_information
This data flow is sent from the Manage Commercial Vehicles function to the Manage Emergency Services function and contains information about hazardous materials that are on-board the vehicle and details of the vehicle itself. The data

flow consists of the following data items each of which is defined in its own DDE: hazmat_load_data + hazmat_vehicle_data.

Fleet and Freight Management => **Information Service Provider**

Physical Architecture Flow Name: route request

Request for a tailored route based on given constraints.

Logical Architecture Reference Flow: cf_route_request
This data flow is sent from the Manage Commercial Vehicles function to the Provide Driver and Traveler Services function. It is used to request the preparation of a dynamic route for a commercial vehicle and originates with the commercial vehicle fleet manager. The data flow consists of the following items each of which is defined in its own DDE: constraint_on_acceptable_travel_time + constraint_on_eta_change + constraint_on_load_classification + constraint_on_ahs_lanes + constraint_on_interstate + constraint_on_urban + constraint_on_vehicle_type + cv_route_number + destination + departure_time + desired_arrival_time + modes + origin + preferred_routes + preferred_alternate_routes + preferred_route_segments + preferred_weather_conditions.

Logical Architecture Reference Flow: cv_route_request
This data flow is sent from the Manage Commercial Vehicles function to the Provide Driver and Traveler Services function. It is used to request the preparation of a dynamic route for a commercial vehicle and originates with the commercial vehicle driver acting in the role of fleet manager. The data flow consists of the following items each of which is defined in its own DDE: constraint_on_acceptable_travel_time + constraint_on_eta_change + constraint_on_load_classification + constraint_on_ahs_lanes + constraint_on_interstate + constraint_on_urban + constraint_on_vehicle_type + destination + departure_time + desired_arrival_time + modes + origin + preferred_routes + preferred_alternate_routes + preferred_route_segments + preferred_weather_conditions + vehicle_identity.

Fleet and Freight Management => **Intermodal Freight Depot**

Physical Architecture Flow Name: intermod CVO coord

Cargo movement logs and cargo ID’s exchanged between freight shippers.

Logical Architecture Reference Flow: To_Intermodal_Freight_Depot
This data flow is a request for details of the movement of freight by means that may include methods other than commercial vehicles, e.g. heavy rail, air, sea, river, etc.

Fleet and Freight Management => **Intermodal Freight Shipper**

Physical Architecture Flow Name: intermod CVO coord

Cargo movement logs and cargo ID’s exchanged between freight shippers.

Logical Architecture Reference Flow: To_Intermodal_Freight_Shipper
This data flow is a request for data about the services available to ship freight by means other than commercial vehicles, e.g. heavy rail, air, sea, river, etc.

Fleet and Freight Management => **Payment Instrument**

Physical Architecture Flow Name: request for payment

Request to deduct cost of service from user's payment account.

Logical Architecture Reference Flow: tpi_debited_commercial_manager_payment
This data flow is sent to the payment instrument from the Provide Electronic Payment Services function. It is a request to deduct the cost of payment for commercial vehicle electronic credential filing and tax payment from the value of credit currently stored on the payment instrument being used by the commercial fleet manager.

Information Service Provider => **Fleet and Freight Management**

Physical Architecture Flow Name: route plan

Tailored route provided by ISP in response to a specific request.

Logical Architecture Reference Flow: cf_route
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Commercial Vehicles function . It contains details of a dynamic route provided for a commercial vehicle the request for which originated with the commercial vehicle fleet manager. The data flow consists of the following data items each of which is defined in its own DDE: cv_route_data + cv_route_number.

Logical Architecture Reference Flow: cv_route
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Commercial Vehicles function . It contains details of a dynamic route provided for a commercial vehicle the request for which originated with the commercial vehicle driver acting in the role of fleet manager. The data flow consists of the following data items each of which is defined in its own DDE: cv_route_data + vehicle_identity.

Intermodal Freight Depot => **Fleet and Freight Management**

Physical Architecture Flow Name: intermod CVO coord

Cargo movement logs and cargo ID’s exchanged between freight shippers.

Logical Architecture Reference Flow: From_Intermodal_Freight_Depot
This data flow is used within the Manage Commercial Vehicles function and contains data about the movement of freight by means that may include methods other than commercial vehicles, e.g. heavy rail, air, sea, river, etc.

Intermodal Freight Shipper => **Fleet and Freight Management**

Physical Architecture Flow Name: intermod CVO coord

Cargo movement logs and cargo ID’s exchanged between freight shippers.

Logical Architecture Reference Flow: From_Intermodal_Freight_Shipper
This data flow is used within the Manage Commercial Vehicles function and contains data about the services available to ship freight by means other than commercial vehicles, e.g. heavy rail, air, sea, river, etc.

Payment Instrument => **Fleet and Freight Management**

Physical Architecture Flow Name: payment

Payment of some kind (e.g., toll, parking, fare) by traveler which in most cases can be related to a credit account.

Logical Architecture Reference Flow: fpi_commercial_manager_input_credit_identity

Fleet and Freight Management (FMS)

This data flow is sent from the payment instrument to the Provide Electronic Payment Services function. It contains the data necessary to allow automatic billing of the user of the payment instrument when they are the commercial vehicle fleet manager, working in an office environment. A payment instrument is a device that can be used to make payments, e.g. a debit card, a credit card. It will belong to the financial institution responsible for its issue and not to the user.

2.10.4 Subsystem Architecture Flow Diagram

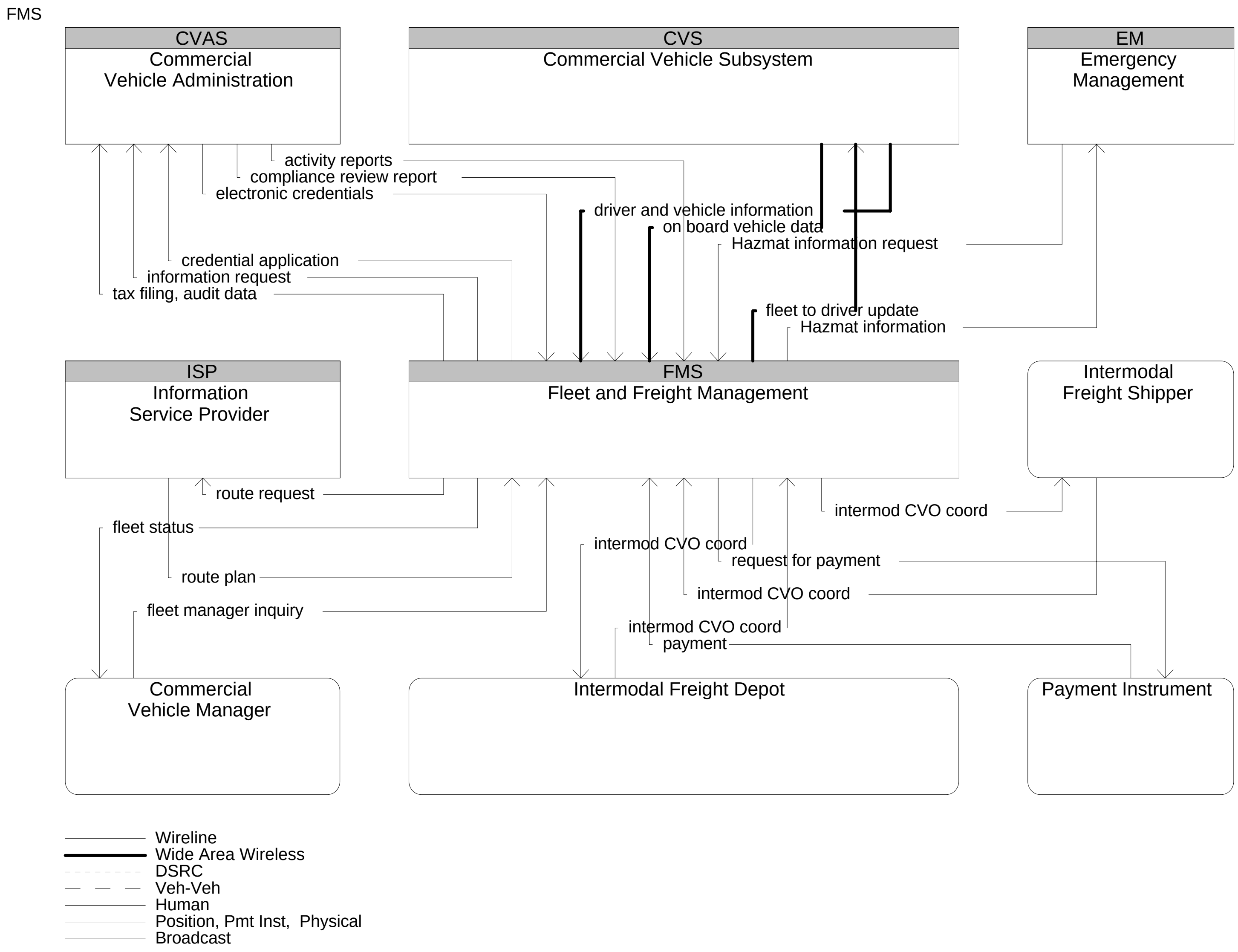


Figure 2.10-2 Architecture Flow Diagram for FMS

2.11 Information Service Provider

This subsystem collects, processes, stores, and disseminates transportation information to system operators and the traveling public. The subsystem can play several different roles in an integrated ITS. In one role, the ISP provides a general data warehousing function, collecting information from transportation system operators and redistributing this information to other system operators in the region and other ISPs. In this information redistribution role, the ISP provides a bridge between the various transportation systems that produce the information and the other ISPs and their subscribers that use the information. The second role of an ISP is focused on delivery of traveler information to subscribers and the public at large. Information provided includes basic advisories, real time traffic condition and transit schedule information, yellow pages information, ridematching information, and parking information. The subsystem also provides the capability to provide specific directions to travelers by receiving origin and destination requests from travelers, generating route plans, and returning the calculated plans to the users. In addition to general route planning for travelers, the ISP also supports specialized route planning for vehicle fleets. In this third role, the ISP function may be dedicated to, or even embedded within, the dispatch system. Reservation services are also provided in advanced implementations. The information is provided to the traveler through the Personal Information Access Subsystem, Remote Traveler Support Subsystem, and various Vehicle Subsystems through available communications links. Both basic one-way (broadcast) and personalized two-way information provision is supported. The subsystem provides the capability for an informational infrastructure to connect providers and consumers, and gather that market information needed to assist in the planning of service improvements and in maintenance of operations.

2.11.1 Alternative Configurations

The Information Service Provider subsystem is responsible for dissemination of information to travelers, supplying route planning and customized route planning for travelers, vehicles, and cargo, and has the capability to collect probe data from cooperative travelers Figure 2.11-1. The subsystem may be integrated with other centers to provide the indicated functionality or may be a separate entity specializing in one, some, or all of the above functions. An ISP integrated with traffic management disseminates traffic information to the public and media. An ISP integrated with a transit organization disseminates information regarding transit schedules. An independent ISP may integrate traffic, yellow pages, transit, and route planning together in a pay for use service.

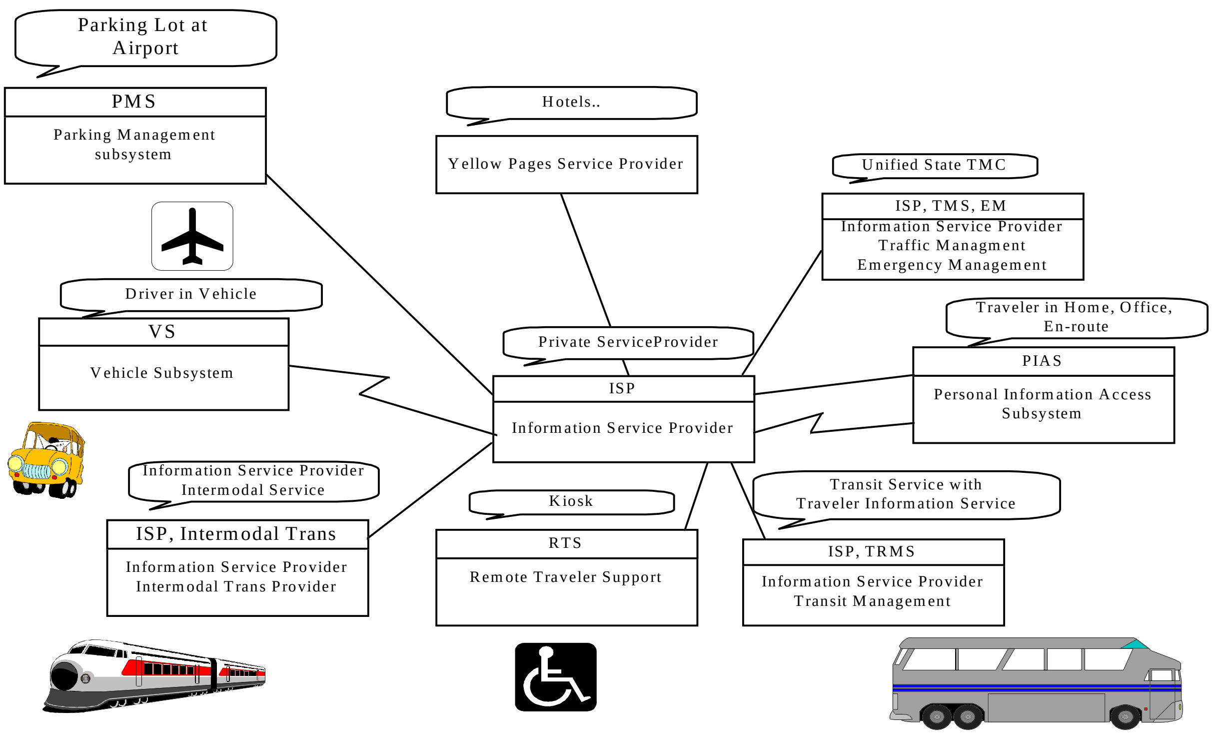


Figure 2.11-1 Alternative Configurations for an ISP

2.11.2 Subsystem Equipment Packages and Supporting Process Specifications for ISP

Basic Information Broadcast
This Equipment package provides the capabilities to collect, process, store, bill, and disseminate traveler information including traveler, transit, ridematching, traffic, and parking information. The traveler information shall include maintaining a database of local area services available to travelers with up-to-the-minute information and providing an interactive connectivity between, sponsors, and providers of services. The transit information shall include the latest available information on transit routes and schedules, transit transfer options, transit fares, and real-time schedule adherence. The traffic information shall include latest available information on traffic and highway conditions, and current situation information in real-time including incidents, road construction, recommended routes, current speeds on specific routes, current parking conditions in key areas, schedules for any current or soon to start events, and current weather situations. This Equipment package shall also provide users with real-time travel related information while they are traveling, and disseminate to assist the travelers in making decisions about transfers and modification of trips. These capabilities shall be provided using equipment such as a fixed facility with a communications system such as a data subcarrier multiplexing device.

Process Specifications
1.1.4.6 Provide Traffic Data Retrieval Interface

Overview: This process shall provide customized sets of traffic data for broadcast, advisories, and personalized data to travelers and the media. This process shall use the parameters in the data store 'traffic_data_retreival_parameters' to define exactly what data shall be retrieved as a result of each request. The process shall select the appropriate subset of traffic data which will be sent to each ITS function which is requesting data. The process shall accept traveler profiles for use in determining what personalized data to send to the traveler.

6.2.1.3 Collect Transit Data for Advisory Messages

Overview: This process shall collect and fuse transit advisory data that will be used to create broadcast or advisory messages to travelers. The process shall extract from the data those elements appropriate for advisory or broadcast messages and load it into the traveler_transit_information_data store. The data can be provided to the process either via direct request from the process or as a result of periodic (unrequested) updates. The process shall fuse all the received data into a coherent set, which is loaded into a traveler_transit_information_data store for access by other processes.

6.2.1.4 Provide Traffic and Transit Broadcast Messages

Overview: This process shall extract advisory data from stores of traveler traffic and transit information at locally determined intervals and send it out to drivers or transit users in vehicles as wide area broadcast messages. The content and rate of these messages shall be based upon parameters from the broadcast_parameters_data store, which is managed by the ISP operator.

6.2.1.5 Provide ISP Operator Broadcast Parameters Interface

Overview: This process shall provide the interface through which the ISP operator can manipulate data in the broadcast_parameters_data store. The data, in this store, shall be used by another process to define the scope and rate of wide area broadcast messages to vehicles. The process shall provide the ISP operator with the ability to request parameter data output and/or update the data store with new parameter values.

6.2.4 Collect Yellow Pages Data

Overview: This process shall collect and fuse data about (yellow pages) services in order to provide information to users in vehicles. The process shall fuse all the received yellow pages data into a coherent set and loaded into the yellow_pages_information_data store for access by processes in response to requests from users in vehicles.

EM Route Plan Information Dissemination

This Equipment package provides route plan information for the Emergency Management Subsystem. Routes are computed based on the request for route plan and current traffic conditions. Special algorithms are employed which take into account the special needs and capabilities (e.g., traveling along normally non-navigable links) of dispatched emergency vehicles. Special security, logging, and performance requirements may also be associated with the operation of this equipment package.

Process Specifications

6.6.2.2 Provide Vehicle Route Calculation Data

Overview: This process shall update the data stores containing information which is used by the another process to calculate vehicle routes. This process shall also provide data about links (speed or travel times), and queues to be broadcast to vehicles (to support autonomous guidance with dynamic link updates). The process shall fuse link and queue data received from Manage Traffic sources with probe data received from vehicles under infrastructure based route guidance, or with probe data obtained from other sources (such as from an electronic toll collection system). The process shall obtain route segment data as requested data or as data received at periodic intervals from other ITS functions. The process shall have the capability to request data about route segments outside its own area by sending a data request to the process that provides the interface with other ISP's. Link addresses, mapped to other ISPs, will be maintained by this process and stored in the link_data_store.

Infrastructure Provided Dynamic Ridesharing

This Equipment package shall have as prerequisite the capabilities of the Interactive Infrastructure Information Equipment package. In addition, this Equipment package provides the capability to provide specific dynamic ridesharing, including rider and driver information and reservations.

Process Specifications

6.1.1 Provide Trip Planning Information to Traveler

Overview: This process shall obtain all the information needed to fulfill the traveler's request for a trip. The process shall support the request for trips that require the use of one or more modes of transport, and shall use the preferences and constraints specified by the traveler in the trip request, plus data from the store of trip planning parameters, to select the most appropriate modes. It shall send details of the trip requirements to the specialized processes that provide route information for the different modes of transport. When route data is received back from these processes, this process shall ensure that the whole trip is covered by one coherent route for which all the data such as costs, arrival times, and modal change points are known. The information provided to the traveler by the process shall be sufficient to enable the traveler to understand the routing, modes and cost of the trip. The trip information shall be stored for possible use in subsequent trip confirmation. This process shall exchange all input and output data from and to the traveler with the appropriate traveler interface process.

6.1.2 Confirm Traveler's Trip Plan

Overview: This process shall confirm a trip previously requested by a traveler and any financial transactions that this may require. The process shall base the trip confirmation upon information created by the process responsible for trip planning and stored locally. Confirmation details shall be sent to specialized processes (such as those responsible for demand responsive transit, ridesharing, etc.) to make reservations for their services. The response to these reservation requests and any necessary payment transactions shall be sent to the traveler. This process shall exchange all input and output data to and from the traveler via the appropriate traveler interface process.

6.1.4 Provide ISP Operator Interface for Trip Planning Parameters

Overview: This process shall manage the data store containing parameters used by the trip planning processes. These parameters shall govern the way in which multimodal trips are planned by other processes within Provide Trip Planning Services. This process shall accept inputs from the ISP Operator to define or update trip planning parameters. This process shall output these trip planning parameters to the ISP Operator.

6.2.4 Collect Yellow Pages Data

Overview: This process shall collect and fuse data about (yellow pages) services in order to provide information to users in vehicles. The process shall fuse all the received yellow pages data into a coherent set and loaded into the yellow_pages_information_data store for access by processes in response to requests from users in vehicles.

6.4.1 Screen Rider Requests

Overview: This process shall accept and screen traveler requests for ride-sharing. These requests shall be sent to the process as a result of trip plan requests received from travelers by other processes. This process shall use eligibility data from a local rideshare_data store, to screen travelers before they are matched with other travelers and to enable ridesharing for all or part of their proposed trips.

6.4.2 Match Rider and Provider

Overview: This process shall match travelers for ridesharing trips. The process shall attempt to achieve a match by considering some or all of the following: the origin and destination of the traveler's proposed trip, any routing constraints, preferences specified by the traveler, compatibility of this rideshare with rideshares confirmed by other travelers, the requesting traveler's eligibility data, and traffic data obtained on request from the Manage Traffic function. The process shall consider the possible disbenefits to other travelers who will be part of the same rideshare when finding the rideshare best suited to the traveler’s requirements. The process shall store data about selected rideshares in the rideshare_data store, and shall update the data when confirmation of the rideshare acceptance is received from another process.

6.4.3 Report Ride Match Results to Requestor

Overview: This process shall report ridesharing match results to requesters. The data for the results shall be provided to this process by other processes responsible for assessing traveler eligibility, and the actual match with travelers in other rideshares. The process shall output data indicating a failure when either the data from the eligibility process shows a failure, or no ridesharing match can be found. The process shall also determine that no ridesharing match can be found if no match is found between the traveler's rideshare request and the rideshare data provided as input to it by another process. When a successful match is found, the process shall output the rideshare data to the process from which the traveler's request was received.

6.4.4 Confirm Traveler Rideshare Request

Overview: This process shall confirm the traveler's rideshare match and initiate a payment transaction where appropriate. The process shall send the payment transaction data for action by a process in the Provide Electronic Payment Services function. The results of this transaction shall be sent by this process to the process providing the overall trip confirmation. Once a rideshare match is confirmed, this data is sent to the rideshare match process where it can be factored in to subsequent matches.

6.6.2.1 Calculate Vehicle Route

Overview: This process shall calculate trip planning and real-time dynamic guidance routes for all types of vehicles. The route data provided by the process in response to requests from vehicles using infrastructure based in-vehicle guidance shall only contain data necessary for the vehicle to provide guidance (since the data is intended for use by an in-vehicle navigation unit). The route provided for trip planning purposes shall contain data in a form which can be presented to a user via display (or alternatively in audio form). The process shall select the route according to the data included in the route request. Data provided by the requesting process includes preferences and constraints. The process shall have the capability of using current and/or predicted conditions of the road network in route calculation. The process shall have the capability of including additional factors such as current or predicted weather in the calculation of route. If the process cannot find the data it needs in the route_segment_details_data store, it shall request the process responsible for providing route calculation data to obtain it from the appropriate source. The process shall have the capability of outputting routes for special priority vehicles to the Manage Traffic function so that signal preemption could be provided for the special priority vehicle. The process shall send details of routes for commercial vehicles with hazardous or unusual loads to the Manage Incidents function for monitoring (as a potential, or a predicted incident).

6.6.2.3 Provide Route Segment Data for Other Areas

Overview: This process shall obtain from another ISP current or predicted data for road links that are outside the area served by the local supplier. This area, which may be defined on a geographic or jurisdictional basis, is the portion of the transportation network on which the ISP maintains real time information. Identification of which ISP to contact is based upon a store that maps a link to the ISP which maintains real time information about this link. If there is no map to another ISP in the data store, then the process will return default or static data for the link(s). This process shall also respond to similar requests from other ISPs for real time data on links within the local database.

6.6.2.4 Update Vehicle Route Selection Map Data

Overview: This process shall provide the interface to map update providers, or other appropriate data sources, through which updates of the digitized map data can be obtained. The process shall request new data from the provider on request from the ISP operator interface process. The data received from the supplier shall be loaded into a the map_data_for_route_selection data store by the process in such a way that it can be easily used by the route selection process in determining vehicle routes, trip planning, and on-line vehicle guidance.

6.6.3 Update Other Routes Selection Map Data

Overview: This process shall provide the interface to a map update providers through which to obtain fresh updates of digitized map data used in identification of non-vehicle portions of routes. The process shall request new data from the provider on request from the ISP operator interface process. The data received from the supplier shall be loaded into the map_data_for_general_use data store by the process in such a way that it can be easily used by the route selection process in determining non-vehicle routes for use in on-line traveler guidance and trip planning.

7.4.1.8 Process Traveler Rideshare Payments

Overview: This process shall be responsible for transacting payments for ridesharing that are required for the confirmation of a traveler's trip. The process shall start the transaction by receiving data from a process in the Provide Driver and Traveler Services function and shall send the data to the appropriate financial institution. The process shall send the response from the financial institution to the requesting process and shall send details of the transaction to another process for entry into a store of transaction records.

Infrastructure Provided Route Selection
This Equipment package shall have as prerequisite the capabilities of the Interactive Infrastructure Information Equipment package. In addition, this Equipment package provides the capability to provide specific directions to travelers by receiving origin and destination requests from travelers, generating route plans, returning the calculated plans to the users, and then potentially logging the route plans with Traffic Management Subsystem. This additional capability shall be provided using equipment such as a workstation type processor and software for route planning and traffic measurements along with additional communications capabilities including dialup lines, PCS telephones, and wireless data transceivers.

Process Specifications
1.1.4.6 Provide Traffic Data Retrieval Interface

Overview: This process shall provide customized sets of traffic data for broadcast, advisories, and personalized data to travelers and the media. This process shall use the parameters in the data store 'traffic_data_retreival_parameters' to define exactly what data shall be retrieved as a result of each request. The process shall select the appropriate subset of traffic data which will be sent to each ITS function which is requesting data. The process shall accept traveler profiles for use in determining what personalized data to send to the traveler.

6.6.1 Provide multimodal Route Selection

Overview: This process shall manage the creation of multimodal routes (those with one or more modes in them) in response to traveler's trip or route requests. It shall support on-line route guidance for travelers using personal devices, route guidance for vehicles, selection of specialized vehicle based routes for other ITS functions, (such as Manage Emergency Services and Manage Commercial Vehicles), and selection of multimodal routes in response to trip planning requests from travelers. The multimodal routes provided by the process shall take account of the traveler's preferences and constraints. Constraints can include the access needs of those with disabilities. Preferences can include minimizing waiting time at modal interchange points, level of traveler security, or minimum cost.

6.6.2.1 Calculate Vehicle Route

Overview: This process shall calculate trip planning and real-time dynamic guidance routes for all types of vehicles. The route data provided by the process in response to requests from vehicles using infrastructure based in-vehicle guidance shall only contain data necessary for the vehicle to provide guidance (since the data is intended for use by an in-vehicle navigation unit). The route provided for trip planning purposes shall contain data in a form which can be presented to a user via display (or alternatively in audio form). The process shall select the route according to the data included in the route request. Data provided by the requesting process includes preferences and constraints. The process shall have the capability of using current and/or predicted conditions of the road network in route calculation. The process shall have the capability of including additional factors such as current or predicted weather in the calculation of route. If the process cannot find the data it needs in the route_segment_details_data store, it shall request the process responsible for providing route calculation data to obtain it from the appropriate source. The process shall have the capability of outputing routes for special priority vehicles to the Manage Traffic function so that signal preemption could be provided for the special priority vehicle. The process shall send details of routes for commercial vehicles with hazardous or unusual loads to the Manage Incidents function for monitoring (as a potential, or a predicted incident).

6.6.2.1 Calculate Vehicle Route

Overview: This process shall calculate trip planning and real-time dynamic guidance routes for all types of vehicles. The route data provided by the process in response to requests from vehicles using infrastructure based in-vehicle guidance shall only contain data necessary for the vehicle to provide guidance (since the data is

intended for use by an in-vehicle navigation unit). The route provided for trip planning purposes shall contain data in a form which can be presented to a user via display (or alternatively in audio form). The process shall select the route according to the data included in the route request. Data provided by the requesting process includes preferences and constraints. The process shall have the capability of using current and/or predicted conditions of the road network in route calculation. The process shall have the capability of including additional factors such as current or predicted weather in the calculation of route. If the process cannot find the data it needs in the route_segment_details_data store, it shall request the process responsible for providing route calculation data to obtain it from the appropriate source. The process shall have the capability of outputting routes for special priority vehicles to the Manage Traffic function so that signal preemption could be provided for the special priority vehicle. The process shall send details of routes for commercial vehicles with hazardous or unusual loads to the Manage Incidents function for monitoring (as a potential, or a predicted incident).

6.6.2.2 Provide Vehicle Route Calculation Data

Overview: This process shall update the data stores containing information which is used by the another process to calculate vehicle routes. This process shall also provide data about links (speed or travel times), and queues to be broadcast to vehicles (to support autonomous guidance with dynamic link updates). The process shall fuse link and queue data received from Manage Traffic sources with probe data received from vehicles under infrastructure based route guidance, or with probe data obtained from other sources (such as from an electronic toll collection system). The process shall obtain route segment data as requested data or as data received at periodic intervals from other ITS functions. The process shall have the capability to request data about route segments outside its own area by sending a data request to the process that provides the interface with other ISP's. Link addresses, mapped to other ISPs, will be maintained by this process and stored in the link_data_store.

6.6.2.3 Provide Route Segment Data for Other Areas

Overview: This process shall obtain from another ISP current or predicted data for road links that are outside the area served by the local supplier. This area, which may be defined on a geographic or jurisdictional basis, is the portion of the transportation network on which the ISP maintains real time information. Identification of which ISP to contact is based upon a store that maps a link to the ISP which maintains real time information about this link. If there is no map to another ISP in the data store, then the process will return default or static data for the link(s). This process shall also respond to similar requests from other ISPs for real time data on links within the local database.

6.6.2.4 Update Vehicle Route Selection Map Data

Overview: This process shall provide the interface to map update providers, or other appropriate data sources, through which updates of the digitized map data can be obtained. The process shall request new data from the provider on request from the ISP operator interface process. The data received from the supplier shall be loaded into a the map_data_for_route_selection data store by the process in such a way that it can be easily used by the route selection process in determining vehicle routes, trip planning, and on-line vehicle guidance.

6.6.2.5 Provide ISP Operator Route Parameters Interface

Overview: This process shall provide the interface through which the ISP operator can input and update route calculation parameters used by the Provide Driver and Traveler Services function. The process shall provide an interface through which the ISP operator can review and request update of map data. The operator shall be able to use the process to request digitized map updates from suppliers, request output of trip planning and route selection control parameters, or to update the control parameters in the route_selection_parameters data store. The process shall support inputs from the ISP operator in manual or audio form, and shall provide its outputs in audible or visual forms. It shall enable the visual output to be in hardcopy, and/or display.

6.6.3 Update Other Routes Selection Map Data

Overview: This process shall provide the interface to a map update providers through which to obtain fresh updates of digitized map data used in identification of non-vehicle portions of routes. The process shall request new data from the provider on request from the ISP operator interface process. The data received from the supplier shall be loaded into the map_data_for_general_use data store by the process in such a way that it can be easily used by the route selection process in determining non-vehicle routes for use in on-line traveler guidance and trip planning.

6.6.4 Select Transit Route

Overview: This process shall determine routes that are based on regular transit services. Routes shall be provided by the process to travelers in response to trip planning and on-line personal guidance requests. The data provided by the process shall be different for the two types of requests since trip planning information will not need the detail that guidance data requires. The process shall base routes on the current state of the regular transit services using data obtained from processes in the Manage Transit function. It shall also respond to any preferences and constraints, such as those for travelers with special needs, that are specified in the route request. Data on the current use of transit routes in on-line guidance shall be provided by the process to the Manage Demand function to aid in demand management. This data on current use of the transit routes in on-line guidance is stored in the transit_mode_routes data store.

6.6.5 Select Other Routes

Overview: This process shall determine routes, or portions of routes, not based on use of vehicles or regular transit services. Routes shall be provided by the process for travelers in response to trip planning, and on-line personal guidance requests. Data provided by the process will be different for the two types of requests since the data for trip planning will not need the level of detailed that guidance data requires. The process shall calculate its routes using digitized map data obtained and updated by another process. It shall make use of the alternative modes, (such as ferries, walking, cycling, etc.) that have been specified in the route request, and shall also take account of any preferences and constraints, (such as those for travelers with special needs). Data on current use of routes in on-line guidance shall be provided by the process to the Manage Demand function.

Infrastructure Provided Yellow Pages & Reservation

This Equipment package shall have as prerequisite the capabilities of the Interactive Infrastructure Information Equipment package. In addition, this Equipment package provides the capability to provide specific traveler information, such as Yellow Pages information, with reservation capabilities.

Process Specifications

6.1.1 Provide Trip Planning Information to Traveler

Overview: This process shall obtain all the information needed to fulfill the traveler's request for a trip. The process shall support the request for trips that require the use of one or more modes of transport, and shall use the preferences and constraints specified by the traveler in the trip request, plus data from the store of trip planning parameters, to select the most appropriate modes. It shall send details of the trip requirements to the specialized processes that provide route information for the different modes of transport. When route data is received back from these processes, this process shall ensure that the whole trip is covered by one coherent route for which all the data such as costs, arrival times, and modal change points are known. The information provided to the traveler by the process shall be sufficient to enable the traveler to understand the routing, modes and cost of the trip. The trip information shall be stored for possible use in subsequent trip confirmation. This process shall exchange all input and output data from and to the traveler with the appropriate traveler interface process.

6.1.2 Confirm Traveler's Trip Plan

Overview: This process shall confirm a trip previously requested by a traveler and any financial transactions that this may require. The process shall base the trip confirmation upon information created by the process responsible for trip planning and stored locally. Confirmation details shall be sent to specialized processes (such as those responsible for demand responsive transit, ridesharing, etc.) to make reservations for their services. The response to these reservation requests and any necessary payment transactions shall be sent to the traveler. This process shall exchange all input and output data to and from the traveler via the appropriate traveler interface process.

6.1.3 Manage Intermodal Service Provider Interface

Overview: This process shall collect data about services that are available to travelers from intermodal service providers. These suppliers shall be those that provide travel services that are not part of regular transit or demand responsive transit operations, e.g. heavy rail, and may not involve surface transportation, e.g. ferry and airline operations. The process shall provide data formatted for use as part of a traveler's multimodal trip, and shall support subsequent confirmation of any portion provided by the Intermodal Service Provider.

6.2.1.2 Provide Traffic and Transit Advisory Messages

Overview: This process shall provide advisory data to users in vehicles (drivers or transit users) as a result of a request from the driver or transit user. (e.g. This process supports a request/response type of exchange with the user.) The advisory information is extracted from the data stores of traveler traffic and transit information. The process shall have the capability to filter the advisory data, read from the data stores, store so that the output only contains data that is relevant to the current location of the vehicle from which the request was made. When the user requests location specific data, the vehicle's location shall be provided to the process in the request message.

6.2.4 Collect Yellow Pages Data

Overview: This process shall collect and fuse data about (yellow pages) services in order to provide information to users in vehicles. The process shall fuse all the received yellow pages data into a coherent set and loaded into the yellow_pages_information_data store for access by processes in response to requests from users in vehicles.

6.2.6 Provide Yellow Pages Data and Reservations

Overview: This process shall extract data from the yellow_pages_information_data store upon request for data from the driver or a transit user in a vehicle. The data read from the store may be filtered, by the process, so that output only contains that which is relevant to the current location of the vehicle. The process shall also enable the user to make reservations for yellow pages services from a vehicle.

6.5.2 Provide Traveler Yellow Pages Information and Reservations

Overview: This process shall provide information and reservation services obtained from yellow pages service providers. The process shall provide the information and reservation data so that it can easily form part of a traveler's information request or trip planning activities. The process shall be able to request additional yellow pages information if the process cannot find the required data in the tourist_information data store. The process shall send requests for payment to a process in the Provide Electronic Payment Services function for action, and shall send the response back to the process from which the payment request was received.

6.5.2 Provide Traveler Yellow Pages Information and Reservations

Overview: This process shall provide information and reservation services obtained from yellow pages service providers. The process shall provide the information and reservation data so that it can easily form part of a traveler's information request or trip planning activities. The process shall be able to request additional yellow pages information if the process cannot find the required data in the tourist_information data store. The process shall send requests for payment to a process in the Provide Electronic Payment Services function for action, and shall send the response back to the process from which the payment request was received.

6.5.3 Register Yellow Pages Service Providers

Overview: This process shall register yellow pages service providers. The process shall accept requests for registration from the providers and shall pass the data to a process in the Provide Electronic Payment Services function for action. The process shall send the result of this action to the provider, and if successful, shall send a request for the process that manages the contents of the store of tourist information to request data from the provider. The details of the provider shall also be loaded into the store used by that process, so that data from the provider can readily be obtained in the future. This process shall perform updating of the yellow pages service provider details.

6.6.2.3 Provide Route Segment Data for Other Areas

Overview: This process shall obtain from another ISP current or predicted data for road links that are outside the area served by the local supplier. This area, which may be defined on a geographic or jurisdictional basis, is the portion of the transportation network on which the ISP maintains real time information. Identification of which ISP to contact is based upon a store that maps a link to the ISP which maintains real time information about this link. If there is no map to another ISP in the data store, then the process will return default or static data for the link(s). This process shall also respond to similar requests from other ISPs for real time data on links within the local database.

6.6.2.4 Update Vehicle Route Selection Map Data

Overview: This process shall provide the interface to map update providers, or other appropriate data sources, through which updates of the digitized map data can be obtained. The process shall request new data from the provider on request from the ISP operator interface process. The data received from the supplier shall be loaded into a the map_data_for_route_selection data store by the process in such a way that it can be easily used by the route selection process in determining vehicle routes, trip planning, and on-line vehicle guidance.

7.1.6 Distribute Advanced Charges and Fares

Overview: This process shall be responsible for receiving requests for advanced payment of tolls from the parking lot charge or transit fare collection facilities within the Provide Electronic Payment Services function. It shall pass the requests on to another process in the toll collection facility, and shall return transaction success or failure details to the requesting process. The process shall also receive requests for the advanced payment of parking lot charges and transit fares from the toll payment interface process. It shall send these requests to other processes in the Provide Electronic Payment Services function and when received, return the results to the toll payment interface process.

7.2.6 Distribute Advanced Tolls and Fares

Overview: This process shall be responsible for receiving requests for advanced payment of parking lot charges from the toll or transit fare collection facilities within the Provide Electronic Payment Services function. It shall pass the requests on to another process in the Provide Electronic Parking Lot Payment facility, and shall return transaction success or failure details to the requesting process. The process shall also receive requests for the advanced payment of tolls and transit fares from the parking lot payment interface process. It shall send these requests to other processes in the Provide Electronic Payment Services function and when received, return the results to the Parking Lot payment interface process.

7.3.2 Distribute Advanced Tolls and Parking Lot Charges

Overview: This process shall be responsible for receiving requests for advanced payment of transit fares from the toll and parking lot charge collection facilities within the Provide Electronic Payment Services function. It shall pass the advanced fare requests on to another process in the Process Electronic Transit Fare Payment facility, and when received, shall return transit success or failure details to the requesting process. The process shall also receive requests for advanced payment of tolls and parking lot charges from transit vehicle and roadside (transit stop) fare collection facilities. It shall send these requests to other processes in the Provide Electronic Payment Services function and when received, return the results to the requesting process.

7.4.1.2 Process Yellow Pages Services Provider Payments

Overview: This process shall be responsible for transacting payments for the registration of other (yellow pages) service providers. The process shall be initiated by receiving data from a process in the Provide Driver and Traveler Services function and shall send the data to the financial institution. The process shall send the response from the financial institution to the requesting process and shall send details of the transaction to another process for entry into a store of transaction records.

7.4.1.3 Process Driver Map Update Payments

Overview: This process shall be responsible for transacting payments from the driver for updates to the navigable map database in the vehicle. The process shall receive the transaction request data from a process in the Provide Driver and Traveler Services function and shall send the data to the financial institution for action. The process shall send the response from the financial institution to the requesting process and shall send details of the transaction to another process for entry into the payment_transaction_records data store.

7.4.1.4 Process Traveler Map Update Payments

Overview: This process shall be responsible for transacting payments from the traveler for updates to the navigable map database carried in the personal device. The process shall receive the transaction request data from a process in the Provide Driver and Traveler Services function and shall send the data to the financial institution. The process shall send the response from the financial institution to the requesting process and shall send details of the transaction to another process for entry into the payment_transaction_records data store.

7.4.1.6 Process Traveler Trip and Other Services Payments

Overview: This process shall be responsible for transacting advance payments required for the confirmation of a trip by a traveler. Payments supported by the process shall comprise those for any tolls, parking lot charges, transit fares, or other (yellow pages) services that need to be paid for the trip to be confirmed. The process shall receive the transaction data from a process in the Provide Driver and Traveler Services function and shall send the data to the financial institution. Tolls, fares and parking lot charges are sent to the Route Traveler Advanced Payment function for processing. The process shall send the response from the financial institution to the requesting process and shall send details of the transaction to another process for entry into the payment_transaction_records data store.

7.4.2 Collect Price Data for ITS Use

Overview: This process shall be responsible for collecting data on the prices being charged for tolls, parking lots and transit fares. This process shall accept data sent to it by the other processes when they have updated their data and automatically sent it, or this process shall request a transfer of data from the other processes. The process shall load the data into the price_data_for_services data store from which some or all of it can be read on request from processes in other ITS functions.

7.4.3 Route Traveler Advanced Payments

Overview: This process shall be responsible for receiving a traveler's request for advanced payment (for tolls, parking lot charges, and/or transit fares) and routing it to the appropriate part of the Provide Electronic Payment Services function. The process shall also receive responses to the advanced payment requests and shall return them to the originating process.

Interactive Infrastructure Information

This Equipment package shall have as prerequisite the capabilities of the Basic Information Broadcast Equipment package. This Equipment package augments the Basic Information Broadcast Equipment package by providing the capabilities for interactive traveler information.

Process Specifications

1.1.4.5 Provide Media System Traffic Data Interface

Overview: This process shall provide the interface through which traffic and incident data can be output to the Media. The output shall comprise traffic and incident data that is suitable for output to the Media System as determined by traffic managers. This interface is only for the output of data that has been requested by the Media.

1.1.4.6 Provide Traffic Data Retrieval Interface

Overview: This process shall provide customized sets of traffic data for broadcast, advisories, and personalized data to travelers and the media. This process shall use the parameters in the data store 'traffic_data_retreival_parameters' to define exactly what data shall be retrieved as a result of each request. The process shall select the appropriate subset of traffic data which will be sent to each ITS function which is requesting data. The process shall accept traveler profiles for use in determining what personalized data to send to the traveler.

4.1.8 Provide Transit Operations Data Distribution Interface

Overview: This process shall provide customized sets of transit vehicle schedule deviations to travelers and to the media. The process shall only provide data to the media when prompted by the arrival of new deviation data in the transit_vehicle_operational_data store, which is maintained by another process in the Manage Transit function. The outputs shall be made available following a direct request from the other ITS function, or as a part of subscription process relating to a travelers transit profile. The process shall obtain the required data from the process that manages the store of transit vehicle operating data.

6.1.1 Provide Trip Planning Information to Traveler

Overview: This process shall obtain all the information needed to fulfill the traveler's request for a trip. The process shall support the request for trips that require the use of one or more modes of transport, and shall use the preferences and constraints specified by the traveler in the trip request, plus data from the store of trip planning parameters, to select the most appropriate modes. It shall send details of the trip requirements to the specialized processes that provide route information for the different modes of transport. When route data is received back from these processes, this process shall ensure that the whole trip is covered by one coherent route for which all the data such as costs, arrival times, and modal change points are known. The information provided to the traveler by the process shall be sufficient to enable the traveler to understand the routing, modes and cost of the trip. The trip information shall be stored for possible use in subsequent trip confirmation. This process shall exchange all input and output data from and to the traveler with the appropriate traveler interface process.

6.1.2 Confirm Traveler's Trip Plan

Overview: This process shall confirm a trip previously requested by a traveler and any financial transactions that this may require. The process shall base the trip confirmation upon information created by the process responsible for trip planning and stored locally. Confirmation details shall be sent to specialized processes (such as those responsible for demand responsive transit, ridesharing, etc.) to make reservations for their services. The response to these reservation requests and any necessary payment transactions shall be sent to the traveler. This process shall exchange all input and output data to and from the traveler via the appropriate traveler interface process.

6.1.3 Manage Intermodal Service Provider Interface

Overview: This process shall collect data about services that are available to travelers from intermodal service providers. These suppliers shall be those that provide travel services that are not part of regular transit or demand responsive transit operations, e.g. heavy rail, and may not involve surface transportation, e.g. ferry and airline operations. The process shall provide data formatted for use as part of a traveler's multimodal trip, and shall support subsequent confirmation of any portion provided by the Intermodal Service Provider.

6.1.4 Provide ISP Operator Interface for Trip Planning Parameters

Overview: This process shall manage the data store containing parameters used by the trip planning processes. These parameters shall govern the way in which multimodal trips are planned by other processes within Provide Trip Planning Services. This process shall accept inputs from the ISP Operator to define or update trip planning parameters. This process shall output these trip planning parameters to the ISP Operator.

6.2.1.1 Collect Traffic Data for Advisory Messages

Overview: This process shall collect and fuse traffic data that will be used to create broadcast or advisory messages to travelers. The input data for this process shall consist of historical, current, and predicted traffic and incident data. The process shall extract from the data those elements appropriate for advisory or broadcast messages and load it into the store of traveler_traffic_information_data store. The data can be provided to the process either via direct request from the process or as a result of periodic (unrequested) updates.

6.2.1.2 Provide Traffic and Transit Advisory Messages

Overview: This process shall provide advisory data to users in vehicles (drivers or transit users) as a result of a request from the driver or transit user. (e.g. This process supports a request/response type of exchange with the user.) The advisory information is extracted from the data stores of traveler traffic and transit information. The process shall have the capability to filter the advisory data, read from the data stores, store so that the output only contains data that is relevant to the current location of the vehicle from which the request was made. When the user requests location specific data, the vehicle's location shall be provided to the process in the request message.

6.2.4 Collect Yellow Pages Data

Overview: This process shall collect and fuse data about (yellow pages) services in order to provide information to users in vehicles. The process shall fuse all the received yellow pages data into a coherent set and loaded into the yellow_pages_information_data store for access by processes in response to requests from users in vehicles.

6.5.1 Collect and Update Traveler Information

Overview: This process shall collect and update data about incidents, road construction, weather, events and yellow pages data. This data shall be obtained by the process from other ITS functions and from outside sources such as the weather service, yellow pages service providers and the media. The process shall load the data into a local store for use by the process that provides yellow pages information and reservations.

6.6.1 Provide Multimodal Route Selection

Overview: This process shall manage the creation of multimodal routes (those with one or more modes in them) in response to traveler's trip or route requests. It shall support on-line route guidance for travelers using personal devices, route guidance for vehicles, selection of specialized vehicle based routes for other ITS functions, (such as Manage Emergency Services and Manage Commercial Vehicles), and selection of multimodal routes in response to trip planning requests from travelers. The multimodal routes provided by the process shall take account of the traveler's preferences and constraints. Constraints can include the access needs of those with disabilities. Preferences can include minimizing waiting time at modal interchange points, level of traveler security, or minimum cost.

6.6.2.3 Provide Route Segment Data for Other Areas

Overview: This process shall obtain from another ISP current or predicted data for road links that are outside the area served by the local supplier. This area, which may be defined on a geographic or jurisdictional basis, is the portion of the transportation network on which the ISP maintains real time information. Identification of which ISP to contact is based upon a store that maps a link to the ISP which maintains real time information about this link. If there is no map to another ISP in the data store, then the process will return default or static data for the link(s). This process shall also respond to similar requests from other ISPs for real time data on links within the local database.

6.6.2.4 Update Vehicle Route Selection Map Data

Overview: This process shall provide the interface to map update providers, or other appropriate data sources, through which updates of the digitized map data can be obtained. The process shall request new data from the provider on request from the ISP operator interface process. The data received from the supplier shall be loaded into a the map_data_for_route_selection data store by the process in such a way that it can be easily used by the route selection process in determining vehicle routes, trip planning, and on-line vehicle guidance.

6.6.4 Select Transit Route

Overview: This process shall determine routes that are based on regular transit services. Routes shall be provided by the process to travelers in response to trip planning and on-line personal guidance requests. The data provided by the process shall be different for the two types of requests since trip planning information will not need the detail that guidance data requires. The process shall base routes on the current state of the regular transit services using data obtained from processes in the Manage Transit function. It shall also respond to any preferences and constraints, such as those for travelers with special needs, that are specified in the route request. Data on the current use of transit routes in on-line guidance shall be provided by the process to the Manage Demand function to aid in demand management. This data on current use of the transit routes in on-line guidance is stored in the transit_mode_routes data store.

6.6.5 Select Other Routes

Overview: This process shall determine routes, or portions of routes, not based on use of vehicles or regular transit services. Routes shall be provided by the process for travelers in response to trip planning, and on-line personal guidance requests. Data provided by the process will be different for the two types of requests since the data for trip planning will not need the level of detailed that guidance data requires. The process shall calculate its routes using digitized map data obtained and updated by another process. It shall make use of the alternative modes, (such as ferries, walking, cycling, etc.) that have been specified in the route request, and shall also take account of any preferences and constraints, (such as those for travelers with special needs). Data on current use of routes in on-line guidance shall be provided by the process to the Manage Demand function.

7.1.6 Distribute Advanced Charges and Fares

Overview: This process shall be responsible for receiving requests for advanced payment of tolls from the parking lot charge or transit fare collection facilities within the Provide Electronic Payment Services function. It shall pass the requests on to another process in the toll collection facility, and shall return transaction success or failure details to the requesting process. The process shall also receive requests for the advanced payment of parking lot charges and transit fares from the toll payment interface process. It shall send these requests to other processes in the Provide Electronic Payment Services function and when received, return the results to the toll payment interface process.

7.2.6 Distribute Advanced Tolls and Fares

Overview: This process shall be responsible for receiving requests for advanced payment of parking lot charges from the toll or transit fare collection facilities within the Provide Electronic Payment Services function. It shall pass the requests on to another process in the Provide Electronic Parking Lot Payment facility, and shall return transaction success or failure details to the requesting process. The process shall also receive requests for the advanced payment of tolls and transit fares from the parking lot payment interface process. It shall send these requests to other processes in the Provide Electronic Payment Services function and when received, return the results to the Parking Lot payment interface process.

7.3.2 Distribute Advanced Tolls and Parking Lot Charges

Overview: This process shall be responsible for receiving requests for advanced payment of transit fares from the toll and parking lot charge collection facilities within the Provide Electronic Payment Services function. It shall pass the advanced fare requests on to another process in the Process Electronic Transit Fare Payment facility, and when received, shall return transit success or failure details to the requesting process. The process shall also receive requests for advanced payment of tolls and parking lot charges from transit vehicle and roadside (transit stop) fare collection facilities. It shall send these requests to other processes in the Provide Electronic Payment Services function and when received, return the results to the requesting process.

7.4.1.3 Process Driver Map Update Payments

Overview: This process shall be responsible for transacting payments from the driver for updates to the navigable map database in the vehicle. The process shall receive the transaction request data from a process in the Provide Driver and Traveler Services function and shall send the data to the financial institution for action. The process shall send the response from the financial institution to the requesting process and shall send details of the transaction to another process for entry into the payment_transaction_records data store.

7.4.1.4 Process Traveler Map Update Payments

Overview: This process shall be responsible for transacting payments from the traveler for updates to the navigable map database carried in the personal device. The process shall receive the transaction request data from a process in the Provide Driver and Traveler Services function and shall send the data to the financial institution. The process shall send the response from the financial institution to the requesting process and shall send details of the transaction to another process for entry into the payment_transaction_records data store.

7.4.1.6 Process Traveler Trip and Other Services Payments

Overview: This process shall be responsible for transacting advance payments required for the confirmation of a trip by a traveler. Payments supported by the process shall comprise those for any tolls, parking lot charges, transit fares, or other (yellow pages) services that need to be paid for the trip to be confirmed. The process shall receive the transaction data from a process in the Provide Driver and Traveler Services function and shall send the data to the financial institution. Tolls, fares and parking lot charges are sent to the Route Traveler Advanced Payment function for processing. The process shall send the response from the financial institution to the requesting process and shall send details of the transaction to another process for entry into the payment_transaction_records data store.

7.4.2 Collect Price Data for ITS Use

Overview: This process shall be responsible for collecting data on the prices being charged for tolls, parking lots and transit fares. This process shall accept data sent to it by the other processes when they have updated their data and automatically sent it, or this process shall request a transfer of data from the other processes. The process shall load the data into the price_data_for_services data store from which some or all of it can be read on request from processes in other ITS functions.

7.4.3 Route Traveler Advanced Payments

Overview: This process shall be responsible for receiving a traveler's request for advanced payment (for tolls, parking lot charges, and/or transit fares) and routing it to the appropriate part of the Provide Electronic Payment Services function. The process shall also receive responses to the advanced payment requests and shall return them to the originating process.

ISP Advanced Integrated Control Support
This Equipment package supports the traffic management center provide real-time optimized signal control by providing ISP route planning information.

Process Specifications
6.6.2.2 Provide Vehicle Route Calculation Data

Overview: This process shall update the data stores containing information which is used by the another process to calculate vehicle routes. This process shall also provide data about links (speed or travel times), and queues to be broadcast to vehicles (to support autonomous guidance with dynamic link updates). The process shall fuse link and queue data received from Manage Traffic sources with probe data received from vehicles under infrastructure based route guidance, or with probe data obtained from other sources (such as from an electronic toll collection system). The process shall obtain route segment data as requested data or as data received at periodic intervals from other ITS functions. The process shall have the capability to request data about route segments outside its own area by sending a data request to the process that provides the interface with other ISP's. Link addresses, mapped to other ISPs, will be maintained by this process and stored in the link_data_store.

ISP Probe Information Collection
This Equipment package supports the collection of vehicle probe data by the ISP. It provides the capability to accept and process probe vehicle information. This capability shall be provided through the use of additional hardware and probe vehicle control and tracking software.

Process Specifications
6.6.2.2 Provide Vehicle Route Calculation Data

Overview: This process shall update the data stores containing information which is used by the another process to calculate vehicle routes. This process shall also provide data about links (speed or travel times), and queues to be broadcast to vehicles (to support autonomous guidance with dynamic link updates). The process shall fuse link and queue data received from Manage Traffic sources with probe data received from vehicles under infrastructure based route guidance, or with probe data obtained from other sources (such as from an electronic toll collection system). The process shall obtain route segment data as requested data or as data received at periodic intervals from other ITS functions. The process shall have the capability to request data about route segments outside its own area by sending a data request to the process that provides the interface with other ISP's. Link addresses, mapped to other ISPs, will be maintained by this process and stored in the link_data_store.

6.6.2.6 Calculate Vehicle Probe Data for Guidance

Overview: This process shall calculate route segment travel times from vehicle probe data. The probe data shall be accepted by the process from a variety of sources including toll collection points and vehicles receiving on-line infrastructure based guidance. The process shall be responsible for combining the data obtained from these sources and producing one set of route segment travel times or route segment speeds. The process shall indicate route segments for which no data, or insufficient data, is available (this indication could be by setting the link time or speed to zero).

2.11.3 Subsystem Interfaces for ISP

Emergency Management => **Information Service Provider**

Physical Architecture Flow Name: emergency vehicle route request

Special routing instructions and signal priority for emergency vehicles.

Logical Architecture Reference Flow: emergency_vehicle_route_request
This data flow is used by the Manage Emergency Services function to request a dynamic route for an emergency vehicle. It contains the following data items each of which is defined in its own DDE: trip_request + vehicle_identity + emergency_request.

Physical Architecture Flow Name: incident information

Notification of existence of incident and expected severity, location, time and nature of incident.

Logical Architecture Reference Flow: incident_information
This data flow is sent from the Manage Emergency Services function to the Provide Driver and Traveler Services function and contains information that has been requested about incidents. It consists of the following items each of which is defined in its own DDE: incident_number + incident_location + incident_start_time + incident_duration + incident_type + incident_severity + incident_traffic_impact.

Financial Institution => **Information Service Provider**

Physical Architecture Flow Name: transaction status

Response to transaction request. Normally dealing with a request for payment.

Logical Architecture Reference Flow: ffi_driver_map_payment_confirm
This data flow is sent from the financial institution to the Provide Electronic Payment Services function. It contains confirmation that a previously submitted request from a driver for payment for the update of the digitized map data used for on-line vehicle guidance has been accepted and made.

Logical Architecture Reference Flow: ffi_registration_payment_confirm
This data flow is sent from the financial institution to the Provide Electronic Payment Services function. It contains confirmation that a previously submitted request from a yellow services provider for payment to register as a supplier of these services and have details of them made available to travelers and transit users has been accepted and made.

Logical Architecture Reference Flow: ffi_traveler_display_payment_confirm
This data flow is sent from the financial institution to the Provide Electronic Payment Services function. It contains confirmation that a previously submitted request from a traveler for payment for the update of the digitized map data

used as the background for displays of traffic and travel information on a traveler's personal device has been accepted and made.

Logical Architecture Reference Flow: ffi_traveler_map_payment_confirm
This data flow is sent from the financial institution to the Provide Electronic Payment Services function. It contains confirmation that a previously submitted request from a traveler for payment for the update of the navigable map database used for on-line personal guidance has been accepted and made.

Logical Architecture Reference Flow: ffi_traveler_other_services_payments_confirm
This data flow is sent from the financial institution to the Provide Electronic Payment Services function. It contains confirmation that a previously submitted request from a traveler for payment for other (yellow pages) services has been accepted and made.

Logical Architecture Reference Flow: ffi_traveler_rideshare_payment_confirm
This data flow is sent from the financial institution to the Provide Electronic Payment Services function. It contains confirmation that a previously submitted request from a traveler for payment for the provision of rideshare services has been accepted and made.

Fleet and Freight Management => Information Service Provider

Physical Architecture Flow Name: route request

Request for a tailored route based on given constraints.

Logical Architecture Reference Flow: cf_route_request
This data flow is sent from the Manage Commercial Vehicles function to the Provide Driver and Traveler Services function. It is used to request the preparation of a dynamic route for a commercial vehicle and originates with the commercial vehicle fleet manager. The data flow consists of the following items each of which is defined in its own DDE: constraint_on_acceptable_travel_time + constraint_on_eta_change + constraint_on_load_classification + constraint_on_ahs_lanes + constraint_on_interstate + constraint_on_urban + constraint_on_vehicle_type + cv_route_number + destination + departure_time + desired_arrival_time + modes + origin + preferred_routes + preferred_alternate_routes + preferred_route_segments + preferred_weather_conditions.

Logical Architecture Reference Flow: cv_route_request
This data flow is sent from the Manage Commercial Vehicles function to the Provide Driver and Traveler Services function. It is used to request the preparation of a dynamic route for a commercial vehicle and originates with the commercial vehicle driver acting in the role of fleet manager. The data flow consists of the following items each of which is defined in its own DDE: constraint_on_acceptable_travel_time + constraint_on_eta_change + constraint_on_load_classification + constraint_on_ahs_lanes + constraint_on_interstate + constraint_on_urban + constraint_on_vehicle_type + destination + departure_time + desired_arrival_time + modes + origin + preferred_routes + preferred_alternate_routes + preferred_route_segments + preferred_weather_conditions + vehicle_identity.

Information Service Provider => Emergency Management

Physical Architecture Flow Name: emergency vehicle route

Routing for emergency vehicle including greenwave paths.

Logical Architecture Reference Flow: emergency_vehicle_route
This data flow is sent from the Manage Emergency Services function to the Manage Traffic function. It contains details of the emergency vehicle's route and is used to trigger a special green wave route for the emergency vehicle. The data flow consists of the following items each of which is described in its own DDE: route + emergency_request +

vehicle_identity.

Physical Architecture Flow Name: incident information request

Request for incident information, clearing time, severity. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Logical Architecture Reference Flow: incident_information_request
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Emergency Services function and is a request for information about incidents It contains the following data items each of which is defined in its own DDE: incident_type + incident_oldest_time.

Information Service Provider => Financial Institution

Physical Architecture Flow Name: payment request

Request for payment from financial institution.

Logical Architecture Reference Flow: tfi_driver_map_payment_request
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains a request from a driver for payment for the update of the digitized map data used for on-line vehicle guidance.

Logical Architecture Reference Flow: tfi_registration_payment_request
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains a request from a yellow services provider for payment to register as a supplier of these services and have details of them made available to travelers and transit users.

Logical Architecture Reference Flow: tfi_traveler_display_payment_request
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains a request from a traveler for payment for the update of the digitized map data used as the background for displays of traffic and travel information on a traveler's personal device.

Logical Architecture Reference Flow: tfi_traveler_map_payment_request
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains a request from a traveler for payment for the update of the navigable map database used for on-line personal guidance.

Logical Architecture Reference Flow: tfi_traveler_other_services_payments_request
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It a request from a traveler for payment for other (yellow pages) services.

Logical Architecture Reference Flow: tfi_traveler_rideshare_payment_request
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains a request from a traveler for payment for the provision of rideshare services as part of a confirmed trip.

Information Service Provider => Fleet and Freight Management

Physical Architecture Flow Name: route plan

Tailored route provided by ISP in response to a specific request.

Logical Architecture Reference Flow: cf_route
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Commercial Vehicles function . It contains details of a dynamic route provided for a commercial vehicle the request for which originated with

the commercial vehicle fleet manager. The data flow consists of the following data items each of which is defined in its own DDE: cv_route_data + cv_route_number.

Logical Architecture Reference Flow: cv_route
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Commercial Vehicles function . It contains details of a dynamic route provided for a commercial vehicle the request for which originated with the commercial vehicle driver acting in the role of fleet manager. The data flow consists of the following data items each of which is defined in its own DDE: cv_route_data + vehicle_identity.

Information Service Provider => Intermodal Transportation Service Provider

Physical Architecture Flow Name: intermodal information

Schedule information for alternate mode transportation providers such as train, ferry, air and bus.

Logical Architecture Reference Flow: titsp_air_services_request
This data flow is sent to the intermodal transportation service provider from the Provide Driver and Traveler Services function and contains a request for details of the regular and charter air services available to move travelers.

Logical Architecture Reference Flow: titsp_confirm_intermodal_service
This data flow is sent to the intermodal transportation service provider from the Provide Driver and Traveler Services function and contains a request for provision of an alternate mode service as part of a traveler's proposed trip.

Logical Architecture Reference Flow: titsp_ferry_services_request
This data flow is sent to the intermodal transportation service provider from the Provide Driver and Traveler Services function and contains a request for details of the sea and river ferry services available to move travelers.

Logical Architecture Reference Flow: titsp_rail_services_request
This data flow is sent to the intermodal transportation service provider from the Provide Driver and Traveler Services function and contains a request for details of the heavy rail services (i.e. those which do not form part of a transit operation) available to move travelers.

Information Service Provider => ISP Operator

Physical Architecture Flow Name: ISP operating parameters

Parameters provided to the ISP Operator by the ISP including broadcast information settings, route selection controls, and travel optimization algorithms.

Logical Architecture Reference Flow: tispo_broadcast_data_parameters_output
This data flow is sent to the ISP operator by the Provide Driver and Traveler Services function. It contains output of the parameters used in wide area information broadcast and is a result of an output request from the operator.

Logical Architecture Reference Flow: tispo_route_selection_parameters
This data is sent to the ISP Operator from the Provide Driver and Traveler Services function. It contains output of the parameters used by the route selection processes to best determine the routes used for travelers proposed trips and for on-line vehicle guidance. The output may be in audible, visual or hardcopy form and will require no further processing to be understood by the operator.

Logical Architecture Reference Flow: tispo_trip_planning_parameters
This data is sent to the ISP Operator from the Provide Driver and Traveler Services function. It contains output of the parameters used by the trip planning processes to best determine the routes for travelers. The output may be in audible, visual or hardcopy form and will require no further processing to be understood by the operator.

Information Service Provider => Map Update Provider

Physical Architecture Flow Name: map update request

Request for a map update which could include a new underlying map or map layer updates.

Logical Architecture Reference Flow: tmup_request_other_routes_map_update
This data flow is sent to the map update provider by the Provide Driver and Traveler Services function and contains a request for a new copy of the digitized map data used by the process that selects other, i.e. non-vehicle and non-transit, routes for travelers.

Logical Architecture Reference Flow: tmup_request_route_selection_map_update
This data flow is sent to the map update provider by the Provide Driver and Traveler Services function and contains a request for a new copy of the digitized map data used by the process that selects vehicle based routes for travelers and drivers.

Information Service Provider => Media

Physical Architecture Flow Name: traveler information for media

General traveler information regarding incidents, unusual traffic conditions, transit issues, or other advisory information that has been desensitized and provided to the media.

Logical Architecture Reference Flow: tm_incident_information
This data flow contains data on current incidents in a form which will be readily understood by Media Systems.

Logical Architecture Reference Flow: tm_traffic_information
This data flow gives information on a particular current traffic situation in a form which will be readily understood by Media Systems .

Logical Architecture Reference Flow: tm_transit_vehicle_deviations
This data flow contains details of deviations from schedule of specific transit vehicles, or routes. The information will enable the media to broadcast the details to travelers via such things as local radio, bulletin boards, etc.

Logical Architecture Reference Flow: tm_traveler_information_request
This data flow is sent to the media from the Provide Driver and Traveler Services function and contains a request for any information that the media has that might be of interest to travelers planning trips. This may include but not be limited to such things as special events. sports fixtures, etc.

Information Service Provider => Other ISP

Physical Architecture Flow Name: ISP coordination

Coordination and exchange of transportation information between centers. This flow allows a broad range of transportation information collected by one ISP to be redistributed to many other ISPs and their clients.

Logical Architecture Reference Flow: toisp_data_supply
This data flow is sent by the Plan System Deployment and Implementation function and contains a road data covering the local geographic area for use by the similar function in an ITS covering another geographic or jurisdictional area. It consists of the following data item which is defined in its own DDE: road_data.

Logical Architecture Reference Flow: toisp_request_data

This data flow is sent by the Plan System Deployment and Implementation function and contains a request for road data covering another geographic area outside that covered by the local function.

Logical Architecture Reference Flow: toisp_traffic_data_request
This data flow contains a request (either as a subscription or as individual request) to another ISP for available traffic data to be provided. This allows an ISP to act as a wholesaler and send data from an Other ISP to the process.

Logical Architecture Reference Flow: toisp_traffic_information
This data flow contains a complete (or partial) set of the traffic data which has been created through fusion of available data sources. This includes current, long term (historical) and predicted link data as well as incident data. This data flow allows one ISP to act as a wholesaler and provide information to other ISPs. The data flow consists of the following items each of which is defined in its own DDE: source_identity + current_data_for_retrieval + long_term_data_for_retrieval + predicted_incidents + predictive_model_data_for_retrieval.

Logical Architecture Reference Flow: toisp_transit_data_request
This data flow contains a request (either as a subscription or as individual request) to another ISP for available transit data to be provided. This allows an ISP to act as a wholesaler and send data from Other ISP to the process.

Logical Architecture Reference Flow: toisp_transit_information
This data flow is used to provide data on the current state of transit operations (regarding both incidents and transit vehicle schedule status) for use by the Other ISP (information service provider). It consists of the following items each of which is defined in its own DDE: transit_running_data_for_advisory_output + transit_incident_data.

Logical Architecture Reference Flow: transit_running_data_for_advisory_output
This data flow is sent from the Manage Transit function to the Provide Driver and Traveler Services function. It is used to provide data on the current state of transit operation for use in driver and traveler advisories and consists of the following items each of which is defined in its own DDE: list_size + list_size{transit_vehicle_passenger_loading + transit_vehicle_running_times + transit_vehicle_schedule_deviations + transit_vehicle_eta}.

Information Service Provider => Parking Management

Physical Architecture Flow Name: parking lot data request

Request for parking lot occupancy, fares, and availability. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Logical Architecture Reference Flow: advanced_other_charges_request
This data flow is used within the Provide Electronic Payment Services function to request that a parking lot charge be paid for in advance by either a driver who is paying a toll or a traveler (as a transit user) who is paying a transit fare. It consists of the following data items each of which is defined in its own DDE: credit_identity + parking_lot_identity + parking_space_details + stored_credit + vehicle_identity.

Logical Architecture Reference Flow: advanced_traveler_charges_request
This data flow is used within the Provide Electronic Payment Services function to request that a parking lot charge be paid for in advance by a traveler who is planning a trip. It consists of the following data items each of which is defined in its own DDE: credit_identity + parking_space_details + stored_credit + vehicle_identity.

Logical Architecture Reference Flow: parking_lot_data_request
This data flow is sent from the Provide Driver and Traveler Services function to the Provide Electronic Payment Services function and contains a request for data about the number of spaces that are available in a particular parking lot at the specified data and time. This data is requested as part of the process of putting together a proposed trip in response to a traveler's trip request This data flow contains the following items each of which is defined in its own DDE: date + parking_lot_identity + time + traveler_identity.

Logical Architecture Reference Flow: parking_lot_price_data_request

This data flow is used within the Provide Electronic Payment Services function. It contains a request for the current parking lot price data to be provided from the store that is being used to calculate parking lot charges.

Physical Architecture Flow Name: parking reservations request

Reservation request for parking lot.

Logical Architecture Reference Flow: parking_lot_reservation_request
This data flow is sent from the Provide Driver and Traveler Services function to the Provide Electronic Payment Services function and contains a request for an advance reservation to be made at a parking lot. This reservation is the result of a traveler confirming a proposed trip. This data flow contains the following items each of which is defined in its own DDE: date + parking_lot_identity + time + traveler_identity.

Information Service Provider => Personal Information Access

Physical Architecture Flow Name: broadcast information

General broadcast information that contains link travel times, incidents, advisories, transit services and a myriad of other traveler information.

Logical Architecture Reference Flow: traffic_data_for_personal_devices
This data flow is used to provide data on the traffic flowing in the road network, plus that which is predicted to flow in the network for output to a traveler's personal device. The data flow has been sized to take account of the need to only supply a subset of the full traffic data to the case where the device is a portable device. This subset will depend on the route segment list included in the request, which for sizing purposes has been assumed to have a maximum value of 25. Thus, the size expression below is based on sampling a percentage of the data equivalent to an N route segment square area (i.e. N/total number of segments). The flow consists of the following items each of which is defined in its own DDE: source_identity + current_data_for_broadcast + predicted_data_for_broadcast + traveler_identity.

Logical Architecture Reference Flow: transit_deviations_for_personal_devices
This data flow contains current transit service deviations for a particular route. This data will be output to a traveler's personal device. It consists of the following data items each of which is defined in its own DDE: traveler_identity + list_size + list_size{transit_vehicle_identity + transit_vehicle_achieved_time + transit_route_segment_number}.

Physical Architecture Flow Name: traveler information

Traveler information comprised of traffic status, advisories, incidents, responses to traveler requests (e.g., traveler routing, yellow pages), payment information and many other travel-related data updates and confirmations.

Logical Architecture Reference Flow: traffic_data_for_personal_devices
This data flow is used to provide data on the traffic flowing in the road network, plus that which is predicted to flow in the network for output to a traveler's personal device. The data flow has been sized to take account of the need to only supply a subset of the full traffic data to the case where the device is a portable device. This subset will depend on the route segment list included in the request, which for sizing purposes has been assumed to have a maximum value of 25. Thus, the size expression below is based on sampling a percentage of the data equivalent to an N route segment square area (i.e. N/total number of segments). The flow consists of the following items each of which is defined in its own DDE: source_identity + current_data_for_broadcast + predicted_data_for_broadcast + traveler_identity.

Logical Architecture Reference Flow: transit_deviations_for_personal_devices
This data flow contains current transit service deviations for a particular route. This data will be output to a traveler's personal device. It consists of the following data items each of which is defined in its own DDE: traveler_identity + list_size + list_size{transit_vehicle_identity + transit_vehicle_achieved_time + transit_route_segment_number}.

Logical Architecture Reference Flow: traveler_map_update_payment_response
This data flow is sent from the Provide Electronic Payment Services function to the Provide Driver and Traveler Services function and contains the response to a previous request from the traveler that payment be made for an update of the navigable map database used for on-line traveler guidance. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + traveler_identity.

Logical Architecture Reference Flow: traveler_personal_display_update_payment_response
This data flow is sent from the Provide Electronic Payment Services function to the Provide Driver and Traveler Services function and contains the response to a previous request from the traveler that payment be made for an update of the digitized map data used as background to the displays of traffic and travel information on a traveler's personal device. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + traveler_identity.

Logical Architecture Reference Flow: traveler_personal_payment_confirmation
This data flow is used within the Provide Driver and Traveler Services function to indicate the payment for a confirmed trip has been successfully completed, or that the total cost can now be deducted from the credit stored on the traveler's payment instrument. The request for payment will have been initiated by input from the traveler to a personal device. The data flow consists of the following data items each of which is defined in its own DDE: advanced_tolls_confirm + advanced_fares_confirm + advanced_parking_lot_charges_confirm + credit_identity + stored_credit + traveler_identity + traveler_total_trip_cost.

Logical Architecture Reference Flow: traveler_personal_transaction_confirmation
This data flow is used within the Provide Driver and Traveler Services function to confirm any reservations made by the traveler from a personal device. These reservations will be based on information obtained by the traveler from previous data input and output through the device. The data flow consists of the following data items each of which is defined in its own DDE: credit_identity + traveler_identity + transaction_number + yellow_pages_cost + yellow_pages_lodging_reservation_confirmation + yellow_pages_dining_reservation_confirmation + yellow_pages_ticket_purchase_confirmation.

Logical Architecture Reference Flow: traveler_personal_yellow_pages_data
This data flow is used within the Provide Driver and Traveler Services function and contains details of other (yellow pages) services which is to be sent to the traveler interface facility for output using a kiosk. The size of the data flow has been set to take account of the need to provide only a small percentage of the total yellow pages data that is available. The data flow consists of the following data items each of which is defined in its own DDE: traveler_identity + yellow_pages_general_information + yellow_pages_specific_information + yellow_pages_transaction_information.

Physical Architecture Flow Name: trip plan

A sequence of links and special instructions comprising a trip plan indicating efficient routes for navigating the links. Normally coordinated with traffic conditions, other incidents, preemption and prioritization plans.

Logical Architecture Reference Flow: traveler_guidance_route
This data flow is used within the Provide Driver and Traveler Services function and contains the data for a traveler's route which has been produced following a route request from the traveler. It consists of the following data items each of which is defined in its own DDE: route_identity + traveler_route + traveler_identity.

Logical Architecture Reference Flow: traveler_personal_trip_information
This data flow is used within the Provide Driver and Traveler Services function and contains information about a proposed trip that the traveler has requested earlier from the personal device. It consists of the following data items each of which is defined in its own DDE: current_conditions + [paratransit_personal_schedule | route | rideshare_response] + traveler_identity + traveler_total_trip_cost.

Information Service Provider => Planning Subsystem

Physical Architecture Flow Name: aggregate travel data

Aggregated transportation infrastructure data and associated traveler transaction data for planning purposes.

Logical Architecture Reference Flow: current_other_routes_use

This data flow is used within the Provide Driver and Traveler Services function and contains data about the non-vehicle portion(s) of routes that have been requested by travelers. These route portions will involve the use of modes such as cycling, walking, etc. The data will be stored in ascending route segment number order (i.e. from 1 to the maximum number of route segments),r and consists of the following data items each of which is defined in its own DDE: route_segment_total_number + route_segment_total_number{route_segment_identity + time_period{route_segment_guided_travelers} + route_segment_journey_time}.

Logical Architecture Reference Flow: current_road_network_use

This data flow is sent from the Provide Driver and Traveler Services function to the Manage Traffic function. It contains information about how many vehicles are being guided down each route segment and the average journey time for each route segment provided by guided vehicles. The data will be stored in ascending route segment number order (i.e. from 1 to the maximum number of route segments), and consists of the following data items each of which is defined in its own DDE: route_segment_total_number + route_segment_total_number{route_segment_identity + route_segment_use_prediction + route_segment_journey_time}.

Logical Architecture Reference Flow: driver_map_update_payments_transactions

This data flow is used within the Provide Payment Electronic Services function and contains records of all payment transactions for driver in-vehicle guidance map updates.

Logical Architecture Reference Flow: traveler_map_update_payments_transactions

This data flow is used within the Provide Payment Electronic Services function and contains records of all payment transactions for driver map updates.

Logical Architecture Reference Flow: traveler_rideshare_payments_transactions

This data flow is used within the Provide Payment Electronic Services function and contains records of all payment transactions for traveler ridesharing provision.

Logical Architecture Reference Flow: traveler_trip_payments_transactions

This data flow is used within the Provide Payment Electronic Services function and contains records of all payment transactions for the provision of other (yellow pages) services and advance toll, parking lot charges, or transit fares as part of travelers' confirmed trips.

Logical Architecture Reference Flow: yellow_pages_provder_payments_transactions

This data flow is used within the Provide Payment Electronic Services function and contains records of all payment transactions for the provision of other (yellow pages) services and registration of suppliers of these services.

Information Service Provider => **Remote Traveler Support**

Physical Architecture Flow Name: broadcast information

General broadcast information that contains link travel times, incidents, advisories, transit services and a myriad of other traveler information.

Logical Architecture Reference Flow: traffic_data_for_kiosks

This data flow is sent from the Manage Traffic function to the Provide Driver and Traveler Services function. It is used to provide data on the traffic flowing in the road network, plus that which is predicted to flow in the network for output at a kiosk. The flow consists of the following items each of which is defined in its own DDE: kiosk_identity + source_identity + current_data_for_broadcast + predicted_data_for_broadcast.

Logical Architecture Reference Flow: transit_deviations_for_kiosks
This data flow is sent by the Manage Transit function to the Provide Driver and Traveler Services function and contains current transit service deviations for output to a kiosk. It consists of the following data items each of which is defined in its own DDE: kiosk_identity + transit_vehicle_schedule_deviations.

Physical Architecture Flow Name: traveler information

Traveler information comprised of traffic status, advisories, incidents, responses to traveler requests (e.g., traveler routing, yellow pages), payment information and many other travel-related data updates and confirmations.

Logical Architecture Reference Flow: advanced_tolls_and_charges_roadside_confirm
This data flow is used within the Provide Electronic Payment Services function and contains the result of the requested advanced payment transaction from a traveler (as a transit user) at the roadside, i.e. a transit stop. It consists of the following data items each of which is defined in its own DDE: advanced_charges_confirm + advanced_tolls_confirm + confirmation_flag.

Logical Architecture Reference Flow: traffic_data_for_kiosks
This data flow is sent from the Manage Traffic function to the Provide Driver and Traveler Services function. It is used to provide data on the traffic flowing in the road network, plus that which is predicted to flow in the network for output at a kiosk. The flow consists of the following items each of which is defined in its own DDE: kiosk_identity + source_identity + current_data_for_broadcast + predicted_data_for_broadcast.

Logical Architecture Reference Flow: transit_deviations_for_kiosks
This data flow is sent by the Manage Transit function to the Provide Driver and Traveler Services function and contains current transit service deviations for output to a kiosk. It consists of the following data items each of which is defined in its own DDE: kiosk_identity + transit_vehicle_schedule_deviations.

Logical Architecture Reference Flow: traveler_payment_confirmation
This data flow is used within the Provide Driver and Traveler Services function to indicate the payment for a confirmed trip has been successfully completed., or that the total cost can now be deducted from the credit stored on the traveler's payment instrument The request for payment will have been initiated by input from the traveler to a kiosk. The data flow consists of the following data items each of which is defined in its own DDE: advanced_tolls_confirm + advanced_fares_confirm + advanced_parking_lot_charges_confirm + credit_identity + kiosk_identity + stored_credit + traveler_total_trip_cost.

Logical Architecture Reference Flow: traveler_transaction_confirmation
This data flow is used within the Provide Driver and Traveler Services function to confirm any reservations made by the traveler from a kiosk. These reservations will be based on information obtained by the traveler from previous data input and output through the kiosk. The data flow consists of the following data items each of which is defined in its own DDE: credit_identity + kiosk_identity + transaction_number + yellow_pages_cost + yellow_pages_lodging_reservation_confirmation + yellow_pages_dining_reservation_confirmation + yellow_pages_ticket_purchase_confirmation.

Logical Architecture Reference Flow: traveler_yellow_pages_data
This data flow is used within the Provide Driver and Traveler Services function and contains details of other (yellow pages) services which is to be sent to the traveler interface facility for output using a kiosk. The size of the data flow has been set to take account of the need to provide only a small percentage of the total yellow pages data that is available. The data flow consists of the following data items each of which is defined in its own DDE: kiosk_identity + yellow_pages_general_information + yellow_pages_specific_information + yellow_pages_transaction_information.

Physical Architecture Flow Name: trip plan

A sequence of links and special instructions comprising a trip plan indicating efficient routes for navigating the links. Normally coordinated with traffic conditions, other incidents, preemption and prioritization plans.

Logical Architecture Reference Flow: traveler_trip_information

This data flow is used within the Provide Driver and Traveler Services function and contains information about a proposed trip that the traveler has requested earlier from the kiosk. It consists of the following data items each of which is defined in its own DDE: current_conditions + kiosk_identity + paratransit_personal_schedule + rideshare_response + 1{route + route_cost}4.

Information Service Provider => Toll Administration

Physical Architecture Flow Name: toll data request

Request made to obtain toll schedule information or pay a toll in advance. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Logical Architecture Reference Flow: advanced_other_tolls_request
This data flow is used within the Provide Electronic Payment Services function to request that a toll be paid for in advance by either a driver who is paying a parking lot charge or a traveler (as a transit user) who is paying a transit fare. It consists of the following data items each of which is defined in its own DDE: credit_identity + stored_credit + toll_route_segments + vehicle_identity.

Logical Architecture Reference Flow: advanced_traveler_tolls_request
This data flow is used within the Provide Electronic Payment Services function to request that a toll be paid for in advance by a traveler who is planning a trip. It consists of the following data items each of which is defined in its own DDE: credit_identity + stored_credit + toll_route_segments + vehicle_identity.

Logical Architecture Reference Flow: toll_price_data_request
This data flow is used within the Provide Electronic Payment Services function. It contains a request for the current toll price data to be provided from the store that is being used to calculate toll costs.

Information Service Provider => Traffic Management

Physical Architecture Flow Name: fare and price information

Current transit, parking, and toll fee schedule information.

Logical Architecture Reference Flow: parking_lot_charge_details
This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function and contains the prices being charged by each parking lot for each of its spaces, together with the time and date for which they apply. parking_lot_identity + parking_lot_price + parking_lot_charge_application_time + vehicle_type_for_charges.

Logical Architecture Reference Flow: toll_price_details
This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function and contains the price for each road segment to which a toll applies, with the time and date for when it applies. This data will be used by the Manage Travel Demand facility in its efforts to re-distribute travel demand to the more efficient providers. The data flow consists of the following data items each of which is defined in its own DDE: toll_segments + toll_price + toll_price_application_time + vehicle_type_for_tolls.

Logical Architecture Reference Flow: transit_fare_details
This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function and contains details of the fares being currently charged for transit services. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{transit_route_number + transit_route_segment_list + transit_user_category + transit_route_use_time}.

Physical Architecture Flow Name: logged special vehicle route

Anticipated route information for special vehicles (e.g., oversize vehicles) or groups of vehicles (e.g., governer's motorcade) that may require changes in traffic control strategy.

Logical Architecture Reference Flow: logged_special_vehicle_route

This data contains details about a route that has been requested by a special vehicle. This could be a commercial vehicle that is carrying cargo which could be viewed as being liable to cause a potential incident. Loads falling into this category are those containing hazardous (HAZMAT) material, or those which are outsize, e.g. wide, heavy, or fragile and hence slow moving. This could also include vehicles which must be specially routed (e.g. the governors motorcade). The data flow is derived from the route that has been produced for the special vehicle. hazmat_load_data + list_size + list_size{route_segment_end_point + route_segment_estimated_arrival_time + route_segment_estimated_travel_time + route_segment_identity + route_segment_start_point}.

Logical Architecture Reference Flow: special_vehicle_priority_routing

This data flow is a special form of route similar to an emergency vehicle route, but for use by other vehicle types which may be given special priority routing (e.g. green wave routing). This could be applied to HOV vehicles, special HAZMAT, priority vehicles (e.g. governors motorcade), or even to regular vehicles under a low traffic volume period (e.g. in the early hours of the morning). This flow contains the items shown below each of which is defined in its own DDE: route + vehicle_identity.

Physical Architecture Flow Name: request for traffic information

Request for traffic information that specifies the region/route of interest, the desired effective time period, and other parameters that allow preparation of a tailored response. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Logical Architecture Reference Flow: traffic_data_distribution_request

This data flow contains a request for particular data to be retrieved from the stores of long term, current, and predicted traffic data. The request is in response to a variety of requests received from ITS Users.

Physical Architecture Flow Name: road network use

Aggregated route usage and associated travel data from clients for planning and analysis.

Logical Architecture Reference Flow: current_other_routes_use

This data flow is used within the Provide Driver and Traveler Services function and contains data about the non-vehicle portion(s) of routes that have been requested by travelers. These route portions will involve the use of modes such as cycling, walking, etc. The data will be stored in ascending route segment number order (i.e. from 1 to the maximum number of route segments),r and consists of the following data items each of which is defined in its own DDE: route_segment_total_number + route_segment_total_number{route_segment_identity + time_period{route_segment_guided_travelers} + route_segment_journey_time}.

Logical Architecture Reference Flow: current_road_network_use

This data flow is sent from the Provide Driver and Traveler Services function to the Manage Traffic function. It contains information about how many vehicles are being guided down each route segment and the average journey time for each route segment provided by guided vehicles. The data will be stored in ascending route segment number order (i.e. from 1 to the maximum number of route segments), and consists of the following data items each of which is defined in its own DDE: route_segment_total_number + route_segment_total_number{route_segment_identity + route_segment_use_prediction + route_segment_journey_time}.

Logical Architecture Reference Flow: current_transit_routes_use

This data flow is used within the Provide Driver and Traveler Services and Manage Traffic functions. It contains data showing the numbers of travelers using all or part of the available transit routes, either for personal guidance or as part of trip requests. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{transit_route_number + transit_route_current_use}.

Information Service Provider => Transit Management

Physical Architecture Flow Name: demand responsive transit request

Request for paratransit support.

Logical Architecture Reference Flow: paratransit_trip_request
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Transit function to action a trip request using the paratransit operation. It contains the following data items each of which is defined in its own DDE: traveler_identity + trip_request.

Physical Architecture Flow Name: selected routes

Routes selected based on route request criteria.

Logical Architecture Reference Flow: advanced_tolls_and_charges_vehicle_confirm
This data flow is used within the Provide Electronic Payment Services function and contains the result of the requested advanced payment transaction from a traveler (as a transit user) in a transit vehicle. It consists of the following data items each of which is defined in its own DDE: advanced_charges_confirm + advanced_tolls_confirm + confirmation_flag.

Logical Architecture Reference Flow: paratransit_service_confirmation
This data flow is sent by the Provide Driver and Traveler Services function to the Manage Transit function to confirm that the traveler wants to use the previously identified paratransit service. It contains the following data items each of which is defined in its own DDE: paratransit_service_identity + transit_confirmation_flag + traveler_identity.

Physical Architecture Flow Name: transit information request

Request for transit operations information including schedule and fare information. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Logical Architecture Reference Flow: advanced_other_fares_request
This data flow is used within the Provide Electronic Payment Services function to request that a transit fare be paid in advance by a driver who is paying either a parking lot charge or a toll. It consists of the following data items each of which is defined in its own DDE: credit_identity + stored_credit + transit_route_origin + transit_route_destination + transit_journey_date + traveler_identity.

Logical Architecture Reference Flow: advanced_traveler_fares_request
This data flow is used within the Provide Electronic Payment Services function to request that a transit fare be paid for in advance by a traveler who is planning a trip using facilities in the Provide Driver and Traveler Services function. It consists of the following items each of which is defined in its own DDE: credit_identity + stored_credit + transit_route_origin + transit_route_destination + transit_journey_date + traveler_identity.

Logical Architecture Reference Flow: transit_fare_data_request
This data flow is used within the Provide Electronic Payment Services function. It contains a request for the current transit fare price data to be provided from the store that is being used to calculate transit fares.

Logical Architecture Reference Flow: transit_services_advisories_request
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Transit function. It is a request for supply of details of the services currently being provided by the transit fleet and will be used in the preparation of on-line driver and traveler advisory data for output to vehicles.

Logical Architecture Reference Flow: transit_services_guidance_request
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Transit function. It is a

request for supply of details of the services being currently provided by the transit fleet and will be used in the preparation of on-line traveler guidance data. The process(es) that are providing the interface through which the traveler is obtaining the on-line guidance will have to provide the origin and destination so that the receiving process in the Manage Transit function can work out for which transit route(s) data will be provided. The data flow consists of the following data items each of which is defined in its own DDE: destination + origin + traveler_identity.

Logical Architecture Reference Flow: transit_vehicle_deviations_details_request
This data flow is used within the Manage Transit function. It contains a request for output of the details of the deviations of transit vehicles from their published routes and schedules for use as a source of data to be sent to processes in other functions.

Information Service Provider => Vehicle

Physical Architecture Flow Name: broadcast information

General broadcast information that contains link travel times, incidents, advisories, transit services and a myriad of other traveler information.

Logical Architecture Reference Flow: broadcast_data
This data flow is used within the Provide Driver and Traveler Services function to provide traffic and travel advisory data via a wide area broadcast message to drivers and travelers in vehicles. It consists of the following data items each of which is defined in its own DDE: traffic_data_for_broadcast + predicted_incidents_for_broadcast + prediction_data_for_broadcast + transit_services_for_broadcast + transit_running_data_for_broadcast.

Logical Architecture Reference Flow: link_and_queue_data
This data contains, for each link, the average journey time, speed, and occupancy. For queues it contains the queue times for each link. This data is computed from traffic data and (if available) vehicle probe data. link_state_data_for_broadcast + list_size + list_size{link_identity + link_journey_time + link_queue_time}.

Physical Architecture Flow Name: traveler information

Traveler information comprised of traffic status, advisories, incidents, responses to traveler requests (e.g., traveler routing, yellow pages), payment information and many other travel-related data updates and confirmations.

Logical Architecture Reference Flow: advanced_fares_and_charges_response
This data flow is used within the Provide Electronic Payment Services function and contains the result of the requested advanced transit fare and/or parking lot charge payment transaction from a driver. It consists of the following data item which is defined in its own DDE: confirmation_flag + credit_identity + parking_lot_cost + stored_credit + transit_fare.

Logical Architecture Reference Flow: advanced_tolls_and_fares_response
This data flow is used within the Provide Electronic Payment Services function and contains the result of the requested advanced toll and/or transit fare payment transaction from a driver. It consists of the following data item which is defined in its own DDE: confirmation_flag + credit_identity + stored_credit + toll_cost + transit_fare.

Logical Architecture Reference Flow: advisory_data
This data flow is used within the Provide Driver and Traveler Services function to provide traffic and travel advisory data to drivers and travelers in vehicles. It consists of the following data items each of which is defined in its own DDE: traffic_data_for_advisories + predicted_incidents_for_advisories + prediction_data_for_advisories + transit_services_for_advisories + transit_running_data_for_advisories.

Logical Architecture Reference Flow: driver_map_update_payment_response
This data flow is sent from the Provide Electronic Payment Services function to the Provide Driver and Traveler Services function and contains the response to a previous request from the driver that payment be made for an update of the navigable map database used for on-line vehicle guidance. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + vehicle_identity.

Logical Architecture Reference Flow: link_and_queue_data
This data contains, for each link, the average journey time, speed, and occupancy. For queues it contains the queue times for each link. This data is computed from traffic data and (if available) vehicle probe data.
link_state_data_for_broadcast + list_size + list_size{link_identity + link_journey_time + link_queue_time}.

Logical Architecture Reference Flow: yellow_pages_advisory_data
This data flow is used within the Provide Driver and Traveler Services function to provide yellow pages data to drivers and transit users in vehicles and/or confirmation of a previously requested reservation. It consists of the following data items each of which is defined in its own DDE: yellow_pages_data_for_advisories + yellow_pages_cost + yellow_pages_dining_reservation_confirmation + yellow_pages_lodging_reservation_confirmation + yellow_pages_ticket_purchase_confirmation.

Physical Architecture Flow Name: trip plan

A sequence of links and special instructions comprising a trip plan indicating efficient routes for navigating the links. Normally coordinated with traffic conditions, other incidents, preemption and prioritization plans.

Logical Architecture Reference Flow: vehicle_guidance_route
This data flow is used within the Provide Driver and Traveler Services function and is a special form of %034route%034 for vehicle guidance only. It contains a subset of the data items included in the %034route%034 data flow to meet the requirements of in-vehicle infrastructure based guidance as opposed to the more general requirements for a route need as part of a trip planning activity. The data flow consists of the following data items each of which is defined in its own DDE: route_identity + route_segment_number{route_segment_description + route_segment_end_point + route_segment_estimated_travel_time + route_segment_report_position_points + route_segment_start_point} + vehicle_identity.

Information Service Provider => Yellow Pages Service Providers

Physical Architecture Flow Name: provider profile confirm

Confirmation of profile information received by a service provider (e.g. for a hotel or restaurant).

Logical Architecture Reference Flow: typsp_provider_update_confirm
This data flow is sent to confirm that a request for registration or update of the yellow pages service provider's profile has been successfully completed. The provider can now expect to receive requests for yellow pages data from another process within this function.

Physical Architecture Flow Name: travel service request

Request for reservation or other service (e.g., yellow pages).

Logical Architecture Reference Flow: typsp_transaction_request
This data flow is sent to the information and service provider from the Provide Driver and Traveler Services function and contains a request that payment for the associated yellow pages services is transacted.

Logical Architecture Reference Flow: typsp_yellow_pages_info_request
This data flow is used by the Provide Driver and Traveler Services function to obtain data from the information and service providers. It contains requests for information of a general nature, or specific information, or information an available transactions.

Intermodal Transportation Service Provider => Information Service Provider

Physical Architecture Flow Name: intermodal information

Schedule information for alternate mode transportation providers such as train, ferry, air and bus.

Logical Architecture Reference Flow: fitsp_air_services
This data flow is sent from the intermodal transportation service provider to the Provide Driver and Traveler Services function and contains details of the regular and charter air services available to move travelers.

Logical Architecture Reference Flow: fitsp_ferry_services
This data flow is sent from the intermodal transportation service provider to the Provide Driver and Traveler Services function and contains details of the sea and river ferry services available to move travelers.

Logical Architecture Reference Flow: fitsp_intermodal_service_confirmation
This data flow is sent from the intermodal transportation service provider to the Provide Driver and Traveler Services function and contains confirmation that a previous request from a traveler for an alternate mode service has been accepted.

Logical Architecture Reference Flow: fitsp_rail_services
This data flow is sent from the intermodal transportation service provider to the Provide Driver and Traveler Services function and contains details of the heavy rail services (i.e. those which do not form part of a transit operation) available to move travelers.

ISP Operator => **Information Service Provider**

Physical Architecture Flow Name: ISP operating parameter updates

Tuning and performance enhancement parameters to ISP algorithms

Logical Architecture Reference Flow: fispo_broadcast_data_parameters_request
This data flow is sent from the ISP operator to the Provide Driver and Traveler Services function. It contains a request for the output of the parameters used in the output of wide area information broadcasts.

Logical Architecture Reference Flow: fispo_broadcast_data_parameters_update
This data flow is sent from the ISP operator to the Provide Driver and Traveler Services function. It contains updates of parameters used in wide area information broadcast to drivers and transit users in vehicle. These parameters are used to %034filter%034 the available data to pick out that which is unusual or which is useful if broadcast.

Logical Architecture Reference Flow: fispo_request_other_routes_selection_map_data_update
This data flow is sent from the ISP operator to the Provide Driver and Traveler Services function. It contains a request for an update of the digitized map data used by the process that selects non-vehicle or transit based routes for use in traveler's trip plans. The request will go to the process that manages the store of this data and provides the interface with the map data supplier.

Logical Architecture Reference Flow: fispo_request_route_selection_map_data_update
This data flow is sent from the ISP operator to the Provide Driver and Traveler Services function. It contains a request for an update of the digitized map data used by the process that selects vehicle based routes for use in traveler's trip plans and on-line drover guidance. The request will go to the process that manages the store of this data and provides the interface with the map data supplier.

Logical Architecture Reference Flow: fispo_route_selection_parameters_request
This data flow is sent from the ISP operator to the Provide Driver and Traveler Services function and contains a request for output of the parameters currently being used to govern the selection of routes by the trip planning processes.

Logical Architecture Reference Flow: fispo_route_selection_parameters_update
This data flow is sent from the ISP operator to the Provide Driver and Traveler Services function and contains an update of the parameters used to govern the selection of routes by the route selection process that provide both vehicle routes to travelers and on-line guidance to drivers.

Logical Architecture Reference Flow: fispo_trip_planning_parameters_request
This data flow is sent from the ISP operator to the Provide Driver and Traveler Services function and contains a request for output of the parameters used to govern the selection of routes by the trip planning process.

Logical Architecture Reference Flow: fispo_trip_planning_parameters_update
This data flow is sent from the ISP operator to the Provide Driver and Traveler Services function and contains an update of the parameters used to govern the selection of routes by the trip planning processes.

Map Update Provider => Information Service Provider

Physical Architecture Flow Name: map updates

Map update which could include a new underlying static or real-time map or map layer(s) update.

Logical Architecture Reference Flow: fmup_other_routes_map_data
This data flow is sent by the map update provider to the Provide Driver and Traveler Services function and contains a new copy of the digitized map data used by the process that selects other, i.e. non-vehicle and non-transit, routes.

Logical Architecture Reference Flow: fmup_route_selection_map_data
This data flow is sent by the map update provider to the Provide Driver and Traveler Services function. It contains an update of the digitized map data used by the route selection facility to produce vehicle based routes for trip planning and on-line guidance purposes.

Media => Information Service Provider

Physical Architecture Flow Name: external reports

Traffic and incident information that is collected by the media through a variety of mechanisms (e.g., radio station call-in programs, air surveillance).

Logical Architecture Reference Flow: fm_incident_details
This data flow contains data about an incident that has been reported by a member of the traveling public to the media by mechanisms that are outside of ITS, e.g. car phone. The data flow consists of the following items each of which is defined in its own DDE: media_identity + incident_location + incident_start_time + incident_duration + incident_severity + incident_type.

Logical Architecture Reference Flow: fm_traveler_information
This data flow is sent from the media to the Provide Driver and Traveler Services function and contains information that the media has that might be of interest to travelers planning trips. This may include but not be limited to such things as special events, sports fixtures, etc.

Physical Architecture Flow Name: media information request

Request from the media for current transportation information.

Logical Architecture Reference Flow: fm_incident_information_request
This data flow contains a request for data on incidents to be sent to the Media. The request must specify whether all, current or predicted incidents are required, in the latter case state the time period by date and hour range, and the

geographic area(s) to which it should relate.

Logical Architecture Reference Flow: fm_traffic_information_request
This data flow contains a request from the Media for traffic information. The request must specify the type of information required (flow/congestion) and the geographic area(s) to which it should relate.

Logical Architecture Reference Flow: fm_transit_vehicle_deviations_request
This data flow contains a request for data on schedule deviations of specific transit vehicles or routes.

Other ISP => Information Service Provider

Physical Architecture Flow Name: ISP coordination

Coordination and exchange of transportation information between centers. This flow allows a broad range of transportation information collected by one ISP to be redistributed to many other ISPs and their clients.

Logical Architecture Reference Flow: foisp_data_supply
This data flow is sent to the Provide Driver and Traveler Services function and contains a set of road data covering the local geographic area for use by the similar function in an ISP covering another geographic area. It consists of the following data item which is defined in its own DDE: road_data.

Logical Architecture Reference Flow: foisp_request_data
This data flow is sent to the Provide Driver and Traveler Services function and contains a request for a set of road data covering another geographic area outside that covered by the ISP supporting the local function.

Logical Architecture Reference Flow: foisp_traffic_data
This data flow contains a complete (or partial) set of the traffic data which has been created through fusion of available data sources and sent from another ISP (e.g. a wholesaler of information). This includes current, long term (historical) and predicted link data as well as incident data. The data flow consists of the following items each of which is defined in its own DDE: source_identity + current_data_for_retrieval + long_term_data_for_retrieval + predicted_incidents + predictive_model_data_for_retrieval.

Logical Architecture Reference Flow: foisp_traffic_information_request
This data flow contains a request (either as a subscription or as individual request) from another ISP for available traffic data to be sent. This allows an ISP to act as a wholesaler and send data to other ISPs.

Logical Architecture Reference Flow: foisp_transit_data
This data flow is used to provide data on the current state of transit operations (regarding both incidents and transit vehicle schedule status) and is sent from anther ISP (e.g. a wholesaler of information). The data flow consists of the following items each of which is defined in its own DDE: transit_running_data_for_advisory_output + transit_incident_data.

Logical Architecture Reference Flow: foisp_transit_information_request
This data flow contains a request (either as a subscription or as individual request) from another ISP for available transit data to be sent. This allows an ISP to act as a wholesaler and send data to other ISPs.

Parking Management => Information Service Provider

Physical Architecture Flow Name: parking availability

Parking lot occupancy, availability and payment information.

Logical Architecture Reference Flow: parking_lot_availability

This data flow is sent from the Provide Electronic Payment Services function to the Provide Driver and Traveler Services function. It contains details of the number of spaces available in the lot in response to a previous request for this data. The data flow consists of the following items each of which is defined in its own DDE: parking_lot_identity + parking_lot_spaces + traveler_identity.

Physical Architecture Flow Name: parking lot reservation confirmation

Confirmation for parking lot reservation.

Logical Architecture Reference Flow: advanced_other_charges_confirm

This data flow is used within the Provide Electronic Payment Services function and shows whether or not an advanced parking lot payment transaction has been confirmed or not. It consists of the following data item which is defined in its own DDE: confirmation_flag + credit_identity + stored_credit + parking_lot_cost + vehicle_identity.

Logical Architecture Reference Flow: advanced_traveler_charges_confirm

This data flow is used within the Provide Electronic Payment Services function It contains data about an advanced parking lot charge transaction requested by a traveler and consists of the following data items each of which is defined in its own DDE: confirmation_flag + parking_lot_cost + stored_credit + traveler_identity.

Logical Architecture Reference Flow: parking_lot_price_data

This data flow is used within the Provide Electronic Payment Services function and contains the prices being charged by each parking lot for each of its spaces, together with the time and date for which they apply. It consists of the following data items each of which is defined in its own DDE: parking_lot_identity + parking_lot_price + parking_lot_charge_application_time + vehicle_type_for_charges.

Logical Architecture Reference Flow: parking_lot_reservation_confirm

This data flow is sent from the Provide Electronic Payment Services function to the Provide Driver and Traveler Services function. It contains the confirmation that a previously requested reservation of a space at a parking lot has been confirmed and can be included in a traveler's confirmed trip plan. This data flow consists of the following items each of which is defined in its own DDE: parking_lot_identity + reservation_status + traveler_identity.

Personal Information Access => **Information Service Provider**

Physical Architecture Flow Name: traveler profile

Information about a traveler including equipment capabilities, personal preferences and recurring trip characteristics.

Logical Architecture Reference Flow: traveler_traffic_profile

This data flow contains a traveler's personal profile for obtaining traffic information. This profile is submitted one time and then used to generate future personalized trip information. This profile supports a subscription type of information dissemination to the traveler. It consists of the following data items each of which is defined in its own DDE: traveler_identity + traveler_contact_setting + traveler_device_setting + traveler_traffic_preference_setting.

Logical Architecture Reference Flow: traveler_transit_profile

This data flow contains a traveler's personal profile for obtaining transit information. This profile is submitted one time and then used to generate future personalized trip information. This profile supports a subscription type of information dissemination to the traveler. It consists of the following data items each of which is defined in its own DDE: traveler_identity + traveler_contact_setting + traveler_device_setting + traveler_transit_preference_setting.

Physical Architecture Flow Name: traveler request

Request by a traveler to summon assistance, request information, make a reservation, or initiate any other traveler service.

Logical Architecture Reference Flow: traffic_data_personal_request

This data flow contains the request for the provision of traffic data for output at a traveler's personal device. It consists of the following data items each of which is defined in its own DDE: traffic_data_request + traveler_identity.

Logical Architecture Reference Flow: transit_deviations_personal_request

This data flow is sent by the Provide Driver and Traveler Services function to the Manage Transit function and is a request for data on current transit service deviations for output to a traveler's personal device. When a portable device is being used by the traveler. The request can be modified so that only the data for a transit route specified by the traveler from the portable device is requested. The data flow consists of the following data items each of which is defined in its own DDE: traveler_identity + transit_vehicle_deviation_request + transit_route_number.

Logical Architecture Reference Flow: traveler_map_update_payment_request

This data flow is sent from the Provide Driver and Traveler Services function to the Provide Electronic Payment Services function and contains a request that payment be made for an update of the navigable map database used by the traveler for on-line personal guidance. The payment will be made by debiting the credit identity with the cost through the financial institution terminator. It consists of the following data items each of which is defined in its own DDE: traveler_identity + credit_identity + navigable_map_traveler_update_cost.

Logical Architecture Reference Flow: traveler_personal_current_condition_request

This data flow is used within the Provide Driver and Traveler Services function and contains a request for details of the current conditions, e.g. weather, events, incidents, etc. The request includes the identity of the personal device from which the request was input by the traveler so that the response can be correctly returned. The data flow consists of the following item which is defined in its own DDE: traveler_identity.

Logical Architecture Reference Flow: traveler_personal_display_update_payment_request

This data flow is sent from the Provide Driver and Traveler Services function to the Provide Electronic Payment Services function and contains a request that payment be made for an update of the digitized map data used as background to the displays of traffic and travel information on a traveler's personal device. The payment will be made by debiting the credit identity with the cost through the financial institution terminator. It consists of the following data items each of which is defined in its own DDE: traveler_identity + credit_identity + display_map_traveler_update_cost.

Physical Architecture Flow Name: trip confirmation

Acknowledgement by the driver/traveler of acceptance of a route.

Logical Architecture Reference Flow: traveler_personal_payment_information

This data flow is used within the Provide Driver and Traveler Services function. It contains details of the components of a trip which a traveler has obtained from the input of data to a personal device and for which advanced payment is needed following a trip confirmation. The traveler's identity and credit identity or stored credit from the payment instrument are therefore also included to enable payment to be made. The data flow consists of the following items each of which is defined in its own DDE: credit_identity + parking_space_details + ride_segments + stored_credit + toll_route_segments + traveler_identity.

Logical Architecture Reference Flow: traveler_personal_trip_confirmation

This data flow is used within the Provide Driver and Traveler Services function to confirm the trip details provided as the result of a traveler's previous trip request input from a personal device. It contains the following data item which is defined in its own DDE: paratransit_service_confirmation + traveler_identity + traveler_rideshare_confirmation.

Logical Architecture Reference Flow: traveler_route_accepted

This data flow is used within the Provide Driver and Traveler Services function and contains the acceptance by the traveler of the previously provided route for on-line infrastructure based guidance. Acceptance must be provided before guidance can begin. The data flow consists of the following data item which is defined in its own DDE: route_identity.

Physical Architecture Flow Name: trip request

Request by a driver/traveler for special routing.

Logical Architecture Reference Flow: traveler_personal_trip_request
This data flow is used within the Provide Driver and Traveler Services function and contains data about a traveler's trip request which has been input from a personal device. It consists of the following data items each of which is defined in its own DDE: trip_request + traveler_identity + traveler_rideshare_request.

Logical Architecture Reference Flow: traveler_route_request
This data flow is used within the Provide Driver and Traveler Services function and contains data from which the route requested by a traveler can be determined, or that the previously provided data has been accepted. It consists of the following data items each of which is defined in its own DDE: origin + destination + desired_arrival_time + modes + preferred_routes + preferred_alternate_routes + preferred_ridesharing_options + preferred_route_segments + preferred_transit_options + constraint_on_acceptable_travel_time + constraint_on_number_of_mode_changes + constraint_on_number_of_transfers + constraint_on_eta_change + constraint_on_special_needs + traveler_route_accepted + traveler_identity + traveler_location.

Physical Architecture Flow Name: yellow pages request

Request for information through a yellow pages type service.

Logical Architecture Reference Flow: traveler_personal_transaction_request
This data flow is used within the Provide Driver and Traveler Services function and contains data input by the traveler at a personal device to make reservations for various other (yellow pages) services. It contains the following data items, each of which is defined in its own DDE: yellow_pages_dining_reservation + yellow_pages_lodging_reservation + yellow_pages_ticket_purchase.

Logical Architecture Reference Flow: traveler_personal_yellow_pages_information_request
This data flow is used within the Provide Driver and Traveler Services function and contains a request for data on other (yellow pages) services to be provided to a traveler using a personal device. The traveler identity is used as the means of ensuring that the data produced in response to the request is returned to the correct traveler. As no filtering components are included, all the data currently available will be provided. The data flow consists of the following data item which is defined in its own DDE: traveler_identity.

Remote Traveler Support => **Information Service Provider**

Physical Architecture Flow Name: traveler request

Request by a traveler to summon assistance, request information, make a reservation, or initiate any other traveler service.

Logical Architecture Reference Flow: advanced_tolls_and_charges_roadside_request
This data flow is used by the Manage Transit function to transfer requests for advanced payments for toll and parking lot charges from the traveler (as a transit user) fare payment interface at the roadside, i.e. a transit stop, to the Provide Electronic Payment Services function for subsequent processing. The size of the data flow has been set at less than the sum of the two constituent flows to allow for the fact that they will both not be present for every data transfer. It consists of the following data items each of which is defined in its own DDE: advanced_charges + advanced_tolls.

Logical Architecture Reference Flow: traffic_data_kiosk_request
This data flow is sent by the Provide Driver and Traveler Services function to the Manage Traffic function and contains the request for the provision of traffic data for output at a kiosk. It consists of the following data items each of which is defined in its own DDE: kiosk_identity + traffic_data_request.

Logical Architecture Reference Flow: transit_deviation_kiosk_request
This data flow is sent by the Provide Driver and Traveler Services function to the Manage Transit function and is a request for data on current transit service deviations for output to a kiosk. It consists of the following data items each of which is defined in its own DDE: kiosk_identity + transit_vehicle_deviation_request.

Logical Architecture Reference Flow: traveler_current_condition_request
This data flow is used within the Provide Driver and Traveler Services function and contains a request for details of the current conditions, e.g. weather, events, incidents, etc. The request includes the identity of the kiosk from which the request was input by the traveler so that the response can be correctly returned. The data flow consists of the following item which is defined in its own DDE: kiosk_identity.

Logical Architecture Reference Flow: traveler_payment_information
This data flow is used within the Provide Driver and Traveler Services function. It contains details of the components of a trip which a traveler has obtained from the input of data to a kiosk and for which advanced payment is needed following trip confirmation. The traveler's identity and credit identity or stored credit from the payment instrument are therefore also included to enable payment to be made. The data flow consists of the following items each of which is defined in its own DDE: credit_identity + kiosk_identity + parking_space_details + ride_segments + stored_credit + toll_route_segments.

Logical Architecture Reference Flow: traveler_transaction_request
This data flow is used within the Provide Driver and Traveler Services function and contains data input by the traveler at a kiosk to make reservations for various other (yellow pages) services. It contains the following data items, each of which is defined in its own DDE: yellow_pages_dining_reservation + yellow_pages_lodging_reservation + yellow_pages_ticket_purchase.

Logical Architecture Reference Flow: traveler_yellow_pages_information_request
This data flow is used within the Provide Driver and Traveler Services function and contains a request for data on other (yellow pages) services to be provided to a traveler at the identified kiosk. As no filtering components are included, all the data currently available will be provided. The data flow consists of the following data item which is defined in its own DDE: kiosk_identity.

Physical Architecture Flow Name: trip confirmation

Acknowledgement by the driver/traveler of acceptance of a route.

Logical Architecture Reference Flow: traveler_trip_confirmation
This data flow is used within the Provide Driver and Traveler Services function to confirm the trip details provided as the result of a traveler's previous trip request input from a kiosk. It contains the following data item which is defined in its own DDE: paratransit_service_confirmation + traveler_identity + traveler_rideshare_confirmation.

Physical Architecture Flow Name: trip request

Request by a driver/traveler for special routing.

Logical Architecture Reference Flow: traveler_trip_request
This data flow is used within the Provide Driver and Traveler Services function and contains data about a traveler's trip request which has been input from a kiosk. It consists of the following data items each of which is defined in its own DDE: trip_request + traveler_identity + traveler_rideshare_request.

Physical Architecture Flow Name: yellow pages request

Request for information through a yellow pages type service.

Logical Architecture Reference Flow: traveler_yellow_pages_information_request
This data flow is used within the Provide Driver and Traveler Services function and contains a request for data on other

(yellow pages) services to be provided to a traveler at the identified kiosk. As no filtering components are included, all the data currently available will be provided. The data flow consists of the following data item which is defined in its own DDE: kiosk_identity.

Toll Administration => **Information Service Provider**

Physical Architecture Flow Name: probe data

Aggregate data from probe vehicles including location, speed for a given link or collection of links.

Logical Architecture Reference Flow: vehicle_toll_probe_data

This data flow is sent from the Provide Electronic Payment Services function to the Provide Driver and Traveler Services function. It contains the smoothed average vehicle journey times for the route segment between two toll collection points, and the identity of the route segment. The data is used to calculate link journey times for in-vehicle guidance purposes. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{route_segment_identity + route_segment_journey_time_from_tolls}.

Physical Architecture Flow Name: toll data

Current toll schedules for different types of vehicles as well as advanced toll payment information.

Logical Architecture Reference Flow: advanced_other_tolls_confirm

This data flow is used within the Provide Electronic Payment Services function to confirm the advanced payment of tolls by a driver. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + stored_credit + toll_cost + vehicle_identity.

Logical Architecture Reference Flow: advanced_traveler_tolls_confirm

This data flow is used within the Provide Electronic Payment Services function. It contains data about an advanced toll transaction requested by a traveler and consists of the following data items each of which is defined in its own DDE: confirmation_flag + stored_credit + toll_cost + traveler_identity.

Logical Architecture Reference Flow: toll_price_data

This data flow is used within the Provide Electronic Payment Services function. It contains the price for each road segment to which a toll applies, with the time and date for when it applies. The data flow consists of the following data items each of which is defined in its own DDE: toll_segment_identity + toll_price + toll_price_application_time + vehicle_type_for_tolls.

Traffic Management => **Information Service Provider**

Physical Architecture Flow Name: request fare and price information

Requests for current fare and price information from a service provider that can be used to augment the traffic manager's overall view of current transportation network status.

Logical Architecture Reference Flow: parking_lot_charge_request

This data flow is sent from the Manage Traffic function to the Provide Electronic Payment Services function and contains a request for the current prices being charged for parking lot spaces.

Logical Architecture Reference Flow: toll_price_request

This data flow is sent from the Manage Traffic function to the Provide Electronic Payment Services function and contains a request for the current prices being charged for toll segments on the road and highway network.

Logical Architecture Reference Flow: transit_fare_request

This data flow is sent from the Manage Traffic function to the Provide Electronic Payment Services function and contains a request for the current prices being charged for transit fares.

Physical Architecture Flow Name: traffic information

Current and predicted traffic information, road and weather conditions, incident information, and pricing data. Either raw data, processed data, or some combination of both may be provided by this architecture flow.

Logical Architecture Reference Flow: current_highway_network_state
This data flow is sent by the Manage Traffic function to the Provide Driver and Traveler Services function and contains data about traffic conditions on links in the road network served by the function. The data is used by the route selection and guidance processes in determining the best vehicle routes. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{link_identity + link_journey_time + link_delay}.

Logical Architecture Reference Flow: current_road_network_state
This data flow is sent by the Manage Traffic function to the Provide Driver and Traveler Services function and contains data about traffic conditions on links in the highway network served by the function. The data is used by the route selection and guidance processes in determining the best vehicle routes. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{link_identity + link_journey_time + link_delay}.

Logical Architecture Reference Flow: link_data_for_guidance
This data flow is sent from the Manage Traffic function to the Provide Driver and Traveler Services function. It contains data for use in determining which other ISP('s) must be contacted to obtain data about roads and highways in geographic area(s) outside that served by the local function. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{link_attributes + link_identity + link_ISP_identity}.

Logical Architecture Reference Flow: predicted_incidents
This data flow is used within the Manage Traffic function and contains details of known incidents due to take place in the future. It contains the following data items each of which is defined in its own DDE: list_size + list_size{incident_location + incident_type + incident_severity + incident_description + incident_traffic_impact}.

Logical Architecture Reference Flow: prediction_data
This data flow is used within the Manage Traffic function and is also sent by that function to the Manage Transit and Provide Driver and Traveler Services function. It contains output from the predictive model process showing predictions of traffic data for route segments on the road and highway network served by the Manage Traffic function. The data flow consists of the following items each of which is defined in its own DDE: list_size + list_size{route_segment_identity + route_segment_volume_delay_predictions + route_segment_queue_delay_predictions + route_segment_speed_predictions + route_segment_occupancy_predictions}.

Logical Architecture Reference Flow: sensor_data_for_distribution
This data flow contains raw and processed sensor data. The data flow consists of the following data items each of which is defined in its own DDE: sensor_output_data + roadway_environment_conditions.

Logical Architecture Reference Flow: traffic_data_for_distribution
This data flow is used within the Manage Traffic function. It contains the response to a request for particular data to be retrieved from the stores of current, long term and predictive model data. This data will be used as the basis for traffic information data that is provided to other ITS functions. The data flow consists of the following data items each of which is defined in its own DDE: current_data_for_retrieval + long_term_data_for_retrieval + predictive_model_data_for_retrieval.

Transit Management => **Information Service Provider**

Physical Architecture Flow Name: demand responsive transit plan

Plan regarding overall demand responsive transit schedules and deployment.

Logical Architecture Reference Flow: paratransit_personal_schedule
This data flow is sent from the Manage Transit function to the Provide Driver and Traveler Services function. It consists of the following data items each of which is defined in its own DDE: paratransit_service_details + paratransit_service_cost + traveler_identity.

Physical Architecture Flow Name: transit and fare schedules

Specific transit and fare schedule information including schedule adherence.

Logical Architecture Reference Flow: transit_deviation_data_received
This data flow is used within the Manage Transit function and contains an indication that new data about transit service deviations has been received and is now in the local store of this data. The process(es) receiving this data is(are) expected to take action automatically to output the new data to other functions that are outside the scope of the ITS.

Logical Architecture Reference Flow: transit_fare_data
This data flow is used within the Provide Electronic Payment Services function and contains details of the fares being currently charged for transit services. It consists of the following item which is defined in its own DDE: list_size + list_size{transit_fares}.

Logical Architecture Reference Flow: transit_services_for_advisory_data
This data flow is sent from the Manage Transit function to the Provide Driver and Traveler Services function. It contains a complete set of all the transit routes and the services that run upon them, including timings, etc. that are provided by the transit fleet from which the data was requested, for use in the preparation of driver and traveler advisory information for output on-board vehicles. It consists of the following data item which is defined in its own DDE: transit_services.

Logical Architecture Reference Flow: transit_services_for_guidance
This data flow is sent from the Manage Transit function to the Provide Driver and Traveler Services function. It contains a complete set of all the transit routes and the services that run upon them, including timings, etc. that are provided by the transit fleet from which the data was requested, for use in the preparation of data for output as on-line driver and traveler guidance data. The data flow consists of the following data items each of which is defined in its own DDE: 1{transit_services_for_output}2 + traveler_identity.

Logical Architecture Reference Flow: transit_vehicle_deviations_details
This data flow is used within the Manage Transit function. It contains details of the deviations of transit vehicles from their published routes and schedules and is used as a source of data to be sent to processes in other functions. The data flow consists of the following data items each of which is defined in is own DDE: transit_vehicle_eta + transit_vehicle_collected_trip_data + transit_vehicle_deviation_update + transit_vehicle_location + transit_vehicle_schedule_deviations.

Physical Architecture Flow Name: transit incident information

Information on transit incidents that impact transit services for public dissemination.

Logical Architecture Reference Flow: transit_incident_data
This data flow contains information about an incident that has occurred within part of the transit operations network, e.g. transit stop or mode interchange point. The location and details of the incident will be included in the information, subject to any constraints applied by the transit agency on providing information to outside sources. The data flow has been sized to enable two pages of text at 65 lines/page and 80 characters/line to be accommodated.

Physical Architecture Flow Name: transit request confirmation

Confirmation of a request for transit information or service.

Logical Architecture Reference Flow: advanced_other_fares_confirm
This data flow is used within the Provide Electronic Payment Services function to confirm the advanced payment of a transit fare by a transit user. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + stored_credit + transit_fare + traveler_identity.

Logical Architecture Reference Flow: advanced_tolls_and_charges_vehicle_request
This data flow is used by the Manage Transit function to transfer requests for advanced payments for toll and parking lot charges from the traveler (as a transit user) fare payment interface in a transit vehicle to the Provide Electronic Payment Services function for subsequent processing. The size of the data flow has been set at less than the sum of the two constituent flows to allow for the fact that they will both not be present for every data transfer. It consists of the following data items each of which is defined in its own DDE: advanced_charges + advanced_tolls.

Logical Architecture Reference Flow: advanced_traveler_fares_confirm
This data flow is used within the Provide Electronic Payment Services function to show whether advanced fare payment by a traveler planning a trip has been refused or cleared. The traveler will be using facilities in the Provide Driver and Traveler Services function to generate the trip request. The data flow consists of the following data items each of which is defined in its own DDE: confirmation_flag + stored_credit + transit_fare + traveler_identity.

Vehicle => Information Service Provider

Physical Architecture Flow Name: traveler profile

Information about a traveler including equipment capabilities, personal preferences and recurring trip characteristics.

Logical Architecture Reference Flow: traveler_profile_from_vehicle
This data flow contains a traveler's personal profile which is submitted one time and then used to generate future personalized trip information. This profile supports a subscription type of information dissemination to the traveler. It consists of the following data items each of which is defined in its own DDE: traveler_traffic_profile +traveler_transit_profile.

Physical Architecture Flow Name: traveler request

Request by a traveler to summon assistance, request information, make a reservation, or initiate any other traveler service.

Logical Architecture Reference Flow: advanced_fares_and_charges_request
This data flow is used within the Provide Electronic Payment Services function to transfer requests for advanced payments from the driver interface for subsequent processing. The size of the data flow has been set at less than the sum of the two constituent flows to allow for the fact that they will both not be present for every data transfer. It consists of the following data items each of which is defined in its own DDE: advanced_fare_details + advanced_parking_lot_charges.

Logical Architecture Reference Flow: advanced_tolls_and_fares_request
This data flow is used within the Provide Electronic Payment Services function to transfer requests for advanced payments from the driver parking lot charge payment interface for subsequent processing. The size of the data flow has been set at less than the sum of the two constituent flows to allow for the fact that they will both not be present for every data transfer. It consists of the following data items each of which is defined in its own DDE: advanced_fare_details + advanced_tolls.

Logical Architecture Reference Flow: advisory_data_request
This data flow is used within the Provide Driver and Traveler Services function to request that advisory data be output to a driver or a traveler in a vehicle. The scope and transit route number data will be provided by the driver or transit

user, whilst the vehicle location will be provided automatically. The data flow consists of the following data items each of which is defined in its own DDE: advisory_data_scope + vehicle_location_for_advisories + transit_route_number + transit_vehicle_identity.

Logical Architecture Reference Flow: driver_map_update_payment_request
This data flow is sent from the Provide Driver and Traveler Services function to the Provide Electronic Payment Services function and contains a request that payment be made for an update of the navigable map database used for on-line vehicle guidance. The payment will be made by debiting the credit identity with the cost through the financial institution terminator. It consists of the following data items each of which is defined in its own DDE: vehicle_identity + credit_identity + navigable_map_vehicle_update_cost.

Physical Architecture Flow Name: trip confirmation

Acknowledgement by the driver/traveler of acceptance of a route.

Logical Architecture Reference Flow: vehicle_guidance_route_accepted
This data flow is used within the Provide Driver and Traveler Services function and contains the acceptance by the driver of the previously provided route for on-line guidance. Acceptance must be provided before guidance can begin. The data flow consists of the following data item which is defined in its own DDE: route_identity.

Physical Architecture Flow Name: trip request

Request by a driver/traveler for special routing.

Logical Architecture Reference Flow: vehicle_route_request
This data is flow used within the Provide Driver and Traveler Services function and contains a request for on-line guidance of the vehicle. This will have been generated by the driver and will include the necessary source and destination data from which a route can be computed. It consists of the following data items each of which is defined in its own DDE: constraint_on_acceptable_travel_time + constraint_on_eta_change + constraint_on_special_needs + constraint_on_load_classification + constraint_on_ahs_lanes + constraint_on_interstate + constraint_on_urban + constraint_on_vehicle_type + destination + departure_time + desired_arrival_time + origin + preferred_routes + preferred_alternate_routes + preferred_route_segments + vehicle_location_for_dynamic_guidance + vehicle_identity.

Physical Architecture Flow Name: vehicle probe data

Vehicle probe data indicating identity, route segment identity, link time and location.

Logical Architecture Reference Flow: vehicle_guidance_probe_data
This data flow is used within the Provide Driver and Traveler Services function and contains the time at which a vehicle was at a route segment end point. This data will be used to calculate the actual vehicle journey time for the route segment which may supplement or replace data gathered from other sources. This data will be used for in-vehicle guidance purposes. The data flow consists of the following data items each of which is defined in its own DDE route_segment_identity + time + vehicle_identity.

Physical Architecture Flow Name: yellow pages request

Request for information through a yellow pages type service.

Logical Architecture Reference Flow: yellow_pages_advisory_requests
This data flow is used within the Provide Driver and Traveler Services function to request that data about yellow pages services be output to a driver or a transit user in a vehicle or that a yellow pages services reservation b made. The scope and transit route number data will be provided by the driver or transit user, whilst the vehicle location will be provided automatically. The data flow consists of the following data items each of which is defined in its own DDE: advisory_data_scope + vehicle_location_for_advisories + transit_route_number + transit_vehicle_identity + yellow_pages_dining_reservation + yellow_pages_lodging_reservation + yellow_pages_ticket_purchase.

Weather Service => **Information Service Provider**

Physical Architecture Flow Name: weather information

Accumulated predicted and current weather data (e.g., temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.).

Logical Architecture Reference Flow: fws_current_weather
This data flow is sent to the Manage Traffic function and the Provide Driver and Traveler Services functions. It contains details of the current weather conditions, e.g. temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.

Logical Architecture Reference Flow: fws_predicted_weather
This data flow is sent to the Manage Traffic and Provide Driver and Traveler Services functions. It contains details of the predicted weather conditions, e.g. temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.

Yellow Pages Service Providers => **Information Service Provider**

Physical Architecture Flow Name: provider profile data

Information supplied by a service provider (e.g., a hotel or restaurant) that identifies the service provider and provides details of the service offering. This flow covers initial registration of a service provider and subsequent submittal of new information and status updates so that data currency is maintained.

Logical Architecture Reference Flow: fypsp_provider_profile_update
This data flow is used to update the current yellow pages service provider profile. This update could be in the form of a change to provider information or services, or could be the deletion of the provider from the database.

Logical Architecture Reference Flow: fypsp_request_provider_registration
This data flow is sent to the Provide Driver and Traveler Services function from the information and service provider to request registration as a provider of yellow pages data for another process within this function. The data flow includes details of the provider, credit identity of the provider, geographic area for which data can be provided and available yellow pages services.

Physical Architecture Flow Name: travel service info

Reservation information or yellow pages data.

Logical Architecture Reference Flow: fypsp_transaction_confirmation
This data flow is sent by the information and service provider to the Provide Driver and Traveler Services function and contains confirmation that a transaction requested by a traveler has (or has not) successfully taken place.

Logical Architecture Reference Flow: fypsp_yellow_pages_data
This data flow is sent from the information and services provider to the Provide Driver and Traveler Services function. It provides information on yellow pages services in three forms comprising that of general interest, more specific items and transaction information.

2.11.4 Subsystem Architecture Flow Diagram

ISP

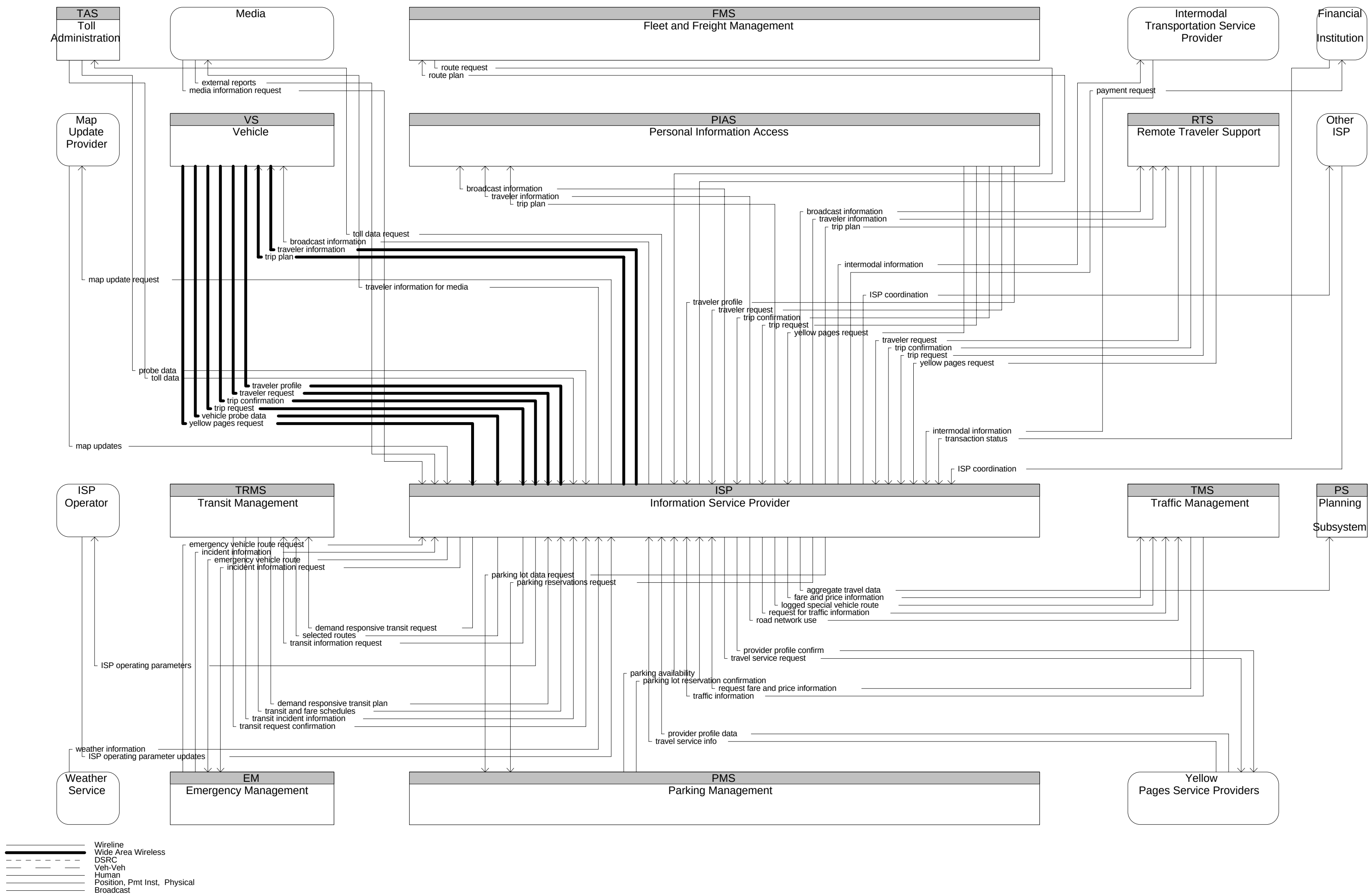


Figure 2.11-2 Architecture Flow Diagram for ISP

2.12 Personal Information Access

This subsystem provides the capability for travelers to receive formatted traffic advisories from their homes, place of work, major trip generation sites, personal portable devices, and over multiple types of electronic media. These capabilities shall also provide basic routing information and allow users to select those transportation modes that allow them to avoid congestion, or more advanced capabilities to allow users to specify those transportation parameters that are unique to their individual needs and receive travel information. This subsystem shall provide capabilities to receive route planning from the infrastructure at fixed locations such as in their homes, their place of work, and at mobile locations such as from personal portable devices and in the vehicle or perform the route planning process at a mobile information access location. This subsystem shall also provide the capability to initiate a distress signal and cancel a prior issued manual request for help.

2.12.1 Alternative Configurations

Personal Information Access and Remote Traveler Access are two ways that the traveler interacts with ITS services Figure 2.12-1. These subsystems represent the equipment accepting commands from the traveler and displaying information to the traveler. When a traveler is in a vehicle, the Vehicle Subsystem performs ITS access type functions. A traveler on a bus uses the transit vehicle to access ITS services. The Personal Information Access subsystem is owned by the traveler (as opposed to the agency or organization providing data) and may be found at home, in the office, or mobile with the traveler. The access to information is through an information provider of some kind. Through the PIAS, the traveler has access to any information source available (as opposed to an RTS which typically provides access to the agency responsible for the kiosk).

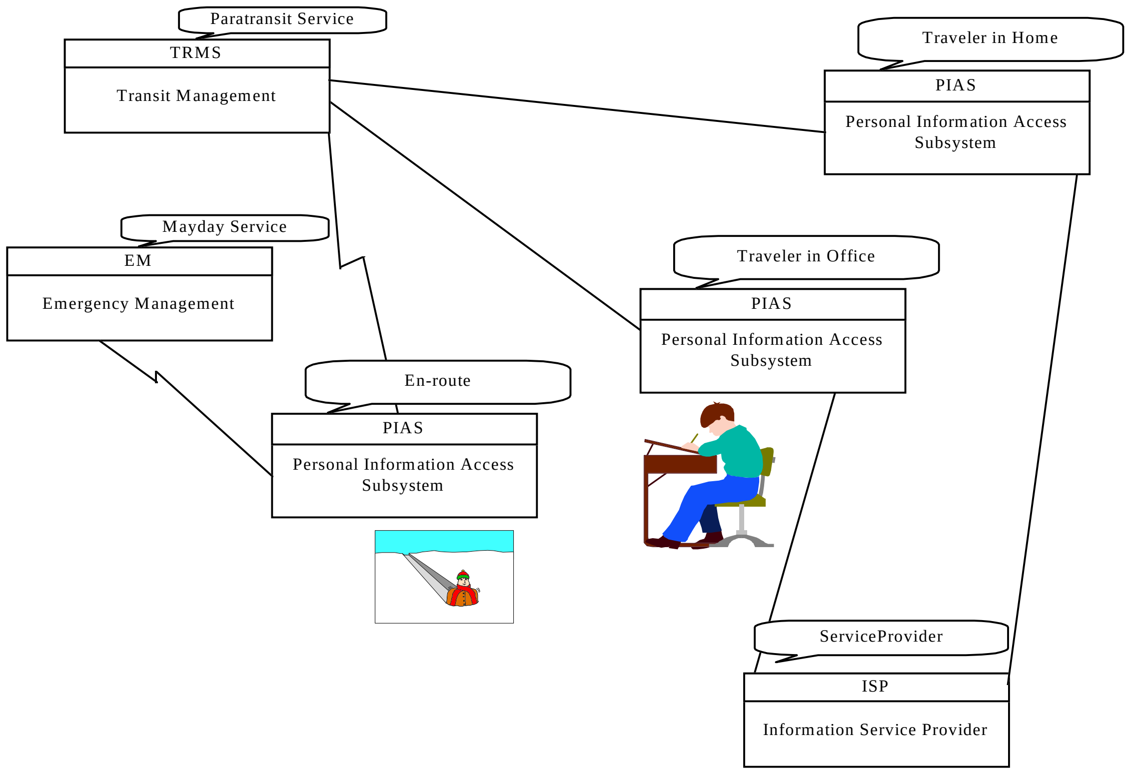


Figure 2.12-1 Alternative Configurations for Personal Information Access

2.12.2 Subsystem Equipment Packages and Supporting Process Specifications for PIAS

Personal Autonomous Route Guidance
This Equipment package provides multi-modal route planning and transition by transition route guidance. It provides autonomous route guidance in the absence of real-time information or factors information provided by the infrastructure into its route selection and guidance algorithms. The equipment package also includes those truly autonomous systems that are not configured to receive or process any external data.

Process Specifications
6.8.1.1.3 Provide Personal Portable Device Autonomous Guidance

Overview: This process shall provide autonomous on-line guidance when requested by the traveler from a personal portable device. It shall calculate the route using data obtained from a navigable map database stored in the traveler's personal portable device. Guidance shall be provided by the process in the form of actual instructions to the traveler, e.g. cross the road here, take the subway to a specific station. The process shall provide guidance for the shortest route, within the preferences and constraints specified by the traveler in the guidance request.

6.8.1.2 Provide Personal Portable Device Guidance Interface

Overview: This process shall be responsible for providing a user interface for the traveler through which personal guidance can be delivered. The process shall enable the traveler to input data to request a suitable route. This process shall be capable of supporting two types of route guidance: dynamic (infrastructure based guidance is provided to the personal portable device), and autonomous (the personal portable device uses only locally available data- there is no information provided by the infrastructure). The process shall also act as the interface for output of on-line guidance to the traveler. Multimodal routes shall be supported by the process. The process shall not provide on-line guidance until the route has been accepted by the traveler. For those forms of guidance that require an on-board map database, the process shall provide an interface through which the traveler may obtain and pay for an initial copy of the database plus updates when needed. The process shall support inputs from the traveler in either manual or audio form, and shall provide outputs in audible or visual forms. It shall enable the visual output to be either in hardcopy, or display. Both types of output shall be produced in such a way that in using them the traveler does not become a hazard to other travelers.

6.8.1.3 Process Personal Portable Device Location Data

Overview: This process shall provide the traveler's current location. It shall calculate the location from one or more sources of position data such as GPS or DGPS, and shall refine its calculations using techniques such as map matching, dead reckoning, etc. The process shall provide the location to the to other processes for use in autonomous and dynamic guidance. This location should be precise as is practical within cost and technology constraints. It is intended for use by traveler personal navigation and guidance systems, as well as emergency notification systems.

6.8.1.4 Update Traveler Navigable Map Database

Overview: This process shall update the traveler's navigable database based on digitized data obtained from a map provider, or other appropriate data source. The update shall be initiated by the traveler through another process. The process shall have the capability to allow a financial transaction (to pay for the update) to be completed using processes in the Provide Electronic Payment Services function. When the new map data is received, it shall be loaded by the process into the traveler_map_database data store for use by other processes. The result of the update request (successful or not) shall be sent back to the traveler interface process for output to the traveler.

6.8.3.4 Update Traveler Personal Display Map Data

Overview: This process shall provide updates to the digitized map data used as the background for displays on travelers' personal devices. These displays include details of traffic, trip and travel information for use by travelers. The process shall obtain the new map data from a map provider process or some other appropriate data source on request from the traveler via the traveler interface process. The process shall load data into the map_data_for_traveler_personal_displays data store. The data will be compatible with the types of displays that are found on personal devices.

Personal Basic Information Reception

This Equipment package shall provide the capability for travelers to interface with the ISP Subsystem Basic Information Broadcast Equipment package and receive formatted traffic advisories including accurate traveling information concerning available travel options and their availability, and congestion information from their Personal Information Access Subsystem to include their homes, place of work, major trip generation sites, personal portable devices, and over multiple types of electronic media such as facsimile machines, portable AM/FM radios, and a pager processor.

Process Specifications

6.8.3.2 Provide Traveler with Personal Travel Information

Overview: This process shall provide the traveler (using a personal device) with data about all requested trip, traffic, transit, other (yellow pages) services information, confirmation of any requested reservations, and payments made as part of confirmed trip plans. The data shall be sent by the process to an interface process which is responsible for its actual output to the traveler. This data shall include digitized map data to act as the background to the output when the data is shown in a suitable format. This process shall request data from other ITS functions or be sent it as a result of requests from another process.

6.8.3.3 Provide Traveler Personal Interface

Overview: This process shall provide an interface in a personal device through which travelers can plan and confirm trips, as well as obtain current traffic and transit information. The process shall support trip planning and confirmation of other (yellow pages) services such as lodging, restaurants, theaters, and other tourist activities. The process shall be able to load in the traveler_personal_regular_data store frequently used information such as traveler identity (the owner of the personal device), home and work locations, etc. This will reduce the amount of input needed by the traveler for each trip request. The process shall also carry out input data verification and require input confirmation, with the traveler, before passing the data to other processes. The traveler's payment information and location (when traveler is using a portable device) shall be obtained by this process from other processes. The process shall support inputs from the traveler in both manual and audio form, and shall provide its outputs in audible and visual forms that are consistent with a personal device. This process shall include forms suitable for travelers with hearing and vision physical disabilities. The process shall display data for as long as required by the traveler and must enable viewing of previously output data. When used with a portable device, the process shall provide the traveler the option to filter the data (to be displayed) relevant to the travelers current location.

Personal Interactive Information Reception

This Equipment package shall provide the capability for travelers to interface with the ISP Subsystem Infrastructure Equipment packages including the Interactive Infrastructure Information Equipment package, and the Infrastructure Provided Route Selection, Yellow Pages and Reservation, and Dynamic Ridesharing Equipment packages. These capabilities shall be provided using the Personal Information Access Subsystem equipment such as cellular telephone, interactive TV, Personal Computer, and pager with alpha display using communication medium and equipment such as two-way radio, CATV, and wireless data transceivers.

Process Specifications

6.8.3.1 Get Traveler Personal Request

Overview: This process shall receive traveler requests from a personal device (portable, or non portable) then provide support for trip planning, traffic, transit and other (yellow pages) services information, trip confirmation, yellow pages services confirmation, and payment requests. The process shall send these requests to the appropriate processes within the Provide Driver and Traveler Services function for further processing to generate responses. The interface to the traveler shall be provided through a separate process, from which input to this process originates.

6.8.3.2 Provide Traveler with Personal Travel Information

Overview: This process shall provide the traveler (using a personal device) with data about all requested trip, traffic, transit, other (yellow pages) services information, confirmation of any requested reservations, and payments made as part of confirmed trip plans. The data shall be sent by the process to an interface process which is responsible for its actual output to the traveler. This data shall include digitized map data to act as the background to the output when the data is shown in a suitable format. This process shall request data from other ITS functions or be sent it as a result of requests from another process.

6.8.3.3 Provide Traveler Personal Interface

Overview: This process shall provide an interface in a personal device through which travelers can plan and confirm trips, as well as obtain current traffic and transit information. The process shall support trip planning and confirmation of other (yellow pages) services such as lodging, restaurants, theaters, and other tourist activities. The process shall be able to load in the traveler_personal_regular_data store frequently used information such as traveler identity (the owner of the personal device), home and work locations, etc. This will reduce the amount of input needed by the traveler for each trip request. The process shall also carry out input data verification and require input confirmation, with the traveler, before passing the data to other processes. The traveler's payment information and location (when traveler is using a portable device) shall be obtained by this process from other processes. The process shall support inputs from the traveler in both manual and audio form, and shall provide its outputs in audible and visual forms that are consistent with a personal device. This process shall include forms suitable for travelers with hearing and vision physical disabilities. The process shall display data for as long as required by the traveler and must enable viewing of previously output data. When used with a portable device, the process shall provide the traveler the option to filter the data (to be displayed) relevant to the travelers current location.

7.5.3 Provide Personal Payment Instrument Interface

Overview: This process shall be responsible for providing the interface through which credit identity or stored credit may be collected from the tag being used by a traveler with a personal device. The process shall support the collection of this data from any location in which the device (and hence the tag) is being used. It shall provide an interface through which the credit identity can be used for the payment of advanced tolls, parking lot charges, transit fares, display updates, and/or map updates. The process shall also enable the stored credit value on the tag to be used for the same purposes.

Personal Mayday I/F
This Equipment package shall provide the capability to initiate a distress signal and cancel a prior issued manual request for help using the Personal Information Access Subsystem. This capability shall be provided using equipment such as a processor to automatically dial the Emergency Management Subsystem and provide location.

Process Specifications
6.8.1.5 Provide Traveler Emergency Message Interface

Overview: This process shall provide an emergency notification interface for a traveler using a personal portable device. The emergency notification interface shall enable the output of messages generated by a traveler's emergency request to another process.

6.8.2.1 Build Traveler Personal Security Message

Overview: This process shall respond to the input of a request from a traveler for action by the emergency services. Input of the request shall be received by the process from the traveler via a panic button or some other functionally similar form of input device provided as part of the traveler's personal portable device. When the input is received, the process shall send a message to the communications process, containing the traveler's current location and identity.

6.8.2.2 Provide Traveler Emergency Communications Function

Overview: This process shall prepare and send an emergency message from a traveler's personal portable device to the Manage Emergency Services function. The message shall only be sent by the process in response to data received from another process that monitors traveler inputs. Once an emergency message has been sent, the process shall send a message to that effect to another process for output to the traveler. The process shall then await a response from the Manage Emergency Services function, and when received again send a message to the other process for output to the traveler. Output of the emergency message to the Manage Emergency Services function shall be repeated by the process at regular intervals until a response is received.

Personal Provider-Based Route Guidance
This Equipment package coordinates with an ISP-Based route planning service to select a suggested route plan that is tailored to the traveler's preferences. Coordination may continue during the trip so that the route plan can be modified to account for new information. Many equipment configurations are possible including systems that provide a basic route plan to the traveler as well as more sophisticated systems that can provide transition by transition guidance to the traveler along a multi-modal route plan.

Process Specifications
6.8.1.1.1 Determine Personal Portable Device Guidance Method

Overview: This process shall act as the interface for personal guidance requests received from travelers with personal portable devices. The process shall select the best method for personal guidance based on data in the traveler's request Two methods shall be available to the process, comprising dynamic infrastructure based guidance is provided to the personal portable device), and autonomous (the personal portable device uses only locally available data- there is no information provided by the infrastructure) If the communications link to the central source fails, the process shall use the last set of guidance data that was received, and if this is not sufficient for the traveler to reach the requested destination, automatically revert to the use of autonomous guidance using local data only.

6.8.1.1.2 Provide Personal Portable Device Dynamic Guidance

Overview: This process shall enable dynamic traveler guidance data to be calculated. The process shall base its guidance request on the data input by the traveler from a personal portable device through other processes, and on the traveler's current location as provided by another process.

6.8.1.2 Provide Personal Portable Device Guidance Interface

Overview: This process shall be responsible for providing a user interface for the traveler through which personal guidance can be delivered. The process shall enable the traveler to input data to request a a suitable route. This process shall be capable of supporting two types of route guidance: dynamic (infrastructure based guidance is provided to the personal portable device), and autonomous (the personal portable device uses only locally available data- there is no information provided by the infrastructure). The process shall also act as the interface for output of on-line guidance to the traveler. Multimodal routes shall be supported by the process. The process shall not provide on-line guidance until the route has been accepted by the traveler. For those forms of guidance that require an on-board map database, the process shall provide an interface through which the traveler may obtain and pay for an initial copy of the database plus updates when needed. The process shall support inputs from the traveler in either manual or audio form, and shall provide outputs in audible or visual forms. It shall enable the visual output to be either in hardcopy, or display. Both types of output shall be produced in such a way that in using them the traveler does not become a hazard to other travelers.

6.8.1.3 Process Personal Portable Device Location Data

Overview: This process shall provide the traveler's current location. It shall calculate the location from one or more sources of position data such as GPS or DGPS, and shall refine its calculations using techniques such as map matching, dead reckoning, etc. The process shall provide the location to the to other processes for use in autonomous and dynamic guidance. This location should be precise as is practical within cost and technology constraints. It is intended for use by traveler personal navigation and guidance systems, as well as emergency notification systems.

6.8.1.4 Update Traveler Navigable Map Database

Overview: This process shall update the traveler's navigable database based on digitized data obtained from a map provider, or other appropriate data source. The update shall be initiated by the traveler through another process. The process shall have the capability to allow a financial transaction (to pay for the update) to be completed using processes in the Provide Electronic Payment Services function. When the new map data is received, it shall be loaded by the process into the traveler_map_database data store for use by other processes. The result of the update request (successful or not) shall be sent back to the traveler interface process for output to the traveler.

2.12.3 Subsystem Interfaces for PIAS

Emergency Management => **Personal Information Access**

Physical Architecture Flow Name: emergency acknowledge

Acknowledge request for emergency assistance and provide additional details regarding actions and verification requirements.

Logical Architecture Reference Flow: emergency_request_personal_traveler_acknowledge

This data flow is used by the Manage Emergency Services function to confirm that the request for emergency services previously sent by the traveler has been received from a personal device and is therefore sent to the Provide Driver and Traveler Services function for output. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Information Service Provider => **Personal Information Access**

Physical Architecture Flow Name: broadcast information

General broadcast information that contains link travel times, incidents, advisories, transit services and a myriad of other traveler information.

Logical Architecture Reference Flow: traffic_data_for_personal_devices

This data flow is used to provide data on the traffic flowing in the road network, plus that which is predicted to flow in the network for output to a traveler's personal device. The data flow has been sized to take account of the need to only supply a subset of the full traffic data to the case where the device is a portable device. This subset will depend on the route segment list included in the request, which for sizing purposes has been assumed to have a maximum value of 25. Thus, the size expression below is based on sampling a percentage of the data equivalent to an N route segment square area (i.e. N/total number of segments). The flow consists of the following items each of which is defined in its own DDE: source_identity + current_data_for_broadcast + predicted_data_for_broadcast + traveler_identity.

Logical Architecture Reference Flow: transit_deviations_for_personal_devices

This data flow contains current transit service deviations for a particular route. This data will be output to a traveler's personal device. It consists of the following data items each of which is defined in its own DDE: traveler_identity + list_size + list_size{transit_vehicle_identity + transit_vehicle_achieved_time + transit_route_segment_number}.

Physical Architecture Flow Name: traveler information

Traveler information comprised of traffic status, advisories, incidents, responses to traveler requests (e.g., traveler routing, yellow pages), payment information and many other travel-related data updates and confirmations.

Logical Architecture Reference Flow: traffic_data_for_personal_devices

This data flow is used to provide data on the traffic flowing in the road network, plus that which is predicted to flow in the network for output to a traveler's personal device. The data flow has been sized to take account of the need to only supply a subset of the full traffic data to the case where the device is a portable device. This subset will depend on the route segment list included in the request, which for sizing purposes has been assumed to have a maximum value of 25. Thus, the size expression below is based on sampling a percentage of the data equivalent to an N route segment square area (i.e. N/total number of segments). The flow consists of the following items each of which is defined in its own DDE: source_identity + current_data_for_broadcast + predicted_data_for_broadcast + traveler_identity.

Logical Architecture Reference Flow: transit_deviations_for_personal_devices

This data flow contains current transit service deviations for a particular route. This data will be output to a traveler's personal device. It consists of the following data items each of which is defined in its own DDE: traveler_identity + list_size + list_size{transit_vehicle_identity + transit_vehicle_achieved_time + transit_route_segment_number}.

Logical Architecture Reference Flow: traveler_map_update_payment_response

This data flow is sent from the Provide Electronic Payment Services function to the Provide Driver and Traveler Services function and contains the response to a previous request from the traveler that payment be made for an update of the navigable map database used for on-line traveler guidance. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + traveler_identity.

Logical Architecture Reference Flow: traveler_personal_display_update_payment_response

This data flow is sent from the Provide Electronic Payment Services function to the Provide Driver and Traveler Services function and contains the response to a previous request from the traveler that payment be made for an update of the digitized map data used as background to the displays of traffic and travel information on a traveler's personal device. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + traveler_identity.

Logical Architecture Reference Flow: traveler_personal_payment_confirmation

This data flow is used within the Provide Driver and Traveler Services function to indicate the payment for a confirmed trip has been successfully completed, or that the total cost can now be deducted from the credit stored on the traveler's payment instrument. The request for payment will have been initiated by input from the traveler to a personal device. The data flow consists of the following data items each of which is defined in its own DDE: advanced_tolls_confirm + advanced_fares_confirm + advanced_parking_lot_charges_confirm + credit_identity + stored_credit + traveler_identity + traveler_total_trip_cost.

Logical Architecture Reference Flow: traveler_personal_transaction_confirmation
This data flow is used within the Provide Driver and Traveler Services function to confirm any reservations made by the traveler from a personal device. These reservations will be based on information obtained by the traveler from previous data input and output through the device. The data flow consists of the following data items each of which is defined in its own DDE: credit_identity + traveler_identity + transaction_number + yellow_pages_cost + yellow_pages_lodging_reservation_confirmation + yellow_pages_dining_reservation_confirmation + yellow_pages_ticket_purchase_confirmation.

Logical Architecture Reference Flow: traveler_personal_yellow_pages_data
This data flow is used within the Provide Driver and Traveler Services function and contains details of other (yellow pages) services which is to be sent to the traveler interface facility for output using a kiosk. The size of the data flow has been set to take account of the need to provide only a small percentage of the total yellow pages data that is available. The data flow consists of the following data items each of which is defined in its own DDE: traveler_identity + yellow_pages_general_information + yellow_pages_specific_information + yellow_pages_transaction_information.

Physical Architecture Flow Name: trip plan

A sequence of links and special instructions comprising a trip plan indicating efficient routes for navigating the links. Normally coordinated with traffic conditions, other incidents, preemption and prioritization plans.

Logical Architecture Reference Flow: traveler_guidance_route
This data flow is used within the Provide Driver and Traveler Services function and contains the data for a traveler's route which has been produced following a route request from the traveler. It consists of the following data items each of which is defined in its own DDE: route_identity + traveler_route + traveler_identity.

Logical Architecture Reference Flow: traveler_personal_trip_information
This data flow is used within the Provide Driver and Traveler Services function and contains information about a proposed trip that the traveler has requested earlier from the personal device. It consists of the following data items each of which is defined in its own DDE: current_conditions + [paratransit_personal_schedule | route | rideshare_response] + traveler_identity + traveler_total_trip_cost.

Location Data Source => Personal Information Access

Physical Architecture Flow Name: position fix

Information which provides a traveler or vehicles geographical position.

Logical Architecture Reference Flow: From_Location_Data_Source
This data is sent to a number of processes in the Provide Driver and Traveler Services function. It contains the current state of such things as GPS signals, magnetic flux and other location data sources. The data is sent to sensor processes for conversion into an actual vehicle location. The size estimate below is based on an equivalent digitized form (e.g. location_identity) even though it may actually be an entirely form .

Map Update Provider => Personal Information Access

Physical Architecture Flow Name: map updates

Map update which could include a new underlying static or real-time map or map layer(s) update.

Logical Architecture Reference Flow: fmup_traveler_map_update
This data flow is sent from the map update provider to the Provide Driver and Traveler Services function and contains data for a new guidance map to be used by the on-line traveler guidance facility.

Logical Architecture Reference Flow: fmup_traveler_map_update_cost
This data flow is sent from the map update provider to the Provide Driver and Traveler Services function and contains the cost for a new navigable map database to be used by the traveler personal on-line guidance facility.

Logical Architecture Reference Flow: fmup_traveler_personal_display_update
This data flow is sent from the map update provider to the Provide Driver and Traveler Services function. It contains the digitized map data that can be used as background to displays of traffic, trip and travel information that are output to a personal device for use by travelers. The data will be different to that sent for output of similar displays at a kiosk since the type of display is likely to be different (smaller) in this case.

Logical Architecture Reference Flow: fmup_traveler_personal_display_update_cost
This data flow is sent from the map update provider to the Provide Driver and Traveler Services function and contains the cost for a new set of digitized map data to be used as the background to displays of traffic and travel information being output by a traveler's personal device.

Payment Instrument => **Personal Information Access**

Physical Architecture Flow Name: payment

Payment of some kind (e.g., toll, parking, fare) by traveler which in most cases can be related to a credit account.

Logical Architecture Reference Flow: fpi_traveler_personal_input_credit_identity
This data flow is sent from the payment instrument to the Provide Electronic Payment Services function. It contains the data necessary to allow automatic billing of the user of the payment instrument when they are using a personal device, such as a Personal Digital Assistant (PDA) or similar type of unit, that can be used by the traveler to provide travel information, trip planning, or on-line guidance during a intermodal trip. A payment instrument is a device that can be used to make payments, e.g. a debit card, or a stored value card which itself contains credit that can be used to make payments. It will belong to the financial institution responsible for its issue and not to the user.

Personal Information Access => **Emergency Management**

Physical Architecture Flow Name: emergency notification

An emergency request for assistance originated by a traveler using an in-vehicle, public access, or personal device. Sufficient information is provided so that the recipient can determine the location of the emergency as a minimum. Additional information identifying the requestor and requesting device and the nature and severity of the emergency may also be provided (and required) by some systems.

Logical Architecture Reference Flow: emergency_request_personal_traveler_details
This data flow is used by the Provide Driver and Traveler Services function to send data about an emergency declared by a traveler using a personal device to the Manage Emergency Services function. It contains the following data items each of which is defined in its own DDE: date + time + traveler_personal_emergency_request.

Personal Information Access => **Information Service Provider**

Physical Architecture Flow Name: traveler profile

Information about a traveler including equipment capabilities, personal preferences and recurring trip characteristics.

Logical Architecture Reference Flow: traveler_traffic_profile

This data flow contains a traveler's personal profile for obtaining traffic information. This profile is submitted one time and then used to generate future personalized trip information. This profile supports a subscription type of information dissemination to the traveler. It consists of the following data items each of which is defined in its own DDE: traveler_identity + traveler_contact_setting + traveler_device_setting + traveler_traffic_preference_setting.

Logical Architecture Reference Flow: traveler_transit_profile

This data flow contains a traveler's personal profile for obtaining transit information. This profile is submitted one time and then used to generate future personalized trip information. This profile supports a subscription type of information dissemination to the traveler. It consists of the following data items each of which is defined in its own DDE: traveler_identity + traveler_contact_setting + traveler_device_setting + traveler_transit_preference_setting.

Physical Architecture Flow Name: traveler request

Request by a traveler to summon assistance, request information, make a reservation, or initiate any other traveler service.

Logical Architecture Reference Flow: traffic_data_personal_request

This data flow contains the request for the provision of traffic data for output at a traveler's personal device. It consists of the following data items each of which is defined in its own DDE: traffic_data_request + traveler_identity.

Logical Architecture Reference Flow: transit_deviations_personal_request

This data flow is sent by the Provide Driver and Traveler Services function to the Manage Transit function and is a request for data on current transit service deviations for output to a traveler's personal device. When a portable device is being used by the traveler. The request can be modified so that only the data for a transit route specified by the traveler from the portable device is requested. The data flow consists of the following data items each of which is defined in its own DDE: traveler_identity + transit_vehicle_deviation_request + transit_route_number.

Logical Architecture Reference Flow: traveler_map_update_payment_request

This data flow is sent from the Provide Driver and Traveler Services function to the Provide Electronic Payment Services function and contains a request that payment be made for an update of the navigable map database used by the traveler for on-line personal guidance. The payment will be made by debiting the credit identity with the cost through the financial institution terminator. It consists of the following data items each of which is defined in its own DDE: traveler_identity + credit_identity + navigable_map_traveler_update_cost.

Logical Architecture Reference Flow: traveler_personal_current_condition_request

This data flow is used within the Provide Driver and Traveler Services function and contains a request for details of the current conditions, e.g. weather, events, incidents, etc. The request includes the identity of the personal device from which the request was input by the traveler so that the response can be correctly returned. The data flow consists of the following item which is defined in its own DDE: traveler_identity.

Logical Architecture Reference Flow: traveler_personal_display_update_payment_request

This data flow is sent from the Provide Driver and Traveler Services function to the Provide Electronic Payment Services function and contains a request that payment be made for an update of the digitized map data used as background to the displays of traffic and travel information on a traveler's personal device. The payment will be made by debiting the credit identity with the cost through the financial institution terminator. It consists of the following data items each of which is defined in its own DDE: traveler_identity + credit_identity + display_map_traveler_update_cost.

Physical Architecture Flow Name: trip confirmation

Acknowledgement by the driver/traveler of acceptance of a route.

Logical Architecture Reference Flow: traveler_personal_payment_information

This data flow is used within the Provide Driver and Traveler Services function. It contains details of the components of a trip which a traveler has obtained from the input of data to a personal device and for which advanced payment is needed following a trip confirmation. The traveler's identity and credit identity or stored credit from the payment instrument are therefore also included to enable payment to be made. The data flow consists of the following items each of which is defined in its own DDE: credit_identity + parking_space_details + ride_segments + stored_credit + toll_route_segments + traveler_identity.

Logical Architecture Reference Flow: traveler_personal_trip_confirmation

This data flow is used within the Provide Driver and Traveler Services function to confirm the trip details provided as the result of a traveler's previous trip request input from a personal device. It contains the following data item which is defined in its own DDE: paratransit_service_confirmation + traveler_identity + traveler_rideshare_confirmation.

Logical Architecture Reference Flow: traveler_route_accepted

This data flow is used within the Provide Driver and Traveler Services function and contains the acceptance by the traveler of the previously provided route for on-line infrastructure based guidance. Acceptance must be provided before guidance can begin. The data flow consists of the following data item which is defined in its own DDE: route_identity.

Physical Architecture Flow Name: trip request

Request by a driver/traveler for special routing.

Logical Architecture Reference Flow: traveler_personal_trip_request

This data flow is used within the Provide Driver and Traveler Services function and contains data about a traveler's trip request which has been input from a personal device. It consists of the following data items each of which is defined in its own DDE: trip_request + traveler_identity + traveler_rideshare_request.

Logical Architecture Reference Flow: traveler_route_request

This data flow is used within the Provide Driver and Traveler Services function and contains data from which the route requested by a traveler can be determined, or that the previously provided data has been accepted. It consists of the following data items each of which is defined in its own DDE: origin + destination + desired_arrival_time + modes + preferred_routes + preferred_alternate_routes + preferred_ridesharing_options + preferred_route_segments + preferred_transit_options + constraint_on_acceptable_travel_time + constraint_on_number_of_mode_changes + constraint_on_number_of_transfers + constraint_on_eta_change + constraint_on_special_needs + traveler_route_accepted + traveler_identity + traveler_location.

Physical Architecture Flow Name: yellow pages request

Request for information through a yellow pages type service.

Logical Architecture Reference Flow: traveler_personal_transaction_request

This data flow is used within the Provide Driver and Traveler Services function and contains data input by the traveler at a personal device to make reservations for various other (yellow pages) services. It contains the following data items, each of which is defined in its own DDE: yellow_pages_dining_reservation + yellow_pages_lodging_reservation + yellow_pages_ticket_purchase.

Logical Architecture Reference Flow: traveler_personal_yellow_pages_information_request

This data flow is used within the Provide Driver and Traveler Services function and contains a request for data on other (yellow pages) services to be provided to a traveler using a personal device. The traveler identity is used as the means of ensuring that the data produced in response to the request is returned to the correct traveler. As no filtering components are included, all the data currently available will be provided. The data flow consists of the following data item which is defined in its own DDE: traveler_identity.

Personal Information Access => **Map Update Provider**

Physical Architecture Flow Name: map update request

Request for a map update which could include a new underlying map or map layer updates.

Logical Architecture Reference Flow: tmup_request_traveler_personal_display_update
This data flow is sent to the map update provider from the Provide Driver and Traveler Services function. It contains a request for an update to the digitized map data used for displays that can be output as background for traffic, trip and travel information for use by a traveler at a personal device. It consists of the following data items each of which is defined in its own DDE. The map update provider is expected to use them as a means of obtaining payment for providing the map data, but not for %034tuning%034 the data so that it will be relevant to the area in which the kiosk is located since there is no knowledge of where the device may be used. credit_identity + traveler_identity.

Logical Architecture Reference Flow: tmup_request_traveler_personal_display_update_cost
This data flow is sent to the map update provider from the Provide Driver and Traveler Services function and contains a request for the cost of an update to the digitized map data used for providing the background to displays of traffic and travel information on a traveler personal device.

Logical Architecture Reference Flow: tmup_traveler_map_update_cost_request
This data flow is sent to the map update provider from the Provide Driver and Traveler Services function and contains a request for the cost of an update to the navigable map database used for providing traveler personal on-line guidance.

Logical Architecture Reference Flow: tmup_traveler_map_update_request
This data flow is sent to the map update provider from the Provide Driver and Traveler Services function and contains a request for an update of the digitized map database used for guiding travelers on their selected routes.

Personal Information Access => **Payment Instrument**

Physical Architecture Flow Name: request for payment

Request to deduct cost of service from user's payment account.

Logical Architecture Reference Flow: tpi_debited_payment_at_personal_device
This data flow is sent to the payment instrument from the Provide Electronic Payment Services function. It is a request to deduct either the cost of the update to the navigable map database used by the traveler for on-line personal guidance, or the cost of a traveler's confirmed trip from the value of credit currently stored by the payment instrument.

Personal Information Access => **Transit Management**

Physical Architecture Flow Name: transit information user request

Request for special transit routing, real-time schedule information, and availability information.

Logical Architecture Reference Flow: transit_services_personal_request
This data flow is a request for supply of details of transit services for output to a traveler's personal device. The traveler will have to provide the origin and destination so that the receiving process can work out for which transit route(s) data will be provided. The data flow consists of the following data items each of which is defined in its own DDE: destination + origin + traveler_identity.

Personal Information Access => **Traveler**

Physical Architecture Flow Name: traveler interface updates

Visual or audio information (e.g., routes, messages, guidance) to the traveler.

Logical Architecture Reference Flow: tt_emergency_message
This data flow is used as part of the interface to the traveler by the Provide Driver and Traveler Services function. It contains the acknowledgment of a request for action by the Emergency Services previously submitted by the traveler.

Logical Architecture Reference Flow: tt_guidance
This data flow is used as part of the interface to the traveler by the Provide Driver and Traveler Services function. It contains output (displays - text and/or graphics, and/or audio based information) which gives the traveler instructions on how to follow the route, e.g. cross the road, take a particular transit service, use the lift for wheelchairs, etc.

Logical Architecture Reference Flow: tt_guidance_input_request
This data flow is sent to the traveler by the Provide Driver and Traveler Services function. It contains a request for the traveler to input a specific item of data needed to determine the best route for on-line guidance. The data may comprise such things as the destination, preferred arrival time, plus route choice preferences and constraints. The latter will include limitations on the choice of mode(s) for all or part of the route.

Logical Architecture Reference Flow: tt_guidance_map_update_response
This data flow is sent to the traveler by the Provide Driver and Traveler Services function and contains the response to a previous request from the driver for an update to the digitized map data used to provide on-line traveler guidance.

Logical Architecture Reference Flow: tt_guidance_route_details
This data flow is sent to the traveler from the Provide Driver and Traveler Services function and contains details of the route that has been selected in response to the traveler's request for on-line guidance. The route and choice of guidance method will have been based on previous input from the traveler. Guidance will not begin until the traveler has positively accepted this data.

Logical Architecture Reference Flow: tt_personal_extra_trip_data_request
This data flow is used as part of the interface to the traveler by the Provide Driver and Traveler Services function. It contains outputs about the trip that the traveler has previously requested from a personal device, or messages about the previous confirmation of this trip.

Logical Architecture Reference Flow: tt_personal_trip_planning_responses
This data flow is used as part of the interface to the traveler by the Provide Driver and Traveler Services function. It contains the result of requests for more data about the trip on which the traveler is requesting information from a personal device.

Transit Management => **Personal Information Access**

Physical Architecture Flow Name: personal transit information

General and personalized transit information for a particular fixed route, flexible route, or paratransit system.

Logical Architecture Reference Flow: transit_services_for_personal_devices
This data flow contains details of the current transit services for output to a traveler's personal device and consists of the following data items each of which is defined in its own DDE: traveler_identity + 1{transit_services_for_output}2.

Traveler => **Personal Information Access**

Physical Architecture Flow Name: traveler request

Request by a traveler to summon assistance, request information, make a reservation, or initiate any other traveler service.

Logical Architecture Reference Flow: ft_guidance_data
This data flow is sent from the traveler to the Provide Driver and Traveler Services function and contains data to be used in preparing the request for a traveler route.

Logical Architecture Reference Flow: ft_guidance_map_update_request
This data flow is sent from the traveler to the Provide Driver and Traveler Services function and contains a request for an update of the digitized map data used to provide on-line traveler guidance.

Logical Architecture Reference Flow: ft_guidance_request
This data flow is sent from the traveler to the Provide Driver and Traveler Services function. It contains the a request to provide on-line traveler guidance and specifies a choice of the type of prefered guidance, i.e. infrastructure based dynamic, or totally autonomous. The traveler will be prompted for further data in order that the guidance can begin. .

Logical Architecture Reference Flow: ft_guidance_route_accepted
This data flow is sent from the driver to the Provide Driver and Traveler Services function. It contains acceptance of the route that has been generated in response to a previous request from the traveler for on-line guidance. Guidance will not begin until the acceptance has been received.

Logical Architecture Reference Flow: ft_personal_emergency_request
This data flow is sent from the traveler to the Provide Driver and Traveler Services function. It contains analog data from which sensors can determine that a traveler has an emergency situation that has given rise to an input from the traveler's personal device. This data must be forwarded to the Emergency Services function.

Logical Architecture Reference Flow: ft_personal_extra_trip_data
This data flow is sent from the traveler to the Provide Driver and Traveler Services function and contains analog data from which sensors can determine extra trip request data to supplement that already input by a traveler using a personal device. Examples of these inputs are speech, signals from relays driven by switches, buttons, etc., or input from touch screens.

Logical Architecture Reference Flow: ft_personal_map_display_update_request
This data flow is sent from the traveler to the Provide Driver and Traveler Services function and contains analog input from which may be determined a traveler's request for the update of the digitized map data used as the background to traffic, trip and travel information displays in a personal device. Examples of these inputs are speech, signals from relays driven by switches, buttons, etc., or input from touch screens.

Logical Architecture Reference Flow: ft_personal_trip_planning_requests
This data flow is sent from the traveler to the Provide Driver and Traveler Services function and contains analog inputs from which a traveler's trip request to a personal device may be determined. Examples of these inputs are speech, signals from relays driven by switches, buttons, etc., or input from touch screens.

2.12.4 Subsystem Architecture Flow Diagram

Personal Information Access (PIAS)

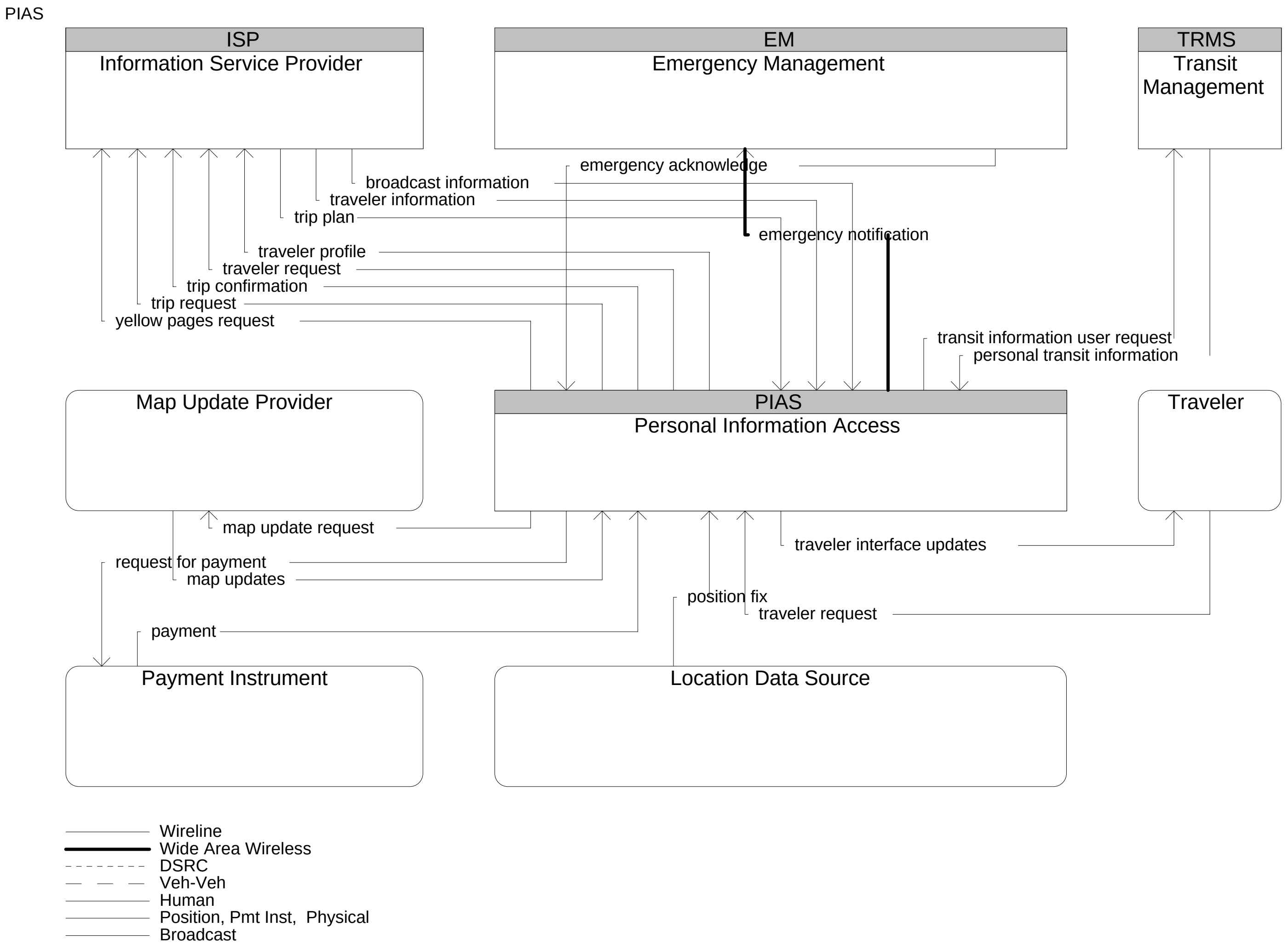


Figure 2.12-2 Architecture Flow Diagram for Personal Access

2.13 Planning Subsystem

The Planning Subsystem provides a data archiving and analysis function for the National ITS Architecture. It collects historical, current, and predicted transportation information from the other center subsystems. The collected information is used in analysis and evaluation of current transportation system performance and in planning for future transportation improvements. The broad data interfaces supported by this subsystem make transportation data available to researchers and planners to facilitate the deployment and operation of ITS services.

2.13.1 Alternative Configurations

The planning subsystem represents the planing agencies who may obtain historical or operational data from all subsystems Figure 2.13-1.

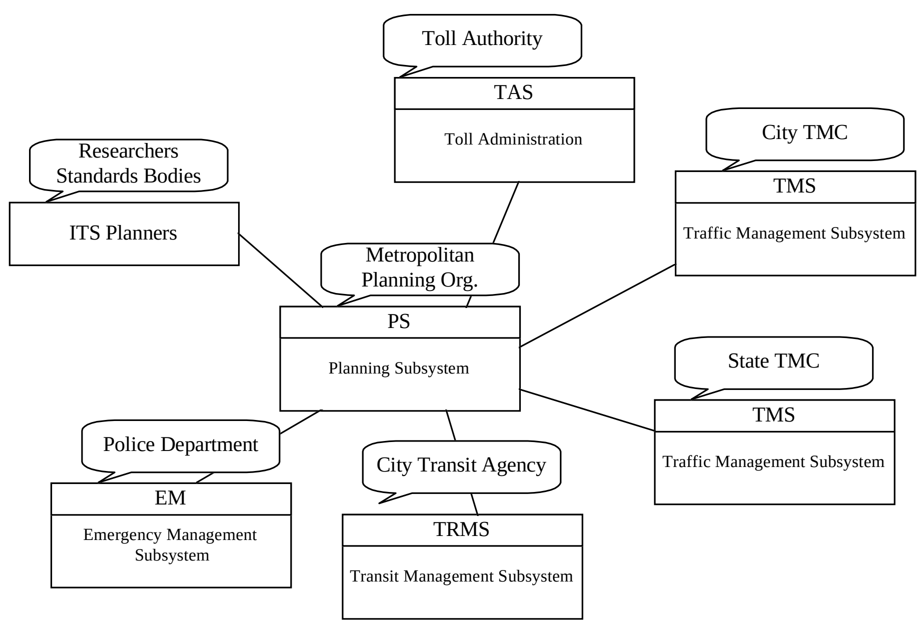


Figure 2.13-1 Configuration of Planning Subsystem

2.13.2 Subsystem Equipment Packages and Supporting Process Specifications for PS

Data Collection and ITS Planning

This service collects data from all center functions in support of ITS planning activities

Process Specifications

7.4.1.7 Collect Payment Transaction Records

Overview: This process shall be responsible for the collection and maintenance of a data store that contains transaction records for payments made for various services provided. The process shall load information into the payment_transaction_records data store for services comprising electronic credential and tax filing purchases for commercial vehicle operations, updates of map databases for drivers and travelers, registration of other (yellow pages) service providers (so that information about what they have to offer is available to travelers and transit users), advanced payment of tolls, parking lot charges, transit fares and other (yellow pages) services that form part of travelers' trips. The data shall be stored by the process with all references to the identity of the payment source, i.e. driver, traveler, commercial vehicle fleet manager, and any other payment information removed.

8.1 Import Static and Historical Data

Overview: This process shall be responsible for collecting static and historical data from other functions within ITS, as well as digitized map data from a specialist supplier. The process shall collect some of the data on request as a result of input received from transportation planners via an interface process, and the rest of the data in the form of regular unsolicited inputs. When received, this process shall send the data to another process for loading into stores that are used by other processes in the function as a source of data for simulation and evaluation of the ITS network.

8.2.1 Update Data Stores

Overview: This process shall be responsible for updating the four data stores used by processes in the Plan System Deployment function. The data for loading into the stores shall be provided by the data import process.

8.2.2 Communicate with Transportation Planner

Overview: This process shall be responsible for acting as the interface between the transportation planner and other processes in this function. It shall receive transportation planner inputs via an interface process, and shall generate the required outputs, or send data to other processes in response to the inputs. Data for output shall be sent to the to the transportation planner interface process.

8.2.3 Generate Static Data

Overview: This process shall be responsible for providing the facility to generate a new set of static data for use by the Manage Traffic function. The process shall produce a new set of static data on receipt of a set of network generation parameters from the transportation planner, via the interface processes. It shall use the current set of network static data and historical data to produce the new static data.

8.2.4 Evaluate System

Overview: This process shall be responsible for the evaluation of the performance of the network of roads and freeways served by the local ITS function. This evaluation shall take into account any factors that may affect performance such as, weather, travel demand, the desire to reduce emissions, transit services, proportion of guided and non-guided vehicles, historic performance, etc. From the results produced by the process it shall be possible to deduce the effects that any changes to the network have had on its performance in terms of its ability to move people and goods and the efficiency with which this is achieved. The process shall be driven by the input of evaluation parameters received from the transportation planner via the interface processes. The same processes shall be supplied with the results of the evaluation.

8.2.5 Simulate System

Overview: This process shall be responsible for providing a simulation of the way in which the road and freeway network served by the local ITS functions will operate in terms of the traffic volumes, delays, etc. experienced. The simulation performed by the process shall take into account factors which may affect system performance such as weather, travel demand, the desire to reduce emissions, transit services, proportion of guided and non-guided vehicles, historic performance, etc. The process shall be driven by the input of simulation parameters received from the transportation planner via the interface processes. The same processes shall be supplied with the results of the simulation.

8.2.6 Document System

Overview: This process shall be responsible for providing the transportation planner with the facility to create proper documentation of the network served by the ITS functions. The process shall provide output that is capable of being used as an aid in communicating to others how the ITS network will work and what improvements are proposed to improve the efficiency with which it moves people and goods and/or reduce the emissions which this movement generates. The process shall be driven by the input of a documentation request received from the transportation planner via the interface processes. The same processes shall be supplied with the output of the generated documentation in electronic form, which it shall be their responsibility to convert into the required output format.

8.3 Export Static Data

Overview: This process shall be responsible for sending a new data to other ITS functions. The process send a copy of the static data to processes in the Manage Traffic function, and a new copy of the database than maps link ownership to a process in the Provide Driver and Traveler Services function. This output shall only be produced by the process on receipt of a new set of static data from the process responsible for its generation.

8.4 Provide Transportation Planner Interface

Overview: This process shall be responsible for the interface between the transportation planner and the other processes in the Plan System Deployment function. It shall enable the transportation planner to control the processes in the function, input data for use in the simulation, evaluation and generation of improvements to the ITS road and freeway network, obtain the results of the simulation and evaluation processes, plus network documentation, and import and export data. The process shall support inputs from the transportation planner in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be either in hardcopy, or as a display.

8.5 Provide Map Update Provider Deployment Interface

Overview: This process shall be responsible for providing the interface through which the Plan System Deployment function can exchange data with the map update provider. This exchange of data shall enable the process to send data about the road and highway network to the supplier for inclusion in the digitized map data that is provided to other function within ITS. The supplier shall provide the process with digitized map data for use in the evaluation and simulation of the traffic conditions in the road and freeway network so that it can be continually updated to reflect the changing pattern of use.

2.13.3 Subsystem Interfaces for PS

Commercial Vehicle Administration => Planning Subsystem

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: cv_operational_data

This data is sent from the Manage Commercial Vehicle function to the Plan System Deployment function. It contains data about the number of commercial vehicles passing each roadside checking facility and how many are passing or failing their checks. The data is obtained from the roadside facility log, with details of vehicle identities is removed by the Manage Commercial Vehicles function before the data is sent.

Emergency Management => Planning Subsystem

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: emergency_vehicle_operational_data

This data flow is sent from the manage Emergency Services function to the Plan System Deployment function. It contains information about the movements of emergency vehicles whilst attending incidents, and therefore shows the usage of green wave routes, the times at which the vehicles passed various points in the road and highway network etc.

Emissions Management => Planning Subsystem

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: pollution_operational_data

This data flow is sent from the Manage Traffic function to the Plan System Deployment function. It contains data about the levels of pollution detected by roadside, wide area, and individual vehicle sensors located in the geographic area served by the function. The data will be that which has been collected since the last set of data was sent and will therefore have a time stamp record. The data flow consists of the following data items each of which is defined in its own DDE: date + list_size + list_size{pollution_state_roadside_collection + pollution_state_vehicle_collection} + pollution_state_area_collection + time.

Information Service Provider => Planning Subsystem

Physical Architecture Flow Name: aggregate travel data

Aggregated transportation infrastructure data and associated traveler transaction data for planning purposes.

Logical Architecture Reference Flow: current_other_routes_use

This data flow is used within the Provide Driver and Traveler Services function and contains data about the non-vehicle portion(s) of routes that have been requested by travelers. These route portions will involve the use of modes such as cycling, walking, etc. The data will be stored in ascending route segment number order (i.e. from 1 to the maximum number of route segments),r and consists of the following data items each of which is defined in its own DDE: route_segment_total_number + route_segment_total_number{route_segment_identity + time_period{route_segment_guided_travelers} + route_segment_journey_time}.

Logical Architecture Reference Flow: current_road_network_use

This data flow is sent from the Provide Driver and Traveler Services function to the Manage Traffic function. It contains information about how many vehicles are being guided down each route segment and the average journey time for each route segment provided by guided vehicles. The data will be stored in ascending route segment number order (i.e. from 1 to the maximum number of route segments), and consists of the following data items each of which is defined in its own DDE: route_segment_total_number + route_segment_total_number{route_segment_identity + route_segment_use_prediction + route_segment_journey_time}.

Logical Architecture Reference Flow: driver_map_update_payments_transactions

This data flow is used within the Provide Payment Electronic Services function and contains records of all payment transactions for driver in-vehicle guidance map updates.

Logical Architecture Reference Flow: traveler_map_update_payments_transactions

This data flow is used within the Provide Payment Electronic Services function and contains records of all payment transactions for driver map updates.

Logical Architecture Reference Flow: traveler_rideshare_payments_transactions
This data flow is used within the Provide Payment Electronic Services function and contains records of all payment transactions for traveler ridesharing provision.

Logical Architecture Reference Flow: traveler_trip_payments_transactions
This data flow is used within the Provide Payment Electronic Services function and contains records of all payment transactions for the provision of other (yellow pages) services and advance toll, parking lot charges, or transit fares as part of travelers' confirmed trips.

Logical Architecture Reference Flow: yellow_pages_provder_payments_transactions
This data flow is used within the Provide Payment Electronic Services function and contains records of all payment transactions for the provision of other (yellow pages) services and registration of suppliers of these services.

Map Update Provider => **Planning Subsystem**

Physical Architecture Flow Name: map updates

Map update which could include a new underlying static or real-time map or map layer(s) update.

Logical Architecture Reference Flow: fmup_deployment_map_update
This data flow is sent from the map update provider to the Plan System Deployment function and contains a new set of digitized map data. This will be used in the simulation and evaluation of the road and highway network served by the ITS, so that any changes to improve its efficiency can be determined.

Parking Management => **Planning Subsystem**

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: parking_lot_operational_data
This data flow is sent from the Manage Traffic function to the Provide System Deployment function. It contains counts of the number of spaces used in the parking lot and the parking lot state during the previous time period, i.e. since the data was last sent. Regardless of the actual value of the time period, the data is provided in five (5) minute steps, i.e. the number of spaces used and state are recorded every five minutes. The data flow consists of the following data items each of which is defined in its own DDE: date + list_size + list_size{parking_lot_current_state} + parking_lot_identity + time.

Planning Subsystem => **Map Update Provider**

Physical Architecture Flow Name: map update request

Request for a map update which could include a new underlying map or map layer updates.

Logical Architecture Reference Flow: tmup_deployment_map_update_request
This data flow is sent to the map update provider from the Plan System Deployment function and contains a request for an update of the digitized map data. This data is used in the simulation and evaluation of the road and highway network served by the ITS so that changes to improve its efficiency can be designed and implemented.

Logical Architecture Reference Flow: tmup_map_static_data
This data flow is sent to the map update provider from the Plan System Deployment function and contains a new set of static data or a set of transit route data. This will be used by the map provider in the preparation of a new set of digitized map data which can then be used by this and other functions within ITS.

Planning Subsystem => Traffic Management

Physical Architecture Flow Name: planning data

Data and commands from Transportation Planners.

Logical Architecture Reference Flow: link_data
This data flow is sent by the Plan System Deployment function to the Manage Traffic function. It contains a new version of the store of link data used to determine which other TMC to contact to obtain traffic data relating to another geographic area. The data flow consists of the following data items each of which is defined in its own DDE: link_attributes + link_identity + link_TMC_identity.

Logical Architecture Reference Flow: static_data_request
This data flow is sent from the Provide System Deployment function to the Manage Traffic function to request a copy of some or all of the current static data being used by the traffic control, sensor processing and incident management facilities. The data flow is sized at three bytes to enable the use of three alphanumeric characters, one for each type of data, i.e. 'i' for incident, 's' for sensor and 't' for traffic. The receiving process will check that the alphanumeric character for its data is present before retrieving and returning it to the Plan System Deployment function.

Logical Architecture Reference Flow: supply_incident_static_data
This data flow is sent by the Provide System Deployment function to the Manage Traffic function and includes new and/or amended static data for use in incident management. This data consists of details of the road network plus the location and relationship between links in the network. It therefore contains the contents of the following data store which is defined in its own DDE: static_data_for_incident_management.

Logical Architecture Reference Flow: supply_traffic_static_data
This data flow is sent by the Provide System Deployment function to the Manage Traffic function and includes new and/or amended static data for use in traffic control and sensor processing. This data consists of traffic signal data (timings, permitted phase changes, etc.), which items of traffic data are used by certain processes, VMS data, etc. It therefore contains the contents of the following data stores, each of which is defined in its own DDE: static_data_for_traffic_control + static_data_for_sensor_processing.

Logical Architecture Reference Flow: traffic_data_deployment_request
This data flow is used by the Plan System Deployment and Improvement function to request the Manage Traffic function to provide it with traffic data. It must contain a processor source identity so that the Manage Traffic function knows where to send the retrieved traffic data.

Planning Subsystem => Transportation Planners

Physical Architecture Flow Name: planning data

Data and commands from Transportation Planners.

Logical Architecture Reference Flow: ttp_evaluation_results
This data flow is sent to the transportation planner by the Plan System Deployment function. It contains the formatted output of the results from the evaluation process.

Logical Architecture Reference Flow: ttp_output_data_store

This data flow is sent to the transportation planner by the Plan System Deployment function. It contains the formatted output of either the historic, static or planning data stores.

Logical Architecture Reference Flow: ttp_output_documentation
This data flow is sent to the transportation planner by the Plan System Deployment function. It contains output formatted as a document covering some or all of the data stored for the ITS network in the historic, static or planning data stores.

Logical Architecture Reference Flow: ttp_output_link_data
This data flow is sent to the transportation planner by the Plan System Deployment function. It contains the formatted output of the contents of the store containing the link database.

Logical Architecture Reference Flow: ttp_simulation_data
This data flow is sent to the transportation planner by the Plan System Deployment function. It contains the formatted output of the results from the simulation process.

Toll Administration => **Planning Subsystem**

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: toll_operational_data
This data flow is sent from the Provide Electronic Payment Services function to the Plan System Deployment function. It contains data about the cost of toll segments and the number of users of those segments during the time period since the data was last sent. The data flow consists of the following data items each of which is defined in its own DDE: date + list_size + list_size{toll_cost + toll_segment_identity + toll_segment_users} + time.

Traffic Management => **Planning Subsystem**

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: ahs_operational_data
This data flow is sent from the Provide Vehicle Monitoring and Control function to the Plan System Deployment function. It contains details of the number of vehicles that have been checked into the automatic highway system (ahs), plus details about the use of ahs lanes during the previous time period, e.g. one (1) hour. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{ahs_checking_records + ahs_lane_use_data + date + time}.

Logical Architecture Reference Flow: current_incident_static_data
This data flow is sent from the Manage Traffic function to the Plan System Deployment function and contains of some or all of the current static data used for the management of incidents. It therefore contains the contents of the following data store, which is defined in its own DDE: static_data_for_incident_management.

Logical Architecture Reference Flow: current_traffic_static_data
This data flow is sent from the Manage Traffic function to the Plan System Deployment function and contains of some or all of the current static data consisting of signal data (timings, permitted phase changes, etc.), VMS data, which items of traffic data are used by certain processes, etc. It therefore contains the contents of the following data stores, each of which is defined in its own DDE: static_data_for_traffic_control + static_data_for_sensor_processing.

Logical Architecture Reference Flow: traffic_data_for_deployment

This data flow is sent from the Manage Traffic function to the Plan System Deployment and Implementation function. It is used to provide data on the traffic flowing in the road network, plus that which is predicted to flow in the network and consists the following items each of which is defined in its own DDE: long_term_data + predictive_model_data.

Transit Management => **Planning Subsystem**

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: transit_passenger_operational_data
This data flow is sent from the Manage Transit function to the Plan System Deployment function. It contains information about the number of passengers (transit users) who have used transit stops and vehicles being operated by the Manage transit function. There are therefore two sets of data, one showing the numbers of passengers using each transit stop and the other showing the number of passengers on-board transit vehicles on each route segment. The data is recorded as the average values over short time intervals, e.g. five (5) minutes, for a whole day. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{date + time + transit_roadside_operational_data + transit_route_operational_data}.

Logical Architecture Reference Flow: transit_services_for_deployment
This data flow is sent from the Manage Transit function to the Plan System Deployment function. It contains details of the current transit services for use in the analysis of ITS operating performance and consists of the following data item which is defined in its own DDE: transit_services.

Logical Architecture Reference Flow: transit_user_payments_transactions
This data flow is used within the Provide Electronic Payment Services function and contains records of all payment transactions for the provision of other (yellow pages) services to transit users.

Transportation Planners => **Planning Subsystem**

Physical Architecture Flow Name: planning data

Data and commands from Transportation Planners.

Logical Architecture Reference Flow: ftp_evaluation_request
This data flow is sent from the transportation planner to the Plan System Deployment function contains a request to start the evaluation process plus (optionally) additional parameters.

Logical Architecture Reference Flow: ftp_export_request
This data flow is sent from the transportation planner to the Plan System Deployment function and specifies the type of data to be exported to other functions within ITS.

Logical Architecture Reference Flow: ftp_generation_request
This data flow is sent from the transportation planner to the Plan System Deployment function and contains parameters used to create a new set of static data for use by other functions within ITS.

Logical Architecture Reference Flow: ftp_import_request
This data flow is sent from the transportation planner to the Plan System Deployment function and contains details of data to be imported from other functions within ITS. This will only cover data that is not automatically sent to the function by other functions within ITS.

Logical Architecture Reference Flow: ftp_output_request
This data flow is sent from the transportation planner to the Plan System Deployment function and contains details of

the part(s) of the historic, static, or planning data stores are to be output.

Logical Architecture Reference Flow: ftp_parameters
This data flow is sent from the transportation planner to the Plan System Deployment function and contains parameters use by the simulation, evaluation and static data generation processes within the function.

Logical Architecture Reference Flow: ftp_request_documentation
This data flow is sent from the transportation planner to the Plan System Deployment function and contains details of the planner's choice of the planning data or simulation data that is to be documented.

Logical Architecture Reference Flow: ftp_request_link_data
This data flow is sent from the transportation planner to the Plan System Deployment function and contains a request for output of the store containing the link database.

Logical Architecture Reference Flow: ftp_simulation_request
This data flow is sent from the transportation planner to the Plan System Deployment function and contains a request plus (optionally) additional parameters needed to start the simulation process.

Logical Architecture Reference Flow: ftp_static_data
This data flow is sent from the transportation planner to the Plan System Deployment function and contains items of new or amended data for the static data store which the transportation planner wishes to add to that store.

Logical Architecture Reference Flow: ftp_traffic_data_request
This data flow is sent from the transportation planner to the Plan System Deployment function and contains a request for a new set of traffic data to be obtained from the Manage Traffic function.

Logical Architecture Reference Flow: ftp_update_link_data
This data flow is sent from the transportation planner to the Plan System Deployment function and contains updates to the contents of the store containing the link database.

2.13.4 Subsystem Architecture Flow Diagram

Planning Subsystem (PS)

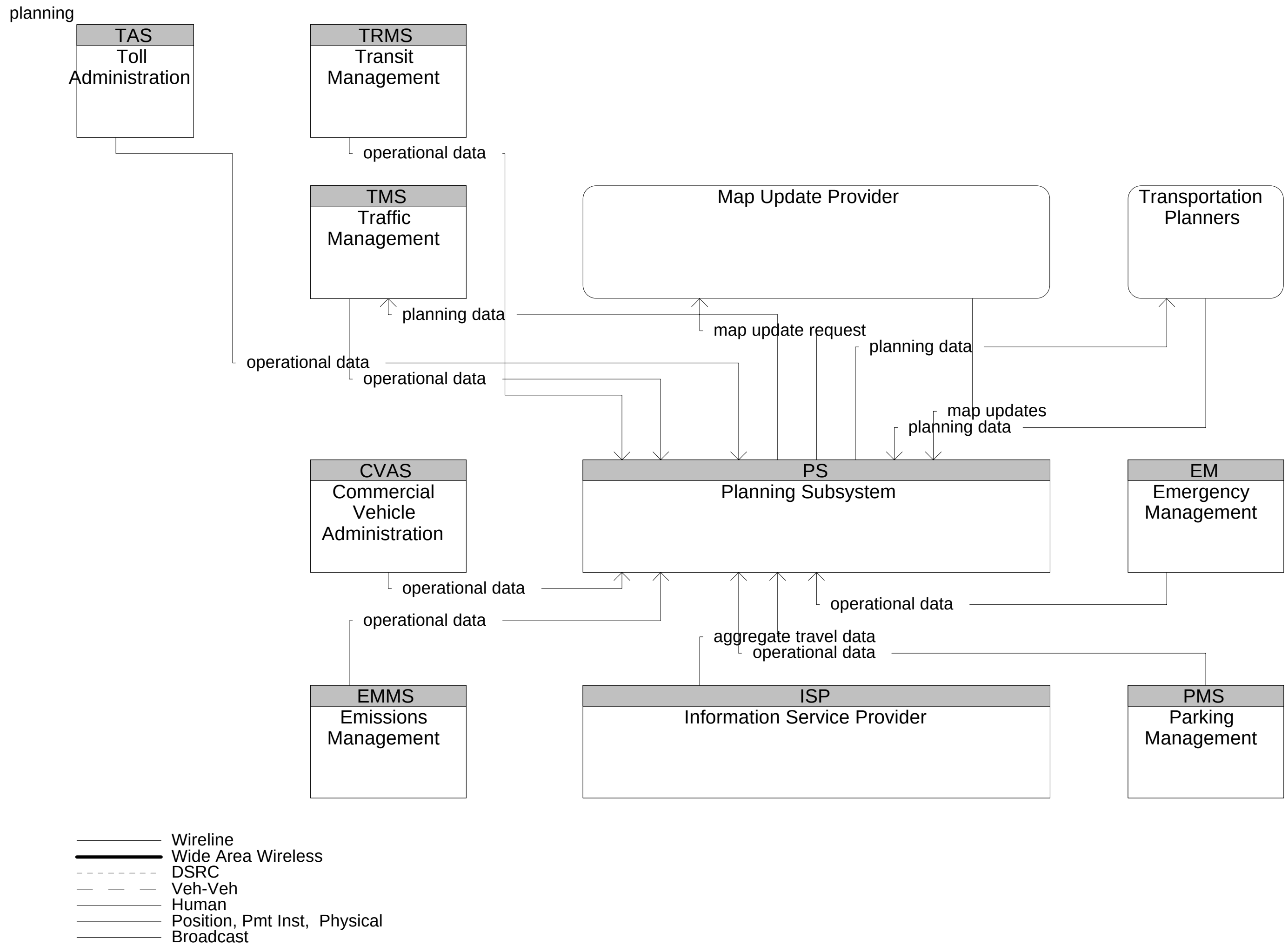


Figure 2.13-2 Architecture Flow Diagram for Planning

2.14 Parking Management

The Parking Subsystem provides the capability to provide parking availability and parking fee information, allow for parking payment without the use of cash with a multiple use medium, and support the detection, classification, and control of vehicles seeking parking.

2.14.1 Alternative Configurations

Parking management could represent any parking lot or collection of lots. They could be affiliated with intermodal providers or simply independent organizations Figure 2.14-1. Parking lots can make their occupancy and predicted occupancy available to ISP's so that travelers are aware of parking options. Parking lots can also exchange occupancy and price information with traffic management to implement some future demand management policies.

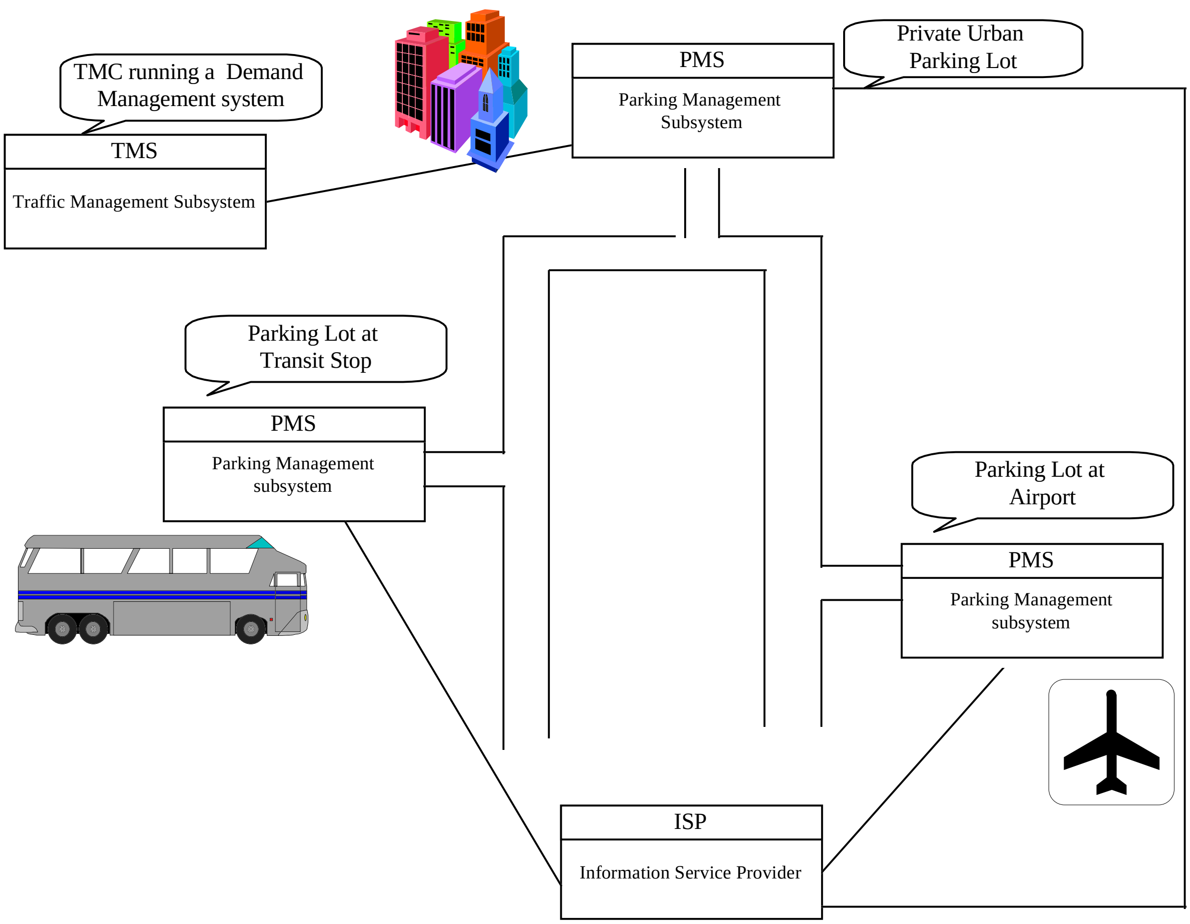


Figure 2.14-1 Alternative Configurations for Parking

2.14.2 Subsystem Equipment Packages and Supporting Process Specifications for PMS

Parking Management

This Equipment package provides the capability to detect and classify properly equipped vehicles entering and exiting the parking facility, and to maintain database information with parking availability and pricing structure information. This capability shall be provided through the utilization of active/passive tag readers and database software containing parking pricing structure and current availability. Wireline communications with clearinghouse operators (the Financial Institution terminator) and the back office (the parking service provider terminator) enable processing of financial transactions and external coordination.

Process Specifications

1.2.5.1 Determine Parking Lot State

Overview: This process shall implement the selected control strategies on some or all of the parking lots in the road and freeway network served by the Manage Traffic function. It shall use the current parking lot occupancy provided by another process to determine the parking lot state to be used in sign settings implemented by other processes in the function, when this is not subject to a strategy override. The parking lot state shall be determined using threshold occupancy values contained in the static data provided by the Plan System Deployment function. Fixed thresholds for the states:'full','almost full' and 'available' are in the data flow'static_data_for_parking_lots'. In addition, threshold transitions can depend on whether the actual occupancy of the lot is increasing or decreasing, allowing hysteresis in the parking lot state transitions, so as to control jitter between parking lot states. The process shall also provide operational data to the Plan System Deployment function and the current parking lot occupancy to other processes in the function.

1.2.5.2 Provide Parking Service Provider Interface

Overview: This process shall provide the interface to a local parking lot System (i.e. parking service provider) that controls the use of the lot. The parking lot system shall provide inputs of occupancy and/or the current lot state. It shall respond to a request from this process to change the lot state by activating local variable message signs (vms) and controlling the use of entry/exit barriers, and to output through local vms, information about transit services that provide a park and ride (P+R) operation.

1.2.5.3 Provide Parking Lot Operator Interface

Overview: This process shall provide the interface to a local parking lot operator that controls the use of the lot. The operator shall provide inputs of occupancy and/or the current lot state to this process. This process shall provide the operator with outputs that request a change to the lot state, which the operator shall implement by activating local variable message signs (vms) and controlling the use of entry/exit barriers, and data about transit services that provide a park and ride (P+R) operation to be output through local vms.

1.2.5.4 Determine P+R needs for Transit Management

Overview: This process shall be responsible for calculating the need for transit services to provide a park and ride (P+R) operation at a parking lot. This calculation shall be based on the rate of change of the current parking lot occupancy. The results of the calculation shall be sent to the Manage Transit function in the form of a request for an additional (or reduced) level of service, depending on demand at the parking lot. The results of the request shall also be passed to other processes within the function.

1.2.5.5 Calculate Parking Lot Occupancy

Overview: This process shall calculate the occupancy of a parking lot based on processed traffic sensor data provided by other processes within the Manage Traffic function. The process shall use the static data for parking lots to determine the part(s) of the supplied data that apply to its entry and exit lanes, so that the numbers of vehicles entering and leaving can be calculated. These calculated flows shall be used by the process to generate the current parking lot occupancy.

5.4.3 Process Parking Lot Violations

Overview: This process shall manage the details of parking lot payment violations reported by the Provide Electronic Payment Services function. The process shall use the parameters in the store of parking lot violation (enforcement) data to obtain the vehicle registration data from the appropriate State Department of Motor

Vehicles (DMV) office (or alternate source) for vehicles that are not equipped with a tag, before sending all of the received information to the correct law enforcement agency. This process shall also maintain the store of parking lot violation (enforcement) data, entering all information received from other processes.

7.2.1.1 Read Parking Lot Tag Data

Overview: This process shall be responsible for requesting the data from the parking lot tag being carried on-board the vehicle and used as the payment instrument be read. If there is no tag or the data it contains cannot be properly read, the process shall send a message for the vehicle to pull in for output by another process, and send a request to other processes to obtain an image of the vehicle. If there is no entry time data on the tag, then the process shall re-write this data plus the number of the entry lane onto the tag and so that it can be used as the mechanism for charging for the use of the parking lot. If the entry time is present, the process shall combine it with the vehicle characteristics, e.g. size, type, etc. to form the data upon which the parking lot payment transaction can be based, and send it to another process.

7.2.1.10 Determine Advanced Charges

Overview: This process shall be responsible for receiving a request to pay an advanced parking lot charge. It shall obtain the required parking lot charge from a data store and shall then forward the data to the billing processes. The store of parking lot charges shall be maintained by another process.

7.2.1.2 Calculate Vehicle Parking Lot Charges

Overview: This process shall be responsible for calculating the parking lot charge for the detected vehicle based on its characteristics and data obtained from the tag being carried by the vehicle. The process shall obtain the cost of the use of the parking lot by reading data from a store that contains the standard prices for parking lot charges.

7.2.1.3 Collect Bad Charge Payment Data

Overview: This process shall be responsible for maintaining a data store containing a list of invalid driver credit identities. The process shall use this data to check credit identities provided for checking by the billing process. This checking shall ensure that the current parking lot payment transaction is using a credit identity that has not previously had an invalid transaction. Details of possible invalid credit identities shall be sent by the process to the financial institution for verification. The process shall also receive from the financial institution details of invalid payment instrument data that has been found by other means.

7.2.1.4 Check for Advanced Parking Lot Payment

Overview: This process shall be responsible for checking to see if the required parking lot charge payment has already been made. The process shall determine the existence of an advance payment for the parking lot charges by comparing the received payment information with that in the store containing the list of advanced payments. If the payment has already been made then the process shall remove the requirement for local billing and remove the record of the advanced payment from the store. Details of each payment transaction shall be sent by the process to another process with the payment information received from the driver removed.

7.2.1.5 Bill Driver for Parking Lot Charges

Overview: This process shall be responsible for either obtaining payment for the current or advanced parking lot charge. The process shall achieve this either by requesting that the charge be deducted from the credit being stored by the parking lot tag that is acting as the payment instrument, or informing the driver that payment for the charge will be debited to the credit identity provided by the tag. Before sending data to the tag, the process shall check that either the credit identity is not already in the last of bad payers, or the stored credit is not less that the parking lot charge. If either of these conditions is true the process shall obtain an image of the driver and vehicle which can be forwarded to the appropriate enforcement agency via another process. When the appropriate payment transaction has been completed, the parking lot entry time data shall be cleared from the tag so that it can be used for the next visit by the vehicle to a parking lot. The tag may be in the form of some type of credit or debit card, or an electronic purse. Details of the transaction shall always be sent to the process that manages parking lot transactions which will also send details to the financial institution if a credit or debit card is involved. Where an advanced parking lot charge payment is identified, no action is taken if the credit identity is on the bad payers list, or the stored credit is less than the charge, other than the payment is not confirmed.

7.2.1.6 Manage Parking Lot Financial Processing

Overview: This process shall be responsible for maintaining a log of all transactions that are carried out by other processes in the Process Electronic Parking Lot Payment facility. The identity of the payee shall have been removed from the data before it is stored. At periodic intervals the process shall output the accumulated records to another process in the Provide Electronic Payment Services function. It shall also output the same data on request to the parking service provider, either in hardcopy form, or as a visual display. Both audible and manual inputs from the parking service provider shall be supported by the process. The process shall also be responsible for sending details of transactions to the financial institution to enable the users to be billed through their credit identities.

7.2.1.7 Update Parking Lot Data

Overview: This process shall be responsible for maintaining a store of data containing the parking lot charges, which may vary according to the type of vehicle. The process shall also act as the interface to the parking service provider to enable changes to be made to the stored data, for the output and input of responses to parking lot price change requests from the Manage Traffic function, and for requests for parking lot price data from the Centralized Payments facility. The process shall support inputs from the parking service provider in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be either in hardcopy, or as a display.

7.2.1.8 Register for Advanced Parking Lot Payment

Overview: This process shall be responsible for responding to requests for parking lot charges to be paid in advance. It shall provide the parking service provider with the opportunity to deny the request for advanced payment of a parking lot charge. If approved, the advanced parking lot charge data shall be forwarded by the process to other processes for the actual cost to be obtained and the payment transactions initiated.

7.2.1.9 Manage Parking Lot Reservations

Overview: This process shall be responsible for maintaining a store of parking lot data. This data shall cover the capacity of the parking lot, i.e. the maximum number of spaces available, which may vary according to the type of vehicle. The process shall also act as the interface for inquiries from other ITS functions both for details of parking lot capacity, both now and in the future and for the reservation of spaces as part of travelers' confirmed trips.

7.2.2 Produce Parking Lot Displays

Overview: This process shall be responsible for driving the displays that tell vehicles whether or not their parking lot charge payment has been confirmed or rejected. The process shall receive the data for output via the displays from other processes. The process shall provide its outputs in audible and visual forms, with the latter using an appropriate form of display that shall be easily readable under all lighting conditions and over the range of speeds that vehicles are expected to use when entering or leaving a parking lot.

7.2.3 Obtain Parking Lot Violator Image

Overview: This process shall be responsible for obtaining an image of a violator for use by other processes. The form which the image data is obtained by this process shall be so accurate that there can be no mistake of the determination of the identity of the vehicle and/or driver, and shall be easily passed on by the other processes to the appropriate law enforcement agency(ies) so that any punitive action that may be taken. The process shall be capable of obtaining an image of the required accuracy under all lighting conditions and over the range of speeds with which vehicles will enter or leave parking lots.

7.2.5 Detect Vehicle for Parking Lot Payment

Overview: This process shall be responsible for producing a vehicle's characteristics from data received by sensors located at or near the parking lot entry and exit lanes. The data shall be sent by the process to another process in a form suitable for use in calculating the parking lot charge for the vehicle. The process shall ensure that the data includes such things as vehicle size, type, identifiable features , etc.

2.14.3 Subsystem Interfaces for PMS

DMV => **Parking Management**

Physical Architecture Flow Name: registration

Registered owner of vehicle and associated vehicle information.

Logical Architecture Reference Flow: fdmv_parking_lot_violation_state_identity
This data flow is sent from the department of motor vehicles to the Manage Emergency Services function and contains the identity of the state that is supplying the requested vehicle registration data to enable a parking lot payment violation to be processed.

Logical Architecture Reference Flow: fdmv_parking_lot_violation_vehicle_registration
This data flow is sent from the department of motor vehicles to the Manage Emergency Services function and contains the requested vehicle registration data to enable a parking lot payment violation to be processed.

Financial Institution => **Parking Management**

Physical Architecture Flow Name: transaction status

Response to transaction request. Normally dealing with a request for payment.

Logical Architecture Reference Flow: ffi_bad_charges_payment_updates
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains data about a toll transaction that was either attempted but did not work, or for which the subsequent payment transaction by the Financial Institution failed. The data is to be used within the function for checking against future toll transaction data.

Logical Architecture Reference Flow: ffi_confirm_charges_payment
This data flow is sent from the Financial Institution to the Provide Electronic Payment Services function and is set to either zero (0) for a parking lot charge payment transaction not completed, or one (1) for a valid completion.

Information Service Provider => **Parking Management**

Physical Architecture Flow Name: parking lot data request

Request for parking lot occupancy, fares, and availability. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Logical Architecture Reference Flow: advanced_other_charges_request
This data flow is used within the Provide Electronic Payment Services function to request that a parking lot charge be paid for in advance by either a driver who is paying a toll or a traveler (as a transit user) who is paying a transit fare. It consists of the following data items each of which is defined in its own DDE: credit_identity + parking_lot_identity + parking_space_details + stored_credit + vehicle_identity.

Logical Architecture Reference Flow: advanced_traveler_charges_request
This data flow is used within the Provide Electronic Payment Services function to request that a parking lot charge be paid for in advance by a traveler who is planning a trip. It consists of the following data items each of which is defined in its own DDE: credit_identity + parking_space_details + stored_credit + vehicle_identity.

Logical Architecture Reference Flow: parking_lot_data_request
This data flow is sent from the Provide Driver and Traveler Services function to the Provide Electronic Payment Services function and contains a request for data about the number of spaces that are available in a particular parking lot at the specified data and time. This data is requested as part of the process of putting together a proposed trip in response to a traveler's trip request This data flow contains the following items each of which is defined in its own DDE: date + parking_lot_identity + time + traveler_identity.

Logical Architecture Reference Flow: parking_lot_price_data_request
This data flow is used within the Provide Electronic Payment Services function. It contains a request for the current parking lot price data to be provided from the store that is being used to calculate parking lot charges.

Physical Architecture Flow Name: parking reservations request

Reservation request for parking lot.

Logical Architecture Reference Flow: parking_lot_reservation_request
This data flow is sent from the Provide Driver and Traveler Services function to the Provide Electronic Payment Services function and contains a request for an advance reservation to be made at a parking lot. This reservation is the result of a traveler confirming a proposed trip. This data flow contains the following items each of which is defined in its own DDE: date + parking_lot_identity + time + traveler_identity.

Parking Management => DMV

Physical Architecture Flow Name: license request

Request supporting registration data based on license plate read during violation.

Logical Architecture Reference Flow: tdmv_parking_lot_violation_identity_code
This data flow is sent to the department of motor vehicles from the Manage Emergency Services function and contains the identity code of the ITS that is requesting the vehicle registration data so that a parking lot payment violation can be processed.

Logical Architecture Reference Flow: tdmv_parking_lot_violation_vehicle_license
This data flow is sent to the department of motor vehicles from the Manage Emergency Services function and contains the vehicle license for which the corresponding registration data is required so that a parking lot payment violation can be processed.

Parking Management => Driver

Physical Architecture Flow Name: transaction status

Response to transaction request. Normally dealing with a request for payment.

Logical Architecture Reference Flow: td_parking_lot_payment_confirmed
This data flow is sent to the driver from the Provide Driver and Traveler Services function to confirm that the parking lot payment transaction has been successfully completed.

Logical Architecture Reference Flow: td_parking_lot_payment_invalid

This data flow is sent to the driver from the Provide Driver and Traveler Services function to indicate that the parking lot payment transaction is invalid.

Parking Management => **Enforcement Agency**

Physical Architecture Flow Name: violation notification

Notification to enforcement agency of violation or regulations.

Logical Architecture Reference Flow: tea_parking_violation_data
This data flow is sent from the Manage Emergency Services function to the enforcement agency and contains information about parking lot charge payment violations that have been detected by the Provide Electronic Payment Services function. The data in the flow will enable the notified enforcement agency to take the appropriate action against those committing the violation.

Parking Management => **Financial Institution**

Physical Architecture Flow Name: payment request

Request for payment from financial institution.

Logical Architecture Reference Flow: tfi_parking_lot_payment_violator_data
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains data about a toll payment transaction that was attempted but did not work and is to be used by the Financial Institution.

Logical Architecture Reference Flow: tfi_request_charges_payment
This data flow is sent to the Financial Institution by the Provide Electronic Payment Services function and requests payment of a parking lot charge. It is sent periodically, e.g. once per day, and requests payment of the parking lot charge transactions since the previous request. The data flow will include the parking lot charge and credit identity for each transaction.

Parking Management => **Information Service Provider**

Physical Architecture Flow Name: parking availability

Parking lot occupancy, availability and payment information.

Logical Architecture Reference Flow: parking_lot_availability
This data flow is sent from the Provide Electronic Payment Services function to the Provide Driver and Traveler Services function. It contains details of the number of spaces available in the lot in response to a previous request for this data. The data flow consists of the following items each of which is defined in its own DDE: parking_lot_identity + parking_lot_spaces + traveler_identity.

Physical Architecture Flow Name: parking lot reservation confirmation

Confirmation for parking lot reservation.

Logical Architecture Reference Flow: advanced_other_charges_confirm
This data flow is used within the Provide Electronic Payment Services function and shows whether or not an advanced parking lot payment transaction has been confirmed or not. It consists of the following data item which is defined in its own DDE: confirmation_flag + credit_identity + stored_credit + parking_lot_cost + vehicle_identity.

Logical Architecture Reference Flow: advanced_traveler_charges_confirm
This data flow is used within the Provide Electronic Payment Services function It contains data about an advanced parking lot charge transaction requested by a traveler and consists of the following data items each of which is defined in its own DDE: confirmation_flag + parking_lot_cost + stored_credit + traveler_identity.

Logical Architecture Reference Flow: parking_lot_price_data
This data flow is used within the Provide Electronic Payment Services function and contains the prices being charged by each parking lot for each of its spaces, together with the time and date for which they apply. It consists of the following data items each of which is defined in its own DDE: parking_lot_identity + parking_lot_price + parking_lot_charge_application_time + vehicle_type_for_charges.

Logical Architecture Reference Flow: parking_lot_reservation_confirm
This data flow is sent from the Provide Electronic Payment Services function to the Provide Driver and Traveler Services function. It contains the confirmation that a previously requested reservation of a space at a parking lot has been confirmed and can be included in a traveler's confirmed trip plan. This data flow consists of the following items each of which is defined in its own DDE: parking_lot_identity + reservation_status + traveler_identity.

Parking Management => **Parking Operator**

Physical Architecture Flow Name: parking status

Parking lot operational status.

Logical Architecture Reference Flow: tpo_change_lot_state
This data flow is sent to a parking lot operator by the Manage Traffic function and is a request for the operator to change the apparent parking lot state. This is the state that is apparent to users and may be closed, open, or in some cases almost full, although this may not be the true state according to its occupancy.

Parking Management => **Parking Service Provider**

Physical Architecture Flow Name: parking availability

Parking lot occupancy, availability and payment information.

Logical Architecture Reference Flow: tpsp_change_lot_state
This data flow is sent to a parking lot service provider by the Manage Traffic function and is a request for the provider to change the apparent state of the parking lot. This is the state that is seen by users and may be 'CLOSED', 'OPEN', or in some cases 'ALMOST FULL', although this may not be the true state according to its current occupancy.

Logical Architecture Reference Flow: tpsp_parking_lot_charge_change_request
This data flow is sent from the Provide Electronic Payment Services function to the parking services provider. It contains data requesting a change to the current parking lot charging structure so that travelers my be encouraged to change the modal split in their journeys.

Logical Architecture Reference Flow: tpsp_request_advanced_parking_payment
This data flow is used by the Provide Electronic Payment Services function to send a request the parking service provider to enable a particular advanced parking lot payment and includes information on the payee's credit identity, the vehicle identity and the location of the required parking lot space.

Logical Architecture Reference Flow: tpsp_transaction_reports
This data flow is sent to the parking service provider from the Provide Electronic Payment Services function to the parking service provider and contains the output of the log of toll transaction reports.

Parking Management => **Planning Subsystem**

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: parking_lot_operational_data
This data flow is sent from the Manage Traffic function to the Provide System Deployment function. It contains counts of the number of spaces used in the parking lot and the parking lot state during the previous time period, i.e. since the data was last sent. Regardless of the actual value of the time period, the data is provided in five (5) minute steps, i.e. the number of spaces used and state are recorded every five minutes. The data flow consists of the following data items each of which is defined in its own DDE: date + list_size + list_size{parking_lot_current_state} + parking_lot_identity + time.

Parking Management => **Traffic Management**

Physical Architecture Flow Name: demand management response

Response to various demand management change requests indicating level of compliance with request.

Logical Architecture Reference Flow: parking_lot_charge_change_response
This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function. It contains the response to a previous request for the current parking lot charges to be changed to help produce a change in the current modal split of trips being undertaken by all types of travelers. If sent to one (1) the change was accepted and if set to zero (0), the change was rejected.

Logical Architecture Reference Flow: parking_lot_charge_direct_details
This data flow is contains the prices being charged by each parking lot for each of its spaces, together with the time and date for which they apply. parking_lot_identity + parking_lot_price + parking_lot_charge_application_time + vehicle_type_for_charges.

Physical Architecture Flow Name: parking availability

Parking lot occupancy, availability and payment information.

Logical Architecture Reference Flow: parking_guidance_for_vms
This data flow is used within the Manage Traffic function. It contains the variable message sign (vms) states that will be used to implement the desired traffic control strategy at the parking lots served by the function. This data may be used to guide vehicles towards those parking lots where spaces are currently available or to show which lots have been closed, i.e. are not currently in use. The data flow consists of the following data items each of which is defined in its own DDE: indicator_list + 1{parking_lot_vms_controls}list_size.

Logical Architecture Reference Flow: parking_lot_current_state
This data flow is used within the Manage Traffic function and contains the identity of the parking lot plus its current status and occupancy. It consists of the following data items each of which is defined in its own DDE: parking_lot_identity + parking_lot_state + parking_lot_current_occupancy.

Parking Management => **Transit Management**

Physical Architecture Flow Name: transit parking coordination

Request for coordinated fare payment and parking lot price data.

Logical Architecture Reference Flow: parking_lot_transit_request
This data flow is sent by the Manage Traffic function to the Manage Transit function and contains a request for new or additional park and ride (P+R) transit services to be provided from the parking lot.

Parking Management => **Vehicle**

Physical Architecture Flow Name: request tag data

Request for tag information including credit identity, stored value card cash, etc.

Logical Architecture Reference Flow: parking_lot_payment_request
This data flow is used within the Provide Electronic Payment Services function and contains the request for the cost of the current parking lot charge to be deducted from the credit currently stored by the payment instrument. It is only sent when a value of stored credit has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: parking_lot_cost.

Logical Architecture Reference Flow: parking_lot_tag_data_request
This data flow is used within the Provide Electronic Payment Services function and contains a request for the parking lot tag data to be read from the store that is held on-board the vehicle.

Physical Architecture Flow Name: tag update

Update data held in tag which can be read at another screening.

Logical Architecture Reference Flow: advanced_parking_lot_charges_confirm
This data flow is used within the Provide Electronic Payment Services function to show that payment for an advanced parking lot charge has been confirmed or not. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + parking_lot_cost + stored_credit.

Logical Architecture Reference Flow: parking_lot_payment_debited
This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the cost of the current parking lot charge will be deducted by the financial institution from the credit identity previously provided by the payment instrument. It is only sent when a credit identity has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: parking_lot_tag_data_clear
This data flow is used within the Provide Electronic Payment Services function and contains the parking lot tag data from which any arrival time has been cleared. The data will have been used to charge for use of the parking lot, and is being cleared to enable its use for future charging. The data flow consists of the following data item which is defined in its own DDE: parking_lot_tag_data.

Logical Architecture Reference Flow: parking_lot_tag_data_update
This data flow is used within the Provide Electronic Payment Services function and contains the parking lot tag data that has been updated. The updated will have loaded the time at which the vehicle entered the parking lot and is for use in charging for the vehicle's use of the lot. The data flow consists of the following data item which is defined in its own DDE: parking_lot_tag_data.

Parking Operator => **Parking Management**

Physical Architecture Flow Name: parking instructions

Instructions from traffic manager or parking operator regarding operation strategy of a parking facility.

Logical Architecture Reference Flow: fpo_current_lot_state
This data flow is sent from a parking lot system to the Manage Traffic function and contains the current parking lot state as provided by the operator. This state may be defined as closed, almost full, full, or spaces.

Logical Architecture Reference Flow: fpo_lot_occupancy
This data flow is sent from a parking lot operator to the Manage Traffic function and contains the current parking lot occupancy in terms of the number of vehicles present as provided by the operator.

Parking Service Provider => Parking Management

Physical Architecture Flow Name: request for performance data

Request issued by a service provider for current parking service performance data.

Logical Architecture Reference Flow: fpsp_confirm_advanced_parking_payment
This data flow is sent from the parking service provider to the Provide Electronic Payment Services function to confirm that an advanced payment of a parking lot charge will be accepted.

Logical Architecture Reference Flow: fpsp_current_lot_state
This data flow is sent from a parking lot service provider to the Manage Traffic function and contains the current parking lot state. This state may be defined as closed, almost full, full, or spaces.

Logical Architecture Reference Flow: fpsp_lot_occupancy
This data flow is sent from a parking lot service provider to the Manage Traffic function and contains the current parking lot occupancy in terms of the number of vehicles present.

Logical Architecture Reference Flow: fpsp_parking_lot_charge_change_response
This data flow is sent to the Provide Electronic Payment Services function by the parking services provider. It contains the response to a previous request for a change to the current parking lot charging structure so that travelers my be encouraged to change the modal split in their journeys.

Logical Architecture Reference Flow: fpsp_parking_lot_data
This data flow is sent from the parking service provider to the Provide Electronic Payment Services function. It contains input of parking lot price and capacity data.

Logical Architecture Reference Flow: fpsp_transaction_reports_request
This data flow is sent form the parking service provider to the Provide Electronic Payment Services function and is a request for the report on parking lot transactions. The request must include the time period and identity of the lots which the report must cover.

Traffic Management => Parking Management

Physical Architecture Flow Name: demand management request

Request to change the demand for road facility use through pricing or other mechanisms.

Logical Architecture Reference Flow: parking_lot_charge_change_request
This data flow is sent from the Manage Traffic function to the Provide Electronic Payment Services function and contains a request for a change to the current parking lot charging structure that will help to influence a change in modal split of journeys currently being undertaken by travelers of all types, i.e. including drivers and transit users, by encouraging then to use certain parking lots, e.g. those near park and ride sites on the edge of an urban area. It consists of the following data items each of which is defined in its own DDE: parking_lot_identity + parking_lot_price +

parking_lot_charge_application_time + vehicle_type_for_charges.

Logical Architecture Reference Flow: parking_lot_charge_direct_request
This data flow is sent from the Manage Traffic function to the Provide Electronic Payment Services function and contains a request for the current prices being charged for parking lot spaces.

Physical Architecture Flow Name: parking instructions

Instructions from traffic manager or parking operator regarding operation strategy of a parking facility.

Logical Architecture Reference Flow: parking_lot_input_data
This data flow is used within the Manage Traffic function and contains data that is used to calculate the occupancy of parking lots. It consists of the following items each of which is defined in its own DDE: parking_lot_list + 1{vehicle_count + vehicle_queue_length}list_size.

Logical Architecture Reference Flow: selected_parking_lot_control_strategy
This data flow is used within the Manage Traffic function and contains the strategy which has been selected for implementation at parking lots to control their use. The strategy will be designed to promote or discourage the use of a parking lot by directing vehicles to or away from it through the use of variable message signs (vms). The decision on which strategy to employ will depend upon such things as the overall traffic management strategy, the need to restrict vehicle use because of a number of factors e.g. congestion, pollution, and the desire to encourage travelers to make use of alternative modes of transport by using park and ride (P+R) facilities. The strategy may be applied to some or all of the parking lots in the geographic area served by the TMC. The data flow consists of the following data items each of which is defined in its own DDE: the may be one of 'open' or 'close' the lot and may be applied to some or all of the lots in the geographic area served by the function. parking_lot_list + selected_parking_lot_strategy_type.

Logical Architecture Reference Flow: static_data_for_parking_lots
This data flow is used within the Manage Traffic function and is provided by the Plan System Deployment function. It contains data that relates vehicle sensors, queue counting sensors and signs to individual parking lots, and the lot occupancy(ies) at which states such as 'almost full' and 'full' will apply. The data is sent to each parking lot for its own use. The data flow consists of the following data items each of which is defined in its own DDE: parking_lot_identity + parking_lot_sensor_allocation + parking_lot_state_thresholds + parking_lot_vms_allocation.

Transit Management => **Parking Management**

Physical Architecture Flow Name: transit parking lot response

Response to transit occupancy inquiries and coordination with parking lots.

Logical Architecture Reference Flow: parking_lot_transit_response
This data flow is sent by the Manage Transit function to the Manage Traffic function and contains the response to a request for new or additional park and ride (P+R) transit services to be provided from the parking lot.

Vehicle => **Parking Management**

Physical Architecture Flow Name: tag data

Unique tag ID and related vehicle information for the purposes of payment for services.

Logical Architecture Reference Flow: advanced_parking_lot_charges_request
This data flow is used within the Provide Electronic Payment Services function to request that a parking lot charge be paid for in advance by a driver who is already paying a charge for the immediate use of a parking lot space. It consists of the following data items each of which is defined in its own DDE: credit_identity + parking_lot_identity +

parking_space_details + stored_credit + vehicle_identity.

Logical Architecture Reference Flow: parking_lot_payment_confirmation
This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the previous request for the cost of the current parking lot charge to be deducted from the credit currently stored by the payment instrument has been completed successfully. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: parking_lot_tag_data_collect
This data flow is used within the Provide Electronic Payment Services function and contains the parking lot tag data that is being collected from on-board the vehicle. This data will be used as the means by which the vehicle will be charged for its use of the parking lot and will consist of the following data item which is defined in its own DDE: parking_lot_tag_data.

Vehicle Characteristics => **Parking Management**

Physical Architecture Flow Name: vehicle characteristics

The physical or visible characteristics of an individual vehicle that can be measured to classify a vehicle and imaged to uniquely identify a vehicle.

Logical Architecture Reference Flow: From_Vehicle_Characteristics
This data flow is sent from the vehicle characteristics terminator. It represents the presence of a vehicle near a sensor, which allows the sensor to create an output that can be used to identify a particular vehicle and its characteristics, such as the number of wheels, size, pollution parameters, etc., for toll payment and parking lot charging purposes. The sensor may also determine the visible characteristics of a vehicle and use that data to obtain information about toll and parking lot charge violators.

2.14.4 Subsystem Architecture Flow Diagram

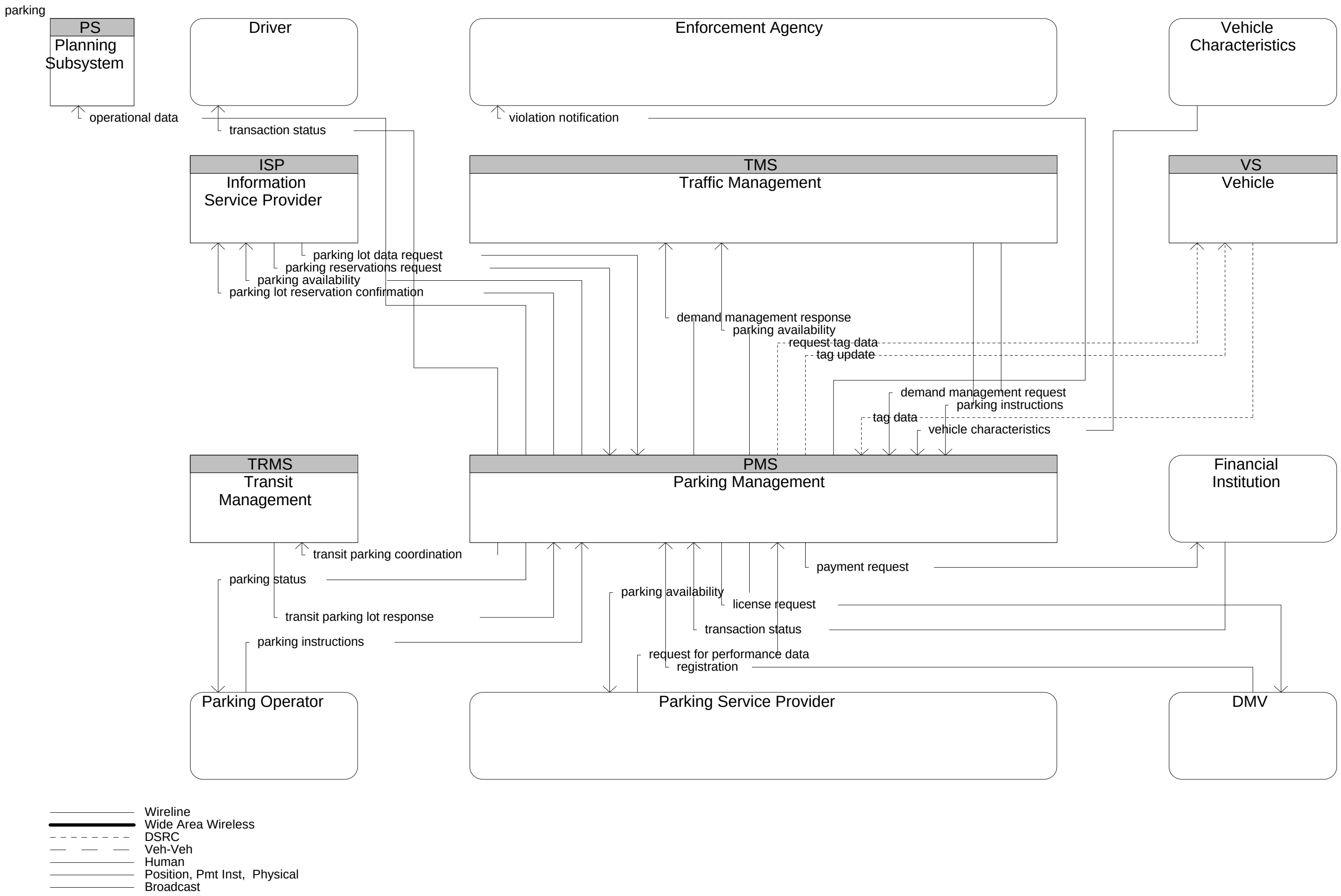


Figure 2.14-2 Architecture Flow Diagram for Parking

2.15 Remote Traveler Support

This subsystem provides access to traveler information at transit stations, transit stops, other fixed sites along travel routes, and at major trip generation locations such as special event centers, hotels, office complexes, amusement parks, and theatres. Traveler information access points include kiosks and informational displays supporting varied levels of interaction and information access. At transit stops, simple displays providing schedule information and imminent arrival signals can be provided. This basic information may be extended to include multi-modal information including traffic conditions and transit schedules along with yellow pages information to support mode and route selection at major trip generation sites. Personalized route planning and route guidance information can also be provided based on criteria supplied by the traveler. In addition to traveler information provision, this subsystem also supports public safety monitoring using CCTV cameras or other surveillance equipment and emergency notification within these public areas. Fare card maintenance, and other features which enhance traveler convenience may also be provided at the discretion of the deploying agency.

2.15.1 Alternative Configurations

The Remote Traveler Support subsystem represents the functions found in a kiosk installed by some service provider to supply traveler information to the user (Figure 2.15-1). It's functions are similar to the Personal Information Access subsystem but it is normally publicly accessible and found at a relatively fixed location such as at a transit stop, in a hotel or convention center. Communication is typically directly to the service provider which installed the kiosk as opposed to the PIAS which may access any service provider directly.

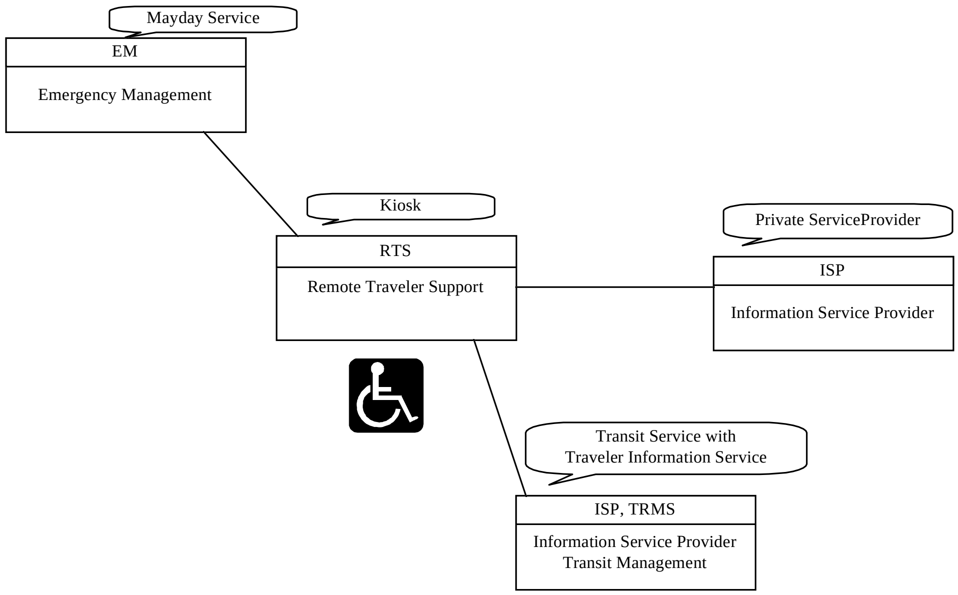


Figure 2.15-1 Alternative Remote Traveler Support Configurations

2.15.2 Subsystem Equipment Packages and Supporting Process Specifications for RTS

Remote Basic Information Reception

This Equipment package shall provide the capability for travelers to interface with the ISP Subsystem Basic Information Broadcast Equipment package and receive formatted traffic advisories including accurate traveling information concerning available travel options and their availability, and congestion information at the Remote Traveler Support Subsystem.

Process Specifications

6.3.2 Inform Traveler

Overview: This process provides the traveler (located at a kiosk) with data about all requested trip, traffic, transit, yellow pages services information, confirmation of any requested reservations, and payments made as part of confirmed trip plans. The data is sent by the process to an interface process that is responsible for its actual output to the traveler. This data may include digitized map data to act as the background to the output when the data is to be shown in a suitable format. This process shall receive data from other ITS functions by requesting it or be sent data as a result of requests from another process.

6.3.3 Provide Traveler Kiosk Interface

Overview: This process shall provide an interface at a kiosk through which travelers can input data and can receive data. The functions that the traveler can perform include plan and confirm trips, obtain current traffic and transit information, and declare emergencies. The process shall support the inclusion of yellow pages services such as lodging, restaurants, theaters, and other tourist activities as a part of trip planning and confirmation. The process shall be able to store frequently used data, such as the kiosk location, to reduce the amount of input needed by the traveler for each request. The process shall also carry out input data verification and require input confirmation before passing any of the traveler data to other processes (except when an emergency is being declared). The traveler's payment information shall be obtained by this process from another process specially designed for that purpose. The process shall support traveler inputs in manual or audio form, and shall provide its outputs in audible or visual forms consistent with a kiosk. These forms shall include those that are suitable for travelers with hearing or vision physical disabilities. The process shall enable viewing of data that has been previously output. Where it is appropriate, the process shall use the kiosk's location to filter data being displayed to only show information relevant to the kiosk's location, or to a specific location requested by the user.

Remote Interactive Information Reception

This Equipment package shall provide the capability for travelers to interface with the ISP Subsystem Infrastructure Equipment packages including the Interactive Infrastructure Information Equipment package, the Infrastructure Provided Route Selection, Yellow Pages and Reservation, and Dynamic Ridesharing Equipment packages. These capabilities shall be provided using the Remote Traveler Support Subsystem equipment such as interactive TV and kiosk using communication medium and equipment such as CATV and wireline and wireless data transceivers.

Process Specifications

6.3.1 Get Traveler Request

Overview: This process shall receive input data from a traveler located at a kiosk and send requests to the appropriate processes within the Provide Driver and Traveler Services function for further processing. The process shall provide support for trip planning, traffic, transit, yellow pages services information requests, trip confirmation, yellow pages confirmation, and payment requests. The actual interface to the traveler is provided through a separate process, which creates the input flow to this process.

6.3.2 Inform Traveler

Overview: This process provides the traveler (located at a kiosk) with data about all requested trip, traffic, transit, yellow pages services information, confirmation of any requested reservations, and payments made as part of confirmed trip plans. The data is sent by the process to an interface process that is responsible for its actual output to the traveler. This data may include digitized map data to act as the background to the output when the data is to be shown in a suitable format. This process shall receive data from other ITS functions by requesting it or be sent data as a result of requests from another process.

6.3.3 Provide Traveler Kiosk Interface

Overview: This process shall provide an interface at a kiosk through which travelers can input data and can receive data. The functions that the traveler can perform include plan and confirm trips, obtain current traffic and transit information, and declare emergencies. The process shall support the inclusion of yellow pages services such as lodging, restaurants, theaters, and other tourist activities as a part of trip planning and confirmation. The process shall be able to store frequently used data, such as the kiosk location, to reduce the amount of input needed by the traveler for each request. The process shall also carry out input data verification and require input confirmation before passing any of the traveler data to other processes (except when an emergency is being declared). The traveler's payment information shall be obtained by this process from another process specially designed for that purpose. The process shall support traveler inputs in manual or audio form, and shall provide its outputs in audible or visual forms consistent with a kiosk. These forms shall include those that are suitable for travelers with hearing or vision physical disabilities. The process shall enable viewing of data that has been previously output. Where it is appropriate, the process shall use the kiosk's location to filter data being displayed to only show information relevant to the kiosk's location, or to a specific location requested by the user.

6.3.4 Update Traveler Display Map Data at Kiosk

Overview: This process shall provide updates to the digitized map data used as the background for displays of trip, traffic and transit information. This data shall be suitable for use in kiosk displays. The process shall obtain the new data from map data suppliers or some other appropriate data source.

7.3.4 Provide Remote Terminal Payment Instrument Interface

Overview: This process shall be responsible for providing the interface through which payment information can be read from a transit user tag. The process shall support reading this data from transit users at the roadside, e.g. a transit stop, for use in paying the current transit fare and (if required) advanced payments. The process shall support advanced payments for tolls, parking lot charges, and/or transit fares. The process shall collect either the credit identity or the stored credit value data from the tag, and update the stored credit value as a result of the fare and (possibly) advanced charges.

7.5.2 Provide Transit User Roadside Payment Instrument Interface

Overview: This process shall be responsible for providing the interface through which credit identities and stored credit values may be collected from tags being used by transit users. The process shall support the collection of this data at the roadside (which in this instance is a transit stop). Payments by the transit user for fares, other services, payment of advanced tolls, and/or parking lot charges shall be supported by the process. It shall also provide an interface through which stored credit held by the tag can be debited for the same types of payment.

7.5.5 Provide Traveler Kiosk Payment Instrument Interface

Overview: This process shall be responsible for providing the interface through which credit identities and stored credit values may be collected from payment instruments being used by travelers. The process shall support the collection of data at the roadside (which in this instance is a kiosk) and use this data for payments needed to confirm a traveler's trip. Payments supported by the process shall include those for advanced tolls, parking lot charges, transit fares, and/or other (yellow pages) services. It shall also provide an interface through which the stored credit held by the tag can be debited for the same types of payment.

Remote Mayday I/F

This Equipment package provides the capability to report an emergency and summons assistance. The equipment includes a traveler interface that facilitates generation of a distress signal under duress and wireline communications that carries this distress signal and allows follow-up verification and determination of the nature of the emergency and the required response. This equipment package notifies either the Emergency Management or Transit Management Subsystem depending on the implementation.

Process Specifications

4.4.1.8 Report Traveler Emergencies

Overview: This process shall provide an interface in the Provide Driver and Traveler Services function through which travelers can declare emergencies. The traveler may be at a kiosk or other device, transit stop, transit depot, etc. The input and output forms shall include those that are suitable for travelers with physical disabilities.

6.3.3 Provide Traveler Kiosk Interface

Overview: This process shall provide an interface at a kiosk through which travelers can input data and can receive data. The functions that the traveler can perform include plan and confirm trips, obtain current traffic and transit information, and declare emergencies. The process shall support the inclusion of yellow pages services such as lodging, restaurants, theaters, and other tourist activities as a part of trip planning and confirmation. The process shall be able to store frequently used data, such as the kiosk location, to reduce the amount of input needed by the traveler for each request. The process shall also carry out input data verification and require input confirmation before passing any of the traveler data to other processes (except when an emergency is being declared). The traveler's payment information shall be obtained by this process from another process specially designed for that purpose. The process shall support traveler inputs in manual or audio form, and shall provide its outputs in audible or visual forms consistent with a kiosk. These forms shall include those that are suitable for travelers with hearing or vision physical disabilities. The process shall enable viewing of data that has been previously output. Where it is appropriate, the process shall use the kiosk's location to filter data being displayed to only show information relevant to the kiosk's location, or to a specific location requested by the user.

Remote Transit Fare Management

This Equipment package provides the capability for the traveler to use a common fare medium for all applicable surface transportation services, to pay without stopping, have payment media automatically identified as void and/or invalid and eligibility verified. This may be implemented as a payment instrument reader at a kiosk. In addition, capability to provide expansion into other uses for payment medium such as retail and telephone and for off-line billing for fares paid by agencies shall be supported.

Process Specifications

4.7.2.1 Detect Transit User at Roadside

Overview: This process shall detect transit users embarking at a roadside transit stop and read data from the payment instrument that they are carrying. The process shall provide an image of all transit users which shall be used for violation processing of those who do not have a payment instrument or whose transit fare transaction fails. It shall obtain an image of the required accuracy under all lighting conditions and over the range of speeds with which transit users will pass through the fare collection point at the roadside, i.e., a transit stop.

4.7.2.2 Determine Transit User Needs at Roadside

Overview: This process shall determine the transit user's travel routing based on the user's destination and the location of the roadside transit stop from which the route request is being made. The process shall support the transit user's routing enabling it to include travel on all or part of the route(s) operating from the stop and (possibly) transfer to another route. In order for this to be achieved, the process requires access to the complete range of transit services (routes and schedules) that are available to the transit user. Details of all transactions with the transit user's payment details removed, shall be sent by this process to the interface process for loading into the transit roadside fare collection data store.

4.7.2.3 Determine Transit Fare at Roadside

Overview: This process shall calculate the transit user's fare based on the origin and destination provided by the user. The process shall calculate the fare using the transit routing, transit fare category, and transit user history components of the ride data together with data provided by the interface process to the database of transit fares. The accumulated data shall be sent by the process to another process for the actual implementation of the fare payment transaction.

4.7.2.4 Manage Transit Fare Billing at Roadside

Overview: This process shall generate the data necessary to enable the financial transaction between the transit user and the transit provider to be completed at the roadside, i.e., at a transit stop. The process shall accept and process current transit passenger fare collection information. The process shall perform the front end transaction between the transit user and the transit system, and use the infrastructure interactive mode of operation to complete the back end processing. This means that the process shall send data about each transaction to processes in the Provide Electronic Payment Services function for the back end financial authorization and transaction processing. The process shall then await the return of the result for display to the transit user before accepting the next transaction. A failed transaction shall result in the transmission of an image of the transit user to another process. A record of the status of all transit fare processing shall be sent to another process for storage in a fare collection database.

4.7.2.5 Provide Transit User Roadside Fare Interface

Overview: This process shall provide the interface for the transit user at the roadside, i.e., at a transit stop. The interface shall enable the transit user to specify the required destination of a transit service ride and request other (yellow pages) services. The process shall prompt the transit user for information necessary for the transaction that has not been provided. The result of the transit service ride fare payment plus other services request and payment, shall be reported back to the transit user by the process. The input and output forms shall include those that are suitable for travelers with physical disabilities.

4.7.2.6 Update Roadside Transit Fare Data

Overview: This process shall provide a database at the roadside, i.e., a transit stop, for use in fare processing. The database shall contain transit fare information from which the fares for all possible trips within the transit operational network can be determined.

4.7.2.7 Provide Transit Roadside Passenger Data

Overview: This process shall create passenger loading and fare statistics data based upon data collected at the roadside and send this data to the store of transit operations data. The process may send the data at regular periodic intervals, on-demand, or through some other trigger mechanism. The process shall create its outputs using information collected in the store of fare transaction data. This data is received from other processes at the roadside, i.e., at a transit stop.

7.3.4 Provide Remote Terminal Payment Instrument Interface

Overview: This process shall be responsible for providing the interface through which payment information can be read from a transit user tag. The process shall support reading this data from transit users at the roadside, e.g. a transit stop, for use in paying the current transit fare and (if required) advanced payments. The process shall support advanced payments for tolls, parking lot charges, and/or transit fares. The process shall collect either the credit identity or the stored credit value data from the tag, and update the stored credit value as a result of the fare and (possibly) advanced charges.

7.5.2 Provide Transit User Roadside Payment Instrument Interface

Overview: This process shall be responsible for providing the interface through which credit identities and stored credit values may be collected from tags being used by transit users. The process shall support the collection of this data at the roadside (which in this instance is a transit stop). Payments by the transit user for fares, other services, payment of advanced tolls, and/or parking lot charges shall be supported by the process. It shall also provide an interface through which stored credit held by the tag can be debited for the same types of payment.

Remote Transit Information Services

The Equipment package furnishes transit users with real-time travel-related information at transit stops, multi-modal transfer points, and other public transportation areas. It provides transit users with the latest available information on transit routes, schedules, transfer options, fares, real-time schedule adherence, current incidents, weather conditions, and special events. In addition to tailored information for individual transit users, this equipment package supports general annunciation and/or display of imminent arrival information and other information of general interest to transit users.

Process Specifications

4.7.1.1 Provide Transit User Roadside Data Interface

Overview: This process shall communicate with the transit management center (TRM) by providing public transit information at roadside locations. These locations may consist of transit vehicle stops or other locations that provide general public transit information. The process shall enable the roadside unit to obtain information about the transit services on request from the local transit user interface process and to receive data about late running services from other processes within the Manage Transit function. The received data shall be loaded into a local data store for future use. The input and output forms shall include those that are suitable for travelers with physical disabilities.

4.7.1.2 Provide Transit User Roadside Vehicle Data Interface

Overview: This process shall provide the roadside (transit stop) interface through which transit users receive information about an approaching transit vehicle or one that has already arrived. The process shall output the data to the transit user as soon as it is received and shall load the data into the local store for future use. Output of the data shall be maintained until the vehicle leaves the stop, when the process shall cease output of the data and delete it from the local store. The input and output forms shall include those that are suitable for travelers with physical disabilities.

Secure Area Monitoring
This Equipment package provides the capability to monitor the safety of transit users at Remote Traveler Subsystem locations. It collects surveillance images and data and relays this information back to the Transit Management Subsystem.

Process Specifications
4.4.1.7 Monitor Secure Area

Overview: This process shall monitor the secure area environment. Data shall be obtained by the process from a variety of sources and assessed for any security problems. Problems shall be passed by the process to other processes for review and the required action. Information about incidents shall also be sent by this process to another process for output to the media, using interface parameters set up by the transit system operator. The process shall also provide facilities for the control of video cameras and audio output in the secure area environment.

2.15.3 Subsystem Interfaces for RTS

Emergency Management => **Remote Traveler Support**

Physical Architecture Flow Name: emergency acknowledge

Acknowledge request for emergency assistance and provide additional details regarding actions and verification requirements.

Logical Architecture Reference Flow: emergency_request_traveler_acknowledge
This data flow is used by the Manage Emergency Services function to confirm that the request for emergency services previously sent by the traveler has been received from a kiosk or other device and is therefore sent to the Provide Driver and Traveler Services function for output. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Information Service Provider => **Remote Traveler Support**

Physical Architecture Flow Name: broadcast information

General broadcast information that contains link travel times, incidents, advisories, transit services and a myriad of other traveler information.

Logical Architecture Reference Flow: traffic_data_for_kiosks
This data flow is sent from the Manage Traffic function to the Provide Driver and Traveler Services function. It is used to provide data on the traffic flowing in the road network, plus that which is predicted to flow in the network for output at a kiosk. The flow consists of the following items each of which is defined in its own DDE: kiosk_identity + source_identity + current_data_for_broadcast + predicted_data_for_broadcast.

Logical Architecture Reference Flow: transit_deviations_for_kiosks
This data flow is sent by the Manage Transit function to the Provide Driver and Traveler Services function and contains current transit service deviations for output to a kiosk. It consists of the following data items each of which is defined in its own DDE: kiosk_identity + transit_vehicle_schedule_deviations.

Physical Architecture Flow Name: traveler information

Traveler information comprised of traffic status, advisories, incidents, responses to traveler requests (e.g., traveler routing, yellow pages), payment information and many other travel-related data updates and confirmations.

Logical Architecture Reference Flow: advanced_tolls_and_charges_roadside_confirm
This data flow is used within the Provide Electronic Payment Services function and contains the result of the requested advanced payment transaction from a traveler (as a transit user) at the roadside, i.e. a transit stop. It consists of the following data items each of which is defined in its own DDE: advanced_charges_confirm + advanced_tolls_confirm + confirmation_flag.

Logical Architecture Reference Flow: traffic_data_for_kiosks
This data flow is sent from the Manage Traffic function to the Provide Driver and Traveler Services function. It is used to provide data on the traffic flowing in the road network, plus that which is predicted to flow in the network for output at a kiosk. The flow consists of the following items each of which is defined in its own DDE: kiosk_identity + source_identity + current_data_for_broadcast + predicted_data_for_broadcast.

Logical Architecture Reference Flow: transit_deviations_for_kiosks
This data flow is sent by the Manage Transit function to the Provide Driver and Traveler Services function and contains current transit service deviations for output to a kiosk. It consists of the following data items each of which is defined in its own DDE: kiosk_identity + transit_vehicle_schedule_deviations.

Logical Architecture Reference Flow: traveler_payment_confirmation
This data flow is used within the Provide Driver and Traveler Services function to indicate the payment for a confirmed trip has been successfully completed., or that the total cost can now be deducted from the credit stored on the traveler's payment instrument The request for payment will have been initiated by input from the traveler to a kiosk. The data flow consists of the following data items each of which is defined in its own DDE: advanced_tolls_confirm + advanced_fares_confirm + advanced_parking_lot_charges_confirm + credit_identity + kiosk_identity + stored_credit + traveler_total_trip_cost.

Logical Architecture Reference Flow: traveler_transaction_confirmation
This data flow is used within the Provide Driver and Traveler Services function to confirm any reservations made by the traveler from a kiosk. These reservations will be based on information obtained by the traveler from previous data input and output through the kiosk. The data flow consists of the following data items each of which is defined in its own DDE: credit_identity + kiosk_identity + transaction_number + yellow_pages_cost + yellow_pages_lodging_reservation_confirmation + yellow_pages_dining_reservation_confirmation + yellow_pages_ticket_purchase_confirmation.

Logical Architecture Reference Flow: traveler_yellow_pages_data
This data flow is used within the Provide Driver and Traveler Services function and contains details of other (yellow pages) services which is to be sent to the traveler interface facility for output using a kiosk. The size of the data flow has been set to take account of the need to provide only a small percentage of the total yellow pages data that is available. The data flow consists of the following data items each of which is defined in its own DDE: kiosk_identity + yellow_pages_general_information + yellow_pages_specific_information + yellow_pages_transaction_information.

Physical Architecture Flow Name: trip plan

A sequence of links and special instructions comprising a trip plan indicating efficient routes for navigating the links. Normally coordinated with traffic conditions, other incidents, preemption and prioritization plans.

Logical Architecture Reference Flow: traveler_trip_information
This data flow is used within the Provide Driver and Traveler Services function and contains information about a proposed trip that the traveler has requested earlier from the kiosk. It consists of the following data items each of which is defined in its own DDE: current_conditions + kiosk_identity + paratransit_personal_schedule + rideshare_response + 1{route + route_cost}4.

Map Update Provider => **Remote Traveler Support**

Physical Architecture Flow Name: map updates

Map update which could include a new underlying static or real-time map or map layer(s) update.

Logical Architecture Reference Flow: fmup_traveler_display_update
This data flow is sent from the map update provider to the Provide Driver and Traveler Services function. It contains the digitized map data that can be used as background to displays of traffic, trip and travel information that are output to a kiosk for use by travelers. The data will be different from that sent for output of similar displays at a personal device since the type of display is likely to be different (larger) in this case.

Payment Instrument => **Remote Traveler Support**

Physical Architecture Flow Name: Payment

Payment of some kind (e.g., toll, parking, fare) by traveler which in most cases can be related to a credit account.

Logical Architecture Reference Flow: fpi_confirm_fare_payment_at_roadside
This data flow is sent from the payment instrument to the Provide Electronic Payment Services function. It contains confirmation that the requested payment of the current transit fare, plus if required the cost of advanced tolls, and/or parking lot charges, and/or transit fares, has been successfully deducted from the total credit previously stored by the payment instrument. This data flow will only apply to those types of payment instrument that can carry stored credit and will not be set by those that only contain a credit identity.

Logical Architecture Reference Flow: fpi_transit_roadside_tag_data
This data flow is sent from the payment instrument to the Provide Electronic Payment Services function and is used to either identify a particular payment instrument or the amount of credit that it currently has stored, when the instrument is being used at the roadside, i.e. a transit stop. In either case the data will be used to enable automatic billing for the current transit fare, plus if required, advanced payments for tolls, and/or parking lot charges and/or transit fares. A payment instrument is a device that can be used to make payments, e.g. a debit card, or a device that contains stored credit that can be used for actual payments. It will belong to the financial institution responsible for its issue and not to the user, who in this instance is the transit user.

Logical Architecture Reference Flow: fpi_transit_user_roadside_input_credit_identity
This data flow is sent from the payment instrument to the Provide Electronic Payment Services function. It contains the data necessary to allow automatic billing of the transit user for advanced tolls, and/or parking lot charges, and/or transit fares, when the user is at the roadside, i.e. a transit stop. A payment instrument is a device that can be used to make payments, e.g. a debit card, or a stored value card. It will belong to the financial institution responsible for its issue and not to the user.

Logical Architecture Reference Flow: fpi_traveler_roadside_input_credit_identity
This data flow is sent from the payment instrument to the Provide Electronic Payment Services function. It contains the data necessary to allow automatic billing of the user of the payment instrument when they are at a kiosk that provides facilities for traveler information and trip planning. A payment instrument is a device that can be used to make payments, e.g. a debit card, or a stored value card which itself contains credit that can be used to make payments. It will belong to the financial institution responsible for its issue and not to the user.

Remote Traveler Support => **Emergency Management**

Physical Architecture Flow Name: emergency notification

An emergency request for assistance originated by a traveler using an in-vehicle, public access, or personal device. Sufficient information is provided so that the recipient can determine the location of the emergency as a minimum. Additional information identifying the requestor and requesting device and the nature and severity of the emergency may also be provided (and required) by some systems.

Logical Architecture Reference Flow: emergency_request_traveler_details
This data flow is used by the Provide Driver and Traveler Services function to send data about an emergency declared by a traveler using a kiosk or other device to the Manage Emergency Services function. It contains the following data items each of which is defined in its own DDE: date + time + traveler_personal_emergency_request.

Remote Traveler Support => **Information Service Provider**

Physical Architecture Flow Name: traveler request

Request by a traveler to summon assistance, request information, make a reservation, or initiate any other traveler service.

Logical Architecture Reference Flow: advanced_tolls_and_charges_roadside_request
This data flow is used by the Manage Transit function to transfer requests for advanced payments for toll and parking lot charges from the traveler (as a transit user) fare payment interface at the roadside, i.e. a transit stop, to the Provide Electronic Payment Services function for subsequent processing. The size of the data flow has been set at less than the sum of the two constituent flows to allow for the fact that they will both not be present for every data transfer. It consists of the following data items each of which is defined in its own DDE: advanced_charges + advanced_tolls.

Logical Architecture Reference Flow: traffic_data_kiosk_request
This data flow is sent by the Provide Driver and Traveler Services function to the Manage Traffic function and contains the request for the provision of traffic data for output at a kiosk. It consists of the following data items each of which is defined in its own DDE: kiosk_identity + traffic_data_request.

Logical Architecture Reference Flow: transit_deviation_kiosk_request
This data flow is sent by the Provide Driver and Traveler Services function to the Manage Transit function and is a request for data on current transit service deviations for output to a kiosk. It consists of the following data items each of which is defined in its own DDE: kiosk_identity + transit_vehicle_deviation_request.

Logical Architecture Reference Flow: traveler_current_condition_request
This data flow is used within the Provide Driver and Traveler Services function and contains a request for details of the current conditions, e.g. weather, events, incidents, etc. The request includes the identity of the kiosk from which the request was input by the traveler so that the response can be correctly returned. The data flow consists of the following item which is defined in its own DDE: kiosk_identity.

Logical Architecture Reference Flow: traveler_payment_information
This data flow is used within the Provide Driver and Traveler Services function. It contains details of the components of a trip which a traveler has obtained from the input of data to a kiosk and for which advanced payment is needed following trip confirmation. The traveler's identity and credit identity or stored credit from the payment instrument are therefore also included to enable payment to be made. The data flow consists of the following items each of which is defined in its own DDE: credit_identity + kiosk_identity + parking_space_details + ride_segments + stored_credit + toll_route_segments.

Logical Architecture Reference Flow: traveler_transaction_request
This data flow is used within the Provide Driver and Traveler Services function and contains data input by the traveler at a kiosk to make reservations for various other (yellow pages) services. It contains the following data items, each of which is defined in its own DDE: yellow_pages_dining_reservation + yellow_pages_lodging_reservation + yellow_pages_ticket_purchase.

Logical Architecture Reference Flow: traveler_yellow_pages_information_request
This data flow is used within the Provide Driver and Traveler Services function and contains a request for data on other (yellow pages) services to be provided to a traveler at the identified kiosk. As no filtering components are included, all

the data currently available will be provided. The data flow consists of the following data item which is defined in its own DDE: kiosk_identity.

Physical Architecture Flow Name: trip confirmation

Acknowledgement by the driver/traveler of acceptance of a route.

Logical Architecture Reference Flow: traveler_trip_confirmation
This data flow is used within the Provide Driver and Traveler Services function to confirm the trip details provided as the result of a traveler's previous trip request input from a kiosk. It contains the following data item which is defined in its own DDE: paratransit_service_confirmation + traveler_identity + traveler_rideshare_confirmation.

Physical Architecture Flow Name: trip request

Request by a driver/traveler for special routing.

Logical Architecture Reference Flow: traveler_trip_request
This data flow is used within the Provide Driver and Traveler Services function and contains data about a traveler's trip request which has been input from a kiosk. It consists of the following data items each of which is defined in its own DDE: trip_request + traveler_identity + traveler_rideshare_request.

Physical Architecture Flow Name: yellow pages request

Request for information through a yellow pages type service.

Logical Architecture Reference Flow: traveler_yellow_pages_information_request
This data flow is used within the Provide Driver and Traveler Services function and contains a request for data on other (yellow pages) services to be provided to a traveler at the identified kiosk. As no filtering components are included, all the data currently available will be provided. The data flow consists of the following data item which is defined in its own DDE: kiosk_identity.

Remote Traveler Support => **Map Update Provider**

Physical Architecture Flow Name: map update request

Request for a map update which could include a new underlying map or map layer updates.

Logical Architecture Reference Flow: tmup_request_traveler_display_update
This data flow is sent to the map update provider from the Provide Driver and Traveler Services function. It contains a request for an update to the digitized map data used for displays that can be output as background for traffic, trip and travel information for use by travelers at kiosks. It must include the identity of the kiosk from which the request has originated so that the map update provider can determine the map data to supply that will be relevant to the area in which the kiosk is located.

Remote Traveler Support => **Payment Instrument**

Physical Architecture Flow Name: request for payment

Request to deduct cost of service from user's payment account.

Logical Architecture Reference Flow: tpi_debited_fare_payment_at_roadside
This data flow is sent to the payment instrument by the Provide Electronic Payment Services function and contains

confirmation that the cost of the current transit fare incurred at the roadside, i.e. a transit stop, plus if required the cost of advanced tolls, and/or parking lot charges, and/or transit fares, will be debited to the credit identity provided by the payment instrument. The debit transaction will be carried out through the financial institution through other processes within the function.

Logical Architecture Reference Flow: tpi_debited_transit_user_payment_at_roadside
This data flow is sent to the payment instrument from the Provide Electronic Payment Services function. It is a request to deduct the cost of advanced payments from the value of credit currently stored by the payment instrument belonging being used by a transit user at the roadside, i.e. a transit stop. The advanced payments may cover tolls, and/or parking lot charges, and/or transit fares.

Logical Architecture Reference Flow: tpi_debited_traveler_payment_at_roadside
This data flow is sent to the payment instrument from the Provide Electronic Payment Services function. It is a request to deduct the cost of a traveler's confirmed trip from the value of credit currently stored by the payment instrument being used by the traveler.

Logical Architecture Reference Flow: tpi_request_fare_payment_at_roadside
This data flow is sent to the payment instrument from the Provide Electronic Payment Services function. It is a request to deduct the total cost of the current transit fare, or if required, that for advanced tolls, and/or parking lot charges and/or transit fares from the credit currently stored by the payment instrument when used at the roadside, i.e. a transit stop.

Remote Traveler Support => Transit Management

Physical Architecture Flow Name: emergency notification

An emergency request for assistance originated by a traveler using an in-vehicle, public access, or personal device. Sufficient information is provided so that the recipient can determine the location of the emergency as a minimum. Additional information identifying the requestor and requesting device and the nature and severity of the emergency may also be provided (and required) by some systems.

Logical Architecture Reference Flow: emergency_request_transit_details
This data flow is sent by the Provide Driver and Traveler Services function to the Manage Transit function to send data about an emergency declared by a traveler at a transit stop using a kiosk or other device to the Manage Emergency Services function. This can also be used by the transit user to alert the transit system operator to an emergency situation or incident within the transit operational network, i.e. not on-board a transit vehicle, or at a transit stop, but in such things as a modal interchange facility, transit depot, etc. It contains the following data items each of which is defined in its own DDE: date + time + transit_emergency_request.

Logical Architecture Reference Flow: transit_user_roadside_image
This data flow is sent from the Manage Transit function to the Provide Electronic Payment Services function. It contains an JPEG compressed image of the transit user who has violated the transit fare collection process at the roadside, i.e. at a transit stop. The data will be used in subsequent transit fare violation processing.

Physical Architecture Flow Name: secure area surveillance data

Data collected from surveillance systems used to monitor secure areas. Includes video, audio, and other security sensor outputs.

Logical Architecture Reference Flow: secure_area_surveillance_information
This data flow is sent from the Provide Driver and Traveler Services Function to the Manage Transit function and represents information about conditions in a secure area environment such as that found in a transit network. This information is sensed/detected by sensors contained in the Manage Transit function, and includes video, audio, and other image data. The data may have been obtained from closed circuit television (cctv), or other systems that are monitoring activity in the transit operational network, i.e. not on-board a transit vehicle, but at a transit stop, or in such

things as a modal interchange facility, transit depot, etc. The data can be used for incident detection, etc., using automatic analysis techniques.

Physical Architecture Flow Name: transit fare payment requests

Information provided from the transit user location that supports fare payments and associated record keeping.

Logical Architecture Reference Flow: fare_collection_roadside_violation_information
This data is used by the Manage Transit function to send data about a violator of the transit fare collection processes at the roadside, i.e. a transit stop to the Manage Emergency Services function. This data flow will contain a digitized video image of the transit user who is trying to violate the fare collection process at the roadside. It is assumed that this digitized data will include other data such as date and time, plus camera identity from which the roadside (transit stop) location can be determined. The data flow consists of the following data items each of which is defined in its own DDE: transit_route_number + transit_route_segment_number + transit_user_roadside_image + transit_user_roadside_tag_identity.

Logical Architecture Reference Flow: request_roadside_fare_payment
This data flow is sent from the Manage Transit function to the Provide Electronic Payment Services function to request payment of a transit fare from the roadside, i.e. a transit stop. It consists of the following data items each of which is defined in its own DDE: transit_fare + transit_roadside_fare_collection_identity + transit_user_roadside_tag_identity.

Logical Architecture Reference Flow: transit_roadside_fare_payment_confirmation
This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the previous request for the cost of the current transit fare has been deducted successfully from the credit currently stored by the transit user's payment instrument. The data flow is used when the transit user is paying for the transit fare at the roadside and consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: transit_roadside_passenger_data
This data flow is used within the Manage Transit function. It contains the number of transit users (passengers) who in a twenty four hour period, have passed through a transit stop plus data about the ride which they purchased. The data is derived from roadside fare collection data and is for use in the determination of future transit services. The data flow consists of the following data items each of which is defined in its own DDE: transit_route_stop_number + transit_user_journey_start + 24{list_size + list_size{transit_passenger_numbers + transit_user_journey_end + transit_route_use_time + transit_user_category}}.

Physical Architecture Flow Name: transit information user request

Request for special transit routing, real-time schedule information, and availability information.

Logical Architecture Reference Flow: other_services_roadside_request
This data flow is sent from the Manage Transit function to the Provide Electronic Payment Services function and contains the transit user's request from the roadside, i.e. a transit stop, for other (yellow pages) services. It consists of the following data items each of which is defined in its own DDE : traveler_identity + credit_identity + other_services_data.

Logical Architecture Reference Flow: transit_services_kiosk_request
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Transit function. It is a request for details of transit services for output to a kiosk. The traveler will have to provide the origin and destination so that the receiving process can work out for which transit route(s) data will have to be provided. The data flow consists of the following data items each of which is defined in its own DDE: destination + kiosk_identity + origin.

Logical Architecture Reference Flow: transit_services_travelers_request
This data flow is used within the Manage Transit function to request the details of the current transit services for a transit user at the roadside. The transit user will have to provide the origin and destination so that the receiving process

can work out for which transit route(s) data will be provided. The data flow consists of the following data items each of which is defined in its own DDE: destination + origin + traveler_identity.

Remote Traveler Support => Transit User

Physical Architecture Flow Name: transit user fare status

Status of fare transaction for transit user.

Logical Architecture Reference Flow: ttu_roadside_access_message
This data flow is sent to the transit user by the Manage Transit function. It contains a message giving the transit user details of the success or failure of the fare transaction previously initiated from at the roadside, i.e. the transit stop.

Logical Architecture Reference Flow: ttu_roadside_payment_confirmed
This data flow is sent to the transit user by the Manage Transit function. It contains a message giving the transit user at the roadside, i.e. the transit stop, details of the success or failure of the payment transaction..

Physical Architecture Flow Name: transit user outputs

Information for traveler from either an on-board or fixed location traveler information station.

Logical Architecture Reference Flow: ttu_other_services_roadside_confirmed
This data flow is sent to the transit user by the Manage Transit function. It contains a message giving the transit user at the roadside, i.e. the transit stop, details of the success or failure of the request for other (yellow pages) services.

Logical Architecture Reference Flow: ttu_transit_information
This data flow is sent to the transit user by the Manage Transit function. It contains a message giving the transit user details of transit services that are currently available.

Logical Architecture Reference Flow: ttu_transit_vehicle_information
This data flow is sent to the transit user from the Manage transit function and contains information such as route number, service number, etc. about a transit vehicle that has just arrived at a transit stop.

Remote Traveler Support => Traveler

Physical Architecture Flow Name: traveler interface updates

Visual or audio information (e.g., routes, messages, guidance) to the traveler.

Logical Architecture Reference Flow: tt_emergency_response
This data flow is used as part of the interface to the traveler by the Provide Driver and Traveler Services function. It contains acknowledgment of an emergency previously declared by a traveler from a kiosk.

Logical Architecture Reference Flow: tt_extra_trip_data_request
This data flow is used as part of the interface to the traveler by the Provide Driver and Traveler Services function. It contains outputs about the trip that the traveler has previously requested from a kiosk, or messages about the previous confirmation of this trip.

Logical Architecture Reference Flow: tt_trip_planning_responses
This data flow is used as part of the interface to the traveler by the Provide Driver and Traveler Services function. It contains the result of requests for more data about the trip on which the traveler is requesting information from a kiosk.

Secure Area Environment => **Remote Traveler Support**

Physical Architecture Flow Name: secure area characteristics

Characteristics (visual, audible, other) that are monitored by surveillance security systems via sensors.

Logical Architecture Reference Flow: fsa_area_image

This data flow is sent from the secure area environment to the Manage Transit function and represents information about conditions in a secure area environment such as that found in a transit network. Also represented is information about visual and audible characteristics of travelers and surrounding areas that are monitored with security surveillance systems. This data is sensed/detected by sensors contained in the Manage Transit function, and includes video, audio, and other image data. The data may have been obtained from closed circuit television (cctv), or other, systems that are monitoring activity in the transit operational network, i.e. not on-board a transit vehicle, but at a transit stop, or in such things as a modal interchange facility, transit depot, etc. The data can be used for incident detection, etc., using automatic analysis techniques.

Transit Management => **Remote Traveler Support**

Physical Architecture Flow Name: emergency acknowledge

Acknowledge request for emergency assistance and provide additional details regarding actions and verification requirements.

Logical Architecture Reference Flow: emergency_acknowledge_transit_details

This data flow is sent by the Manage Transit function to the Provide Driver and Traveler Services function to confirm that the request for emergency services previously sent by the traveler has been received from a kiosk or other device. This data flow may also contain the response to input from a panic button that has been activated by a transit user in part of the transit operational network, i.e. not on-board a transit vehicle, or at a transit stop, but in such things as a modal interchange facility, transit depot, etc. The information will be sent out as part of the response to an emergency or incident being detected within the network. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: request_transit_user_roadside_image

This data flow is sent from the Provide Electronic Payment Services function to the Manage Transit function. It contains a request for the supply of the image of a transit user who has violated the transit fare payment process at a roadside fare collection point.

Physical Architecture Flow Name: secure area monitoring support

Commands that control surveillance equipment and security sensors that monitor secure public transportation areas. Also includes information for general advisories and alerts intended for general dissemination in these same public areas.

Logical Architecture Reference Flow: secure_area_broadcast_message

This data flow is sent to the Provide Driver and Traveler Services function by the Manage Transit function and contains textual information for transit users in part of the transit operational network, i.e. not on-board a transit vehicle, or at a transit stop, but in such things as a modal interchange facility, transit depot, etc. The information will be sent out as part of the response to an emergency or incident being detected within the network.

Logical Architecture Reference Flow: secure_area_monitoring_control

This data flow is sent to the Provide Driver and Traveler Services function by the Manage Transit function and contains control data for closed circuit television (cctv) systems, or audio equipment, located in the secure area environment. This data may change the pan, tilt, zoom, or other camera or audio operating parameters and may be generated automatically or as a result of input from the transit system operator.

Physical Architecture Flow Name: transit fare payment responses

Information provided by transit management that supports a fare payment transaction

Logical Architecture Reference Flow: confirm_roadside_fare_payment

This data flow is sent from the Provide Electronic Payment Services function to the Manage transit function to confirm that transaction processing of the payment of a transit fare from the roadside, i.e. a transit stop, has been completed. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + transit_roadside_fare_collection_identity.

Logical Architecture Reference Flow: transit_roadside_fare_data

This data flow is sent by the Provide Electronic Payment Services function to the Manage Transit function and contains details of the fares being currently charged for regular transit services. It is for use in calculating fares that are to be paid by transit users at the roadside, i.e. a transit stop, and consists of the following data item which is defined in its own DDE: transit_fare_data.

Logical Architecture Reference Flow: transit_roadside_fare_payment_debited

This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the cost of the current transit fare will be deducted by the financial institution from the credit identity previously provided by the payment instrument being used by the transit user on-board a transit vehicle. It is only sent when a credit identity has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: transit_roadside_fare_payment_request

This data flow is used within the Provide Electronic Payment Services function and contains the request for the cost of the current transit fare to be deducted from the credit currently stored by the transit user's payment instrument, when it is being used at the roadside, i.e. a transit stop. It is only sent when a value of stored credit has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: transit_fare.

Logical Architecture Reference Flow: transit_services_for_roadside_fares

This data flow is used within the Manage Transit function. It contains details of the transit user fares for all the transit routes operated by the transit fleet from which the request was made. This data is for use in processing transit fare payments initiated by transit users at the roadside (a transit stop). The data flow consists of the following data item which is defined in its own DDE: transit_route_fare_data.

Physical Architecture Flow Name: transit traveler information

Transit information prepared to support transit users and other travelers. It contains transit schedules, real-time arrival information, fare schedules, and general transit service information.

Logical Architecture Reference Flow: other_services_roadside_response

This data flow is sent from the Provide Electronic Payment Services function to the Manage Transit function and contains the response to the transit user's request from the roadside, i.e. a transit stop, for other (yellow pages) services. It consists of the following data items each of which is defined in its own DDE : traveler_identity + credit_identity + other_services_data.

Logical Architecture Reference Flow: transit_services_for_kiosks

This data flow is sent from the Manage Transit function to the Provide Driver and Traveler Services function. It contains details of the transit services that satisfy a traveler's request and are for output to a kiosk. The data flow consists of the following data items each of which is defined in its own DDE: kiosk_identity + 1{transit_services_for_output}2.

Logical Architecture Reference Flow: transit_services_for_travelers

This data flow is used within the Manage Transit function. It is sent to the Provide Traveler Transit Interface facility and contains a complete set of all the transit routes and the services that run upon them, including timings, etc. that are provided by the transit fleet from which the data was requested. The data flow consists of the following data items each of which is defined in its own DDE: traveler_identity + 1{transit_services_for_output}2.

Logical Architecture Reference Flow: transit_vehicle_arrival_time
This data flow is used within the Manage Transit function. It contains the estimated time of arrival of a transit vehicle at a stop plus the route and service number on which it is operating.

Logical Architecture Reference Flow: transit_vehicle_user_data
This data flow is used within the Manage Transit function and contains data about a transit vehicle for automatic output to transit users at transit stops. The data is output at the transit stop as the vehicle approaches and contains information about the vehicle such as the route number. It therefore consists of the following data items each of which is defined in its own DDE: transit_route_number + transit_vehicle_time.

Transit User => **Remote Traveler Support**

Physical Architecture Flow Name: transit user inputs

Requests from transit user through either an on-board or fixed location traveler information station.

Logical Architecture Reference Flow: ftu_destination_at_roadside
This data flow is sent by the transit user to the Manage Transit function and is used to specify the destination of a desired service for which a fare has to be paid at the roadside, i.e. a transit stop.

Logical Architecture Reference Flow: ftu_other_services_roadside_request
This data flow is sent by the transit user to the Manage Transit function to specify other non-transit services that are needed by a transit user at the roadside, i.e. a transit stop.

Logical Architecture Reference Flow: ftu_transit_information_request
This data flow is sent by the transit user to the Manage Transit functions to request information on transit services from a kiosk or other information point.

Logical Architecture Reference Flow: ftu_transit_user_roadside_image
This data flow is used within the Manage Transit function and contains analog information from which sensors can produce an image of the transit user when a fare transaction violation has been detected at the roadside. The size estimate represents a digitized equivalent of the analog image .

Traveler => **Remote Traveler Support**

Physical Architecture Flow Name: traveler request

Request by a traveler to summon assistance, request information, make a reservation, or initiate any other traveler service.

Logical Architecture Reference Flow: ft_extra_trip_data
This data flow is sent from the traveler to the Provide Driver and Traveler Services function and contains analog data from which sensors can determine extra trip request data to supplement that already input by a traveler at a kiosk. Examples of these inputs are speech, signals from relays driven by switches, buttons, etc., or input from touch screens.

Logical Architecture Reference Flow: ft_remote_emergency_request
This data flow is sent from the traveler to the Provide Driver and Traveler Services function. It contains image data from which sensors can determine that a traveler has an emergency situation that has given rise to an input from a kiosk or other device. This data is used by a process at the kiosk or other device to determine the nature of the emergency.

Logical Architecture Reference Flow: ft_trip_planning_requests
This data flow is sent from the traveler to the Provide Driver and Traveler Services function and contains analog inputs from which a traveler's trip request to a kiosk may be determined. Examples of these inputs are speech, signals from relays driven by switches, buttons, etc., or input from touch screens.

2.15.4 Subsystem Architecture Flow Diagram

Remote Traveler Subsystem (RTS)

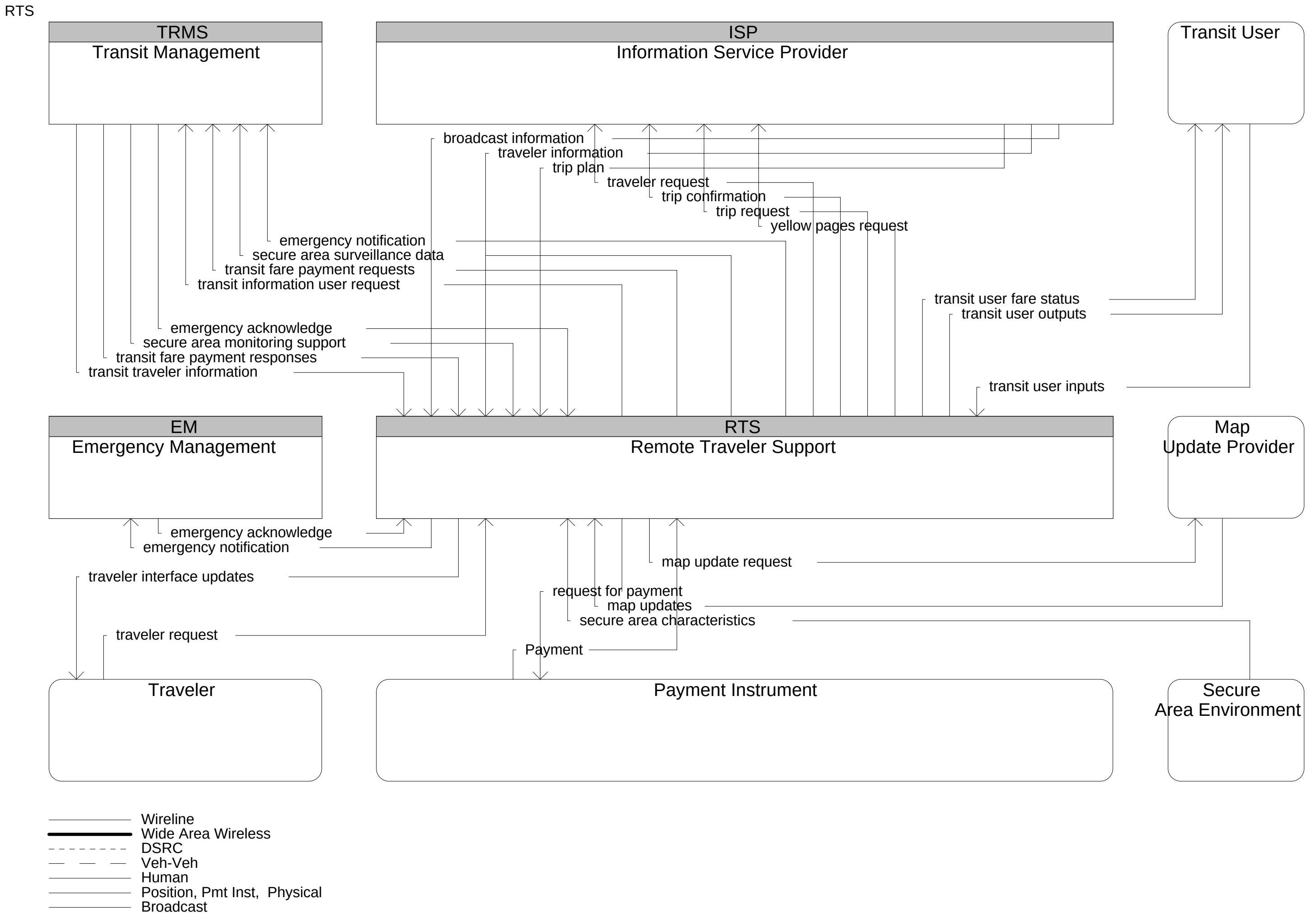


Figure 2.15-2 Architecture Flow Diagram for RTS

2.16 Roadway Subsystem

This subsystem includes the equipment distributed on and along the roadway which monitors and controls traffic. Equipment includes highway advisory radios, variable message signs, cellular call boxes, CCTV cameras and video image processing systems for incident detection and verification, vehicle detectors, traffic signals, grade crossing warning systems, and freeway ramp metering systems. This subsystem also provides the capability for emissions and environmental condition monitoring including weather sensors, pavement icing sensors, fog etc. HOV lane management and reversible lane management functions are also available. In advanced implementations, this subsystem supports automated vehicle safety systems by safely controlling access to and egress from an Automated Highway System through monitoring of, and communications with, AHS vehicles. Intersection collision avoidance functions are provided by determining the probability of a collision in the intersection and sending appropriate warnings and/or control actions to the approaching vehicles.

2.16.1 Alternative Configurations

The roadside contains all of the traffic sensors and controls associated with a traffic management center Figure 2.16-1. These devices interface with vehicles, drivers, pedestrians, and potentially with multi-modal crossings such as rail crossings. Roadway subsystem devices typically communicate with a single traffic management subsystem. Coordination between devices is handled either within and between TMS subsystems or locally at the roadside.

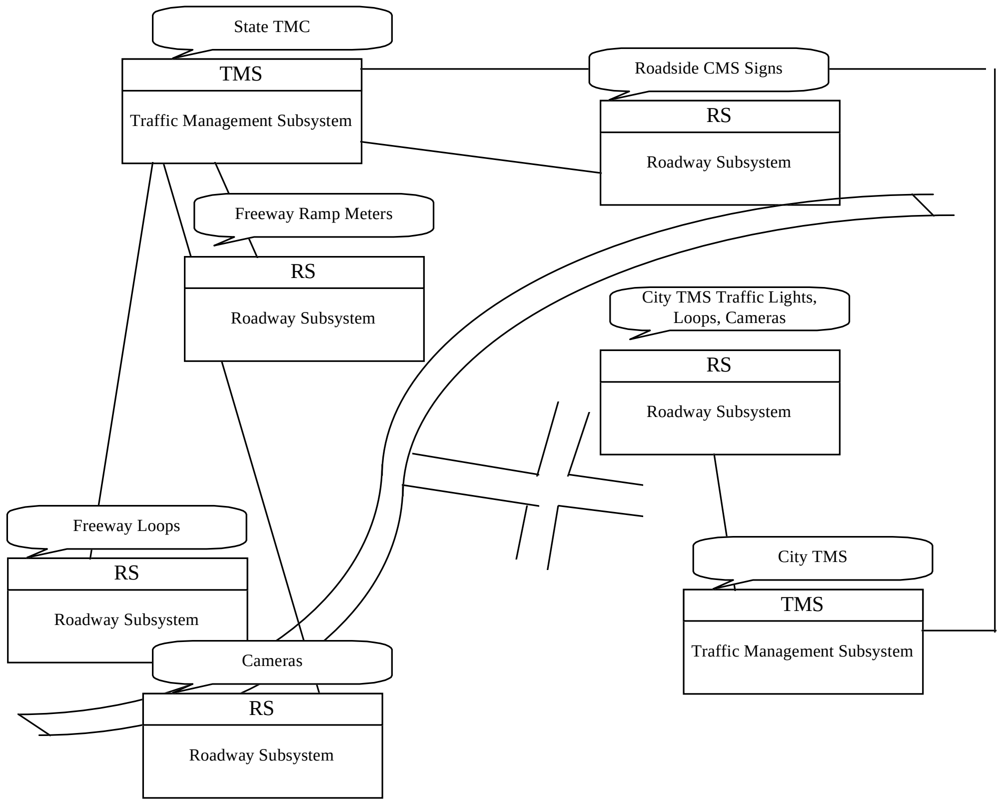


Figure 2.16-1 Alternative Configurations for the Roadway Subsystem

2.16.2 Subsystem Equipment Packages and Process Specifications for RS

Advanced Rail Crossing
This equipment package manages highway traffic at highway-rail intersections (HRIs) where operational requirements demand advanced features (e.g., where rail operational speeds are greater than 80 miles per hour). It includes all capabilities from the Standard Rail Crossing equipment package and augments these with additional safety features. The active warning systems supported by this market package includes positive barrier systems which preclude entrance into the intersection when the barriers are activated. Like the Standard package, the HRI equipment is activated on notification by wayside interface equipment which detects, or communicates with the approaching train. In this equipment package, additional information about the arriving train is also provided by the wayside interface equipment so that the train's direction of travel, its estimated time of arrival, and the estimated duration of closure may be derived. This enhanced information may be conveyed to the driver prior to, or in context with, warning system activation. This equipment package also includes detection capabilities which enable it to detect an entrapped or otherwise immobilized vehicle within the HRI and provide an immediate notification to the wayside interface equipment and traffic management.

Process Specifications
1.1.1.1 Process Traffic Sensor Data

Overview: This process shall be responsible for collecting surveillance obtained from the roadside, vehicles, pedestrians (travelers using other modes of transport), railroad grade and multimodal crossings. The process shall be able to receive both passive (e.g. presence) and active (e.g. tag) data from vehicles. Where any of the data is provided in analog form, the process shall be responsible for converting it into digital form and calibrating. The converted data shall be sent to other processes for distribution, further analysis and storage.

1.6.1.1 Detect Roadway Events

Overview: This process is responsible for monitoring local sensor data obtained from traffic surveillance and then determining and reporting the current state of all traffic in the HRI vicinity. The process provides triggers for other processes within Manage HRI Vehicle Traffic. It also monitors the device controls as they are initiated by the Activate HRI Device Controls process.

1.6.1.2.4 Provide HSR Device Controls

Overview: This process is responsible for initiating the activation of HRI devices, barriers and other special safety features for High Speed Rail at active vehicular and pedestrian grade crossings. This process responds to requests sent by the Control HRI Traffic Signals process based on detection of an oncoming train. This process sends command information to the Manage Device Control containing control signals and commands that are unique to the HSR functions, such as trapped vehicle detection. State information is also sent to the Maintain Device State process to monitor the last known state of the controls commands being processed.

1.6.1.4.1 Generate Alerts and Advisories

Overview: This process is responsible for generating the messages to advise and protect motorists, travelers and train crews approaching and crossing railroad grade crossings. Based on the severity of the hazard condition sent by the Detect HRI Hazards process this process will either send an hri_advisory command for non-time critical data or an hri_alert command for time critical data to the Report Alerts and Advisories. These users that will receive these messages include drivers, bicyclists, and pedestrians.

1.6.1.4.3 Report Alerts and Advisories

Overview: This process is responsible for reporting real-time HRI vehicle traffic advisories and real-time highway traffic alerts. Depending on the input received from the Generate Alerts and Advisories process, this process sends alerts or advisories to a train to describe the operational status of the intersection and alerts about any hazards. This process also sends the commands to Output Control Data for Roads process that will control the variable message signs in the area of an HRI to display the appropriate alert or advisory. Messages for local beacon broadcast are processed and sent to the Report HRI Status on Approach process.

1.6.1.5 Detect HRI Hazards

Overview: This process is responsible for detecting real-time HRI blockages or collisions in the vicinity of an HRI that create a blockage or other hazard at the HRI. Based upon information received from the Provide Advance Warnings process this process can send a request to the Control Vehicle Traffic at Active HRI that the local signal strategy be preempted. A hazard condition message can also be sent to the Generate Alerts and Advisories process for further action or the Provide Closures Parameters process to possibly adjust the time to closing.

1.6.1.6.1 Close HRI on Detection

Overview: This process is responsible for protecting highway vehicles approaching and crossing railroad grade crossings by initiating the closure up to 3 minutes before train arrival. This process receives the near term status of the crossing including any approaching trains or trapped vehicles. With this information along with the local control plan data the predicted hri state is computed and sent to the Detect Imminent Vehicle/Train Collision process. If a hri_predicted_collision message is returned then this process sends out an hri_hazard message to the Detect HRI Hazard which will in turn result in a change to the device control strategy. This process also receives rail operations advisories for processing along with the state and control plan data. As needed this process will output any rail_operations_message data to the Interact with Rail Operations process.

1.6.1.6.2 Detect Imminent Vehicle/Train Collision

Overview: This process is responsible for detecting imminent collisions between vehicles and trains at railroad grade crossings. Using the data contained in the predicted_hri_state message this process performs the necessary calculations to determine whether a collision is imminent. If so, this process returns a hri_predicted_collision message to the Close_HRI_on_Detection process.

1.6.3.1 Interact with Wayside Systems

Overview: This process is responsible for interfacing to railroad owned and maintained wayside equipment, such as Wayside Interface Units, Crossing Gate Controllers, etc. All these devices are expected to provide real-time information to the HRI about approaching trains and their own health. In addition, advanced implementations will make use of a communications path back to approaching trains provided by the railroad's equipment.

1.6.3.1 Interact with Wayside Systems

Overview: This process is responsible for interfacing to railroad owned and maintained wayside equipment’s, such as Wayside Interface Units, Crossing Gate Controllers, etc. All these devices are expected to provide real-time information to the HRI about approaching trains and their own health. In addition, advanced implementations will make use of a communications path back to approaching trains provided by the railroad's equipment.

1.6.3.2 Advise and Protect Train Crews

Overview: This process is responsible for generating advisories/ alerts that are routed to the wayside equipment for transmission to the train crews. If the intersection is blocked, or there is an incident at the intersection this information will be passed to the Interact with Wayside Systems process for routing to the wayside equipment. The wayside equipment can then route the information directly to the train crews, or to rail operations.

1.6.3.3 Provide ATS Alerts

Overview: This process is responsible for automatically protecting commuter, intercity, transit and freight trains as they approach and cross grade crossings. It also reports HRI rail traffic advisories to traffic management and rail operations. It is responsible for verifying and reporting overall HRI status to approaching trains so that crews can act within safe service braking distances. It provides for notification of Automatic Train Stop systems (ATS, PTS, etc) with sufficient advance warning to allow emergency brake application time to stop a train before it encounters an HRI hazard. Finally, it provides automatic status indications about the HRI to the crews of approaching trains.

1.6.5.1 Provide Interactive Interface

Overview: This process is responsible for initiating reports of the health status of the HRI to both Traffic Management and Rail Operations. In addition the process initiates reporting of the health status of the HRI to the wayside interface equipment (and ultimately to the train when the advanced HRI functionality is in place).

Automated Road Signing
Roadside beacons which may be locally and autonomously controlled from probe transmissions or centrally controlled from the virtual TMC

Process Specifications
1.1.2.6 Process Collected Vehicle Smart Probe Data

Overview: This process shall be responsible for the processing of vehicle smart probe data. The process receives data from vehicle subsystems and processes the data to estimate type and level of road conditions and hazards. The process shall send the road condition and hazard estimates to the Provide Device Control facility for output to future passing vehicles. It shall send this data, together with the fixed unit identity and fixed location to the traffic data storage process for loading into the current and long term data stores.

1.1.7 Collect Vehicle Smart Probe Data

Overview: This process shall collect data from vehicle smart probes. This data shall include information about conditions in the vicinity of the vehicle operating as a smart probe. It shall receive this data from passing vehicles and shall add its own identity and location before sending the data on to the process which outputs the data.

1.2.7.4 Process In-vehicle Signage Data

Overview: This process shall output data for use by in-vehicle signage equipment on vehicles traveling along the road (surface street) and freeway network served by the Manage Traffic function. This data shall be able to provide information from any of the types of indicators that are supported by the function, e.g. intersection controller, pedestrian controller, dynamic message sign (dms), plus data about incidents and link information such as speed, travel times or roadway conditions. The process shall be responsible for its own fault monitoring, which shall check that output data is being sent and that it is an accurate representation of the input data. When a fault is detected this process shall report it to the process responsible for the monitoring of roadside equipment faults.

1.2.7.7 Process Vehicle Smart Probe Data for Output

Overview: This process shall output data about the conditions on roads and freeways. The process shall be provided with this data by other processes in the Manage Traffic function, which have received and processed data output by smart probes in vehicles. The data shall be output by the process for reception by those vehicles that are passing the deployed instance of this process (e.g. by dedicated short range communications). The process shall perform its own fault detection and report faults that are found to the fault monitoring process.

Roadside Signal Priority
This Equipment package shall provide the capability to receive vehicle signal priority requests and control roadside signals accordingly.

Process Specifications
1.2.7.1 Process Indicator Output Data for Roads

Overview: This process shall implement the indicator output data generated by other processes within the Manage Traffic function for use on the roads (surface streets) served by the function. It shall perform the functions needed to provide control at intersections or pedestrian crossings, generate the output for dynamic message signs (dms) and highway advisory radios (HAR), or provide the interface for data to be sent to the units (or systems) that manage multimodal crossings. The dms may be either those that display variable text messages, or those that have fixed format display(s)(e.g. vehicle restrictions, or lane open/close).

1.2.7.3 Manage Indicator Preemptions

Overview: This process shall receive indicator (e.g. signal) preemption and priority requests from other functions within ITS. These requests shall enable the process to give selected vehicles (e.g. those that belong to Transit Authorities or Emergency Services) signal preemption or priority at intersections, pedestrian crossings and multimodal crossings in the road (surface street) and freeway network served by the instance of the Manage Traffic function. Sending of the priority request output shall also generate an output to the monitoring process to suspend its activities while the priority request is being served. This process shall only generate its data flow outputs when input data is received.

Roadway Basic Surveillance
This Equipment package provides the capabilities to monitor traffic flow in major intersections and on main highways for urban areas and to monitor road conditions using fixed equipment such as loop detectors and wireline communication.

Process Specifications
1.1.1.1 Process Traffic Sensor Data

Overview: This process shall be responsible for collecting surveillance obtained from the roadside, vehicles, pedestrians (travelers using other modes of transport), railroad grade and multimodal crossings. The process shall be able to receive both passive (e.g. presence) and active (e.g. tag) data from vehicles. Where any of the data is provided in analog form, the process shall be responsible for converting it into digital form and calibrating. The converted data shall be sent to other processes for distribution, further analysis and storage.

1.3.1.3 Process Traffic Images

Overview: This process shall process raw traffic image data received from sensors located on the road (surface street) and freeway network served by the Manage Traffic function. The process shall transform the raw data into images that can be sent to another process within the Manage Incidents facility. It shall also act as the control interface through which the images of traffic conditions which are analyzed for incidents can be changed by the traffic operations personnel, who shall also be supplied with images for viewing.

Roadway Emissions Monitoring
This Equipment package monitors emissions and general air quality and communicates the collected information back to the emissions management subsystem where it can be monitored, analyzed, and used. This equipment package supports point monitoring of individual vehicle emissions as well as general monitoring of standard air quality measures.

Process Specifications
1.5.5 Process Vehicle Pollution Data

Overview: This process shall obtain pollution data about individual vehicles and analyze it against reference data obtained from another process within the Manage Emissions facility of the Manage Traffic function. The process shall use this reference data to determine whether or not a vehicle is possibly violating the acceptable levels of pollution output. When the process determines that a possible violation has occurred, it shall send the detected pollution levels and the vehicle identity to the process responsible for law enforcement in the Manage Emergency Services function for action.

1.5.6 Detect Roadside Pollution Levels

Overview: This process shall process the local area pollution data analyzed by sensors looking at the levels of pollution at the roadside within the geographic area served by the Manage Traffic function. The process shall pass the data on to another process within the Manage Emissions facility for integration with wide area pollution data and comparison with thresholds for pollution incidents.

Roadway Environmental Monitoring
This Equipment package measures environmental conditions and communicates the collected information back to a center where it can be monitored and analyzed. A broad array of general weather and road surface information may be collected. Weather conditions that may be measured include temperature, wind, humidity, precipitation, and visibility. Surface and sub-surface sensors can measure road surface temperature, moisture, icing, salinity, and other measures. Air quality monitoring can include point monitoring of individual vehicles as well as general monitoring of standard air quality measures.

Process Specifications
1.1.1.3 Process Environmental Sensor Data

Overview: This process shall be responsible for collecting data obtained from environmental sensors. Where any of the data is provided in analog form, the process shall be responsible for converting it into digital form and calibrating. The converted data shall be sent to other processes for distribution, further analysis and storage.

Roadway Freeway Control
Ramp meters, CMS and other freeway control effectors which will control traffic on freeways.

Process Specifications
1.1.1.1 Process Traffic Sensor Data

Overview: This process shall be responsible for collecting surveillance obtained from the roadside, vehicles, pedestrians (travelers using other modes of transport), railroad grade and multimodal crossings. The process shall be able to receive both passive (e.g. presence) and active (e.g. tag) data from vehicles. Where any of the data is provided in analog form, the process shall be responsible for converting it into digital form and calibrating. The converted data shall be sent to other processes for distribution, further analysis and storage.

1.2.7.2 Monitor Roadside Equipment Operation for Faults

Overview: This process shall monitor the operation of the processes that output in-vehicle signage, highway advisory radio, as well as indicator data in the road (surface street) and freeway network. It shall report any instances where the indicator response does not match that expected from the contents of the indicator control data it is receiving, the in-vehicle signage process reports a fault, or the HAR processes report a fault.

1.2.7.5 Process Indicator Output Data for Freeways

Overview: This process shall implement the indicator output data generated by other processes within the Manage Traffic function for use on freeways served by the function. It shall perform the functions needed to output control data to ramp metering controllers and multimodal crossings, generate the output for dynamic message signs (dms), or generate the output for highway advisory radios(HAR). The dms may be either those that display variable text messages, or those that have fixed format display(s), for such things as vehicle restrictions, or lane open/close.

Roadway HOV Control

This Equipment package provides the capability to detect the HOV lane usage using sensor equipment. For lanes that become HOV or High Occupancy Toll (HOT) lanes during certain time of the day, it provides display equipment to notify users of their status.

Process Specifications

1.1.1.1 Process Traffic Sensor Data

Overview: This process shall be responsible for collecting surveillance obtained from the roadside, vehicles, pedestrians (travelers using other modes of transport), railroad grade and multimodal crossings. The process shall be able to receive both passive (e.g. presence) and active (e.g. tag) data from vehicles. Where any of the data is provided in analog form, the process shall be responsible for converting it into digital form and calibrating. The converted data shall be sent to other processes for distribution, further analysis and storage.

1.2.7.5 Process Indicator Output Data for Freeways

Overview: This process shall implement the indicator output data generated by other processes within the Manage Traffic function for use on freeways served by the function. It shall perform the functions needed to output control data to ramp metering controllers and multimodal crossings, generate the output for dynamic message signs (dms), or generate the output for highway advisory radios(HAR). The dms may be either those that display variable text messages, or those that have fixed format display(s), for such things as vehicle restrictions, or lane open/close.

Roadway In-Vehicle Signing

This Equipment package provides the capability to detect local traffic flow conditions, corroborate them with a traffic management subsystem, and distribute them to the user over a short-range interface such as a radio beacon.

Process Specifications

1.2.7.4 Process In-vehicle Signage Data

Overview: This process shall output data for use by in-vehicle signage equipment on vehicles traveling along the road (surface street) and freeway network served by the Manage Traffic function. This data shall be able to provide information from any of the types of indicators that are supported by the function, e.g. intersection controller, pedestrian controller, dynamic message sign (dms), plus data about incidents and link information such as speed, travel times or roadway conditions. The process shall be responsible for its own fault monitoring, which shall check that output data is being sent and that it is an accurate representation of the input data. When a fault is detected this process shall report it to the process responsible for the monitoring of roadside equipment faults.

Roadway Incident Detection

This Equipment package provides incident detection capability to reside at the roadside. For example, advanced CCTV's with built-in incident detection algorithms would allow the actual detection function to be roadside rather than transmitting images to a center for visual or automated detection.

Process Specifications

1.1.1.1 Process Traffic Sensor Data

Overview: This process shall be responsible for collecting surveillance obtained from the roadside, vehicles, pedestrians (travelers using other modes of transport), railroad grade and multimodal crossings. The process shall be able to receive both passive (e.g. presence) and active (e.g. tag) data from vehicles. Where any of the data is provided in analog form, the process shall be responsible for converting it into digital form and calibrating. The converted data shall be sent to other processes for distribution, further analysis and storage.

Roadway Intersection Collision Warning
This Equipment package provides the capability to determine the probability of a collision in the intersection and send appropriate warnings and/or control actions to the approaching vehicles using a short-range interface. This Equipment package also provides the capability that the traffic control signals provide signal indication information to the vehicles using a short-range interface and the vehicle performs the determination of the probability of collision in the intersection. This package covers intersections between vehicles and railroad at grade crossings.

Process Specifications
1.2.7.6 Provide Intersection Collision Avoidance Data

Overview: This process shall provide collision avoidance data to vehicles that are approaching intersections served by the Manage Traffic function. The process shall use the data available from traffic sensors to determine any vehicle position conflict(s) that will arise if no action is taken. This process shall output data giving the direction from which the potential collision hazard will arise to the vehicle(s) that is(are) likely to receive the impact.

Roadway Probe Beacons
This Equipment package provides the capabilities to monitor traffic flow in major intersections and on main highways for urban areas and to monitor road conditions using mobile equipment and wireless communication. For example, vehicle probe data or aerial surveillance data.

Process Specifications
1.1.6 Collect Vehicle Tag Data for Link Time Calculations

Overview: This process shall collect data from toll and parking tags on passing vehicles. This shall be achieved by transmitting a tag data request message and collecting any tag reply data that is received. This reply data shall be translated by the process into a unique but anonymous ID that does not store or transmit the identity of the tag in any way that is traceable to the tag owner, e.g. credit identity or stored credit value. This ID is then passed on to another process for further link travel time calculation analysis.

Roadway Reversible Lanes
This Equipment package provides the capability for control of reversible lanes using sensor and actuator type equipment. This Equipment package also provides the capability to notify users the direction of the reversible lanes using electronic lane signs.

Process Specifications
1.1.1.1 Process Traffic Sensor Data

Overview: This process shall be responsible for collecting surveillance obtained from the roadside, vehicles, pedestrians (travelers using other modes of transport), railroad grade and multimodal crossings. The process shall be able to receive both passive (e.g. presence) and active (e.g. tag) data from vehicles. Where any of the data is provided in analog form, the process shall be responsible for converting it into digital form and calibrating. The converted data shall be sent to other processes for distribution, further analysis and storage.

1.2.7.1 Process Indicator Output Data for Roads

Overview: This process shall implement the indicator output data generated by other processes within the Manage Traffic function for use on the roads (surface streets) served by the function. It shall perform the functions needed to provide control at intersections or pedestrian crossings, generate the output for dynamic message signs (dms) and highway advisory radios (HAR), or provide the interface for data to be sent to the units (or systems) that manage multimodal crossings. The dms may be either those that display variable text messages, or those that have fixed format display(s)(e.g. vehicle restrictions, or lane open/close).

1.2.7.5 Process Indicator Output Data for Freeways

Overview: This process shall implement the indicator output data generated by other processes within the Manage Traffic function for use on freeways served by the function. It shall perform the functions needed to output control data to ramp metering controllers and multimodal crossings, generate the output for dynamic message signs (dms), or generate the output for highway advisory radios(HAR). The dms may be either those that display variable text messages, or those that have fixed format display(s), for such things as vehicle restrictions, or lane open/close.

Roadway Signal Controls

This Equipment package provides the capabilities to control traffic signals at major intersections and on main highways for urban areas. This Equipment package is generally constrained to a single jurisdiction.

Process Specifications

1.1.1.1 Process Traffic Sensor Data

Overview: This process shall be responsible for collecting surveillance obtained from the roadside, vehicles, pedestrians (travelers using other modes of transport), railroad grade and multimodal crossings. The process shall be able to receive both passive (e.g. presence) and active (e.g. tag) data from vehicles. Where any of the data is provided in analog form, the process shall be responsible for converting it into digital form and calibrating. The converted data shall be sent to other processes for distribution, further analysis and storage.

1.2.7.1 Process Indicator Output Data for Roads

Overview: This process shall implement the indicator output data generated by other processes within the Manage Traffic function for use on the roads (surface streets) served by the function. It shall perform the functions needed to provide control at intersections or pedestrian crossings, generate the output for dynamic message signs (dms) and highway advisory radios (HAR), or provide the interface for data to be sent to the units (or systems) that manage multimodal crossings. The dms may be either those that display variable text messages, or those that have fixed format display(s)(e.g. vehicle restrictions, or lane open/close).

1.2.7.2 Monitor Roadside Equipment Operation for Faults

Overview: This process shall monitor the operation of the processes that output in-vehicle signage, highway advisory radio, as well as indicator data in the road (surface street) and freeway network. It shall report any instances where the indicator response does not match that expected from the contents of the indicator control data it is receiving, the in-vehicle signage process reports a fault, or the HAR processes report a fault.

Roadway Systems for AHS

This Equipment package provides the capability of safely controlling access to and egress from an Automated Highway System. This Equipment package also provides the capability for roadside to vehicle communication. These capabilities shall be provided using equipment such as a lane check-in or check-out beacon and special purpose vehicle signing beacons.

Process Specifications

3.2.5 Check Vehicle for AHS eligibility

Overview: This process shall be responsible for checking that vehicles are eligible for using the automated highway system (ahs) lanes on a highway. The process shall decide whether or not the vehicle is suitable for has operation by checking locally stored data that has been provided by a process in the Manage Traffic function, against data from the vehicle provided through the check request by a process on-board the vehicle. The process shall send the results of the check to the process on-board the vehicle that requested the ahs check-in. The vehicles that are successfully checked-in shall also be down loaded with ahs control data from this process.

3.2.6 Manage AHS Check-in and Check-out

Overview: This process shall be responsible for managing the checking in and checking out of suitably equipped vehicles requesting to use automated highway system (ahs) lanes. The process shall provide the special vehicle control parameters needed for ahs operation to the process that manages ahs check-in and collect data on vehicles that request check-in and check-out from that process. This process shall send a record of all check-in and check-out transactions regardless of whether they are successful or not, to the process responsible for managing ahs operational data.

Roadway Traffic Information Dissemination
This Equipment package provides the roadside elements of traffic information dissemination including DMS and HAR.

Process Specifications
1.2.7.1 Process Indicator Output Data for Roads

Overview: This process shall implement the indicator output data generated by other processes within the Manage Traffic function for use on the roads (surface streets) served by the function. It shall perform the functions needed to provide control at intersections or pedestrian crossings, generate the output for dynamic message signs (dms) and highway advisory radios (HAR), or provide the interface for data to be sent to the units (or systems) that manage multimodal crossings. The dms may be either those that display variable text messages, or those that have fixed format display(s)(e.g. vehicle restrictions, or lane open/close).

1.2.7.5 Process Indicator Output Data for Freeways

Overview: This process shall implement the indicator output data generated by other processes within the Manage Traffic function for use on freeways served by the function. It shall perform the functions needed to output control data to ramp metering controllers and multimodal crossings, generate the output for dynamic message signs (dms), or generate the output for highway advisory radios(HAR). The dms may be either those that display variable text messages, or those that have fixed format display(s), for such things as vehicle restrictions, or lane open/close.

Standard Rail Crossing
This Equipment Package manages highway traffic at highway-rail intersections (HRIs) where operational requirements do not dictate advanced features (e.g., where rail operational speeds are less than 80 miles per hour). Either passive (e.g., the crossbuck sign) or active warning systems (e.g., flashing lights and gates) are supported depending on the specific requirements for each intersection. These traditional HRI warning systems may also be augmented with other standard traffic management devices. The warning systems are activated on notification by interfaced wayside equipment of an approaching train. The equipment at the HRI may also be interconnected with adjacent signalized intersections so that local control can be adapted to highway-rail intersection activities. Health monitoring of the HRI equipment and interfaces is performed; detected abnormalities are reported through interfaces to the wayside interface equipment and the traffic management subsystem.

Process Specifications
1.1.1.1 Process Traffic Sensor Data

Overview: This process shall be responsible for collecting surveillance obtained from the roadside, vehicles, pedestrians (travelers using other modes of transport), railroad grade and multimodal crossings. The process shall be able to receive both passive (e.g. presence) and active (e.g. tag) data from vehicles. Where any of the data is provided in analog form, the process shall be responsible for converting it into digital form and calibrating. The converted data shall be sent to other processes for distribution, further analysis and storage.

1.2.7.1 Process Indicator Output Data for Roads

Overview: This process shall implement the indicator output data generated by other processes within the Manage Traffic function for use on the roads (surface streets) served by the function. It shall perform the functions needed to provide control at intersections or pedestrian crossings, generate the output for dynamic message signs (dms) and highway advisory radios (HAR), or provide the interface for data to be sent to the units (or systems) that manage multimodal crossings. The dms may be either those that display variable text messages, or those that have fixed format display(s)(e.g. vehicle restrictions, or lane open/close).

1.6.1.2.1 Control HRI Traffic Signals

Overview: This process is responsible for interpreting the hri_control message and safely directing the activation of the appropriate devices. This process will both directly command devices at the HRI and will disseminate necessary control information to the Process Indicator Output Data for Roads function to allow integrated control of adjacent traffic signals. Data will also be sent to SSR and/or HSR Device Control functions to control these specialized devices at the crossing. When sensor data indicates an approaching train this process notifies the Process Indicator Output Data for Roads function to allow the signal timing to be adjusted and variable message signs, if available, to be updated. This allows the traffic signals in the area adjacent to an HRI to be used to clear the Storage Area in advance of an approaching train and to manage traffic around the intersection.

1.6.1.2.2 Control HRI Warnings and Barriers

Overview: This process is responsible for initiating the activation of HRI barriers at active vehicular and pedestrian grade crossings. When a request is sent to activate the HRI barriers perhaps because of a detection of an oncoming train, this process sends the device control signal to the Manage Device Controls process to activate the barriers. This process also returns state information to the Maintain Device State process concerning the commands that have been initiated by this process.

1.6.1.2.3 Provide SSR Device Controls

Overview: This process is responsible for initiating the activation of HRI Standard Speed Rail control devices at active vehicular and pedestrian grade crossings. This process responds to requests sent by the Control HRI Traffic Signals process based on detection of an oncoming train. This process sends command information to the Manage Device Control containing control signals and commands that are unique to the SSR functions. State information is also sent to the Maintain Device State process to monitor the last known state of the controls commands being processed.

1.6.1.2.5 Manage Device Control

Overview: This process is responsible for managing and selecting the appropriate device control messages. This process gathers the control signals from the other Activate HRI Device Control processes and forwards them as needed to the Process Indicator Output Data for Roads process within Provide Device Control. These control signals are used to activate all of the HRI unique roadside devices such as gates or other barriers, lights, adjacent traffic signals, message signs or in-vehicle signage beacons.

1.6.1.2.6 Maintain Device State

Overview: This process is responsible for managing and selecting the appropriate device control state messages. This process collects the device state messages that are produced by the other Activate HRI Device Controls processes and forwards the appropriate signals to the Detect Roadway Events process that monitors the status of the HRI commands being processed. This information is also used in the equipment diagnostic monitoring and testing.

1.6.1.3 Perform Equipment Self-Test

Overview: This process is responsible for performing real-time equipment checks and reporting the status of the equipment associated with an active grade crossing. Based on receipt of the sensor data of the surrounding highway and rail traffic and receipt of any near term events this process can execute a real-time check of the equipment and determine the relative health and status of the active grade crossing equipment. The output is sent onto the Monitor HRI Status process for further processing with other diagnostic data.

1.6.1.4.2 Provide Closure Parameters

Overview: This process is responsible for providing the HRI predicted time to closure to be used in broadcast message alerts to approaching vehicles. This time is calculated from data provided by the Detect HRI Hazards process.

1.6.1.4.4 Report HRI Status on Approach

Overview: This process is responsible for providing real-time HRI status to vehicles as they approach an HRI. It must discriminate between vehicles near, but not approaching, the HRI (e.g. on parallel side streets, etc.). This process develops the message to be broadcast to nearby vehicles by receiving time_to_closing data and the hazard_condition signal and calculating the appropriate window of time to display the message. The message is built from the approach_warning data received from the Report Alerts and Advisories process.

1.6.1.7.1 Control Vehicle Traffic at Passive HRI

Overview: This process is responsible for controlling vehicle traffic at passive grade crossings. It provides a mechanism for rail operations to close grade crossings that have active traffic devices but no real-time train detection mechanisms. This process also will allow for a train crew member to manually activate closure of the crossing. In such an event a crew_close_hri signal is sent to the Close_HRI_on_Command process.

1.6.1.7.2 Control Vehicle Traffic at Active HRI

Overview: This process is responsible for controlling vehicular traffic at an active HRI by controlling the operation of traffic control devices in accordance with a predetermined local control plan. The local_control_plan is communicated to the Close_HRI_on_Detection process. This local control plan can be preempted by a strategy_preemption message from the Detect_HRI_Hazards process or by such inputs as an event_notice from the Detect_Roadway_Events process or hri_traffic_surveillance data. The outputs of this process include the command messages to close the HRI, requests for information from the Manage Traffic function, and information about the current hri_traffic_data.

1.6.1.7.3 Close HRI on Command

Overview: This process is responsible for closing the HRI to vehicular traffic, either on command from the Control Vehicle traffic at Active HRI process, or from direct command from rail operations (as an override). A third mechanism for closing the HRI is defined for passive crossings, i.e. crossings without active train detection systems. Upon command from rail operations, or via manual operation by a train crewman, active traffic devices at an otherwise passive grade crossing may be activated to close the crossing *.

1.6.3.1 Interact with Wayside Systems

Overview: This process is responsible for interfacing to railroad owned and maintained wayside equipment, such as Wayside Interface Units, Crossing Gate Controllers, etc. All these devices are expected to provide real-time information to the HRI about approaching trains and their own health. In addition, advanced implementations will make use of a communications path back to approaching trains provided by the railroad's equipment.

1.6.5.2 Determine HRI Status

Overview: This process is responsible for monitoring critical HRI functions and merging them into a single coherent picture of the state of the hri. It also is responsible for assuring that the HRI always reverts to the safest possible operating condition in the event of any operational malfunctions.

1.6.5.3 Maintain HRI Closure Data

Overview: This process is responsible for managing a log of the HRI operation for use in strategy planning, demand management and traffic management.

2.16.3 Subsystem Interfaces for RS

Emergency Vehicle Subsystem => Roadway Subsystem

Physical Architecture Flow Name: local signal preemption request

Direct control signal or message to a signalized intersection that results in preemption of the current control plan and grants right-of-way to the requesting vehicle.

Logical Architecture Reference Flow: emergency_vehicle_preemptions

This data flow is sent by the Manage Emergency Services function to the Manage Traffic function. It contains the data necessary for an individual emergency services vehicle and the stream of traffic in which the vehicle is traveling to be given preemption (priority) at an indicator controller. This will be at the controller for a particular road junction, pedestrian crossing, or highway entrance ramp. The data is sent directly from the emergency vehicle to the next controller along its route and therefore is not subject to any centralized coordination. Local coordination may be provided if there are links between adjacent controllers. The data flow consists of the following data items each of which is defined in its own DDE: emergency_vehicle_junction_preemption + emergency_vehicle_pedestrian_preemption + emergency_vehicle_ramp_preemption + emergency_vehicle_sign_preemption.

Emissions Management => **Roadway Subsystem**

Physical Architecture Flow Name: vehicle pollution criteria

Vehicular pollution acceptance criteria.

Logical Architecture Reference Flow: pollution_state_vehicle_acceptance_criteria
This data flow is used within the Manage Traffic function and contains data on the pollution levels produced by different types of vehicle under various operating conditions. It consists of the following items of data each of which is defined in its own DDE: pollution_vehicle_acceptance_data + pollution_vehicle_acceptance_conditions.

Environment => **Roadway Subsystem**

Physical Architecture Flow Name: pollution data

Measured emissions data comprised of various atmospheric pollutants.

Logical Architecture Reference Flow: fe_roadside_pollutant_levels
This data flow is used within the Manage Traffic function. It contains analog data from which sensors can determine the actual levels of various atmospheric pollutants, such as nitrous oxide, sulfur dioxide, hydrocarbons, carbon monoxide and ozone, that are generally present at the roadside. The sizing assumption below is based on an estimate of the digitized equivalent information.

Multimodal Crossings => **Roadway Subsystem**

Physical Architecture Flow Name: multimodal crossing status

Indication of operational status and pending requests for right-of-way from equipment supporting the non-highway mode at multimodal crossings.

Logical Architecture Reference Flow: fmmc_crossing_close_duration
This data flow is sent to the Provide Traffic Surveillance facility within the Manage Traffic function from multimodal crossings. It contains the time duration for which a crossing must close to vehicular traffic to permit the passage of the alternate flow, e.g. river traffic, aircraft, etc.

Logical Architecture Reference Flow: fmmc_crossing_close_time
This data flow is sent to the Provide Traffic Surveillance facility within the Manage Traffic function from multimodal crossings. It contains the time period before a crossing must close to vehicular traffic to permit the passage of the alternate flow, e.g. river traffic, aircraft, etc.

Logical Architecture Reference Flow: fmmc_crossing_status_for_highways
This data flow allows multimodal crossing equipment to provide its operational status to control processes on the

roadside. This can be a as simple as a binary indication of status all the way to a full maintenance report.

Logical Architecture Reference Flow: fmmc_crossing_status_for_roads
This data flow allows multimodal crossing equipment to provide its operational status to control processes on the roadside. This can be a as simple as a binary indication of status all the way to a full maintenance report.

Pedestrians => **Roadway Subsystem**

Physical Architecture Flow Name: crossing call

Request for pedestrian crossing.

Logical Architecture Reference Flow: fp_pedestrian_data
This data flow is used within the Manage Traffic function. It contains analog data about the presence of pedestrians waiting to cross, or approaching the crossing points of roads and highways from which pedestrian surveillance data such as pedestrian demand, numbers of pedestrians, etc. can be obtained by sensors within ITS.

Logical Architecture Reference Flow: fp_pedestrian_images
This data flow is used within the Manage Traffic function. It contains visual information (analog data) about pedestrians waiting to cross, or approaching the crossing points of roads and highways from which pedestrian surveillance data can be obtained by image processors within ITS. The size estimate is representative of a digitization of this analog data.

Roadway Environment => **Roadway Subsystem**

Physical Architecture Flow Name: weather conditions

Collected weather condition data from sensors.

Logical Architecture Reference Flow: fre_environmental_conditions
This data flow is sent from the roadside environment to the Manage Traffic function and contains analog data. This data is used by sensors within the function to determine environmental roadside conditions such as air temperature, wind speed, humidity and precipitation, fog, ice, snow, rain, etc. that are affecting the road and highway network served by the function.

Logical Architecture Reference Flow: fre_physical_conditions
This data flow is sent from the roadside environment to the Manage Traffic function and contains analog data. This data is used by sensors within the function to determine the physical conditions such as fog, ice, snow, rain, etc. that are affecting the road and highway network served by the function.

Roadway Subsystem => **Basic Vehicle**

Physical Architecture Flow Name: broadcast advisories

General broadcast advisories that are provided over wide-area wireless communications direct to the vehicle radio. These analog advisory messages may provide similar content to ITS broadcast information flows, but include no digital data component. Existing Highway-Advisory Radio (HAR) advisory messages are a prime example of this flow.

Logical Architecture Reference Flow: tbv_har_broadcast_for_highways
This data flow contains the output of a Highway Advisory Radio (HAR) operating at the roadside on highways in the geographic and/or jurisdictional area(s) served by the function. This output, the HAR program, is broadcast to an existing communications device (e.g. AM Radio) in a vehicle.

Logical Architecture Reference Flow: tbv_har_broadcast_for_roads
This data flow contains the output of a Highway Advisory Radio (HAR) operating at the roadside on roads (surface streets) in the geographic and/or jurisdictional area(s) served by the function. This output, the HAR program, is broadcast to an existing communications device (e.g. AM Radio) in a vehicle.

Roadway Subsystem => **Driver**

Physical Architecture Flow Name: driver information

General advisory and traffic control information provided to the driver while en-route.

Logical Architecture Reference Flow: td_lane_use_indication_for_highways
This data flow is sent to the driver by the Manage Traffic function and contains an indication that a particular vehicle lane on a freeway served by the function is available for use (green) or closed to vehicle use (red).

Logical Architecture Reference Flow: td_lane_use_indication_for_roads
This data flow is sent to the driver by the Manage Traffic function and contains an indication that a particular vehicle lane on a road (surface street) served by the function is available for use (green) or closed to vehicle use (red).

Logical Architecture Reference Flow: td_ramp_state_indication
This data flow is sent to the driver by the Manage Traffic function and contains an indication that a particular highway entrance ramp is available for use (green) or closed to vehicle use (red).

Logical Architecture Reference Flow: td_signal_indication
This data flow is sent to the driver by the Manage Traffic function and contains instructions for drivers to stop their vehicles (red), prepare to stop their vehicles (amber), or that they have permission to proceed (green) along the road or highway.

Logical Architecture Reference Flow: td_vms_indication_for_highways
This data flow is sent to the driver by the Manage Traffic function and contains a textual message either warning drivers of a potential hazard, or providing mandatory instructions as to the availability of all or part of a freeway served by the function.

Logical Architecture Reference Flow: td_vms_indication_for_roads
This data flow is sent to the driver by the Manage Traffic function and contains a textual message either warning drivers of a potential hazard, or providing mandatory instructions as to the availability of all or part of a road (surface street) served by the function.

Roadway Subsystem => **Emissions Management**

Physical Architecture Flow Name: pollution data

Measured emissions data comprised of various atmospheric pollutants.

Logical Architecture Reference Flow: pollution_state_roadside_collection
This data flow is used within the Manage Traffic function and contains the digitized values of pollution levels obtained from roadside sensors in the geographic area served by the function. It consists of the following data items each of which is defined in its own DDE: current_ozone_pollution + current_nitrous_oxide_pollution + current_sulfur_dioxide_pollution + current_hydrocarbon_pollution + current_carbon_monoxide_pollution + current_particulate_pollution + current_roadside_pollution_location.

Logical Architecture Reference Flow: pollution_state_vehicle_collection
This data flow is used within the Manage Traffic function and contains the average levels of the various types of

pollution that were being output by a particular type of violating vehicle. It consists of the following data items each of which is defined in its own DDE: current_ozone_pollution + current_nitrous_oxide_pollution + current_sulfur_dioxide_pollution + current_hydrocarbon_pollution + current_carbon_monoxide_pollution + current_particulate_pollution + vehicle_type.

Logical Architecture Reference Flow: pollution_state_vehicle_log_data
This data flow is used within the Manage Traffic function and contains a periodic average of the pollution data measured by sensors from actual vehicles. It consists of the following data item which is defined in its own DDE: pollution_state_vehicle_collection.

Roadway Subsystem => **Multimodal Crossings**

Physical Architecture Flow Name: highway control status

Current traffic control equipment status that indicates operational status and right-of-way availability to the non-highway transportation mode at a multimodal crossing.

Logical Architecture Reference Flow: tmmc_crossing_clear_at_highways
This data flow is sent to the multimodal crossings from the Manage Traffic function to indicate that freeway traffic has been stopped and the crossing may be used by the other (non-road or highway) transportation system.

Logical Architecture Reference Flow: tmmc_crossing_clear_at_roads
This data flow is sent to the multimodal crossings from the Manage Traffic function to indicate that road (surface street) traffic has been stopped and the crossing may be used by the other (non-road or highway) transportation system.

Logical Architecture Reference Flow: tmmc_highway_equipment_status
This data flow is sent to the multimodal crossing from the Manage Traffic function to indicate the operational status of the roadway equipment (e.g. traffic control devices, lane closure indicators, dynamic message sing, etc).

Logical Architecture Reference Flow: tmmc_road_equipment_status
This data flow is sent to the multimodal crossing from the Manage Traffic function to indicate the operational status of the roadway equipment (e.g. traffic control devices, lane closure indicators, dynamic message sing, etc).

Logical Architecture Reference Flow: tmmc_stop_alternate_mode_at_highways
This data flow is sent to the multimodal crossings from the Manage Traffic function to indicate that the alternate mode traffic must if possible be stopped, to enable the crossing to be used by emergency vehicles. Output of this data does not guarantee that the emergency vehicle(s) will have priority, since in some cases it may be too late to stop the alternate mode traffic.

Logical Architecture Reference Flow: tmmc_stop_alternate_mode_at_roads
This data flow is sent to the multimodal crossings from the Manage Traffic function to indicate that the alternate mode traffic must if possible be stopped, to enable the crossing to be used by emergency vehicles. Output of this data does not guarantee that the emergency vehicle(s) will have priority, since in some cases it may be too late to stop the alternate mode traffic.

Roadway Subsystem => **Pedestrians**

Physical Architecture Flow Name: crossing permission

Signal to pedestrians indicating permission to cross roadway.

Logical Architecture Reference Flow: tp_cross_request_received

This data flow is sent to pedestrians by the Manage Traffic function and is an indication that a request by a pedestrian to cross the road or highway has been received and will be actioned in the future. When no requests have been received the indication must be blank or not visible to pedestrians.

Logical Architecture Reference Flow: tp_cross_road
This data flow is sent to pedestrians by the Manage Traffic function and is an indication that a pedestrian may cross the road or highway (green) or wait until instructed to cross (red).

Logical Architecture Reference Flow: tp_vms_indication
This data flow is sent to pedestrians by the Manage Traffic function and contains a textual message either warning them of a potential hazard, or providing mandatory instructions as to the availability of pedestrian access.

Roadway Subsystem => Traffic Management

Physical Architecture Flow Name: AHS status

Status of AHS equipment, lane controls etc.

Logical Architecture Reference Flow: ahs_checking_details
This data flow is used in the Provide Vehicle Monitoring and Control function and contains updates to the counts of successful and failed check-ins to the automatic highway system (ahs) lanes from roadside locations. It consists of the following data items each of which is defined in its own DDE: ahs_failed_checks_count + ahs_successful_checks_count.

Physical Architecture Flow Name: emissions data

Emissions data and associated imagery collected by roadside equipment.

Logical Architecture Reference Flow: vehicle_pollution_alert
This data flow is used by the Manage Traffic function as a means of transferring current vehicle pollution data from the Manage Emissions facility to the Manage Emergency Services function to enable enforcement of air quality standards. It contains data about the current levels of pollution being output by a vehicle. This data is held in the following data items each of which is defined in its own DDE: vehicle_identity + vehicle_license + pollution_data_violation.

Logical Architecture Reference Flow: vehicle_pollution_message_for_highways
This data flow is used within the Manage Traffic function and contains data about the current levels of pollution being output by a vehicle. It is for output to the vehicle driver who is on a freeway in the geographic and/or jurisdictional area(s) served by the function. The data flow consists of the following data items each of which is defined in its own DDE: indicator_identity + pollution_output_message.

Logical Architecture Reference Flow: vehicle_pollution_message_for_roads
This data flow is used within the Manage Traffic function and contains data about the current levels of pollution being output by a vehicle. It is for output to the vehicle driver who is on a road (surface street) in the geographic and/or jurisdictional area(s) served by the function. The data flow consists of the following data items each of which is defined in its own DDE: indicator_identity + pollution_output_message.

Physical Architecture Flow Name: environmental conditions

Current environment conditions (e.g., air temperature, wind speed, surface temperature) as measured by environmental sensors and communicated by supporting field equipment.

Logical Architecture Reference Flow: environment_sensor_data
This data flow is used within the Manage Traffic function and contains a set of outputs from individual environment sensors. It consists of the following data items each of which is defined in its own DDE: 1{station_id +

sensor_identity + environment_sensor_output}list_size.

Logical Architecture Reference Flow: environmental_sensor_status
This data flow is used within the Manage Traffic function to report the status of an environmental sensor. By monitoring this data flow, the receiving process can monitor the health and current status of field equipment. It consists of the following items each of which are defined in its own DDE: list_size+ 1{station_id+ sensor_identity}list_size.

Physical Architecture Flow Name: fault reports

Reports from field equipment (sensors, signals, signs, controllers, etc.) which indicate current operational status.

Logical Architecture Reference Flow: environment_sensor_fault_data
This data flow is used within the Manage Traffic function to show that an environment sensor has developed a fault that means it is not operating correctly. The fault will have been found by a process that is local to the sensor itself.

Logical Architecture Reference Flow: sensor_fault_data
This data flow is used within the Manage Traffic function to show that a sensor has developed a fault that means it is not operating correctly. The fault will have been found by a process that is local to the sensor itself.

Logical Architecture Reference Flow: traffic_control_device_status
This data flow is used within the Manage Traffic function to show any faults that have been found in roadside equipment. This may be in an indicator or in a traffic sensor. The data flow consists of the following data items each of which is defined in its own DDE: indicator_fault_data + vehicle_smart_probe_data_output_fault + traffic_sensor_status.

Logical Architecture Reference Flow: traffic_sensor_status
This data flow is used within the Manage Traffic function to report the status of a sensor. By monitoring this data flow, the receiving process can monitor the health and current status of field equipment.

Physical Architecture Flow Name: freeway control status

Current operational status and operating parameters for ramp meters, dynamic message signs, mainline metering/lane controls and other control equipment associated with freeway operations.

Logical Architecture Reference Flow: indicator_input_data_from_highways
This data flow is used within the Manage Traffic function and contains the actual state of operation of the roadside indicators used to pass instructions to drivers and travelers on freeways within the geographic and/or jurisdictional area(s) served by the function. It is used for centralized monitoring the operation of the indicators and consists of the following data items each of which is defined in its own DDE: list_size + list_size{indicator_identity + indicator_response_state}.

Physical Architecture Flow Name: hov data

Current HOV lane information including both standard traffic flow measures and information regarding vehicle occupancy in HOV lanes.

Logical Architecture Reference Flow: hov_lane_data_input
This data is used within the Manage Traffic function and contains data from which the use of High Occupancy Vehicle (HOV) lanes can be monitored. It consists of the following data items each of which is defined in its own DDE: private_vehicle_occupants + traffic_video_image + vehicle_count.

Logical Architecture Reference Flow: hov_sensor_data
This data flow is used within the Manage Traffic function and contains the HOV data obtained from processing the inputs from sensors around the road network. It consists of the following data items each of which is defined in its own DDE: 1{private_vehicle_occupants+ hov_priority}link_list.

Physical Architecture Flow Name: hri status

Status of the highway-rail intersection equipment including both the current state or mode of operation and the current equipment condition.

Logical Architecture Reference Flow: hri_guidance_for_beacon_message

This data flow is used to control which message is to be broadcast to drivers approaching an HRI. The sizing assumption is based on a coded implementation that uses a 3 character ASCII code to select one of many prerecorded messages.

Logical Architecture Reference Flow: hri_guidance_for_vms

This data flow is used to control which message is to be displayed on a variable message sign (vms) as drivers approach an HRI. The sizing assumption is based on a coded implementation that uses a 3 character ASCII code to select on of several predefined messages.

Logical Architecture Reference Flow: hri_status

This data flow represents the complete status of an HRI, including train situation, vehicle traffic, equipment health and predictable near term events. It consists of the following data items each of which is defined in its own DDE: hri_state + hri_closure_data_response.

Logical Architecture Reference Flow: hri_traffic_data

This data flow contains data to be used by traffic management to coordinate its overall operations with the hri activity.

Logical Architecture Reference Flow: rail_operations_message

This data flow contains advanced (predictive) data about an HRI operational status to be passed to rail operations. It is generated by the Manage HRI Traffic process for use by the Interact with Rail Operations process.

Logical Architecture Reference Flow: traffic_management_request

This data flow is used by hri to request services or data from other traffic management functions .

Physical Architecture Flow Name: incident data

Data and imagery from the roadside supporting incident detection and verification.

Logical Architecture Reference Flow: incident_analysis_data

This data flow is used within the Manage Traffic function and contains processed traffic sensor data that can be analyzed for the possible presence of incidents. The data is provided directly from the local traffic sensor process rather than from some regional/area based process and so must originate in sensors that are within a small geographic area.

Logical Architecture Reference Flow: incident_video_image

This data flow is used within the Manage Traffic function and contains a high resolution digitized image of a potential or current incident at a particular point on the road or freeway network. The size expression below assumes that the image is compressed.

Physical Architecture Flow Name: intersection blockage notification

Notification that a highway-rail intersection is obstructed and supporting information.

Logical Architecture Reference Flow: hri_blockage

This data flow contains information, obtained from sensors in the intersection, regarding blockage of the hri by a vehicle or other object. This data will be passed to Rail Operations.

Logical Architecture Reference Flow: intersection_blocked

This data flow contains information, obtained from sensors in the intersection, regarding blockage of the hri by a vehicle or other object. This data will be used by the traffic management functions to begin incident management procedures.

Physical Architecture Flow Name: request for right-of-way

Forwarded request from signal prioritization, signal preemption, pedestrian call, multi-modal crossing activation, or other source for right-of-way.

Logical Architecture Reference Flow: indicator_input_data_from_roads

This data flow is used within the Manage Traffic function and contains the actual state of operation of the roadside and grade crossing indicators used to pass instructions to drivers and travelers on roads (surface streets) within the geographic and/or jurisdictional area(s) served by the function. It is used for centralized monitoring the operation of the indicators and consists of the following data items each of which is defined in its own DDE: list_size + list_size{indicator_identity + indicator_response_state}.

Logical Architecture Reference Flow: multimodal_crossing_sensor_data

This data flow is used within the Manage Traffic function and contains the multimodal crossing data obtained from processing the other inputs from sensors around the road network. It consists of the following data items each of which is defined in its own DDE: 1{crossing_close_time+ crossing_close_duration}crossing_list.

Logical Architecture Reference Flow: pedestrian_sensor_data

This data flow is used within the Manage Traffic function and contains the pedestrian data obtained from processing the other inputs from sensors around the road network. It consists of the following data items each of which is defined in its own DDE: 1{pedestrian_demand}node_list.

Physical Architecture Flow Name: reversible lane status

Current reversible lane status including traffic sensor and surveillance data and the operational status and mode of the reversible lane control equipment.

Logical Architecture Reference Flow: reversible_lane_video_images

This data flow is used within the Manage Traffic function. It contains video images of the reversible lanes. It consists of the following data item which is defined in its own DDE: incident_video_image.

Logical Architecture Reference Flow: sensor_data_for_reversible_lanes

This data flow is used within the Manage Traffic function and contains data from which a wrong way vehicle is detected in a reversible lane through the use of sensors located. It consists of the following data items each of which is defined in its own DDE: traffic_video_image + vehicle_detection_data.

Physical Architecture Flow Name: roadway information system status

Current operating status of dynamic message signs, highway advisory radios, beacon systems, or other configurable field equipment that provides dynamic information to the driver.

Logical Architecture Reference Flow: dms_status_for_highways

This data flow is used within the Manage Traffic function and contains data about the text strings of information to be output to drivers on freeways in the geographic and/or jurisdictional area(s) served by the function. The data flow consists of the following data items each of which is defined in its own DDE: vms_updates_for_highways.

Logical Architecture Reference Flow: dms_status_for_roads

This data flow contains the Dynamic Message Sign status for sign control data, operating at the roadside on roads (surface streets) in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{indicator_identity + vms_advisory_text}.

Logical Architecture Reference Flow: har_status_for_highways

This data flow contains the Highway Advisory Radio status for HARs, operating at the roadside on highways in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: har_status+ har_identity.

Logical Architecture Reference Flow: har_status_for_roads

This data flow contains the Highway Advisory Radio status for HARs, operating at the roadside on roads (surface streets) in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: har_status+ har_identity.

Logical Architecture Reference Flow: information_device_fault_status

This data flow is used to show any faults that have been found in roadside information dissemination equipment. This includes highway advisory radio, dynamic message signs, or in-vehicle signs. The data flow consists of the following data items each of which is defined in its own DDE: har_fault_data_for_roads+ har_fault_data_for_highways+ vehicle_sign_data_output_fault.

Physical Architecture Flow Name: signal control status

Status of surface street signal controls.

Logical Architecture Reference Flow: indicator_input_data_from_roads

This data flow is used within the Manage Traffic function and contains the actual state of operation of the roadside and grade crossing indicators used to pass instructions to drivers and travelers on roads (surface streets) within the geographic and/or jurisdictional area(s) served by the function. It is used for centralized monitoring the operation of the indicators and consists of the following data items each of which is defined in its own DDE: list_size + list_size{indicator_identity + indicator_response_state}.

Physical Architecture Flow Name: traffic flow

Raw and/or processed traffic detector information which allows derivation of traffic flow, speed, and density measures.

Logical Architecture Reference Flow: traffic_image_data

This data flow is used within the Manage Traffic function and contains the data produced by processing image data obtained from visual detection systems. This data is therefore that which can be obtained from systems such as traffic surveillance closed circuit television (cctv). It is analyzed and used to detect traffic conditions such as flow, occupancy, possible incidents, etc. The size expression below assumes that the image is compressed.

Logical Architecture Reference Flow: traffic_sensor_data

This data flow is used within the Manage Traffic function and contains the data obtained from processing the inputs from sensors around the road network. It consists of the following data items each of which is defined in its own DDE: list_size + 1{station_id + sensor_identity + traffic_sensor_output}list_size.

Physical Architecture Flow Name: traffic images

High fidelity, real-time traffic images suitable for surveillance monitoring by the operator or for use in machine vision applications.

Logical Architecture Reference Flow: traffic_video_image

This data flow is used within the Manage Traffic function and contains a video image of sufficient fidelity to support operator monitoring applications. This image can be a by-product of a machine vision application or the end-product of a system dedicated to traffic surveillance.

Logical Architecture Reference Flow: traffic_video_image_for_display

This data flow contains the video image which is used by a roadside device to measure traffic flow measures. The data flow consists of the following item which is defined in its own DDE: traffic_video_image.

Physical Architecture Flow Name: vehicle probe data

Vehicle probe data indicating identity, route segment identity, link time and location.

Logical Architecture Reference Flow: vehicle_smart_probe_data_for_storage
This data flow is used within the Manage Traffic function. It contains the processed vehicle smart probe data collected from a roadside unit, which in turn have received data output by suitably equipped vehicles as they pass by. The data flow consists of the following data items each of which is defined in its own DDE: vehicle_smart_probe_data_source + vehicle_smart_probe_data_indication.

Logical Architecture Reference Flow: vehicle_tag_data
This data flow is used within the Manage Traffic function. It contains the data from parking lot and toll tags on-board vehicles plus the identity of the unit which received the data. The data flow consists of the following data items each of which is defined in its own DDE: vehicle_tag_data_source_identity + vehicle_tag_input_data.

Roadway Subsystem => **Vehicle**

Physical Architecture Flow Name: AHS control data

Information required for vehicles to operate on AHS lanes.

Logical Architecture Reference Flow: ahs_check_response
This data flow is used within the Provide Vehicle Monitoring and Control function and contains the response to the checking of data from on-board a vehicle to see if it is suitable for operating on automatic highway system (ahs) lanes. The data flow consists of the following data items each of which is defined in its own DDE: ahs_control_data_update + confirmation_flag.

Physical Architecture Flow Name: intersection status

Status of intersection congestion, approaching vehicles, etc.

Logical Architecture Reference Flow: intersection_collision_avoidance_data
This data flow is sent from the Manage Traffic function to the Provide Vehicle Monitoring and Control function. It contains data for a vehicle that shows that it is likely to be involved in a collision at an intersection, unless it takes some avoiding action. The data flow is sized at two (2) bytes so that it can show the direction from which the other vehicle(s) is (are) approaching. This will help the vehicle to decide which is the most appropriate avoiding action to take.

Physical Architecture Flow Name: request tag data

Request for tag information including credit identity, stored value card cash, etc.

Logical Architecture Reference Flow: parking_lot_tag_data_needed
This data flow is used within the Manage Traffic and Provide Electronic Payment Services functions to request the output of the data from a toll tag that may be on-board a vehicle. This data will be used to calculate vehicle journey times for links in the road (surface street) and freeway network served by the Manage Traffic function.

Logical Architecture Reference Flow: toll_tag_data_needed
This data flow is used within the Manage Traffic and Provide Electronic Payment Services functions to request the output of the data from a parking lot tag that may be on-board a vehicle. This data will be used to calculate vehicle journey times for links in the road (surface street) and freeway network served by the Manage Traffic function.

Physical Architecture Flow Name: vehicle signage data

In-vehicle signage data generated by the roadway infrastructure indicating either road conditions, street names, or special information which will be useful for a vehicle passing a specific point on the roadway.

Logical Architecture Reference Flow: vehicle_signage_data
This data flow is sent from the Manage Traffic function to the Provide Driver and Traveler Services function. It contains data for use in producing in-vehicle signage displays. The data flow consists of the following data items each of which is defined in its own DDE: vehicle_sign_data_output_location + 6{vehicle_signage_output_data} + 8{vehicle_signage_traffic_data}.

Logical Architecture Reference Flow: vehicle_smart_probe_data_output
This data flow is sent from the Manage Traffic function to the Provide Driver and Traveler Services function. It contains the data obtained from vehicle smart probes, processsed and formatted for output to vehicles as they pass by. The data flow is sized at one (1) byte and will contain a compressed version of the internally used character codes for the various types of hazard that may be found on roads and freeways.

Roadway Subsystem => **Wayside Equipment**

Physical Architecture Flow Name: hri status

Status of the highway-rail intersection equipment including both the current state or mode of operation and the current equipment condition.

Logical Architecture Reference Flow: twe_hri_status
This data flow provides a real-time indication of the status at a highway grade crossing (e.g. operational, not-operational, obstructed, etc.). The sizing assumption is based on a coded aspect in a single byte that is interpreted by railroad equipment, either at the wayside, a central site, or on-board the locomotive or hi-rail vehicle.

Physical Architecture Flow Name: intersection blockage notification

Notification that a highway-rail intersection is obstructed and supporting information.

Logical Architecture Reference Flow: twe_stop_highway_indication
This data flow provides a real-time confirmation that a highway grade crossing is closed to highway non-rail traffic and all trains may proceed at full authorized speed. Alternative indications are possible, i.e. proceed at reduced speed - prepared to stop. The sizing assumption is based on a coded aspect in a single byte that is interpreted by railroad equipment, either at the wayside or on-board the locomotive or hi-rail vehicle.

Logical Architecture Reference Flow: twe_stop_train_indication
This data flow provides a real-time indication that a highway grade crossing is obstructed or otherwise closed and all trains must stop prior to entering it. Alternative indications to full stop are possible, i.e. proceed at reduced speed - prepared to stop. The sizing assumption is based on a coded aspect in a single byte that is interpreted by railroad equipment, either at the wayside, a central site, or on-board the locomotive or hi-rail vehicle.

Traffic => **Roadway Subsystem**

Physical Architecture Flow Name: traffic characteristics

Physical traffic characteristics which are monitored and translated into macroscopic measures like occupancy, volume, density, and average speed. Point measures support presence detection and individual vehicle measures like speed.

Logical Architecture Reference Flow: ft_traffic_data
This data flow is used within the Manage Traffic function. It contains analog data about the presence of traffic flowing on roads and highways from which traffic surveillance data such as vehicle speed, occupancy, flow volume, headway,

etc. can be obtained by sensors within ITS.

Logical Architecture Reference Flow: ft_traffic_images
This data flow is used within the Manage Traffic function. It contains visual information (analog data) about the traffic flowing on roads and highways (and at highway rail grade crossings) from which traffic surveillance data can be obtained by image processors within ITS. The size shown below is representative of a digitized equivalent of this analog information.

Logical Architecture Reference Flow: ft_vehicle_pollutant_levels
This data flow is used within the Manage Traffic function. It contains analog data from which sensors within ITS can determine the actual levels of various atmospheric pollutants, such as nitrous oxide, sulphur dioxide, hydrocarbons, carbon monoxide and ozone, that are being produced by particular passing vehicles.

Traffic Management => Roadway Subsystem

Physical Architecture Flow Name: AHS control information

Control data to AHS roadway equipment

Logical Architecture Reference Flow: ahs_control_data_changes
This data flow is used within the Provide Vehicle Monitoring and Control function to send data from the management facility to the automatic highway system (ahs) check-in facilities. It contains data defining parameters to be used by vehicles participating in platoon following and running in ahs controlled lanes, which will override any already loaded into the vehicles. The data flow consists of the following items of data each of which is defined in its own DDE: ahs_demand_accel_decel_profile + ahs_demand_headway.

Physical Architecture Flow Name: freeway control data

Control commands and operating parameters for ramp meters, dynamic message signs, mainline metering/lane controls and other systems associated with freeway operations.

Logical Architecture Reference Flow: indicator_control_data_for_highways
This data flow is used within the Manage Traffic function and contains the actual data from which instructions to the driver and traveler can be produced by indicators at the roadside on freeways in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: indicator_crossing_control_data_for_highways + indicator_ramp_control_data.

Logical Architecture Reference Flow: indicator_control_monitoring_data_for_highways
This data flow is used within the Manage Traffic function. It contains the actual data from which instructions to the driver and traveler can be produced by indicators on the freeways in the geographic and/or jurisdictional area(s) served by the function. In this case the data is used by a process to monitor the operation of the indicators rather than actual message output. The data flow consists of the following data items each of which is defined in its own DDE: indicator_control_data_for_highways.

Physical Architecture Flow Name: hri control data

Data required for HRI information transmitted at railroad grade crossings and within railroad operations.

Logical Architecture Reference Flow: hri_traffic_surveillance
This data flow represents the various traffic sensor inputs to HRI from the Traffic Surveillance processes .

Logical Architecture Reference Flow: indicator_sign_control_data_for_hri
This data flow is used within the Manage Traffic function and contains the actual data from which instructions to the driver and traveler can be produced by indicators, variable message (vms), advisory beacons, and other types of signs

on the roads (surface streets) in the vicinity of railroad grade crossings. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{crossing_id + hri_sign_control_data}.

Logical Architecture Reference Flow: rail_operations_advisories
This data flow contains advisory information for HRI vehicular traffic that has been derived from information received from rail operations.

Logical Architecture Reference Flow: rail_operations_device_command
This data flow contains HRI device commands that have been derived from information received from rail operations and provides for rail operations preemption capability.

Physical Architecture Flow Name: hri request

A request for highway-rail intersection status or a specific control request intended to modify HRI operation.

Logical Architecture Reference Flow: ro_requests
This data flow is generated in response to a need for hri status for rail operations.

Logical Architecture Reference Flow: tms_requests
This data flow is generated in response to a need for hri status for traffic management.

Physical Architecture Flow Name: roadway information system data

Information used to initialize, configure, and control roadside systems that provide driver information (e.g., dynamic message signs, highway advisory radio, beacon systems). This flow can provide message content and delivery attributes, local message store maintenance requests, control mode commands, status queries, and all other commands and associated parameters that support remote management of these systems.

Logical Architecture Reference Flow: dms_data_for_highways
This data flow is used within the Manage Traffic function and contains DMS data about text strings of information to be output to drivers on freeways in the geographic and/or jurisdictional area(s) served by the function. The data flow consists of the following data items each of which is defined in its own DDE: indicator_sign_control_data_for_highways.

Logical Architecture Reference Flow: dms_data_for_roads
This data flow is used within the Manage Traffic function and contains the actual data from which instructions to the driver and traveler can be produced by indicators at dynamic message (dms) and other types of signs on the roads. It consists of the following data items each of which is defined in its own DDE: indicator_sign_control_data_for_roads + indicator_sign_control_data_for_hri.

Logical Architecture Reference Flow: har_data_for_highways
This data flow contains the HAR data, both program and management, used to define the output of a Highway Advisory Radio (HAR) operating at the roadside on highways in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: har_identity + har_program + har_management_data.

Logical Architecture Reference Flow: har_data_for_roads
This data flow contains the HAR data, both program and management, used to define the output of a Highway Advisory Radio (HAR) operating at the roadside on roads (surface streets) in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: har_identity + har_program + har_management_data.

Logical Architecture Reference Flow: vehicle_sign_data
This data flow is used within the Manage Traffic function and contains data for use in producing in-vehicle signage

displays. The information sent can fall into two categories - sign data and situation data. Sign data includes permanent fixed signs (e.g. STOP, YIELD, etc), temporary signs (e.g. detours), and variable message signs. Situation data provides information about traffic conditions, e.g. congestion and speed, on up to eight (8) links. Up to six (6) sets of indicator data may be contained within the data flow and they may be for the same or different types of indicator, or for incidents. All data is filtered so that the receiving processes only get that which is relevant to their local geographic area. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{vehicle_signage_output_identity + 6{vehicle_signage_output_data} + 8{vehicle_signage_traffic_data}}.

Physical Architecture Flow Name: sensor and surveillance control

Information used to configure and control sensor and surveillance systems at the roadside.

Logical Architecture Reference Flow: environment_sensor_configuration_data
This data flow is used within the Manage Traffic function to provide environmental sensor control commands. It consists of the following data items each of which is defined in its own DDE:

Logical Architecture Reference Flow: incident_video_image_control
This data flow is used within the Manage Traffic function and contains control parameters for closed circuit television (cctv) systems located used to provide incident management information. These parameters may cover things such as camera pan, tilt, and zoom, plus other picture controls.

Logical Architecture Reference Flow: sensor_configuration_data
This data flow provides control commands for advanced sensors, including video sensing systems .

Physical Architecture Flow Name: signal control data

Information used to configure and control traffic signal control systems.

Logical Architecture Reference Flow: indicator_control_data_for_roads
This data flow is used within the Manage Traffic function and contains the actual data from which instructions to the driver and traveler can be produced by indicators at the roadside on roads (surface streets) in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: indicator_crossing_control_data_for_roads + indicator_junction_control_data + indicator_pedestrian_control_data.

Logical Architecture Reference Flow: indicator_control_monitoring_data_for_roads
This data flow is used within the Manage Traffic function. It contains the actual data from which instructions to the driver and traveler can be produced by indicators on the roads (surface streets) and at railroad grade crossings in the geographic and/or jurisdictional area(s) served by the function. In this case the data is used by a process to monitor the operation of the indicators rather than actual message output. The data flow consists of the following data items each of which is defined in its own DDE: indicator_control_data_for_roads + indicator_control_data_for_hri.

Transit Vehicle Subsystem => Roadway Subsystem

Physical Architecture Flow Name: local signal priority request

Request from a vehicle to a signalized intersection for priority at that intersection.

Logical Architecture Reference Flow: transit_vehicle_roadway_preemptions
This data flow is sent from the Manage Transit function to the Manage Traffic function and contains data necessary for an individual transit vehicle to be given preemption (priority) at indicator controllers. This will be at the controller for a particular road junction, pedestrian crossing, or highway entrance ramp. The data is sent directly from the transit vehicle to the next controller along its route and therefore is not subject to any centralized coordination. Local

coordination may be provided if there are links between adjacent controllers. The data flow consists of the following data items each of which is defined in its own DDE: transit_vehicle_junction_preemption + transit_vehicle_pedestrian_preemption + transit_vehicle_ramp_preemption + transit_vehicle_sign_preemption.

Vehicle => Roadway Subsystem

Physical Architecture Flow Name: AHS vehicle data

AHS route and vehicle condition data

Logical Architecture Reference Flow: ahs_route_data

This data flow is used within the Provide Vehicle Monitoring and Control function and contains a list of the route segments that will be used by a vehicle. These route segments will be those that contain automatic highway system (ahs) lanes, and will be used by the vehicle on its ahs controlled route. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{route_segment_identity}.

Logical Architecture Reference Flow: ahs_vehicle_condition

This data flow is used within the Provide Vehicle Monitoring and Control function. It contains data processed from on-board vehicle sensors that show the vehicle's current operating condition. This data is used to determine its suitability for operating on automatic highway system (ahs) lanes.

Physical Architecture Flow Name: vehicle probe data

Vehicle probe data indicating identity, route segment identity, link time and location.

Logical Architecture Reference Flow: parking_lot_tag_data_input

This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function. It contains the data from parking lot and toll tags on-board vehicles which will be used to calculate vehicle journey times for links in the road (surface street) and freeway network served by the Manage Traffic function. The data consists of a unique identity number which is assigned to each tag as it is read parking_lot_tag_data.

Logical Architecture Reference Flow: toll_tag_data_input

This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function. It contains the data from a toll tag on-board a vehicle which will be used to calculate vehicle journey times for links in the road (surface street) and freeway network served by the Manage Traffic function. The data flow consists of the following data item which is defined in its own DDE: toll_tag_data.

Logical Architecture Reference Flow: vehicle_smart_probe_data

This data flow contains data which provides information about conditions in the vicinity of the smart probe. These conditions, which may be the indication of a hazard on the road or freeway that has been detected by sensors on-board the vehicle. The type of information measured could comprise but not be limited to such things as, temperature, fog, ice, snow, and road condition (e.g. wet, icy, dry). The data may be provided as distinct elements with actual measured values (e.g. temperature) or it could provide conditions from a list of codes.

Logical Architecture Reference Flow: vehicle_status_details_for_emissions

This data flow is sent from the Provide Vehicle Control and Monitoring function to the Manage Traffic function. It contains the operational status of the vehicle which is important because the levels of pollution vary according to how long the vehicle has been running, i.e. how warm is the engine, and what it is actually doing, e.g. is it stationary, or pulling away from a stop. The data flow consists of the following data item which is defined in its own DDE: vehicle_status_details.

Vehicle Characteristics => Roadway Subsystem

Physical Architecture Flow Name: vehicle characteristics

The physical or visible characteristics of an individual vehicle that can be measured to classify a vehicle and imaged to uniquely identify a vehicle.

Logical Architecture Reference Flow: From_Vehicle_Characteristics

This data flow is sent from the vehicle characteristics terminator. It represents the presence of a vehicle near a sensor, which allows the sensor to create an output that can be used to identify a particular vehicle and its characteristics, such as the number of wheels, size, pollution parameters, etc., for toll payment and parking lot charging purposes. The sensor may also determine the visible characteristics of a vehicle and use that data to obtain information about toll and parking lot charge violators.

Wayside Equipment => **Roadway Subsystem**

Physical Architecture Flow Name: arriving train information

Information for a train approaching a highway-rail intersection that may include direction and allow calculation of approximate arrival time and closure duration.

Logical Architecture Reference Flow: fwe_train_data

This data flow contains time critical data about an approaching train and is provided to HRI at the roadside by railroad owned and maintained equipment and/or communications networks. This data, if available, will be provided concurrently with the approaching train announcement and must include data sufficient for the HRI to determine crossing close time, and the anticipated closing duration. The sizing below assumes the flow includes arrival time, speed, and length of train. A smaller alternative flow might be train arrival time and a pre-computed closing duration. A train identification is required to allow the HRI to manage multiple train arrivals at a single crossing within short time intervals.

Physical Architecture Flow Name: track status

Current status of the wayside equipment and notification of an arriving train.

Logical Architecture Reference Flow: fwe_approaching_train_announcement

This data flow is a real-time notification to the HRI that a train is approaching a specific grade crossing. In its simplest form it is a simple binary indication. Two bytes are assumed so that more sophisticated implementations may also include the crossing ID to allow for wireless short-range broadcast (e.g. to multiple crossings in a local configuration).

Logical Architecture Reference Flow: fwe_wayside_equipment_status

This data flow allows the railroad operated and maintained equipment to verify its operational status to dependent HRI processes. This can be as simple as a binary indication of status, to a full maintenance report. The sizing assumption is based on a single status byte.

2.16.4 Subsystem Architecture Flow Diagram

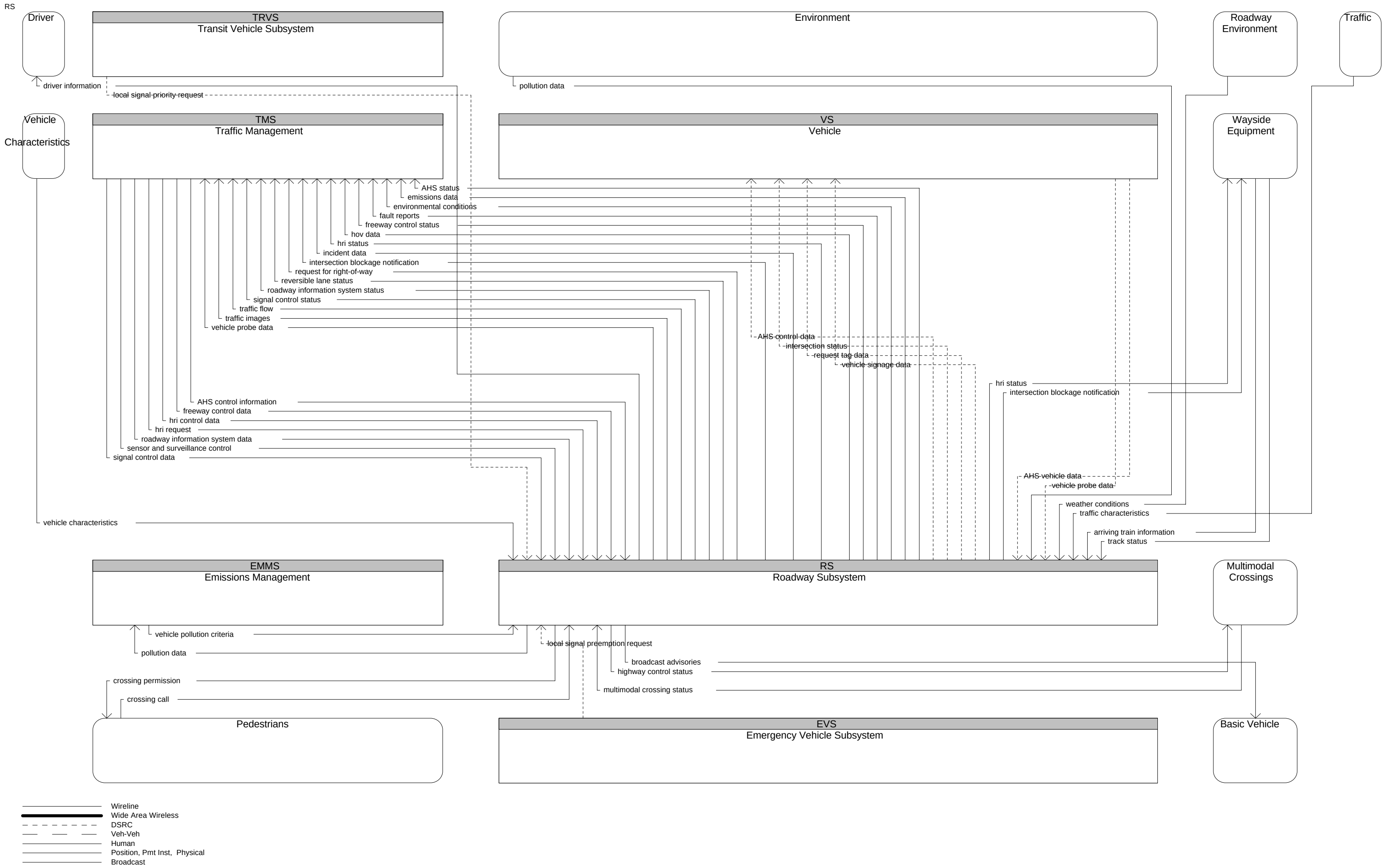


Figure 2.16-2 Architecture Flow Diagram for RS

2.17 Toll Administration

The Toll Administration Subsystem provides general payment administration capabilities and supports the electronic transfer of authenticated funds from the customer to the transportation system operator. This subsystem supports traveler enrollment and collection of both pre-payment and post-payment transportation fees in coordination with the existing, and evolving financial infrastructure supporting electronic payment transactions. The system may establish and administer escrow accounts depending on the clearinghouse scheme and the type of payments involved. This subsystem posts a transaction to the customer account and generates a bill (for post-payment accounts), debits an escrow account, or interfaces to the financial infrastructure to debit a customer designated account. It supports communications with the Toll Collection Subsystem to support fee collection operations. The subsystem also sets and administers the pricing structures and includes the capability to implement road pricing policies in coordination with the Traffic Management Subsystem. The electronic financial transactions in which this subsystem is an intermediary between the customer and the financial infrastructure shall be cryptographically protected and authenticated to preserve privacy and ensure authenticity and audibility.

2.17.1 Alternative Configurations

The Toll Administration Subsystem coordinates fees from toll collection booths Figure 2.17-1. Toll Administration subsystems may be deployed by any agency charging fees for use of roadways. This may include a Traffic Management Center using the toll collection facilities for road pricing if it is instituted by local policy.

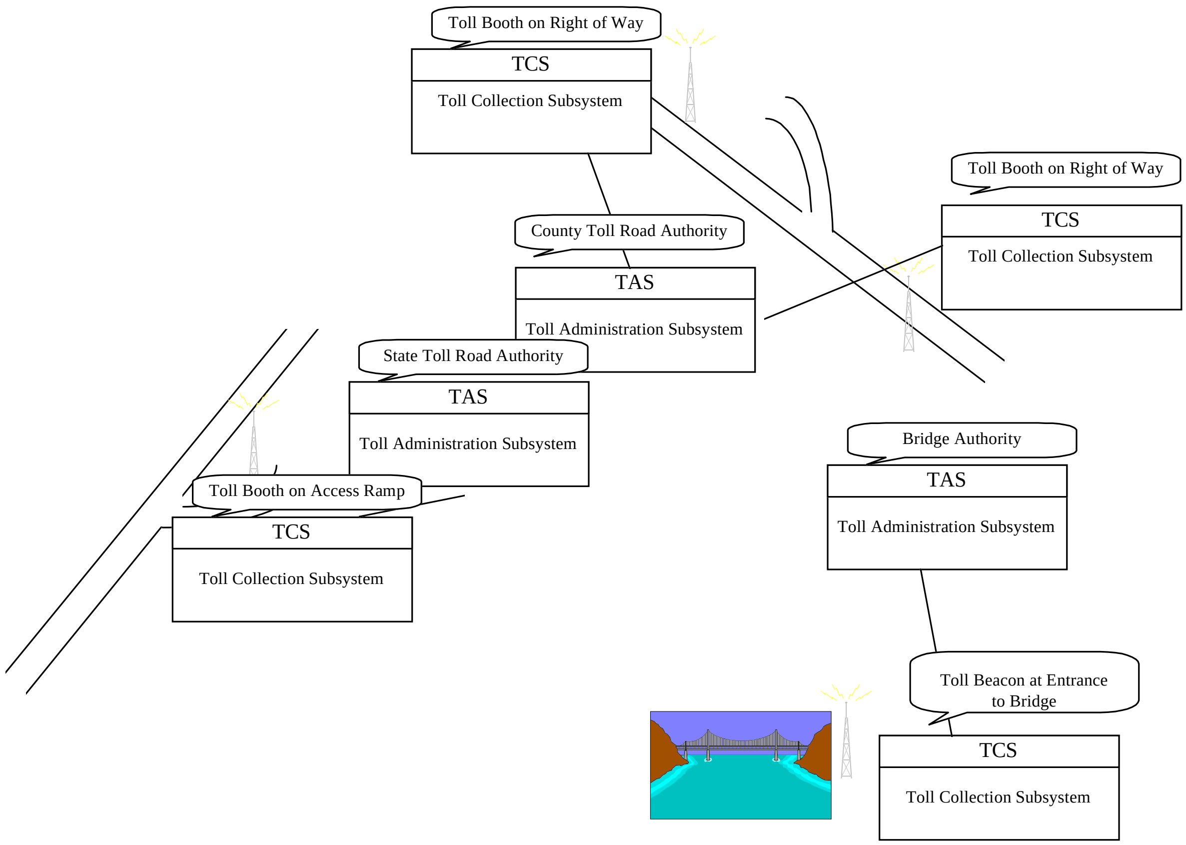


Figure 2.17-1 Alternative Configurations for Toll Administration

2.17.2 Subsystem Equipment Packages and Supporting Process Specifications for TAS

Toll Administration

This Equipment package provides the capability to maintain database information with pricing structure information. This capability shall be provided through database software containing pricing structure and current traffic conditions on the transportation network obtained from the transportation management center. This capability allows the determination of dynamic tolls according to congestion levels for demand management. Secure wireline communications with the financial infrastructure and distributed toll plazas supports electronic payments and other ancillary requirements such as lost/stolen tag identification and management. Dependent on customer account requirements and the clearinghouse arrangement, this Equipment package may also contain a billing database.

Process Specifications

5.4.2 Process Violations for Tolls

Overview: This process shall manage the details of toll payment violations reported by the Provide Electronic Payments Services function. The process shall use the parameters in the store of toll payment violation (enforcement) data to obtain the vehicle registration data from the appropriate State Department of Motor Vehicles (DMV) office (or alternate source) for vehicles that are not equipped with a tag, before sending all of the received information to the correct law enforcement agency. This process shall also maintain the toll payment enforcement data-store, entering all information received from other processes.

7.1.1.3 Manage Bad Toll Payment Data

Overview: This process shall be responsible for maintaining a data store containing a list of invalid driver credit identities. This process shall use this data to verify credit identities and commercial vehicle carrier numbers provided for checking by the billing process. Verification shall ensure that the current toll payment transaction is using a credit identity or carrier identity that has not previously had an invalid transaction. Details of potential invalid credit identities or carrier numbers shall be sent by this process to the financial institution for verification. This process shall also receive from the financial institution details of invalid payment instrument data that has been found by other means.

7.1.1.6 Collect Probe Data From Toll Transactions

Overview: This process shall calculate the time taken for vehicles to travel between successive toll plazas and sending it to the Manage Traffic and Provide Driver and Traveler Services functions. The process shall periodically request the data from the process that manages toll financial processing and ensure that any references to the driver and/or vehicle identity plus any other payment information are removed from the data before it is sent to the other functions.

7.1.1.7 Update Toll Price Data

Overview: This process shall be responsible for maintaining a store of data containing the toll price, which may vary according to the type of vehicle. The process shall also act as the interface for the output and input of responses to toll price change requests from the Manage Traffic function, the provision of toll price information to the Centralized Payments facility, and to the toll service provider to enable changes to be made to the stored data. The process shall support inputs from the toll service provider in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be either in hardcopy, or as a display.

7.1.1.8 Register for Advanced Toll Payment

Overview: This process shall be responsible for responding to requests for tolls to be paid in advance. It shall provide the toll service provider with the opportunity to deny the request for an advanced toll. If approved, the advanced toll data shall be forwarded by the process to other processes for the actual toll cost to be obtained and payment transactions initiated.

7.1.1.9 Manage Toll Financial Processing

Overview: This process shall be responsible for maintaining a log of all toll transactions that are carried out by other processes in the toll payments facility. At periodic intervals the process shall output the accumulated records to the toll service provider, the toll operator and the Plan System Deployment function. It shall also output the data on request to the process that calculates probe data from the average travel time between toll plazas. The identity of the payee shall be removed from the data before it is used in any of these outputs. The first two outputs shall be produced by the process in visual form, which may be either hardcopy or a visual display. The process shall also be responsible for sending details of transactions to the financial institution to enable the users to be billed through their credit identities. For commercial vehicles, this will be done using the data provided by the vehicle's on-board tag and shall enable billing to the financial institution to be made by carrier.

2.17.3 Subsystem Interfaces for TAS

DMV => Toll Administration

Physical Architecture Flow Name: registration

Registered owner of vehicle and associated vehicle information.

Logical Architecture Reference Flow: fdmv_toll_violation_state_identity

This data flow is sent from the department of motor vehicles to the Manage Emergency Services function and contains the identity of the state that is supplying the requested vehicle registration data to enable a toll payment violation to be processed.

Logical Architecture Reference Flow: fdmv_toll_violation_vehicle_registration
This data flow is sent from the department of motor vehicles to the Manage Emergency Services function and contains the requested vehicle registration data to enable a toll payment violation to be processed.

Financial Institution => Toll Administration

Physical Architecture Flow Name: transaction status

Response to transaction request. Normally dealing with a request for payment.

Logical Architecture Reference Flow: ffi_bad_toll_payment_updates
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains data about a parking lot charge transaction that was either attempted but did not work, or for which the subsequent payment transaction by the Financial Institution failed. The data is to be used within the function for checking against future parking lot charge transaction data.

Logical Architecture Reference Flow: ffi_confirm_toll_payment
This data flow is sent from the Financial Institution to the Provide Electronic Payment Services function. It is used to confirm that a previous request for toll payments is being processed by the Financial Institution.

Information Service Provider => Toll Administration

Physical Architecture Flow Name: toll data request

Request made to obtain toll schedule information or pay a toll in advance. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Logical Architecture Reference Flow: advanced_other_tolls_request
This data flow is used within the Provide Electronic Payment Services function to request that a toll be paid for in advance by either a driver who is paying a parking lot charge or a traveler (as a transit user) who is paying a transit fare. It consists of the following data items each of which is defined in its own DDE: credit_identity + stored_credit + toll_route_segments + vehicle_identity.

Logical Architecture Reference Flow: advanced_traveler_tolls_request
This data flow is used within the Provide Electronic Payment Services function to request that a toll be paid for in advance by a traveler who is planning a trip. It consists of the following data items each of which is defined in its own DDE: credit_identity + stored_credit + toll_route_segments + vehicle_identity.

Logical Architecture Reference Flow: toll_price_data_request
This data flow is used within the Provide Electronic Payment Services function. It contains a request for the current toll price data to be provided from the store that is being used to calculate toll costs.

Toll Administration => DMV

Physical Architecture Flow Name: license request

Request supporting registration data based on license plate read during violation.

Logical Architecture Reference Flow: tdmv_toll_violation_identity_code
This data flow is sent to the department of motor vehicles from the Manage Emergency Services function and contains the identity code of the ITS that is requesting the vehicle registration data so that a toll payment violation can be processed.

Logical Architecture Reference Flow: tdmv_toll_violation_vehicle_license
This data flow is sent to the department of motor vehicles from the Manage Emergency Services function and contains the vehicle license for which the corresponding registration data is required so that a toll payment violation can be processed.

Toll Administration => **Enforcement Agency**

Physical Architecture Flow Name: violation notification

Notification to enforcement agency of violation or regulations.

Logical Architecture Reference Flow: tea_toll_violation_data
This data flow is sent from the Manage Emergency Services function to the enforcement agency and contains information about toll violations that have been detected by processes in the Provide Electronic Payment Services function. The data in the flow will enable the notified enforcement agency to take the appropriate action against those committing the violation.

Toll Administration => **Financial Institution**

Physical Architecture Flow Name: payment request

Request for payment from financial institution.

Logical Architecture Reference Flow: tfi_request_toll_payment
This data flow is sent to the Financial Institution by the Provide Electronic Payment Services function. It is sent periodically, e.g. once per day, and requests payment of the toll transactions since the previous request. The data flow will include the toll cost and credit identity for each transaction.

Logical Architecture Reference Flow: tfi_toll_payment_violator_data
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains data about a parking lot payment transaction that was attempted but did not work and is to be used by the Financial Institution.

1) Toll Administration => Information Service Provider

Physical Architecture Flow Name: probe data

Aggregate data from probe vehicles including location, speed for a given link or collection of links.

Logical Architecture Reference Flow: vehicle_toll_probe_data
This data flow is sent from the Provide Electronic Payment Services function to the Provide Driver and Traveler Services function. It contains the smoothed average vehicle journey times for the route segment between two toll collection points, and the identity of the route segment. The data is used to calculate link journey times for in-vehicle guidance purposes. The data flow consists of the following data items each of which is defined in its own DDE:
list_size + list_size{route_segment_identity + route_segment_journey_time_from_tolls}.

Physical Architecture Flow Name: toll data

Current toll schedules for different types of vehicles as well as advanced toll payment information.

Logical Architecture Reference Flow: advanced_other_tolls_confirm
This data flow is used within the Provide Electronic Payment Services function to confirm the advanced payment of tolls by a driver. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + stored_credit + toll_cost + vehicle_identity.

Logical Architecture Reference Flow: advanced_traveler_tolls_confirm
This data flow is used within the Provide Electronic Payment Services function. It contains data about an advanced toll transaction requested by a traveler and consists of the following data items each of which is defined in its own DDE: confirmation_flag + stored_credit + toll_cost + traveler_identity.

Logical Architecture Reference Flow: toll_price_data
This data flow is used within the Provide Electronic Payment Services function. It contains the price for each road segment to which a toll applies, with the time and date for when it applies. The data flow consists of the following data items each of which is defined in its own DDE: toll_segment_identity + toll_price + toll_price_application_time + vehicle_type_for_tolls.

Toll Administration => **Planning Subsystem**

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: toll_operational_data
This data flow is sent from the Provide Electronic Payment Services function to the Plan System Deployment function. It contains data about the cost of toll segments and the number of users of those segments during the time period since the data was last sent. The data flow consists of the following data items each of which is defined in its own DDE: date + list_size + list_size{toll_cost + toll_segment_identity + toll_segment_users} + time.

Toll Administration => **Toll Collection**

Physical Architecture Flow Name: toll instructions

Demand management toll pricing information based on current congestion.

Logical Architecture Reference Flow: advanced_toll_needed
This data flow is used within the Provide Electronic Payment Services function to show that an advanced toll cost must be determined. It contains the following data items each of which is defined in its own DDE: credit_identity + stored_credit + toll_route_segments + vehicle_identity.

Logical Architecture Reference Flow: toll_bad_payment_check_response
This data flow is used within the Provide Electronic Payment Services function and contains the response to a request for a check that a driver requesting toll payment is not on the list of bad payers. In the case of commercial vehicles it will be the carrier that is used for the check. The data flow consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + cv_carrier_number.

Toll Administration => **Toll Operator**

Physical Architecture Flow Name: toll transaction reports

Summary report sent to toll collection point operator containing previous toll transactions.

Logical Architecture Reference Flow: tto_transaction_reports
This data flow is sent to the toll operator by the Provide Electronic Payments Services function. It contains details of the toll transactions that have taken place in the last day. The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Toll Administration => **Toll Service Provider**

Physical Architecture Flow Name: toll revenues and summary reports

Summary of toll revenues and toll-related reports to toll service provider.

Logical Architecture Reference Flow: ttsp_credit_identity
This data flow is sent to the toll service provider by the Provide Electronic Payment Services function and contains the identity number of a payment instrument which is to be used to pay advanced tolls.

Logical Architecture Reference Flow: ttsp_toll_price_changes_request
This data flow is sent from the Provide Electronic Payment Services function to the toll services provider. It contains data requesting a change to the current toll pricing structure so that travelers my be encouraged to change the modal split in their journeys.

Logical Architecture Reference Flow: ttsp_toll_segments
This data flow is sent to the toll service provider by the Provide Electronic Payment Services function and provides information on the route segment(s) for which advanced toll payment is being requested.

Logical Architecture Reference Flow: ttsp_transaction_reports
This data flow is sent to the toll service provider by the Provide Electronic Payment Services function. It contains details of the toll transactions that have taken place in the last period. The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Logical Architecture Reference Flow: ttsp_vehicle_identity
This data item is sent to the toll service provider by the Provide Electronic Payment Services function and contains the identity of a vehicle which is to be associated with an advanced toll payment.

Toll Administration => **Traffic Management**

Physical Architecture Flow Name: demand management response

Response to various demand management change requests indicating level of compliance with request.

Logical Architecture Reference Flow: toll_price_changes_response
This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function. It contains the response to a previous request for the current toll prices to be changed to help produce a change in the current modal split of trips being undertaken by all types of travelers. If sent to one (1) the change was accepted and if set to zero (0), the change was rejected.

Logical Architecture Reference Flow: toll_price_direct_details
This data flow contains the price for each road segment to which a toll applies, with the time and date for when it applies. This data will be used by the Manage Travel Demand facility in its efforts to re-distribute travel demand to the more efficient providers. The data flow consists of the following data items each of which is defined in its own DDE: toll_segments + toll_price + toll_price_application_time + vehicle_type_for_tolls.

Physical Architecture Flow Name: probe data

Aggregate data from probe vehicles including location, speed for a given link or collection of links.

Logical Architecture Reference Flow: probe_data_for_traffic
This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function and contains journey times between toll collection points for those vehicles equipped for electronic toll collection. It is used to calculate link journey times for use in adaptive traffic control techniques and route selection and guidance. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{route_segment_identity + route_segment_journey_time_from_tolls}.

Toll Collection => **Toll Administration**

Physical Architecture Flow Name: toll transactions

Detailed list of transactions from a toll station.

Logical Architecture Reference Flow: advanced_toll_transactions
This data flow is used within the Provide Electronic Payment Services function and contains data about advanced toll transactions as they take place for recording in the log of toll transactions. It consists of the following data items each of which is defined in its own DDE: credit_identity + cv_carrier_number + cv_vehicle_number + date + stored_credit + time + toll_cost + toll_plaza_identity + toll_route_segments + vehicle_identity.

Logical Architecture Reference Flow: confirm_advanced_tolls_payment
This data flow is used within the Provide Electronic Payment Services function to confirm that advanced payment for a toll has been made. This may have originated as a request from a traveler making pre-trip planning, as an advanced toll payment from a driver at a toll plaza or parking lot, or from a transit user in the transit network, i.e. on a transit vehicle or at a transit stop. The data flow consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + stored_credit + toll_cost + traveler_identity.

Logical Architecture Reference Flow: current_toll_transactions
This data flow is used within the Provide Electronic Payment Services function and contains data about current toll transactions as they take place for recording in the log of toll transactions. It consists of the following data items each of which is defined in its own DDE: credit_identity + cv_carrier_number + cv_vehicle_number + date + stored_credit + time + toll_cost + toll_plaza_identity + toll_route_segments + vehicle_identity.

Logical Architecture Reference Flow: toll_bad_payment_check_request
This data flow is used within the Provide Electronic Payment Services function and contains a request for a check that a driver requesting toll payment is not on the list of bad payers. In the case of commercial vehicles it will be the carrier that is used for the check. The data flow consists of the following data items each of which is defined in its own DDE: credit_identity + cv_carrier_number.

Logical Architecture Reference Flow: toll_payment_violator_data
This data flow is used within the Provide Electronic Payment Services function and contains data about a toll transaction that was attempted but did not work. It consists of the data items shown below, each of which is defined in its own DDE. For each particular set of data some of the data items may be blank depending on the reason(s) for which the transaction did not work. credit_identity + vehicle_identity + toll_cost.

Logical Architecture Reference Flow: toll_violation_information
This data is used by the Provide Electronic Payment Services functions to send data about a violator of the toll collection processes to the Manage Emergency Services function. This data will contain a digitized video image of the vehicle trying to violate the toll collection process.

Toll Operator => **Toll Administration**

Physical Architecture Flow Name: toll operator requests

Request for information from toll operator at toll collection site.

Logical Architecture Reference Flow: fto_local_toll_price_variations
This data flow is sent from the toll operator to the Provide Electronic Payment Services function and defines changes to the toll prices that are provided locally by the toll operator.

Toll Service Provider => **Toll Administration**

Physical Architecture Flow Name: toll fees

Instructions indicating toll fees which should be charged.

Logical Architecture Reference Flow: ftsp_confirm_advanced_toll
This data flow is sent from the toll service provider to the Provide Electronic Payment Services function and confirms the acceptance of payment of an advanced toll.

Logical Architecture Reference Flow: ftsp_toll_price_changes_response
This data flow is sent to the Provide Electronic Payment Services function by the toll services provider. It contains the response to a previous request for a change to the current toll pricing structure so that travelers my be encouraged to change the modal split in their journeys.

Logical Architecture Reference Flow: ftsp_toll_price_data
This data flow is sent from the toll service provider to the Provide Electronic Payment Services function and contains updates to the current toll prices.

Traffic Management => **Toll Administration**

Physical Architecture Flow Name: demand management request

Request to change the demand for road facility use through pricing or other mechanisms.

Logical Architecture Reference Flow: toll_price_changes_request
This data flow is sent from the Manage Traffic function to the Provide Electronic Payment Services function and contains a request for a change to the current toll pricing structure that will help to influence a change in modal split of journeys currently being undertaken by travelers of all types, i.e. including drivers and transit users. It consists of the following data items each of which is defined in its own DDE: toll_segments + toll_price + toll_price_application_time + vehicle_type_for_tolls.

Logical Architecture Reference Flow: toll_price_direct_request
This data flow contains a request for the current prices being charged for toll segments on the road and highway network.

2.17.4 Subsystem Architecture Flow Diagram

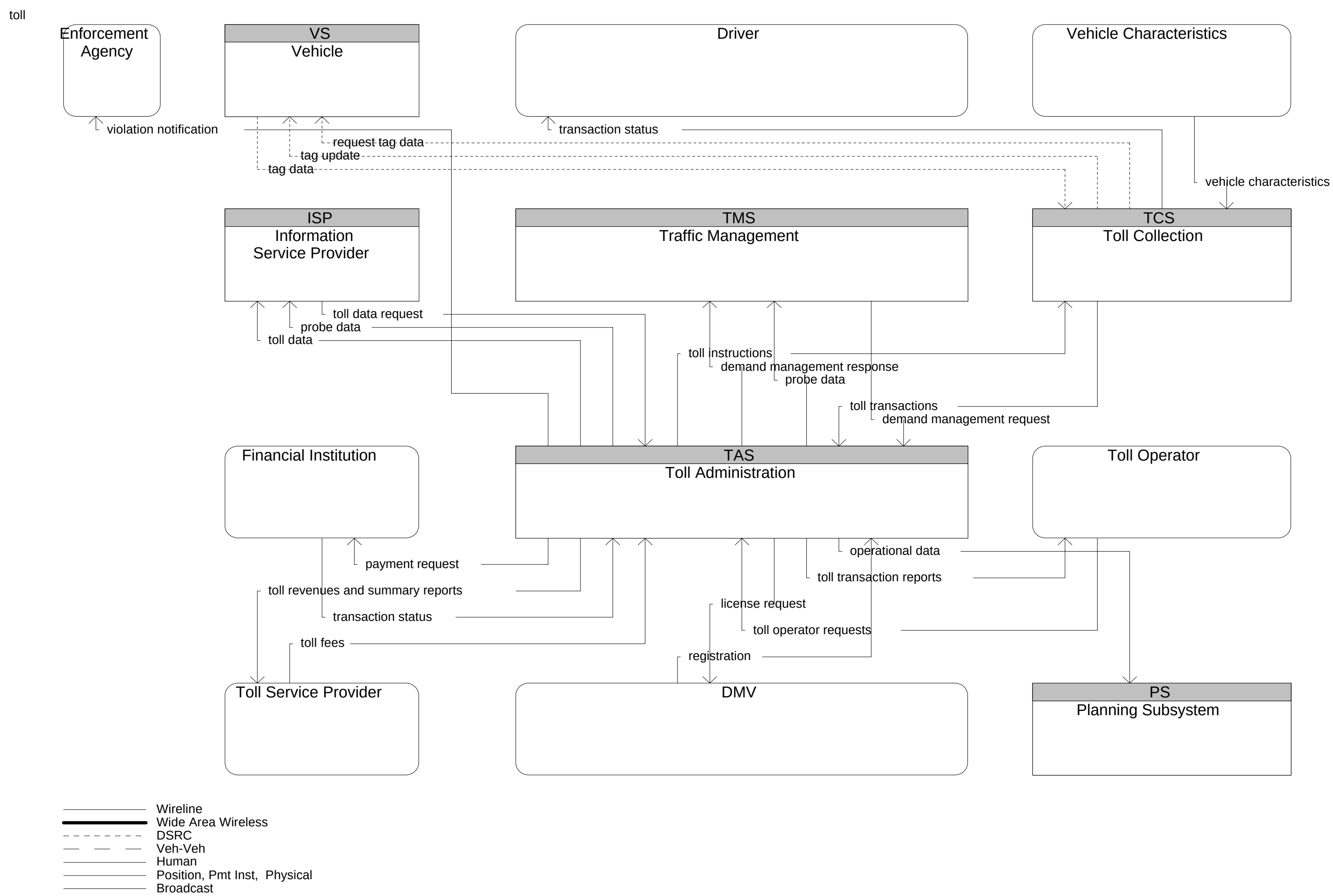


Figure 2.17-2 Architecture Flow Diagram for Toll Administration

2.18 Toll Collection

The Toll Collection Subsystem provides the capability for vehicle operators to pay tolls without stopping their vehicles using locally determined pricing structures and including the capability to implement various variable road pricing policies. Each transaction is accompanied by feedback to the customer which indicates the general status of the customer account. A record of the transactions is provided to the Toll Administration subsystem for reconciliation and so that the customer can periodically receive a detailed record of the transactions.

2.18.1 Alternative Configurations

The Toll Collection Subsystem represents a toll collection facility which may be manned or unmanned Figure 2.18-1. The facility will contain equipment to communicate with vehicles through beacon type technology as well as visual feed back to drivers and violator processing.

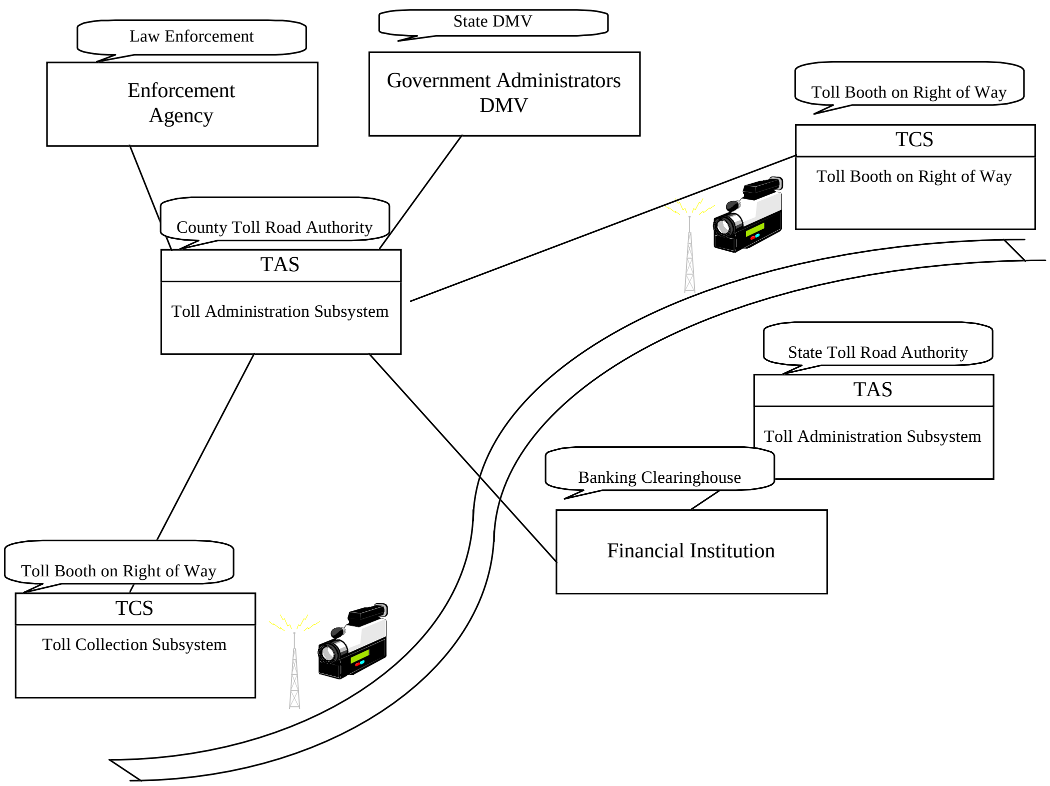


Figure 2.18-1 Alternative Configurations for Toll Collection

2.18.2 Subsystem Equipment Packages and Supporting Process Specifications for TCS

Toll Plaza Toll Collection

This Equipment package provides existing toll plazas the capability to identify properly equipped vehicles and automatically perform toll collection. These capabilities are provided with active tag readers and vehicle identification software running on a workstation type processor. A camera for performing violation identification shall interface to a monitor, the workstation, and database software. Automated account reconciliation and notification to authorities of violations shall be supported using wireline communications.

Process Specifications

7.1.1.1 Read Tag Data for Tolls

Overview: This process shall be responsible for requesting the data from the toll tag being carried on-board the vehicle and used as a payment instrument. If there is no tag or the data it contains cannot be properly read, this process shall provide a message for the vehicle operator to contact the toll authority (or toll system operator). The process shall send a request to other processes to obtain an image of the vehicle. If the vehicle is exiting a closed toll system the data shall be checked by this process to see if it contains an entry point toll segment number. If not present, the process would be referred to another process for off-line resolution. If the toll segment identity is present, it shall be combined with the vehicle characteristics, e.g. size, type, etc. to form the data upon which the toll payment transaction can be based, and the data sent to another process. If the vehicle is entering a closed toll system, the entry point toll segment shall be written onto the tag so that it can be used as the mechanism for charging for the use of the toll road.

7.1.1.10 Determine Advanced Toll Bill

Overview: This process shall be responsible for receiving a request to pay an advanced toll. It shall obtain the price of the toll segment(s) for which advanced payment is being requested from a local data store and shall then forward it to the billing processes. The store of toll prices shall be maintained by another process.

7.1.1.2 Calculate Vehicle Toll

Overview: This process shall be responsible for calculating the toll for the detected vehicle based on the vehicle's characteristics and data obtained from the tag being carried by the vehicle. This process shall calculate the cost of the toll using segment(s) traveled by the vehicle. Segment information is obtained by reading data from a store that contains standard prices for toll segments.

7.1.1.4 Check for Advanced Tolls Payment

Overview: This process shall be responsible for checking to see if the required toll payment has already been made. The process shall determine the existence of an advance payment for the toll segment(s) by comparing the received payment information with that in the store containing the list of advanced payments. If the payment has already been made then the process shall remove the requirement for local billing and remove the record of the advanced payment from the store. Details of each payment transaction shall be sent by the process to another process with the payment information received from the driver removed.

7.1.1.5 Bill Driver for Tolls

Overview: This process shall be responsible for either obtaining payment for the current or advanced toll. The process shall achieve this either by requesting that the toll cost be deducted from the credit being stored by the toll tag that is acting as the payment instrument, or informing the driver that payment for the toll will be debited to the credit identity provided by the tag. Before sending data to the tag, the process shall check that either the credit identity is not already in the list of bad payers, or the stored credit is not less that the toll cost. If either of these conditions is true the process shall obtain an image of the driver and vehicle which can be forwarded to the appropriate enforcement agency via another process. When the appropriate payment transaction has been completed, the toll entry segment identity shall be cleared from the tag so that it can be used the next time that the vehicle is on a toll road. The tag may be in the form of some type of credit or debit card, or an electronic purse. Details of the transaction shall always be sent by this process to the process that manages toll transactions. Where an advanced toll payment is identified, the process shall take no action if the credit identity is on the bad payers list, or the stored credit is less than the toll cost, other than the payment is not confirmed.

7.1.2 Produce Roadside Displays

Overview: This process shall be responsible for driving the displays that tell vehicles whether or not their driver's toll payment has been confirmed or rejected. The process shall receive the data for output via the displays from other processes. The process shall provide its outputs in audible and visual forms, with the latter using an appropriate form of display that shall be easily readable under all lighting conditions and over the range of speeds that vehicles are expected to use when passing through the toll plaza.

7.1.3 Obtain Toll Violator Image

Overview: This process shall be responsible for obtaining an image of a violator for use by other processes. The form which the image data is obtained by this process shall be so accurate that there can be no mistake of the determination of the identity of the vehicle and/or driver, and shall be easily passed on by the other processes to the appropriate law enforcement agency(ies) so that any punitive action that may be taken. The process shall be capable of obtaining an image of the required accuracy under all lighting conditions and over the range of speeds with which vehicles will pass through the toll plaza.

7.1.5 Detect Vehicle for Tolls

Overview: This process shall be responsible for producing a vehicle's characteristics from data received by sensors located at the roadside, at or near the toll collection point. The data shall be sent by the process to another process in a form suitable for use in calculating the toll cost for the vehicle. The process shall ensure that the data includes such things as vehicle size, weight, axle count, type, identifiable features, etc.

2.18.3 Subsystem Interfaces for TCS

Toll Administration => Toll Collection

Physical Architecture Flow Name: toll instructions

Demand management toll pricing information based on current congestion.

Logical Architecture Reference Flow: advanced_toll_needed

This data flow is used within the Provide Electronic Payment Services function to show that an advanced toll cost must be determined. It contains the following data items each of which is defined in its own DDE: credit_identity + stored_credit + toll_route_segments + vehicle_identity.

Logical Architecture Reference Flow: toll_bad_payment_check_response

This data flow is used within the Provide Electronic Payment Services function and contains the response to a request for a check that a driver requesting toll payment is not on the list of bad payers. In the case of commercial vehicles it will be the carrier that is used for the check. The data flow consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + cv_carrier_number.

Toll Collection => Driver

Physical Architecture Flow Name: transaction status

Response to transaction request. Normally dealing with a request for payment.

Logical Architecture Reference Flow: td_toll_payment_confirmed

This data flow is sent to the driver from the Provide Driver and Traveler Services function to confirm that the toll payment transaction has been successfully completed.

Logical Architecture Reference Flow: td_toll_payment_invalid
This data flow is sent to the driver from the Provide Driver and Traveler Services function to indicate that the toll payment transaction is invalid.

Toll Collection => **Toll Administration**

Physical Architecture Flow Name: toll transactions

Detailed list of transactions from a toll station.

Logical Architecture Reference Flow: advanced_toll_transactions
This data flow is used within the Provide Electronic Payment Services function and contains data about advanced toll transactions as they take place for recording in the log of toll transactions. It consists of the following data items each of which is defined in its own DDE: credit_identity + cv_carrier_number + cv_vehicle_number + date + stored_credit + time + toll_cost + toll_plaza_identity + toll_route_segments + vehicle_identity.

Logical Architecture Reference Flow: confirm_advanced_tolls_payment
This data flow is used within the Provide Electronic Payment Services function to confirm that advanced payment for a toll has been made. This may have originated as a request from a traveler making pre-trip planning, as an advanced toll payment from a driver at a toll plaza or parking lot, or from a transit user in the transit network, i.e. on a transit vehicle or at a transit stop. The data flow consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + stored_credit + toll_cost + traveler_identity.

Logical Architecture Reference Flow: current_toll_transactions
This data flow is used within the Provide Electronic Payment Services function and contains data about current toll transactions as they take place for recording in the log of toll transactions. It consists of the following data items each of which is defined in its own DDE: credit_identity + cv_carrier_number + cv_vehicle_number + date + stored_credit + time + toll_cost + toll_plaza_identity + toll_route_segments + vehicle_identity.

Logical Architecture Reference Flow: toll_bad_payment_check_request
This data flow is used within the Provide Electronic Payment Services function and contains a request for a check that a driver requesting toll payment is not on the list of bad payers. In the case of commercial vehicles it will be the carrier that is used for the check. The data flow consists of the following data items each of which is defined in its own DDE: credit_identity + cv_carrier_number.

Logical Architecture Reference Flow: toll_payment_violator_data
This data flow is used within the Provide Electronic Payment Services function and contains data about a toll transaction that was attempted but did not work. It consists of the data items shown below, each of which is defined in its own DDE. For each particular set of data some of the data items may be blank depending on the reason(s) for which the transaction did not work. credit_identity + vehicle_identity + toll_cost.

Logical Architecture Reference Flow: toll_violation_information
This data is used by the Provide Electronic Payment Services functions to send data about a violator of the toll collection processes to the Manage Emergency Services function. This data will contain a digitized video image of the vehicle trying to violate the toll collection process.

Toll Collection => **Vehicle**

Physical Architecture Flow Name: request tag data

Request for tag information including credit identity, stored value card cash, etc.

Logical Architecture Reference Flow: toll_payment_request
This data flow is used within the Provide Electronic Payment Services function and contains the request for the cost of the current toll to be deducted from the credit currently stored by the payment instrument. It is only sent when a value of stored credit has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: toll_cost.

Logical Architecture Reference Flow: toll_tag_data_request
This data flow is used within the Provide Electronic Payment Services function and contains a request for the toll tag data to be read from the store that is held on-board the vehicle.

Physical Architecture Flow Name: tag update

Update data held in tag which can be read at another screening.

Logical Architecture Reference Flow: toll_payment_debited
This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the cost of the current toll will be deducted by the financial institution from the credit identity previously provided by the payment instrument being used by the driver. It is only sent when a credit identity has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: toll_tag_data_clear
This data flow is used within the Provide Electronic Payment Services function and contains the toll tag data from which any toll segment identity has been cleared. The data will have been used to charge for use of the toll road, and is being cleared to enable its use for future charging. The data flow consists of the following data item which is defined in its own DDE: toll_tag_data.

Logical Architecture Reference Flow: toll_tag_data_update
This data flow is used within the Provide Electronic Payment Services function and contains the toll tag data that has been updated. The updated will have loaded the identity of the toll segment at which the vehicle entered the toll road and is for use in charging for the vehicle's use of the toll road. The data flow consists of the following data item which is defined in its own DDE: toll_tag_data.

Vehicle => **Toll Collection**

Physical Architecture Flow Name: tag data

Unique tag ID and related vehicle information for the purposes of payment for services.

Logical Architecture Reference Flow: toll_payment_confirmation
This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the previous request for the cost of the current toll to be deducted from the credit currently stored by the driver's payment instrument has been completed successfully. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: toll_tag_data_collect
This data flow is used within the Provide Electronic Payment Services function and contains the toll tag data that is being collected from on-board the vehicle. This data will be used as the means by which the vehicle will be charged for its use of the toll road and will consist of the following data item which is defined in its own DDE: toll_tag_data.

Vehicle Characteristics => **Toll Collection**

Physical Architecture Flow Name: vehicle characteristics

The physical or visible characteristics of an individual vehicle that can be measured to classify a vehicle and imaged to uniquely identify a vehicle.

Logical Architecture Reference Flow: From_Vehicle_Characteristics
This data flow is sent from the vehicle characteristics terminator. It represents the presence of a vehicle near a sensor, which allows the sensor to create an output that can be used to identify a particular vehicle and its characteristics, such as the number of wheels, size, pollution parameters, etc., for toll payment and parking lot charging purposes. The sensor may also determine the visible characteristics of a vehicle and use that data to obtain information about toll and parking lot charge violators.

2.18.4 Subsystem Architecture Flow Diagram

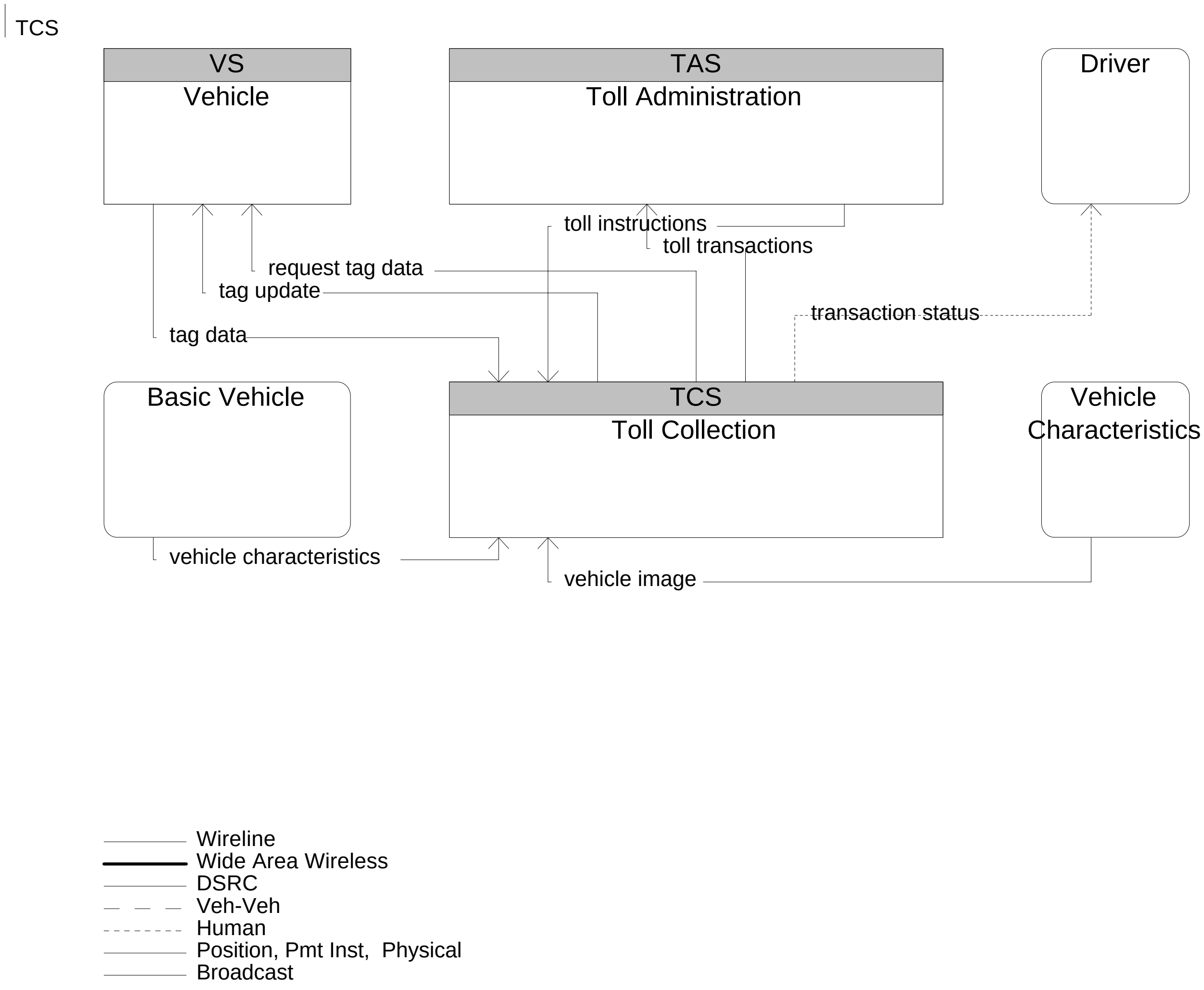


Figure 2.18-2 Architecture Flow Diagram for TCS

2.19 Traffic Management

The Traffic Management Subsystem operates within a traffic management center or other fixed location. This subsystem communicates with the Roadway Subsystem to monitor and manage traffic flow. Incidents are detected and verified and incident information is provided to the Emergency Management Subsystem, travelers (through Roadway Subsystem Highway Advisory Radio and Dynamic Message Signs), and to third party providers. The subsystem supports HOV lane management and coordination, road pricing, and other demand management policies that can alleviate congestion and influence mode selection. The subsystem monitors and manages maintenance work and disseminates maintenance work schedules and road closures. The subsystem also manages reversible lane facilities, and processes probe vehicle information. The subsystem communicates with other Traffic Management Subsystems to coordinate traffic information and control strategies in neighboring jurisdictions. It also coordinates with rail operations to support safer and more efficient highway traffic management at highway-rail intersections. Finally, the Traffic Management Subsystem provides the capabilities to exercise control over those devices utilized for AHS traffic and vehicle control.

2.19.1 Alternative Configurations

Traffic Management Subsystems reside in TMC's and may be coupled within a TMC with other subsystems such as ISP's to interface with the public or EM to handle incidents Figure 2.19-1. TMS subsystems also communicate with other TMS subsystems such as with freeway and surface street TMC's. TMS subsystems may also be organized to accept control from other TMS subsystems under special arrangements.

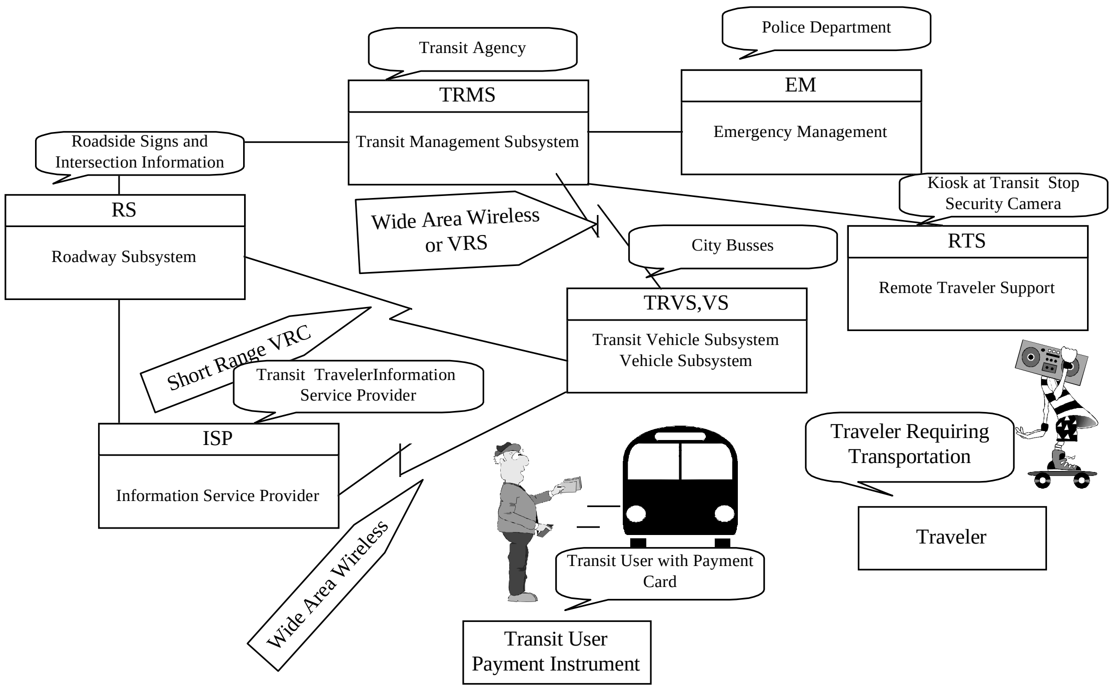


Figure 2.19-1 Alternative Configurations for TMC

2.19.2 Subsystem Equipment Packages and Process Specifications for TMS

Collect Traffic Surveillance

This Equipment package collects, stores, and provides electronic access to the traffic surveillance data.

Process Specifications

1.1.2.1 Process Traffic Data for Storage

Overview: This process shall receive data from other processes and store the data into the long term and current data stores. The data shall comprise sensor data, both smoothed and unsmoothed: processed sensor surveillance data, data sent to control indicators (output devices e.g. intersection controllers, pedestrian controllers, variable message signs, ramp metering equipment), parking lot management data and other street equipment, the status data received from the indicators, plus current traffic conditions, predicted incidents, current incidents, parking lot states, ramp states, link travel times, road conditions provided by vehicle probes, and selected traffic control strategy. The data stored by the process in the current data store shall be the values collected over a relatively short period of time. The data stored in the long term data store shall be retained for a longer period. The data retained in the long term data store may be aggregated so as to reduce the storage requirements for long historical records, the amount of aggregation to be an implementation decision.

1.1.2.2 Process Traffic Data

Overview: This process shall receive and process data from sensors (both traffic and environmental) at the roadway. The process distributes data to Provide Device Control processes that are responsible for freeway, highway rail intersections, parking lot, ramp and road management. It also sends the data to another Provide Traffic Surveillance process for loading into the stores of current and long term data. Information about the various sensors to aid in this processing and distribution of data is accessed from the data store static_data_for_sensor_processing.

1.1.2.3 Update Data Source Static Data

Overview: This process shall be responsible for the maintenance of the store of static data used in the processing of sensor data. This sensor data shall be used to provide traffic surveillance information for use by other processes within the Manage Traffic function. The store shall contain data showing the relationship between sensors and the road and freeway network, i.e. where they are located, to which part(s) of the network their data applies, the type of data, etc. It shall also hold information about the ownership of each link (that is, the agency or entity responsible for collecting and storing surveillance of the link) in the network which shall be used by processes involved in exchanging surveillance information (and optionally control) with other Traffic Management Subsystems (TMS's). The contents of the store shall be provided by the Plan System Deployment function.

1.1.4.1 Retrieve Traffic Data

Overview: This process shall on request retrieve traffic data from the data stores managed by other processes in the Provide Traffic Surveillance facility of the Manage Traffic function. It shall be possible for requests to originate from traffic operations personnel, the media operator, the Manage Demand facility within the Manage Traffic function, the Plan System Deployment function and the Provide Driver and Traveler Services function. With the exception of those from the Manage Demand facility and the Plan System Deployment function, all requests shall be provided by interface processes. The process shall also generate traffic data for output by other processes to in-vehicle signage functions.

1.1.4.2 Provide Traffic Operations Personnel Traffic Data Interface

Overview: This process shall provide the interface through which traffic operations personnel can obtain access to the data stored by other processes in the Provide Traffic Surveillance facility of the Manage Traffic function, and set up the parameters that govern the data that is available to non-traffic operations people via a separate process to the media operator. This stored data shall comprise current and long term (historic) data on traffic conditions, weather conditions and roadside equipment activity, plus prediction estimates of traffic conditions. The data shall apply to some or all of the road and freeway network served by the specific instance of the Manage Traffic function. Where appropriate and/or requested by the traffic operations personnel, the process shall provide the data output in the form of an overlay onto a map of the relevant part(s) of the road and freeway network served by the instance of the function. The process shall obtain the map from a local data store, which it shall enable the traffic operations personnel to update as and when required.

1.1.4.4 Update Traffic Display Map Data

Overview: This process shall provide updates to a store of digitized map data when a request is received from traffic operations personnel via their interface process. The map data shall be for use as the background for displays of traffic data requested by traffic operations personnel and media operators through their respective interface processes. This process shall obtain the new map data from either a specialized data supplier or some other appropriate data source.

HRI Traffic Management

This equipment package monitors highway-rail intersection (HRI) equipment at the roadside which manages highway traffic. Various levels of roadside equipment may be interfaced to, and supported by, this equipment package to include standard speed active warning systems and high speed systems which provide additional information on approaching trains and detect and report on obstructions in the HRI. This equipment package remotely monitors and reports the status of this roadside equipment. A two way interface supports explicitly status requests or remote control plan updates to be generated by this equipment package. Status may also be received periodically in the absence of a request or asynchronously in the event of a detected failure or other unsafe condition at the intersection.

Process Specifications

1.6.2.1 Exchange Data with Rail Operations

Overview: This process is responsible for exchanging routine data with rail operations. Such data being sent to the rail operators includes event schedules, requests for information from the Rail Operators, incident notification based on rail operations messages received from Close_HRI_on_Detection process and hri_priority_message data received from the Manage Alerts and Advisories process. This process receives maintenance schedules, train schedules, and incident notifications from the rail operators. This information is used to develop the rail operations update data that is passed onto the Manage Rail Traffic Control Data process and the rail operations priority data that is sent to the Manage Alerts and Advisories process.

1.6.2.2 Manage Alerts and Advisories

Overview: This process is responsible for acquiring HRI advisory or alert data from rail operations and for providing HRI status to rail operations. The data managed by this process may be time critical, as in the case of alerts or priority messages, or not time critical, as in the case of advisories.

1.6.4.1 Manage HRI Closures

Overview: This process is responsible for coordination and managing of HRI closures at the Traffic management Center. It interfaces with Manage Incidents process to provide incident information and to receive strategy overrides as required by the larger incident management function.

1.6.4.2 Exchange Data with Traffic Management

Overview: This process is responsible for interacting with traffic management processes. It collects data from processes that are within the HRI elements located at the roadside and forwards the data as needed to other processes within traffic management. It also acts as the interface between rail operations and traffic management processes through its interface with the Interact with Rail Operations process.

Rail Operations Coordination

This equipment package provides strategic coordination between rail operations and traffic management centers. It receives train schedules, maintenance schedules, and any other forecast events which will result in highway-rail intersection (HRI) closures from Rail Operations. The provided information is used to develop forecast HRI closure times and durations which may be applied in advanced traffic control strategies or delivered as enhanced traveler information. This equipment package includes the processing and algorithms necessary to derive HRI closure times and the communications capabilities necessary to communicate with rail operations and interface to the traffic control and information distribution capabilities included in other Traffic Management Subsystem equipment packages.

Process Specifications

1.1.3 Generate Predictive Traffic Model

Overview: This process shall be responsible for continually producing and updating a predictive model of the traffic flow conditions in the road or freeway network served by the Manage Traffic function that an instance of this process is allocated to. The prediction shall be based on current surveillance, historic traffic conditions and surveillance, current and predicted incidents, current traffic control strategy, data received from other Traffic Management Subsystems (TMS's) serving other geographic and/or jurisdictional areas, and current and predicted weather conditions. The predictive model of traffic flow produced by this process shall be used by processes in the Manage Traffic function and other ITS functions.

1.6.2.1 Exchange Data with Rail Operations

Overview: This process is responsible for exchanging routine data with rail operations. Such data being sent to the rail operators includes event schedules, requests for information from the Rail Operators, incident notification based on rail operations messages received from Close_HRI_on_Detection process and hri_priority_message data received from the Manage Alerts and Advisories process. This process receives maintenance schedules, train schedules, and incident notifications from the rail operators. This information is used to develop the rail operations update data that is passed onto the Manage Rail Traffic Control Data process and the rail operations priority data that is sent to the Manage Alerts and Advisories process.

1.6.2.3 Manage Rail Traffic Control Data

Overview: This process is responsible for providing and maintaining a current store of rail operations data. The data is assembled from the rail_operations_update information sent by the Exchange Data with Rail Operations process. Queries for this information are received from the Manage Alerts and Advisories process and the Interact with Vehicle Traffic Management processes.

TMC for AHS

This Equipment package provides the capability to exercise control over those devices utilized for AHS traffic and vehicle control.

Process Specifications

3.2.7 Manage AHS Operations

Overview: This process shall be responsible for recording data about vehicles that have requested check-in and check-out for the use of the automated highway system (ahs) lanes, and for receiving ahs control parameters from a process in the Manage Traffic function. The process shall provide a process at the roadside with the vehicle control parameters needed for ahs operation. The process shall keep a log of all ahs check-in and check-out transactions received from the roadside process regardless of whether they are successful or not, and periodically pass this data on to the Plan System Deployment function.

TMC Freeway Management

Control system for efficient freeway management including integration of surveillance information with freeway road geometry, vehicle control such as ramp metering, CMS, HAR. Interface to coordinated traffic subsystems for information dissemination to the public.

Process Specifications

1.1.4.2 Provide Traffic Operations Personnel Traffic Data Interface

Overview: This process shall provide the interface through which traffic operations personnel can obtain access to the data stored by other processes in the Provide Traffic Surveillance facility of the Manage Traffic function, and set up the parameters that govern the data that is available to non-traffic operations people via a separate process to the media operator. This stored data shall comprise current and long term (historic) data on traffic conditions, weather conditions and roadside equipment activity, plus prediction estimates of traffic conditions. The data shall apply to some or all of the road and freeway network served by the specific instance of the Manage Traffic function. Where appropriate and/or requested by the traffic operations personnel, the process shall provide the data output in the form of an overlay onto a map of the relevant part(s) of the road and freeway network served by the instance of the function. The process shall obtain the map from a local data store, which it shall enable the traffic operations personnel to update as and when required.

1.2.2.1 Determine Indicator State for Freeway Management

Overview: This process shall implement selected traffic control strategies and transit vehicle overall priority on some or all of the indicators covering the freeway network served by the Manage Traffic function. It shall implement the strategies only using the indicators (e.g. variable message signs (vms)) specified in the implementation request and shall coordinate its actions with those of the process that controls the road network. The process shall also be capable of monitoring the extra inputs that will arise where tunnels are involved, including the detection of fire and the consequent requirement to re-route traffic.

1.2.3 Determine Ramp State

Overview: This process shall implement the selected control strategies on some or all of the freeway entry ramps in the freeway network served by the Manage Traffic function. It shall implement the strategies only using the ramps that are specified in the implementation request and shall coordinate its actions with those of the process that controls the road network. The process shall base its ramp metering decisions on the data from sensors and ramp meters monitoring traffic conditions upstream and downstream of the ramps. Data from sensors on the ramp used to detect flow past the meter, extent of queues on the ramp, and the presence of vehicles will also be used as the basis for the ramp metering decisions. The decision making process shall use an algorithm to determine the ramp's state based on the ramp control strategy and the sensor input data received.The process shall coordinate its activities with the process responsible for controlling the road(surface street) network.

1.2.4.2 Output Control Data for Freeways

Overview: This process shall transfer data to processes responsible for controlling equipment located at the roadside within the freeway network served by the Manage Traffic function. This data shall contain outputs for use by roadside indicators, such as variable message signs (vms), etc. Data for use by in-vehicle signage equipment shall be sent to another process for output to roadside processes. All data shall have been sent to this process by processes within the Manage Traffic function. This process shall also be responsible for the monitoring of input data showing the way in which the indicators are responding to the data that they are being sent, and the reporting of any errors in their responses as faults to the Collect and Process Indicator Fault Data facility within the Manage Traffic function. All output and input data shall be sent by the process to another process in the Manage Traffic function to be loaded into the store of long term data.

TMC HOV Lane Management

This Equipment package provides the capability to manage HOV lanes by coordinating freeway ramp meters and connector signals with HOV lane usage signals, and giving preferential treatments to HOV lanes to encourage drivers to carpool.

Process Specifications

1.1.2.4 Monitor HOV lane use

Overview: This process shall be responsible for monitoring the use of High Occupancy Vehicle (HOV) lanes and detecting vehicles that do not have the required number of occupants. The process also provides data on HOV lane usage for storage in the stores of current and long term data.

1.2.4.2 Output Control Data for Freeways

Overview: This process shall transfer data to processes responsible for controlling equipment located at the roadside within the freeway network served by the Manage Traffic function. This data shall contain outputs for use by roadside indicators, such as variable message signs (vms), etc. Data for use by in-vehicle signage equipment shall be sent to another process for output to roadside processes. All data shall have been sent to this process by processes within the Manage Traffic function. This process shall also be responsible for the monitoring of input data showing the way in which the indicators are responding to the data that they are being sent, and the reporting of any errors in their responses as faults to the Collect and Process Indicator Fault Data facility within the Manage Traffic function. All output and input data shall be sent by the process to another process in the Manage Traffic function to be loaded into the store of long term data.

5.4.1 Process TM Detected Violations

Overview: This process shall manage the details of high occupancy vehicle (hov) lane use, wrong-way vehicle detection in reversible lanes, and pollution violations reported by the Manage Traffic function. The process shall use the parameters in the store of traffic management (TM) violation (enforcement) data to obtain the vehicle registration data from the appropriate State Department of Motor Vehicles (DMV) office, before sending all of the received information to the correct law enforcement agency. This process shall also maintain the TM enforcement data store, entering all information received from other processes.

TMC Incident Detection

This Equipment package provides the capability to traffic managers to detect and verify incident. This capability includes analyzing and reducing the collected data from traffic surveillance equipment, including predicted incidents and hazardous conditions.

Process Specifications

1.3.1.1 Analyze Traffic Data for Incidents

Overview: This process shall analyze traffic sensor data, vehicle probe data, or video images for anomalies that could indicate occurrence of an incident. The data may be collected from roads(surface street) and/or highways served by the Manage Traffic function. The process shall pass on any anomalies that it detects to another process in the Manage Incidents facility as possible detected incidents.

1.3.1.2 Maintain Static Data for Incident Management

Overview: This process shall maintain the store of static data (data about the location and features of the road or highway links in the transportation network). This data store is used by another process within the Manage Incidents facility to identify and locate incidents. The static data shall be input to this process from another process within the Planning for Deployment function, and it shall be possible for that process to request a copy of the current static data.

1.3.2.1 Store Possible Incident Data

Overview: This process shall receive data on possible incidents from other processes within the Manage Incidents facility and other ITS functions. The process shall load all data that it receives into the store of possible incidents. As part of the loading activity, the process shall enter the data into the relevant parts of the standard format for incident data, and shall assign a level of confidence (e.g. related to the source of the data or time of its detection) to that data.

1.3.2.2 Review and Classify Possible Incidents

Overview: This process shall review input data about possible incidents and provide verification of the incident. The process shall have the capability of using algorithms to automatically identify and verify an incident. The process shall have the capability to classify an incident as current or predicted (e.g. a planned road closure) and shall be load the data into the store of possible incidents as either current or predicted incidents. The process shall report any incidents that it is unable to verify or classify to the traffic operations personnel for manual verification and classification. The process shall allow the traffic operations personnel to request all possible incidents and carry out the verification and classification process manually.

1.3.2.3 Review and Classify Predicted Incidents

Overview: This process shall receive updates of predicted incidents (e.g. planned events) and review the complete list of predicted incidents to determine when an incident should be reclassified from predicted to current. It shall carry out the re-classification process automatically either upon receiving notice that the store of predicted incidents has been updated, or at some periodic rate. The criteria for reclassifying an incident could be that the planned start time of the event has passed. The process shall request details of predicted incidents from the process that manages their data store and shall send details of any new (re- classified) current incidents to the process that manages their data store. It shall also provide updates of predicted and current incidents to other ITS functions, and details of any new predicted incidents to the process responsible for the output of data to vehicle signage functions.

1.3.2.4 Provide Predicted Incidents Store Interface

Overview: This process shall provide the interface to, and manage the use of the store containing details of predicted incidents. The process shall enter details of all new predicted incidents into the store, retrieve details on request, and delete details of an incident when it has been re-classified as a current incident. The process shall be able to receive details of predicted incidents from within the local Manage Incidents facility, and from similar facilities in other Traffic Management Subsystems (TMS's). When requested, the process shall also be able to provide details of its predicted incidents to the Manage Incidents facilities in other TMS's.

1.3.2.5 Provide Current Incidents Store Interface

Overview: This process shall provide the interface to, and manage the use of the store of current incident details. The process shall enter the details of all new current incidents into the store, retrieve details on request, and delete details of incidents when they cease to be current. The process shall be able to receive details of current incidents from within the local Manage Incidents facility, and from similar facilities in other Traffic Management Subsystems (TMS's). When requested, the process shall also be able to provide details of its current incidents to the Manage Incidents facilities in other TMS's.

1.3.4.3 Provide Media Incident Data Interface

Overview: This process shall provide the interface between the Media and the Manage Incidents facility. It shall enable the media to request details of incidents and shall allow transmission of incident information to the media. The media shall also provide raw input data on possible incidents. The process shall enable the output to incorporate a map of the area to which the incidents relate.

TMC Incident Dispatch Coordination/Communication

This Equipment package provides the capability for an incident response formulation function minimizing the incident potential, incident impacts, and/or resources required for incident management including proposing and facilitating the dispatch of emergency response and service vehicles as well as coordinating response with all appropriate cooperating agencies.

Process Specifications

1.1.5 Exchange data with Other Traffic Centers

Overview: This process shall exchange data with similar processes in other Traffic Management Subsystems (TMS's). The other TMS can be adjacent geographically, under control of a different jurisdiction, or part of a more complex hierarchy. The exchange of data shall be triggered by either a request from a remote TMS for data from the TMS to which the Manage Traffic function belongs, or because data needs to be sent from the local TMS to a remote TMS. This data shall include emergency vehicle preemption ('green waves') or special commercial vehicle routes which pass through the local network but have a destination in an area served by a remote TMS, or include data about an incident that has an impact on the traffic conditions in the network served by a remote TMS. The data received from remote TMS's shall be used either to vary the current traffic control strategy to give signal preemption to emergency vehicles or enable the passage of commercial vehicles with unusual loads, or as input to the local traffic predictive model estimation process.

1.2.4.1 Output Control Data for Roads

Overview: This process shall transfer data to processes responsible for controlling equipment located at the roadside within the road (surface street) network served by the Manage Traffic function. This data shall contain outputs for use by roadside indicators, such as intersection and pedestrian controllers, variable message signs (vms), highway advisory radio (har), etc. Data for use by in-vehicle signage equipment shall be sent to another process for output to roadside processes. All data shall be sent to this process by processes within the Manage Traffic function. This process shall also be responsible for the monitoring of input data showing the way in which the indicators are responding to the data that they are being sent, and the reporting of any errors in their responses as faults to the Collect and Process Indicator Fault Data facility within the Manage Traffic function. All output and input data shall be sent by the process to another process in the Manage Traffic function to be loaded into the store of long term data.

1.2.4.2 Output Control Data for Freeways

Overview: This process shall transfer data to processes responsible for controlling equipment located at the roadside within the freeway network served by the Manage Traffic function. This data shall contain outputs for use by roadside indicators, such as variable message signs (vms), etc. Data for use by in-vehicle signage equipment shall be sent to another process for output to roadside processes. All data shall have been sent to this process by processes within the Manage Traffic function. This process shall also be responsible for the monitoring of input data showing the way in which the indicators are responding to the data that they are being sent, and the reporting of any errors in their responses as faults to the Collect and Process Indicator Fault Data facility within the Manage Traffic function. All output and input data shall be sent by the process to another process in the Manage Traffic function to be loaded into the store of long term data.

1.3.2.3 Review and Classify Predicted Incidents

Overview: This process shall receive updates of predicted incidents (e.g. planned events) and review the complete list of predicted incidents to determine when an incident should be reclassified from predicted to current. It shall carry out the re-classification process automatically either upon receiving notice that the store of predicted incidents has been updated, or at some periodic rate. The criteria for reclassifying an incident could be that the planned start time of the event has passed. The process shall request details of predicted incidents from the process that manages their data store and shall send details of any new (re- classified) current incidents to the process that manages their data store. It shall also provide updates of predicted and current incidents to other ITS functions, and details of any new predicted incidents to the process responsible for the output of data to vehicle signage functions.

1.3.3 Respond to Current Incidents

Overview: This process shall provide responses to incidents that become current, i.e. active. Three general strategies for response to incidents can be supported by the process: 1) Operator enters response (there is no set of predetermined responses), 2) the operator selects response from a set of predetermined responses (possibly modifying the response), and 3) the process automatically accesses and implements a response from a set of predetermined responses (while informing the operator of the actions taken). Where predetermined responses are utilized, the operator shall have the capability to view, modify, or override the predetermined response. The predetermined response to each type of incident shall be defined for the process in the store defined_responses_data. If the process cannot find a predetermined response for a particular incident, it shall send the details of the incident to the traffic operations personnel so that they can provide an update to the store of predetermined responses. The process shall output the predetermined responses to an incident when it receives notification from another process in the Manage Incidents facility that a new current incident has occurred. At the same time it shall also output the incident data to the process responsible for providing broadcast data to roadside processes. The other process in the Manage Incidents facility shall also provide details of incidents that have ceased to be current (terminated) so that this process can send out data to clear the actions requested and roadside broadcast information output in response to its occurrence.

1.3.4.1 Retrieve Incident Data

Overview: This process shall retrieve incident data from the stores of predicted and current incidents that are managed by other processes in the Manage Incidents facility of the Manage Traffic function. The process shall retrieve data as the result of a request which may come from the traffic operations personnel or the media operator. The output shall be returned to the source of the request, except where the media operator has specified that the data should be output to the media system.

1.3.4.2 Provide Traffic Operations Personnel Incident Data Interface

Overview: This process shall provide the interface between the traffic operations personnel and the Manage Incidents facility of the Manage Traffic function. It shall enable the personnel to request and amend details of predicted and current incidents and predetermined incident responses, obtain and control incident video image data and manually re-classify incidents as possible, predicted or current. It shall also output to the traffic operations personnel incident details to which no predetermined response currently exists. The process shall support inputs from and outputs to the traffic operations personnel Where appropriate and/or requested by the traffic operations personnel, the process shall provide the output' display' in a form incorporating a map of the relevant part(s) of the road and freeway network served by the function. The process shall obtain the map from a local data store, which it shall request to be updated by another process as and when required.

1.3.4.4 Update Incident Display Map Data

Overview: This process shall provide updates to the store of digitized map data used with displays of incident data produced by processes in the Manage Incidents facility of the Manage Traffic function. The process shall obtain the new data from a map provider or other appropriate data source, on receiving an update request from the traffic operations personnel interface process within the Manage Incidents facility.

1.3.4.5 Manage Resources for Incidents

Overview: This process shall provide the capability for the Manage Traffic function to generate and receive requests for resources in responding to incidents. The process shall provide the capability for traffic operations personnel to request resources from the Construction and Maintenance to provide equipment and support for incident response and clean up. The process shall be able to receive resource requests from the Manage Emergency function and respond with the status of the response by Construction and Maintenance or the traffic operations personnel.

1.3.5 Manage Possible Predetermined Responses Store

Overview: This process shall manage the data store containing possible predetermined responses to incidents used within the Manage Incidents facility. These responses shall be those that another process within the facility has found to be worth including in the store of predetermined responses from an analysis of the incident response log. This process shall enable retrieval of the data from the store for presentation to traffic operations personnel and its possible transfer to the process that manages the store of predetermined incident responses that are actually used by other processes in the Manage Incidents facility.

1.3.6 Manage Predetermined Incident Response Data

Overview: This process shall manage data held in the store of predetermined incident responses that are used by processes within the Manage Incidents facility of the Manage Traffic function. The process shall provide details of the current predetermined responses in response to requests from traffic operations personnel, and shall also update the store with new responses received from the process that manages the store of possible predetermined responses.

1.3.7 Analyze Incident Response Log

Overview: This process shall analyze the data in the log of incident responses within the Manage Incidents facility of the Manage Traffic functions. The process shall analyze the log so that possible standard predetermined incident responses can be identified from the data in the incident_response_log data store. Any such possible standard predetermined responses that are identified shall be passed by this process to the process that manages the store of possible predetermined responses.

TMC Input to In-Vehicle Signing

This Equipment package shall provide the capability to allow traffic managers input to operation and maintenance of the roadway vehicle signing devices.

Process Specifications

1.2.4.3 Output In-vehicle Signage Data

Overview: This process shall format and output data for use by roadside processes in creating in-vehicle signage. This process supports a full range of functionality for in-vehicle signage (from display of signage to location specific advisory data). The process shall be capable of outputting some or all of the following advisory data: link state data, current incidents, planned events, and highway rail intersection status. The process shall be capable of outputting some or all of the following signage data: dynamic message sign contents or fixed signage. The data shall be structured by this process so that it can be output by each roadside process to vehicles for use by in-vehicle signage equipment.

TMC Multi-Modal Coordination

This Equipment package provides the capability of signal control at the traffic management subsystem to provide signal priority for transit vehicles.

Process Specifications

1.2.2.1 Determine Indicator State for Freeway Management

Overview: This process shall implement selected traffic control strategies and transit vehicle overall priority on some or all of the indicators covering the freeway network served by the Manage Traffic function. It shall implement the strategies only using the indicators (e.g. variable message signs (vms)) specified in the implementation request and shall coordinate its actions with those of the process that controls the road network. The process shall also be capable of monitoring the extra inputs that will arise where tunnels are involved, including the detection of fire and the consequent requirement to re-route traffic.

1.2.2.2 Determine Indicator State for Road Management

Overview: This process shall implement selected traffic control strategies and transit priority on some or all of the indicators covering the road (surface street) network served by the Manage Traffic function. It shall implement the strategies only using the indicators (intersection and pedestrian controllers, variable message signs (vms), etc.) that are specified in the implementation request and shall coordinate its actions with those of the processes that control the freeway network and the ramps that give access to the freeway network.

1.2.3 Determine Ramp State

Overview: This process shall implement the selected control strategies on some or all of the freeway entry ramps in the freeway network served by the Manage Traffic function. It shall implement the strategies only using the ramps that are specified in the implementation request and shall coordinate its actions with those of the process that controls the road network. The process shall base its ramp metering decisions on the data from sensors and ramp meters monitoring traffic conditions upstream and downstream of the ramps. Data from sensors on the ramp used to detect flow past the meter, extent of queues on the ramp, and the presence of vehicles will also be used as the basis for the ramp metering decisions. The decision making process shall use an algorithm to determine the ramp's state based on the ramp control strategy and the sensor input data received. The process shall coordinate its activities with the process responsible for controlling the road(surface street) network.

1.4.2 Collect Demand Forecast Data

Overview: This process shall collect data from other ITS functions for use as input to the demand forecasting process within the Manage Demand facility of the Manage Traffic function. The process shall support data retrieval from other functions on request from the traffic operations personnel and through the receipt of unsolicited data from ITS functions. It shall load all the data that it receives in a consistent format into the input store used by the demand forecasting process.

TMC Probe Information Collection

This Equipment package provides the capability to accept and process probe vehicle information. This capability shall be provided through the use of additional hardware and probe vehicle control and tracking software.

Process Specifications

1.1.2.5 Process Tag/AVL Data for Link Time Data

Overview: This process shall be responsible for processing tag and AVL data collected from roadside readers or obtained from an analysis of toll transaction records. The process shall receive the processed tag and AVL data from the data collection process and shall calculate the travel time for the links under tag surveillance that have been traveled by the vehicles carrying the tags. This shall be achieved by noting the successive times at which the tag data is received and calculating the travel time from the difference. The data obtained from the toll tag transaction record analysis and/or tag reader locations shall not need any further processing as it shall contain the average travel times between successive toll collection plazas and tag reading locations. The process shall maintain a data store which contains the average travel time for each link in the road and freeway network under tag surveillance calculated from the previously described data. Calculation of the actual average values shall employ some type of aggregation processing (e.g. smoothing or similar technique) and be stored for differing time categories (e.g. times of day, day of week, holidays) in periodic increments. The current delay time for a link shall be the difference between current travel time value and the aggregate processed (e.g. average) value for that time category.

1.1.4.1 Retrieve Traffic Data

Overview: This process shall on request retrieve traffic data from the data stores managed by other processes in the Provide Traffic Surveillance facility of the Manage Traffic function. It shall be possible for requests to originate from traffic operations personnel, the media operator, the Manage Demand facility within the Manage Traffic function, the Plan System Deployment function and the Provide Driver and Traveler Services function. With the exception of those from the Manage Demand facility and the Plan System Deployment function, all requests shall be provided by interface processes. The process shall also generate traffic data for output by other processes to in-vehicle signage functions.

TMC Regional Traffic Control

This Equipment package provides capabilities in addition to those provided by the TMC Basic Signal Control Equipment package for analyzing, controlling, and optimizing area-wide traffic flow. These capabilities provide for wide area optimization integrating control of a network signal system with control of freeway, considering current demand as well as expected demand with a goal of providing the capability for real-time traffic adaptive control while balancing inter-jurisdictional control issues to achieve regional solutions. These capabilities are best provided using a Traffic Management Center (TMC) to monitor and manage freeway ramp meters and intersection traffic signals and software to process traffic information and implement traffic management measures (e.g., ramp metering, signalization, and traffic coordination between both local and regional jurisdiction). The TMC shall be able to communicate with other TMCs in order to receive and transmit traffic information on other jurisdictions within the region.

Process Specifications

1.1.4.2 Provide Traffic Operations Personnel Traffic Data Interface

Overview: This process shall provide the interface through which traffic operations personnel can obtain access to the data stored by other processes in the Provide Traffic Surveillance facility of the Manage Traffic function, and set up the parameters that govern the data that is available to non-traffic operations people via a separate process to the media operator. This stored data shall comprise current and long term (historic) data on traffic conditions, weather conditions and roadside equipment activity, plus prediction estimates of traffic conditions. The data shall apply to some or all of the road and freeway network served by the specific instance of the Manage Traffic function. Where appropriate and/or requested by the traffic operations personnel, the process shall provide the data output in the form of an overlay onto a map of the relevant part(s) of the road and freeway network served by the instance of the function. The process shall obtain the map from a local data store, which it shall enable the traffic operations personnel to update as and when required.

1.1.5 Exchange data with Other Traffic Centers

Overview: This process shall exchange data with similar processes in other Traffic Management Subsystems (TMS's). The other TMS can be adjacent geographically, under control of a different jurisdiction, or part of a more complex hierarchy. The exchange of data shall be triggered by either a request from a remote TMS for data from the TMS to which the Manage Traffic function belongs, or because data needs to be sent from the local TMS to a remote TMS. This data shall include emergency vehicle preemption ('green waves') or special commercial vehicle routes which pass through the local network but have a destination in an area served by a remote TMS, or include data about an incident that has an impact on the traffic conditions in the network served by a remote TMS. The data received from remote TMS's shall be used either to vary the current traffic control strategy to give signal preemption to emergency vehicles or enable the passage of commercial vehicles with unusual loads, or as input to the local traffic predictive model estimation process.

1.2.4.1 Output Control Data for Roads

Overview: This process shall transfer data to processes responsible for controlling equipment located at the roadside within the road (surface street) network served by the Manage Traffic function. This data shall contain outputs for use by roadside indicators, such as intersection and pedestrian controllers, variable message signs (vms), highway advisory radio (har), etc. Data for use by in-vehicle signage equipment shall be sent to another process for output to roadside processes. All data shall be sent to this process by processes within the Manage Traffic function. This process shall also be responsible for the monitoring of input data showing the way in which the indicators are responding to the data that they are being sent, and the reporting of any errors in their responses as faults to the Collect and Process Indicator Fault Data facility within the Manage Traffic function. All output and input data shall be sent by the process to another process in the Manage Traffic function to be loaded into the store of long term data.

1.2.4.2 Output Control Data for Freeways

Overview: This process shall transfer data to processes responsible for controlling equipment located at the roadside within the freeway network served by the Manage Traffic function. This data shall contain outputs for use by roadside indicators, such as variable message signs (vms), etc. Data for use by in-vehicle signage equipment shall be sent to another process for output to roadside processes. All data shall have been sent to this process by processes within the Manage Traffic function. This process shall also be responsible for the monitoring of input data showing the way in which the indicators are responding to the data that they are being sent, and the reporting of any errors in their responses as faults to the Collect and Process Indicator Fault Data facility within the Manage Traffic function. All output and input data shall be sent by the process to another process in the Manage Traffic function to be loaded into the store of long term data.

TMC Reversible Lane Management
This Equipment package provides the capability for access and management of reversible lane facilities, including the direction of traffic flow changes during the day, especially between the peak hours and dedication of more lanes to the congestion direction during special events.

Process Specifications
1.1.2.7 Monitor Reversible Lanes

Overview: This process shall be responsible for monitoring the use of reversible lanes and detecting wrong-way vehicles in reversible lanes. The process shall monitor sensor data and video images from the reversible lanes, and use this information along with the lane status (which direction it is currently operating) to identify when a vehicle is traveling in the wrong direction on the reversible lane.

1.2.4.2 Output Control Data for Freeways

Overview: This process shall transfer data to processes responsible for controlling equipment located at the roadside within the freeway network served by the Manage Traffic function. This data shall contain outputs for use by roadside indicators, such as variable message signs (vms), etc. Data for use by in-vehicle signage equipment shall be sent to another process for output to roadside processes. All data shall have been sent to this process by processes within the Manage Traffic function. This process shall also be responsible for the monitoring of input data showing the way in which the indicators are responding to the data that they are being sent, and the reporting of any errors in their responses as faults to the Collect and Process Indicator Fault Data facility within the Manage Traffic function. All output and input data shall be sent by the process to another process in the Manage Traffic function to be loaded into the store of long term data.

1.3.4.2 Provide Traffic Operations Personnel Incident Data Interface

Overview: This process shall provide the interface between the traffic operations personnel and the Manage Incidents facility of the Manage Traffic function. It shall enable the personnel to request and amend details of predicted and current incidents and predetermined incident responses, obtain and control incident video image data and manually re-classify incidents as possible, predicted or current. It shall also output to the traffic operations personnel incident details to which no predetermined response currently exists. The process shall support inputs from and outputs to the traffic operations personnel Where appropriate and/or requested by the traffic operations personnel, the process shall provide the output 'display' in a form incorporating a map of the relevant part(s) of the road and freeway network served by the function. The process shall obtain the map from a local data store, which it shall request to be updated by another process as and when required.

5.4.1 Process TM Detected Violations

Overview: This process shall manage the details of high occupancy vehicle (hov) lane use, wrong-way vehicle detection in reversible lanes, and pollution violations reported by the Manage Traffic function. The process shall use the parameters in the store of traffic management (TM) violation (enforcement) data to obtain the vehicle registration data from the appropriate State Department of Motor Vehicles (DMV) office, before sending all of the received information to the correct law enforcement agency. This process shall also maintain the TM enforcement data store, entering all information received from other processes.

TMC Road Weather Monitoring
This equipment package assimilates current and forecast road conditions and weather information using a combination of weather service information and an array of environmental sensors deployed on and about the roadway. The collected road weather information is monitored and analyzed to detect and forecast environmental hazards such as icy road conditions and dense fog. This information can be used to more effectively deploy road maintenance resources, issue general traveler advisories, and support location specific warnings to drivers.

Process Specifications

1.1.2.1 Process Traffic Data for Storage

Overview: This process shall receive data from other processes and store the data into the long term and current data stores. The data shall comprise sensor data, both smoothed and unsmoothed: processed sensor surveillance data, data sent to control indicators (output devices e.g. intersection controllers, pedestrian controllers, variable message signs, ramp metering equipment), parking lot management data and other street equipment, the status data received from the indicators, plus current traffic conditions, predicted incidents, current incidents, parking lot states, ramp states, link travel times, road conditions provided by vehicle probes, and selected traffic control strategy. The data stored by the process in the current data store shall be the values collected over a relatively short period of time. The data stored in the long term data store shall be retained for a longer period. The data retained in the long term data store may be aggregated so as to reduce the storage requirements for long historical records, the amount of aggregation to be an implementation decision.

1.1.2.2 Process Traffic Data

Overview: This process shall receive and process data from sensors (both traffic and environmental) at the roadway. The process distributes data to Provide Device Control processes that are responsible for freeway, highway rail intersections, parking lot, ramp and road management. It also sends the data to another Provide Traffic Surveillance process for loading into the stores of current and long term data. Information about the various sensors to aid in this processing and distribution of data is accessed from the data store static_data_for_sensor_processing.

1.1.4.1 Retrieve Traffic Data

Overview: This process shall on request retrieve traffic data from the data stores managed by other processes in the Provide Traffic Surveillance facility of the Manage Traffic function. It shall be possible for requests to originate from traffic operations personnel, the media operator, the Manage Demand facility within the Manage Traffic function, the Plan System Deployment function and the Provide Driver and Traveler Services function. With the exception of those from the Manage Demand facility and the Plan System Deployment function, all requests shall be provided by interface processes. The process shall also generate traffic data for output by other processes to in-vehicle signage functions.

1.3.2.1 Store Possible Incident Data

Overview: This process shall receive data on possible incidents from other processes within the Manage Incidents facility and other ITS functions. The process shall load all data that it receives into the store of possible incidents. As part of the loading activity, the process shall enter the data into the relevant parts of the standard format for incident data, and shall assign a level of confidence (e.g. related to the source of the data or time of its detection) to that data.

1.3.2.2 Review and Classify Possible Incidents

Overview: This process shall review input data about possible incidents and provide verification of the incident. The process shall have the capability of using algorithms to automatically identify and verify an incident. The process shall have the capability to classify an incident as current or predicted (e.g. a planned road closure) and shall be load the data into the store of possible incidents as either current or predicted incidents. The process shall report any incidents that it is unable to verify or classify to the traffic operations personnel for manual verification and classification. The process shall allow the traffic operations personnel to request all possible incidents and carry out the verification and classification process manually.

1.3.4.2 Provide Traffic Operations Personnel Incident Data Interface

Overview: This process shall provide the interface between the traffic operations personnel and the Manage Incidents facility of the Manage Traffic function. It shall enable the personnel to request and amend details of predicted and current incidents and predetermined incident responses, obtain and control incident video image data and manually re-classify incidents as possible, predicted or current. It shall also output to the traffic operations personnel incident details to which no predetermined response currently exists. The process shall support inputs from and outputs to the traffic operations personnel Where appropriate and/or requested by the traffic operations personnel, the process shall provide the output 'display' in a form incorporating a map of the relevant part(s) of the road and freeway network served by the function. The process shall obtain the map from a local data store, which it shall request to be updated by another process as and when required.

1.3.4.5 Manage Resources for Incidents

Overview: This process shall provide the capability for the Manage Traffic function to generate and receive requests for resources in responding to incidents. The process shall provide the capability for traffic operations personnel to request resources from the Construction and Maintenance to provide equipment and support for incident response and clean up. The process shall be able to receive resource requests from the Manage Emergency function and respond with the status of the response by Construction and Maintenance or the traffic operations personnel.

TMC Signal Control

This Equipment package provides the capability for traffic managers to monitor and manage the traffic flow at signalized intersections. This capability includes analyzing and reducing the collected data from traffic surveillance equipment and developing and implementing control plans for signalized intersections. Control plans may be developed and implemented that coordinate signals at many intersections under the domain of a single traffic management subsystem.

In advanced implementations, this package collects route planning information and integrates and uses this information in predicting future traffic conditions and optimizing the traffic control strategy for these conditions. These capabilities are achieved through real-time communication of logged routes from an Information Service Provider. The planned control strategies can be passed back to the Information Service Provider so that the intended strategies can be reflected in future route planning.

Process Specifications

1.1.2.2 Process Traffic Data

Overview: This process shall receive and process data from sensors (both traffic and environmental) at the roadway. The process distributes data to Provide Device Control processes that are responsible for freeway, highway rail intersections, parking lot, ramp and road management. It also sends the data to another Provide Traffic Surveillance process for loading into the stores of current and long term data. Information about the various sensors to aid in this processing and distribution of data is accessed from the data store static_data_for_sensor_processing.

1.1.4.2 Provide Traffic Operations Personnel Traffic Data Interface

Overview: This process shall provide the interface through which traffic operations personnel can obtain access to the data stored by other processes in the Provide Traffic Surveillance facility of the Manage Traffic function, and set up the parameters that govern the data that is available to non-traffic operations people via a separate process to the media operator. This stored data shall comprise current and long term (historic) data on traffic conditions, weather conditions and roadside equipment activity, plus prediction estimates of traffic conditions. The data shall apply to some or all of the road and freeway network served by the specific instance of the Manage Traffic function. Where appropriate and/or requested by the traffic operations personnel, the process shall provide the data output in the form of an overlay onto a map of the relevant part(s) of the road and freeway network served by the instance of the function. The process shall obtain the map from a local data store, which it shall enable the traffic operations personnel to update as and when required.

1.2.1 Select Strategy

Overview: This process shall select the appropriate traffic control strategy to be implemented over a road and/or freeway section served by the specific instance of the Manage Traffic function. The strategy shall be selected by the process from a number that are available, e.g. adaptive control, fixed time control, local operations. The selected strategy shall be passed by the process to the actual control processes for implementation according to the part of the network to which it is to be applied, i.e. surface roads, freeways (i.e. limited access roads), ramps and/or parking lots. The definition of strategy can be extended to include a strategy for the operations of sensors such as video cameras used to provide traffic surveillance data. The process shall make it possible for the current strategy selection to be modified to accommodate the effects of such things as incidents, emergency vehicle preemption, the passage of commercial vehicles with unusual loads, equipment faults and overrides from the traffic operations personnel. The strategy for control of freeways and parking lots is through use of vms signs and lane indicators. The strategy for control of ramps is through the timing plans for ramp meters. The selected strategy shall be sent to the process within the Provide Traffic Surveillance facility responsible for maintaining the store of long term data.

1.2.2.2 Determine Indicator State for Road Management

Overview: This process shall implement selected traffic control strategies and transit priority on some or all of the indicators covering the road (surface street) network served by the Manage Traffic function. It shall implement the strategies only using the indicators (intersection and pedestrian controllers, variable message signs (vms), etc.) that are specified in the implementation request and shall coordinate its actions with those of the processes that control the freeway network and the ramps that give access to the freeway network.

1.2.4.1 Output Control Data for Roads

Overview: This process shall transfer data to processes responsible for controlling equipment located at the roadside within the road (surface street) network served by the Manage Traffic function. This data shall contain outputs for use by roadside indicators, such as intersection and pedestrian controllers, variable message signs (vms), highway advisory radio (har), etc. Data for use by in-vehicle signage equipment shall be sent to another process for output to roadside processes. All data shall be sent to this process by processes within the Manage Traffic function. This process shall also be responsible for the monitoring of input data showing the way in which the indicators are responding to the data that they are being sent, and the reporting of any errors in their responses as faults to the Collect and Process Indicator Fault Data facility within the Manage Traffic function. All output and input data shall be sent by the process to another process in the Manage Traffic function to be loaded into the store of long term data.

TMC Toll/Parking Coordination
This Equipment package provides the transportation management center with the capability to transform and transmit network traffic congestion information to the Toll Administration or Parking Management so that dynamic pricing for demand management is possible. Communications shall be supported using a wireline modem.

Process Specifications
1.4.4 Implement Demand Management Policy

Overview: This process shall implement the traffic and travel demand forecast data produced by the demand forecasting process in the Manage Travel Demand facility of the Manage Traffic function. The new demand forecast data shall be implemented in such a way that it can influence the demand from travelers for various types of services provided by ITS functions. The process shall when required, request changes to transit services, and/or the charges for tolls, and/or the use of parking lot spaces (as per the locally determined demand policy). It shall communicate the results of its policy implementation to the process that provides the interface to the traffic operations personnel.

TMC Traffic Information Dissemination
This Equipment package provides the capability to disseminate incident related information to travelers, potential travelers, and private information service providers. These capabilities shall be provided using a workstation type processor within a facility connected to traveler information providers by utilizing existing wireline links.

Process Specifications
1.1.4.1 Retrieve Traffic Data

Overview: This process shall on request retrieve traffic data from the data stores managed by other processes in the Provide Traffic Surveillance facility of the Manage Traffic function. It shall be possible for requests to originate from traffic operations personnel, the media operator, the Manage Demand facility within the Manage Traffic function, the Plan System Deployment function and the Provide Driver and Traveler Services function. With the exception of those from the Manage Demand facility and the Plan System Deployment function, all requests shall be provided by interface processes. The process shall also generate traffic data for output by other processes to in-vehicle signage functions.

1.1.4.2 Provide Traffic Operations Personnel Traffic Data Interface

Overview: This process shall provide the interface through which traffic operations personnel can obtain access to the data stored by other processes in the Provide Traffic Surveillance facility of the Manage Traffic function, and set up the parameters that govern the data that is available to non-traffic operations people via a separate process to the media operator. This stored data shall comprise current and long term (historic) data on traffic conditions, weather conditions and roadside equipment activity, plus prediction estimates of traffic conditions. The data shall apply to some or all of the road and freeway network served by the specific instance of the Manage Traffic function. Where appropriate and/or requested by the traffic operations personnel, the process shall provide the data output in the form of an overlay onto a map of the relevant part(s) of the road and freeway network served by the instance of the function. The process shall obtain the map from a local data store, which it shall enable the traffic operations personnel to update as and when required.

1.1.4.3 Provide Direct Media Traffic Data Interface

Overview: This process shall be responsible for providing the interface between the media and the process responsible for obtaining data from the stores of traffic data maintained by other processes within the Provide Traffic Surveillance facility of the Manage Traffic function. The process shall enable the media to request and be provided with current, long term (historic) and predicted traffic data. The data may be provided in one or more formats: as a data stream, as processed and displayed to Traffic Operations Personnel (e.g. graphical summaries of link speeds),or as a display (with data included on a map of relevant part(s) of the road and freeway served by the Manage Traffic function. The media shall only be able to request and see displayed that data that the traffic operations personnel have made available, through the use of the definition in the traffic data media parameters.

1.3.4.3 Provide Media Incident Data Interface

Overview: This process shall provide the interface between the Media and the Manage Incidents facility. It shall enable the media to request details of incidents and shall allow transmission of incident information to the media. The media shall also provide raw input data on possible incidents. The process shall enable the output to incorporate a map of the area to which the incidents relate.

TMC Traffic Network Performance Evaluation

This Equipment package provides the capability to predict travel demand patterns to support traffic flow optimization, demand management, and incident management. This Equipment package requires the data collected by surveillance Equipment packages as well as input from other management subsystems including the ISP Subsystem, Transit Management Subsystem.

Process Specifications

1.1.2.1 Process Traffic Data for Storage

Overview: This process shall receive data from other processes and store the data into the long term and current data stores. The data shall comprise sensor data, both smoothed and unsmoothed: processed sensor surveillance data, data sent to control indicators (output devices e.g. intersection controllers, pedestrian controllers, variable message signs, ramp metering equipment), parking lot management data and other street equipment, the status data received from the indicators, plus current traffic conditions, predicted incidents, current incidents, parking lot states, ramp states, link travel times, road conditions provided by vehicle probes, and selected traffic control strategy. The data stored by the process in the current data store shall be the values collected over a relatively short period of time. The data stored in the long term data store shall be retained for a longer period. The data retained in the long term data store may be aggregated so as to reduce the storage requirements for long historical records, the amount of aggregation to be an implementation decision.

1.1.2.2 Process Traffic Data

Overview: This process shall receive and process data from sensors (both traffic and environmental) at the roadway. The process distributes data to Provide Device Control processes that are responsible for freeway, highway rail intersections, parking lot, ramp and road management. It also sends the data to another Provide Traffic Surveillance process for loading into the stores of current and long term data. Information about the various sensors to aid in this processing and distribution of data is accessed from the data store static_data_for_sensor_processing.

1.1.3 Generate Predictive Traffic Model

Overview: This process shall be responsible for continually producing and updating a predictive model of the traffic flow conditions in the road or freeway network served by the Manage Traffic function that an instance of this process is allocated to. The prediction shall be based on current surveillance, historic traffic conditions and surveillance, current and predicted incidents, current traffic control strategy, data received from other Traffic Management Subsystems (TMS's) serving other geographic and/or jurisdictional areas, and current and predicted weather conditions. The predictive model of traffic flow produced by this process shall be used by processes in the Manage Traffic function and other ITS functions.

1.1.5 Exchange data with Other Traffic Centers

Overview: This process shall exchange data with similar processes in other Traffic Management Subsystems (TMS's). The other TMS can be adjacent geographically, under control of a different jurisdiction, or part of a more complex hierarchy. The exchange of data shall be triggered by either a request from a remote TMS for data from the TMS to which the Manage Traffic function belongs, or because data needs to be sent from the local TMS to a remote TMS. This data shall include emergency vehicle preemption ('green waves') or special commercial vehicle routes which pass through the local network but have a destination in an area served by a remote TMS, or include data about an incident that has an impact on the traffic conditions in the network served by a remote TMS. The data received from remote TMS's shall be used either to vary the current traffic control strategy to give signal preemption to emergency vehicles or enable the passage of commercial vehicles with unusual loads, or as input to the local traffic predictive model estimation process.

1.2.6.1 Maintain Traffic and Sensor Static Data

Overview: This process shall maintain the store of static and link data used by other processes within the Manage Traffic function. Link data shall also be sent to the Provide Driver and Traveler Services function to enable it to obtain data about links that are not in the geographic area which it serves. The data held in the store maintained by this process shall be provided to it by the Plan System Deployment function, to which this process shall be capable of providing the current data and receiving from it updates to the current data.

1.2.6.2 Provide Static Data Store Output Interface

Overview: This process shall provide updates of static data to other processes in the Provide Traffic Control facility of the Manage Traffic function. An update of the data shall only be provided when this process has been notified by another process that the contents of the store of static data has been changed. This other process shall be responsible for receiving the new data from the Plan System Deployment function and for providing copies of the current data to that function.

1.4.1 Provide Traffic Operations Personnel Demand Interface

Overview: This process shall provide the interface between the traffic operations personnel and the processes and data stores used within the Manage Demand facility of the Manage Traffic function. It shall enable the traffic operations personnel to access the data used as input by the demand forecasting process and the results of that process, to request that the input data be updated, set the policies used as input to the Calculate Forecast Demand process, to request that the demand forecasting process runs, and to run the process that implements the results. Where appropriate and/or requested by the traffic operations personnel, the process shall provide the output in a form that includes a map of the relevant part(s) of the road and freeway network served by the Manage Travel Demand function. The process shall obtain the map from a local data store, which it shall request to be updated by another process when required.

1.4.2 Collect Demand Forecast Data

Overview: This process shall collect data from other ITS functions for use as input to the demand forecasting process within the Manage Demand facility of the Manage Traffic function. The process shall support data retrieval from other functions on request from the traffic operations personnel and through the receipt of unsolicited data from ITS functions. It shall load all the data that it receives in a consistent format into the input store used by the demand forecasting process.

1.4.3 Update Demand Display Map Data

Overview: This process shall provide updates to the store of map data used for displays of forecast traffic and travel demand produced by processes in the Manage Travel Demand facility of the Manage Traffic function. The process shall obtain the new data from a specialist map data supplier or some other appropriate source, on receiving an update request from the traffic operations personnel interface process within the Manage Travel Demand facility.

1.4.4 Implement Demand Management Policy

Overview: This process shall implement the traffic and travel demand forecast data produced by the demand forecasting process in the Manage Travel Demand facility of the Manage Traffic function. The new demand forecast data shall be implemented in such a way that it can influence the demand from travelers for various types of services provided by ITS functions. The process shall when required, request changes to transit services, and/or the charges for tolls, and/or the use of parking lot spaces (as per the locally determined demand policy). It shall communicate the results of its policy implementation to the process that provides the interface to the traffic operations personnel.

1.4.5 Calculate Forecast Demand

Overview: This process shall provide a forecast of traffic and travel demand in the geographic area served by the Manage Traffic function to which this instance of the Manage Travel Demand facility belongs. The process shall base its forecast on the current and predicted traffic levels traveler demand patterns obtained from an analysis of data obtained from elsewhere within the Manage Traffic function and from other ITS functions as well as locally determined demand policy. The process shall produce a demand forecast that changes the way that services are provided by ITS functions according to locally determined demand policy.

Traffic Maintenance
This Equipment package provides monitoring and remote diagnostics of field equipment to detect field equipment failures, issues problem reports, and tracks the repair or replacement of the failed equipment.

Process Specifications
1.1.1.2 Collect and Process Sensor Fault Data

Overview: This process shall be responsible for collecting sensor status, identifying faults, and logging faults that have been detected by processes in other parts of the Manage Traffic function. It shall be possible for the faults to have been detected locally at the sensors, or centrally through communications links with the sensors. The process shall pass on new fault data to another processes for communication to the Construction and Maintenance terminator and shall receive fault clearances from the same terminator. It shall also maintain a store of the current fault state of all sensors. The process shall provide facilities that enable traffic operations personnel to review and update the current fault status of all sensors. Details of faulty and fixed equipment shall be passed by the process to the traffic control strategy selection process so that it can adjust its strategy to take account of the fault(s).

1.2.8.1 Collect Indicator Fault Data

Overview: This process shall collect data about faults in the operation of indicators (e.g. signals, vms, har) that have been detected by processes in other parts of the Manage Traffic function. It shall be possible for the faults to be detected locally at the indicators, or centrally through communications links with the indicators.

1.2.8.2 Maintain Indicator Fault Data Store

Overview: This process shall collect data about indicator faults that have been detected by processes in other parts of the Manage Traffic function. It shall be possible for the faults to have been detected locally at the indicators, or centrally through communications links with the indicators. The process shall pass on new fault data to another process for communication to the Construction and Maintenance terminator and shall receive fault clearances from the same process communicating with that terminator. It shall also maintain a store of the current fault state of all indicators. The process shall provide facilities that enable traffic operations personnel to review and update the current fault status of all indicators. Details of faulty and fixed equipment shall be passed by the process to the traffic control strategy selection process so that it can adjust its strategy to take account of the current fault(s).

1.2.8.3 Provide Indicator Fault Interface for C and M

Overview: This process shall provide an interface for the exchange of data with the Construction and Maintenance terminator. The interface shall be used to both send data containing details of new indicator equipment faults, and to receive clearances when the faults are cleared. The details of new equipment faults and the clearances shall be received from and sent to another process.

1.2.8.4 Provide Traffic Operations Personnel Indicator Fault Interface

Overview: This process shall provide the interface through which traffic operations personnel access data about faults on indicator equipment controlled by the Manage Traffic function. The process shall enable the personnel to monitor all indicator equipment faults that have been detected, and if necessary, amend that data. It shall also enable the traffic operations personnel to manually input faults in cases where they cannot otherwise be detected.

2.19.3 Subsystem Interfaces for TMS

Construction and Maintenance => Traffic Management

Physical Architecture Flow Name: equipment maintenance status

Current status of field equipment maintenance actions.

Logical Architecture Reference Flow: fcm_fault_clearance
This data flow is sent from the construction and maintenance terminator to the Manage Traffic function. It contains a report showing that a particular fault in an indicator has been cleared and that it has been restored to normal operation.

Logical Architecture Reference Flow: fcm_sensor_fault_data
This data flow is sent from the construction and maintenance terminator to the Manage Traffic function. It contains a report showing that a particular fault in a sensor has been cleared and that it has been restored to normal operation.

Physical Architecture Flow Name: maintenance resource response

Current status of maintenance resources included availability and deployment status.

Logical Architecture Reference Flow: fcm_resource_response
This data flow is sent to the Manage Traffic function from the construction and maintenance operations personnel to provide the status of the requested resources by the Construction and Maintenance terminator .

Physical Architecture Flow Name: work zone status

Status of maintenance work zone.

Logical Architecture Reference Flow: fcm_incident_information
This data flow is sent from the construction and maintenance terminator to the Manage Traffic function. It contains information about an incident that is about to be created by the proposed start of road maintenance activity which will affect the flow of traffic on one or more lanes of a road or highway. Information contained is location, number of lanes closed, and duration of closure.

DMV => **Traffic Management**

Physical Architecture Flow Name: registration

Registered owner of vehicle and associated vehicle information.

Logical Architecture Reference Flow: fdmv_traffic_violation_state_identity
This data flow is sent from the department of motor vehicles to the Manage Emergency Services function and contains the identity of the state that is supplying the requested vehicle registration data to enable a high occupancy vehicle (hov) lane or pollution violation to be processed.

Logical Architecture Reference Flow: fdmv_traffic_violation_vehicle_registration
This data flow is sent from the department of motor vehicles to the Manage Emergency Services function and contains the requested vehicle registration data to enable a high occupancy vehicle (hov) lane or pollution violation to be processed.

Emergency Management => **Traffic Management**

Physical Architecture Flow Name: emergency traffic control request

Special request to preempt the current traffic control strategy in effect at one or more signalized intersections or highway segments. For example, this flow can request all signals to red-flash, request a green wave for an emergency vehicle, or request another special traffic control plan.

Logical Architecture Reference Flow: emergency_traffic_control_request
This data flow is sent from the Manage Emergency Services function to the Manage Traffic function. It contains a list of the route segments that have been provided for use by an emergency vehicle, together with the arrival time at each segment. The data will be used by the Manage Traffic function to generate changes to the current traffic management strategy to give the emergency vehicle priority. The data flow consists of the following data items each of which is defined in its own DDE: date + list_size + list_size{route_segment_identity + route_segment_estimated_arrival_time} + time.

Physical Architecture Flow Name: incident information

Notification of existence of incident and expected severity, location, time and nature of incident.

Logical Architecture Reference Flow: incident_details
This data flow is sent from the Manage Emergency Services function to the Manage Traffic function and provides information about current incidents. It contains the following data items each of which is defined in its own DDE: incident_number + incident_location + incident_start_time + incident_duration + incident_type + incident_severity.

Physical Architecture Flow Name: incident response status

Status of the current incident response including traffic management strategies implemented at the site (e.g., closures, diversions, traffic signal control overrides).

Logical Architecture Reference Flow: incident_response_status
This data flow provides the current status of an incident response indicating site management strategies in effect, incident clearance status, the incident command structure that is in place, and points of contact.

Physical Architecture Flow Name: remote surveillance control

The control commands used to remotely operate another center's sensors or surveillance equipment so that roadside surveillance assets can be shared by more than one agency.

Logical Architecture Reference Flow: remote_video_image_control
This data flow is used within the Manage Traffic function. It is a request from the Emergency Management function to control closed circuit televisions(cctv) images of incidents that occurred on roadways.

Physical Architecture Flow Name: resource request

A request for traffic management resources to implement special traffic control measures, assist in clean up, verify an incident, etc.

Logical Architecture Reference Flow: resource_request
This data flow is used within the Manage Emergency Services function and contains data for the request for traffic management resources to implement special traffic control measures, assist in clean up, etc. It consists of the following data item which is defined in its own DDE. traffic_resource_request.

Emissions Management => **Traffic Management**

Physical Architecture Flow Name: wide-area statistical pollution information

Aggregated region-wide measured emissions data and possible pollution incident information.

Logical Architecture Reference Flow: pollution_incident
This data flow is used within the Manage Traffic function and contains details of a current or predicted pollution incident. The incident type will be set to the three character code for a pollution incident and will depend on the type of pollutant that is involved. The data flow consists of the following data items each of which is defined in its own DDE: incident_start_time + incident_duration + incident_location + incident_severity + incident_type.

Logical Architecture Reference Flow: pollution_state_data
This data flow is used within the Manage Traffic function as a means of transferring current pollution data from the Manage Emissions facility to the Manage Demand facility. It contains data about the current levels of pollution obtained from the store of pollution data and consists of the following data items each of which is defined in its own DDE: current_ozone_pollution + current_nitrous_oxide_pollution + current_sulfur_dioxide_pollution + current_hydrocarbon_pollution + current_carbon_monoxide_pollution + current_particulate_pollution + current_pollution_location + vehicle_type.

Logical Architecture Reference Flow: wide_area_pollution_data
This data flow is used within the Manage Traffic function as a means of transferring current pollution data from the Manage Emissions facility to the Provide Traffic Surveillance facility. It contains data about the current levels of pollution obtained from the store of pollution data in the area covered by the Traffic Management Center (TMC) and consists of the following data items each of which is defined in its own DDE: pollution_state_area_collection + list_size + list_size{pollution_state_roadside_collection}.

Event Promoters => **Traffic Management**

Physical Architecture Flow Name: event plans

Plans for major events possibly impacting traffic.

Logical Architecture Reference Flow: fep_event_information
This data flow is sent from the event promoters terminator to the Manage Traffic function and contains details of a special event that may become a possible incident due to its impact on the traffic flowing on one or more lanes of a road or highway.

Information Service Provider => **Traffic Management**

Physical Architecture Flow Name: fare and price information

Current transit, parking, and toll fee schedule information.

Logical Architecture Reference Flow: parking_lot_charge_details
This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function and contains the prices being charged by each parking lot for each of its spaces, together with the time and date for which they apply. parking_lot_identity + parking_lot_price + parking_lot_charge_application_time + vehicle_type_for_charges.

Logical Architecture Reference Flow: toll_price_details
This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function and contains the price for each road segment to which a toll applies, with the time and date for when it applies. This data will be used by the Manage Travel Demand facility in its efforts to re-distribute travel demand to the more efficient providers. The data flow consists of the following data items each of which is defined in its own DDE: toll_segments + toll_price + toll_price_application_time + vehicle_type_for_tolls.

Logical Architecture Reference Flow: transit_fare_details
This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function and contains details of the fares being currently charged for transit services. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{transit_route_number + transit_route_segment_list + transit_user_category + transit_route_use_time}.

Physical Architecture Flow Name: logged special vehicle route

Anticipated route information for special vehicles (e.g., oversize vehicles) or groups of vehicles (e.g., governer's motorcade) that may require changes in traffic control strategy.

Logical Architecture Reference Flow: logged_special_vehicle_route
This data contains details about a route that has been requested by a special vehicle. This could be a commercial vehicle that is carrying cargo which could be viewed as being liable to cause a potential incident. Loads falling into this category are those containing hazardous (HAZMAT) material, or those which are outsize, e.g. wide, heavy, or fragile and hence slow moving. This could also include vehicles which must be specially routed (e.g. the governors motorcade). The data flow is derived from the route that has been produced for the special vehicle. hazmat_load_data + list_size + list_size{route_segment_end_point + route_segment_estimated_arrival_time + route_segment_estimated_travel_time + route_segment_identity + route_segment_start_point}.

Logical Architecture Reference Flow: special_vehicle_priority_routing
This data flow is a special form of route similar to an emergency vehicle route, but for use by other vehicle types which may be given special priority routing (e.g. green wave routing). This could be applied to HOV vehicles, special HAZMAT, priority vehicles (e.g. governors motorcade), or even to regular vehicles under a low traffic volume period (e.g. in the early hours of the morning). This flow contains the items shown below each of which is defined in its own DDE: route + vehicle_identity.

Physical Architecture Flow Name: request for traffic information

Request for traffic information that specifies the region/route of interest, the desired effective time period, and other parameters that allow preparation of a tailored response. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Logical Architecture Reference Flow: traffic_data_distribution_request
This data flow contains a request for particular data to be retrieved from the stores of long term, current, and predicted

traffic data. The request is in response to a variety of requests received from ITS Users.

Physical Architecture Flow Name: road network use

Aggregated route usage and associated travel data from clients for planning and analysis.

Logical Architecture Reference Flow: current_other_routes_use

This data flow is used within the Provide Driver and Traveler Services function and contains data about the non-vehicle portion(s) of routes that have been requested by travelers. These route portions will involve the use of modes such as cycling, walking, etc. The data will be stored in ascending route segment number order (i.e. from 1 to the maximum number of route segments),r and consists of the following data items each of which is defined in its own DDE: route_segment_total_number + route_segment_total_number{route_segment_identity + time_period{route_segment_guided_travelers} + route_segment_journey_time}.

Logical Architecture Reference Flow: current_road_network_use

This data flow is sent from the Provide Driver and Traveler Services function to the Manage Traffic function. It contains information about how many vehicles are being guided down each route segment and the average journey time for each route segment provided by guided vehicles. The data will be stored in ascending route segment number order (i.e. from 1 to the maximum number of route segments), and consists of the following data items each of which is defined in its own DDE: route_segment_total_number + route_segment_total_number{route_segment_identity + route_segment_use_prediction + route_segment_journey_time}.

Logical Architecture Reference Flow: current_transit_routes_use

This data flow is used within the Provide Driver and Traveler Services and Manage Traffic functions. It contains data showing the numbers of travelers using all or part of the available transit routes, either for personal guidance or as part of trip requests. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{transit_route_number + transit_route_current_use}.

Map Update Provider => **Traffic Management**

Physical Architecture Flow Name: map updates

Map update which could include a new underlying static or real-time map or map layer(s) update.

Logical Architecture Reference Flow: fmup_demand_display_update

This data flow is sent from the map update provider to the Manage Demand facility within the Manage Traffic function. It contains the digitized map data for displays that can be used as background for the output of data on traffic and travel demand levels.

Logical Architecture Reference Flow: fmup_incident_display_update

This data flow is sent from the map update provider to the Display and Update Incident Data facility within the Manage Traffic function. It contains the digitized map data for displays that can be used as background for the output of data on current or predicted incidents.

Logical Architecture Reference Flow: fmup_traffic_display_update

This data flow is sent from the map update provider to the Display and Output Traffic Data facility within the Manage Traffic function. It contains the digitized map data for displays that can be used as background for the output of data on current or predicted traffic levels.

Media => **Traffic Management**

Physical Architecture Flow Name: external reports

Traffic and incident information that is collected by the media through a variety of mechanisms (e.g., radio station call-in programs, air surveillance).

Logical Architecture Reference Flow: fm_incident_information
This data flow contains data about an incident that has been reported by a member of the traveling public to the media by mechanisms that are outside of ITS, e.g. car phone. The data flow consists of the following items each of which is defined in its own DDE: media_identity + incident_location + incident_start_time + incident_duration + incident_severity + incident_type.

Physical Architecture Flow Name: media information request

Request from the media for current transportation information.

Logical Architecture Reference Flow: fm_incident_data_request
This data flow contains a request for data on incidents to be sent to the Media. The request must specify whether all, current or predicted incidents are required, in the latter case state the time period by date and hour range, and the geographic area(s) to which it should relate.

Logical Architecture Reference Flow: fm_traffic_data_request
This data flow is contains a request from the Media for traffic information. The request must specify the type of information required (flow/congestion) and the geographic area(s) to which it should relate.

Other TM => Traffic Management

Physical Architecture Flow Name: traffic control coordination

Information transfers that enable remote monitoring and control of traffic management devices. This flow is intended to allow cooperative access to, and control of, field equipment during incidents and special events and during day-to-day operations. This flow also allows 24-hour centers to monitor and control assets of other centers during off-hours, allows system redundancies and fail-over capabilities to be established, and otherwise enables integrated traffic control strategies in a region.

Logical Architecture Reference Flow: fotc_identity
This data flow is sent from the other traffic centers to the Manage Traffic function and contains the identity of the remote (originating) Traffic Management Center - TMC for the accompanying data that covers geographic or jurisdictional area(s) outside that served by the local TMC. It consists of the following data item which is defined in its own DDE: tmc_identity.

Logical Architecture Reference Flow: fotc_traffic_control_and_status
This data flow is sent from the other traffic centers to the Manage Traffic function and contains status and control data which is being transferred from one (remote) Traffic Management Center (TMC) to another, in this case the local TMC. It consists of the following data items each of which is defined in its own DDE: other_control_data_for_roads + other_control_data_for_highways + other_status_for_roads + other_status_for_highways.

Physical Architecture Flow Name: traffic information coordination

Traffic information exchanged between TMC's. Normally would include incidents, congestion data, traffic data, signal timing plans, and real-time signal control information.

Logical Architecture Reference Flow: fotc_data_request
This data flow is sent to from other traffic centers to the Manage Traffic function and contains a request for data to be sent to another Traffic Management Center (TMC) from the local TMC. It consists of the following data item which is defined in its own DDE: other_TMC_data_request + local_TMC_incidents_request.

Logical Architecture Reference Flow: fotc_identity
This data flow is sent from the other traffic centers to the Manage Traffic function and contains the identity of the

remote (originating) Traffic Management Center - TMC for the accompanying data that covers geographic or jurisdictional area(s) outside that served by the local TMC. It consists of the following data item which is defined in its own DDE: tmc_identity.

Logical Architecture Reference Flow: fotc_transfer_data
This data flow is sent from the other traffic centers to the Manage Traffic function and contains the data which is being transferred from one (remote) Traffic Management Center (TMC) to another, in this case the local TMC. It consists of the following data items each of which is defined in its own DDE: other_current_data + other_long_term_data + other_predicted_incidents + other_TMC_cv_incidents + permit_coordination + other_TMC_emergency_data.

Parking Management => Traffic Management

Physical Architecture Flow Name: demand management response

Response to various demand management change requests indicating level of compliance with request.

Logical Architecture Reference Flow: parking_lot_charge_change_response
This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function. It contains the response to a previous request for the current parking lot charges to be changed to help produce a change in the current modal split of trips being undertaken by all types of travelers. If sent to one (1) the change was accepted and if set to zero (0), the change was rejected.

Logical Architecture Reference Flow: parking_lot_charge_direct_details
This data flow is contains the prices being charged by each parking lot for each of its spaces, together with the time and date for which they apply. parking_lot_identity + parking_lot_price + parking_lot_charge_application_time + vehicle_type_for_charges.

Physical Architecture Flow Name: parking availability

Parking lot occupancy, availability and payment information.

Logical Architecture Reference Flow: parking_guidance_for_vms
This data flow is used within the Manage Traffic function. It contains the variable message sign (vms) states that will be used to implement the desired traffic control strategy at the parking lots served by the function. This data may be used to guide vehicles towards those parking lots where spaces are currently available or to show which lots have been closed, i.e. are not currently in use. The data flow consists of the following data items each of which is defined in its own DDE: indicator_list + 1{parking_lot_vms_controls}list_size.

Logical Architecture Reference Flow: parking_lot_current_state
This data flow is used within the Manage Traffic function and contains the identity of the parking lot plus its current status and occupancy. It consists of the following data items each of which is defined in its own DDE: parking_lot_identity + parking_lot_state + parking_lot_current_occupancy.

Planning Subsystem => Traffic Management

Physical Architecture Flow Name: planning data

Data and commands from Transportation Planners.

Logical Architecture Reference Flow: link_data
This data flow is sent by the Plan System Deployment function to the Manage Traffic function. It contains a new version of the store of link data used to determine which other TMC to contact to obtain traffic data relating to another

geographic area. The data flow consists of the following data items each of which is defined in its own DDE: link_attributes + link_identity + link_TMC_identity.

Logical Architecture Reference Flow: static_data_request

This data flow is sent from the Provide System Deployment function to the Manage Traffic function to request a copy of some or all of the current static data being used by the traffic control, sensor processing and incident management facilities. The data flow is sized at three bytes to enable the use of three alphanumeric characters, one for each type of data, i.e. 'i' for incident, 's' for sensor and 't' for traffic. The receiving process will check that the alphanumeric character for its data is present before retrieving and returning it to the Plan System Deployment function.

Logical Architecture Reference Flow: supply_incident_static_data

This data flow is sent by the Provide System Deployment function to the Manage Traffic function and includes new and/or amended static data for use in incident management. This data consists of details of the road network plus the location and relationship between links in the network. It therefore contains the contents of the following data store which is defined in its own DDE: static_data_for_incident_management.

Logical Architecture Reference Flow: supply_traffic_static_data

This data flow is sent by the Provide System Deployment function to the Manage Traffic function and includes new and/or amended static data for use in traffic control and sensor processing. This data consists of traffic signal data (timings, permitted phase changes, etc.), which items of traffic data are used by certain processes, VMS data, etc. It therefore contains the contents of the following data stores, each of which is defined in its own DDE: static_data_for_traffic_control + static_data_for_sensor_processing.

Logical Architecture Reference Flow: traffic_data_deployment_request

This data flow is used by the Plan System Deployment and Improvement function to request the Manage Traffic function to provide it with traffic data. It must contain a processor source identity so that the Manage Traffic function knows where to send the retrieved traffic data.

Rail Operations => Traffic Management

Physical Architecture Flow Name: railroad advisories

Real-time notification of railway-related incident or advisory.

Logical Architecture Reference Flow: fro_incident_notification

This data flow is used by a rail operator to notify an ITS traffic management function that a rail incident has been detected that will impact vehicle traffic. This could be an HRI collision incident or merely a stalled train that is blocking an HRI. It could also be a rail incident NOT associated with an HRI, but that may cause abnormal traffic patterns, or blockage of a non-crossing or grade separated roadway. The sizing assumption is based on the assumption that this is a free form text message.

Physical Architecture Flow Name: railroad schedules

Train schedules, maintenance schedules, and other information from the railroad that supports forecast of HRI closures.

Logical Architecture Reference Flow: fro_maintenance_schedules

This data flow provides the information traffic management needs to plan around scheduled maintenance by railroad crews at highway grade crossings that may affect highway traffic. The sizing assumption assumes that this is a planned closing notice (crossing, time, and duration) with an appended text message, and covers some average number of crossings needing maintenance per day (HRI_MAINT_PER_DAY).

Logical Architecture Reference Flow: fro_train_schedules

This data flow is used by railroads to provide ITS traffic management functions of train movement schedules that may be pertinent to traffic and route planning, highway maintenance planning, etc. As scheduled information, it may be used to determine the probability of grade crossing blockage by trains and therefore the expected traffic flow rates on specific vehicle routes. The sizing assumption is based on a daily schedule consisting of an average number of grade crossing events per day. Each event would typically associated with a specific crossing, train identification, a scheduled arrival time and an estimated closure time.

Roadway Subsystem => Traffic Management

Physical Architecture Flow Name: AHS status

Status of AHS equipment, lane controls etc.

Logical Architecture Reference Flow: ahs_checking_details
This data flow is used in the Provide Vehicle Monitoring and Control function and contains updates to the counts of successful and failed check-ins to the automatic highway system (ahs) lanes from roadside locations. It consists of the following data items each of which is defined in its own DDE: ahs_failed_checks_count + ahs_successful_checks_count.

Physical Architecture Flow Name: emissions data

Emissions data and associated imagery collected by roadside equipment.

Logical Architecture Reference Flow: vehicle_pollution_alert
This data flow is used by the Manage Traffic function as a means of transferring current vehicle pollution data from the Manage Emissions facility to the Manage Emergency Services function to enable enforcement of air quality standards. It contains data about the current levels of pollution being output by a vehicle. This data is held in the following data items each of which is defined in its own DDE: vehicle_identity + vehicle_license + pollution_data_violation.

Logical Architecture Reference Flow: vehicle_pollution_message_for_highways
This data flow is used within the Manage Traffic function and contains data about the current levels of pollution being output by a vehicle. It is for output to the vehicle driver who is on a freeway in the geographic and/or jurisdictional area(s) served by the function. The data flow consists of the following data items each of which is defined in its own DDE: indicator_identity + pollution_output_message.

Logical Architecture Reference Flow: vehicle_pollution_message_for_roads
This data flow is used within the Manage Traffic function and contains data about the current levels of pollution being output by a vehicle. It is for output to the vehicle driver who is on a road (surface street) in the geographic and/or jurisdictional area(s) served by the function. The data flow consists of the following data items each of which is defined in its own DDE: indicator_identity + pollution_output_message.

Physical Architecture Flow Name: environmental conditions

Current environment conditions (e.g., air temperature, wind speed, surface temperature) as measured by environmental sensors and communicated by supporting field equipment.

Logical Architecture Reference Flow: environment_sensor_data
This data flow is used within the Manage Traffic function and contains a set of outputs from individual environment sensors. It consists of the following data items each of which is defined in its own DDE: 1{station_id + sensor_identity + environment_sensor_output}list_size.

Logical Architecture Reference Flow: environmental_sensor_status
This data flow is used within the Manage Traffic function to report the status of an environmental sensor. By monitoring this data flow, the receiving process can monitor the health and current status of field equipment. It consists

of the following items each of which are defined in its own DDE: list_size+ 1{station_id+ sensor_identity}list_size.

Physical Architecture Flow Name: fault reports

Reports from field equipment (sensors, signals, signs, controllers, etc.) which indicate current operational status.

Logical Architecture Reference Flow: environment_sensor_fault_data
This data flow is used within the Manage Traffic function to show that an environment sensor has developed a fault that means it is not operating correctly. The fault will have been found by a process that is local to the sensor itself.

Logical Architecture Reference Flow: sensor_fault_data
This data flow is used within the Manage Traffic function to show that a sensor has developed a fault that means it is not operating correctly. The fault will have been found by a process that is local to the sensor itself.

Logical Architecture Reference Flow: traffic_control_device_status
This data flow is used within the Manage Traffic function to show any faults that have been found in roadside equipment. This may be in an indicator or in a traffic sensor. The data flow consists of the following data items each of which is defined in its own DDE: indicator_fault_data + vehicle_smart_probe_data_output_fault + traffic_sensor_status.

Logical Architecture Reference Flow: traffic_sensor_status
This data flow is used within the Manage Traffic function to report the status of a sensor. By monitoring this data flow, the receiving process can monitor the health and current status of field equipment.

Physical Architecture Flow Name: freeway control status

Current operational status and operating parameters for ramp meters, dynamic message signs, mainline metering/lane controls and other control equipment associated with freeway operations.

Logical Architecture Reference Flow: indicator_input_data_from_highways
This data flow is used within the Manage Traffic function and contains the actual state of operation of the roadside indicators used to pass instructions to drivers and travelers on freeways within the geographic and/or jurisdictional area(s) served by the function. It is used for centralized monitoring the operation of the indicators and consists of the following data items each of which is defined in its own DDE: list_size + list_size{indicator_identity + indicator_response_state}.

Physical Architecture Flow Name: hov data

Current HOV lane information including both standard traffic flow measures and information regarding vehicle occupancy in HOV lanes.

Logical Architecture Reference Flow: hov_lane_data_input
This data is used within the Manage Traffic function and contains data from which the use of High Occupancy Vehicle (HOV) lanes can be monitored. It consists of the following data items each of which is defined in its own DDE: private_vehicle_occupants + traffic_video_image + vehicle_count.

Logical Architecture Reference Flow: hov_sensor_data
This data flow is used within the Manage Traffic function and contains the HOV data obtained from processing the inputs from sensors around the road network. It consists of the following data items each of which is defined in its own DDE: 1{private_vehicle_occupants+ hov_priority}link_list.

Physical Architecture Flow Name: hri status

Status of the highway-rail intersection equipment including both the current state or mode of operation and the current equipment condition.

Logical Architecture Reference Flow: hri_guidance_for_beacon_message
This data flow is used to control which message is to be broadcast to drivers approaching an HRI. The sizing assumption is based on a coded implementation that uses a 3 character ASCII code to select one of many prerecorded messages.

Logical Architecture Reference Flow: hri_guidance_for_vms
This data flow is used to control which message is to be displayed on a variable message sign (vms) as drivers approach an HRI. The sizing assumption is based on a coded implementation that uses a 3 character ASCII code to select on of several predefined messages.

Logical Architecture Reference Flow: hri_status
This data flow represents the complete status of an HRI, including train situation, vehicle traffic, equipment health and predictable near term events .It consists of the following data items each of which is defined in its own DDE: hri_state + hri_closure_data_response.

Logical Architecture Reference Flow: hri_traffic_data
This data flow contains data to be used by traffic management to coordinate its overall operations with the hri activity.

Logical Architecture Reference Flow: rail_operations_message
This data flow contains advanced (predictive) data about an HRI operational status to be passed to rail operations. It is generated by the Manage HRI Traffic process for use by the Interact with Rail Operations process.

Logical Architecture Reference Flow: traffic_management_request
This data flow is used by hri to request services or data from other traffic management functions .

Physical Architecture Flow Name: incident data

Data and imagery from the roadside supporting incident detection and verification.

Logical Architecture Reference Flow: incident_analysis_data
This data flow is used within the Manage Traffic function and contains processed traffic sensor data that can be analyzed for the possible presence of incidents. The data is provided directly from the local traffic sensor process rather than from some regional/area based process and so must originate in sensors that are within a small geographic area.

Logical Architecture Reference Flow: incident_video_image
This data flow is used within the Manage Traffic function and contains a high resolution digitized image of a potential or current incident at a particular point on the road or freeway network. The size expression below assumes that the image is compressed.

Physical Architecture Flow Name: intersection blockage notification

Notification that a highway-rail intersection is obstructed and supporting information.

Logical Architecture Reference Flow: hri_blockage
This data flow contains information, obtained from sensors in the intersection, regarding blockage of the hri by a vehicle or other object. This data will be passed to Rail Operations.

Logical Architecture Reference Flow: intersection_blocked
This data flow contains information, obtained from sensors in the intersection, regarding blockage of the hri by a vehicle or other object. This data will be used by the traffic management functions to begin incident management procedures.

Physical Architecture Flow Name: request for right-of-way

Forwarded request from signal prioritization, signal preemption, pedestrian call, multi-modal crossing activation, or other source for right-of-way.

Logical Architecture Reference Flow: indicator_input_data_from_roads
This data flow is used within the Manage Traffic function and contains the actual state of operation of the roadside and grade crossing indicators used to pass instructions to drivers and travelers on roads (surface streets) within the geographic and/or jurisdictional area(s) served by the function. It is used for centralized monitoring the operation of the indicators and consists of the following data items each of which is defined in its own DDE: list_size + list_size{indicator_identity + indicator_response_state}.

Logical Architecture Reference Flow: multimodal_crossing_sensor_data
This data flow is used within the Manage Traffic function and contains the multimodal crossing data obtained from processing the other inputs from sensors around the road network. It consists of the following data items each of which is defined in its own DDE: 1{crossing_close_time+ crossing_close_duration}crossing_list.

Logical Architecture Reference Flow: pedestrian_sensor_data
This data flow is used within the Manage Traffic function and contains the pedestrian data obtained from processing the other inputs from sensors around the road network. It consists of the following data items each of which is defined in its own DDE: 1{pedestrian_demand}node_list.

Physical Architecture Flow Name: reversible lane status

Current reversible lane status including traffic sensor and surveillance data and the operational status and mode of the reversible lane control equipment.

Logical Architecture Reference Flow: reversible_lane_video_images
This data flow is used within the Manage Traffic function. It contains video images of the reversible lanes. It consists of the following data item which is defined in its own DDE: incident_video_image.

Logical Architecture Reference Flow: sensor_data_for_reversible_lanes
This data flow is used within the Manage Traffic function and contains data from which a wrong way vehicle is detected in a reversible lane through the use of sensors located. It consists of the following data items each of which is defined in its own DDE: traffic_video_image + vehicle_detection_data.

Physical Architecture Flow Name: roadway information system status

Current operating status of dynamic message signs, highway advisory radios, beacon systems, or other configurable field equipment that provides dynamic information to the driver.

Logical Architecture Reference Flow: dms_status_for_highways
This data flow is used within the Manage Traffic function and contains data about the text strings of information to be output to drivers on freeways in the geographic and/or jurisdictional area(s) served by the function. The data flow consists of the following data items each of which is defined in its own DDE: vms_updates_for_highways.

Logical Architecture Reference Flow: dms_status_for_roads
This data flow contains the Dynamic Message Sign status for sign control data, operating at the roadside on roads (surface streets) in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{indicator_identity + vms_advisory_text}.

Logical Architecture Reference Flow: har_status_for_highways
This data flow contains the Highway Advisory Radio status for HARs, operating at the roadside on highways in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: har_status+ har_identity.

Logical Architecture Reference Flow: har_status_for_roads
This data flow contains the Highway Advisory Radio status for HARs, operating at the roadside on roads (surface

streets) in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: har_status+ har_identity.

Logical Architecture Reference Flow: information_device_fault_status
This data flow is used to show any faults that have been found in roadside information dissemination equipment. This includes highway advisory radio, dynamic message signs, or in-vehicle signs. The data flow consists of the following data items each of which is defined in its own DDE: har_fault_data_for_roads+ har_fault_data_for_highways+ vehicle_sign_data_output_fault.

Physical Architecture Flow Name: signal control status

Status of surface street signal controls.

Logical Architecture Reference Flow: indicator_input_data_from_roads
This data flow is used within the Manage Traffic function and contains the actual state of operation of the roadside and grade crossing indicators used to pass instructions to drivers and travelers on roads (surface streets) within the geographic and/or jurisdictional area(s) served by the function. It is used for centralized monitoring the operation of the indicators and consists of the following data items each of which is defined in its own DDE: list_size + list_size{indicator_identity + indicator_response_state}.

Physical Architecture Flow Name: traffic flow

Raw and/or processed traffic detector information which allows derivation of traffic flow, speed, and density measures.

Logical Architecture Reference Flow: traffic_image_data
This data flow is used within the Manage Traffic function and contains the data produced by processing image data obtained from visual detection systems. This data is therefore that which can be obtained from systems such as traffic surveillance closed circuit television (cctv). It is analyzed and used to detect traffic conditions such as flow, occupancy, possible incidents, etc. The size expression below assumes that the image is compressed.

Logical Architecture Reference Flow: traffic_sensor_data
This data flow is used within the Manage Traffic function and contains the data obtained from processing the inputs from sensors around the road network. It consists of the following data items each of which is defined in its own DDE: list_size + 1{station_id + sensor_identity + traffic_sensor_output}list_size.

Physical Architecture Flow Name: traffic images

High fidelity, real-time traffic images suitable for surveillance monitoring by the operator or for use in machine vision applications.

Logical Architecture Reference Flow: traffic_video_image
This data flow is used within the Manage Traffic function and contains a video image of sufficient fidelity to support operator monitoring applications. This image can be a by-product of a machine vision application or the end-product of a system dedicated to traffic surveillance.

Logical Architecture Reference Flow: traffic_video_image_for_display
This data flow contains the video image which is used by a roadside device to measure traffic flow measures. The data flow consists of the following item which is defined in its own DDE: traffic_video_image.

Physical Architecture Flow Name: vehicle probe data

Vehicle probe data indicating identity, route segment identity, link time and location.

Logical Architecture Reference Flow: vehicle_smart_probe_data_for_storage
This data flow is used within the Manage Traffic function. It contains the processed vehicle smart probe data collected

from a roadside unit, which in turn have received data output by suitably equipped vehicles as they pass by. The data flow consists of the following data items each of which is defined in its own DDE: vehicle_smart_probe_data_source + vehicle_smart_probe_data_indication.

Logical Architecture Reference Flow: vehicle_tag_data

This data flow is used within the Manage Traffic function. It contains the data from parking lot and toll tags on-board vehicles plus the identity of the unit which received the data. The data flow consists of the following data items each of which is defined in its own DDE: vehicle_tag_data_source_identity + vehicle_tag_input_data.

Toll Administration => Traffic Management

Physical Architecture Flow Name: demand management response

Response to various demand management change requests indicating level of compliance with request.

Logical Architecture Reference Flow: toll_price_changes_response

This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function. It contains the response to a previous request for the current toll prices to be changed to help produce a change in the current modal split of trips being undertaken by all types of travelers. If sent to one (1) the change was accepted and if set to zero (0), the change was rejected.

Logical Architecture Reference Flow: toll_price_direct_details

This data flow contains the price for each road segment to which a toll applies, with the time and date for when it applies. This data will be used by the Manage Travel Demand facility in its efforts to re-distribute travel demand to the more efficient providers. The data flow consists of the following data items each of which is defined in its own DDE: toll_segments + toll_price + toll_price_application_time + vehicle_type_for_tolls.

Physical Architecture Flow Name: probe data

Aggregate data from probe vehicles including location, speed for a given link or collection of links.

Logical Architecture Reference Flow: probe_data_for_traffic

This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function and contains journey times between toll collection points for those vehicles equipped for electronic toll collection. It is used to calculate link journey times for use in adaptive traffic control techniques and route selection and guidance. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{route_segment_identity + route_segment_journey_time_from_tolls}.

Traffic Management => Construction and Maintenance

Physical Architecture Flow Name: closure coordination

Coordination between subsystems regarding construction and maintenance closure times and durations.

Logical Architecture Reference Flow: tcm_incident_confirmation

This data flow is sent to the construction and maintenance terminator from the Manage Traffic function to provide confirmation that work by the Construction and Maintenance terminator which has been recorded as a possible incident can take place at the requested time.

Logical Architecture Reference Flow: tcm_request_incident_change

This data flow is sent to the construction and maintenance terminator from the Manage Traffic function to request changes to the timing of work requested by the construction and maintenance terminator. This will have been provided as input to the function and been recorded as a possible incident.

Physical Architecture Flow Name: maintenance resource request

Request for road maintenance resources that can be used in the diversion of traffic (cones, portable signs), clearance of an incident, and repair of ancillary damage.

Logical Architecture Reference Flow: tcm_resource_request

This data flow is used to request traffic management resources to include temporary signs, cones, and other assets that can be used to divert traffic, create detours, and otherwise mange traffic at the incident scene. It also includes requests for any other assets that may be needed to support incident clearance.

Physical Architecture Flow Name: traffic equipment status

Identification of field equipment requiring repair and known information about the associated faults.

Logical Architecture Reference Flow: tcm_fault_data

This data flow is sent to the construction and maintenance terminator from the Manage Traffic function. It contains a report showing that a particular fault has been found in an indicator be either a local or a roadside process. This report acts as a request for the construction and maintenance terminator to effect repairs to restore the indicator to normal operation as soon as possible.

Logical Architecture Reference Flow: tcm_sensor_fault_data

This data flow is sent to the construction and maintenance terminator from the Manage Traffic function. It contains a report showing that a particular fault has been found in a sensor be either a local or a roadside process. This report acts as a request for the construction and maintenance terminator to effect repairs to restore the sensor to normal operation as soon as possible.

Traffic Management => **DMV**

Physical Architecture Flow Name: license request

Request supporting registration data based on license plate read during violation.

Logical Architecture Reference Flow: tdmv_traffic_violation_identity_code

This data flow is sent to the department of motor vehicles from the Manage Emergency Services function and contains the identity code of the ITS that is requesting the vehicle registration data so that a high occupancy vehicle (hov) lane or pollution violation can be processed.

Logical Architecture Reference Flow: tdmv_traffic_violation_vehicle_license

This data flow is sent to the department of motor vehicles from the Manage Emergency Services function and contains the vehicle license for which the corresponding registration data is required so that a high occupancy vehicle (hov) lane or pollution violation can be processed.

Traffic Management => **Emergency Management**

Physical Architecture Flow Name: current network conditions

Current traffic information, road conditions, and camera images that can be used to locate and verify reported incidents, and plan and implement an appropriate response.

Logical Architecture Reference Flow: incident_video_for_emergency_services

This data flow is used within the Manage Traffic function. It contains current video images of incidents requested by the Manage Emergency Services facility. It consists of the following data item which is defined in its own DDE: incident_video_image.

Logical Architecture Reference Flow: traffic_data_for_emergency_services
This data flow is used within the Manage Traffic function and contains current traffic information and roadway environmental conditions for the emergency management system. It consists of the following items each of which is defined in it own DDE: roadway_environment_conditions + link_state_data.

Physical Architecture Flow Name: emergency traffic control response

Status of the green wave or other special traffic signal control strategy implemented in response to the emergency traffic control request.

Logical Architecture Reference Flow: emergency_traffic_control_response
This data flow is sent from the Manage Traffic function to the Manage Emergency Services function. It contains a list of the route segments that have been provided for use by an emergency vehicle and indicates the traffic management strategy that is in effect for these route segments. date + list_size + list_size{route_segment_identity + route_segment_estimated_arrival_time} + time + selected_emergency_vehicle_strategy.

Physical Architecture Flow Name: incident information request

Request for incident information, clearing time, severity. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Logical Architecture Reference Flow: incident_details_request
This data flow is used by the Manage Traffic function to request details of incidents from the Manage Emergency Services function. It contains the following data items each of which is defined in its own DDE: incident_type + incident_oldest_time.

Physical Architecture Flow Name: incident notification

The notification of an incident including its nature, severity, and location.

Logical Architecture Reference Flow: incident_alert
This data flow is used to send details of an incident from the Manage Traffic function to the Manage Emergency Services function. It contains the following data items each of which is defined in its own DDE: incident_location + incident_start_time + incident_duration + incident_severity + incident_type + incident_traffic_impact.

Logical Architecture Reference Flow: incident_response_clear
This data flow is sent from the Manage Traffic function to the Manage Emergency Services function and is an indication that the Manage Traffic function has data indicating that an incident has been cleared. It consists of the following items of data each of which is defined in its own DDE: incident_location + incident_type.

Logical Architecture Reference Flow: wrong_way_vehicle_detection
This data flow is sent by the Manage Traffic function to the Manage Emergency Services function and contains data about wrong-way vehicles detected in reversible lanes. It consists of the following data items each of which is defined in its own DDE: vehicle_identity + vehicle_license + incident_video_image + traffic_video_image + vehicle_detection_data.

Physical Architecture Flow Name: resource deployment status

Status of traffic management center resource deployment identifying the resources available and their current deployment status.

Logical Architecture Reference Flow: resource_deployment_status
This data flow is sent to the Manage Traffic function indicating the availability of the requested traffic management resources and provides current status of their deployment.

Traffic Management => **Emissions Management**

Physical Architecture Flow Name: pollution state data request

Aggregated emissions data information request.

Logical Architecture Reference Flow: pollution_state_data_request
This data flow is used within the Manage Traffic function as a means of requesting current pollution data to be sent from the Manage Emissions facility to the Manage Demand facility. It contains request for data about current levels of pollution. This data can be requested for a roadside or wide area location.

Traffic Management => Enforcement Agency

Physical Architecture Flow Name: violation notification

Notification to enforcement agency of violation or regulations.

Logical Architecture Reference Flow: tea_traffic_violation_data
This data flow is sent from the Manage Emergency Services function to the enforcement agency and contains information about high occupancy vehicle (hov) lane use and pollution violations that have been detected by processes within the Manage Traffic function. The data in this flow will enable the notified enforcement agency to take the appropriate action against those committing the violations.

Traffic Management => Event Promoters

Physical Architecture Flow Name: event confirmation

Confirmation that special event details have been received and processed.

Logical Architecture Reference Flow: tep_event_confirmation
This data flow is sent from the Manage Traffic function to event promoters and is the confirmation that the previously submitted details of an event have been accepted as a possible incident.

Traffic Management => Information Service Provider

Physical Architecture Flow Name: request fare and price information

Requests for current fare and price information from a service provider that can be used to augment the traffic manager's overall view of current transportation network status.

Logical Architecture Reference Flow: parking_lot_charge_request
This data flow is sent from the Manage Traffic function to the Provide Electronic Payment Services function and contains a request for the current prices being charged for parking lot spaces.

Logical Architecture Reference Flow: toll_price_request
This data flow is sent from the Manage Traffic function to the Provide Electronic Payment Services function and contains a request for the current prices being charged for toll segments on the road and highway network.

Logical Architecture Reference Flow: transit_fare_request
This data flow is sent from the Manage Traffic function to the Provide Electronic Payment Services function and contains a request for the current prices being charged for transit fares.

Physical Architecture Flow Name: traffic information

Current and predicted traffic information, road and weather conditions, incident information, and pricing data. Either raw data, processed data, or some combination of both may be provided by this architecture flow.

Logical Architecture Reference Flow: current_highway_network_state
This data flow is sent by the Manage Traffic function to the Provide Driver and Traveler Services function and contains data about traffic conditions on links in the road network served by the function. The data is used by the route selection and guidance processes in determining the best vehicle routes. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{link_identity + link_journey_time + link_delay}.

Logical Architecture Reference Flow: current_road_network_state
This data flow is sent by the Manage Traffic function to the Provide Driver and Traveler Services function and contains data about traffic conditions on links in the highway network served by the function. The data is used by the route selection and guidance processes in determining the best vehicle routes. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{link_identity + link_journey_time + link_delay}.

Logical Architecture Reference Flow: link_data_for_guidance
This data flow is sent from the Manage Traffic function to the Provide Driver and Traveler Services function. It contains data for use in determining which other ISP('s) must be contacted to obtain data about roads and highways in geographic area(s) outside that served by the local function. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{link_attributes + link_identity + link_ISP_identity}.

Logical Architecture Reference Flow: predicted_incidents
This data flow is used within the Manage Traffic function and contains details of known incidents due to take place in the future. It contains the following data items each of which is defined in its own DDE: list_size + list_size{incident_location + incident_type + incident_severity + incident_description + incident_traffic_impact}.

Logical Architecture Reference Flow: prediction_data
This data flow is used within the Manage Traffic function and is also sent by that function to the Manage Transit and Provide Driver and Traveler Services function. It contains output from the predictive model process showing predictions of traffic data for route segments on the road and highway network served by the Manage Traffic function. The data flow consists of the following items each of which is defined in its own DDE: list_size + list_size{route_segment_identity + route_segment_volume_delay_predictions + route_segment_queue_delay_predictions + route_segment_speed_predictions + route_segment_occupancy_predictions}.

Logical Architecture Reference Flow: sensor_data_for_distribution
This data flow contains raw and processed sensor data. The data flow consists of the following data items each of which is defined in its own DDE: sensor_output_data + roadway_environment_conditions.

Logical Architecture Reference Flow: traffic_data_for_distribution
This data flow is used within the Manage Traffic function. It contains the response to a request for particular data to be retrieved from the stores of current, long term and predictive model data. This data will be used as the basis for traffic information data that is provided to other ITS functions. The data flow consists of the following data items each of which is defined in its own DDE: current_data_for_retrieval + long_term_data_for_retrieval + predictive_model_data_for_retrieval.

Traffic Management => Map Update Provider

Physical Architecture Flow Name: map update request

Request for a map update which could include a new underlying map or map layer updates.

Logical Architecture Reference Flow: tmup_request_demand_display_update
This data flow is sent to the map update provider from the Manage Demand facility within the Manage Traffic function. It contains a request for an update to the digitized map data for displays that can be used as background for

the output of data on traffic and travel demand levels.

Logical Architecture Reference Flow: tmup_request_incident_display_update
This data flow is sent to the map update provider from the Display and Update Incident Data facility within the Manage Traffic function. It contains a request for an update to the digitized map data for displays that can be used as background for the output of data on current and predicted incidents.

Logical Architecture Reference Flow: tmup_request_traffic_display_update
This data flow is sent to the map update provider from the Display and Output Traffic Data facility within the Manage Traffic function. It contains a request for an update to the digitized map data for displays that can be used as background for the output of data on current and predicted traffic levels.

Traffic Management => Media

Physical Architecture Flow Name: traffic information for media

Report of current traffic conditions, incidents, maintenance activities and other traffic-related information prepared for public dissemination through the media.

Logical Architecture Reference Flow: tm_incident_data
This data flow contains data on current and/or predicted incidents in a form which will be readily understood by the Media. The data is sent in response to a request for information from the media.

Logical Architecture Reference Flow: tm_traffic_data
This data flow gives information on a particular current traffic situation in a form which will be readily understood by Media Systems .

Traffic Management => Other TM

Physical Architecture Flow Name: traffic control coordination

Information transfers that enable remote monitoring and control of traffic management devices. This flow is intended to allow cooperative access to, and control of, field equipment during incidents and special events and during day-to-day operations. This flow also allows 24-hour centers to monitor and control assets of other centers during off-hours, allows system redundancies and fail-over capabilities to be established, and otherwise enables integrated traffic control strategies in a region.

Logical Architecture Reference Flow: totc_identity
This data flow is sent to the other traffic centers by the Manage Traffic function and contains the identity of the local Traffic Management Center - TMC which is sending the accompanying data about traffic conditions in the geographic or jurisdictional area which is serves. The data flow consists of the following data item which is defined in its own DDE: tmc_identity.

Logical Architecture Reference Flow: totc_traffic_control_and_status
This data flow is sent to the other traffic centers by the Manage Traffic function and contains traffic control and status data which is being transferred from one Traffic Management Center (TMC) to one or more others. It consists of the following data items each of which is defined in its own DDE: control_data_for_roads + status_data_for_roads + control_data_for_highways + status_data_for_highways.

Physical Architecture Flow Name: traffic information coordination

Traffic information exchanged between TMC's. Normally would include incidents, congestion data, traffic data, signal timing plans, and real-time signal control information.

Logical Architecture Reference Flow: totc_data_request

This data flow is sent to the other traffic centers by the Manage Traffic function and contains a request for data to be sent from another Traffic Management Center (TMC) to the local TMC. It consists of the following data item which is defined in its own DDE: other_TMC_data_request + other_TMC_incidents_request.

Logical Architecture Reference Flow: totc_identity

This data flow is sent to the other traffic centers by the Manage Traffic function and contains the identity of the local Traffic Management Center - TMC which is sending the accompanying data about traffic conditions in the geographic or jurisdictional area which is serves. The data flow consists of the following data item which is defined in its own DDE: tmc_identity.

Logical Architecture Reference Flow: totc_transfer_data

This data flow is sent to the other traffic centers by the Manage Traffic function and contains the data which is being transferred from one Traffic Management Center (TMC) to one or more others. It consists of the following data items each of which is defined in its own DDE: current_data + cv_incidents_for_other_TMC + emergency_data_for_other_TMC + long_term_data + permit_coordination + predicted_incidents_local_data.

Traffic Management => Parking Management

Physical Architecture Flow Name: demand management request

Request to change the demand for road facility use through pricing or other mechanisms.

Logical Architecture Reference Flow: parking_lot_charge_change_request

This data flow is sent from the Manage Traffic function to the Provide Electronic Payment Services function and contains a request for a change to the current parking lot charging structure that will help to influence a change in modal split of journeys currently being undertaken by travelers of all types, i.e. including drivers and transit users, by encouraging then to use certain parking lots, e.g. those near park and ride sites on the edge of an urban area. It consists of the following data items each of which is defined in its own DDE: parking_lot_identity + parking_lot_price + parking_lot_charge_application_time + vehicle_type_for_charges.

Logical Architecture Reference Flow: parking_lot_charge_direct_request

This data flow is sent from the Manage Traffic function to the Provide Electronic Payment Services function and contains a request for the current prices being charged for parking lot spaces.

Physical Architecture Flow Name: parking instructions

Instructions from traffic manager or parking operator regarding operation strategy of a parking facility.

Logical Architecture Reference Flow: parking_lot_input_data

This data flow is used within the Manage Traffic function and contains data that is used to calculate the occupancy of parking lots. It consists of the following items each of which is defined in its own DDE: parking_lot_list + 1{vehicle_count + vehicle_queue_length}list_size.

Logical Architecture Reference Flow: selected_parking_lot_control_strategy

This data flow is used within the Manage Traffic function and contains the strategy which has been selected for implementation at parking lots to control their use. The strategy will be designed to promote or discourage the use of a parking lot by directing vehicles to or away from it through the use of variable message signs (vms). The decision on which strategy to employ will depend upon such things as the overall traffic management strategy, the need to restrict vehicle use because of a number of factors e.g. congestion, pollution, and the desire to encourage travelers to make use of alternative modes of transport by using park and ride (P+R) facilities. The strategy may be applied to some or all of the parking lots in the geographic area served by the TMC. The data flow consists of the following data items each of which is defined in its own DDE: the may be one of 'open' or 'close' the lot and may be applied to some or all of the lots in the geographic area served by the function. parking_lot_list + selected_parking_lot_strategy_type.

Logical Architecture Reference Flow: static_data_for_parking_lots
This data flow is used within the Manage Traffic function and is provided by the Plan System Deployment function. It contains data that relates vehicle sensors, queue counting sensors and signs to individual parking lots, and the lot occupancy(ies) at which states such as 'almost full' and 'full' will apply. The data is sent to each parking lot for its own use. The data flow consists of the following data items each of which is defined in its own DDE: parking_lot_identity + parking_lot_sensor_allocation + parking_lot_state_thresholds + parking_lot_vms_allocation.

Traffic Management => Planning Subsystem

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: ahs_operational_data
This data flow is sent from the Provide Vehicle Monitoring and Control function to the Plan System Deployment function. It contains details of the number of vehicles that have been checked into the automatic highway system (ahs), plus details about the use of ahs lanes during the previous time period, e.g. one (1) hour. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{ahs_checking_records + ahs_lane_use_data + date + time}.

Logical Architecture Reference Flow: current_incident_static_data
This data flow is sent from the Manage Traffic function to the Plan System Deployment function and contains of some or all of the current static data used for the management of incidents. It therefore contains the contents of the following data store, which is defined in its own DDE: static_data_for_incident_management.

Logical Architecture Reference Flow: current_traffic_static_data
This data flow is sent from the Manage Traffic function to the Plan System Deployment function and contains of some or all of the current static data consisting of signal data (timings, permitted phase changes, etc.), VMS data, which items of traffic data are used by certain processes, etc. It therefore contains the contents of the following data stores, each of which is defined in its own DDE: static_data_for_traffic_control + static_data_for_sensor_processing.

Logical Architecture Reference Flow: traffic_data_for_deployment
This data flow is sent from the Manage Traffic function to the Plan System Deployment and Implementation function. It is used to provide data on the traffic flowing in the road network, plus that which is predicted to flow in the network and consists the following items each of which is defined in its own DDE: long_term_data + predictive_model_data.

Traffic Management => Rail Operations

Physical Architecture Flow Name: hri advisories

Notification of Highway-Rail Intersection equipment failure, intersection blockage, or other condition requiring attention, and maintenance actives at or near highway rail intersections.

Logical Architecture Reference Flow: tro_equipment_status
This data flow contains information about the status of the wayside equipment and the intelligent intersection controller. It is used to pass information to rail operations about the overall health and status of the HRI.

Logical Architecture Reference Flow: tro_event_schedules
This data flow contains highway event schedules for use by a rail operator. Typically the rail operator would be interested in highway maintenance at or near grade crossings that may interfere with the rail right-of-way. The sizing assumption is based on an assumption of a free form text message.

Logical Architecture Reference Flow: tro_incident_notification

This data flow contains a highway incident notification relevant to a rail operator. Typically the rail operator would be interested in highway incidents at or near railroads that may interfere with the safe operation of passing trains (e.g. a HAZMAT spill, equipment failure, or an intersection blockage). The sizing assumption is based on an assumption of a free form text message.

Traffic Management => Roadway Subsystem

Physical Architecture Flow Name: AHS control information

Control data to AHS roadway equipment

Logical Architecture Reference Flow: ahs_control_data_changes

This data flow is used within the Provide Vehicle Monitoring and Control function to send data from the management facility to the automatic highway system (ahs) check-in facilities. It contains data defining parameters to be used by vehicles participating in platoon following and running in ahs controlled lanes, which will override any already loaded into the vehicles. The data flow consists of the following items of data each of which is defined in its own DDE: ahs_demand_accel_decel_profile + ahs_demand_headway.

Physical Architecture Flow Name: freeway control data

Control commands and operating parameters for ramp meters, dynamic message signs, mainline metering/lane controls and other systems associated with freeway operations.

Logical Architecture Reference Flow: indicator_control_data_for_highways

This data flow is used within the Manage Traffic function and contains the actual data from which instructions to the driver and traveler can be produced by indicators at the roadside on freeways in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: indicator_crossing_control_data_for_highways + indicator_ramp_control_data.

Logical Architecture Reference Flow: indicator_control_monitoring_data_for_highways

This data flow is used within the Manage Traffic function. It contains the actual data from which instructions to the driver and traveler can be produced by indicators on the freeways in the geographic and/or jurisdictional area(s) served by the function. In this case the data is used by a process to monitor the operation of the indicators rather than actual message output. The data flow consists of the following data items each of which is defined in its own DDE: indicator_control_data_for_highways.

Physical Architecture Flow Name: hri control data

Data required for HRI information transmitted at railroad grade crossings and within railroad operations.

Logical Architecture Reference Flow: hri_traffic_surveillance

This data flow represents the various traffic sensor inputs to HRI from the Traffic Surveillance processes .

Logical Architecture Reference Flow: indicator_sign_control_data_for_hri

This data flow is used within the Manage Traffic function and contains the actual data from which instructions to the driver and traveler can be produced by indicators, variable message (vms), advisory beacons, and other types of signs on the roads (surface streets) in the vicinity of railroad grade crossings. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{crossing_id + hri_sign_control_data}.

Logical Architecture Reference Flow: rail_operations_advisories

This data flow contains advisory information for HRI vehicular traffic that has been derived from information received from rail operations.

Logical Architecture Reference Flow: rail_operations_device_command

This data flow contains HRI device commands that have been derived from information received from rail operations and provides for rail operations preemption capability.

Physical Architecture Flow Name: hri request

A request for highway-rail intersection status or a specific control request intended to modify HRI operation.

Logical Architecture Reference Flow: ro_requests

This data flow is generated in response to a need for hri status for rail operations.

Logical Architecture Reference Flow: tms_requests

This data flow is generated in response to a need for hri status for traffic management.

Physical Architecture Flow Name: roadway information system data

Information used to initialize, configure, and control roadside systems that provide driver information (e.g., dynamic message signs, highway advisory radio, beacon systems). This flow can provide message content and delivery attributes, local message store maintenance requests, control mode commands, status queries, and all other commands and associated parameters that support remote management of these systems.

Logical Architecture Reference Flow: dms_data_for_highways

This data flow is used within the Manage Traffic function and contains DMS data about text strings of information to be output to drivers on freeways in the geographic and/or jurisdictional area(s) served by the function. The data flow consists of the following data items each of which is defined in its own DDE: indicator_sign_control_data_for_highways.

Logical Architecture Reference Flow: dms_data_for_roads

This data flow is used within the Manage Traffic function and contains the actual data from which instructions to the driver and traveler can be produced by indicators at dynamic message (dms) and other types of signs on the roads. It consists of the following data items each of which is defined in its own DDE: indicator_sign_control_data_for_roads + indicator_sign_control_data_for_hri.

Logical Architecture Reference Flow: har_data_for_highways

This data flow contains the HAR data, both program and management, used to define the output of a Highway Advisory Radio (HAR) operating at the roadside on highways in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: har_identity + har_program + har_management_data.

Logical Architecture Reference Flow: har_data_for_roads

This data flow contains the HAR data, both program and management, used to define the output of a Highway Advisory Radio (HAR) operating at the roadside on roads (surface streets) in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: har_identity + har_program + har_management_data.

Logical Architecture Reference Flow: vehicle_sign_data

This data flow is used within the Manage Traffic function and contains data for use in producing in-vehicle signage displays. The information sent can fall into two categories - sign data and situation data. Sign data includes permanent fixed signs (e.g. STOP, YIELD, etc), temporary signs (e.g. detours), and variable message signs. Situation data provides information about traffic conditions, e.g. congestion and speed, on up to eight (8) links. Up to six (6) sets of indicator data may be contained within the data flow and they may be for the same or different types of indicator, or for incidents. All data is filtered so that the receiving processes only get that which is relevant to their local geographic area. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{vehicle_signage_output_identity + 6{vehicle_signage_output_data} + 8{vehicle_signage_traffic_data}}.

Physical Architecture Flow Name: sensor and surveillance control

Information used to configure and control sensor and surveillance systems at the roadside.

Logical Architecture Reference Flow: environment_sensor_configuration_data
This data flow is used within the Manage Traffic function to provide environmental sensor control commands. It consists of the following data items each of which is defined in its own DDE:

Logical Architecture Reference Flow: incident_video_image_control
This data flow is used within the Manage Traffic function and contains control parameters for closed circuit television (cctv) systems located used to provide incident management information. These parameters may cover things such as camera pan, tilt, and zoom, plus other picture controls.

Logical Architecture Reference Flow: sensor_configuration_data
This data flow provides control commands for advanced sensors, including video sensing systems .

Physical Architecture Flow Name: signal control data

Information used to configure and control traffic signal control systems.

Logical Architecture Reference Flow: indicator_control_data_for_roads
This data flow is used within the Manage Traffic function and contains the actual data from which instructions to the driver and traveler can be produced by indicators at the roadside on roads (surface streets) in the geographic and/or jurisdictional area(s) served by the function. It consists of the following data items each of which is defined in its own DDE: indicator_crossing_control_data_for_roads + indicator_junction_control_data + indicator_pedestrian_control_data.

Logical Architecture Reference Flow: indicator_control_monitoring_data_for_roads
This data flow is used within the Manage Traffic function. It contains the actual data from which instructions to the driver and traveler can be produced by indicators on the roads (surface streets) and at railroad grade crossings in the geographic and/or jurisdictional area(s) served by the function. In this case the data is used by a process to monitor the operation of the indicators rather than actual message output. The data flow consists of the following data items each of which is defined in its own DDE: indicator_control_data_for_roads + indicator_control_data_for_hri.

Traffic Management => **Toll Administration**

Physical Architecture Flow Name: demand management request

Request to change the demand for road facility use through pricing or other mechanisms.

Logical Architecture Reference Flow: toll_price_changes_request
This data flow is sent from the Manage Traffic function to the Provide Electronic Payment Services function and contains a request for a change to the current toll pricing structure that will help to influence a change in modal split of journeys currently being undertaken by travelers of all types, i.e. including drivers and transit users. It consists of the following data items each of which is defined in its own DDE: toll_segments + toll_price + toll_price_application_time + vehicle_type_for_tolls.

Logical Architecture Reference Flow: toll_price_direct_request
This data flow contains a request for the current prices being charged for toll segments on the road and highway network.

Traffic Management => **Traffic Operations Personnel**

Physical Architecture Flow Name: traffic operations data

Presentation of traffic operations data to the operator including traffic conditions, current operating status of traffic control equipment, maintenance activity status, incident status, and other information. This data keeps the operator appraised of current road network status, provides feedback to the operator as traffic control actions are implemented, and supports review of historical data and preparation for future traffic operations activities.

Logical Architecture Reference Flow: ttop_current_indicator_faults
This data flow is sent by the Manage Traffic function to the traffic operations personnel. It contains details of the data currently held in the store of current indicator faults and shows which indicators are faulty and the nature of the fault.

Logical Architecture Reference Flow: ttop_current_sensor_faults
This data flow is sent by the Manage Traffic function to the traffic operations personnel. It contains details of the data currently held in the store of current sensor faults and shows which sensors are faulty and the nature of the fault.

Logical Architecture Reference Flow: ttop_defined_incident_responses_data
This data flow is used by the Manage Traffic function to send the traffic operations personnel details of the data currently held in the store or defined incident responses used by the Manage Incidents facility.

Logical Architecture Reference Flow: ttop_demand_data
This data flow is sent to the traffic operations personnel by the Manage Traffic function and contains input data to be used in the calculation of demand forecasts.

Logical Architecture Reference Flow: ttop_demand_forecast_data
This data flow is used by the Manage Traffic function to send the Traffic Operations Personnel details of the predicted trends in traffic and travel demand and its effects on the overall transportation service.

Logical Architecture Reference Flow: ttop_demand_forecast_result
This data flow is used by the Manage Traffic function to send the traffic operations personnel results of progress in producing new data on traffic and travel demand and its effects on the overall transportation service.

Logical Architecture Reference Flow: ttop_demand_policy_activation_result
This data flow is sent to the traffic operations personnel from the Manage Traffic function and provides confirmation of the result of the implementation of the current demand management policy data.

Logical Architecture Reference Flow: ttop_demand_policy_information
This data flow is used by the Manage Traffic function to send traffic operations personnel details of the current travel demand controlling parameters being used to generate demand forecasts. It is sent in response to an operator request.

Logical Architecture Reference Flow: ttop_incident_information_display
This data flow is used by the Manage Traffic function to send requested information on either current or predicted incidents, plus defined incident response data to traffic operations personnel.

Logical Architecture Reference Flow: ttop_incident_video_image_output
This data flow is sent by the Manage Traffic function to the traffic operations personnel. It contains a video image from a closed circuit television (cctv) system which shows incident conditions at a point in the road and freeway network served by the function.

Logical Architecture Reference Flow: ttop_possible_defined_response_output
This data flow is sent from the Manage Traffic function to traffic operations personnel and contains the current contents of the store of possible defined incident responses.

Logical Architecture Reference Flow: ttop_possible_incidents_data
This data flow is sent from the Manage Traffic function and provides traffic operations personnel with details of incidents that are currently held in the store of possible incidents.

Logical Architecture Reference Flow: ttop_resource_response
This data flow is used by the Manage Traffic function to send the traffic operations personnel details of incident resource data used by the manage incidents data process.

Logical Architecture Reference Flow: ttop_traffic_control_information_display
This data flow is sent by the Manage Traffic function to the traffic operations personnel. It contains information on traffic conditions. The information may concern current, long term, or predicted traffic data, or a combination of some or all of these three.

Logical Architecture Reference Flow: ttop_undefined_response_details
This data flow is sent from the Manage Traffic function and provides traffic operations personnel with details of incidents for which no predefined response is available. The Manage Incidents facility will take no action concerning this type of incident until the traffic operations personnel have provided a defined response.

Logical Architecture Reference Flow: ttop_video_image_output
This data flow is sent by the Manage Traffic function to the traffic operations personnel. It contains a video image from a closed circuit television (cctv) system which shows traffic conditions at a point in the road and freeway network served by the function.

Logical Architecture Reference Flow: ttop_weather_information
This data flow is sent by the Manage Traffic function to the traffic operations personnel. It contains information on weather conditions. The information may contain current or predicted weather conditions, or a combination of both conditions.

Logical Architecture Reference Flow: ttop_wrong_way_detection
This data flow is used by the Manage Traffic function to send the traffic operations personnel details of wrong-way vehicle detection in reversible lanes.

Traffic Management => **Transit Management**

Physical Architecture Flow Name: demand management request

Request to change the demand for road facility use through pricing or other mechanisms.

Logical Architecture Reference Flow: transit_services_changes_request
This data flow is sent by the Manage Traffic function to the Manage Transit function and is a request to change the current transit services in response to changes in demand, or a desire to change the modal split currently being used by travelers.

Physical Architecture Flow Name: request transit information

Request for transit service information and current transit status.

Logical Architecture Reference Flow: transit_conditions_demand_request
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Traffic function. It is used to request details of the current state of transit vehicle operations for use in demand forecasting calculations carried out by the Manage Demand facility.

Logical Architecture Reference Flow: transit_fare_direct_request
This data flow contains a request for the current prices being charged for transit fares.

Logical Architecture Reference Flow: transit_services_demand_request
This data flow is sent from the Manage Traffic function to the Manage Transit function. It is a request for supply of details of the transit services and will be used in the preparation of demand forecasts by the Manage Demand facility.

The data flow consists of the following data items each of which is defined in its own DDE: tmc_identity + transit_services_request.

Physical Architecture Flow Name: traffic control priority status

Status of signal priority request functions at the roadside (e.g. enabled or disabled).

Logical Architecture Reference Flow: transit_highway_priority_given
This data flow is sent from the Manage Traffic function to the Manage Transit function. It contains confirmation that the requested priority has been given to transit vehicles throughout the freeway network served by the function. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: transit_ramp_priority_given
This data flow is sent from the Manage Transit function to the Manage Traffic function. It contains confirmation that the overall priority request for one or more transit vehicles over the ramp signals in a wide area as opposed to priority requests from individual vehicles at a particular set of ramp signals has been given. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: transit_road_priority_given
This data flow is sent from the Manage Traffic function to the Manage Transit function. It contains confirmation that the requested priority has been given to transit vehicles throughout the road network served by the function. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Physical Architecture Flow Name: traffic information for transit

Current and predicted traffic information and incident information.

Logical Architecture Reference Flow: prediction_data
This data flow is used within the Manage Traffic function and is also sent by that function to the Manage Transit and Provide Driver and Traveler Services function. It contains output from the predictive model process showing predictions of traffic data for route segments on the road and highway network served by the Manage Traffic function. The data flow consists of the following items each of which is defined in its own DDE: list_size + list_size{route_segment_identity + route_segment_volume_delay_predictions + route_segment_queue_delay_predictions + route_segment_speed_predictions + route_segment_occupancy_predictions}.

Logical Architecture Reference Flow: traffic_data_for_transit
This data flow contains information about environmental conditions, current incidents on the road, traffic flow state, and air quality data. The data flow consists of the following data items each of which is defined in its own DDE: list_size + roadway_environment_conditions + traffic_flow_state + link_state_data + current_incidents_data + area_air_quality_index.

Traffic Operations Personnel => **Traffic Management**

Physical Architecture Flow Name: traffic operations requests

Traffic operations requests for information, commands to adjust current traffic control strategies (e.g., adjust signal timing plans, change VMS messages).

Logical Architecture Reference Flow: ftop_defined_incident_response_data_request
This data flow is sent to the Manage Traffic function and enables Traffic Operations Personnel to request the data currently held by the store of defined incident responses held by the Manage Incidents facility.

Logical Architecture Reference Flow: ftop_defined_incident_response_data_update

This data flow is sent to the Manage Traffic function and enables Traffic Operations Personnel to amend data currently held by the store of defined incident responses held by the Manage Incidents facility.

Logical Architecture Reference Flow: ftop_demand_data_request
This data flow is sent to the Manage Traffic function from the traffic operations personnel and is a request for output of the current contents of the store of input data to be used in demand forecasting.

Logical Architecture Reference Flow: ftop_demand_data_update_request
This data flow is sent to the Manage Traffic function from the traffic operations personnel and is a request to update the current contents of the store of input data to be used in demand forecasting.

Logical Architecture Reference Flow: ftop_demand_forecast_request
This data flow is sent to the Manage Traffic function from the traffic operations personnel and is a request for a calculation of a new demand forecast based on the data currently available in the store of input data to be used in demand forecasting.

Logical Architecture Reference Flow: ftop_demand_policy_activation
This data flow is sent to the Manage Traffic function from the traffic operations personnel and is a request for the activation of the demand policies dictated by the data in the demand forecast data store.

Logical Architecture Reference Flow: ftop_demand_policy_information_request
This data flow is sent to the Manage Traffic function and is a request from the traffic operations personnel for the data currently held in the demand policy data store.

Logical Architecture Reference Flow: ftop_demand_policy_updates
This data flow is sent to the Manage Traffic function from the traffic operations personnel and contains permanent changes to the data held in the store of demand policy data.

Logical Architecture Reference Flow: ftop_incident_camera_action_request
This data flow is sent to the Manage Traffic function from the traffic operations personnel. It contains a request for a change to the operating parameters of a closed circuit television (cctv) system used to provide incident management data. These parameters may cover things such as camera pan, tilt, and zoom, plus other picture controls.

Logical Architecture Reference Flow: ftop_incident_data_amendment
This data flow is sent to the Manage Traffic function and enables traffic operations personnel to amend data currently held by the stores of current and predicted incidents held by the Manage Incidents facility.

Logical Architecture Reference Flow: ftop_incident_information_requests
This data flow is sent to the Manage Traffic function and contains requests for output of the data held in the stores of current or predicted incidents maintained by the Manage Incidents facility.

Logical Architecture Reference Flow: ftop_indicator_fault_data_input
This data flow is sent to the Manage Traffic function from the traffic operations personnel and contains new data for loading into the store of current indicator faults.

Logical Architecture Reference Flow: ftop_indicator_fault_data_request
This data flow is sent to the Manage Traffic function from the traffic operations personnel and is a request for output of the contents of the store of current indicator faults.

Logical Architecture Reference Flow: ftop_indicator_fault_data_update
This data flow is sent to the Manage Traffic function from the traffic operations personnel and is an update data in the store of current indicator faults.

Logical Architecture Reference Flow: ftop_output_possible_defined_reponses
This data flow is sent from the traffic operations personnel to the Manage Traffic function and contains a request for output of the current contents of the store of possible defined incident responses to enable traffic operations personnel to review them to see if any should be made available for on-line use as and when incidents occur.

Logical Architecture Reference Flow: ftop_request_possible_incidents_data
This data flow is sent from the traffic operations personnel to the Manage Traffic function and contains requests for the output of some or all of the data currently held in the store of possible incidents maintained by the Manage Incidents facility.

Logical Architecture Reference Flow: ftop_resource_request
This data flow is sent to the Manage Traffic function from the traffic operations personnel and is a request for output of the resource data contents to be used in the manage incidents data process.

Logical Architecture Reference Flow: ftop_sensor_fault_data_input
This data flow is sent to the Manage Traffic function from the traffic operations personnel and contains new data for loading into the store of current sensor faults.

Logical Architecture Reference Flow: ftop_strategy_override
This data flow is sent from the traffic operations personnel to the Manage Traffic function and contains an override for some aspect of the current method of traffic control currently being implemented.

Logical Architecture Reference Flow: ftop_traffic_data_parameter_updates
This data flow is sent from the traffic operations personnel to the Manage Traffic function and contains updates to the parameters used to define the traffic data that is retrieved in response to information requests from the media and from other functions within ITS that are outside of the Manage Traffic function.

Logical Architecture Reference Flow: ftop_traffic_information_requests
This data flow is sent from the traffic operations personnel to the Manage Traffic function and contains requests for traffic information to enable the personnel to review and seek clarification of the way the current traffic situation is being managed, or something which is taking place on the road or highway, etc.

Logical Architecture Reference Flow: ftop_update_defined_incident_responses
This data flow is sent from the traffic operations personnel to the Manage Traffic function and contains a request for possible defined incident responses to be transferred from their store to that of the actual defined responses that will be used as and when incidents occur.

Logical Architecture Reference Flow: ftop_video_camera_strategy_change
This data flow contains a request for a change to the strategy of operation of a closed circuit television (cctv) system used to provide traffic surveillance data. This strategy does not cover specific camera action such as things as pan, tilt, and zoom, plus other picture controls.

Logical Architecture Reference Flow: ftop_weather_request_information
This data flow is sent from the traffic operations personnel to the Manage Traffic function and contains requests for weather conditions and service information.

Transit Management => **Traffic Management**

Physical Architecture Flow Name: demand management response

Response to various demand management change requests indicating level of compliance with request.

Logical Architecture Reference Flow: transit_services_changes_response
This data flow is sent by the Manage Transit function to the Manage Traffic function and is the response to the previous request for changes in the transit services.

Physical Architecture Flow Name: traffic control priority request

Request for signal priority at one or more intersections along a particular route.

Logical Architecture Reference Flow: transit_highway_overall_priority

This data flow is sent from the Manage Transit function to the Manage Traffic function. It contains requests and information about the overall priority which should be given to one or more transit vehicles at all points in the freeway network served by the function, as opposed to priority requests from individual vehicles at specific locations. This priority will apply at an individual junction, or along a selected transit route if that is specified. The data flow size assumption below is based on an assumed percentage of transit vehicles running late (TRANSIT_VEH_DEVS) and the percentage of freeway miles to total miles (HIGHWAY_MILES/MILES).

Logical Architecture Reference Flow: transit_ramp_overall_priority

This data flow is sent from the Manage Transit function to the Manage Traffic function. It contains requests and information on the overall priority which should be given to one or more transit vehicles over a wide area as opposed to priority requests from individual vehicles at a particular set of ramp signals.

Logical Architecture Reference Flow: transit_road_overall_priority

This data flow is sent from the Manage Transit function to the Manage Traffic function. It contains requests and information about the overall priority which should be given to one or more transit vehicles at all junctions and/or pedestrian crossings in the road network served by the function, as opposed to priority requests for individual vehicles at specific locations. As this is a 'blanket' application of priority, no list of indicators is needed. The size assumption is based on an assumed percentage of transit vehicles running late (TRANSIT_VEH_DEVS) and the percentage of street miles to total miles (ROAD_MILES/MILES).

Physical Architecture Flow Name: transit system data

Current transit system operations information indicating current transit routes, the level of service on each route, and the progress of individual vehicles along their routes for use in forecasting demand and estimating current transportation network performance.

Logical Architecture Reference Flow: transit_fare_direct_details

This data flow contains details of the fares being currently charged for transit services. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{transit_route_number + transit_route_segment_list + transit_user_category + transit_route_use_time}.

Logical Architecture Reference Flow: transit_probe_data

This data flow contains the location of the transit vehicle on each part of its route, i.e., each transit route segment. This data will be used along with other probe data to calculate the link speed or travel time. The data flow consists of the following items each of which is defined in its own DDE: list_size + transit_route_number + transit_route_segment_number + transit_vehicle_location_for_store + transit_vehicle_time.

Logical Architecture Reference Flow: transit_running_data_for_demand

This data flow is sent from the Manage Transit function to the Manage Traffic function. It is used to provide data on the current state of transit operation for use in demand forecasting calculations carried out by the Manage Demand facility and consists of the following items each of which is defined in its own DDE: transit_vehicle_passenger_loading + transit_vehicle_deviation_update + transit_vehicle_running_times + transit_vehicle_schedule_deviations + transit_vehicle_eta.

Logical Architecture Reference Flow: transit_services_for_demand

This data flow is sent from the Manage Transit function to the Manage Traffic function. It contains a complete set of all the transit routes and the services that run upon them, including timings, etc. that are provided by the transit fleet from which the data was requested, for use in the calculation of demand forecasts by the Manage Demand facility. The data flow consists of the following data items each of which is defined in its own DDE: transit_services.

Weather Service => **Traffic Management**

Physical Architecture Flow Name: weather information

Accumulated predicted and current weather data (e.g., temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.).

Logical Architecture Reference Flow: fws_current_weather
This data flow is sent to the Manage Traffic function and the Provide Driver and Traveler Services functions. It contains details of the current weather conditions, e.g. temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.

Logical Architecture Reference Flow: fws_predicted_weather
This data flow is sent to the Manage Traffic and Provide Driver and Traveler Services functions. It contains details of the predicted weather conditions, e.g. temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.

2.19.4 Subsystem Architecture Flow Diagram

TMS

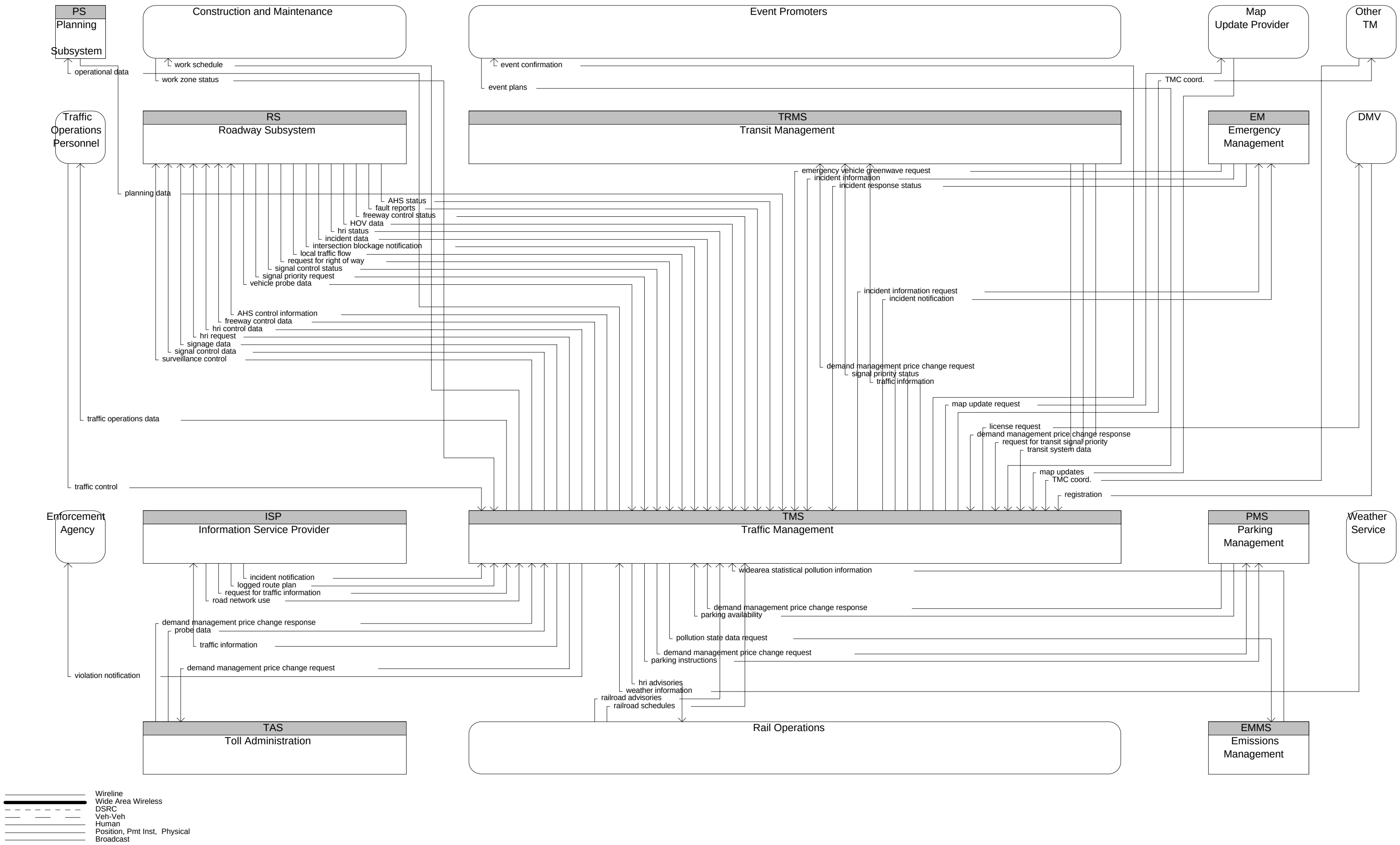


Figure 2.19-2 Architecture Flow Diagram for TMS

2.20 Transit Management

The transit management subsystem manages transit vehicle fleets and coordinates with other modes and transportation services. It provides operations, maintenance, customer information, planning and management functions for the transit property. It spans distinct central dispatch and garage management systems and supports the spectrum of fixed route, flexible route, and paratransit services. The subsystem's interfaces allow for communication between transit departments and with other operating entities such as emergency response services and traffic management systems. This subsystem receives special event and real-time incident data from the traffic management subsystem. It provides current transit operations data to other center subsystems. The Transit Management Subsystem collects and stores accurate ridership levels and implements corresponding fare structures. It collects operational and maintenance data from transit vehicles, manages vehicle service histories, and assigns drivers and maintenance personnel to vehicles and routes. The Transit Management Subsystem also provides the capability for automated planning and scheduling of public transit operations. It furnishes travelers with real-time travel information, continuously updated schedules, schedule adherence information, transfer options, and transit routes and fares. In addition, the monitoring of key transit locations with both video and audio systems is provided with automatic alerting of operators and police of potential incidents including support for traveler activated alarms.

2.20.1 Alternative Configurations

Transit management can support any management organization responsible for moving people. This include taxis, transit agencies, tour agencies. Figure.. Communication between transit management subsystems and intermodal providers provide schedule coordination for smooth travel options for the public. Communication between transit vehicles and the management center provides for tracking and vehicle management.

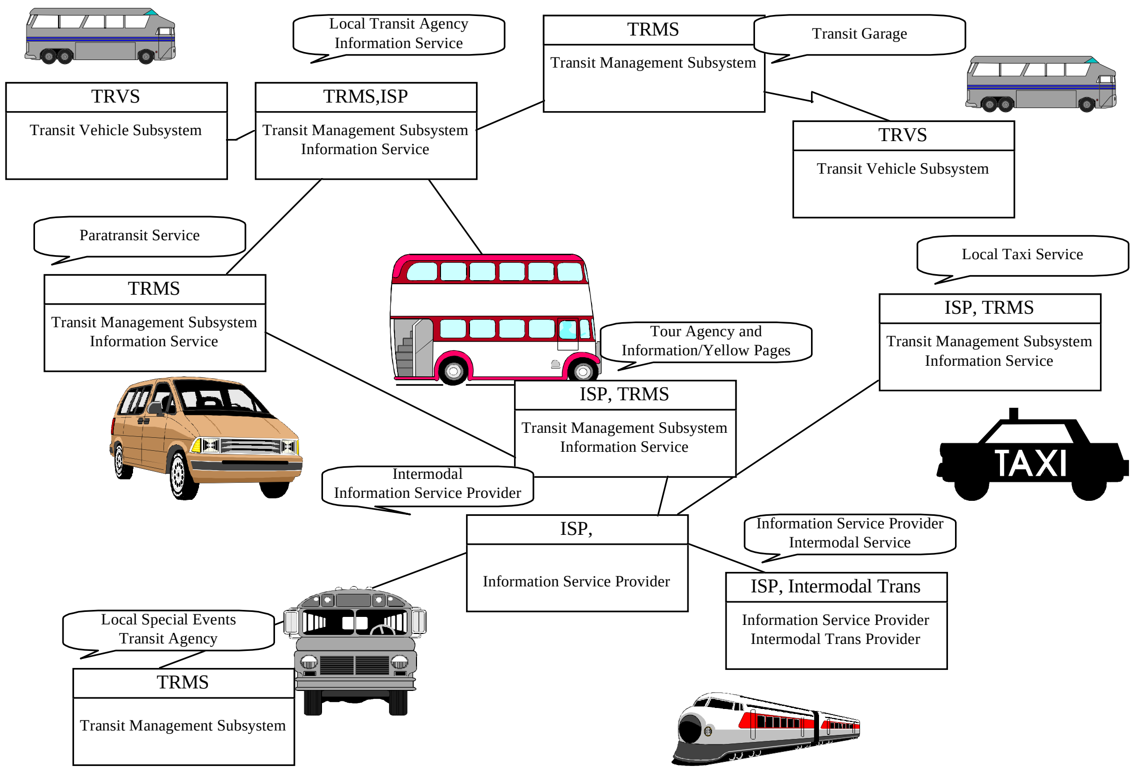


Figure 2.20-1 Alternative Configurations for Transit Management

2.20.2 Subsystem Equipment Packages and Supporting Process Specifications for TRMS

Transit Center Fare and Load Management

This Equipment package provides the capability to accept collected data required to determine accurate ridership levels and implement variable and flexible fare structures. Support shall be provided for the traveler for use of a fare medium for all applicable surface transportation services, to pay without stopping, have payment media automatically identified as void and/or invalid and eligibility verified, and allow for third party payment. In addition, capability to provide expansion into other uses for payment medium such as retail and telephone and for off-line billing for fares paid by agencies shall be supported. This Equipment package also supports the capability for two-way voice communication between the transit vehicle driver and a facility, two-way data communication between the transit vehicles and a facility, sensor data to be transmitted from the transit vehicles to a facility, and data transmission from individual facilities to a central facility for processing/analysis if desired. These capabilities shall be provided through a workstation type processor with GUI, high capacity storage, ride share software housed in a building with dialup lines and wireline telephone and require integration with an existing Transit Center Tracking and Dispatch Equipment package.

Process Specifications

4.6.8 Manage Transit Vehicle Advanced Payments

Overview: This process shall act as the interface for advanced payment of tolls and parking lot charges from the transit user. Requests for these advanced payments shall be passed to other processes in the Provide Electronic Payment Services function for transaction processing. The process shall ensure that the response to these requests from transit users is returned to the transit vehicle from which it was made.

5.4.4 Process Fare Payment Violations

Overview: This process shall manage the details of fare payment violations reported by the Provide Electronic Payments function. The process shall use the parameters in the store of fare payment violation (enforcement) data to process and send the data to the correct law enforcement agency. This process shall also maintain the fare payment enforcement data store, entering all information received from other processes.

5.4.5 Process Vehicle Fare Collection Violations

Overview: This process shall manage the details of fare collection violations reported by the Manage Transit function that have taken place on-board a transit vehicle. The process shall use the parameters in the store of vehicle fare collection violation (enforcement) data to process and send the information to the correct law enforcement agency. This process shall also maintain the vehicle fare collection enforcement data store, entering all information received from other processes.

5.4.7 Process Roadside Fare Collection Violations

Overview: This process shall manage the details of fare collection violations reported by the Manage Transit function that have taken place at the roadside, i.e., at a transit stop. The process shall use the parameters in the store of roadside fare collection violation (enforcement) data to process and send the information to the correct law enforcement agency. This process shall also maintain the roadside fare collection enforcement data store, entering all information received from other processes.

7.3.1.1 Register for Advanced Transit Fare Payment

Overview: This process shall be responsible for responding to requests for transit fares to be paid in advance. The advanced transit fare data shall be forwarded by the process to other processes for the actual cost to be obtained and the payment transactions initiated.

7.3.1.2 Determine Advanced Transit Fares

Overview: This process shall be responsible for receiving a request to pay an advanced transit fare. It shall obtain the required transit fare data from a local store of transit fares and shall then forward the data to the billing processes. The store of fare data shall be maintained by another process.

7.3.1.3 Manage Transit Fare Financial Processing

Overview: This process shall be responsible for maintaining a log of all the transactions carried out by other processes in the Process Electronic Transit Fare Payment facility. The identity of the payee shall have been removed from the data before it is stored. At periodic intervals the process shall output the accumulated records to the transit fleet manager, the transit system operator and to another process in the Provide Electronic Payment Services function. The first two outputs shall be produced by the process in visual form, which may be either hardcopy or a visual display. The process shall also be responsible for sending details of transactions to the financial institution to enable the users to be billed through their credit identities.

7.3.1.4 Check for Advanced Transit Fare Payment

Overview: This process shall be responsible for checking to see if the required transit fare payment has already been made. The process shall determine the existence of an advance payment for the transit fare by comparing the received payment information with that in the store containing the list of advanced payments. If the payment has already been made then the process shall remove the requirement for local billing and remove the record of the advanced payment from the store. Details of each payment transaction shall be sent by the process to another process with the payment information received from the transit user removed.

7.3.1.5 Bill Transit User for Transit Fare

Overview: This process shall be responsible for obtaining payment for a transit fare transaction using data provided by the transit user. The process shall achieve this either by requesting that the fare be deducted from the credit being stored by the tag that is acting as the payment instrument for the transit user, or informing the transit user that payment for the fare will be debited to the credit identity provided by the tag. Before sending data to the tag, the process shall check that the transit user's credit identity is not already in the list of bad payers, and if it is request an image of the user which can be forwarded to the appropriate enforcement agency via another process. The tag may be in the form of cash, some type of credit or debit card, an electronic purse, or an intelligent transit ticket upon which pre-payment has been recorded, etc. Details of the transaction shall always be sent by the process to the process that manages transit fare transactions. The process shall load details of advanced transit fare payments into a data store for use by another process when the transit user eventually passes a fare payment point. If requested the process shall provide a copy of the current bad payers list to processes in the transit vehicle fare collection facility for use in on-board payment validation.

7.3.1.6 Collect Bad Transit Fare Payment Data

Overview: This process shall be responsible for maintaining a data store containing a list of invalid transit user credit identities. The process shall use this data to check credit identities provided for checking by the billing process. This checking shall ensure that the current transit fare payment transaction is using a credit identity that has not previously had an invalid transaction. Details of possible invalid credit identities shall be sent by the process to the financial institution for verification. The process shall also receive from the financial institution details of invalid payment instrument data that has been found by other means.

7.3.1.7 Update Transit Fare Data

Overview: This process shall be responsible for managing the store of data that provides the actual value of transit fares for each segment of each regular transit route. The process shall also act as the interface through which the transit system operator can output and make changes to the stored data, and copies of this data can be provided to the Centralized Payments facility on request. The process shall support inputs from the transit system operator in both manual and audio form, and shall provide its outputs in audible and visual forms. It shall enable the visual output to be either in hardcopy, or as a display. The process shall automatically output the new fares for use by process on-board a transit vehicle and at the roadside, as well as by other ITS functions.

7.4.1.5 Process Transit User Other Services Payments

Overview: This process shall be responsible for collecting advance payments for other(yellow pages) services. The transaction data shall be provided by processes in the Manage Transit function in response to reservation requests from a transit user either at the roadside, i.e. a transit stop, or on-board a transit vehicle. The process shall send the received transaction data to the financial institution and shall send the response to the requesting process. It shall also send details of the transaction to another process for entry into a store of transaction records.

Transit Center Fixed-Route Operations

This Equipment package enhances the planning and scheduling associated with fixed route transit services. The package allows fixed-route services to develop, print and disseminate schedules and automatically updates customer service operator systems with the most current schedule information. Current vehicle schedule adherence and optimum scenarios for schedule adjustment shall also be provided.

Process Specifications

4.1.6 Manage Transit Vehicle Operations Data

Overview: This process shall manage the store of transit vehicle operating data. When any new data is received from another process, this process shall load it into the data store. This process shall also retrieve selected data on request from other processes in the Manage Transit function.

4.2.2 Provide Transit Plans Store Interface

Overview: This process shall provide the interface to the store of current regular transit plans, i.e., routes and schedules and demand responsive transit schedules. The process shall enable the store to be used by the Demand Responsive Transit facility as a source of data about regular transit services when it is generating its schedules. The demand responsive transit schedule data shall be accessible as input to the regular transit route and schedule generation processes.

4.2.3.1 Generate Transit Routes

Overview: This process shall generate new transit routes. The process shall use parameters set up by the transit fleet manager, operational data for the current routes and schedules, plus the current routes and digitized map data, as sources of input from which the new routes are generated. The process shall also use the requested input data containing the demand responsive transit routes and schedules. The generation of new routes by the process shall be initiated as a result of data received from the transit fleet manager interface process, with the output being sent to other processes for storage. The output data produced by the process shall include sufficient data for a specialist map data provider to generate maps showing transit routes and stops, either as separate data or as part of the general digitized map data provided to other ITS functions.

4.2.3.2 Generate Schedules

Overview: This process shall generate new transit schedules for use by the regular transit operation. The process shall use parameters set up by the transit fleet manager, operational data for the current routes and schedules, plus the current routes and schedules themselves, as sources of input from which the new schedules are generated. The process shall also use the data containing the demand responsive transit routes and schedules to generate the new schedules. The generation of new schedules by the process shall be initiated as a result of data received from the transit fleet manager interface process or a request for services to a parking lot. The process shall send its output to another process for storage.

4.2.3.4 Provide Transit Fleet Manager Interface for Services Generation

Overview: This process shall provide the interface through which the transit fleet manager controls the generation of new routes and schedules (transit services). The transit fleet manager shall be able to review and update the parameters used by the routes and schedules generation processes and to initiate these processes. This process shall also act as the interface through which the Manage Demand facility in the Manage Traffic function can request changes to the current routes and schedules in its efforts to adjust the modal split of travelers' trips in order to make the most efficient use of the road and highway network served by the local ITS functions. The input and output forms shall include those that are suitable for travelers with physical disabilities.

4.2.3.5 Manage Transit Operational Data Store

Overview: This process shall collect transit operational data and load it into a data store for use by the routes and schedules generation processes. The data shall be provided to this process by other processes in the Manage Transit function and shall enable an accurate picture of how routes and schedules are currently operating in terms of the numbers of vehicles that are available, the numbers of passengers that they are carrying, and the numbers of passengers passing through each roadside facility (transit stop).

4.2.3.6 Produce Transit Service Data for Manage Transit Use

Overview: This process shall obtain transit routes and services data and distribute it internally to other processes in the Manage Transit function. The process shall only provide its outputs when fresh data is received from another process. If this does not happen for a long period of time (days) then the process shall initiate its own request for fresh data.

Transit Center Information Services

This equipment package collects the latest available information for a transit service and makes it available to transit customers and to Information Service Providers for further distribution. Customers are provided information at transit stops and other public transportation areas before they embark and on-board the transit vehicle once they are enroute. Information provided can include the latest available information on transit routes, schedules, transfer options, fares, real-time schedule adherence, current incidents, weather conditions, and special events. In addition to general service information, tailored information (e.g., itineraries) are provided to individual transit users.

Process Specifications

4.1.5 Provide Transit Vehicle Status Information

Overview: This process shall provide transit vehicle operational data to processes within the Manage Transit function, and on request to the transit fleet manager and the Manage Travel Demand facility in the Manage Traffic function. This process shall also provide transit probe and AVL information to the Manage Traffic function. The data shall be obtained by this process from another process that manages a store of transit vehicle operating data.

4.1.6 Manage Transit Vehicle Operations Data

Overview: This process shall manage the store of transit vehicle operating data. When any new data is received from another process, this process shall load it into the data store. This process shall also retrieve selected data on request from other processes in the Manage Transit function.

4.2.3.3 Produce Transit Service Data for External Use

Overview: This process shall obtain transit routes and services data and distribute it to ITS functions that are outside the transit center. The process shall run when a request for data is received from an external source, or when fresh data is received. In the latter case, the data shall only be sent by the process to the intermodal transportation service provider. For data requests that include an origin and a destination, the process shall only provide details of the transit service(s) that link the two points. The details shall only cover those portion(s) of the service(s) that are needed to complete the requested trip and not full details of the services.

4.6.8 Manage Transit Vehicle Advanced Payments

Overview: This process shall act as the interface for advanced payment of tolls and parking lot charges from the transit user. Requests for these advanced payments shall be passed to other processes in the Provide Electronic Payment Services function for transaction processing. The process shall ensure that the response to these requests from transit users is returned to the transit vehicle from which it was made.

Transit Center Multi-Modal Coordination

This Equipment package provides the transit management subsystem the capability to determine the need for transit priority on routes and at certain intersections and request transit vehicle priority at these locations. It also supports schedule coordination between transit properties and coordinates with other surface and air transportation modes.

Process Specifications

4.1.2.4 Provide Transit Vehicle Correction Data Output Interface

Overview: This process shall provide the interface through which intermodal transportation service providers are informed of a transit vehicle schedule deviation. The output delivered by the process results from input received from another process in the Manage Transit function, and shall relate to the deviation of an individual transit vehicle. The process shall provide the output in a form that enables adjustments to be made to any connecting services being provided by the intermodal supplier so that transit users are not inconvenienced by the deviation of a transit vehicle on one service. A zero (or null) output shall be provided when no deviations are present.

4.1.4 Manage Transit Vehicle Deviations

Overview: This process shall manage large deviations of individual transit vehicles, deviations in rural areas, and deviations of large numbers of vehicles. The process shall generate the necessary corrective actions which may involve more than the vehicles concerned and more far reaching action, such as, the introduction of extra vehicles, wide area signal preemption by the Manage Traffic function, the premature termination of some services, etc. All corrective actions generated by this process shall be subject to the approval of the transit fleet manager before being implemented. Confirmation that the requested overall priority has been given by the Manage Traffic function shall be received by the process.

4.1.5 Provide Transit Vehicle Status Information

Overview: This process shall provide transit vehicle operational data to processes within the Manage Transit function, and on request to the transit fleet manager and the Manage Travel Demand facility in the Manage Traffic function. This process shall also provide transit probe and AVL information to the Manage Traffic function. The data shall be obtained by this process from another process that manages a store of transit vehicle operating data.

4.1.7 Provide Transit Vehicle Deviation Data Output Interface

Overview: This process shall provide the interface through which intermodal transportation service providers are informed of transit vehicle schedule deviations. The output delivered by the process shall result from input received from another process in the Manage Transit function, and shall relate to the deviation of a number of transit vehicles such that the disruption will affect several services, possibly on different routes. The process shall provide the output in a form that enables adjustments to be made to any connecting services being provided by the intermodal supplier so that transit users are not inconvenienced by the deviations. A zero (or null) output shall be provided when no deviations are present.

4.2.3.2 Generate Schedules

Overview: This process shall generate new transit schedules for use by the regular transit operation. The process shall use parameters set up by the transit fleet manager, operational data for the current routes and schedules, plus the current routes and schedules themselves, as sources of input from which the new schedules are generated. The process shall also use the data containing the demand responsive transit routes and schedules to generate the new schedules. The generation of new schedules by the process shall be initiated as a result of data received from the transit fleet manager interface process or a request for services to a parking lot. The process shall send its output to another process for storage.

4.2.3.7 Provide Interface for Other TRM Data

Overview: This process shall provide the interface through which transit routes and schedules can be exchanged with other transit centers (Other TRM). This data shall be output when data is received from another (local) process and shall enable coordination between services provided by adjacent transit operations, particularly where they serve the same geographic areas. The process shall also provide routes and schedules to the local process when the data is received from other transit centers.

4.2.3.8 Provide Interface for Transit Service Raw Data

Overview: This process shall provide and manage the interface to the store in which the raw transit service data is held. This data shall be sent to the process by the routes and schedules generation processes, which are the only other processes permitted to access the store, and then in read-only mode. The received data shall be loaded into the store and distributed by this process to the three processes that are responsible for distributing the data within the transit center (TRM), to other local ITS functions and to other transit centers (Other TRM) respectively. The process shall read data from the store and return it to whichever of the other three processes has made a data request. Data shall also be received by the process from other transit centers (Other TRM) and from intermodal transportation service providers. The process shall load this data into the data store for use by the local route and schedule generation processes.

Transit Center Paratransit Operations

This Equipment package provides the capability to automate the planning and scheduling, allowing improvements in paratransit routes and services to develop, printing and disseminating schedules, and automatically updating customer service operator systems with the most current schedule. In addition, this Equipment package provides the capability to assign drivers to routes in a fair manner while minimizing labor and overtime services, including driver preferences and qualifications, and automatically tracking and validating the number of work hours performed by each individual driver. These capabilities shall be provided through the utilization of dispatch and fleet management software running on a workstation type processor.

Process Specifications

4.1.6 Manage Transit Vehicle Operations Data

Overview: This process shall manage the store of transit vehicle operating data. When any new data is received from another process, this process shall load it into the data store. This process shall also retrieve selected data on request from other processes in the Manage Transit function.

4.2.1.1 Process Demand Responsive Transit Trip Request

Overview: This process shall provide the interface through which processes in the Provide Driver and Traveler Service function can gain access to the Provide Demand Responsive Transit Service facility. The process shall enable the interface to support the receipt of trip requests, their transfer to another process for the actual demand responsive schedule generation, the output of the proposed schedule and their (possible) subsequent confirmation. The process shall store the input and schedule data relating to each request until such time as the request is confirmed or the data in the request is no longer valid, e.g. the time(s) used in the proposed schedule has(ve) passed. The confirmation of a particular schedule shall be sent by the process to another process that will enable the schedule to be implemented.

4.2.1.2 Compute Demand Responsive Transit Vehicle Availability

Overview: This process shall provide the facility for the calculation of the location and availability of transit vehicles for use in demand responsive transit operations. The process shall base its calculation on the vehicle's current location and on the output from a process that determines vehicle availability from data input to sensors. The output shall be loaded by the process into a store for use by another process.

4.2.1.3 Generate Demand Responsive Transit Schedule and Routes

Overview: This process shall provide dynamic routing and scheduling of transit vehicles so that a demand responsive transit service can be provided. The generation of the specific route and schedule by the process shall be initiated by a request from the management process. The choice of route and schedule produced by the process shall depend on what other demand responsive transit schedules have been planned, the availability and location of vehicles and the relevance of any regular transit routes and schedules. The process shall send its output to another process for output to the requesting process, and shall also load it into a data store for use if the schedule is later confirmed.

4.2.1.4 Confirm Demand Responsive Transit Schedule and Route

Overview: This process shall provide output when a demand responsive transit schedule is confirmed. The outputs shall contain details of the schedule and shall be sent to the transit fleet manager and to processes that provide interfaces to the transit driver, a store of data used by the regular transit routes and schedule generation processes, and the transit driver schedule generation processes. The process shall obtain the data for the outputs from the store of data provided by the schedule generation process.

Transit Center Security

This Equipment package provides the capability to monitor key transit locations and transit vehicles with both video and audio systems automatically alerting operators and police of potential incidents and supporting traveler activated alarms. The monitoring equipment shall also include capabilities to assist in responding to terrorist incidents.

Process Specifications

4.4.1.1 Manage Transit Security

Overview: This process shall manage the security in the transit system by monitoring for potential incidents. Data shall be obtained by the process from a variety of sources and assessed for any security problems. Problems shall be passed by the process to the transit system operator for review and the required action. Information about incidents shall also be sent by this process to another process for output to the media, using interface parameters set up by the transit system operator.

4.4.1.3 Provide Transit System Operator Security Interface

Overview: This process shall provide an interface for the transit system operator to identify and act upon potential information security problems and emergencies. This information shall be provided by other processes through input data flows. This process shall also provide the capability for the transit system operator to update parameters that control the output of data about the potential security problems to the media. The input and output forms shall include those that are suitable for travelers with physical disabilities.

4.4.1.4 Provide Transit External Interface for Emergencies

Overview: This process shall provide the interface through which information about security problems and emergencies detected within the transit system are distributed directly to the media and other information systems. This process shall construct its output from the data supplied by other processes. This data shall contain parameters that define the way (format, content, etc.) in which the information is output by the process. The input and output forms shall include those that are suitable for travelers with physical disabilities.

4.4.1.6 Collect Transit Vehicle Emergency Information

Overview: This process shall collect data about emergencies that occur on-board transit vehicles for output to the media and the Manage Emergency Services function. These emergencies may be reported by either the transit driver or a transit user, the latter through such interfaces as panic buttons, alarm switches, etc. For output to the media interface process, the data shall be combined with the data in the media interface parameters data store.

4.4.2 Coordinate Multiple Agency Responses to Incidents

Overview: This process shall provide transit fleet managers with an interface through which they can control the coordination data sent to the Manage Emergency Services function following the detection of a security problem or emergency within the transit operations network by other processes. The process shall use data from the store of predefined responses to security problems and emergencies in the outputs that it sends to the Manage Emergency Services function. If no match can be found then the process shall send all the available data to the transit fleet manager for action. The input and output forms shall include those that are suitable for travelers with physical disabilities.

4.4.3 Generate Responses for Incidents

Overview: This process shall provide the interface through which the transit fleet manager can enter and review predefined responses to security problems and emergencies that have been detected by other processes within the Manage Transit function. This data shall be stored in a form which can be used by another process to provide coordination data to the Manage Emergency Services function. The input and output forms shall include those that are suitable for travelers with physical disabilities.

7.3.3 Get Transit User Image for Violation

Overview: This process shall be responsible for obtaining an image of a transit user who is trying to carry out an invalid fare payment transaction. The process shall send the image request to other processes either at the roadside, i.e. a transit stop, or on-board a transit vehicle, depending on where the transaction is being attempted. However if the collection method is set to batch, then the process shall take no further action, as an image of the offending transit user will not be available. When the image is received, the process shall use it to form part of the data sent to a processes in the Manage Emergency Services function for forwarding to the appropriate enforcement agency.

Transit Center Tracking and Dispatch

This Equipment package provides the capabilities for monitoring transit vehicle locations and determining vehicle schedule adherence. The Equipment package shall also furnish users with real-time travel related information, continuously updated with real-time information from each transit system within the local area of jurisdiction, inclusive of all transportation modes, from all providers of transportation services, and provide users with the latest available information on transit routes, schedules, transfer options, fares, real-time schedule adherence, current incidents conditions, weather conditions, and special events. This Equipment package also supports the capability for two-way voice communication between the transit vehicle driver and a facility, two-way data communication between the transit vehicles and a facility.

Process Specifications

4.1.6 Manage Transit Vehicle Operations Data

Overview: This process shall manage the store of transit vehicle operating data. When any new data is received from another process, this process shall load it into the data store. This process shall also retrieve selected data on request from other processes in the Manage Transit function.

4.2.3.9 Update Transit Map Data

Overview: This process shall provide updates to the store of digitized map data used by the transit route generation process and as the background for displays of transit services requested by the transit fleet manager. The process shall obtain the new data from a specialist data supplier or some other appropriate data source, after receiving an update request from the transit fleet manager interface process within the function. The processes requiring data for use in transit route generation and as the background to displays will read the data from the store loaded by this process.

Transit Garage Maintenance
This Equipment package provides advanced maintenance functions for the transit property. It collects operational and maintenance data from transit vehicles, manages vehicle service histories, and monitors drivers and vehicles. It collects vehicle mileage data and uses it to automatically generate preventative maintenance schedules for each vehicle by utilizing vehicle tracking data from a prerequisite vehicle tracking equipment package. In addition, it provides information to proper service personnel to support maintenance activities and records and verifies that maintenance work was performed. This equipment package receives special events and real-time incident data from the traffic management subsystem and assigns operators to vehicles and transit routes. Garage maintenance also receives information about incidents involving transit vehicles from the TMC in order to dispatch tow trucks and other repair vehicles.

Process Specifications

4.3.1 Monitor Transit Vehicle Condition

Overview: This process shall monitor the condition of a transit vehicle. It shall use the transit vehicle maintenance specification to analyze brake, drive train, sensors, fuel, steering, tire, processor, communications equipment and transit vehicle mileage to identify mileage based maintenance, out-of-specification or imminent failure conditions. The data resulting from this analysis shall be loaded by the process into the store of transit vehicle operations data, through the output flow transit vehicle maintenance. This data is then sent to the process that generates transit vehicle maintenance schedules.

4.3.2 Generate Transit Vehicle Maintenance Schedules

Overview: This process shall generate transit vehicle maintenance schedules and includes what and when maintenance or repair is to be performed. Transit vehicle availability listings (current and forecast) shall also be generated by the process to support transit vehicle assignment planning. The maintenance and/or repair that is to be performed on the transit vehicle shall be scheduled by the process for a specific month, week, day(s) and hour(s). The availability of the transit vehicle that is also output by the process shall be based upon the transit vehicle maintenance schedule. The process shall load each transit vehicle maintenance schedule that it produces into the store of transit vehicle operations data, through the process that maintains this data store.

4.3.3 Generate Technician Work Assignments

Overview: This process shall assign transit maintenance personnel to a transit vehicle maintenance schedule. The maintenance schedule shall be received from another process and shall define what and when maintenance repair is to be performed to a specific transit vehicle. The process shall base the personnel assignment upon details about the personnel obtained from the transit fleet manager and held in a local data store. These details shall comprise personnel eligibility, work assignments, preferences and seniority. The process shall also provide these details to the transit fleet manager on request. When a work assignment has been generated, the process shall send it to the transit maintenance personnel and also to the process that monitors and verifies maintenance work activity. The input and output forms shall include those that are suitable for travelers with physical disabilities.

4.3.4 Monitor And Verify Maintenance Activity

Overview: This process shall verify that the transit vehicle maintenance activities were performed correctly and that a time stamped maintenance log for record keeping was generated. The correctness of the maintenance activities shall be judged by the process against the transit vehicle's status, the maintenance personnel's work assignment and the transit maintenance schedules produced by other processes. The process shall save a time stamped record of all the maintenance activities performed on the vehicle into the transit vehicle maintenance log.

4.3.5 Report Transit Vehicle Information

Overview: This process shall provide the transit fleet managers with the capability of requesting and receiving transit vehicle maintenance information. The process shall obtain the data for each request from the store of transit vehicle operations data, through the process that manages the data store, and shall produce the output to the transit fleet manager in an easily understood form. The input and output forms shall include those that are suitable for travelers with physical disabilities.

4.3.6 Update Transit Vehicle Information

Overview: This process shall provide the transit maintenance personnel with the capability to update transit vehicle maintenance information. The process shall send the data received from the transit maintenance personnel to the transit vehicle operations data store management process for use by other processes.

4.3.7 Manage Transit Vehicle Operations Data Store

Overview: This process shall manage the store of transit vehicle operations data. It shall be able to load data it receives about vehicle maintenance into the store and provide that data on request to other processes.

Transit Garage Operations

This Equipment package automates and supports the assignment of transit vehicles and drivers to enhance the daily operation of a transit service. It provides the capability to assign drivers to routes or service areas in a fair manner while minimizing labor and overtime services, considering driver preferences and qualifications, and automatically tracking and validating the number of work hours performed by each individual driver.

Process Specifications

4.5.1 Assess Transit Driver Performance

Overview: This process shall assess the transit driver's performance at previous work assignments. The process shall carry out this activity by 1) utilizing standardized performance evaluation criteria set forth by governmental regulations and transit operating company policies, 2) assessing the transit driver's driving history, and 3) assessing comments from the transit driver's supervisor(s). It shall also use the details of any moving violations or accidents, supervisor comments, government regulations, and company policies. The data shall be sent to this process by the process that provides the interface to a local data store, each time that the store is updated with driver performance data.

4.5.2 Assess Transit Driver Availability

Overview: This process shall assess the transit driver's availability based on previous work assignments plus health and vacation commitments. The process shall carry out this activity by 1) utilizing standardized transit driver work criteria set forth by governmental regulations and company policies, 2) monitoring the transit driver's health status and vacation status, and 3) monitoring the transit driver's accumulated work hours. The data shall be sent to this process by the process that provides the interface to a local data store, each time that the store is updated with driver availability data.

4.5.3 Access Transit Driver Cost Effectiveness

Overview: This process shall assess the transit driver's cost effectiveness when carrying out previous work assignments. The process shall perform this activity by 1) utilizing standard transit driver cost criteria set forth by governmental regulations and company policies, and 2) monitoring the transit driver's hourly wage and accumulated work hours. The data shall be sent to this process by the process that provides the interface to a local data store, each time that the store is updated with driver cost effectiveness data.

4.5.4 Assess Transit Driver Eligibility

Overview: This process shall assess the transit driver's eligibility for future work assignments. The process shall carry out this activity by 1) monitoring the transit driver's performance, availability and cost effectiveness, 2) utilizing standardized transit driver eligibility criteria set forth by governmental regulations and company policies, and 3) ensuring that the transit driver has the required experience, education and certifications. The data shall be sent to this process in one of two ways: 1) by the process that provides the interface to a local data store, each time that the store is updated with driver eligibility data, or 2) the data is produced as the result of analysis work carried out by other processes in the Manage Traffic function.

4.5.5 Generate Transit Driver Route Assignments

Overview: This process shall assign transit drivers to transit schedules. The transit driver's eligibility, route preferences, seniority, and transit vehicle availability shall be used by the process to determine the transit driver's route assignment. The output produced by the process shall be sent to the transit driver in the form of the next work assignment. The input and output forms shall include those that are suitable for travelers with physical disabilities.

4.5.6 Update Transit Driver Information

Overview: This process shall provide the interface through which the transit driver can input data to the store of transit driver information. The interface provided by this process shall enable the transit driver to update personal availability and route assignment information. The input and output forms shall include those that are suitable for travelers with physical disabilities.

4.5.7 Report Transit Driver Information

Overview: This process shall provide the interface between the transit fleet manager and the store of driver information. The interface provided by the process shall enable the fleet manager to review and update transit driver information. The input and output forms shall include those that are suitable for travelers with physical disabilities.

4.5.8 Provide Transit Driver Information Store Interface

Overview: This process shall provide the read and write interface to the store of transit driver information. The interface enables the contents of the store to be updated with inputs received from the transit driver and transit fleet manager via other processes, as well as, inputs resulting from analysis of driver availability, cost effectiveness, eligibility, and performance carried out by other processes. The process shall also supply data to these processes when the store is updated with information from the transit driver and fleet manager. It shall also supply data to the process that generates driver route assignments when any of the analysis inputs is received.

2.20.3 Subsystem Interfaces for TRMS

Emergency Management => **Transit Management**

Physical Architecture Flow Name: transit emergency coordination data

Data exchanged between centers dealing with a transit-related incident.

Logical Architecture Reference Flow: transit_incident_coordination_data
This data flow is sent from the Manage Emergency Services function to the Manage Transit function and contains information needed to deal with a transit related incident. It contains the following data items each of which is defined in its own DDE: transit_coordination_information + transit_fleet_operation_request + transit_response_to_incident.

Financial Institution => **Transit Management**

Physical Architecture Flow Name: transaction status

Response to transaction request. Normally dealing with a request for payment.

Logical Architecture Reference Flow: ffi_bad_fare_payment_updates
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains data about a transit fare transaction that was either attempted but did not work, or for which the subsequent payment transaction by the Financial Institution failed. The data is to be used within the function for checking against future fare

transaction data.

Logical Architecture Reference Flow: ffi_confirm_fare_payment
This data flow is sent from the Financial Institution to the Provide Electronic Payment Services function. It is used to confirm that a previous request for toll payments is being processed by the Financial Institution.

Logical Architecture Reference Flow: ffi_other_services_payment_confirm
This data flow is sent from the financial institution to the Provide Electronic Payment Services function. It contains confirmation that a previously submitted request from a transit user for payment for other (yellow pages) services has been accepted and made.

Information Service Provider => Transit Management

Physical Architecture Flow Name: demand responsive transit request

Request for paratransit support.

Logical Architecture Reference Flow: paratransit_trip_request
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Transit function to action a trip request using the paratransit operation. It contains the following data items each of which is defined in its own DDE: traveler_identity + trip_request.

Physical Architecture Flow Name: selected routes

Routes selected based on route request criteria.

Logical Architecture Reference Flow: advanced_tolls_and_charges_vehicle_confirm
This data flow is used within the Provide Electronic Payment Services function and contains the result of the requested advanced payment transaction from a traveler (as a transit user) in a transit vehicle. It consists of the following data items each of which is defined in its own DDE: advanced_charges_confirm + advanced_tolls_confirm + confirmation_flag.

Logical Architecture Reference Flow: paratransit_service_confirmation
This data flow is sent by the Provide Driver and Traveler Services function to the Manage Transit function to confirm that the traveler wants to use the previously identified paratransit service. It contains the following data items each of which is defined in its own DDE: paratransit_service_identity + transit_confirmation_flag + traveler_identity.

Physical Architecture Flow Name: transit information request

Request for transit operations information including schedule and fare information. The request can be a subscription that initiates as-needed information updates as well as a one-time request for information.

Logical Architecture Reference Flow: advanced_other_fares_request
This data flow is used within the Provide Electronic Payment Services function to request that a transit fare be paid in advance by a driver who is paying either a parking lot charge or a toll. It consists of the following data items each of which is defined in its own DDE: credit_identity + stored_credit + transit_route_origin + transit_route_destination + transit_journey_date + traveler_identity.

Logical Architecture Reference Flow: advanced_traveler_fares_request
This data flow is used within the Provide Electronic Payment Services function to request that a transit fare be paid for in advance by a traveler who is planning a trip using facilities in the Provide Driver and Traveler Services function. It consists of the following items each of which is defined in its own DDE: credit_identity + stored_credit + transit_route_origin + transit_route_destination + transit_journey_date + traveler_identity.

Logical Architecture Reference Flow: transit_fare_data_request

This data flow is used within the Provide Electronic Payment Services function. It contains a request for the current transit fare price data to be provided from the store that is being used to calculate transit fares.

Logical Architecture Reference Flow: transit_services_advisories_request
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Transit function. It is a request for supply of details of the services currently being provided by the transit fleet and will be used in the preparation of on-line driver and traveler advisory data for output to vehicles.

Logical Architecture Reference Flow: transit_services_guidance_request
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Transit function. It is a request for supply of details of the services being currently provided by the transit fleet and will be used in the preparation of on-line traveler guidance data. The process(es) that are providing the interface through which the traveler is obtaining the on-line guidance will have to provide the origin and destination so that the receiving process in the Manage Transit function can work out for which transit route(s) data will be provided. The data flow consists of the following data items each of which is defined in its own DDE: destination + origin + traveler_identity.

Logical Architecture Reference Flow: transit_vehicle_deviations_details_request
This data flow is used within the Manage Transit function. It contains a request for output of the details of the deviations of transit vehicles from their published routes and schedules for use as a source of data to be sent to processes in other functions.

Intermodal Transportation Service Provider => **Transit Management**

Physical Architecture Flow Name: intermodal information

Schedule information for alternate mode transportation providers such as train, ferry, air and bus.

Logical Architecture Reference Flow: fitsp_transit_service_data
This data flow is sent from the intermodal transportation service provider to the Manage Transit function and contains details of the services provided by the intermodal transportation service providers. The data is intended for use by other processes in the calculation of new routes and services for the local regular transit operation and will enable the two groups of services to be coordinated for the benefit of the traveler.

Map Update Provider => **Transit Management**

Physical Architecture Flow Name: map updates

Map update which could include a new underlying static or real-time map or map layer(s) update.

Logical Architecture Reference Flow: fmup_transit_map_update
This data flow is sent from the map update provider to the Manage Transit function and contains digitized map data to be used for route generation and as a background to displays of services requested by the transit fleet manager.

Media => **Transit Management**

Physical Architecture Flow Name: media information request

Request from the media for current transportation information.

Logical Architecture Reference Flow: fm_transit_incident_information_request
This data flow contains a request for data on incidents to be sent to the Media. The request must specify whether all, current or predicted incidents are required, in the latter case state the time period by date and hour range, and the

geographic area(s) to which it should relate.

Logical Architecture Reference Flow: fm_transit_schedule_deviations_request
This data flow contains a request for data on details of deviations from schedule of regular transit services.

Other TRM => **Transit Management**

Physical Architecture Flow Name: TRMS coord

Coordination information between local/regional transit organizations including schedule, on-time information and ridership.

Logical Architecture Reference Flow: fotrm_transit_services
This data flow is sent by the other TRM (other transit center) to the Manage Transit function and contains data from the other transit center about services which have an interface into the area covered by services from the local center.

Parking Management => **Transit Management**

Physical Architecture Flow Name: transit parking coordination

Request for coordinated fare payment and parking lot price data.

Logical Architecture Reference Flow: parking_lot_transit_request
This data flow is sent by the Manage Traffic function to the Manage Transit function and contains a request for new or additional park and ride (P+R) transit services to be provided from the parking lot.

Personal Information Access => **Transit Management**

Physical Architecture Flow Name: transit information user request

Request for special transit routing, real-time schedule information, and availability information.

Logical Architecture Reference Flow: transit_services_personal_request
This data flow is a request for supply of details of transit services for output to a traveler's personal device. The traveler will have to provide the origin and destination so that the receiving process can work out for which transit route(s) data will be provided. The data flow consists of the following data items each of which is defined in its own DDE: destination + origin + traveler_identity.

Remote Traveler Support => **Transit Management**

Physical Architecture Flow Name: emergency notification

An emergency request for assistance originated by a traveler using an in-vehicle, public access, or personal device. Sufficient information is provided so that the recipient can determine the location of the emergency as a minimum. Additional information identifying the requestor and requesting device and the nature and severity of the emergency may also be provided (and required) by some systems.

Logical Architecture Reference Flow: emergency_request_transit_details
This data flow is sent by the Provide Driver and Traveler Services function to the Manage Transit function to send data about an emergency declared by a traveler at a transit stop using a kiosk or other device to the Manage Emergency Services function. This can also be used by the transit user to alert the transit system operator to an emergency situation or incident within the transit operational network, i.e. not on-board a transit vehicle, or at a transit stop, but in

such things as a modal interchange facility, transit depot, etc. It contains the following data items each of which is defined in its own DDE: date + time + transit_emergency_request.

Logical Architecture Reference Flow: transit_user_roadside_image
This data flow is sent from the Manage Transit function to the Provide Electronic Payment Services function. It contains an JPEG compressed image of the transit user who has violated the transit fare collection process at the roadside, i.e. at a transit stop. The data will be used in subsequent transit fare violation processing.

Physical Architecture Flow Name: secure area surveillance data

Data collected from surveillance systems used to monitor secure areas. Includes video, audio, and other security sensor outputs.

Logical Architecture Reference Flow: secure_area_surveillance_information
This data flow is sent from the Provide Driver and Traveler Services Function to the Manage Transit function and represents information about conditions in a secure area environment such as that found in a transit network. This information is sensed/detected by sensors contained in the Manage Transit function, and includes video, audio, and other image data. The data may have been obtained from closed circuit television (cctv), or other systems that are monitoring activity in the transit operational network, i.e. not on-board a transit vehicle, but at a transit stop, or in such things as a modal interchange facility, transit depot, etc. The data can be used for incident detection, etc., using automatic analysis techniques.

Physical Architecture Flow Name: transit fare payment requests

Information provided from the transit user location that supports fare payments and associated record keeping.

Logical Architecture Reference Flow: fare_collection_roadside_violation_information
This data is used by the Manage Transit function to send data about a violator of the transit fare collection processes at the roadside, i.e. a transit stop to the Manage Emergency Services function. This data flow will contain a digitized video image of the transit user who is trying to violate the fare collection process at the roadside. It is assumed that this digitized data will include other data such as date and time, plus camera identity from which the roadside (transit stop) location can be determined. The data flow consists of the following data items each of which is defined in its own DDE: transit_route_number + transit_route_segment_number + transit_user_roadside_image + transit_user_roadside_tag_identity.

Logical Architecture Reference Flow: request_roadside_fare_payment
This data flow is sent from the Manage Transit function to the Provide Electronic Payment Services function to request payment of a transit fare from the roadside, i.e. a transit stop. It consists of the following data items each of which is defined in its own DDE: transit_fare + transit_roadside_fare_collection_identity + transit_user_roadside_tag_identity.

Logical Architecture Reference Flow: transit_roadside_fare_payment_confirmation
This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the previous request for the cost of the current transit fare has been deducted successfully from the credit currently stored by the transit user's payment instrument. The data flow is used when the transit user is paying for the transit fare at the roadside and consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: transit_roadside_passenger_data
This data flow is used within the Manage Transit function. It contains the number of transit users (passengers) who in a twenty four hour period, have passed through a transit stop plus data about the ride which they purchased. The data is derived from roadside fare collection data and is for use in the determination of future transit services. The data flow consists of the following data items each of which is defined in its own DDE: transit_route_stop_number + transit_user_journey_start + 24{list_size + list_size{transit_passenger_numbers + transit_user_journey_end + transit_route_use_time + transit_user_category}}.

Physical Architecture Flow Name: transit information user request

Request for special transit routing, real-time schedule information, and availability information.

Logical Architecture Reference Flow: other_services_roadside_request
This data flow is sent from the Manage Transit function to the Provide Electronic Payment Services function and contains the transit user's request from the roadside, i.e. a transit stop, for other (yellow pages) services. It consists of the following data items each of which is defined in its own DDE : traveler_identity + credit_identity + other_services_data.

Logical Architecture Reference Flow: transit_services_kiosk_request
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Transit function. It is a request for details of transit services for output to a kiosk. The traveler will have to provide the origin and destination so that the receiving process can work out for which transit route(s) data will have to be provided. The data flow consists of the following data items each of which is defined in its own DDE: destination + kiosk_identity + origin.

Logical Architecture Reference Flow: transit_services_travelers_request
This data flow is used within the Manage Transit function to request the details of the current transit services for a transit user at the roadside. The transit user will have to provide the origin and destination so that the receiving process can work out for which transit route(s) data will be provided. The data flow consists of the following data items each of which is defined in its own DDE: destination + origin + traveler_identity.

Traffic Management => **Transit Management**

Physical Architecture Flow Name: demand management request

Request to change the demand for road facility use through pricing or other mechanisms.

Logical Architecture Reference Flow: transit_services_changes_request
This data flow is sent by the Manage Traffic function to the Manage Transit function and is a request to change the current transit services in response to changes in demand, or a desire to change the modal split currently being used by travelers.

Physical Architecture Flow Name: request transit information

Request for transit service information and current transit status.

Logical Architecture Reference Flow: transit_conditions_demand_request
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Traffic function. It is used to request details of the current state of transit vehicle operations for use in demand forecasting calculations carried out by the Manage Demand facility.

Logical Architecture Reference Flow: transit_fare_direct_request
This data flow contains a request for the current prices being charged for transit fares.

Logical Architecture Reference Flow: transit_services_demand_request
This data flow is sent from the Manage Traffic function to the Manage Transit function. It is a request for supply of details of the transit services and will be used in the preparation of demand forecasts by the Manage Demand facility. The data flow consists of the following data items each of which is defined in its own DDE: tmc_identity + transit_services_request.

Physical Architecture Flow Name: traffic control priority status

Status of signal priority request functions at the roadside (e.g. enabled or disabled).

Logical Architecture Reference Flow: transit_highway_priority_given
This data flow is sent from the Manage Traffic function to the Manage Transit function. It contains confirmation that the requested priority has been given to transit vehicles throughout the freeway network served by the function. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: transit_ramp_priority_given
This data flow is sent from the Manage Transit function to the Manage Traffic function. It contains confirmation that the overall priority request for one or more transit vehicles over the ramp signals in a wide area as opposed to priority requests from individual vehicles at a particular set of ramp signals has been given. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: transit_road_priority_given
This data flow is sent from the Manage Traffic function to the Manage Transit function. It contains confirmation that the requested priority has been given to transit vehicles throughout the road network served by the function. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Physical Architecture Flow Name: traffic information for transit

Current and predicted traffic information and incident information.

Logical Architecture Reference Flow: prediction_data
This data flow is used within the Manage Traffic function and is also sent by that function to the Manage Transit and Provide Driver and Traveler Services function. It contains output from the predictive model process showing predictions of traffic data for route segments on the road and highway network served by the Manage Traffic function. The data flow consists of the following items each of which is defined in its own DDE: list_size + list_size{route_segment_identity + route_segment_volume_delay_predictions + route_segment_queue_delay_predictions + route_segment_speed_predictions + route_segment_occupancy_predictions}.

Logical Architecture Reference Flow: traffic_data_for_transit
This data flow contains information about environmental conditions, current incidents on the road, traffic flow state, and air quality data. The data flow consists of the following data items each of which is defined in its own DDE: list_size + roadway_environment_conditions + traffic_flow_state + link_state_data + current_incidents_data + area_air_quality_index.

Transit Driver => **Transit Management**

Physical Architecture Flow Name: transit driver availability

Transit driver availability data that can be used to develop driver assignments and detailed operations schedules.

Logical Architecture Reference Flow: ftd_information_updates
This data flow is sent from the transit driver to the Manage Transit function. It contains information from individual transit drivers about their previous work hours and vacations, etc. It is used to assess a driver's eligibility for future work assignments.

Transit Fleet Manager => **Transit Management**

Physical Architecture Flow Name: transit fleet manager inputs

Instructions governing service availability, schedules, emergency response plans, transit personnel assignments, transit maintenance requirements, and other inputs that establish general system operating requirements and procedures.

Logical Architecture Reference Flow: ftfm_approved_corrections
This data flow is sent from the transit fleet manager to the Manage Transit function and contains the transit fleet manager's approval for changes to transit schedules that will bring one or more transit vehicles back to their schedules.

Logical Architecture Reference Flow: ftfm_coordination_data
This data flow is sent from the transit fleet managert to the Manage transit function and contains an acknowledgement of the previous request for action plus incident coordination and response information.

Logical Architecture Reference Flow: ftfm_initiate_service_updates
This data flow is sent from the transit fleet manager to the Manage Transit function and indicates that the transit fleet manager wants the transit services to be re-processed because changes have occurred to the parameters which govern their calculation.

Logical Architecture Reference Flow: ftfm_passenger_loading_updates
This data flow is sent from the transit fleet manager to the Manage Transit function and contains an update to the passenger (transit user) loading data as a result of a miss-match being found in the collected data.

Logical Architecture Reference Flow: ftfm_planning_parameters
This data flow is sent from the transit fleet manager to the Manage Transit function and contains planning parameters used to calculate new transit schedules.

Logical Architecture Reference Flow: ftfm_planning_parameters_update_request
This data flow is sent from the transit fleet manager to the Manage Transit function and contains a request for the current planning transit service planning parameters to be output to the transit fleet manager.

Logical Architecture Reference Flow: ftfm_request_response_parameter_output
This data flow is sent from the transit fleet manager to the Manage Transit function and is used to output of the preplanned responses to incidents by and within the transit operations area.

Logical Architecture Reference Flow: ftfm_request_transit_vehicle_data
This data flow is sent from the transit fleet manager to the Manage Transit function and contains a request for output of the current contents of the store of transit vehicle operational data.

Logical Architecture Reference Flow: ftfm_response_parameters
This data flow is sent from the transit fleet manager to the Manage Transit function and is used to provide data about planned responses to incidents by and within the transit operations area.

Logical Architecture Reference Flow: ftfm_technician_information_request
This data flow is sent from the transit fleet manager to the Manage Transit function and is used to request information about transit maintenance technicians.

Logical Architecture Reference Flow: ftfm_technician_information_updates
This data flow is sent from the transit fleet manage and is used to provide updates to the information held within the Manage Transit function about transit technicians.

Logical Architecture Reference Flow: ftfm_transit_display_update_request
This data flow is sent from the transit fleet manager to the Manage Transit function and is used to request an update of the digitized map database. This is used as a source of data for route generation and for the background to displays of transit services data requested by the transit fleet manager.

Logical Architecture Reference Flow: ftfm_transit_driver_information_request
This data flow is sent from the transit fleet manager to the Manage Transit function and contains a request for the output of the consideration data about one or more transit drivers.

Logical Architecture Reference Flow: ftfm_transit_driver_information_updates
This data flow is sent from the transit fleet manager to the Manage Transit function and contains an update of the consideration data for a particular transit driver.

Logical Architecture Reference Flow: ftfm_transit_driver_route_preferences
This data flow is sent from the transit fleet manager to the Manage Transit function and is used to provide information about the preferences that transit drivers have about working over certain transit routes.

Logical Architecture Reference Flow: ftfm_transit_services_output_request
This data flow is sent from the transit fleet manager to the Manage Transit function and is used to request output of the current transit services. This data is that produced by the transit route and schedule generation processes in the function.

Logical Architecture Reference Flow: ftfm_transit_vehicle_maintenance_information_request
This data flow is sent from the transit fleet manager to the Manage transit function and contains a request for information about the maintenance status of an individual transit vehicle.

Logical Architecture Reference Flow: ftfm_transit_vehicle_maintenance_specs
This data flow is sent from the transit fleet manager to the Manage transit function and contains a new or revised transit vehicle maintenance specification. This will be loaded into the store of transit vehicle operations data.

Transit Maintenance Personnel => **Transit Management**

Physical Architecture Flow Name: maintenance status

Current maintenance status of vehicle.

Logical Architecture Reference Flow: ftmp_transit_vehicle_maintenance_updates
This data flow is sent from transit maintenance personnel to the Manage Transit function and is used to provide updates to the maintenance information being held within the function about individual transit vehicles.

Transit Management => **Emergency Management**

Physical Architecture Flow Name: transit emergency data

Initial notification of transit emergency at a transit stop or on transit vehicles and further coordination as additional details become available and the response is coordinated.

Logical Architecture Reference Flow: transit_coordination_data
This data flow is sent from the Manage Transit function to the Manage Emergency Services function. It is used to provide data on the way in which the response to a transit incident should be coordinated. The data flow consists of the following data items each of which is defined in its own DDE: transit_coordination_information + transit_fleet_operation_acknowledge + transit_response_to_incident.

Logical Architecture Reference Flow: transit_emergency_data
This data flow is sent from the Manage Transit function to the Manage Emergency Services function and contains details of an emergency on-board a transit vehicle. It consists of the following data items each of which is defined in its own DDE: incident_location + incident_start_time + incident_duration + incident_severity.

Logical Architecture Reference Flow: transit_incident_details
This data flow is sent from the Manage Transit function to the Manage Emergency Services function and contains details of an incident in the transit operations network. It consists of the following data items each of which is defined in its own DDE: incident_location + incident_start_time + incident_duration + incident_severity.

Transit Management => **Enforcement Agency**

Physical Architecture Flow Name: violation notification

Notification to enforcement agency of violation or regulations.

Logical Architecture Reference Flow: tea_fare_collection_roadside_violation_data
This data flow is sent from the Manage Emergency Services function to the enforcement agency responsible for dealing with transit fare collection violations. It contains information about a collection violation that has been detected by the Manage Transit function at the roadside, i.e. the transit stop. The data in the flow will enable the notified enforcement agency to take the appropriate action against those who have committed the violation.

Logical Architecture Reference Flow: tea_fare_collection_vehicle_violation_data
This data flow is sent from the Manage Emergency Services function to the enforcement agency responsible for dealing with transit fare collection violations. It contains information about a collection violation that has been detected by the Manage Transit function on-board the transit vehicle. The data in the flow will enable the notified enforcement agency to take the appropriate action against those who have committed the violation.

Logical Architecture Reference Flow: tea_fare_payment_violation_data
This data flow is sent from the Manage Emergency Services function to the enforcement agency and contains information about fare payment violations that have been detected by the Manage Transit and/or Provide Electronic Payment Services functions. The data in the flow will enable the notified enforcement agency to take the appropriate action against those committing the violation.

Transit Management => Financial Institution

Physical Architecture Flow Name: payment request

Request for payment from financial institution.

Logical Architecture Reference Flow: tfi_fare_payment_violator_data
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains data about a transit fare payment transaction that was attempted but did not work and is to be used by the Financial Institution.

Logical Architecture Reference Flow: tfi_other_services_payment_request
This data flow is sent to the financial institution from the Provide Electronic Payment Services function. It contains a request from a transit user for payment for other (yellow pages) services.

Logical Architecture Reference Flow: tfi_request_fare_payment
This data flow is sent to the Financial Institution by the Provide Electronic Payment Services function and requests payment of a transit fare. It is sent periodically, e.g. once per day, and requests payment of the transit fare transactions since the previous request. The data flow will include the transit fare cost and credit identity for each transaction.

Transit Management => Information Service Provider

Physical Architecture Flow Name: demand responsive transit plan

Plan regarding overall demand responsive transit schedules and deployment.

Logical Architecture Reference Flow: paratransit_personal_schedule
This data flow is sent from the Manage Transit function to the Provide Driver and Traveler Services function. It consists of the following data items each of which is defined in its own DDE: paratransit_service_details + paratransit_service_cost + traveler_identity.

Physical Architecture Flow Name: transit and fare schedules

Specific transit and fare schedule information including schedule adherence.

Logical Architecture Reference Flow: transit_deviation_data_received

This data flow is used within the Manage Transit function and contains an indication that new data about transit service deviations has been received and is now in the local store of this data. The process(es) receiving this data is(are) expected to take action automatically to output the new data to other functions that are outside the scope of the ITS.

Logical Architecture Reference Flow: transit_fare_data

This data flow is used within the Provide Electronic Payment Services function and contains details of the fares being currently charged for transit services. It consists of the following item which is defined in its own DDE: list_size + list_size{transit_fares}.

Logical Architecture Reference Flow: transit_services_for_advisory_data

This data flow is sent from the Manage Transit function to the Provide Driver and Traveler Services function. It contains a complete set of all the transit routes and the services that run upon them, including timings, etc. that are provided by the transit fleet from which the data was requested, for use in the preparation of driver and traveler advisory information for output on-board vehicles. It consists of the following data item which is defined in its own DDE: transit_services.

Logical Architecture Reference Flow: transit_services_for_guidance

This data flow is sent from the Manage Transit function to the Provide Driver and Traveler Services function. It contains a complete set of all the transit routes and the services that run upon them, including timings, etc. that are provided by the transit fleet from which the data was requested, for use in the preparation of data for output as on-line driver and traveler guidance data. The data flow consists of the following data items each of which is defined in its own DDE: 1{transit_services_for_output}2 + traveler_identity.

Logical Architecture Reference Flow: transit_vehicle_deviations_details

This data flow is used within the Manage Transit function. It contains details of the deviations of transit vehicles from their published routes and schedules and is used as a source of data to be sent to processes in other functions. The data flow consists of the following data items each of which is defined in is own DDE: transit_vehicle_eta + transit_vehicle_collected_trip_data + transit_vehicle_deviation_update + transit_vehicle_location + transit_vehicle_schedule_deviations.

Physical Architecture Flow Name: transit incident information

Information on transit incidents that impact transit services for public dissemination.

Logical Architecture Reference Flow: transit_incident_data

This data flow contains information about an incident that has occurred within part of the transit operations network, e.g. transit stop or mode interchange point. The location and details of the incident will be included in the information, subject to any constraints applied by the transit agency on providing information to outside sources. The data flow has been sized to enable two pages of text at 65 lines/page and 80 characters/line to be accommodated.

Physical Architecture Flow Name: transit request confirmation

Confirmation of a request for transit information or service.

Logical Architecture Reference Flow: advanced_other_fares_confirm

This data flow is used within the Provide Electronic Payment Services function to confirm the advanced payment of a transit fare by a transit user. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + stored_credit + transit_fare + traveler_identity.

Logical Architecture Reference Flow: advanced_tolls_and_charges_vehicle_request

This data flow is used by the Manage Transit function to transfer requests for advanced payments for toll and parking lot charges from the traveler (as a transit user) fare payment interface in a transit vehicle to the Provide Electronic Payment Services function for subsequent processing. The size of the data flow has been set at less than the sum of the two constituent flows to allow for the fact that they will both not be present for every data transfer. It consists of the following data items each of which is defined in its own DDE: advanced_charges + advanced_tolls.

Logical Architecture Reference Flow: advanced_traveler_fares_confirm
This data flow is used within the Provide Electronic Payment Services function to show whether advanced fare payment by a traveler planning a trip has been refused or cleared. The traveler will be using facilities in the Provide Driver and Traveler Services function to generate the trip request. The data flow consists of the following data items each of which is defined in its own DDE: confirmation_flag + stored_credit + transit_fare + traveler_identity.

Transit Management => **Intermodal Transportation Service Provider**

Physical Architecture Flow Name: intermodal information

Schedule information for alternate mode transportation providers such as train, ferry, air and bus.

Logical Architecture Reference Flow: titsp_transit_arrival_changes
This data flow is sent to the intermodal transportation service provider from the Manage Transit function and contains details of the changes that are currently expected to the arrival time of a transit vehicle at the next modal interchange point. The data is intended to enable the coordination of services between the intermodal and regular transit operations.

Logical Architecture Reference Flow: titsp_transit_arrival_deviations
This data flow is sent to the intermodal transportation service provider from the Manage Transit function and contains details of the changes that are currently expected to the arrival time of transit vehicles at the modal interchange point(s). The data is intended to enable the coordination of services between the intermodal and regular transit operations.

Logical Architecture Reference Flow: titsp_transit_service_data
This data flow is sent to the intermodal transportation service provider from the Manage Transit function and contains details of the regular transit services provided by the local transit operation. The data is intended for use in the coordination of services between the intermodal and regular transit operations.

Transit Management => **Map Update Provider**

Physical Architecture Flow Name: map update request

Request for a map update which could include a new underlying map or map layer updates.

Logical Architecture Reference Flow: tmup_transit_map_update_request
This data flow is sent to the map update provider from the Manage Transit function and contains a request for an update of the map database used for generating new transit routes and as a background to displays of transit services requested by the transit fleet manager.

Transit Management => **Media**

Physical Architecture Flow Name: transit incidents for media

Report of an incident impacting transit operations for public dissemination through the media.

Logical Architecture Reference Flow: tm_transit_emergency_information
This data flow is sent to the media from the Manage Transit function and provides information to the media that an emergency has occurred within a transit vehicle. The location of the transit vehicle and details of the emergency will be included in the information, subject to any constraints applied by the transit media information parameters. These parameters will also control the style and format of the way in which the information is presented to the media. The

data flow has been sized to enable two pages of text at 65 lines/page and 80 characters/line to be accommodated.

Logical Architecture Reference Flow: tm_transit_incident_information
This data flow is sent to the media from the Manage Transit function and contains information about an incident that has occurred within part of the transit operations network, e.g. transit stop or mode interchange point. The location and details of the incident will be included in the information, subject to any constraints applied by the transit media information parameters. These parameters will also control the style and format of the way in which the information is presented to the media. The data flow has been sized to enable two pages of text at 65 lines/page and 80 characters/line to be accommodated.

Physical Architecture Flow Name: transit information for media

Report of transit schedule deviations for public dissemination through the media.

Logical Architecture Reference Flow: tm_transit_schedule_deviations_to_media
This data flow contains details of deviations from schedule of regular transit services. The information will enable the media to broadcast the details to travelers via such things as local radio, bulletin boards, etc.

Transit Management => Other TRM

Physical Architecture Flow Name: TRMS coord

Coordination information between local/regional transit organizations including schedule, on-time information and ridership.

Logical Architecture Reference Flow: totrm_transit_services
This data flow is sent to the other TRM (other transit center) by the Manage Transit function and contains data for the other transit center about services provided by the local center which have an interface into the area(s) covered by services from the other center.

Transit Management => Parking Management

Physical Architecture Flow Name: transit parking lot response

Response to transit occupancy inquiries and coordination with parking lots.

Logical Architecture Reference Flow: parking_lot_transit_response
This data flow is sent by the Manage Transit function to the Manage Traffic function and contains the response to a request for new or additional park and ride (P+R) transit services to be provided from the parking lot.

Transit Management => Personal Information Access

Physical Architecture Flow Name: personal transit information

General and personalized transit information for a particular fixed route, flexible route, or paratransit system.

Logical Architecture Reference Flow: transit_services_for_personal_devices
This data flow contains details of the current transit services for output to a traveler's personal device and consists of the following data items each of which is defined in its own DDE: traveler_identity + 1{transit_services_for_output}2.

Transit Management => Planning Subsystem

Physical Architecture Flow Name: operational data

Statistical data used for planning purposes.

Logical Architecture Reference Flow: transit_passenger_operational_data
This data flow is sent from the Manage Transit function to the Plan System Deployment function. It contains information about the number of passengers (transit users) who have used transit stops and vehicles being operated by the Manage transit function. There are therefore two sets of data, one showing the numbers of passengers using each transit stop and the other showing the number of passengers on-board transit vehicles on each route segment. The data is recorded as the average values over short time intervals, e.g. five (5) minutes, for a whole day. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{date + time + transit_roadside_operational_data + transit_route_operational_data}.

Logical Architecture Reference Flow: transit_services_for_deployment
This data flow is sent from the Manage Transit function to the Plan System Deployment function. It contains details of the current transit services for use in the analysis of ITS operating performance and consists of the following data item which is defined in its own DDE: transit_services.

Logical Architecture Reference Flow: transit_user_payments_transactions
This data flow is used within the Provide Electronic Payment Services function and contains records of all payment transactions for the provision of other (yellow pages) services to transit users.

Transit Management => **Remote Traveler Support**

Physical Architecture Flow Name: emergency acknowledge

Acknowledge request for emergency assistance and provide additional details regarding actions and verification requirements.

Logical Architecture Reference Flow: emergency_acknowledge_transit_details
This data flow is sent by the Manage Transit function to the Provide Driver and Traveler Services function to confirm that the request for emergency services previously sent by the traveler has been received from a kiosk or other device. This data flow may also contain the response to input from a panic button that has been activated by a transit user in part of the transit operational network, i.e. not on-board a transit vehicle, or at a transit stop, but in such things as a modal interchange facility, transit depot, etc. The information will be sent out as part of the response to an emergency or incident being detected within the network. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: request_transit_user_roadside_image
This data flow is sent from the Provide Electronic Payment Services function to the Manage Transit function. It contains a request for the supply of the image of a transit user who has violated the transit fare payment process at a roadside fare collection point.

Physical Architecture Flow Name: secure area monitoring support

Commands that control surveillance equipment and security sensors that monitor secure public transportation areas. Also includes information for general advisories and alerts intended for general dissemination in these same public areas.

Logical Architecture Reference Flow: secure_area_broadcast_message
This data flow is sent to the Provide Driver and Traveler Services function by the Manage Transit function and contains textual information for transit users in part of the transit operational network, i.e. not on-board a transit vehicle, or at a transit stop, but in such things as a modal interchange facility, transit depot, etc. The information will be sent out as part of the response to an emergency or incident being detected within the network.

Logical Architecture Reference Flow: secure_area_monitoring_control
This data flow is sent to the Provide Driver and Traveler Services function by the Manage Transit function and contains control data for closed circuit television (cctv) systems, or audio equipment, located in the secure area environment. This data may change the pan, tilt, zoom, or other camera or audio operating parameters and may be generated automatically or as a result of input from the transit system operator.

Physical Architecture Flow Name: transit fare payment responses

Information provided by transit management that supports a fare payment transaction

Logical Architecture Reference Flow: confirm_roadside_fare_payment
This data flow is sent from the Provide Electronic Payment Services function to the Manage transit function to confirm that transaction processing of the payment of a transit fare from the roadside, i.e. a transit stop, has been completed. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + transit_roadside_fare_collection_identity.

Logical Architecture Reference Flow: transit_roadside_fare_data
This data flow is sent by the Provide Electronic Payment Services function to the Manage Transit function and contains details of the fares being currently charged for regular transit services. It is for use in calculating fares that are to be paid by transit users at the roadside, i.e. a transit stop, and consists of the following data item which is defined in its own DDE: transit_fare_data.

Logical Architecture Reference Flow: transit_roadside_fare_payment_debited
This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the cost of the current transit fare will be deducted by the financial institution from the credit identity previously provided by the payment instrument being used by the transit user on-board a transit vehicle. It is only sent when a credit identity has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: transit_roadside_fare_payment_request
This data flow is used within the Provide Electronic Payment Services function and contains the request for the cost of the current transit fare to be deducted from the credit currently stored by the transit user's payment instrument, when it is being used at the roadside, i.e. a transit stop. It is only sent when a value of stored credit has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: transit_fare.

Logical Architecture Reference Flow: transit_services_for_roadside_fares
This data flow is used within the Manage Transit function. It contains details of the transit user fares for all the transit routes operated by the transit fleet from which the request was made. This data is for use in processing transit fare payments initiated by transit users at the roadside (a transit stop). The data flow consists of the following data item which is defined in its own DDE: transit_route_fare_data.

Physical Architecture Flow Name: transit traveler information

Transit information prepared to support transit users and other travelers. It contains transit schedules, real-time arrival information, fare schedules, and general transit service information.

Logical Architecture Reference Flow: other_services_roadside_response
This data flow is sent from the Provide Electronic Payment Services function to the Manage Transit function and contains the response to the transit user's request from the roadside, i.e. a transit stop, for other (yellow pages) services. It consists of the following data items each of which is defined in its own DDE : traveler_identity + credit_identity + other_services_data.

Logical Architecture Reference Flow: transit_services_for_kiosks
This data flow is sent from the Manage Transit function to the Provide Driver and Traveler Services function. It contains details of the transit services that satisfy a traveler's request and are for output to a kiosk. The data flow

consists of the following data items each of which is defined in its own DDE: kiosk_identity + 1{transit_services_for_output}2.

Logical Architecture Reference Flow: transit_services_for_travelers
This data flow is used within the Manage Transit function. It is sent to the Provide Traveler Transit Interface facility and contains a complete set of all the transit routes and the services that run upon them, including timings, etc. that are provided by the transit fleet from which the data was requested. The data flow consists of the following data items each of which is defined in its own DDE: traveler_identity + 1{transit_services_for_output}2.

Logical Architecture Reference Flow: transit_vehicle_arrival_time
This data flow is used within the Manage Transit function. It contains the estimated time of arrival of a transit vehicle at a stop plus the route and service number on which it is operating.

Logical Architecture Reference Flow: transit_vehicle_user_data
This data flow is used within the Manage Transit function and contains data about a transit vehicle for automatic output to transit users at transit stops. The data is output at the transit stop as the vehicle approaches and contains information about the vehicle such as the route number. It therefore consists of the following data items each of which is defined in its own DDE: transit_route_number + transit_vehicle_time.

Transit Management => Traffic Management

Physical Architecture Flow Name: demand management response

Response to various demand management change requests indicating level of compliance with request.

Logical Architecture Reference Flow: transit_services_changes_response
This data flow is sent by the Manage Transit function to the Manage Traffic function and is the response to the previous request for changes in the transit services.

Physical Architecture Flow Name: traffic control priority request

Request for signal priority at one or more intersections along a particular route.

Logical Architecture Reference Flow: transit_highway_overall_priority
This data flow is sent from the Manage Transit function to the Manage Traffic function. It contains requests and information about the overall priority which should be given to one or more transit vehicles at all points in the freeway network served by the function, as opposed to priority requests from individual vehicles at specific locations. This priority will apply at an individual junction, or along a selected transit route if that is specified. The data flow size assumption below is based on an assumed percentage of transit vehicles running late (TRANSIT_VEH_DEVS) and the percentage of freeway miles to total miles (HIGHWAY_MILES/MILES).

Logical Architecture Reference Flow: transit_ramp_overall_priority
This data flow is sent from the Manage Transit function to the Manage Traffic function. It contains requests and information on the overall priority which should be given to one or more transit vehicles over a wide area as opposed to priority requests from individual vehicles at a particular set of ramp signals.

Logical Architecture Reference Flow: transit_road_overall_priority
This data flow is sent from the Manage Transit function to the Manage Traffic function. It contains requests and information about the overall priority which should be given to one or more transit vehicles at all junctions and/or pedestrian crossings in the road network served by the function, as opposed to priority requests for individual vehicles at specific locations. As this is a 'blanket' application of priority, no list of indicators is needed. The size assumption is based on an assumed percentage of transit vehicles running late (TRANSIT_VEH_DEVS) and the percentage of street miles to total miles (ROAD_MILES/MILES).

Physical Architecture Flow Name: transit system data

Current transit system operations information indicating current transit routes, the level of service on each route, and the progress of individual vehicles along their routes for use in forecasting demand and estimating current transportation network performance.

Logical Architecture Reference Flow: transit_fare_direct_details
This data flow contains details of the fares being currently charged for transit services. It consists of the following data items each of which is defined in its own DDE: list_size + list_size{transit_route_number + transit_route_segment_list + transit_user_category + transit_route_use_time}.

Logical Architecture Reference Flow: transit_probe_data
This data flow contains the location of the transit vehicle on each part of its route, i.e., each transit route segment. This data will be used along with other probe data to calculate the link speed or travel time. The data flow consists of the following items each of which is defined in its own DDE: list_size + transit_route_number + transit_route_segment_number + transit_vehicle_location_for_store + transit_vehicle_time.

Logical Architecture Reference Flow: transit_running_data_for_demand
This data flow is sent from the Manage Transit function to the Manage Traffic function. It is used to provide data on the current state of transit operation for use in demand forecasting calculations carried out by the Manage Demand facility and consists of the following items each of which is defined in its own DDE: transit_vehicle_passenger_loading + transit_vehicle_deviation_update + transit_vehicle_running_times + transit_vehicle_schedule_deviations + transit_vehicle_eta.

Logical Architecture Reference Flow: transit_services_for_demand
This data flow is sent from the Manage Transit function to the Manage Traffic function. It contains a complete set of all the transit routes and the services that run upon them, including timings, etc. that are provided by the transit fleet from which the data was requested, for use in the calculation of demand forecasts by the Manage Demand facility. The data flow consists of the following data items each of which is defined in its own DDE: transit_services.

Transit Management => **Transit Driver**

Physical Architecture Flow Name: route assignment

Route assignment information for transit driver.

Logical Architecture Reference Flow: ttd_route_assignements
This data flow is sent to the transit vehicle driver by the Manage Transit function. It contains information for the driver about route assignments from the regular driver work assignment facility.

Transit Management => **Transit Fleet Manager**

Physical Architecture Flow Name: transit operations planning data

Accumulated schedule and fare information, emergency response plans, transit personnel information, maintenance records, and other information intended to support overall planning and management of a transit property.

Logical Architecture Reference Flow: ttfm_coordination_request
This data flow is sent to the transit fleet manager by the Manage Transit function. It contains requests for input on the responses to be made to a particular emergency situation or incident.

Logical Architecture Reference Flow: ttfm_parameters
This data flow is sent to the transit fleet manager by the Manage Transit function and contains a list of the currently available planning parameters that can be used to prepare transit schedules. The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Logical Architecture Reference Flow: ttfm_paratransit_service
This data flow is sent to the transit fleet manager by the Manage Transit function. It contains details of paratransit services as and when they are provided to travelers. The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Logical Architecture Reference Flow: ttfm_passenger_loading_error
This data flow is sent to the transit fleet manager by the Manage Transit function and contains details of a miss-match in the passenger (transit user) data collected from the fare collection and data collection processes on-board a transit vehicle. The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Logical Architecture Reference Flow: ttfm_proposed_corrections
This data flow is sent to the transit fleet manager by the Manage Transit function. It contains details of the proposed corrective action to return a transit vehicle to the schedule from which it has deviated. The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Logical Architecture Reference Flow: ttfm_response_parameter_output
This data flow is sent to the transit fleet manager by the Manage Transit function. It contains the output of the preplanned responses to incidents by and within the transit operations area.

Logical Architecture Reference Flow: ttfm_technician_information
This data flow is sent to the transit fleet manager by the Manage Transit function. It contains information about transit maintenance technician work assignments. The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Logical Architecture Reference Flow: ttfm_transaction_reports
This data flow is sent to the transit fleet manager by the Provide Electronic Payments Services function. It contains details of the transit fare payment transactions that have taken place. The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Logical Architecture Reference Flow: ttfm_transit_driver_information
This data flow is sent to the transit fleet manager by the Manage Transit function. It contains information about a transit driver that is being used to generate work assignments etc. The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Logical Architecture Reference Flow: ttfm_transit_services_output
This data flow is sent to the transit fleet manager by the Manage Transit function and contains output of the current transit routes and schedules. The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Logical Architecture Reference Flow: ttfm_transit_vehicle_data
This data flow is sent to the transit fleet manager by the Manage Transit function. It contains data about transit vehicle(s). The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Logical Architecture Reference Flow: ttfm_transit_vehicle_maintenance_information
This data flow is sent to the transit fleet manager by the Manage Transit function. It contains a report on the maintenance state of transit vehicle(s). The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Transit Management => **Transit Maintenance Personnel**

Physical Architecture Flow Name: transit work schedule

Orders for maintenance of transit vehicle or other transit system equipment.

Logical Architecture Reference Flow: ttmp_work_schedule
This data flow is sent to the transit maintenance personnel by the Manage Transit function. It contains the schedule of work to be carried out to maintain the transit vehicle(s). The output may be in audio or visual form, with the latter being available in a variety of formats, e.g. displays, vms, or hardcopy (paper) output.

Transit Management => **Transit System Operators**

Physical Architecture Flow Name: transit operator display

Display for transit operations personnel regarding performance of the transit fleet, current ridership and on-time performance.

Logical Architecture Reference Flow: ttso_emergency_request
This data flow is sent to the transit system operator by the Manage Transit function and contains the emergency requests that have been input from either the transit vehicle driver or the transit user.

Logical Architecture Reference Flow: ttso_media_parameters
This data flow is sent to the transit system operator by the Manage Transit function and contains the current set of parameters used to control the style and content of information about incidents affecting the transit network that is automatically sent to the media.

Logical Architecture Reference Flow: ttso_potential_incidents_alarm
This data flow is sent to the transit system operator by the Manage Transit function and contains information about a potential incident on board a transit vehicle.

Logical Architecture Reference Flow: ttso_potential_security_problem
This data flow is sent to the transit system operator by the Manage Transit function and contains information about a potential security problem that has been detected within the transit system operating facilities, e.g. transit stops, travel interchanges, kiosks, etc.

Logical Architecture Reference Flow: ttso_transaction_reports
This data flow is sent to the transit system operator by the Provide Electronic Payments Services function. It contains details of the transit fare payment transactions that have taken place.

Logical Architecture Reference Flow: ttso_transit_fare_output
This data flow is sent from the transit system operator to the Provide Electronic Payment Services function. It contains the output of the current transit fares held in the local data store.

Logical Architecture Reference Flow: ttso_video_image_data
This data flow is sent to the transit system operator by the Manage Transit function and contains video image data. This will have been received by closed circuit television (cctv) systems located in the secure area environment and may contain images of incidents.

Transit Management => **Transit Vehicle Subsystem**

Physical Architecture Flow Name: bad tag list

List of invalid transit user tags which may have previously failed a fare payment transaction.

Logical Architecture Reference Flow: bad_tag_list_update
This data flow is sent from the Provide Electronic Payment Services function to the Manage transit function. It contains a list of current transit user tags that have been found to be bad. This means that a fare payment transaction in which they were involved has failed, or the tag has been invalidated by the financial institution to which it belongs. The data flow consists of the following data items each of which is defined in its own DDE: transit_vehicle_identity + list_size + list_size{credit_identity}.

Physical Architecture Flow Name: driver instructions

Transit service instructions for both transit and paratransit drivers.

Logical Architecture Reference Flow: approved_corrective_plan
This data flow is used within the Manage Transit function and represents a plan of action to respond to a (relatively) long term transit schedule disruption. It consists of the following data items each of which is defined in its own DDE: transit_route_corrections + transit_changes_in_stops + transit_changes_in_speed.

Logical Architecture Reference Flow: paratransit_transit_driver_instructions
This data flow is used within the Manage Transit function and contains the instructions for the transit driver to follow in order that a paratransit service can be executed. The instructions will contain such things as pick-up points, traveler identities, drop off points, etc. The driver can obtain actual route guidance through the on-line vehicle guidance facility available from the Provide Driver and Traveler Services function.

Logical Architecture Reference Flow: transit_services_for_corrections
This data flow is used within the Manage Transit function. It is sent to the Operate Transit Vehicles facility for use in the calculation of corrections to transit vehicle routes and schedules to restore a service to normal operation. It contains a complete set of all the transit routes and the services that run upon them, including timings, etc. that are provided by the transit fleet from which the data was requested. list_size + list_size{transit_services}.

Logical Architecture Reference Flow: transit_services_for_eta
This data flow is used within the Manage Transit function. It is sent to the Operate Transit Vehicles facility for use in the calculation of transit vehicle estimated times of arrival (eta) at transit stops. It only contains details of the schedule for the transit route that is currently being operated by the vehicle. The data flow consists of the following data items each of which is defined in its own DDE: transit_route_number + transit_route_segment_list + transit_route_stop_list.

Physical Architecture Flow Name: emergency acknowledge

Acknowledge request for emergency assistance and provide additional details regarding actions and verification requirements.

Logical Architecture Reference Flow: request_transit_user_vehicle_image
This data flow is sent from the Provide Electronic Payment Services function to the Manage Transit function. It contains a request for the supply of the image of a transit user who has violated the transit fare payment process at an on-board vehicle fare collection point.

Logical Architecture Reference Flow: transit_operator_request_acknowledge
This data flow is used within the Manage Transit function and contains an acknowledgment that the previous notification of an emergency to the transit system operator has been received and is being considered for action.

Physical Architecture Flow Name: fare management information

Transit fare information and transaction data used to manage transit fare processing on the transit vehicle.

Logical Architecture Reference Flow: confirm_vehicle_fare_payment
This data flow is sent from the Provide Electronic Payment Services function to the Manage transit function to confirm that transaction processing of the payment of a single transit fare (interactive operation) or of a group of fares (batch mode) from on-board the vehicle has been completed. If the transaction processing was in batch mode, the transit user's tag identity will be set to zero (0), otherwise it will be set to the identity provided in the transaction request. The data flow consists of the following data items each of which is defined in its own DDE: confirmation_flag + transit_user_vehicle_tag_identity + transit_vehicle_identity.

Logical Architecture Reference Flow: transit_services_for_vehicle_fares
This data flow is used within the Manage Transit function. It contains details of the transit user fares for all the transit routes operated by the transit fleet from which the request was made. This data is for use in processing transit fare

payments initiated by transit users on-board a transit vehicle. The data flow consists of the following data item which is defined in its own DDE: transit_route_fare_data.

Logical Architecture Reference Flow: transit_vehicle_advanced_payment_response

This data flow is used within the Manage Transit function and contains the result of the requested advanced payment transaction from a traveler (as a transit user) in a transit vehicle. It consists of the following data items each of which is defined in its own DDE: advanced_charges_confirm + advanced_tolls_confirm + confirmation_flag + transit_vehicle_identity.

Logical Architecture Reference Flow: transit_vehicle_fare_data

This data flow is sent by the Provide Electronic Payment Services function to the Manage Transit function and contains details of the fares being currently charged for regular transit services. It is for use in calculating fares that are to be paid by transit users on-board a transit vehicle and consists of the following data item which is defined in its own DDE: transit_fares.

Logical Architecture Reference Flow: transit_vehicle_fare_payment_debited

This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the cost of the current transit fare will be deducted by the financial institution from the credit identity previously provided by the payment instrument being used by the transit user at the roadside. It is only sent when a credit identity has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: transit_vehicle_fare_payment_request

This data flow is used within the Provide Electronic Payment Services function and contains the request for the cost of the current transit fare to be deducted from the credit currently stored by the transit user's payment instrument, when it is being used on-board a transit vehicle. It is only sent when a value of stored credit has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: transit_fare.

Physical Architecture Flow Name: request for vehicle measures

Request for vehicle performance and maintenance data collected by onboard sensors.

Logical Architecture Reference Flow: transit_vehicle_collected_maintenance_data_request

This data flow is used by processes within the Manage Transit function and contains a request for data collected on-board the transit vehicle. The data is produced by sensors analyzing conditions on-board the vehicle during the course of its operation.

Physical Architecture Flow Name: transit schedule information

Current and projected transit schedule adherence.

Logical Architecture Reference Flow: transit_vehicle_advisory_eta

This data flow is used as an interface between the Manage Transit and Provide Driver and Traveler Information functions. It contains the estimated time of arrival of a transit vehicle at the end of a transit route segment, which is usually a stop, plus the route and service number on which it is operating. It is used for individual transit vehicle deviations and contains the following data items each of which is defined in its own DDE: transit_vehicle_identity + transit_vehicle_time + transit_route_number.

Physical Architecture Flow Name: traveler information

Traveler information comprised of traffic status, advisories, incidents, responses to traveler requests (e.g., traveler routing, yellow pages), payment information and many other travel-related data updates and confirmations.

Logical Architecture Reference Flow: other_services_vehicle_response

This data flow is sent from the Provide Electronic Payment Services function to the Manage Transit function and contains the response to the transit user's request from a transit vehicle for other (yellow pages) services. It consists of the following data items each of which is defined in its own DDE : traveler_identity + credit_identity + other_services_data.

Transit System Operators => Transit Management

Physical Architecture Flow Name: transit operator management data

Information and control provided by transit system operators involving many aspects of managing transit operations.

Logical Architecture Reference Flow: ftso_emergency_request_acknowledge
This data flow is sent by the transit operator to the Manage Transit function and is used to acknowledge the receipt of an emergency request previously sent by the Manage Transit function.

Logical Architecture Reference Flow: ftso_fare_updates
This data flow is sent from the transit system operator to the Provide Electronic Payment Services function. It contains data identifying the transit route number, the fares for some or all or the segments on that route and the time(s)/day(s) that they apply, which are to be updated in the store of transit fares.

Logical Architecture Reference Flow: ftso_media_parameter_request
This data flow is sent by the transit system operator to the Manage Transit function and contains a request for output of the current parameters used to control the style and content of information about incidents affecting the transit network that is automatically sent to the media.

Logical Architecture Reference Flow: ftso_media_parameter_updates
This data flow is sent by the transit system operator to the Manage Transit function and contains update to the parameters used to control the style and content of information about incidents affecting the transit network that is automatically sent to the media.

Logical Architecture Reference Flow: ftso_request_fare_output
This data flow is sent from the transit system operator to the Provide Electronic Payment Services function. It contains a request for output of the current transit fares held in the local data store.

Logical Architecture Reference Flow: ftso_security_action
This data flow is sent by the transit system operator to the Manage Transit function and contains details of the action needed in response to a previously identified security problem.

Logical Architecture Reference Flow: ftso_video_camera_action_request
This data flow is sent by the transit system operator to the Manage Transit function and contains a request for the change of operating parameters for a closed circuit television (cctv) system located in the secure area environment. These parameters may cover things such as camera pan, tilt, and zoom, plus other picture controls.

Transit Vehicle Subsystem => Transit Management

Physical Architecture Flow Name: emergency notification

An emergency request for assistance originated by a traveler using an in-vehicle, public access, or personal device. Sufficient information is provided so that the recipient can determine the location of the emergency as a minimum. Additional information identifying the requestor and requesting device and the nature and severity of the emergency may also be provided (and required) by some systems.

Logical Architecture Reference Flow: transit_emergency_details
This data flow is used within the Manage Transit function and contains details of emergency requests that have been

input on-board a transit vehicle. It consists of the following data items each of which is defined in its own DDE: transit_driver_emergency_request + transit_user_emergency_request + transit_vehicle_location.

Logical Architecture Reference Flow: transit_emergency_information
This data flow is used within the Manage Transit function and contains details of emergency requests that have been input on-board a transit vehicle. It consists of the following data items each of which is defined in its own DDE: transit_driver_emergency_request + transit_user_emergency_request + transit_vehicle_location.

Logical Architecture Reference Flow: transit_operator_emergency_request
This data flow is used within the Manage Transit function and contains information about an incident that has been detected on board a transit vehicle or at a transit facility following input from a transit user or transit vehicle driver. The data is for output to the transit system operator so that responsive action can be initiated.

Physical Architecture Flow Name: fare and payment status

Current fare collection information including the operational status of the fare collection equipment and financial payment transaction data.

Logical Architecture Reference Flow: fare_collection_vehicle_violation_information
This data is used by the Manage Transit function to send data about a violator of the transit fare collection processes on-board the vehicle to the Manage Emergency Services function. This data flow will contain a digitized video image of the transit user who is trying to violate the fare collection process on-board a vehicle. It is assumed that this digitized data will include other data such as date and time, plus camera identity from which the transit vehicle identity can be determined. The data flow consists of the following data items each of which is defined in its own DDE: transit_route_number + transit_route_segment_number + transit_user_vehicle_image + transit_user_vehicle_tag_identity.

Logical Architecture Reference Flow: request_vehicle_fare_payment
This data flow is sent from the Manage Transit fare billing on vehicle facility to the Provide Electronic Payment Services function. It requests payment processing of one or more transit fare transactions from on-board a transit vehicle. This flow provides for both batch (low value/high usage) fare transactions (e.g. city bus routes) and for high value/low volume, interactive, near real-time transactions (e.g. individualized flexible transit). The size assumption below is appropriate for the interactive mode of operation (one transaction per message) which can be assumed to present the greater stress on the transit vehicle communications process due to the higher frequency of operation. The data flow consists of the following data items each of which is defined in its own DDE: transit_route_number + transit_vehicle_identity + transit_vehicle_fare_collection_method + list_size{transit_fare + transit_user_vehicle_tag_identity}.

Logical Architecture Reference Flow: transit_user_vehicle_image
This data flow is sent from the Manage Transit function to the Provide Electronic Payment Services function. It contains a compressed image of the transit user who has violated the transit fare collection process on-board a transit vehicle. The data will be used in subsequent transit fare violation processing.

Logical Architecture Reference Flow: transit_vehicle_advanced_payment_request
This data flow is used within the Manage Transit function and contains data about advanced fares and tolls requested by travelers (as transit users) from on-board transit vehicles. It consists of the following data items each of which is defined in its own DDE: advanced_charges + advanced_tolls + transit_vehicle_location.

Logical Architecture Reference Flow: transit_vehicle_fare_payment_confirmation
This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the previous request for the cost of the current transit fare has been deducted from the credit currently stored by the transit user's payment instrument has been completed successfully. The data flow is used when the transit user is paying for the transit fare on-board a transit vehicle and consists of the following data item which is defined in its own DDE: confirmation_flag.

Physical Architecture Flow Name: request for bad tag list

Request for list of bad vehicle tag Ids.

Logical Architecture Reference Flow: bad_tag_list_request
This data flow is sent from the Manage Transit fare billing on vehicle facility to the Provide Electronic Payment Services function. It requests that a new copy of the list of bad transit tags be provided for use in fare transaction processing on-board a transit vehicle. The data flow consists of the following data item which is defined in its own DDE: transit_vehicle_identity.

Physical Architecture Flow Name: transit vehicle conditions

Operating conditions of transit vehicle (e.g., mileage).

Logical Architecture Reference Flow: transit_vehicle_collected_maintenance_data
This data flow is used by processes within the Manage Transit function and contains data collected from the transit vehicle. The data is produced by sensors analyzing conditions on-board the vehicle during the course of its operation. The data flow consists of the following data items each of which is defined in its own DDE: transit_vehicle_mileage_accumulated + transit_vehicle_operating_condition.

Physical Architecture Flow Name: transit vehicle location data

Current transit vehicle location and related operational conditions data provided by a transit vehicle.

Logical Architecture Reference Flow: transit_vehicle_collected_trip_data
This data flow is used by processes within the Manage Transit function and contains data collected from the transit vehicle. The data is produced by sensors analyzing conditions on-board the vehicle during the course of its operation. The data flow consists of the following data items each of which is defined in its own DDE: transit_vehicle_passenger_loading + transit_vehicle_running_times.

Logical Architecture Reference Flow: transit_vehicle_location
This data flow is used within the Manage Transit function to provide the exact location of the transit vehicle. It contains the transit vehicle location plus the its identity and consists the following items each of which is defined in its own DDE: transit_vehicle_identity + transit_vehicle_location_data.

Logical Architecture Reference Flow: transit_vehicle_location_for_store
This data flow is used within the Manage Transit function to provide the exact location of the transit vehicle for storage so that it can be used by other facilities and functions within ITS. It contains the transit vehicle location plus the its identity and consists the following items each of which is defined in its own DDE: transit_vehicle_identity + transit_vehicle_location_data.

Physical Architecture Flow Name: transit vehicle passenger and use data

Data collected on board the transit vehicle pertaining to availability and/or passenger count.

Logical Architecture Reference Flow: paratransit_transit_vehicle_availability
This data flow is used within the Manage Transit function and contains the current availability of a transit vehicle for paratransit services. This availability has been computed from processing the inputs from on-board sensors within the transit vehicle.

Logical Architecture Reference Flow: transit_vehicle_passenger_data
This data flow is used within the Manage Transit function and contains the number of passengers carried by a transit vehicle while in service. It is derived from on-board vehicle fare collection data and may be used for calculating future transit schedules. It consists of the following data items each of which is defined in its own DDE: transit_passenger_numbers + transit_route_number + transit_route_segment_number + transit_route_use_time + transit_user_category + transit_vehicle_identity.

Physical Architecture Flow Name: transit vehicle schedule performance

Estimated times of arrival and anticipated schedule deviations reported by a transit vehicle.

Logical Architecture Reference Flow: transit_vehicle_arrival_conditions
This data flow is used within the Manage Transit function and contains the deviations from the published data of a transit service. This data is for output to the intermodal transportation service providers so that adjustments can be made to their services to enable transit users to make their connections at modal interchange points.

Logical Architecture Reference Flow: transit_vehicle_deviations_from_schedule
This data flow is used within the Manage Transit function and contains the deviations of a transit vehicle from its published schedule. It is used in calculating the return to the published schedule where the deviation is major and/or it applies to several vehicles on a particular route.

Logical Architecture Reference Flow: transit_vehicle_eta
This data flow is used within the Manage Transit function. It contains the estimated time of arrival of a transit vehicle at the end of a transit route segment, which is usually a stop, plus the route and service number on which it is operating. It is used for individual transit vehicle deviations and contains the following data items each of which is defined in its own DDE: transit_vehicle_identity + transit_vehicle_time + transit_route_number.

Logical Architecture Reference Flow: transit_vehicle_location_for_deviation
This data flow is used within the Manage Transit function to provide the exact location of the transit vehicle for the calculation of any return to schedule scenarios. It contains the transit vehicle location plus its identity and consists of the following items, each of which is defined in its own DDE: transit_vehicle_identity + transit_vehicle_location_data.

Logical Architecture Reference Flow: transit_vehicle_schedule_deviation
This data flow is used within the Manage Transit function and contains the deviation of a transit vehicle from its published schedule.

Physical Architecture Flow Name: traveler request

Request by a traveler to summon assistance, request information, make a reservation, or initiate any other traveler service.

Logical Architecture Reference Flow: other_services_vehicle_request
This data flow is sent from the Manage Transit function to the Provide Electronic Payment Services function and contains the transit user's request from a transit vehicle for other (yellow pages) services. It consists of the following data items each of which is defined in its own DDE : traveler_identity + credit_identity + other_services_data.

Logical Architecture Reference Flow: transit_services_for_eta_request
This data flow is used within the Manage Transit function to request the details of the current transit service so that a transit vehicle can calculate its current deviation relative to that schedule. It consists of the following data items each of which is defined in its own DDE: transit_vehicle_identity + transit_route_number + transit_route_schedule_number.

Weather Service => **Transit Management**

Physical Architecture Flow Name: weather information

Accumulated predicted and current weather data (e.g., temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.).

Logical Architecture Reference Flow: fws_current_weather
This data flow is sent to the Manage Traffic function and the Provide Driver and Traveler Services functions. It contains details of the current weather conditions, e.g. temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.

Logical Architecture Reference Flow: fws_predicted_weather

This data flow is sent to the Manage Traffic and Provide Driver and Traveler Services functions. It contains details of the predicted weather conditions, e.g. temperature, pressure, wind speed, wind direction, humidity, precipitation, visibility, light conditions, etc.

2.20.4 Subsystem Architecture Flow Diagrams

TRMS

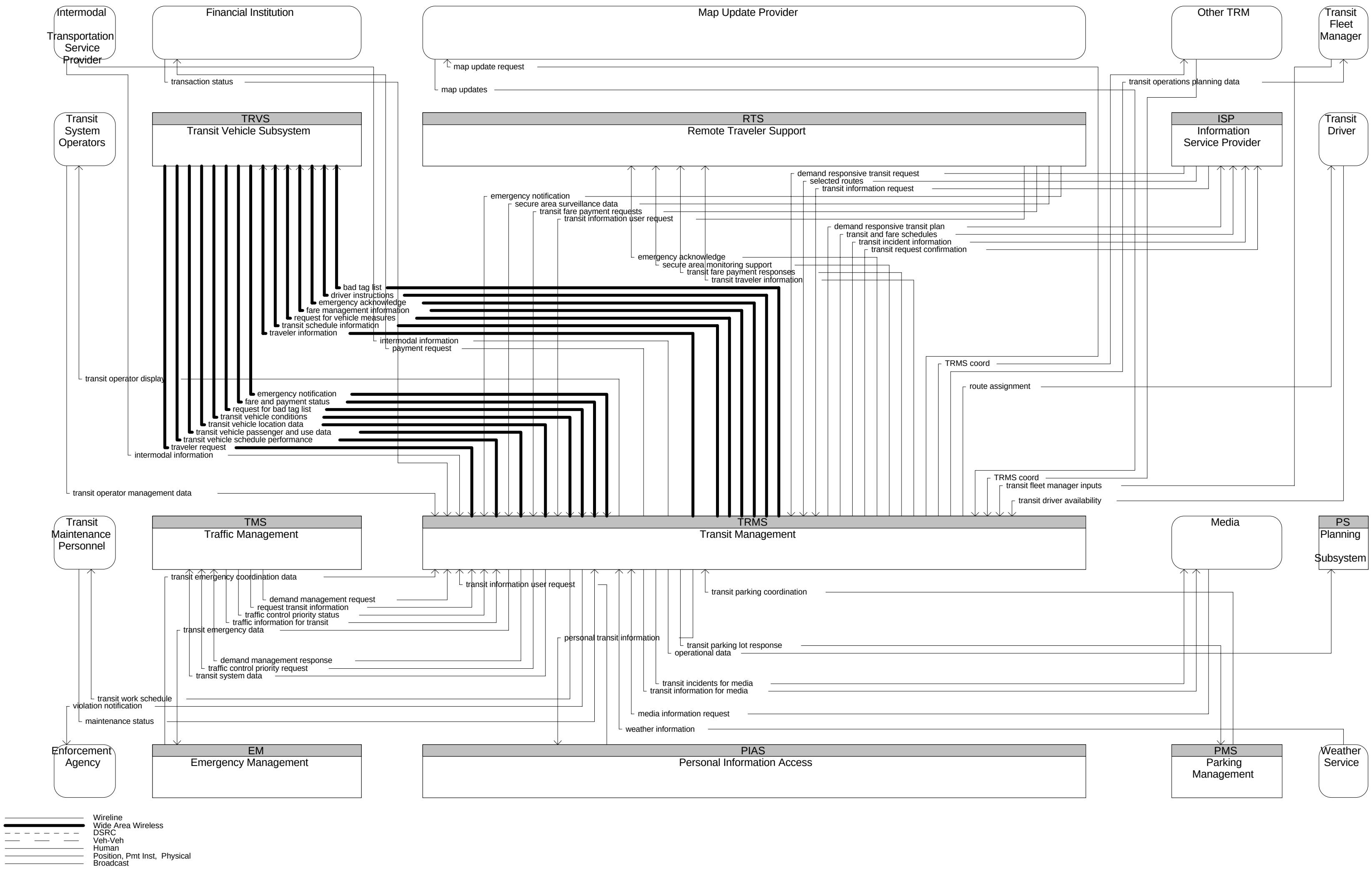


Figure 2.20-2 Architecture Flow Diagram for Transit Management

2.21 Transit Vehicle Subsystem

This subsystem resides in a transit vehicle and provides the sensory, processing, storage, and communications functions necessary to support safe and efficient movement of passengers. The Transit Vehicle Subsystem collects accurate ridership levels and supports electronic fare collection. An optional traffic signal prioritization function communicates with the roadside subsystem to improve on-schedule performance. Automated vehicle location functions enhance the information available to the Transit Management Subsystem enabling more efficient operations. On-board sensors support transit vehicle maintenance. The Transit Vehicle Subsystem also furnishes travelers with real-time travel information, continuously updated schedules, transfer options, routes, and fares.

2.21.1 Alternative Configurations

The Transit Vehicle Subsystem represents the special functions associated with the vehicle moving travelers. Specific features include traveler security, access to schedules, processing of fares, and access to general information. Transit vehicles may be busses, taxies, paratransit vehicles and so on.

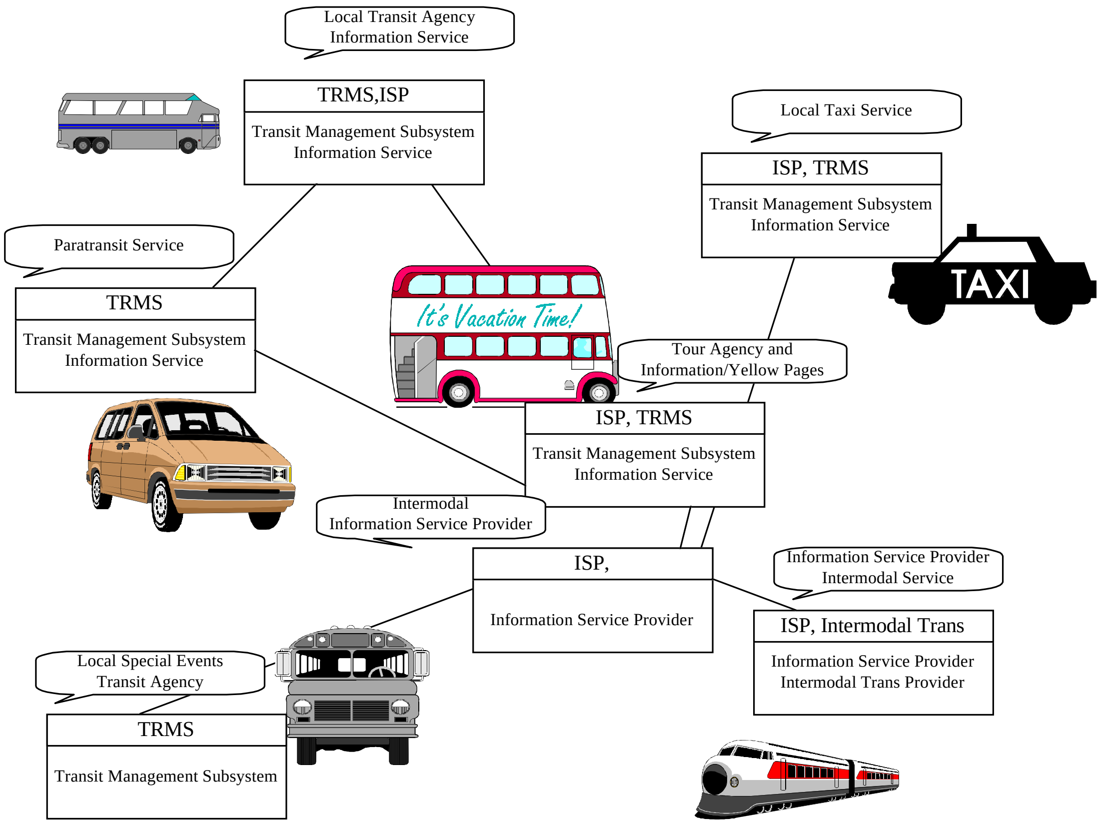


Figure 2.21-1 Alternative Configurations for Transit Vehicles

2.21.2 Subsystem Equipment Packages and Supporting Process Specifications for TRVS

On-board Fixed Route Schedule Management

This Equipment package provides the capabilities for automated planning and scheduling, by collecting data for schedule generation. Capability shall also be provided to automatically determine optimum scenarios for schedule adjustment. This Equipment package also supports the capability for two-way voice communication between the transit vehicle driver and a facility, two-way data communication between the transit vehicles and a facility, on-board safety sensor data to be transmitted from the transit vehicles to a facility, and data transmission from individual facilities to a central facility for processing/analysis if desired.

Process Specifications

4.1.2.1 Determine Transit Vehicle Deviation and ETA

Overview: This process shall determine the schedule deviation and estimated times of arrival (ETA) at transit stops of a transit vehicle. The data shall be sent by this process to other processes in the Manage Transit function for use in calculating corrective instructions for output to the transit vehicle drivers, for use in calculation of a much wider return to schedule strategy where more than one vehicle and/or service is involved, and for storage as transit vehicle operational data. This process shall also send the data to the transit driver interface process, so that the driver is aware of the actual schedule deviation. This output shall be set to zero (no deviation) when that condition occurs, even when it has followed a period of deviation from schedule.

4.1.2.2 Determine Transit Vehicle Corrective Instructions

Overview: This process shall generate outputs that enable a transit vehicle schedule deviation to be corrected. The process shall derive its outputs from data received from another process in the Manage Traffic function. The outputs produced by the process shall consist of corrective instructions for output to the transit vehicle driver by a process on-board the vehicle, and preemption requests for traffic signal controllers at road and freeway intersections. The process shall only produce this output when another process has determined that deviation is small, or the transit vehicle is operating in an urban area. In all other conditions, the process shall provide an output that shows that there are no corrective instructions.

4.1.2.3 Provide Transit Vehicle Driver Interface

Overview: This process shall provide a schedule correction interface for the transit driver in the transit vehicle. The interface shall provide data to the driver about how far the vehicle is from its schedule and what corrective action the driver must take. The data shall be received by the process from other processes in the Manage Traffic function. The output delivered by the process shall be available in audio or visual form in such way that while alerting the driver to the information it contains, it shall in no way impair the driver's ability to operate the vehicle in a manner that is both safe to its passengers and to other vehicles on the roads and freeways. The process shall maintain the output until new data is received from the other processes.

On-board Maintenance

This Equipment package provides the capability to use transit vehicle mileage data to automatically generate preventative maintenance schedules for each specific bus by utilizing vehicle tracking data and storing with a trip computer. It also provides the capability for real-time condition monitoring on board the vehicle, and transmission of this information via two-way communication to the management center.

Process Specifications

4.1.1 Process Transit Vehicle Sensor Trip Data

Overview: This process shall collect and process data available to sensors on-board transit vehicles. This data shall be sent by this process to other processes on-board the transit vehicle and elsewhere in the Manage Traffic function for use in determining vehicle schedule deviations and for storage as operations data.

4.1.9 Process Transit Vehicle Sensor Maintenance Data

Overview: This process shall collect and process vehicle maintenance data available to sensors on-board transit vehicles. When processed, the data shall be sent by this process on request to another process in the Manage Transit function for storage as transit vehicle operating data so that it can subsequently be used for work on future vehicle maintenance.

On-board Paratransit Operations

This equipment package forwards paratransit dispatch requests to the driver and forwards acknowledgements to the center. It coordinates with, and assists the driver in managing multi-stop runs associated with demand responsive, flexibly routed transit services.

Process Specifications

4.2.1.5 Process Demand Responsive Transit Vehicle Availability Data

Overview: This process shall manage data input to sensor(s) on board a transit vehicle. Data including the vehicle's availability for use in demand responsive transit services shall be provided by this process to other processes within the Manage Transit function.

4.2.1.6 Provide Demand Responsive Transit Driver Interface

Overview: This process shall provide the interface through which a transit driver will be sent instructions about the demand responsive transit schedule that has been confirmed. The process shall send the data in a format that will enable the driver to implement the schedule. The output provided by the process shall be available in audio or visual form in such a way that while alerting the driver to the information it contains, it shall in no way impair the driver's ability to operate the vehicle in a manner that is both safe to its passengers, and to other vehicles on the roads and freeways. The input and output forms shall also include those that are suitable for travelers with physical disabilities.

On-board Transit Fare and Load Management

This Equipment package provides the capability to collect data required to determine accurate ridership levels and implement variable and flexible fare structures. Support shall be provided for the traveler for use of a fare medium for all applicable surface transportation services, to pay without stopping, have payment media automatically identified as void and/or invalid and eligibility verified, and allow for third party payment. In addition, capability to provide expansion into other uses for payment medium such as retail and telephone and for off-line billing for fares paid by agencies shall be supported. This Equipment package also supports the capability for two-way voice communication between the transit vehicle driver and a facility, two-way data communication between the transit vehicles and a facility, sensor data to be transmitted from the transit vehicles to a facility, and data transmission from individual facilities to a central facility for processing/analysis if desired. These capabilities require integration with an existing On-board Trip Monitoring Equipment package.

Process Specifications

4.6.1 Detect Transit User on Vehicle

Overview: This process shall detect embarking transit users on-board a transit vehicle and read data from the payment instrument that they are carrying. The process shall provide an image of all transit users which shall be used for violation processing of those who do not have a payment instrument or whose transit fare transaction fails. It shall obtain an image of the required accuracy under all lighting conditions and over the range of speeds with which transit users will pass through the fare collection point on a transit vehicle.

4.6.2 Determine Transit User Needs on Vehicle

Overview: This process shall determine the transit user's travel routing based on the transit vehicle's current location and the user's destination. The process shall support the transit user's routing, enabling it to include travel on the vehicle for all or part of its route and (possibly) transfer to another vehicle on another route. In order to achieve this capability, the process shall have access to the complete range of transit services (routes and schedules) that are available to the transit user. The transit vehicle's location shall be provided by other processes within the Manage Transit function. Details of all transactions with the transit user's payment details removed, shall be sent by this process to the interface process for loading into a data store.

4.6.3 Determine Transit Fare on Vehicle

Overview: This process shall calculate the transit user's fare based on the origin and destination provided by the user. The process shall calculate the fare using the transit routing, transit fare category, and transit user history components of the ride data, in addition to information provided by the interface process for the transit fares data store. The accumulated data shall be sent by this process to another process for the actual implementation of the fare payment transaction.

4.6.4 Manage Transit Fare Billing on Vehicle

Overview: This process shall manage the transit user fare payments on-board a transit vehicle. The process shall receive information about the fare that is to be paid and the method of payment adopted by the transit user. It shall always support two modes of operation to complete the back end financial processing: infrastructure interactive, or semi-autonomous batch processing. The interactive method shall be used for individual transactions, such as those in paratransit type operations where value/volume ratios are high. It shall send transit user fare payment data to processes in the Provide Electronic Payment Services function for financial authorization and transaction processing, plus the return of the result for display to the transit user. A failed transaction shall result in the transmission of an image of the transit user to another process. Batch processing shall be used by the process for routes where value/volume ratios are low. It shall be performed using all the same data flows and processes as in the interactive method, except that transaction records are queued in a transaction buffer store which shall be maintained by this process. The accumulated data for the fare transactions shall be sent to the Provide Electronic Payment Services function on command from the transit vehicle driver, or when the transit vehicle has reached a convenient point on its route. The transit vehicle driver shall be notified when batch processing has completed successfully. In either mode of operation, a record of the status of all transit fare processing shall be sent to an interface process for the fare collection storage database.

4.6.5 Provide Transit User Fare Payment Interface on Vehicle

Overview: This process shall provide the fare payment interface for the transit user on-board a transit vehicle. The process shall prompt the transit user for information necessary for the transaction that has not been provided. The result of the transit service ride fare payment plus other services request and payment, shall be reported back to the transit user by the process. The input and output forms shall include those that are suitable for travelers with physical disabilities.

4.6.6 Update Transit Vehicle Fare Data

Overview: This process shall provide a database on-board the transit vehicle for use in fare processing. The database shall contain transit fare information from which the fares for all possible trips within the transit operational network can be determined.

4.6.7 Provide Transit Vehicle Passenger Data

Overview: This process shall provide passenger loading and fare statistics data to other ITS functions. The process shall send the data automatically at regular periodic intervals using data collected in the store of fare transaction data. This store receives data from the process that interfaces to the user on-board a transit vehicle.

7.3.5 Provide Transit Vehicle Payment Instrument Interface

Overview: This process shall be responsible for providing the interface through which the payment information can be read from a transit user tag. The process shall support the reading of this data from transit users embarking on-board transit vehicles, for use in paying the current transit fare and if required, advanced payments. The process shall support advanced payments for tolls, and/or parking lot charges, and/or transit fares. It shall be possible for the process to collect either the credit identity or the stored credit value data from the tag, and to update the stored credit value as a result of the fare and (possibly) advanced charges having been paid.

On-board Transit Information Services
The Equipment package furnishes enroute transit users with real-time travel-related information. Current information that can be provided to transit users includes transit routes, schedules, transfer options, fares, real-time schedule

adherence, current incidents, weather conditions, and special events are provided. In addition to tailored information for individual transit users, this equipment package also supports general annunciation and/or display of general schedule information, imminent arrival information, and other information of general interest to transit users.

Process Specifications

6.2.1.6 Provide Transit Advisory Data On Vehicle

Overview: This process shall gather transit advisory data and provide it via another process to the transit user on-board a transit vehicle. The interface shall receive requests from the transit user specifying the required destination of a transit service ride and other (yellow pages) type services. The transit user may also request and receive information about the state of traffic on the roadway, as well as transit route and stop data (i.e., traffic and transit advisory data). This process extracts data from the store of traveler transit information upon request for advisory data from the driver or transit user in a vehicle. The process shall filter the data read from the store so that output only contains that which is relevant to the current location of the vehicle from which the request was made. The vehicle's location shall be provided to the process in the request data. The input and output forms shall include those that are suitable for travelers with physical disabilities.

6.2.3 Provide Transit User Advisory Interface

Overview: This process shall provide a data input and output interface for a transit user on-board a transit vehicle. The process shall enable traffic and travel advisory information, plus yellow pages information to be requested and output to the transit user. When constructing the outputs the process shall use the data in the store of vehicle display definitions data. In addition to the traveler's request/ response for information, broadcast advisories about the imminent arrival of the transit vehicle at the next stop are also displayed for the transit user. The process shall handle all inputs and outputs in such a way that they do not impair the vehicle driver's ability to control the transit vehicle in a manner that is safe to both its occupants, to other road and freeway users, and to pedestrians. The input and output forms shall also include those that are suitable for travelers with physical disabilities.

On-board Transit Security

This Equipment package provides the capability to monitor the safety of transit vehicles using on-board safety sensors, processors and communications from the prerequisite On-board Trip Monitoring Equipment package.

Process Specifications

4.4.1.2 Manage Transit Emergencies

Overview: This process shall support the management of emergencies that occur in the transit system by processing information received from transit vehicles. The process shall accept inputs from either the transit vehicle driver or a transit user, the latter through such interfaces as panic buttons, alarm switches, etc. The reported emergencies shall be sent to another process for action by the transit system operator and subsequently for output to the media. The process shall also send acknowledgment data to the process providing the interface to the transit driver.

4.4.1.5 Provide Transit Driver Interface for Emergencies

Overview: This process shall provide an interface to the transit vehicle through which the driver can both report an emergency situation and receive an acknowledgment. The process shall provide this interface in such a way that its operation for both inputs and outputs shall be transparent to transit users on board the vehicle and to anyone outside the vehicle, and shall not compromise the safe operation of the vehicle by the driver.

On-board Transit Signal Priority

This Equipment package provides the capability for transit vehicles to request signal priority through short range communication directly with traffic control equipment at the roadside.

Process Specifications

4.1.2.5 Request Transit Vehicle Preemptions

Overview: This process shall provide the interface through which requests for preemption can be output from a transit vehicle. The output shall be received by the process as a result of data sent from another process in the Manage Transit function. The process shall provide the output in a form that can be used by the controllers at intersections, pedestrian crossings and multimodal crossings on the roads (surface streets) and freeway network served by the Manage Traffic function to provide priority of the transit vehicle. If no data is received from the other process, or it shows that no preemption is needed, the process shall produce no output.

On-board Transit Trip Monitoring

This Equipment package provides the capabilities to support fleet management with automatic vehicle location and automated mileage and fuel reporting and auditing. This package may also record other special events resulting from communication with roadside equipment. This includes only the equipment on board the vehicle to support this function including the vehicle location devices such as GPS equipment, communication interfaces, a processor to record trip length, and the sensors/actuators/interfaces necessary to record mileage and fuel usage.

Process Specifications

4.1.1 Process Transit Vehicle Sensor Trip Data

Overview: This process shall collect and process data available to sensors on-board transit vehicles. This data shall be sent by this process to other processes on-board the transit vehicle and elsewhere in the Manage Traffic function for use in determining vehicle schedule deviations and for storage as operations data.

4.1.3 Provide Transit Vehicle Location Data

Overview: This process shall provide the transit vehicle's current location with a high degree of accuracy. The location shall be computed by this process from data sent by other processes that provides basic vehicle location and on-board vehicle conditions, such as proximity to transit stop, vehicle doors opened or closed, etc. The data shall be output continuously by the process and sent to other processes for their use and for storage.

2.21.3 Subsystem Interfaces for TRVS

Payment Instrument => Transit Vehicle Subsystem

Physical Architecture Flow Name: payment

Payment of some kind (e.g., toll, parking, fare) by traveler which in most cases can be related to a credit account.

Logical Architecture Reference Flow: fpi_confirm_fare_payment_on_transit_vehicle

This data flow is sent from the payment instrument to the Provide Electronic Payment Services function. It contains confirmation that the requested payment of the current transit fare, plus if required the cost of advanced tolls, and/or parking lot charges, and/or transit fares, has been successfully deducted from the total credit previously stored by the payment instrument. This data flow will only apply to those types of payment instrument that can carry stored credit and will not be set by those that only contain a credit identity.

Logical Architecture Reference Flow: fpi_transit_vehicle_tag_data

This data flow is sent from the payment instrument to the Provide Electronic Payment Services function and is used to either identify a particular payment instrument or the amount of credit that it currently has stored, when the instrument is being used on-board a transit vehicle. In either case the data will be used to enable automatic billing for the current transit fare, plus if required, advanced payments for tolls, and/or parking lot charges and/or transit fares. The payments will be for the transit vehicle passengers, i.e. transit users and not for the vehicle itself. A payment instrument is a device that can be used to make payments, e.g. a debit card, or a device that contains stored credit that can be used for actual payments. It will belong to the financial institution responsible for its issue and not to the user, who in this instance is the transit user.

Transit Driver => **Transit Vehicle Subsystem**

Physical Architecture Flow Name: transit driver inputs

Transit driver emergency request as well as fare transaction data.

Logical Architecture Reference Flow: ftd_emergency_request
This data flow is sent by the transit driver to the Manage Transit function. It contains notification that the driver needs help because an emergency has occurred on-board or near a transit vehicle.

Logical Architecture Reference Flow: ftd_fare_transaction_mode_set_up
This data flow is sent from the transit driver to the Manage Transit function. It contains the mode in which the transit fare transaction processing on-baord the transit vehicle is to operate. This may be either batch mode (part processing of each fare transaction carried out and the details of a large number of transactions transferred to the central function for further processing) or interactive mode (complete processing of each transaction carried out without stopping).

Logical Architecture Reference Flow: ftd_request_batch_mode_data_transfer
This data flow is sent from the transit driver to the Manage Transit function. It contains a request that the current contents of the store of transit fare transactions be transferred to processes in the Provide Electronic Payment Services function for further processing.

Transit Management => **Transit Vehicle Subsystem**

Physical Architecture Flow Name: bad tag list

List of invalid transit user tags which may have previously failed a fare payment transaction.

Logical Architecture Reference Flow: bad_tag_list_update
This data flow is sent from the Provide Electronic Payment Services function to the Manage transit function. It contains a list of current transit user tags that have been found to be bad. This means that a fare payment transaction in which they were involved has failed, or the tag has been invalidated by the financial institution to which it belongs. The data flow consists of the following data items each of which is defined in its own DDE: transit_vehicle_identity + list_size + list_size{credit_identity}.

Physical Architecture Flow Name: driver instructions

Transit service instructions for both transit and paratransit drivers.

Logical Architecture Reference Flow: approved_corrective_plan
This data flow is used within the Manage Transit function and represents a plan of action to respond to a (relatively) long term transit schedule disruption. It consists of the following data items each of which is defined in its own DDE: transit_route_corrections + transit_changes_in_stops + transit_changes_in_speed.

Logical Architecture Reference Flow: paratransit_transit_driver_instructions
This data flow is used within the Manage Transit function and contains the instructions for the transit driver to follow in order that a paratransit service can be executed. The instructions will contain such things as pick-up points, traveler identities, drop off points, etc. The driver can obtain actual route guidance through the on-line vehicle guidance facility available from the Provide Driver and Traveler Services function.

Logical Architecture Reference Flow: transit_services_for_corrections
This data flow is used within the Manage Transit function. It is sent to the Operate Transit Vehicles facility for use in the calculation of corrections to transit vehicle routes and schedules to restore a service to normal operation. It contains a complete set of all the transit routes and the services that run upon them, including timings, etc. that are provided by

the transit fleet from which the data was requested. list_size + list_size{transit_services}.

Logical Architecture Reference Flow: transit_services_for_eta
This data flow is used within the Manage Transit function. It is sent to the Operate Transit Vehicles facility for use in the calculation of transit vehicle estimated times of arrival (eta) at transit stops. It only contains details of the schedule for the transit route that is currently being operated by the vehicle. The data flow consists of the following data items each of which is defined in its own DDE: transit_route_number + transit_route_segment_list + transit_route_stop_list.

Physical Architecture Flow Name: emergency acknowledge

Acknowledge request for emergency assistance and provide additional details regarding actions and verification requirements.

Logical Architecture Reference Flow: request_transit_user_vehicle_image
This data flow is sent from the Provide Electronic Payment Services function to the Manage Transit function. It contains a request for the supply of the image of a transit user who has violated the transit fare payment process at an on-board vehicle fare collection point.

Logical Architecture Reference Flow: transit_operator_request_acknowledge
This data flow is used within the Manage Transit function and contains an acknowledgment that the previous notification of an emergency to the transit system operator has been received and is being considered for action.

Physical Architecture Flow Name: fare management information

Transit fare information and transaction data used to manage transit fare processing on the transit vehicle.

Logical Architecture Reference Flow: confirm_vehicle_fare_payment
This data flow is sent from the Provide Electronic Payment Services function to the Manage transit function to confirm that transaction processing of the payment of a single transit fare (interactive operation) or of a group of fares (batch mode) from on-board the vehicle has been completed. If the transaction processing was in batch mode, the transit user's tag identity will be set to zero (0), otherwise it will be set to the identity provided in the transaction request. The data flow consists of the following data items each of which is defined in its own DDE: confirmation_flag + transit_user_vehicle_tag_identity + transit_vehicle_identity.

Logical Architecture Reference Flow: transit_services_for_vehicle_fares
This data flow is used within the Manage Transit function. It contains details of the transit user fares for all the transit routes operated by the transit fleet from which the request was made. This data is for use in processing transit fare payments initiated by transit users on-board a transit vehicle. The data flow consists of the following data item which is defined in its own DDE: transit_route_fare_data.

Logical Architecture Reference Flow: transit_vehicle_advanced_payment_response
This data flow is used within the Manage Transit function and contains the result of the requested advanced payment transaction from a traveler (as a transit user) in a transit vehicle. It consists of the following data items each of which is defined in its own DDE: advanced_charges_confirm + advanced_tolls_confirm + confirmation_flag + transit_vehicle_identity.

Logical Architecture Reference Flow: transit_vehicle_fare_data
This data flow is sent by the Provide Electronic Payment Services function to the Manage Transit function and contains details of the fares being currently charged for regular transit services. It is for use in calculating fares that are to be paid by transit users on-board a transit vehicle and consists of the following data item which is defined in its own DDE: transit_fares.

Logical Architecture Reference Flow: transit_vehicle_fare_payment_debited
This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the cost of the current transit fare will be deducted by the financial institution from the credit identity previously provided by

the payment instrument being used by the transit user at the roadside. It is only sent when a credit identity has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: transit_vehicle_fare_payment_request
This data flow is used within the Provide Electronic Payment Services function and contains the request for the cost of the current transit fare to be deducted from the credit currently stored by the transit user's payment instrument, when it is being used on-board a transit vehicle. It is only sent when a value of stored credit has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: transit_fare.

Physical Architecture Flow Name: request for vehicle measures

Request for vehicle performance and maintenance data collected by onboard sensors.

Logical Architecture Reference Flow: transit_vehicle_collected_maintenance_data_request
This data flow is used by processes within the Manage Transit function and contains a request for data collected on-board the transit vehicle. The data is produced by sensors analyzing conditions on-board the vehicle during the course of its operation.

Physical Architecture Flow Name: transit schedule information

Current and projected transit schedule adherence.

Logical Architecture Reference Flow: transit_vehicle_advisory_eta
This data flow is used as an interface between the Manage Transit and Provide Driver and Traveler Information functions. It contains the estimated time of arrival of a transit vehicle at the end of a transit route segment, which is usually a stop, plus the route and service number on which it is operating. It is used for individual transit vehicle deviations and contains the following data items each of which is defined in its own DDE: transit_vehicle_identity + transit_vehicle_time + transit_route_number.

Physical Architecture Flow Name: traveler information

Traveler information comprised of traffic status, advisories, incidents, responses to traveler requests (e.g., traveler routing, yellow pages), payment information and many other travel-related data updates and confirmations.

Logical Architecture Reference Flow: other_services_vehicle_response
This data flow is sent from the Provide Electronic Payment Services function to the Manage Transit function and contains the response to the transit user's request from a transit vehicle for other (yellow pages) services. It consists of the following data items each of which is defined in its own DDE : traveler_identity + credit_identity + other_services_data.

Transit User => **Transit Vehicle Subsystem**

Physical Architecture Flow Name: emergency notification

An emergency request for assistance originated by a traveler using an in-vehicle, public access, or personal device. Sufficient information is provided so that the recipient can determine the location of the emergency as a minimum. Additional information identifying the requestor and requesting device and the nature and severity of the emergency may also be provided (and required) by some systems.

Logical Architecture Reference Flow: ftu_emergency_request
This data flow is sent from the transit user to the Manage transit function and is used to identify that an emergency of some type has been seen, and may involve the transit user personally.

Logical Architecture Reference Flow: ftu_transit_user_vehicle_image
This data flow is used within the Manage Transit function and contains analog information from which sensors can produce an image of the transit user when a fare transaction violation has been detected on-board the vehicle. The size estimate represents a digitized equivalent of the analog image .

Physical Architecture Flow Name: transit user inputs

Requests from transit user through either an on-board or fixed location traveler information station.

Logical Architecture Reference Flow: ftu_destination_on_vehicle
This data flow is sent by the transit user to the Manage Transit function and is used to specify the destination of a desired service for which a fare has to be paid on-board the transit vehicle.

Logical Architecture Reference Flow: ftu_other_services_vehicle_request
This data flow is sent by the transit user to the Manage Transit function to specify other non-transit services that are needed by a transit user on-board a transit vehicle.

Logical Architecture Reference Flow: ftu_request_advisory_information
This data flow is sent from the traveler to the Provide Driver and Traveler Services function and contains data that enables a display on-board a transit vehicle that will be capable of showing transit, yellow pages information and other traffic and travel advisory information.

Transit Vehicle => **Transit Vehicle Subsystem**

Physical Architecture Flow Name: vehicle measures

Sensing information from vehicle sensors.

Logical Architecture Reference Flow: ftv_availability
This data flow is sent from the transit vehicle to the Manage Transit function. It defines the availability of a transit vehicle in terms of its identity, type, and passenger capacity.

Logical Architecture Reference Flow: ftv_vehicle_maintenance_data
This data flow is sent from the transit vehicle to the Manage Transit function. It contains analog data from which various aspects of the transit vehicle operation can be determined for future maintenance purposes.

Logical Architecture Reference Flow: ftv_vehicle_trip_data
This data flow is sent from the transit vehicle to the Manage Transit function. It contains analog data from which various aspects of the transit vehicle operation can be determined so that its performance on the current trip can be analyzed.

Transit Vehicle Subsystem => **Payment Instrument**

Physical Architecture Flow Name: request for payment

Request to deduct cost of service from user's payment account.

Logical Architecture Reference Flow: tpi_debited_payment_on_transit_vehicle
This data flow is sent to the payment instrument by the Provide Electronic Payment Services function and contains confirmation that the cost of the current transit fare incurred on-board a transit vehicle, plus if required the cost of advanced tolls, and/or parking lot charges, and/or transit fares, will be debited to the credit identity provided by the payment instrument. The debit transaction will be carried out through the financial institution through other processes within the function.

Logical Architecture Reference Flow: tpi_request_fare_payment_on_transit_vehicle
This data flow is sent to the payment instrument from the Provide Electronic Payment Services function. It is a request to deduct the total cost of the current transit fare, or if required, that for advanced tolls, and/or parking lot charges and/or transit fares from the credit currently stored by the payment instrument when used on-board a transit vehicle.

Transit Vehicle Subsystem => Roadway Subsystem

Physical Architecture Flow Name: local signal priority request

Request from a vehicle to a signalized intersection for priority at that intersection.

Logical Architecture Reference Flow: transit_vehicle_roadway_preemptions
This data flow is sent from the Manage Transit function to the Manage Traffic function and contains data necessary for an individual transit vehicle to be given preemption (priority) at indicator controllers. This will be at the controller for a particular road junction, pedestrian crossing, or highway entrance ramp. The data is sent directly from the transit vehicle to the next controller along its route and therefore is not subject to any centralized coordination. Local coordination may be provided if there are links between adjacent controllers. The data flow consists of the following data items each of which is defined in its own DDE: transit_vehicle_junction_preemption + transit_vehicle_pedestrian_preemption + transit_vehicle_ramp_preemption + transit_vehicle_sign_preemption.

Transit Vehicle Subsystem => Transit Driver

Physical Architecture Flow Name: transit driver display

Display (either video or audio) to transit driver containing status of various ITS services.

Logical Architecture Reference Flow: ttd_batch_mode_data_transfer_status
This data flow is sent to the transit vehicle driver by the Manage Transit function. It contains details of the result of the previous request for a transfer of transit fare transaction data to the Provide Electronic Payment Services function for further processing. If the transfer or processing failed the driver may request it to be repeated.

Logical Architecture Reference Flow: ttd_corrective_instructions
This data flow is sent to the transit vehicle driver by the Manage Transit function. It contains items of data which are designed to help the driver restore the transit vehicle to its correct schedule. These will comprise such things as corrections to the current route, changes to the sequence plus inclusion or deletion of stops and changes in the proscribed vehicle speed between stops.

Logical Architecture Reference Flow: ttd_emergency_information
This data flow is sent to the transit vehicle driver by the Manage Transit function. It contains an acknowledgment that the emergency request previously input by the driver has been received.

Logical Architecture Reference Flow: ttd_paratransit_information
This data flow is sent to the transit vehicle driver by the Manage Transit function. It contains information about the paratransit schedule that the transit driver is being requested to perform. This information comprises data such as the route, pick-up time, drop of point, and route. It may involve picking up one or more travelers at different locations along the route.

Logical Architecture Reference Flow: ttd_request_fare_transaction_mode_set_up
This data flow is sent from the transit driver to the Manage Transit function. It contains a request that the driver inputs the mode of transit fare transaction processing that is to be used on-board the vehicle. Either one of the following two modes is possible: batch mode (part processing of each fare transaction carried out and the details of a large number of transactions transferred to the central function for further processing) or interactive mode (complete processing of each

transaction carried out without stopping).

Logical Architecture Reference Flow: ttd_transit_vehicle_schedule_deviations
This data flow is sent to the transit vehicle driver by the Manage Transit function. It contains information for the driver on deviations from the transit route and/or schedule in order that normal service operation can be provided.

Transit Vehicle Subsystem => Transit Management

Physical Architecture Flow Name: emergency notification

An emergency request for assistance originated by a traveler using an in-vehicle, public access, or personal device. Sufficient information is provided so that the recipient can determine the location of the emergency as a minimum. Additional information identifying the requestor and requesting device and the nature and severity of the emergency may also be provided (and required) by some systems.

Logical Architecture Reference Flow: transit_emergency_details
This data flow is used within the Manage Transit function and contains details of emergency requests that have been input on-board a transit vehicle. It consists of the following data items each of which is defined in its own DDE: transit_driver_emergency_request + transit_user_emergency_request + transit_vehicle_location.

Logical Architecture Reference Flow: transit_emergency_information
This data flow is used within the Manage Transit function and contains details of emergency requests that have been input on-board a transit vehicle. It consists of the following data items each of which is defined in its own DDE: transit_driver_emergency_request + transit_user_emergency_request + transit_vehicle_location.

Logical Architecture Reference Flow: transit_operator_emergency_request
This data flow is used within the Manage Transit function and contains information about an incident that has been detected on board a transit vehicle or at a transit facility following input from a transit user or transit vehicle driver. The data is for output to the transit system operator so that responsive action can be initiated.

Physical Architecture Flow Name: fare and payment status

Current fare collection information including the operational status of the fare collection equipment and financial payment transaction data.

Logical Architecture Reference Flow: fare_collection_vehicle_violation_information
This data is used by the Manage Transit function to send data about a violator of the transit fare collection processes on-board the vehicle to the Manage Emergency Services function. This data flow will contain a digitized video image of the transit user who is trying to violate the fare collection process on-board a vehicle. It is assumed that this digitized data will include other data such as date and time, plus camera identity from which the transit vehicle identity can be determined. The data flow consists of the following data items each of which is defined in its own DDE: transit_route_number + transit_route_segment_number + transit_user_vehicle_image + transit_user_vehicle_tag_identity.

Logical Architecture Reference Flow: request_vehicle_fare_payment
This data flow is sent from the Manage Transit fare billing on vehicle facility to the Provide Electronic Payment Services function. It requests payment processing of one or more transit fare transactions from on-board a transit vehicle. This flow provides for both batch (low value/high usage) fare transactions (e.g. city bus routes) and for high value/low volume, interactive, near real-time transactions (e.g. individualized flexible transit). The size assumption below is appropriate for the interactive mode of operation (one transaction per message) which can be assumed to present the greater stress on the transit vehicle communications process due to the higher frequency of operation. The data flow consists of the following data items each of which is defined in its own DDE: transit_route_number + transit_vehicle_identity + transit_vehicle_fare_collection_method + list_size{transit_fare + transit_user_vehicle_tag_identity}.

Logical Architecture Reference Flow: transit_user_vehicle_image
This data flow is sent from the Manage Transit function to the Provide Electronic Payment Services function. It contains a compressed image of the transit user who has violated the transit fare collection process on-board a transit vehicle. The data will be used in subsequent transit fare violation processing.

Logical Architecture Reference Flow: transit_vehicle_advanced_payment_request
This data flow is used within the Manage Transit function and contains data about advanced fares and tolls requested by travelers (as transit users) from on-board transit vehicles. It consists of the following data items each of which is defined in its own DDE: advanced_charges + advanced_tolls + transit_vehicle_location.

Logical Architecture Reference Flow: transit_vehicle_fare_payment_confirmation
This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the previous request for the cost of the current transit fare has been deducted from the credit currently stored by the transit user's payment instrument has been completed successfully. The data flow is used when the transit user is paying for the transit fare on-board a transit vehicle and consists of the following data item which is defined in its own DDE: confirmation_flag.

Physical Architecture Flow Name: request for bad tag list

Request for list of bad vehicle tag Ids.

Logical Architecture Reference Flow: bad_tag_list_request
This data flow is sent from the Manage Transit fare billing on vehicle facility to the Provide Electronic Payment Services function. It requests that a new copy of the list of bad transit tags be provided for use in fare transaction processing on-board a transit vehicle. The data flow consists of the following data item which is defined in its own DDE: transit_vehicle_identity.

Physical Architecture Flow Name: transit vehicle conditions

Operating conditions of transit vehicle (e.g., mileage).

Logical Architecture Reference Flow: transit_vehicle_collected_maintenance_data
This data flow is used by processes within the Manage Transit function and contains data collected from the transit vehicle. The data is produced by sensors analyzing conditions on-board the vehicle during the course of its operation. The data flow consists of the following data items each of which is defined in its own DDE: transit_vehicle_mileage_accumulated + transit_vehicle_operating_condition.

Physical Architecture Flow Name: transit vehicle location data

Current transit vehicle location and related operational conditions data provided by a transit vehicle.

Logical Architecture Reference Flow: transit_vehicle_collected_trip_data
This data flow is used by processes within the Manage Transit function and contains data collected from the transit vehicle. The data is produced by sensors analyzing conditions on-board the vehicle during the course of its operation. The data flow consists of the following data items each of which is defined in its own DDE: transit_vehicle_passenger_loading + transit_vehicle_running_times.

Logical Architecture Reference Flow: transit_vehicle_location
This data flow is used within the Manage Transit function to provide the exact location of the transit vehicle. It contains the transit vehicle location plus the its identity and consists the following items each of which is defined in its own DDE: transit_vehicle_identity + transit_vehicle_location_data.

Logical Architecture Reference Flow: transit_vehicle_location_for_store
This data flow is used within the Manage Transit function to provide the exact location of the transit vehicle for storage so that it can be used by other facilities and functions within ITS. It contains the transit vehicle location plus the its

identity and consists the following items each of which is defined in its own DDE: transit_vehicle_identity + transit_vehicle_location_data.

Physical Architecture Flow Name: transit vehicle passenger and use data

Data collected on board the transit vehicle pertaining to availability and/or passenger count.

Logical Architecture Reference Flow: paratransit_transit_vehicle_availability
This data flow is used within the Manage Transit function and contains the current availability of a transit vehicle for paratransit services. This availability has been computed from processing the inputs from on-board sensors within the transit vehicle.

Logical Architecture Reference Flow: transit_vehicle_passenger_data
This data flow is used within the Manage Transit function and contains the number of passengers carried by a transit vehicle while in service. It is derived from on-board vehicle fare collection data and may be used for calculating future transit schedules. It consists of the following data items each of which is defined in its own DDE: transit_passenger_numbers + transit_route_number + transit_route_segment_number + transit_route_use_time + transit_user_category + transit_vehicle_identity.

Physical Architecture Flow Name: transit vehicle schedule performance

Estimated times of arrival and anticipated schedule deviations reported by a transit vehicle.

Logical Architecture Reference Flow: transit_vehicle_arrival_conditions
This data flow is used within the Manage Transit function and contains the deviations from the published data of a transit service. This data is for output to the intermodal transportation service providers so that adjustments can be made to their services to enable transit users to make their connections at modal interchange points.

Logical Architecture Reference Flow: transit_vehicle_deviations_from_schedule
This data flow is used within the Manage Transit function and contains the deviations of a transit vehicle from its published schedule. It is used in calculating the return to the published schedule where the deviation is major and/or it applies to several vehicles on a particular route.

Logical Architecture Reference Flow: transit_vehicle_eta
This data flow is used within the Manage Transit function. It contains the estimated time of arrival of a transit vehicle at the end of a transit route segment, which is usually a stop, plus the route and service number on which it is operating. It is used for individual transit vehicle deviations and contains the following data items each of which is defined in its own DDE: transit_vehicle_identity + transit_vehicle_time + transit_route_number.

Logical Architecture Reference Flow: transit_vehicle_location_for_deviation
This data flow is used within the Manage Transit function to provide the exact location of the transit vehicle for the calculation of any return to schedule scenarios. It contains the transit vehicle location plus its identity and consists of the following items, each of which is defined in its own DDE: transit_vehicle_identity + transit_vehicle_location_data.

Logical Architecture Reference Flow: transit_vehicle_schedule_deviation
This data flow is used within the Manage Transit function and contains the deviation of a transit vehicle from its published schedule.

Physical Architecture Flow Name: traveler request

Request by a traveler to summon assistance, request information, make a reservation, or initiate any other traveler service.

Logical Architecture Reference Flow: other_services_vehicle_request
This data flow is sent from the Manage Transit function to the Provide Electronic Payment Services function and

contains the transit user's request from a transit vehicle for other (yellow pages) services. It consists of the following data items each of which is defined in its own DDE : traveler_identity + credit_identity + other_services_data.

Logical Architecture Reference Flow: transit_services_for_eta_request
This data flow is used within the Manage Transit function to request the details of the current transit service so that a transit vehicle can calculate its current deviation relative to that schedule. It consists of the following data items each of which is defined in its own DDE: transit_vehicle_identity + transit_route_number + transit_route_schedule_number.

Transit Vehicle Subsystem => Transit User

Physical Architecture Flow Name: transit user fare status

Status of fare transaction for transit user.

Logical Architecture Reference Flow: ttu_vehicle_payment_confirmed
This data flow is sent to the transit user by the Manage Transit function. It contains a message giving the transit user on-board the vehicle details of the success or failure of the payment transaction.

Physical Architecture Flow Name: transit user outputs

Information for traveler from either an on-board or fixed location traveler information station.

Logical Architecture Reference Flow: ttu_advisory_information
This data flow is used as part of the interface to the traveler by the Provide Driver and Traveler Services function. It contains displays of the traffic advisory, incident, and stop annunciation information produced in response to a previous traveler request.

Logical Architecture Reference Flow: ttu_other_services_vehicle_confirmed
This data flow is sent to the transit user by the Manage Transit function. It contains a message giving the transit user on-board the vehicle details of the success or failure of the request for other (yellow pages) services.

Logical Architecture Reference Flow: ttu_traveler_information
This data flow is used as part of the interface to the traveler by the Provide Driver and Traveler Services function. It contains displays of the travel data, transit routes and schedules, and other services (yellow pages) information produced in response to a previous traveler request.

Logical Architecture Reference Flow: ttu_vehicle_access_message
This data flow is sent to the transit user by the Manage Transit function. It contains a message giving the transit user details of the success or failure of the fare transaction previously initiated from on-board the vehicle.

Transit Vehicle Subsystem => Vehicle

Physical Architecture Flow Name: traveler advisory request

In vehicle communication between transit and vehicle systems includes advisories and advance payment deductions.

Logical Architecture Reference Flow: transit_user_advanced_payment_on_vehicle
This data flow is sent from the Manage Transit function to the Provide Electronic Payment Services function. It contains the cost of advanced payments that must be deducted from the credit currently stored on the payment instrument being used by a transit user on-board a transit vehicle. These advanced payments may cover tolls, and/or parking lot charges, and/or transit fares. The data flow consists of the following data items each of which is defined in its own DDE: stored_credit + parking_lot_cost + toll_cost + transit_fare.

Logical Architecture Reference Flow: transit_user_advisory_information_request
This data flow is used within the Provide Driver and Traveler Services function and contains analyzed requests for the various types of transit user display. It contains the following data items each of which is defined in its own DDE: advisory_display_type + advisory_data_scope.

Vehicle => **Transit Vehicle Subsystem**

Physical Architecture Flow Name: vehicle location

Location of vehicle and other vehcile characteristics which are exchanged between vehicle subsystems.

Logical Architecture Reference Flow: transit_user_advisory_information
This data flow is used within the Provide Driver and Traveler Services function. It contains data to be converted into output displays for the transit user and includes the following data item which is defined in its own DDE: advisory_data.

Logical Architecture Reference Flow: transit_user_vehicle_credit_identity
This data flow is sent from the Provide Electronic Payments Services function to the Manage Transit function and contains the credit identity of a transit user on-board a transit vehicle, or a stored credit value. Either data item is obtained by a process within the Provide Electronic Payment Services function as data input from the payment instrument terminator and consists of the following data items each of which is defined in its own DDE: credit_identity + stored_credit.

Logical Architecture Reference Flow: vehicle_location_for_transit
This data flow is sent from the Provide Driver and Traveler Services function to the Manage transit function. It contains the vehicle's location as computed from data input to sensors controlled by the processes that determine vehicle location. This is a low precision data flow that will be refined using data obtained from on-board the transit vehicle. It consists of the following data item which is defined in its own DDE: location_identity.

2.21.4 Subsystem Architecture Flow Diagrams

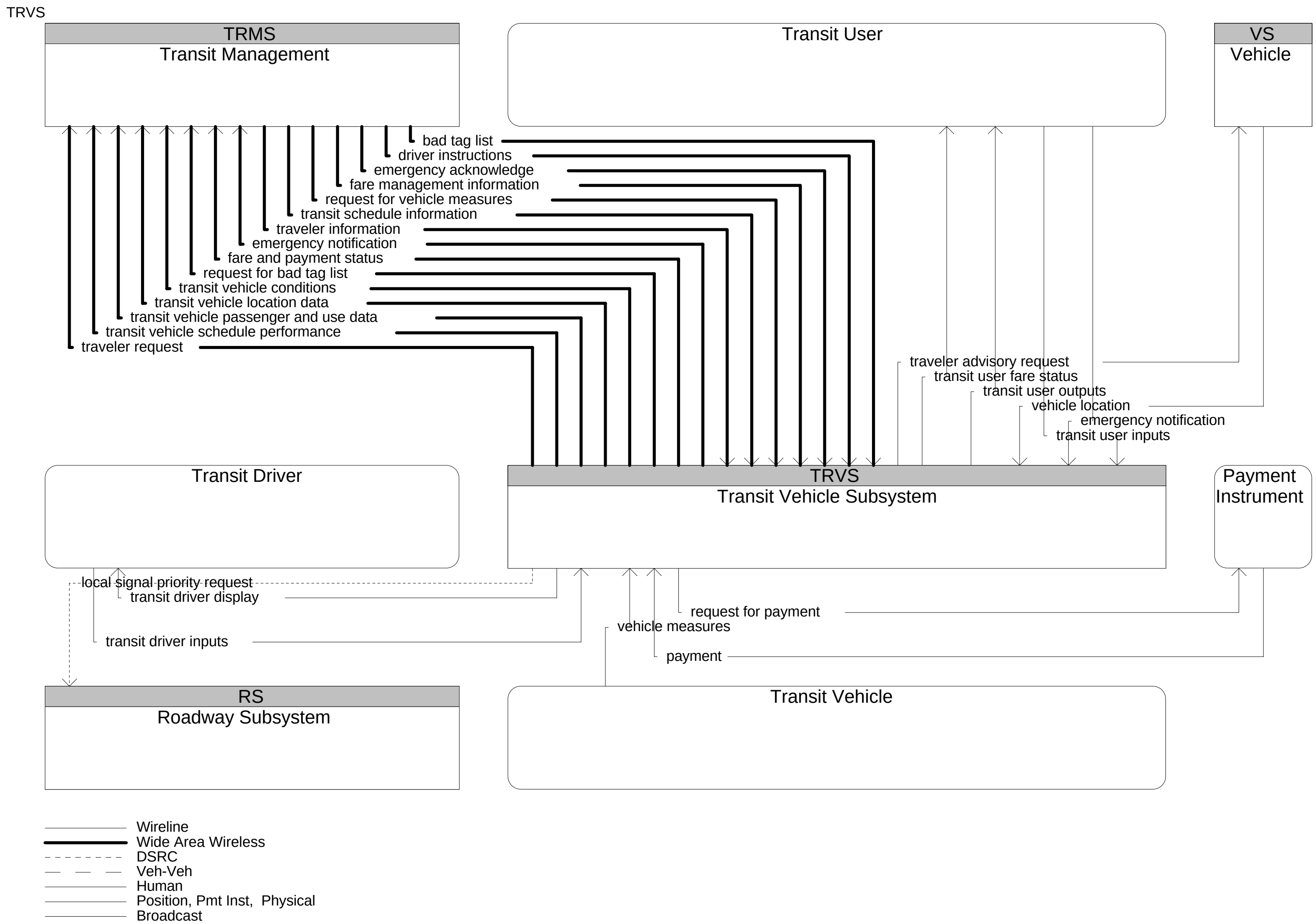


Figure 2.21-2 Architecture Flow Diagram for Transit Vehicle

2.22 Vehicle

This subsystem resides in an automobile and provides the sensory, processing, storage, and communications functions necessary to support efficient, safe, and convenient travel by personal automobile. Information services provide the driver with current travel conditions and the availability of services along the route and at the destination. Both one-way and two-way communications options support a spectrum of information services from low-cost broadcast services to advanced, pay for use personalized information services. Route guidance capabilities assist in formulation of an optimal route and step by step guidance along the travel route. Advanced sensors, processors, enhanced driver interfaces, and actuators complement the driver information services so that, in addition to making informed mode and route selections, the driver travels these routes in a safer and more consistent manner. Initial collision avoidance functions provide "vigilant co-pilot" driver warning capabilities. More advanced functions assume limited control of the vehicle to maintain safe headway. Ultimately, this subsystem supports completely automated vehicle operation through advanced communications with other vehicles in the vicinity and in coordination with supporting infrastructure subsystems. Pre-crash safety systems are deployed and emergency notification messages are issued when unavoidable collisions do occur.

2.22.1 Alternative Configurations

The vehicle subsystem supports not only the driver's requirements for route guidance, traveler information but also the vehicle requirements for interfacing with roadside equipment and other vehicles Figure 2.22-1. The Vehicle subsystem represents the likely subsystems which will appear in private vehicles as well as common functions which will likely appear in transit vehicles, CVO and Emergency vehicles.

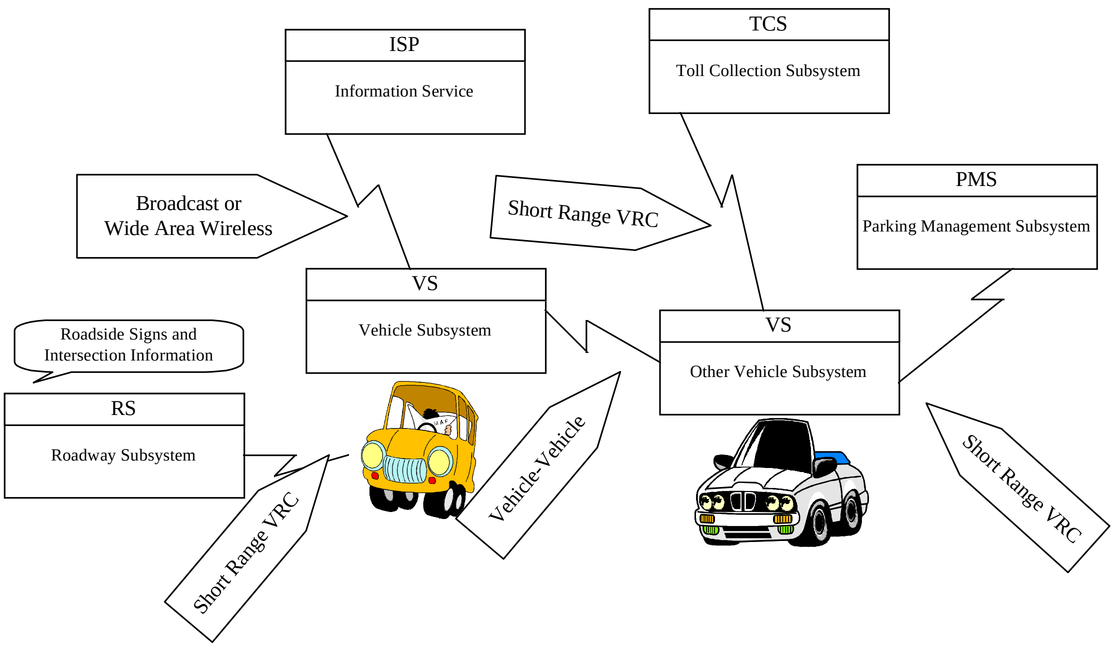


Figure 2.22-1 Alternative Configurations for the Vehicle Subsystem

2.22.2 Subsystem Equipment Packages and Supporting Process Specifications for VS

Basic Vehicle Reception

This Equipment package shall provide the capability for drivers to interface with the ISP Subsystem Basic Information Broadcast Equipment package and receive formatted traffic advisories including accurate traveling information concerning available travel options and their availability, and congestion information in their vehicle. These capabilities shall be based upon the reception of infrastructure information using in-vehicle devices such as an in-vehicle AM/FM radio with data subcarrier connected with the existing audio system and a dash-mounted LCD.

Process Specifications

6.2.2 Prepare and Output In-vehicle Displays

Overview: This process shall provide in-vehicle advisory and broadcast data for output to drivers and transit users. The process shall format requests from users for advisory data and output the requests to other processes. The request for advisory data shall allow the user to request only information relevant to the location of the vehicle. The request may be repeated, periodically, or when the vehicle changes location by a distance determined by the implementation. Data broadcast to the driver shall include traffic related data(incidents, link data and in-vehicle signage), as well as data from the vehicle itself. This vehicle data includes vehicle conditions, smart probe data, safety and position warnings, and enhanced vision images. Safety and warning messages shall be prioritized by the process to supersede advisory and broadcast messages. The process shall also support the transfer of reservation requests from the users in vehicles for other services such as yellow pages.

6.2.5 Provide Driver Interface

Overview: This process shall provide a user interface for a driver through which traffic and travel advisory information can be obtained. The process shall enable traffic and travel advisory information to be requested and output to the driver, and shall also support the automatic output of wide area broadcast information (including in vehicle signage) to the driver. The process shall support output of safety and vision enhancement information to the user. When constructing all outputs the process shall use the vehicle_display_definitions_data store parameters. One purpose of the vehicle_display_definitions_data store is to provide a translation table for road sign and message templates used for in-vehicle display. Part of the input interface provided by the process shall enable the driver to invoke and cancel automatic control of the vehicle including the use of automated highway system (ahs) lanes. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. Visual output may be either in hardcopy, or as a display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

Driver Safety Monitoring System

This Equipment package shall provide the capability todetermine the driver’s condition and warn the driver of potential dangers. This Equipment package includes driver sensors to assess the suitability of the driver (e.g., fitness and alertness) to assume manual control of the vehicle.

Process Specifications

3.1.2 Carry-out Safety Analysis

Overview: This process shall be responsible for producing safety warnings for display to the driver and output to the vehicle control processes. The process shall base its output on input from another process in the vehicle that is analyzing inputs to sensors. When data about a safety situation is received, the process shall output the appropriate messages to another process in the vehicle to warn the driver. If the vehicle is so equipped, the process shall send data to the process in the vehicle responsible for its control.

3.1.3 Process Vehicle On-board Data

Overview: This process shall be responsible for processing data received as input to sensors located on-board a vehicle. The process shall continuously analyze these inputs and produce data from which safety and/or position warnings and actions can be produced by another process. It shall also analyze the data to check for hazardous roadside conditions such as flooding, ice, snow, etc. and if detected shall output this data to processes in the Manage Traffic function.

6.2.5 Provide Driver Interface

Overview: This process shall provide a user interface for a driver through which traffic and travel advisory information can be obtained. The process shall enable traffic and travel advisory information to be requested and output to the driver, and shall also support the automatic output of wide area broadcast information (including in vehicle signage) to the driver. The process shall support output of safety and vision enhancement information to the user. When constructing all outputs the process shall use the vehicle_display_definitions_data store parameters. One purpose of the vehicle_display_definitions_data store is to provide a translation table for road sign and message templates used for in-vehicle display. Part of the input interface provided by the process shall enable the driver to invoke and cancel automatic control of the vehicle including the use of automated highway system (ahs) lanes. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. Visual output may be either in hardcopy, or as a display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

Driver Visibility Improvement System

The Equipment package shall provide the capability to augment the vehicle operator’s ability to see objects in the vehicle path in conditions where driving visibility is poor (e.g., bad weather, night driving, etc.). These capabilities shall be provided using equipment such as on-board sensor system (e.g., an infrared sensor system) to create images that in turn could be relayed to the driver using a heads-up display. The on-board systems to implement this Equipment package shall include a local sensor system, an image creation and processing capability, and a visual display to the driver.

Process Specifications

3.4 Enhance Driver's Vision

Overview: This process shall be responsible for providing data from which a continuously updated display showing an enhanced version of the driver's vision. The process shall produce the data for this display using inputs to sensors mounted on the vehicle. It shall operate at all times and shall send its output to another process for integration with other messages for the driver.

6.2.5 Provide Driver Interface

Overview: This process shall provide a user interface for a driver through which traffic and travel advisory information can be obtained. The process shall enable traffic and travel advisory information to be requested and output to the driver, and shall also support the automatic output of wide area broadcast information (including in vehicle signage) to the driver. The process shall support output of safety and vision enhancement information to the user. When constructing all outputs the process shall use the vehicle_display_definitions_data store parameters. One purpose of the vehicle_display_definitions_data store is to provide a translation table for road sign and message templates used for in-vehicle display. Part of the input interface provided by the process shall enable the driver to invoke and cancel automatic control of the vehicle including the use of automated highway system (ahs) lanes. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. Visual output may be either in hardcopy, or as a display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

In-Vehicle Signing System

This Equipment package shall provide the capability to assist individuals with impaired vision, individuals needing local guidance in areas that the driver is unfamiliar, and implemented in a manner that augments existing signs. This package shall also provide the capability to customize warnings, utilize data from roadside environmental sensors, and provide travelers with information on road conditions and with precautionary reminder messages. These capabilities shall be provided through the use of equipment such as an interface to active tag reader and processor to display the information from the active tag.

Process Specifications

6.2.2 Prepare and Output In-vehicle Displays

Overview: This process shall provide in-vehicle advisory and broadcast data for output to drivers and transit users. The process shall format requests from users for advisory data and output the requests to other processes. The request for advisory data shall allow the user to request only information relevant to the location of the vehicle. The request may be repeated, periodically, or when the vehicle changes location by a distance determined by the implementation. Data broadcast to the driver shall include traffic related data(incidents, link data and in-vehicle signage), as well as data from the vehicle itself. This vehicle data includes vehicle conditions, smart probe data, safety and position warnings, and enhanced vision images. Safety and warning messages shall be prioritized by the process to supersede advisory and broadcast messages. The process shall also support the transfer of reservation requests from the users in vehicles for other services such as yellow pages.

6.2.5 Provide Driver Interface

Overview: This process shall provide a user interface for a driver through which traffic and travel advisory information can be obtained. The process shall enable traffic and travel advisory information to be requested and output to the driver, and shall also support the automatic output of wide area broadcast information (including in vehicle signage) to the driver. The process shall support output of safety and vision enhancement information to the user. When constructing all outputs the process shall use the vehicle_display_definitions_data store parameters. One purpose of the vehicle_display_definitions_data store is to provide a translation table for road sign and message templates used for in-vehicle display. Part of the input interface provided by the process shall

enable the driver to invoke and cancel automatic control of the vehicle including the use of automated highway system (ahs) lanes. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. Visual output may be either in hardcopy, or as a display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

Interactive Vehicle Reception

This Equipment package shall provide the capability for drivers to interface with the ISP Subsystem Infrastructure Equipment packages including the Interactive Infrastructure Information Equipment package, the Infrastructure Provided Route Selection, Yellow Pages and Reservation, and Dynamic Ridesharing Equipment packages. These capabilities shall be provided using the Vehicle Subsystem equipment.

Process Specifications

6.2.2 Prepare and Output In-vehicle Displays

Overview: This process shall provide in-vehicle advisory and broadcast data for output to drivers and transit users. The process shall format requests from users for advisory data and output the requests to other processes. The request for advisory data shall allow the user to request only information relevant to the location of the vehicle. The request may be repeated, periodically, or when the vehicle changes location by a distance determined by the implementation. Data broadcast to the driver shall include traffic related data(incidents, link data and in-vehicle signage), as well as data from the vehicle itself. This vehicle data includes vehicle conditions, smart probe data, safety and position warnings, and enhanced vision images. Safety and warning messages shall be prioritized by the process to supersede advisory and broadcast messages. The process shall also support the transfer of reservation requests from the users in vehicles for other services such as yellow pages.

6.2.5 Provide Driver Interface

Overview: This process shall provide a user interface for a driver through which traffic and travel advisory information can be obtained. The process shall enable traffic and travel advisory information to be requested and output to the driver, and shall also support the automatic output of wide area broadcast information (including in vehicle signage) to the driver. The process shall support output of safety and vision enhancement information to the user. When constructing all outputs the process shall use the vehicle_display_definitions_data store parameters. One purpose of the vehicle_display_definitions_data store is to provide a translation table for road sign and message templates used for in-vehicle display. Part of the input interface provided by the process shall enable the driver to invoke and cancel automatic control of the vehicle including the use of automated highway system (ahs) lanes. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. Visual output may be either in hardcopy, or as a display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

Smart Probe

Vehicle Probes with added capability and intelligence to sense and send road conditions as the vehicle travels. The same vehicle equipment that improves stability in adverse conditions and provides driver information is a potential source for this information. Smart probe data may include road surface conditions and weather information.

Process Specifications

3.1.3 Process Vehicle On-board Data

Overview: This process shall be responsible for processing data received as input to sensors located on-board a vehicle. The process shall continuously analyze these inputs and produce data from which safety and/or position warnings and actions can be produced by another process. It shall also analyze the data to check for hazardous roadside conditions such as flooding, ice, snow, etc. and if detected shall output this data to processes in the Manage Traffic function.

Vehicle Autonomous Route Guidance

This Equipment package provides route planning and turn by turn route guidance. It provides autonomous route guidance in the absence of real-time information or factors information provided by the infrastructure into its route selection and guidance algorithms. The equipment package also includes those truly autonomous systems that are not configured to receive or process any external data.

Process Specifications

6.7.2.1.3 Provide Autonomous In-vehicle Guidance

Overview: This process shall provide autonomous in-vehicle guidance. It shall calculate the route using data obtained from an in-vehicle navigable map database which can be supplemented with link queue and travel time data obtained from a central source, if specified by the driver and available. The process shall provide guidance in the form of actual driving instructions, e.g. turn left at the next intersection, take the right lane, etc. When link queue and travel time data are being used, the process shall provide guidance for the best route for current traffic conditions, within the preferences and constraints specified by the driver in the guidance request.

6.7.2.2 Process Vehicle Location Data

Overview: This process shall provide the vehicle's current location. It shall calculate the location from one or more sources of position data such as GPS, DGPS, odometer and differential odometers, and shall refine its calculations using techniques such as map matching, etc. Location data (intended for use by in-vehicle navigation, guidance systems, and any emergency notification systems) should be provided by the process in a manner that is as precise as is practical within cost and technology constraints. Location data intended for transit vehicles and driver advisories may be less precise.

6.7.2.3 Provide Driver Guidance Interface

Overview: This process shall provide a user interface for the vehicle's driver through which route guidance is provided. Three types of route guidance provided by other processes shall be supported by this process(dynamic infrastructure based, autonomous with infrastructure data update, and autonomous). The process shall enable input by the driver of the type of guidance required, the data from which the route is to be determined and output of the resulting route. The process shall not provide on-line guidance until the route has been accepted by the driver. For those forms of guidance that require an on-board map database, the process shall provide an interface through which the driver may obtain and pay for an initial copy of the database plus updates when needed. The process shall support inputs from the driver in either manual or audio form, and shall provide its outputs in audible or visual forms. It shall enable the visual output to be either in hardcopy, and/or display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

6.7.2.4 Update Vehicle Navigable Map Database

Overview: This process shall update the vehicle's navigable database based on digitized data obtained from a map provider, or other appropriate data source. The update shall be initiated by the driver through another process .The process shall have the capability to allow a financial transaction (to pay for the update) to be successfully completed using processes in the Provide Electronic Payment Services function. When the new map data is received, it shall be loaded by the process into the vehicle_map_database data store for use by other processes. The result of the update request (successful or not) shall be sent back to the driver interface process for output to the driver.

Vehicle Intersection Collision Warning

This Equipment package shall provide the capability for the detection of an impending collision with a moving or stationary object prior to crash impact in an intersection with notification provided to the driver of the presence of potentially hazardous situations and need for immediate collision avoidance action. These capabilities shall be provided through the use of equipment such as an intersection hazard warning sensor and actuator.

Process Specifications

3.1.1 Produce Collision and Crash Avoidance Data

Overview: This process shall be responsible for sensing and evaluating the likelihood of a collision between two vehicles or a vehicle and a stationary object. The process shall base its detection on input from two other processes. One of these processes shall be that which continuously processes sensor inputs on-board the vehicle and the second shall be that which detects collision situations at intersections. When either event is detected this process shall output the appropriate messages to another process in the vehicle to warn the driver. If the vehicle is suitably equipped, the process shall initiate the deployment of crash restraint devices in advance of the collision and/or generate data to initiate direct operation of the vehicle to take evasive maneuvers.

3.1.3 Process Vehicle On-board Data

Overview: This process shall be responsible for processing data received as input to sensors located on-board a vehicle. The process shall continuously analyze these inputs and produce data from which safety and/or position warnings and actions can be produced by another process. It shall also analyze the data to check for hazardous roadside conditions such as flooding, ice, snow, etc. and if detected shall output this data to processes in the Manage Traffic function.

3.2.3.5 Process Vehicle Sensor Data

Overview: This process shall be responsible for providing the facility to decode the input being sent to on-board vehicle sensors. The process shall support inputs to those sensors that monitor conditions both on-board the vehicle and in the way the vehicle relates to its surroundings. The data produced by the process shall be sent to another process which shall determine if any action is required.

6.2.5 Provide Driver Interface

Overview: This process shall provide a user interface for a driver through which traffic and travel advisory information can be obtained. The process shall enable traffic and travel advisory information to be requested and output to the driver, and shall also support the automatic output of wide area broadcast information (including in vehicle signage) to the driver. The process shall support output of safety and vision enhancement information to the user. When constructing all outputs the process shall use the vehicle_display_definitions_data store parameters. One purpose of the vehicle_display_definitions_data store is to provide a translation table for road sign and message templates used for in-vehicle display. Part of the input interface provided by the process shall enable the driver to invoke and cancel automatic control of the vehicle including the use of automated highway system (ahs) lanes. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. Visual output may be either in hardcopy, or as a display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

Vehicle Intersection Control

This Equipment package shall provide the capability for the detection of an impending collision with a moving or stationary object prior to crash impact in an intersection and automatically avoid the intersection collision. These capabilities shall be provided through the use of equipment such as an intersection hazard warning sensor and actuator.

Process Specifications

3.1.1 Produce Collision and Crash Avoidance Data

Overview: This process shall be responsible for sensing and evaluating the likelihood of a collision between two vehicles or a vehicle and a stationary object. The process shall base its detection on input from two other processes. One of these processes shall be that which continuously processes sensor inputs on-board the vehicle and the second shall be that which detects collision situations at intersections. When either event is detected this process shall output the appropriate messages to another process in the vehicle to warn the driver. If the vehicle is suitably equipped, the process shall initiate the deployment of crash restraint devices in advance of the collision and/or generate data to initiate direct operation of the vehicle to take evasive maneuvers.

3.1.3 Process Vehicle On-board Data

Overview: This process shall be responsible for processing data received as input to sensors located on-board a vehicle. The process shall continuously analyze these inputs and produce data from which safety and/or position warnings and actions can be produced by another process. It shall also analyze the data to check for hazardous roadside conditions such as flooding, ice, snow, etc. and if detected shall output this data to processes in the Manage Traffic function.

6.2.5 Provide Driver Interface

Overview: This process shall provide a user interface for a driver through which traffic and travel advisory information can be obtained. The process shall enable traffic and travel advisory information to be requested and output to the driver, and shall also support the automatic output of wide area broadcast information (including in vehicle signage) to the driver. The process shall support output of safety and vision enhancement information to the user. When constructing all outputs the process shall use the vehicle_display_definitions_data store parameters. One purpose of the vehicle_display_definitions_data store is to provide a translation table for road sign and message templates used for in-vehicle display. Part of the input interface provided by the process shall enable the driver to invoke and cancel automatic control of the vehicle including the use of automated highway system (ahs) lanes. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. Visual output may be either in hardcopy, or as a display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

Vehicle Lateral Control

This Equipment package shall provide the capability for lateral control of a vehicle on roads to allow "hands off" driving, automating the steering control function. This capability shall be provided through the use of equipment provided to detect lanes, obstacles or vehicles to the sides of the vehicle. This sensor information shall be processed on board the vehicle, and appropriate steering control actions shall be initiated using steering actuators. Appropriate lane maintenance may thus be maintained automatically.

Process Specifications

3.2.1 Provide Driver Interface

Overview: This process shall be responsible for providing an interface through which a vehicle driver can initiate, monitor and terminate automatic control of the vehicle. The output that any of these actions generates in terms of messages to the driver shall be sent by this process to another process that is in the Provide Driver and Traveler Services function and in the vehicle. The driver inputs shall be received by this process from another process that is also in the Provide Driver and Traveler Services function and in the vehicle.

3.2.3.1 Provide Command Interface

Overview: This process shall be responsible for providing the interface through which all driver commands are passed to the correct processes in the vehicle for action. The process shall also pass all messages about vehicle control status on to another process in the vehicle for output to the driver. It shall also monitor the health of the other in-vehicle processes involved in automatic vehicle control. This process shall take the appropriate mode canceling action when any failures are detected in these processes.

3.2.3.3 Process data for Vehicle Actuators

Overview: This process shall be responsible for providing the interface between other automatic vehicle control process and the actuators which actually change the vehicle's controls. The process shall both implement commands and monitor the operation of the actuators to check that they only move when requested. If they move for any other reason, e.g. the driver has touched the vehicle controls, the process shall disable automatic operation. The process shall perform its own built-in self test (BIST) analysis. It shall report any errors that this shows to another process in the vehicle and shall cease to accept further requests to change the vehicle's actuators.

3.2.3.4.3 Provide Lane Servo Control

Overview: This process shall be responsible for providing the data which enables the vehicle's steering to be adjusted so that it maintains a position that is in the middle of its current lane. The process shall enable this to be temporarily overridden as a result of action being taken by other processes to change lanes. The process shall perform its own built-in self test (BIST) analysis. It shall report any errors that this shows to another process in the vehicle and shall cease to accept further requests to change the vehicle's throttle position.

3.2.3.4.4 Provide Change Lane Servo Control

Overview: This process shall be responsible for providing the data which enables the vehicle's steering to be adjusted so that it will move either left or right from one lane to another. The process shall enable this to temporarily override the lane center holding facility available from another process in the vehicle. The process shall perform its own built-in self test (BIST) analysis. It shall report any errors that this shows to another process in the vehicle and shall cease to accept further requests to change the vehicle's throttle position.

3.2.3.4.5 Provide Vehicle Control Data Interface

Overview: This process shall be responsible for providing a communications and data processing interface between processes in the Provide Vehicle Control and Monitoring function. These processes shall comprise those responsible for controlling individual functions, e.g. throttle, brake, etc., and those that interface to actuators and those that monitor vehicle operation.

Vehicle Lateral Warning System

This Equipment package allows for lateral warning. It utilizes safety sensors and collision sensors . It requires on-board sensors to monitor the areas to the sides of the vehicle and present warnings to the driver about potential hazards.

Process Specifications

3.1.1 Produce Collision and Crash Avoidance Data

Overview: This process shall be responsible for sensing and evaluating the likelihood of a collision between two vehicles or a vehicle and a stationary object. The process shall base its detection on input from two other processes. One of these processes shall be that which continuously processes sensor inputs on-board the vehicle and the second shall be that which detects collision situations at intersections. When either event is detected this process shall output the appropriate messages to another process in the vehicle to warn the driver. If the vehicle is suitably equipped, the process shall initiate the deployment of crash restraint devices in advance of the collision and/or generate data to initiate direct operation of the vehicle to take evasive maneuvers.

3.1.3 Process Vehicle On-board Data

Overview: This process shall be responsible for processing data received as input to sensors located on-board a vehicle. The process shall continuously analyze these inputs and produce data from which safety and/or position warnings and actions can be produced by another process. It shall also analyze the data to check for hazardous roadside conditions such as flooding, ice, snow, etc. and if detected shall output this data to processes in the Manage Traffic function.

3.2.3.5 Process Vehicle Sensor Data

Overview: This process shall be responsible for providing the facility to decode the input being sent to on-board vehicle sensors. The process shall support inputs to those sensors that monitor conditions both on-board the vehicle and in the way the vehicle relates to its surroundings. The data produced by the process shall be sent to another process which shall determine if any action is required.

3.2.3.5 Process Vehicle Sensor Data

Overview: This process shall be responsible for providing the facility to decode the input being sent to on-board vehicle sensors. The process shall support inputs to those sensors that monitor conditions both on-board the vehicle and in the way the vehicle relates to its surroundings. The data produced by the process shall be sent to another process which shall determine if any action is required.

6.2.5 Provide Driver Interface

Overview: This process shall provide a user interface for a driver through which traffic and travel advisory information can be obtained. The process shall enable traffic and travel advisory information to be requested and output to the driver, and shall also support the automatic output of wide area broadcast information (including in vehicle signage) to the driver. The process shall support output of safety and vision enhancement information to the user. When constructing all outputs the process shall use the vehicle_display_definitions_data store parameters. One purpose of the vehicle_display_definitions_data store is to provide a translation table for road sign and message templates used for in-vehicle display. Part of the input interface provided by the process shall enable the driver to invoke and cancel automatic control of the vehicle including the use of automated highway system (ahs) lanes. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. Visual output may be either in hardcopy, or as a display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

Vehicle Longitudinal Control
This Equipment package shall provide the capability for longitudinal control of a vehicle on roads to allow "feet off" driving, automating the function of speed control, acceleration, and braking. This capability shall be provided through the use of equipment to detect obstacles or vehicles in the longitudinal path of the vehicle. This sensor information shall be processed on board the vehicle, and appropriate control actions (acceleration, braking, or maintaining speed) shall be initiated using accelerator and/or brake actuators. Appropriate following distances may thus be maintained automatically.

Process Specifications
3.1.3 Process Vehicle On-board Data

Overview: This process shall be responsible for processing data received as input to sensors located on-board a vehicle. The process shall continuously analyze these inputs and produce data from which safety and/or position warnings and actions can be produced by another process. It shall also analyze the data to check for hazardous roadside conditions such as flooding, ice, snow, etc. and if detected shall output this data to processes in the Manage Traffic function.

3.2.1 Provide Driver Interface

Overview: This process shall be responsible for providing an interface through which a vehicle driver can initiate, monitor and terminate automatic control of the vehicle. The output that any of these actions generates in terms of messages to the driver shall be sent by this process to another process that is in the Provide Driver and Traveler Services function and in the vehicle. The driver inputs shall be received by this process from another process that is also in the Provide Driver and Traveler Services function and in the vehicle.

3.2.3.1 Provide Command Interface

Overview: This process shall be responsible for providing the interface through which all driver commands are passed to the correct processes in the vehicle for action. The process shall also pass all messages about vehicle control status on to another process in the vehicle for output to the driver. It shall also monitor the health of the other in-vehicle processes involved in automatic vehicle control. This process shall take the appropriate mode canceling action when any failures are detected in these processes.

3.2.3.3 Process data for Vehicle Actuators

Overview: This process shall be responsible for providing the interface between other automatic vehicle control process and the actuators which actually change the vehicle's controls. The process shall both implement commands and monitor the operation of the actuators to check that they only move when requested. If they move for any other reason, e.g. the driver has touched the vehicle controls, the process shall disable automatic operation. The process shall perform its own built-in self test (BIST) analysis. It shall report any errors that this shows to another process in the vehicle and shall cease to accept further requests to change the vehicle's actuators.

3.2.3.4.1 Provide Speed Servo Control

Overview: This process shall be responsible for providing data which enables the vehicle's throttle to be regulated in such a way that a desired vehicle speed is maintained. The process shall enable the throttle to be overridden temporarily in order to maintain a desired headway between the vehicle and others in a platoon. The data that actually changes the throttle's position shall be sent to the process that provides data to in-vehicle actuators. The process shall perform its own built-in self test (BIST) analysis. It shall report any errors that this shows to another process in the vehicle and shall cease to accept further requests to change the vehicle's throttle position.

3.2.3.4.2 Provide Headway Servo Control

Overview: This process shall be responsible for providing data which enables the vehicle's brake and throttle to be regulated in such a way that its headway, i.e. the distance between it and the vehicle in front, is maintained. The process shall support the brake movements that either maintain the vehicle's headway for normal operation, or hold it at the value used in platoon following, whether on or off automated highway system (ahs) lanes. The process shall perform its own built-in self test (BIST) analysis. It shall report any errors that this shows to another process in the vehicle and shall cease to accept further requests to change the vehicle's brake setting.

3.2.3.4.5 Provide Vehicle Control Data Interface

Overview: This process shall be responsible for providing a communications and data processing interface between processes in the Provide Vehicle Control and Monitoring function. These processes shall comprise those responsible for controlling individual functions, e.g. throttle, brake, etc., and those that interface to actuators and those that monitor vehicle operation.

6.2.5 Provide Driver Interface

Overview: This process shall provide a user interface for a driver through which traffic and travel advisory information can be obtained. The process shall enable traffic and travel advisory information to be requested and output to the driver, and shall also support the automatic output of wide area broadcast information (including in vehicle signage) to the driver. The process shall support output of safety and vision enhancement information to the user. When constructing all outputs the process shall use the vehicle_display_definitions_data store parameters. One purpose of the vehicle_display_definitions_data store is to provide a translation table for road sign and message templates used for in-vehicle display. Part of the input interface provided by the process shall enable the driver to invoke and cancel automatic control of the vehicle including the use of automated highway system (ahs) lanes. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. Visual output may be either in hardcopy, or as a display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

Vehicle Longitudinal Warning System

This Equipment package allows for longitudinal warning. It utilizes safety sensors and collision sensors. It requires on-board sensors to monitor the areas in front of and behind the vehicle and present warnings to the driver about potential hazards.

Process Specifications

3.1.1 Produce Collision and Crash Avoidance Data

Overview: This process shall be responsible for sensing and evaluating the likelihood of a collision between two vehicles or a vehicle and a stationary object. The process shall base its detection on input from two other processes. One of these processes shall be that which continuously processes sensor inputs on-board the vehicle and the second shall be that which detects collision situations at intersections. When either event is detected this process shall output the appropriate messages to another process in the vehicle to warn the driver. If the vehicle is suitably equipped, the process shall initiate the deployment of crash restraint devices in advance of the collision and/or generate data to initiate direct operation of the vehicle to take evasive maneuvers.

3.1.3 Process Vehicle On-board Data

Overview: This process shall be responsible for processing data received as input to sensors located on-board a vehicle. The process shall continuously analyze these inputs and produce data from which safety and/or position warnings and actions can be produced by another process. It shall also analyze the data to check for hazardous roadside conditions such as flooding, ice, snow, etc. and if detected shall output this data to processes in the Manage Traffic function.

3.2.3.5 Process Vehicle Sensor Data

Overview: This process shall be responsible for providing the facility to decode the input being sent to on-board vehicle sensors. The process shall support inputs to those sensors that monitor conditions both on-board the vehicle and in the way the vehicle relates to its surroundings. The data produced by the process shall be sent to another process which shall determine if any action is required.

3.2.3.5 Process Vehicle Sensor Data

Overview: This process shall be responsible for providing the facility to decode the input being sent to on-board vehicle sensors. The process shall support inputs to those sensors that monitor conditions both on-board the vehicle and in the way the vehicle relates to its surroundings. The data produced by the process shall be sent to another process which shall determine if any action is required.

6.2.5 Provide Driver Interface

Overview: This process shall provide a user interface for a driver through which traffic and travel advisory information can be obtained. The process shall enable traffic and travel advisory information to be requested and output to the driver, and shall also support the automatic output of wide area broadcast information (including in vehicle signage) to the driver. The process shall support output of safety and vision enhancement information to the user. When constructing all outputs the process shall use the vehicle_display_definitions_data store parameters. One purpose of the vehicle_display_definitions_data store is to provide a translation table for road sign and message templates used for in-vehicle display. Part of the input interface provided by the process shall enable the driver to invoke and cancel automatic control of the vehicle including the use of automated highway system (ahs) lanes. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. Visual output may be either in hardcopy, or as a display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

Vehicle Mayday I/F

This Equipment package shall provide the capability for an in-vehicle manually initiated distress signal with cancel a prior issued manual request for help feature. This capability shall include automatically identifying that a collision had occurred using equipment such as collision detection sensors with interface to mayday type equipment that would automatically detect vehicle problems and for some cases, automatically send appropriate distress signals to the Emergency Management Subsystem.

Process Specifications

3.3.2 Provide Communications Function

Overview: This process shall be responsible for sending messages it receives from other processes in this facility to the Manage Emergency Services function. It shall also be responsible for passing on the resulting response to the driver via processes in the Provide Driver and Traveler Services function.

3.3.3 Build Automatic Collision Notification Message

Overview: This process shall be responsible for preparing and submitting data for transmission to the Manage Emergency Services function. The data shall be sent by this process when an emergency situation is detected by analyzing inputs from the vehicle. This process shall produce its outputs regardless of any action by the driver and shall be designed to be as the result of a crash which may have prevented the driver from initiating the emergency request personally.

6.2.5 Provide Driver Interface

Overview: This process shall provide a user interface for a driver through which traffic and travel advisory information can be obtained. The process shall enable traffic and travel advisory information to be requested and output to the driver, and shall also support the automatic output of wide area broadcast information (including in vehicle signage) to the driver. The process shall support output of safety and vision enhancement information to the user. When constructing all outputs the process shall use the vehicle_display_definitions_data store parameters. One purpose of the vehicle_display_definitions_data store is to provide a translation table for road sign and message templates used for in-vehicle display. Part of the input interface provided by the process shall enable the driver to invoke and cancel automatic control of the vehicle including the use of automated highway system (ahs) lanes. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. Visual output may be either in hardcopy, or as a display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

6.7.1.1 Build Driver Personal Security Message

Overview: This process shall respond to the input of a request from a driver for action by the emergency services. Input of the request shall be received by the process from the driver via a panic button or some other functionally similar form of input device provided as part of the in-vehicle equipment. When the input is received, the process shall send a message to the communications process, containing the vehicle's current location, its identity and basic vehicle data relevant to its current condition, as well as any other data, such as personal medical history, vehicle orientation, etc., that may be developed in-vehicle by other systems.

6.7.1.2 Provide Driver In-vehicle Communications Function

Overview: This process shall prepare and send an emergency message from a driver to the Manage Emergency Services function. The message shall only be sent by the process in response to data received from another process that monitors driver inputs. Once an emergency message has been sent, the process shall send a message to that effect to another process in the Provide Vehicle Monitoring and Control function for output to the driver. The process shall then await a response from the Manage Emergency Services function, and then send a detailed message to the other process for output to the driver. Output of the emergency message to the Manage Emergency Services function shall be repeated by the process at regular intervals until a response is received.

Vehicle Pre-Crash Safety Systems

This Equipment package provides the capability to sense local conditions, determine collision probability, and deploy a pre-crash safety system. These capabilities shall be provided by equipment such as on-board sensors to determine the location or distance away and closing rates of neighboring vehicles or other roadway obstacles. These detection systems shall be supplemented by additional sensors for existing weather and roadway conditions and roadway geometry. A processor in the vehicle shall assimilate this information and determine the probability of a collision with the other vehicle or obstacle. If the collision probability is high, it shall deploy a pre-crash safety system either to avoid the accident or to reduce the accident severity.

Process Specifications

3.1.1 Produce Collision and Crash Avoidance Data

Overview: This process shall be responsible for sensing and evaluating the likelihood of a collision between two vehicles or a vehicle and a stationary object. The process shall base its detection on input from two other processes. One of these processes shall be that which continuously processes sensor inputs on-board the vehicle and the second shall be that which detects collision situations at intersections. When either event is detected this process shall output the appropriate messages to another process in the vehicle to warn the driver. If the vehicle is suitably equipped, the process shall initiate the deployment of crash restraint devices in advance of the collision and/or generate data to initiate direct operation of the vehicle to take evasive maneuvers.

3.1.3 Process Vehicle On-board Data

Overview: This process shall be responsible for processing data received as input to sensors located on-board a vehicle. The process shall continuously analyze these inputs and produce data from which safety and/or position warnings and actions can be produced by another process. It shall also analyze the data to check for hazardous roadside conditions such as flooding, ice, snow, etc. and if detected shall output this data to processes in the Manage Traffic function.

3.2.3.5 Process Vehicle Sensor Data

Overview: This process shall be responsible for providing the facility to decode the input being sent to on-board vehicle sensors. The process shall support inputs to those sensors that monitor conditions both on-board the vehicle and in the way the vehicle relates to its surroundings. The data produced by the process shall be sent to another process which shall determine if any action is required.

Vehicle Probe Support

This Equipment package includes capabilities for the probe vehicle to identify its location, measure traffic conditions such as link travel time and speed and possibly environmental hazards such as icy road conditions, and transmit these data to either the ISP or TMC.

Process Specifications

3.2.3.5 Process Vehicle Sensor Data

Overview: This process shall be responsible for providing the facility to decode the input being sent to on-board vehicle sensors. The process shall support inputs to those sensors that monitor conditions both on-board the vehicle and in the way the vehicle relates to its surroundings. The data produced by the process shall be sent to another process which shall determine if any action is required.

6.7.2.1.2 Provide Dynamic In-vehicle Guidance

Overview: This process shall enable dynamic vehicle route guidance data to be calculated. The process shall perform the same dynamic vehicle route guidance services for vehicles that are under automatic control using automatic highway system (ahs)lanes. When providing dynamic guidance, the process provides vehicle travel times as probe data to another process in the Provide Driver and Traveler Services function. The process shall base its guidance request on data input by the driver through another process, and on the vehicle's current location as provided by another process.

6.7.2.2 Process Vehicle Location Data

Overview: This process shall provide the vehicle's current location. It shall calculate the location from one or more sources of position data such as GPS, DGPS, odometer and differential odometers, and shall refine its calculations using techniques such as map matching, etc. Location data (intended for use by in-vehicle navigation, guidance systems, and any emergency notification systems) should be provided by the process in a manner that is as precise as is practical within cost and technology constraints. Location data intended for transit vehicles and driver advisories may be less precise.

Vehicle Provider-Based Route Guidance

This Equipment package coordinates with an ISP-Based route planning service to select a suggested route plan that is tailored to the driver's preferences. Coordination continues during the trip so that the route plan can be modified to account for new information and vehicle probe data can be returned to the ISP. Many equipment configurations are possible including basic systems that provide only a route plan to the driver as well as systems that include the necessary on-board equipment to provide turn by turn route guidance following the selected route.

Process Specifications

6.7.2.1.1 Determine In-vehicle Guidance Method

Overview: This process shall act as the interface for guidance requests received from drivers in vehicles. The process shall select the best method for in-vehicle guidance based on data in the driver's request. Three general methods of route guidance are supported: 1) dynamic (infrastructure based guidance is provided to the vehicle unit), 2) dynamic autonomous (link and queue speed or travel times are obtained from the infrastructure and used by the autonomous in vehicle unit), and autonomous (the in vehicle unit uses only locally available data- there is no information provided by the infrastructure). When dynamic guidance is selected, the vehicle's travel time for each link shall be provided by the process back to a central source of data. If the communications link to the central source fails in either of the modes that use it, the process shall automatically revert to the use of local data only. When the original mode was centralized guidance, the process shall use the last set of guidance data that was received, and if this is not sufficient for the vehicle to reach the requested destination, automatically revert to autonomous guidance using local data only.

6.7.2.1.2 Provide Dynamic In-vehicle Guidance

Overview: This process shall enable dynamic vehicle route guidance data to be calculated. The process shall perform the same dynamic vehicle route guidance services for vehicles that are under automatic control using automatic highway system (ahs)lanes. When providing dynamic guidance, the process provides vehicle travel times as probe data to another process in the Provide Driver and Traveler Services function. The process shall base its guidance request on data input by the driver through another process, and on the vehicle's current location as provided by another process.

6.7.2.2 Process Vehicle Location Data

Overview: This process shall provide the vehicle's current location. It shall calculate the location from one or more sources of position data such as GPS, DGPS, odometer and differential odometers, and shall refine its calculations using techniques such as map matching, etc. Location data (intended for use by in-vehicle navigation, guidance systems, and any emergency notification systems) should be provided by the process in a manner that is as precise as is practical within cost and technology constraints. Location data intended for transit vehicles and driver advisories may be less precise.

6.7.2.3 Provide Driver Guidance Interface

Overview: This process shall provide a user interface for the vehicle's driver through which route guidance is provided. Three types of route guidance provided by other processes shall be supported by this process(dynamic infrastructure based, autonomous with infrastructure data update, and autonomous). The process shall enable input by the driver of the type of guidance required, the data from which the route is to be determined and output of the resulting route. The process shall not provide on-line guidance until the route has been accepted by the driver. For those forms of guidance that require an on-board map database, the process shall provide an interface through which the driver may obtain and pay for an initial copy of the database plus updates when needed. The process shall support inputs from the driver in either manual or audio form, and shall provide its outputs in audible or visual forms. It shall enable the visual output to be either in hardcopy, and/or display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

6.7.2.4 Update Vehicle Navigable Map Database

Overview: This process shall update the vehicle's navigable database based on digitized data obtained from a map provider, or other appropriate data source. The update shall be initiated by the driver through another process .The process shall have the capability to allow a financial transaction (to pay for the update) to be successfully completed using processes in the Provide Electronic Payment Services function. When the new map data is received, it shall be loaded by the process into the vehicle_map_database data store for use by other processes. The result of the update request (successful or not) shall be sent back to the driver interface process for output to the driver.

Vehicle Safety Monitoring System

This Equipment package shall provide the capability to diagnose critical components of the vehicle and warn the driver of potential dangers. These capabilities shall be using equipment such as a set of on-board sensors to monitor continuously the vehicle condition and performance, including steering, braking, acceleration, emissions, fuel economy, engine performance, etc. Problems with any of these systems shall be identified using processors on board the vehicle, providing a timely display to the driver of the situation. The sensors shall provide warnings to the driver in the event of a serious condition (e.g., likely failure or damage).

Process Specifications

3.1.2 Carry-out Safety Analysis

Overview: This process shall be responsible for producing safety warnings for display to the driver and output to the vehicle control processes. The process shall base its output on input from another process in the vehicle that is analyzing inputs to sensors. When data about a safety situation is received, the process shall output the appropriate messages to another process in the vehicle to warn the driver. If the vehicle is so equipped, the process shall send data to the process in the vehicle responsible for its control.

3.1.3 Process Vehicle On-board Data

Overview: This process shall be responsible for processing data received as input to sensors located on-board a vehicle. The process shall continuously analyze these inputs and produce data from which safety and/or position warnings and actions can be produced by another process. It shall also analyze the data to check for hazardous roadside conditions such as flooding, ice, snow, etc. and if detected shall output this data to processes in the Manage Traffic function.

6.2.5 Provide Driver Interface

Overview: This process shall provide a user interface for a driver through which traffic and travel advisory information can be obtained. The process shall enable traffic and travel advisory information to be requested and output to the driver, and shall also support the automatic output of wide area broadcast information (including in vehicle signage) to the driver. The process shall support output of safety and vision enhancement information to the user. When constructing all outputs the process shall use the vehicle_display_definitions_data store parameters. One purpose of the vehicle_display_definitions_data store is to provide a translation table for road sign and message templates used for in-vehicle display. Part of the input interface provided by the process shall enable the driver to invoke and cancel automatic control of the vehicle including the use of automated highway system (ahs) lanes. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. Visual output may be either in hardcopy, or as a display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

Vehicle Systems for AHS

This Equipment package provides the capability for "hands-off" and "feet off" operations of an equipped vehicle on the automated portion of the highway system including the longitudinal control, lateral control for lane change/merge and roadway departure, regulating the vehicle speed and steering control, and sensing impending hazards and responding appropriately. These capabilities shall be provided by systems on board the vehicle to regulate longitudinal and lateral control maneuvers, including acceleration, braking, and steering functions. The capability to control access to the automated highway system shall be provided through an automated check-in procedure in which the vehicle and driver are checked for their fitness.

Process Specifications

3.1.3 Process Vehicle On-board Data

Overview: This process shall be responsible for processing data received as input to sensors located on-board a vehicle. The process shall continuously analyze these inputs and produce data from which safety and/or position warnings and actions can be produced by another process. It shall also analyze the data to check for hazardous roadside conditions such as flooding, ice, snow, etc. and if detected shall output this data to processes in the Manage Traffic function.

3.2.1 Provide Driver Interface

Overview: This process shall be responsible for providing an interface through which a vehicle driver can initiate, monitor and terminate automatic control of the vehicle. The output that any of these actions generates in terms of messages to the driver shall be sent by this process to another process that is in the Provide Driver and Traveler Services function and in the vehicle. The driver inputs shall be received by this process from another process that is also in the Provide Driver and Traveler Services function and in the vehicle.

3.2.2 Provide AHS Control

Overview: This process shall be responsible for providing the facility that enables vehicles to operate in automatic highway system (ahs) lanes. This mode of operation shall only be initiated by the process when a request is received from the driver via other processes in the vehicle. The first action of the process must be to send data to the process that provides the ahs check-in facility. If a positive response is received from that process, i.e. the vehicle's check in is accepted, then the process shall enable ahs operation by sending the data to the vehicle control processes. Once the vehicle is in ahs operation, the process shall continuously monitor for an input from the driver that cancels ahs mode, and when this is received send mode canceling data to the vehicle control processes. Similarly the process shall also continuously monitor input from the process analyzing vehicle condition and the vehicle's presence on an ahs lane. The process shall send mode canceling data to the vehicle control processes, if the condition does not support ahs lane operation, or the vehicle is no longer on an ahs lane.

3.2.3.2 Manage Platoon Following

Overview: This process shall be responsible for providing the facility for the automatic control of vehicles to be extended to cover the platooning of vehicles. The process shall enable vehicles to follow each other very closely (inches apart) in a platoon, responding to changes in speed and direction of the lead vehicle. The process shall monitor data from other vehicles in the platoon received via another process, and shall also send data about itself to the same process for communication to other platoon vehicles. If the data received from the process shows that the vehicle has been left on its own, i.e. there are no other vehicles in front or behind, the process shall send data to another process in the vehicle to increase speed and catch up with any platoon that may be ahead. The process shall only allow the vehicle to join or continue running in a platoon if it and/or the driver are considered to be in a safe condition, using data received from other processes in the vehicle.

3.2.3.3 Process data for Vehicle Actuators

Overview: This process shall be responsible for providing the interface between other automatic vehicle control process and the actuators which actually change the vehicle's controls. The process shall both implement commands and monitor the operation of the actuators to check that they only move when requested. If they move for any other reason, e.g. the driver has touched the vehicle controls, the process shall disable automatic operation. The process shall perform its own built-in self test (BIST) analysis. It shall report any errors that this shows to another process in the vehicle and shall cease to accept further requests to change the vehicle's actuators.

3.2.3.6 Communicate with other Platoon Vehicles

Overview: This process shall be responsible for communicating with the other vehicles that are in a platoon. The process shall support communications with the platoon vehicles that are both immediately in front of and behind the vehicle in which it operates. The passing of data in both directions, i.e. both to and from the vehicles, shall be supported by the process.

3.2.4 Process Sensor Data for AHS input

Overview: This process shall be responsible for analyzing the input from the vehicle that provides information about its condition and that it is on an automatic highway system (ahs) lane. The process shall continuously analyze this data and provide output to the process that provides ahs control.

Vehicle Toll/Parking Interface

This Equipment package shall provide the capability for vehicle operators to pay toll without stopping their vehicles and pay for parking without the use of cash. These capabilities shall be provided through the use of equipment such as an active tag interface and debit/credit card interface.

Process Specifications

7.1.4 Provide Driver Toll Payment Interface

Overview: This process shall be responsible for providing an interface through which drivers can request and pay for other services when paying their tolls at toll plazas. The services supported by this process include advanced payment for parking lot charges and transit fares. The process shall query the driver for sufficient information to enable the advanced parking lot charge and/or transit fare to be determined and the cost either billed to a credit identity provided by the driver's payment instrument, or deducted from credit stored on the instrument. The process shall support inputs from the driver in either manual or audio form, and shall provide its outputs in audible or visual forms. It shall enable the visual output as hardcopy, and/or display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

7.1.7 Provide Payment Instrument Interface for Tolls

Overview: This process shall be responsible for providing the interface through which the payment information can be read from a vehicle tag. The process shall enable the use of the data from the tag for the purposes of paying for current tolls, plus if required, the cost of advanced parking lot charges, and/or transit fares, as well as providing the data for use in traffic flow analysis. The tag data which can be collected by the process shall include credit identity, stored credit value, and the toll segment identity at the vehicle's entry point so that closed toll system can be used. When stored credit is used, the process shall enable the deduction of the cost of the toll and (possibly) advanced payments from the credit value on the tag. The process shall support collection of data from tags on-board a range of vehicle types including private cars or vans, commercial vehicles, transit vehicles, including those used for demand responsive transit services.

7.2.4 Provide Driver Parking Lot Payment Interface

Overview: This process shall be responsible for providing an interface through which drivers can request other services when paying their charges at parking lots. The services supported by this process include advanced parking lot payment, as well as advanced payment for tolls and transit fares. The process shall query the driver for sufficient information to enable the advanced toll, parking lot charge and/or transit fare to be determined and the cost either billed to a credit identity provided by the driver's payment instrument, or deducted from credit stored on the instrument. The process shall support inputs from the driver in manual or audio form, and shall provide its outputs in audible or visual forms. It shall provide visual output in hardcopy, and/or display. Both types of output shall not impair the driver's ability to control the vehicle in a safe manner.

7.2.7 Provide Payment Instrument Interface for Parking

Overview: This process shall be responsible for providing the interface through which the payment information can be read from a vehicle tag. The process shall enable the use of the data from the tag for the purposes of paying the current parking lot charge and if required, advanced payments for tolls and/or transit fares. It shall be possible for the process to collect either the credit identity or the stored credit value data from the tag, and to update the stored credit value as a result of the parking lot charge and (possibly) advanced charges having been paid. The time at which the vehicle entered the parking lot shall also be collected from the tag by the process so that the charge for the use of the lot can be calculated. The process shall support collection of data from tags on-board a range of vehicle types including private cars or vans, commercial vehicles, transit vehicles, including those used for demand responsive transit services.

7.5.1 Provide Vehicle Payment Instrument Interface

Overview: This process shall be responsible for providing the interface through which driver credit identities and stored credit may be entered into the ITS from on-board vehicle tags. The types of vehicle from which data collection is supported by the process shall comprise private cars or vans, commercial vehicles and transit vehicles, including those used for demand responsive transit services. This process shall also provide an interface through which the stored credit held by the tag can be debited for the payment of current or advanced tolls, plus advanced parking lot charges, and/or transit fares.

2.22.3 Subsystem Interfaces for VS

Basic Vehicle => **Vehicle**

Physical Architecture Flow Name: vehicle measures

Sensing information from vehicle sensors.

Logical Architecture Reference Flow: fbv_brake_servo_response
This data flow is sent from the basic vehicle to the Provide Vehicle Monitoring and Control function. It contains analog input which indicates that the vehicle's brake has been moved.

Logical Architecture Reference Flow: fbv_crash_sensor_data
This data flow is sent from the basic vehicle to the Provide Vehicle Control and Monitoring function. It contains analog data from on-board vehicle sensors and consists of information which shows whether the vehicle has been involved in a collision producing a front impact, a left side impact, a right side impact, or a rear impact, plus details of the collision extent and passenger injuries. Size is 4 bytes per sensor with 8 sensors.

Logical Architecture Reference Flow: fbv_diagnostics_data
This data flow is sent from the basic vehicle to the Provide Vehicle Control and Monitoring function and contains data from on-board vehicle diagnostics functions from which the vehicle's operational status can be determined. Size is 4 bytes per function with 8 functions.

Logical Architecture Reference Flow: fbv_driver_safety_status
This data flow is sent from the basic vehicle to the Provide Vehicle Control and Monitoring function and contains sensor data from which the following can be determined: driver_state - the ability of the driver to control the vehicle, negative factors being such things as alcohol on the breath, too many mistakes, etc. injuries - any detectable problems with the vehicle occupants, e.g. sudden change in heart rate, pulse, breathing, etc. Size is 4 bytes per sensor with 8 sensors.

Logical Architecture Reference Flow: fbv_steering_servo_response
This data flow is sent from the basic vehicle to the Provide Vehicle Monitoring and Control function. It contains analog input which indicates that the vehicle's steering has been moved.

Logical Architecture Reference Flow: fbv_throttle_servo_response
This data flow is sent from the basic vehicle to the Provide Vehicle Monitoring and Control function. It contains analog input which indicates that the throttle has been moved.

Logical Architecture Reference Flow: fbv_vehicle_attitude_data
This data flow is sent from the basic vehicle to the Provide Vehicle Monitoring and Control function and contains data obtained from on-vehicle sensors indicating the vehicle's attitude, i.e. right-way up (normal), upside-down, on its side, etc. Size is 4 bytes per sensor.

Logical Architecture Reference Flow: fbv_vehicle_condition
This data flow is sent from the basic vehicle to the Provide Vehicle Monitoring and Control function and contains analog data about the vehicle's current condition from which its suitability for operation on automatic highway system (ahs) lanes can be determined. This flow would be made up of the following components: brake_condition + drive_train_condition + forward_sensors_condition + rear sensors_condition + side_sensors_condition + steering_condition + vehicle_procoessor_condition + fuel_level_condition + tire_wear_and_pressure_condition + vehicle_external_communication_condition + vehicle_internal_communication_condition. Size is 4 bytes per element times 11 elements.

Logical Architecture Reference Flow: fbv_vehicle_headway
This data flow is sent from the basic vehicle to the Provide Vehicle Control and Monitoring function. It contains on-board vehicle sensor input from which the distance between the vehicle and the one in front (headway) can be computed.

Logical Architecture Reference Flow: fbv_vehicle_identity
This data flow is sent from the basic vehicle to the Manage Traffic and the Provide Vehicle Monitoring and Control functions. It contains the identity of the vehicle from which other data such as ownership, vehicle type, plus data from the Department of Motor Vehicles (DMV) can be obtained.

Logical Architecture Reference Flow: fbv_vehicle_lane_position
This data flow is sent from the basic vehicle to the Provide Vehicle Control and Monitoring function. It contains analog data from on-board vehicle sensors input from which the vehicle's position within a lane can be computed. Size is 4 bytes per sensor times 4 sensors.

Logical Architecture Reference Flow: fbv_vehicle_motion_data
This data flow is sent from the basic vehicle to the Provide Vehicle Monitoring and Control function and contains analog data obtained from on-vehicle sensors indicating whether the vehicle is moving in a forward, sideways, or backwards, or in any combined direction. Size is 4 bytes per sensor times 4 sensors.

Logical Architecture Reference Flow: fbv_vehicle_on_ahs_lane
This data flow is sent from the basic vehicle to the Provide Vehicle Control and Monitoring function. It contains analog sensor input from which the type of automatic highway system (ahs) lane (if any) in which the vehicle is traveling can be computed. Size is 4 bytes per sensor times three sensors.

Logical Architecture Reference Flow: fbv_vehicle_proximity_data

This data flow is sent from the basic vehicle to the Provide Vehicle Monitoring and Control function and contains analog data obtained from on-vehicle sensors including but not limited to those covering headway, i.e. distance between vehicle and the next vehicle in front, lateral distance, i.e. the distance between the vehicle and any objects on either side, and rear distance. Size is 4 bytes per sensor times 4 sensors.

Logical Architecture Reference Flow: fbv_vehicle_safety_status
This data flow is sent from the basic vehicle to the Provide Vehicle Control and Monitoring function and contains analog data from on-board vehicle sensors from which the extent of a vehicles collision damage can be determined. Size is 4 bytes per sensor times 8 sensors.

Logical Architecture Reference Flow: fbv_vehicle_speed
This data flow is sent from the basic vehicle to the Provide Vehicle Control and Monitoring function. It contains analog data from on-board vehicle sensors from which the vehicle's speed can be computed.

Commercial Vehicle Subsystem => **Vehicle**

Physical Architecture Flow Name: commercial vehicle data

Information about the commercial vehicles cargo, credentials, and payments.

Logical Architecture Reference Flow: cv_driver_enrollment_cost
This data flow is sent from the Manage Commercial Vehicles function to the Provide Electronic Payment Services function. It contains the cost of the electronic credential filing and taxes, payment of which was previously requested by the commercial vehicle driver acting in the role of fleet manager, and is only sent when the cost is to be deducted from the credit stored on the payment instrument being used by the driver. The data flow consists of the following data items each of which is defined in its own DDE: cv_amount_billed + stored_credit.

Logical Architecture Reference Flow: processed_cargo_data
This data flow is used within the Provide Vehicle Monitoring and Control function. It contains data obtained from the processing by sensors of analog data received on-board the vehicle about the composition and state of its cargo.

Driver => **Vehicle**

Physical Architecture Flow Name: driver inputs

Driver commands to the vehicle.

Logical Architecture Reference Flow: fd_activate_vehicle_control
This data flow is sent from the driver to the Provide Driver and Traveler Services function and is used by a vehicle driver to activate automatic control of the vehicle. The size assumption below allows for some form of electronic signature to be automatically included.

Logical Architecture Reference Flow: fd_guidance_data
This data flow is input by a driver to the Provide Driver and Traveler Services function and contains data requested so that a vehicle route can be determined for which on-line guidance can be provided.

Logical Architecture Reference Flow: fd_guidance_route_accepted
This data flow is sent from the driver to the Provide Driver and Traveler Services function. It contains acceptance of the route that has been generated in response to a previous request from the driver for on-line guidance. Guidance will not begin until the acceptance has been received.

Logical Architecture Reference Flow: fd_request_advisory_information
This data flow is sent from the driver to the Provide Driver and Traveler Services function and contains input from the driver specifying the type of advisory display required or to be disabled.

Physical Architecture Flow Name: request for service

A traveler service request initiated by a driver or traveler. The request may result in a financial transaction, summon an emergency response, or initiate another service at the behest of the driver.

Logical Architecture Reference Flow: fd_emergency_request

This data flow is sent from the driver and to the Provide Driver and Traveler Services function. It consists of the following items : medical_services_required - please send medical assistance; other_vehicle_damage_crash - a crash has occurred involving other vehicles; minor_property_damage_only_crash - a crash has occurred which only involves minor property damage; breakdown - the vehicle has broken down; security_alarm - driver in danger, please send emergency services; cancel - cancel any previous emergency request. Several items may be sent simultaneously with the exception of the last one which will be sent on its own. The size assumption below is intended to allow for some form of electronic signature.

Logical Architecture Reference Flow: fd_guidance_map_update_request

This data flow is sent from the driver to the Provide Driver and Traveler Services function and contains a request for an update of the digitized map data used to provide on-line vehicle guidance.

Logical Architecture Reference Flow: fd_guidance_request

This data flow is sent from the driver to the Provide Driver and Traveler Services function. It contains a request to provide on-line guidance of the driver's vehicle and specifies a choice of the type of preferred guidance, i.e. infrastructure based dynamic, autonomous with link journey and queue times provided from the infrastructure, and totally autonomous. The driver will be prompted for further data in order that the guidance can begin.

Logical Architecture Reference Flow: fd_other_services_parking_request

This data flow is input from a driver to the Provide Driver and Traveler Services function and contains a request for additional services other than simple parking lot charge collection. These may comprise advanced payment of fares, parking lot charges, or toll charges.

Logical Architecture Reference Flow: fd_other_services_toll_request

This data flow is input from a driver to the Provide Driver and Traveler Services function and contains a request for additional services other than simple toll collection. These services may comprise advanced payment of fares, parking lot charges, or toll charges. The information provided by the driver must include sufficient information for the parking lot charge and/or transit fare to be determined.

Emergency Management => Vehicle

Physical Architecture Flow Name: emergency acknowledge

Acknowledge request for emergency assistance and provide additional details regarding actions and verification requirements.

Logical Architecture Reference Flow: emergency_request_driver_acknowledge

This data flow is used by the Manage Emergency Services function to acknowledge that the request for emergency services previously sent by the driver has been received and is therefore sent to the Provide Driver and Traveler Services function for output.

Logical Architecture Reference Flow: emergency_request_vehicle_acknowledge

This data flow is used by the Manage Emergency Services function to acknowledge that the request for emergency services previously sent automatically by the vehicle through processes in the Provide Vehicle Control and Monitoring function has been received. It is sent to the Provide Driver and Traveler Services function for output.

Information Service Provider => Vehicle

Physical Architecture Flow Name: broadcast information

General broadcast information that contains link travel times, incidents, advisories, transit services and a myriad of other traveler information.

Logical Architecture Reference Flow: broadcast_data

This data flow is used within the Provide Driver and Traveler Services function to provide traffic and travel advisory data via a wide area broadcast message to drivers and travelers in vehicles. It consists of the following data items each of which is defined in its own DDE: traffic_data_for_broadcast + predicted_incidents_for_broadcast + prediction_data_for_broadcast + transit_services_for_broadcast + transit_running_data_for_broadcast.

Logical Architecture Reference Flow: link_and_queue_data

This data contains, for each link, the average journey time, speed, and occupancy. For queues it contains the queue times for each link. This data is computed from traffic data and (if available) vehicle probe data. link_state_data_for_broadcast + list_size + list_size{link_identity + link_journey_time + link_queue_time}.

Physical Architecture Flow Name: traveler information

Traveler information comprised of traffic status, advisories, incidents, responses to traveler requests (e.g., traveler routing, yellow pages), payment information and many other travel-related data updates and confirmations.

Logical Architecture Reference Flow: advanced_fares_and_charges_response

This data flow is used within the Provide Electronic Payment Services function and contains the result of the requested advanced transit fare and/or parking lot charge payment transaction from a driver. It consists of the following data item which is defined in its own DDE: confirmation_flag + credit_identity + parking_lot_cost + stored_credit + transit_fare.

Logical Architecture Reference Flow: advanced_tolls_and_fares_response

This data flow is used within the Provide Electronic Payment Services function and contains the result of the requested advanced toll and/or transit fare payment transaction from a driver. It consists of the following data item which is defined in its own DDE: confirmation_flag + credit_identity + stored_credit + toll_cost + transit_fare.

Logical Architecture Reference Flow: advisory_data

This data flow is used within the Provide Driver and Traveler Services function to provide traffic and travel advisory data to drivers and travelers in vehicles. It consists of the following data items each of which is defined in its own DDE: traffic_data_for_advisories + predicted_incidents_for_advisories + prediction_data_for_advisories + transit_services_for_advisories + transit_running_data_for_advisories.

Logical Architecture Reference Flow: driver_map_update_payment_response

This data flow is sent from the Provide Electronic Payment Services function to the Provide Driver and Traveler Services function and contains the response to a previous request from the driver that payment be made for an update of the navigable map database used for on-line vehicle guidance. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + vehicle_identity.

Logical Architecture Reference Flow: link_and_queue_data

This data contains, for each link, the average journey time, speed, and occupancy. For queues it contains the queue times for each link. This data is computed from traffic data and (if available) vehicle probe data. link_state_data_for_broadcast + list_size + list_size{link_identity + link_journey_time + link_queue_time}.

Logical Architecture Reference Flow: yellow_pages_advisory_data

This data flow is used within the Provide Driver and Traveler Services function to provide yellow pages data to drivers and transit users in vehicles and/or confirmation of a previously requested reservation. It consists of the following data items each of which is defined in its own DDE: yellow_pages_data_for_advisories + yellow_pages_cost + yellow_pages_dining_reservation_confirmation + yellow_pages_lodging_reservation_confirmation + yellow_pages_ticket_purchase_confirmation.

Physical Architecture Flow Name: trip plan

A sequence of links and special instructions comprising a trip plan indicating efficient routes for navigating the links. Normally coordinated with traffic conditions, other incidents, preemption and prioritization plans.

Logical Architecture Reference Flow: vehicle_guidance_route
This data flow is used within the Provide Driver and Traveler Services function and is a special form of %034route%034 for vehicle guidance only. It contains a subset of the data items included in the %034route%034 data flow to meet the requirements of in-vehicle infrastructure based guidance as opposed to the more general requirements for a route need as part of a trip planning activity. The data flow consists of the following data items each of which is defined in its own DDE: route_identity + route_segment_number{route_segment_description + route_segment_end_point + route_segment_estimated_travel_time + route_segment_report_position_points + route_segment_start_point} + vehicle_identity.

Location Data Source => **Vehicle**

Physical Architecture Flow Name: position fix

Information which provides a traveler or vehicles geographical position.

Logical Architecture Reference Flow: From_Location_Data_Source
This data is sent to a number of processes in the Provide Driver and Traveler Services function. It contains the current state of such things as GPS signals, magnetic flux and other location data sources. The data is sent to sensor processes for conversion into an actual vehicle location. The size estimate below is based on an equivalent digitized form (e.g. location_identity) even though it may actually be an entirely form .

Map Update Provider => **Vehicle**

Physical Architecture Flow Name: map updates

Map update which could include a new underlying static or real-time map or map layer(s) update.

Logical Architecture Reference Flow: fmup_vehicle_map_update
This data flow is sent from the map update provider to the Provide Driver and Traveler Services function and contains data for a new navigable map database to be used by the on-line vehicle guidance facility.

Logical Architecture Reference Flow: fmup_vehicle_map_update_cost
This data flow is sent from the map update provider to the Provide Driver and Traveler Services function and contains the cost for a new navigable map database to be used by the on-line vehicle guidance facility.

Other Vehicle => **Vehicle**

Physical Architecture Flow Name: vehicle to vehicle coordination

Any type of advanced vehicle to vehicle communication.

Logical Architecture Reference Flow: From_Other_Vehicle
This data flow is used within the Provide Vehicle Monitoring and Control function to receive data from other vehicles in a platoon when in platoon following mode of vehicle operation.

Parking Management => **Vehicle**

Physical Architecture Flow Name: request tag data

Request for tag information including credit identity, stored value card cash, etc.

Logical Architecture Reference Flow: parking_lot_payment_request

This data flow is used within the Provide Electronic Payment Services function and contains the request for the cost of the current parking lot charge to be deducted from the credit currently stored by the payment instrument. It is only sent when a value of stored credit has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: parking_lot_cost.

Logical Architecture Reference Flow: parking_lot_tag_data_request

This data flow is used within the Provide Electronic Payment Services function and contains a request for the parking lot tag data to be read from the store that is held on-board the vehicle.

Physical Architecture Flow Name: tag update

Update data held in tag which can be read at another screening.

Logical Architecture Reference Flow: advanced_parking_lot_charges_confirm

This data flow is used within the Provide Electronic Payment Services function to show that payment for an advanced parking lot charge has been confirmed or not. It consists of the following data items each of which is defined in its own DDE: confirmation_flag + credit_identity + parking_lot_cost + stored_credit.

Logical Architecture Reference Flow: parking_lot_payment_debited

This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the cost of the current parking lot charge will be deducted by the financial institution from the credit identity previously provided by the payment instrument. It is only sent when a credit identity has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: parking_lot_tag_data_clear

This data flow is used within the Provide Electronic Payment Services function and contains the parking lot tag data from which any arrival time has been cleared. The data will have been used to charge for use of the parking lot, and is being cleared to enable its use for future charging. The data flow consists of the following data item which is defined in its own DDE: parking_lot_tag_data.

Logical Architecture Reference Flow: parking_lot_tag_data_update

This data flow is used within the Provide Electronic Payment Services function and contains the parking lot tag data that has been updated. The updated will have loaded the time at which the vehicle entered the parking lot and is for use in charging for the vehicle's use of the lot. The data flow consists of the following data item which is defined in its own DDE: parking_lot_tag_data.

Payment Instrument => Vehicle

Physical Architecture Flow Name: payment

Payment of some kind (e.g., toll, parking, fare) by traveler which in most cases can be related to a credit account.

Logical Architecture Reference Flow: fpi_confirm_payment_at_parking_lot

This data flow is sent from the payment instrument to the Provide Electronic Payment Services function. It contains confirmation that the requested parking lot charge has been successfully deducted from the total credit previously stored by the payment instrument. This data flow will only apply to those types of payment instrument that can carry stored credit and will not be set by those that only contain a credit identity.

Logical Architecture Reference Flow: fpi_confirm_payment_at_toll_plaza

This data flow is sent from the payment instrument to the Provide Electronic Payment Services function. It contains confirmation that the requested payment of the current toll, plus if required the cost of advanced tolls, and/or parking lot charges, and/or transit fares, has been successfully deducted from the total credit previously stored by the payment instrument. This data flow will only apply to those types of payment instrument that can carry stored credit and will not be set by those that only contain a credit identity.

Logical Architecture Reference Flow: fpi_driver_vehicle_input_credit_identity

This data flow is sent from the payment instrument to the Provide Electronic Payment Services function. It contains the data necessary to allow automatic billing of the user of the payment instrument for the update of the navigable map database used within a vehicle for on-line driver guidance, or the cost of commercial vehicle electronic credential filing and tax payments. For the database update, the vehicle may be a private car, a transit vehicle, or a commercial vehicle. A payment instrument is a device that can be used to make payments, e.g. a debit card, a credit card. It will belong to the financial institution responsible for its issue and not to the driver.

Logical Architecture Reference Flow: fpi_parking_tag_data

This data flow is sent from the Payment Instrument to the Provide Electronic Payment Services function and is used to either identify a particular payment instrument or the amount of credit that it currently has stored, when the instrument is being used on-board a vehicle at a parking lot. In either case the data will be used to enable automatic billing for the current parking lot charge, plus if required, advanced payments for tolls, and/or parking lot charges and/or transit fares. The vehicle may be a private car or van, or a transit vehicle, or a commercial vehicle. In the case of a transit vehicle, the payments will be for the vehicle itself and not for its passengers, i.e. transit users. A payment instrument is a device that can be used to make payments, e.g. a debit card, or a device that contains stored credit that can be used for actual payments. It will belong to the financial institution responsible for its issue and not to the user, who in this instance is the vehicle driver.

Logical Architecture Reference Flow: fpi_toll_tag_data

This data flow is sent from the payment instrument to the Provide Electronic Payment Services function and is used to either identify a particular payment instrument or the amount of credit that it currently has stored, when the instrument is being used on-board a vehicle passing through a toll plaza. In either case the data will be used to enable automatic billing for the current toll, plus if required, advanced payments for tolls, and/or parking lot charges and/or transit fares. The vehicle may be a private car or van, a transit vehicle, or a commercial vehicle. In the case of a transit vehicle, the payments will be for the vehicle itself and not for its passengers, i.e. transit users. A payment instrument is a device that can be used to make payments, e.g. a debit card, or a device that contains stored credit that can be used for actual payments. It will belong to the financial institution responsible for its issue and not to the user, who in this instance is the vehicle driver.

Logical Architecture Reference Flow: fpi_transit_user_vehicle_input_credit_identity

This data flow is sent from the payment instrument to the Provide Electronic Payment Services function. It contains the data necessary to allow automatic billing of the transit user for advanced tolls, and/or parking lot charges, and/or transit fares, when the user is on-board a transit vehicle. A payment instrument is a device that can be used to make payments, e.g. a debit card, or a stored value card. It will belong to the financial institution responsible for its issue and not to the user.

Potential Obstacles => Vehicle

Physical Architecture Flow Name: physical presence

Detection of an obstacle by a vehicle. Obstacle could include animals, other vehicles, pedestrians, rocks in roadway etc.

Logical Architecture Reference Flow: From_Potential_Obstacles

This data flow is sent from potential obstacles to the Provide Vehicle Monitoring and Control function. It contains analog data that provides information on potential obstacles that can be encountered by a vehicle.

Roadway => **Vehicle**

Physical Architecture Flow Name: roadway characteristics

Detectable or measurable road characteristics such as friction coefficient and general surface conditions, road geometry and markings, etc. These characteristics are monitored or measured by vehicle ITS components and used to support advanced vehicle safety and control capabilities.

Logical Architecture Reference Flow: From_Roadway
This data flow contains analog information about the roadway. It is sent to the Provide Vehicle Monitoring and Control function.

Roadway Environment => **Vehicle**

Physical Architecture Flow Name: weather conditions

Collected weather condition data from sensors.

Logical Architecture Reference Flow: fre_roadside_data
This data flow is sent by the roadway environment to the Provide Vehicle Monitoring and Control function. It contains analog data from which sensors on-board the vehicle can determine the physical conditions such as fog, ice, snow, rain, etc. at the road or highway.

Roadway Subsystem => **Vehicle**

Physical Architecture Flow Name: AHS control data

Information required for vehicles to operate on AHS lanes.

Logical Architecture Reference Flow: ahs_check_response
This data flow is used within the Provide Vehicle Monitoring and Control function and contains the response to the checking of data from on-board a vehicle to see if it is suitable for operating on automatic highway system (ahs) lanes. The data flow consists of the following data items each of which is defined in its own DDE: ahs_control_data_update + confirmation_flag.

Physical Architecture Flow Name: intersection status

Status of intersection congestion, approaching vehicles, etc.

Logical Architecture Reference Flow: intersection_collision_avoidance_data
This data flow is sent from the Manage Traffic function to the Provide Vehicle Monitoring and Control function. It contains data for a vehicle that shows that it is likely to be involved in a collision at an intersection, unless it takes some avoiding action. The data flow is sized at two (2) bytes so that it can show the direction from which the other vehicle(s) is (are) approaching. This will help the vehicle to decide which is the most appropriate avoiding action to take.

Physical Architecture Flow Name: request tag data

Request for tag information including credit identity, stored value card cash, etc.

Logical Architecture Reference Flow: parking_lot_tag_data_needed
This data flow is used within the Manage Traffic and Provide Electronic Payment Services functions to request the

output of the data from a toll tag that may be on-board a vehicle. This data will be used to calculate vehicle journey times for links in the road (surface street) and freeway network served by the Manage Traffic function.

Logical Architecture Reference Flow: toll_tag_data_needed
This data flow is used within the Manage Traffic and Provide Electronic Payment Services functions to request the output of the data from a parking lot tag that may be on-board a vehicle. This data will be used to calculate vehicle journey times for links in the road (surface street) and freeway network served by the Manage Traffic function.

Physical Architecture Flow Name: vehicle signage data

In-vehicle signage data generated by the roadway infrastructure indicating either road conditions, street names, or special information which will be useful for a vehicle passing a specific point on the roadway.

Logical Architecture Reference Flow: vehicle_signage_data
This data flow is sent from the Manage Traffic function to the Provide Driver and Traveler Services function. It contains data for use in producing in-vehicle signage displays. The data flow consists of the following data items each of which is defined in its own DDE: vehicle_sign_data_output_location + 6{vehicle_signage_output_data} + 8{vehicle_signage_traffic_data}.

Logical Architecture Reference Flow: vehicle_smart_probe_data_output
This data flow is sent from the Manage Traffic function to the Provide Driver and Traveler Services function. It contains the data obtained from vehicle smart probes, processsed and formatted for output to vehicles as they pass by. The data flow is sized at one (1) byte and will contain a compressed version of the internally used character codes for the various types of hazard that may be found on roads and freeways.

Toll Collection => **Vehicle**

Physical Architecture Flow Name: request tag data

Request for tag information including credit identity, stored value card cash, etc.

Logical Architecture Reference Flow: toll_payment_request
This data flow is used within the Provide Electronic Payment Services function and contains the request for the cost of the current toll to be deducted from the credit currently stored by the payment instrument. It is only sent when a value of stored credit has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: toll_cost.

Logical Architecture Reference Flow: toll_tag_data_request
This data flow is used within the Provide Electronic Payment Services function and contains a request for the toll tag data to be read from the store that is held on-board the vehicle.

Physical Architecture Flow Name: tag update

Update data held in tag which can be read at another screening.

Logical Architecture Reference Flow: toll_payment_debited
This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the cost of the current toll will be deducted by the financial institution from the credit identity previously provided by the payment instrument being used by the driver. It is only sent when a credit identity has been previously received from the payment instrument. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: toll_tag_data_clear
This data flow is used within the Provide Electronic Payment Services function and contains the toll tag data from

which any toll segment identity has been cleared. The data will have been used to charge for use of the toll road, and is being cleared to enable its use for future charging. The data flow consists of the following data item which is defined in its own DDE: toll_tag_data.

Logical Architecture Reference Flow: toll_tag_data_update
This data flow is used within the Provide Electronic Payment Services function and contains the toll tag data that has been updated. The updated will have loaded the identity of the toll segment at which the vehicle entered the toll road and is for use in charging for the vehicle's use of the toll road. The data flow consists of the following data item which is defined in its own DDE: toll_tag_data.

Transit Vehicle Subsystem => Vehicle

Physical Architecture Flow Name: traveler advisory request

In vehicle communication between transit and vehicle systems includes advisories and advance payment deductions.

Logical Architecture Reference Flow: transit_user_advanced_payment_on_vehicle
This data flow is sent from the Manage Transit function to the Provide Electronic Payment Services function. It contains the cost of advanced payments that must be deducted from the credit currently stored on the payment instrument being used by a transit user on-board a transit vehicle. These advanced payments may cover tolls, and/or parking lot charges, and/or transit fares. The data flow consists of the following data items each of which is defined in its own DDE: stored_credit + parking_lot_cost + toll_cost + transit_fare.

Logical Architecture Reference Flow: transit_user_advisory_information_request
This data flow is used within the Provide Driver and Traveler Services function and contains analyzed requests for the various types of transit user display. It contains the following data items each of which is defined in its own DDE: advisory_display_type + advisory_data_scope.

Vehicle => Basic Vehicle

Physical Architecture Flow Name: vehicle control

Vehicular control commands

Logical Architecture Reference Flow: tbv_change_brake_setting
This data flow is sent to the basic vehicle from the Provide Vehicle Monitoring and Control function and contains analog output which alters the vehicle's wheel brake setting. A zero value will completely release the brake and any positive value will apply the brake by the set amount.

Logical Architecture Reference Flow: tbv_change_direction
This data flow is sent to the basic vehicle from the Provide Vehicle Monitoring and Control function and contains analog output which changes the vehicle's direction of motion between forward, reverse and neutral.

Logical Architecture Reference Flow: tbv_change_throttle_setting
This data flow is sent to the basic vehicle from the Provide Vehicle Monitoring and Control function and contains analog output which alters the vehicle's throttle setting. A zero output will close the throttle and any positive value will cause the throttle to be opened to the set value.

Logical Architecture Reference Flow: tbv_deploy_crash_restraints
This data flow is sent to the basic vehicle from the Provide Vehicle Monitoring and Control function and contains analog output which initiates the deployment of the vehicle's crash restraint devices.

Logical Architecture Reference Flow: tbv_steer_left
This data flow is sent to the basic vehicle from the Provide Vehicle Monitoring and Control function and is used to

provide analog output which steers the vehicle to the left.

Logical Architecture Reference Flow: tbv_steer_right
This data flow is sent to the basic vehicle from the Provide Vehicle Monitoring and Control function and is used to provide analog output that will steer a vehicle to the right.

Logical Architecture Reference Flow: tbv_steer_straight
This data flow is sent to the basic vehicle from the Provide Vehicle Monitoring and Control function and is used to provide analog output which centralizes the vehicle's steering from a steer left or steer right position.

Vehicle => **Commercial Vehicle Subsystem**

Physical Architecture Flow Name: commercial vehicle data request

Request for commercial vehicle information (cargo, driver's credit, vehicle location).

Logical Architecture Reference Flow: cargo_data_request
This data flow is used within the Provide Vehicle Monitoring and Control function. It contains a request for data about a vehicle's cargo including any damage report to be sent back in reply.

Logical Architecture Reference Flow: cv_driver_credit_identity
This data flow is sent from the Provide Electronic Payments Services function to the Manage Commercial Vehicles function. It contains the credit identity of a commercial vehicle driver or the amount of stored credit obtained from the payment instrument terminator and consists of the following data items each of which is defined in its own DDE: credit_identity + stored_credit.

Logical Architecture Reference Flow: vehicle_location_for_cv
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Commercial Vehicles function. It contains the vehicle location data for use in reporting on-board data to the commercial fleet manager, or the driver acting in this role. The data is computed from data input to sensors controlled by the processes that determines vehicle location. The data flow consists of the following data item which is defined in its own DDE: location_identity.

Vehicle => **Driver**

Physical Architecture Flow Name: driver updates

Information displayed or otherwise conveyed by the vehicle to the driver.

Logical Architecture Reference Flow: td_advisory_information
This data flow is sent to the driver from the Provide Driver and Traveler Services function. It contains displays of the various types of traffic and travel information messages available to the driver. This information will only be output following a specific request from the driver and will be filtered to only include that which is relevant to the vehicle's current location.

Logical Architecture Reference Flow: td_broadcast_information
This data flow is sent to the driver from the Provide Driver and Traveler Services function. It contains displays of the various types of broadcast information which can be output to the driver. This information will comprise but not be limited to such things as safety warnings, position warnings, enhanced vision, vehicle status, data from smart vehicle probes, etc.

Logical Architecture Reference Flow: td_driving_guidance
This data flow is sent to the driver from the Provide Driver and Traveler Services function and contains output (displays - text and/or graphics, and/or audio based information) which gives the driver instructions on how to steer the

vehicle, e.g. turn left at the next intersection, take the middle lane, fork right at the next intersection, etc.

Logical Architecture Reference Flow: td_guidance_input_request
This data flow is sent to the driver from the Provide Driver and Traveler Services function and contains a request for the input a specific item of data needed to determine the vehicle route for on-line guidance. The data may comprise such things as the destination, preferred arrival time, plus route choice preferences and constraints.

Logical Architecture Reference Flow: td_guidance_route_details
This data flow is sent to the driver from the Provide Driver and Traveler Services function and contains details of the route that has been selected in response to the driver's request for on-line guidance. The route and choice of guidance method will have been based on previous input from the driver. Guidance will not begin until the driver has positively accepted this data.

Physical Architecture Flow Name: transaction status

Response to transaction request. Normally dealing with a request for payment.

Logical Architecture Reference Flow: td_guidance_map_update_response
This data flow is sent to the driver from the Provide Driver and Traveler Services function and contains a the response to a previous request for the update of the digitized map data used to provide on-line vehicle guidance.

Logical Architecture Reference Flow: td_other_services_parking_response
This data flow is sent to the driver from the Provide Driver and Traveler Services function and contains the response to the traveler's previously input request for additional services other than simple parking lot charge collection.

Logical Architecture Reference Flow: td_other_services_toll_response
This data flow is sent to the driver from the Provide Driver and Traveler Services function and contains the response to the traveler's previously input request for additional services other than simple toll collection.

Vehicle => **Emergency Management**

Physical Architecture Flow Name: emergency notification

An emergency request for assistance originated by a traveler using an in-vehicle, public access, or personal device. Sufficient information is provided so that the recipient can determine the location of the emergency as a minimum. Additional information identifying the requestor and requesting device and the nature and severity of the emergency may also be provided (and required) by some systems.

Logical Architecture Reference Flow: emergency_request_driver_details
This data flow is used by the Provide Driver and Traveler Services function to send data about an emergency declared by a driver to the Manage Emergency Services function. It contains the following data items each of which is defined in its own DDE: date + driver_personal_emergency_request + time.

Logical Architecture Reference Flow: emergency_request_vehicle_details
This data flow is used by the Provide Vehicle Control and Monitoring function to send data about an emergency automatically declared by a vehicle to the Manage Emergency Services function. It contains the following data items each of which is defined in its own DDE: date + time + vehicle_emergency_request.

Vehicle => **Emergency Vehicle Subsystem**

Physical Architecture Flow Name: vehicle location

Location of vehicle and other vehcile characteristics which are exchanged between vehicle subsystems.

Logical Architecture Reference Flow: vehicle_location_for_emergency_services
This data flow is sent from the Provide Driver and Traveler Services function to the Manage Emergency Services function and contains a vehicle's location as computed from data input to sensors controlled by the processes that determines vehicle location. This is a high precision data flow that enables the location of an emergency vehicle to be pin-pointed to a high degree of accuracy by the Manage Emergency Services function. It consists of the following data item which is defined in its own DDE: location_identity.

Vehicle => **Information Service Provider**

Physical Architecture Flow Name: traveler profile

Information about a traveler including equipment capabilities, personal preferences and recurring trip characteristics.

Logical Architecture Reference Flow: traveler_profile_from_vehicle
This data flow contains a traveler's personal profile which is submitted one time and then used to generate future personalized trip information. This profile supports a subscription type of information dissemination to the traveler. It consists of the following data items each of which is defined in its own DDE: traveler_traffic_profile +traveler_transit_profile.

Physical Architecture Flow Name: traveler request

Request by a traveler to summon assistance, request information, make a reservation, or initiate any other traveler service.

Logical Architecture Reference Flow: advanced_fares_and_charges_request
This data flow is used within the Provide Electronic Payment Services function to transfer requests for advanced payments from the driver interface for subsequent processing. The size of the data flow has been set at less than the sum of the two constituent flows to allow for the fact that they will both not be present for every data transfer. It consists of the following data items each of which is defined in its own DDE: advanced_fare_details + advanced_parking_lot_charges.

Logical Architecture Reference Flow: advanced_tolls_and_fares_request
This data flow is used within the Provide Electronic Payment Services function to transfer requests for advanced payments from the driver parking lot charge payment interface for subsequent processing. The size of the data flow has been set at less than the sum of the two constituent flows to allow for the fact that they will both not be present for every data transfer. It consists of the following data items each of which is defined in its own DDE: advanced_fare_details + advanced_tolls.

Logical Architecture Reference Flow: advisory_data_request
This data flow is used within the Provide Driver and Traveler Services function to request that advisory data be output to a driver or a traveler in a vehicle. The scope and transit route number data will be provided by the driver or transit user, whilst the vehicle location will be provided automatically. The data flow consists of the following data items each of which is defined in its own DDE: advisory_data_scope + vehicle_location_for_advisories + transit_route_number + transit_vehicle_identity.

Logical Architecture Reference Flow: driver_map_update_payment_request
This data flow is sent from the Provide Driver and Traveler Services function to the Provide Electronic Payment Services function and contains a request that payment be made for an update of the navigable map database used for on-line vehicle guidance. The payment will be made by debiting the credit identity with the cost through the financial institution terminator. It consists of the following data items each of which is defined in its own DDE: vehicle_identity + credit_identity + navigable_map_vehicle_update_cost.

Physical Architecture Flow Name: trip confirmation

Acknowledgement by the driver/traveler of acceptance of a route.

Logical Architecture Reference Flow: vehicle_guidance_route_accepted
This data flow is used within the Provide Driver and Traveler Services function and contains the acceptance by the driver of the previously provided route for on-line guidance. Acceptance must be provided before guidance can begin. The data flow consists of the following data item which is defined in its own DDE: route_identity.

Physical Architecture Flow Name: trip request

Request by a driver/traveler for special routing.

Logical Architecture Reference Flow: vehicle_route_request
This data is flow used within the Provide Driver and Traveler Services function and contains a request for on-line guidance of the vehicle. This will have been generated by the driver and will include the necessary source and destination data from which a route can be computed. It consists of the following data items each of which is defined in its own DDE: constraint_on_acceptable_travel_time + constraint_on_eta_change + constraint_on_special_needs + constraint_on_load_classification + constraint_on_ahs_lanes + constraint_on_interstate + constraint_on_urban + constraint_on_vehicle_type + destination + departure_time + desired_arrival_time + origin + preferred_routes + preferred_alternate_routes + preferred_route_segments + vehicle_location_for_dynamic_guidance + vehicle_identity.

Physical Architecture Flow Name: vehicle probe data

Vehicle probe data indicating identity, route segment identity, link time and location.

Logical Architecture Reference Flow: vehicle_guidance_probe_data
This data flow is used within the Provide Driver and Traveler Services function and contains the time at which a vehicle was at a route segment end point. This data will be used to calculate the actual vehicle journey time for the route segment which may supplement or replace data gathered from other sources. This data will be used for in-vehicle guidance purposes. The data flow consists of the following data items each of which is defined in its own DDE route_segment_identity + time + vehicle_identity.

Physical Architecture Flow Name: yellow pages request

Request for information through a yellow pages type service.

Logical Architecture Reference Flow: yellow_pages_advisory_requests
This data flow is used within the Provide Driver and Traveler Services function to request that data about yellow pages services be output to a driver or a transit user in a vehicle or that a yellow pages services reservation b made. The scope and transit route number data will be provided by the driver or transit user, whilst the vehicle location will be provided automatically. The data flow consists of the following data items each of which is defined in its own DDE: advisory_data_scope + vehicle_location_for_advisories + transit_route_number + transit_vehicle_identity + yellow_pages_dining_reservation + yellow_pages_lodging_reservation + yellow_pages_ticket_purchase.

Vehicle => **Map Update Provider**

Physical Architecture Flow Name: map update request

Request for a map update which could include a new underlying map or map layer updates.

Logical Architecture Reference Flow: tmup_vehicle_map_update_cost_request
This data flow is sent to the map update provider from the Provide Driver and Traveler Services function and contains a request for the cost of an update to the navigable map database used for providing in-vehicle on-line guidance.

Logical Architecture Reference Flow: tmup_vehicle_map_update_request
This data flow is sent to the map update provider from the Provide Driver and Traveler Services function and contains

a request for an update of the navigable map database used for providing in-vehicle on-line guidance.

Vehicle => **Other Vehicle**

Physical Architecture Flow Name: vehicle to vehicle coordination

Any type of advanced vehicle to vehicle communication.

Logical Architecture Reference Flow: To_Other_Vehicle
This data flow is used by the provide Vehicle Monitoring and Control function to send data to other vehicles in a platoon when in platoon following mode of vehicle operation.

Vehicle => **Parking Management**

Physical Architecture Flow Name: tag data

Unique tag ID and related vehicle information for the purposes of payment for services.

Logical Architecture Reference Flow: advanced_parking_lot_charges_request
This data flow is used within the Provide Electronic Payment Services function to request that a parking lot charge be paid for in advance by a driver who is already paying a charge for the immediate use of a parking lot space. It consists of the following data items each of which is defined in its own DDE: credit_identity + parking_lot_identity + parking_space_details + stored_credit + vehicle_identity.

Logical Architecture Reference Flow: parking_lot_payment_confirmation
This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the previous request for the cost of the current parking lot charge to be deducted from the credit currently stored by the payment instrument has been completed successfully. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: parking_lot_tag_data_collect
This data flow is used within the Provide Electronic Payment Services function and contains the parking lot tag data that is being collected from on-board the vehicle. This data will be used as the means by which the vehicle will be charged for its use of the parking lot and will consist of the following data item which is defined in its own DDE: parking_lot_tag_data.

Vehicle => **Payment Instrument**

Physical Architecture Flow Name: request for payment

Request to deduct cost of service from user's payment account.

Logical Architecture Reference Flow: tpi_debited_driver_payment_at_vehicle
This data flow is sent to the payment instrument from the Provide Electronic Payment Services function. It is a request to deduct the cost of the update to the navigable map database used within a vehicle for on-line driver guidance, or the cost of commercial vehicle electronic credential filing and tax payments, from the value of credit currently stored on the payment instrument being used by the driver.

Logical Architecture Reference Flow: tpi_debited_payment_at_parking_lot
This data flow is sent to the payment instrument by the Provide Electronic Payment Services function and contains confirmation that the cost of the current parking lot charge, plus if required the cost of advanced tolls, and/or parking lot charges, and/or transit fares, will be debited to the credit identity provided by the payment instrument. The debit

transaction will be carried out through the financial institution through other processes within the function.

Logical Architecture Reference Flow: tpi_debited_payment_at_toll_plaza
This data flow is sent to the payment instrument by the Provide Electronic Payment Services function and contains confirmation that the cost of the current toll, plus if required the cost of advanced tolls, and/or parking lot charges, and/or transit fares, will be debited to the credit identity provided by the payment instrument. The debit transaction will be carried out through the financial institution through other processes within the function.

Logical Architecture Reference Flow: tpi_debited_transit_user_payment_at_vehicle
This data flow is sent to the payment instrument from the Provide Electronic Payment Services function. It is a request to deduct the cost of advanced payments from the value of credit currently stored by the payment instrument belonging being used by a transit user on-board a transit vehicle. The advanced payments may cover tolls, and/or parking lot charges, and/or transit fares.

Logical Architecture Reference Flow: tpi_request_payment_at_parking_lot
This data flow is sent to the payment instrument from the Provide Electronic Payment Services function. It is a request to deduct the total cost of the current parking lot charge, or if required, that for advanced tolls, and/or parking lot charges and/or transit fares from the credit currently stored by the payment instrument when used on-board a vehicle at a parking lot.

Logical Architecture Reference Flow: tpi_request_payment_at_toll_plaza
This data flow is sent to the payment instrument from the Provide Electronic Payment Services function. It is a request to deduct the total cost of the current toll, or if required, that for advanced tolls, and/or parking lot charges and/or transit fares from the credit currently stored by the payment instrument when used on-board a vehicle at a toll plaza.

Vehicle => Roadway Subsystem

Physical Architecture Flow Name: AHS vehicle data

AHS route and vehicle condition data

Logical Architecture Reference Flow: ahs_route_data
This data flow is used within the Provide Vehicle Monitoring and Control function and contains a list of the route segments that will be used by a vehicle. These route segments will be those that contain automatic highway system (ahs) lanes, and will be used by the vehicle on its ahs controlled route. The data flow consists of the following data items each of which is defined in its own DDE: list_size + list_size{route_segment_identity}.

Logical Architecture Reference Flow: ahs_vehicle_condition
This data flow is used within the Provide Vehicle Monitoring and Control function. It contains data processed from on-board vehicle sensors that show the vehicle's current operating condition. This data is used to determine its suitability for operating on automatic highway system (ahs) lanes.

Physical Architecture Flow Name: vehicle probe data

Vehicle probe data indicating identity, route segment identity, link time and location.

Logical Architecture Reference Flow: parking_lot_tag_data_input
This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function. It contains the data from parking lot and toll tags on-board vehicles which will be used to calculate vehicle journey times for links in the road (surface street) and freeway network served by the Manage Traffic function. The data consists of a unique identity number which is assigned to each tag as it is read parking_lot_tag_data.

Logical Architecture Reference Flow: toll_tag_data_input

This data flow is sent from the Provide Electronic Payment Services function to the Manage Traffic function. It contains the data from a toll tag on-board a vehicle which will be used to calculate vehicle journey times for links in the road (surface street) and freeway network served by the Manage Traffic function. The data flow consists of the following data item which is defined in its own DDE: toll_tag_data.

Logical Architecture Reference Flow: vehicle_smart_probe_data

This data flow contains data which provides information about conditions in the vicinity of the smart probe. These conditions, which may be the indication of a hazard on the road or freeway that has been detected by sensors on-board the vehicle. The type of information measured could comprise but not be limited to such things as, temperature, fog, ice, snow, and road condition (e.g. wet, icy, dry). The data may be provided as distinct elements with actual measured values (e.g. temperature) or it could provide conditions from a list of codes.

Logical Architecture Reference Flow: vehicle_status_details_for_emissions

This data flow is sent from the Provide Vehicle Control and Monitoring function to the Manage Traffic function. It contains the operational status of the vehicle which is important because the levels of pollution vary according to how long the vehicle has been running, i.e. how warm is the engine, and what it is actually doing, e.g. is it stationary, or pulling away from a stop. The data flow consists of the following data item which is defined in its own DDE: vehicle_status_details.

Vehicle => Toll Collection

Physical Architecture Flow Name: tag data

Unique tag ID and related vehicle information for the purposes of payment for services.

Logical Architecture Reference Flow: toll_payment_confirmation

This data flow is used within the Provide Electronic Payment Services function and contains confirmation that the previous request for the cost of the current toll to be deducted from the credit currently stored by the driver's payment instrument has been completed successfully. The data flow consists of the following data item which is defined in its own DDE: confirmation_flag.

Logical Architecture Reference Flow: toll_tag_data_collect

This data flow is used within the Provide Electronic Payment Services function and contains the toll tag data that is being collected from on-board the vehicle. This data will be used as the means by which the vehicle will be charged for its use of the toll road and will consist of the following data item which is defined in its own DDE: toll_tag_data.

Vehicle => Transit Vehicle Subsystem

Physical Architecture Flow Name: vehicle location

Location of vehicle and other vehcile characteristics which are exchanged between vehicle subsystems.

Logical Architecture Reference Flow: transit_user_advisory_information

This data flow is used within the Provide Driver and Traveler Services function. It contains data to be converted into output displays for the transit user and includes the following data item which is defined in its own DDE: advisory_data.

Logical Architecture Reference Flow: transit_user_vehicle_credit_identity

This data flow is sent from the Provide Electronic Payments Services function to the Manage Transit function and contains the credit identity of a transit user on-board a transit vehicle, or a stored credit value. Either data item is obtained by a process within the Provide Electronic Payment Services function as data input from the payment instrument terminator and consists of the following data items each of which is defined in its own DDE: credit_identity + stored_credit.

Logical Architecture Reference Flow: vehicle_location_for_transit

This data flow is sent from the Provide Driver and Traveler Services function to the Manage transit function. It contains the vehicle's location as computed from data input to sensors controlled by the processes that determine vehicle location. This is a low precision data flow that will be refined using data obtained from on-board the transit vehicle. It consists of the following data item which is defined in its own DDE: location_identity.

2.22.4 Subsystem Architecture Flow Diagrams

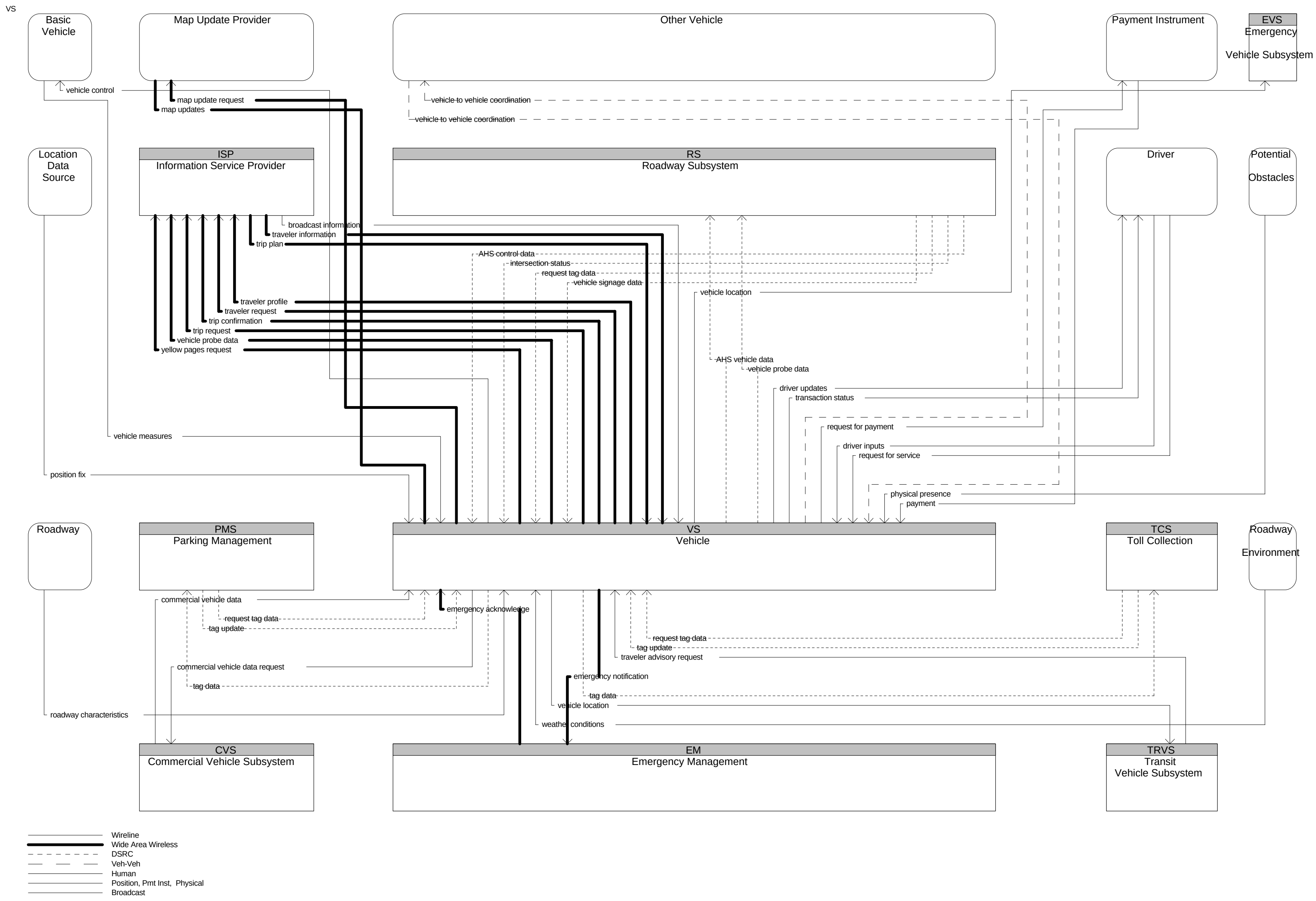


Figure 2.22-2 Architecture Flow Diagrams for Vehicle Subsystem

2.23 Special Constraints

The following section (Communication Layer Section 3) defines the nominal communication requirements for each of the communication channels identified by data flows. Special constraints exist for some of the interconnections between subsystems which should be noted. The following report identifies 5 types of special considerations and the data flows and interconnects which are affected.

2.23.1 Emergency Priority

Communication channel requires priority in emergencies. These data channels require that they be operational even when there is an emergency which might place other loads on the interface. A private communication channel or frequency may be required to satisfy the requirement.

Source	Destination	Interconnect	Architecture Flow
Emergency Management	Emergency Vehicle Subsystem	U1t	assigned route
Emergency Management	Emergency Vehicle Subsystem	U1t	emergency dispatch requests
Emergency Management	Emergency Vehicle Subsystem	U1t	Hazmat information
Emergency Management	Fleet and Freight Management	W	Hazmat information request
Emergency Management	Information Service Provider	W	emergency vehicle route request
Emergency Management	Other EM	W	emergency coordination
Emergency Management	Traffic Management	W	emergency vehicle greenwave request
Emergency Personnel	Emergency Vehicle Subsystem	H	EV driver inputs
Emergency Vehicle Subsystem	Emergency Management	U1t	emergency vehicle driver inputs
Emergency Vehicle Subsystem	Emergency Management	U1t	emergency vehicle driver status update
Emergency Vehicle Subsystem	Emergency Management	U1t	emergency vehicle tracking data
Emergency Vehicle Subsystem	Emergency Personnel	H	emergency dispatch order
Emergency Vehicle Subsystem	Roadway Subsystem	U2	emergency vehicle preemption request
Other EM	Emergency Management	W	emergency coordination

Special Constraints

Personal Information Access	Emergency Management	U1t	emergency notification
Remote Traveler Support	Emergency Management	W,U1t	emergency notification
Remote Traveler Support	Transit Management	W	emergency notification
Traffic Management	Emergency Management	W	incident information request
Traffic Management	Emergency Management	W	incident notification
Transit Management	Emergency Management	W	security alarms
Transit Vehicle Subsystem	Transit Management	U1t	emergency notification
Vehicle	Emergency Management	U1t	emergency notification

2.23.2 Financial Security

Data contains financial information and must be protected accordingly. This data is specifically called out between the user's card and the infrastructure and between the infrastructure and financial institutions. Protections currently exist for the latter. Financial data transmitted over the air must be recognized as private data with an additional reliability requirement. Financial data may exist between other subsystems as part of normal messaging. It is assumed that such data will be treated with the same constraints as the interfaces specifically identified

Source	Destination	Interconnect	Architecture Flow
Commercial Vehicle Administration	Financial Institution	W	payment request
Financial Institution	Commercial Vehicle Administration	W	transaction status
Financial Institution	Information Service Provider	W	transaction status
Financial Institution	Parking Management	W	transaction status
Financial Institution	Toll Administration	W	transaction status
Financial Institution	Transit Management	W	transaction status
Payment Instrument	Remote Traveler Support	S	Payment
Payment Instrument	Transit Vehicle Subsystem	S	payment
Payment Instrument	Vehicle	S	payment
Remote Traveler Support	Payment Instrument	S	request for payment

Special Constraints

Transit Vehicle Subsystem	Payment Instrument	S	request for payment
Transit Vehicle Subsystem	Transit Management	U1t,U2	fare and payment status
Transit Vehicle Subsystem	Transit Management	U1t,U2	request for bad tag list
Vehicle	Payment Instrument	S	request for payment

2.23.3 Personal Privacy

Data contains personal information. Traveler requests and traveler location are private and should be protected. Subsystems aggregate these data and forward specific data with the traveler's permission.

Source	Destination	Interconnect	Architecture Flow
Commercial Vehicle Administration	DMV	W	license request
Commercial Vehicle Administration	Enforcement Agency	W	request for information on violators
Commercial Vehicle Administration	Enforcement Agency	W	violation notification
Commercial Vehicle Check	Commercial Vehicle Administration	W	citation and accident data
DMV	Commercial Vehicle Administration	W	registration
DMV	Toll Administration	W	registration
DMV	Traffic Management	W	registration
Enforcement Agency	Commercial Vehicle Administration	W	information on violators
Fleet and Freight Management	Commercial Vehicle Administration	w	tax filing, audit data
Fleet and Freight Management	Information Service Provider	W	route request
Information Service Provider	Fleet and Freight Management	W	route plan
Information Service Provider	Parking Management	W	parking lot data request
Information Service Provider	Parking Management	W	parking reservations request
Information Service Provider	Personal Information Access	W,U1t	traveler information

Information Service Provider	Personal Information Access	W,U1t	trip plan
Information Service Provider	Remote Traveler Support	W,U1t	traveler information
Information Service Provider	Remote Traveler Support	W	trip plan
Information Service Provider	Traffic Management	W	logged route plan
Information Service Provider	Transit Management	W	demand responsive transit request
Information Service Provider	Transit Management	W	selected routes
Information Service Provider	Vehicle	U1t,U1b	traveler information
Information Service Provider	Vehicle	U1t	trip plan
Information Service Provider	Yellow Pages Service Providers	W	provider registration confirm
Information Service Provider	Yellow Pages Service Providers	W	travel service request
Personal Information Access	Information Service Provider	W,U1t	traveler information request
Personal Information Access	Information Service Provider	U1t	traveler profile
Personal Information Access	Information Service Provider	W,U1t	trip confirmation
Personal Information Access	Information Service Provider	W,U1t	trip request
Personal Information Access	Information Service Provider	W,U1t	yellow pages request
Personal Information Access	Traveler	H	traveler interface updates
Remote Traveler Support	Information Service Provider	W	traveler information request
Remote Traveler Support	Information Service Provider	W	traveler selection
Remote Traveler Support	Information Service Provider	W	trip request
Remote Traveler Support	Information Service Provider	W	yellow pages request

Remote Traveler Support	Transit Management	W	transit request
Remote Traveler Support	Transit Management	W	traveler information request
Remote Traveler Support	Traveler	H	traveler interface updates
Secure Area Environment	Transit Management	P	physical activities
Transit Management	Information Service Provider	W	demand responsive transit plan
Transit Management	Information Service Provider	W	transit request confirmation
Transit Management	Remote Traveler Support	W	traveler information
Transit Management	Transit Vehicle Subsystem	U1t	traveler information
Transit Vehicle Subsystem	Transit Management	U1t	traveler information request
Vehicle	Information Service Provider	U1t	traveler information request
Vehicle	Information Service Provider	U1t	trip confirmation
Vehicle	Information Service Provider	U1t	trip request
Vehicle	Information Service Provider	U1t	vehicle probe data
Vehicle	Information Service Provider	U1t	yellow pages request
Vehicle	Parking Management	U2	tag data
Vehicle	Roadway Subsystem	U2	vehicle probe data
Vehicle	Toll Collection	U2	tag data
Vehicle Characteristics	Parking Management	P	vehicle image
Vehicle Characteristics	Toll Collection	P	vehicle image

2.23.4 Reliability

Failure of the communication medium may result in severe accident. This communication channel may require redundant paths or extra attention paid to potential failure modes. For wireline cases, this may indicate alternate phone or other connections are required. For wireless cases (for AHS applications), special attention will be paid to the transmitters, receivers, and potential interference

Special Constraints

for these connections

Source	Destination	Interconnect	Architecture Flow
Basic Vehicle	Vehicle	W	vehicle measures
Commercial Vehicle	Commercial Vehicle Subsystem	W	vehicle measures
Commercial Vehicle Check	Commercial Vehicle Subsystem	U2	pass/pull-in
Commercial Vehicle Subsystem	Commercial Vehicle Check	U2	screening data
Multimodal Crossings	Roadway Subsystem	W	request for right of way
Multimodal Crossings	Roadway Subsystem	W	right of way preemption request
Other Vehicle	Vehicle	U3	vehicle to vehicle coordination
Parking Management	Vehicle	U2	request tag data
Parking Management	Vehicle	U2	tag update
Roadway Subsystem	Driver	H	driver information
Roadway Subsystem	Multimodal Crossings	W	grant right of way and/or stop traffic
Roadway Subsystem	Traffic Management	W	hri status
Roadway Subsystem	Traffic Management	W	incident data
Roadway Subsystem	Traffic Management	W	intersection blockage notification
Roadway Subsystem	Traffic Management	W	signal priority request
Roadway Subsystem	Vehicle	U2	AHS control data
Roadway Subsystem	Vehicle	U2	intersection status
Roadway Subsystem	Wayside Equipment	W	hri status
Roadway Subsystem	Wayside Equipment	W	intersection blockage notification
Toll Collection	Vehicle	U2	request tag data
Toll Collection	Vehicle	U2	tag update
Traffic Management	Rail Operations	W	hri advisories
Traffic Management	Roadway Subsystem	W	hri control data
Traffic Management	Roadway Subsystem	W	hri request

Special Constraints

Transit Vehicle	Transit Vehicle Subsystem	W	vehicle measures
Vehicle	Basic Vehicle	W	vehicle control
Vehicle	Other Vehicle	U3	vehicle to vehicle coordination
Vehicle	Roadway Subsystem	U2	AHS vehicle data
Wayside Equipment	Roadway Subsystem	W	arriving train information
Wayside Equipment	Roadway Subsystem	W	track status

2.23.5 Performance (Timing)

Timing is critical. Timing for most ITS communication services is based on the response to a request for data. Because of this, common communication media designed to handle voice data will likely support these requirements. The beacon interface has special requirements of identifying the vehicle as well as exchanging information before the vehicle gets out of range. This is more of a problem with vehicles traveling at speed. The architecture constrains such time critical access to data such that the data is available at the beacon site. This obviates the need for explicit specification of other timing information to support data transfer over a short range beacon.

Source	Destination	Interconnect	Architecture Flow
Commercial Vehicle Check	Commercial Vehicle Subsystem	U2	border clearance event record
Commercial Vehicle Check	Commercial Vehicle Subsystem	U2	border clearance request
Commercial Vehicle Check	Commercial Vehicle Subsystem	U2	clearance event record
Commercial Vehicle Check	Commercial Vehicle Subsystem	U2	lock tag data request
Commercial Vehicle Check	Commercial Vehicle Subsystem	U2	on-board safety request
Commercial Vehicle Check	Commercial Vehicle Subsystem	U2	pass/pull-in
Commercial Vehicle Check	Commercial Vehicle Subsystem	U2	safety inspection record
Commercial Vehicle Check	Commercial Vehicle Subsystem	U2	screening request
Commercial Vehicle Subsystem	Commercial Vehicle Check	U2	border clearance data
Commercial Vehicle Subsystem	Commercial Vehicle Check	U2	lock tag data
Commercial Vehicle Subsystem	Commercial Vehicle Check	U2	on board safety data

Commercial Vehicle Subsystem	Commercial Vehicle Check	U2	screening data
CVO Inspector	Commercial Vehicle Check	H	CVC override mode
Emergency Vehicle Subsystem	Roadway Subsystem	U2	emergency vehicle preemption request
Other Vehicle	Vehicle	U3	vehicle to vehicle coordination
Parking Management	Vehicle	U2	request tag data
Parking Management	Vehicle	U2	tag update
Roadway Subsystem	Driver	H	driver information
Roadway Subsystem	Traffic Management	W	incident data
Roadway Subsystem	Traffic Management	W	intersection blockage notification
Roadway Subsystem	Vehicle	U2	AHS control data
Roadway Subsystem	Vehicle	U2	intersection status
Roadway Subsystem	Vehicle	U2	request tag data
Roadway Subsystem	Vehicle	U2	vehicle signage data
Roadway Subsystem	Wayside Equipment	W	intersection blockage notification
Toll Collection	Vehicle	U2	request tag data
Toll Collection	Vehicle	U2	tag update
Traffic Management	Rail Operations	W	hri advisories
Traffic Management	Roadway Subsystem	W	freeway control data
Traffic Management	Roadway Subsystem	W	signal control data
Traffic Management	Roadway Subsystem	W	surveillance control
Transit Vehicle Subsystem	Roadway Subsystem	U2	local signal priority request
Transit Vehicle Subsystem	Transit Management	U1t,U2	fare and payment status
Transit Vehicle Subsystem	Transit Management	U1t,U2	request for bad tag list
Vehicle	Other Vehicle	U3	vehicle to vehicle coordination
Vehicle	Parking Management	U2	tag data
Vehicle	Roadway Subsystem	U2	AHS vehicle data
Vehicle	Roadway Subsystem	U2	vehicle probe data
Vehicle	Toll Collection	U2	tag data
Wayside Equipment	Roadway Subsystem	W	arriving train information
Wayside Equipment	Roadway Subsystem	W	track status

3. COMMUNICATIONS LAYER

The overall ITS physical architecture consists of three layers, the Transportation Layer, the Communication Layer – which is presented in this section and has wireless and wireline components – and the Institutional Layer. This section presents an overview of the Communication Layer and is divided into two main sections: Communication Architecture (Section 3.1), and Communication Layer linkage to the Transportation Layer (Section 3.2).

The Communication Architecture Section (Section 3.1) presents a generic communication model which illustrates the basic relationship between the ITS Physical Architecture's Transportation and Communication Layers. This generic communication model, which should not be confused with the ITS communication network reference model, is based on the International Standards Organization's (ISO) Open Systems Interconnection (OSI) model. The ISO OSI model consists of seven layers: application, presentation, session, transport, network, data link, and physical layer. In general, the application, presentation, and session layers are supported by the Transportation Layer while the transport, network, data link and physical layers are supported by the Communications Layer.

The Communication Architecture Section (Section 3.1) also provides definitions of the various components that make up the communication layer. Some of these components include: communication services, communication logical functions, communication functional entities, and communication network reference model. The communication network reference model is the primary ITS communication model.

The communication architectures for commercial communication systems such as Personal Communication Services (PCS), Group Special Mobile (GSM), TIA-IS-41, Cellular Digital Packet Data (CDPD), to name a few, use communication network reference models. A network reference model is used to identify physical equipment that perform communication functions, and is used to identify reference interfaces between these physical equipment (standards are usually written for these reference interfaces). The ITS network reference model is based on, and presents extensions of, several reference models that were developed for the above mentioned standard communication systems. The model provides a structure that shows how various communication technologies can implement the ITS Architecture Interconnect Diagrams (AIDs), which are presented later in the Communication Layer Linkage Section (Section 3.2).

The Communication Layer Linkage Section also identifies the relationship between the Transportation Layer and Communication Layer definitions. This is accomplished through the following steps:

1. Mapping the communication services to the data flows identified in the Transportation Layer.

2. Generating the Architecture Interconnect Diagrams (AIDs) which define the interconnections between transportation subsystems and modules defined in the Transportation Layer.

3. Identifying the Architecture Renditions (ARs) which are examples, based on the network reference model, of how to provide communication connections between users defined in the Transportation Layer.

4. Mapping of the AIDs to the AR’s (each AR stays one level above technology specification, and comprises a family of systems with similar attributes, e.g., wireless packet data networks).

5. Identifying the Architecture Interconnect Specifications (AISs) which are examples of specific systems to implement an applicable communication technology to a particular rendition, for example, the use of CDPD for cellular wide-area wireless data communication.

To summarize, the Communication Layer Linkage Section presents the communication services/data flow mapping, AIDs, ARs, AID/AR mapping, and AISs.

In general, the Communication architecture for ITS will have two components: one wireless and one wireline. All Transportation Layer entities requiring information transfer are supported by one, or both, of these components. In most cases, the wireless component merely provides a tetherless user, usually one in a vehicle, with access to fixed (or wireline) network resources. The wireless portion will be manifested in three different ways:

- Wide-area wireless infrastructure supporting wide-area information transfer (many data flows). For example, the direct use of existing and emerging mobile wireless systems.

- Short range wireless infrastructure for short-range information transfer (also many data flows, but limited to specific applications), similar to systems used for electronic toll collection.

- Dedicated wireless system handling high data rate, low probability of error, fairly short range, Advanced Highway Systems related (AHS-related) data flows, such as vehicle to vehicle transceiver radio systems.

Because of the variances in the ITS user service requirements (from a communication perspective), it is clear from a cursory examination that the user services do not share a common information transfer capability. Specifically, ITS user services like electronic toll collection demand communication needs that can only be met by dedicated infrastructures for technical and feasibility, notwithstanding institutional, reasons. The ITS user services information transfer needs are supported by a sample deployment of the communication network reference model described in Section 3.1.4. Implementation candidates are identified as a result of a broad, balanced communication technology assessment task. After examining the assessment results for these candidates, an ITS implement or service provider can decide on the mix of communication technologies that are best suited to the implementation scenario at hand.

The wireline portion can be manifested in many different ways, most implementation dependent. Note that in defining the Communication Layer, no assumptions have been made regarding media type.

The process of developing the communications layer (architecture) is illustrated in Figure 3.0-1, and starts from the data flows in the transportation layer. In the following sections, the reader is referred to this figure at each step of the design process description.

The upper left block in Figure 3.0-1 shows the mapping of the identified data flows to communication services. The data flows are derived from the Architecture Flow Diagram (AFD) provided in the Physical Architecture document, which is used to specify which transportation subsystems communicate directly with each other. The communication services are described in Section 3.2.1 in terms of flow response and capabilities (they should not be confused with the ITS user services, which from a communication standpoint, are applications, as will be discussed shortly.) The mapping provides one or more communication services for each of the data flows between transportation entities. The Architecture Interconnect Diagrams (AIDs) encapsulate the type of partition between each of the transportation layer subsystems, as wireline or wireless, accompanied by a description of the communication service and operation mode for all the data flows between each pair of entities.

In parallel, a Network Reference Model (lower left block in Figure 3.0-1) is derived from models for standard commercial communication systems to fit ITS needs. This communication model is then used, in combination with feasibility and cost constraints, to develop renditions, or examples, of how to realize the required communication services. These renditions are based on the communication interface type, and are at one level above specific technology.

As shown in Figure 3.0-1, the MAP AID-AR block is done in an abstract way, identifying which data flows are supported by each rendition. At the same time, the results of the Technology Assessment are used to develop Architecture Interconnect Specifications, which identify and assess specific features of the technology important to interconnecting the transportation layer entities. The AIS involves further specification of the renditions, and completes the description of the ITS Communications Architecture. To illustrate, mobile wireless packet data networks are considered a rendition. Several technologies, like CDPD, RAM, and so on, are specific technologies that belong to this rendition that could be used in the implementation. The AIS Section here is maintained brief, and includes a few examples of the results of the communication technology assessment (from an ITS architecture standpoint) which is presented in Chapter 7 and Appendix D of the Communications Document. The AIS leads to technology recommendations, to be interpreted as implementation examples of the communication elements in the ITS architecture. In a real-world ITS implementation, this last step would be performed by the communication system designer.

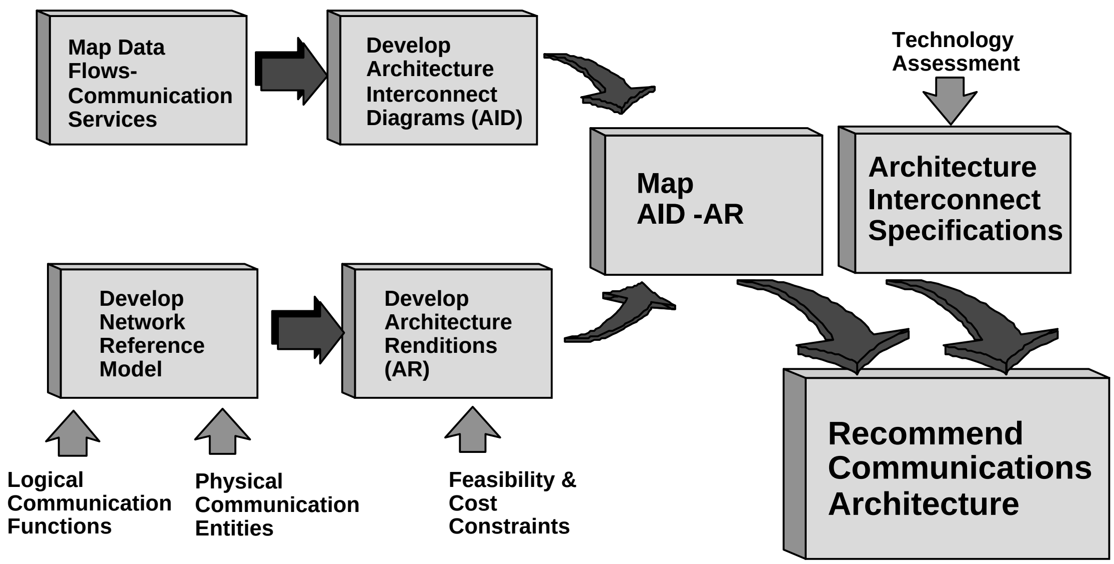


Figure 3.0–1 Communications Architecture Design Process

3.1 Communication Architecture

The generic communication hierarchical model presented in Figure 3.1-1 shows the relationship between the Transportation and Communication Layers. Each data user can be one entity in the Transportation Layer (e.g., the Information Service Provider Subsystem or Personal Vehicle Subsystem in an information exchange). The user does not care about and should not be concerned with the specifics of this information transfer layer. In fact, the Communication Layer can be viewed as plumbing that carries information from one user to another.

The complex makeup of the network is usually defined by system architectures developed to meet specific requirements, performance objectives, and socio-economic drivers. In the absence of crisp specifications and because of the jurisdictional-independence of this particular architecture, the end framework precludes the design of low level implementation details. However, to properly evaluate the communication architecture candidates, select technologies and detailed designs are recommended in an evaluatory design (see the later chapters of the Communications Document.)

The generic hierarchical communication model shown in Figure 3.1-1 follows the Open Systems Interconnection model which organizes the communication network in a highly structured format to reduce its overall design complexity. This model is structured as a series of layers each with the function of providing certain services to the layer above and capable of conversing with the corresponding layer at the other end of the link. Thus the high level layers (e.g. ITS application) are shielded from the actual implementation details of the communication services. Different networks can use layers different from the OSI model, such as the IBM SNA (Systems Network Architecture). When different protocols are used in different networks, an inter working function must provide the conversion between the protocols at the various levels.

The lowest layer in the OSI model is the physical layer (layer 1), which provides the transmission of bits over wires or radio links . Layer 2 is the data link layer, and is concerned with making the link appear to the receiver as bit error-free as possible by implementing error detection and correction (EDAC) coding schemes in the transceiver; one example is the use of a cyclic redundancy code (CRC) to a block or frame of the data and when the data passes the CRC check at the receiver, the returned acknowledgment indicates whether re-transmission is needed. Layer 3 is the network layer, which controls the operation of the network, where the key issue is routing packets, which is also used to generate billing information for the communications service provider; billing is tied to IP addresses. Layer 4 is the transport layer, which mediates between the session layer and the network layer, providing end-to-end accounting for all the data at the receiving end, and isolates the system from the changing physical technologies. Layer 5 is the session layer, which allows users on different machines to establish communications, or sessions, between them, involving ordinary data transport but with enhanced services such as remote log-in or file transfer. Layer 6, the presentation layer, performs syntax and semantic operations on the information transmitted between the users, such as encoding data in a standard way, or compressing or encrypting that data. Layer 7 is the application layer, which provides commonly used protocols for such tasks as terminal emulation, file transfer, electronic mail and remote job entry. (Note that for many ITS applications, layers 5 and 6 are absorbed into the application layer, layer 7.)

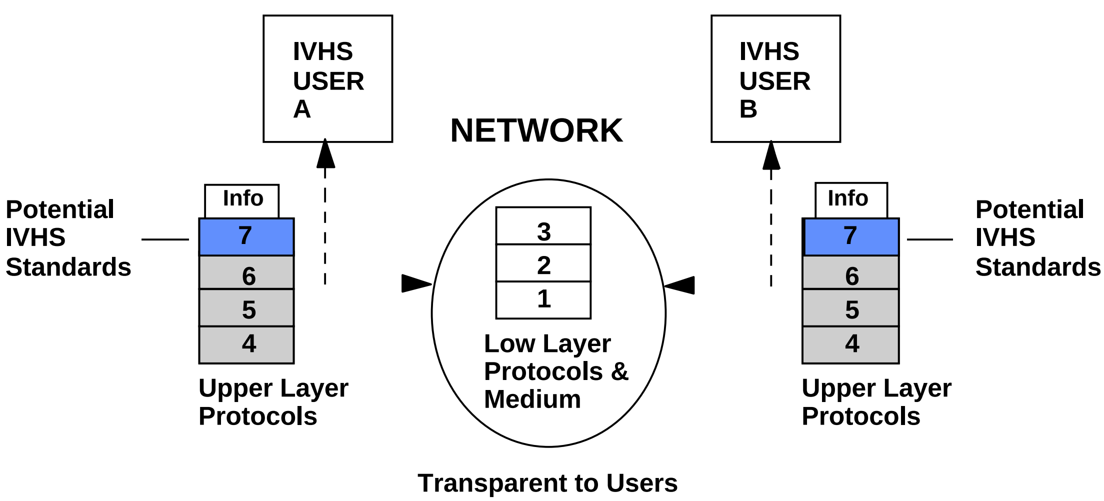


Figure 3.1-1 Generic Hierarchical Communication Model

From the Communication Layer perspective, the term "services" is defined according to communications governing bodies (*e.g.*, ITU, TIA, etc.), and should be used with care. That is, when describing a communications architecture, one should not refer to Route Guidance or Pre-trip Planning as services. Rather, they are applications in need of a communication service. Elaborating more along these lines, ITS appears to the Communication Layer as a collection of applications with markedly different communication requirements. Thus the service provided by the communication model is characterized more by 1) the application's directionality requirements (*e.g.*, one-way or two-way) for information transport, 2) whether it is between mobile elements, mobile and stationary elements or stationary elements, 3) the amounts of data to be transported, and 4) the urgency rather than the precise description as Route Guidance or Pre-trip Planning.

The next section identifies various communication services to which the Transportation Layer data flows can be matched. This matching process will assign broad generic communication services to the data flows without specifying a particular technology.

3.1.1 Communication Services

The communication services define the exchange of information between two points and are independent of media and application (i.e., ITS user service). In essence, they are a specified set of user-information transfer capabilities provided by the communication layer to a user in the transportation layer. Figure 3.1-2 illustrates the hierarchy of communication services, the detailed of these is given in Appendix A-1 in the Communications Document. In what follows a brief description of the services is presented.

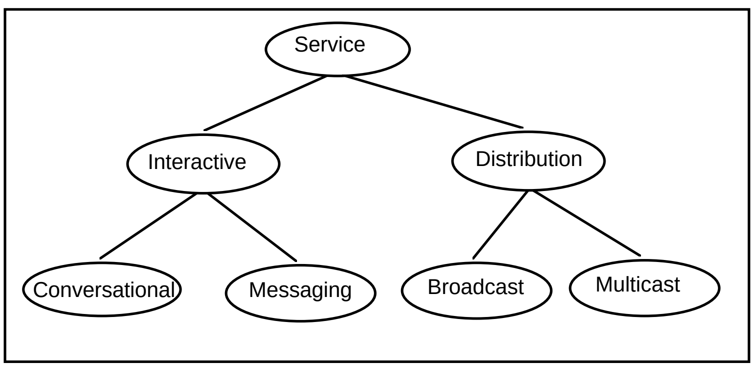


Figure 3.1-2 Communication Services Hierarchy

Communication services consist of two broad categories, interactive and distribution. Interactive services allow the user to exchange data with other users or providers in real or near real time, asking for service or information and receiving it in the time it takes to communicate or look up the information. Distribution services allow the user to send the same message to multiple other users.

Interactive services may be either conversational or messaging. Conversational implies the use of a two-way connection established before information exchange begins and terminated when the exchange is completed. Messaging, on the other hand, works more like electronic mail being exchanged between users. The messages are exchanged without establishing a dedicated path between the two sites. Each message is addressed and placed on the network for transmission, intermixed with messages from other users. The communications community labels this mode of communication a “datagram” service.

Distribution services may be either broadcast or multicast and may be used over wireline and/or wireless communication links. Broadcast messages are those sent to all users while multicast messages are sent only to a subset of users. Multicast differs from broadcast in its use of a designated address for all users and user groups. Examples of broadcast information might include current weather or road conditions, whereas multicast information might be information sent to all drivers working for a specific company. A changing group membership could be the set of users traveling between two locations or with a certain destination, for which unique information must be transmitted. The services that can be supported using circuit or packet connection mode include voice, video, image and data. (see Appendix A-1 of the Communications Document for a complete description.)

Not shown in the Figure 3.1-2 are location services. These fall in two categories: (1) the services that do not use the communication network (i.e., GPS, and stand alone terrestrial systems); (2) location services that use the network for providing the service (e.g., cellular based systems). In the latter case, the location services fall under the interactive services. The service will be rendered by a service provider in response to a request for information or help.

3.1.2 Logical Communication Functions

Based on the objectives of the communication architecture, a list of logical functions to support the ITS system communication requirements are identified. The primary logical communication functions can be confined to: wireless and wireline access, switching, routing, registration authentication, inter working, validation, billing, and operations (see Appendix A-2 of the Communications Document for a detailed description.).

3.1.3 Functional Entities

The functional entities that make up the communication layer were derived from existing and emerging infrastructure specifications and standards (*e.g.*, TIA, ITU, Bellcore, ANSI). These basic building blocks form the foundation of a generic communication system. As with the transportation layer, each functional entity consists of one or more logical functions. These entities include: 1) user device, 2) user profile module, 3) switch, 4) wireless controller, 5) wireless base station, 6) inter working platform, 7) profile data base, and 8) wireline network. The detail description of these functional entities is presented in Appendix A-3 of the Communications Document.

3.1.4 Communication Network Reference Model

As shown previously in Figure 3.0-1, the communication architecture design process consists of several steps. The previous sections listed the communication logical functions and physical entities. The architecture design process now starts on the lower leg of Figure 3.0-1 with the development of the Communication Network Reference Model. This model provides an architecture or structure that shows how various communication technologies can implement the Architecture Interconnect Diagrams developed in the next section.

The network reference model for ITS is depicted in Figure 3.1-3, and is a generic abstraction which builds upon several reference models developed for standard commercial systems. Boxes represent the various physical equipment (with descriptive uppercase letters) that perform the communication functions. Identified by lowercase letters (s, v, u_1, u_2, u_3) are the interfaces important to ITS. They are described in the following paragraphs.

The most important reference point is the wireless interface (*u*) connecting the WBS and the wireless transceiver. To meet the objectives of the national ITS Architecture it will be necessary in some cases that the air interface become standard. The wireless portion of the architecture is manifested in 3 different ways. The *u* interface is realized in three ways: u_1, u_2, u_3, with each interface corresponding to one of the wireless manifestations, as follows:

- u_1 defines the wide area wireless air-link with one of a set of base stations providing connections to mobile or mobile or untethered users. It is typified by the current cellular telephone and data networks or the larger cells of Specialized Mobile Radio for two way communication, as well as paging and broadcast systems.

- u_2 defines the short-range air-link used for close-proximity (less than 50–100 feet) transmissions between a mobile user and a base station, typified by transfers of vehicle identification numbers at toll booths; and

- u_3 addresses the vehicle-vehicle (AHS-type) air-link, for high data rate, burst, usually line-of-sight transmission with high reliability between vehicles, where standards are in their infancy.

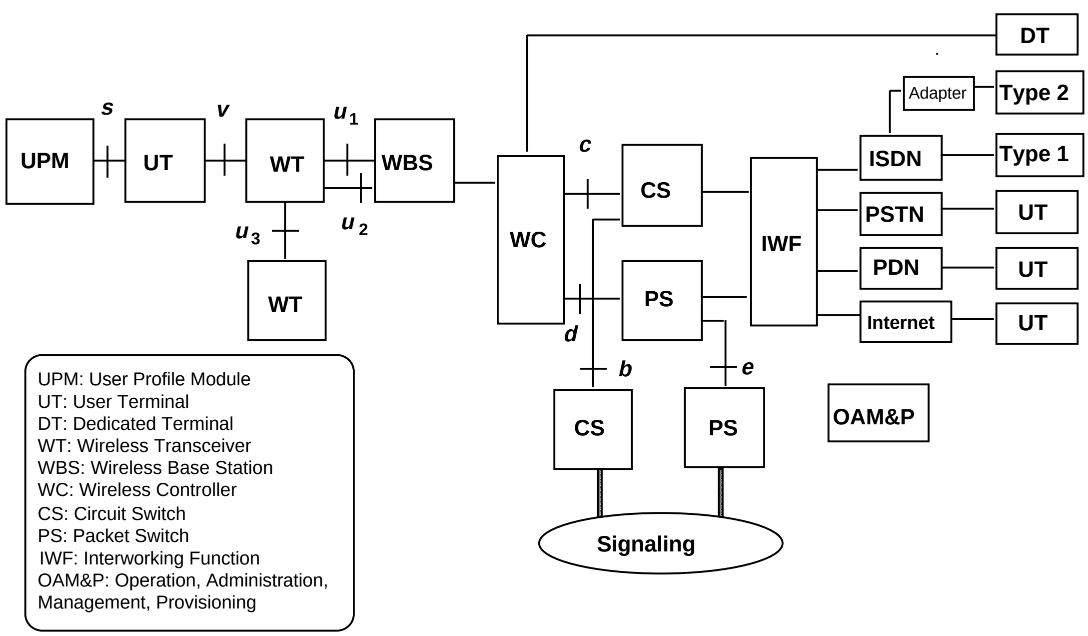


Figure 3.1.-3 Network Reference Model for the Communications Layer

The National ITS architecture provides for implementation flexibility. Various of the data flows in the Architecture can be carried over multiple of these interfaces, and the final choices would be made by the local implementers. This flexibility is depicted conceptually in Figure 3.1.-4.

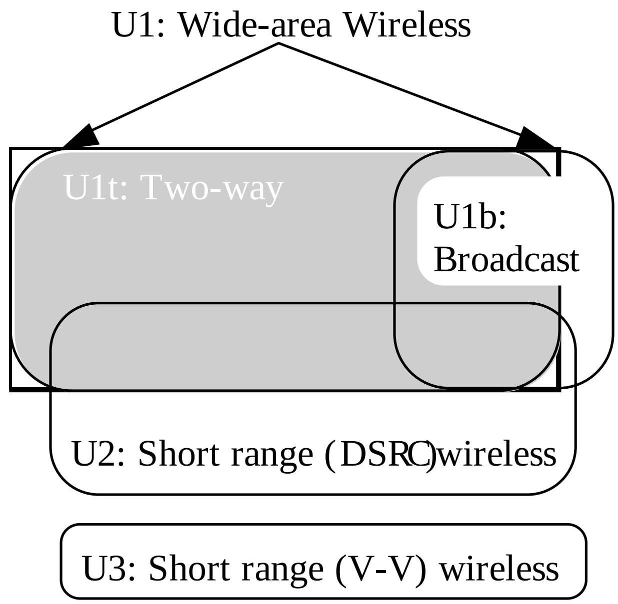


Figure 3.1-4 Implementation Flexibility

Since the wireline segment encompasses standard wireline configurations, the ITS-critical elements from a standards perspective are those comprising the wireless portion on the left side of Figure 3.1-3. The wireless portion consists of the User Profile Module (UPM), the User Terminal (UT), the Wireless Transceiver (WT) and the Wireless Base Station (WBS). The connections through the Dedicated Terminal and various User Terminals are shown in the column of boxes on the right. The equipment in the center is the existing public telecommunications services, so the details are transparent to ITS, which is a major benefit to the ITS community. *All management, operations, expansion, and improvement costs are shared with the wider set of all telecommunications users.*

This is an important point to jurisdictions and agencies who prefer to procure and trench their own network along the right-of-way. Whereas a financial sensitivity analysis may point to a private solution, it frequently does not fully consider the large and sustained Operation, Administration, Management , and Provisioning (OAM&P) fees that the agency will have to pay the telecommunications vendor during the system's life cycle.

Appendix A-4 of the Communications Document presents a detailed description of the wireline side of the above network reference model, in addition to a more thorough treatment for the required interfaces, such as switches, controllers, and terminals. This appendix also presents the network entities, interfaces, and signaling plane, and includes a discussion on circuit connection and data packet transmission.

3.2 Communication Layer Linkage

This Communication Layer Linkage Section further identifies the relationship between the Transportation Layer and Communication Layer definitions. This is accomplished by mapping the communication services to the data flows identified in the Transportation Layer, generating the Architecture Interconnect Diagrams (AIDs), identifying the Architecture Renditions (ARs), mapping the AIDs to the ARs, finally identifying the Architecture Interconnect Specifications (AISs) (based on the technology assessment).

3.2.1 Mapping Communication Services to Data Flows

Mapping of the communication services to the data flows establishes the first link between the transportation layer and the communication layer, and this initial link depends on the completion of two technical architecture milestones. First, the message sizes and data transfer requirements are broadly identified. Second, the physical architecture that allocates logical functions (see Logical Architecture Document) to subsystems necessitates a partitioning exercise, which defines the data flows that require communication. This mapping is an iterative procedure, calibrated by feedback from the logical and physical architectures (and in turn the ITS stakeholders) by retracing the steps shown in Figure 3.0-1.

Appendix A-5 of the Communications Document details the mapping process. It also depicts the assigned communication service for each data flow with the corresponding rationale.

3.2.2 Architecture Interconnect Diagrams

As denoted in Figure 3.1-1, this section presents the development of the Architecture Interconnect Diagrams (AIDs). These diagrams show the subsystem-to-subsystem communication interfaces of all transportation subsystem entities (defined in the transportation part of the Physical Architecture). The diagrams identify the communication mode and partition, either wireline or one of three types of wireless connection, as well as documenting the rationale for of these choices when needed for clarification. The diagrams identify the requirements, developed from the physical relationships of the various subsystem entities, but do not force any specific communication technology to be used. The information contained in the AIDs can be traced to the information provided in the Data Flow-Communications Service Mapping Table (Appendix A-5 of the Communications Document).

A template is used to illustrate the interconnections between entities and between modules and is described next. At this stage in the physical architecture, no AIDs are defined for inter-module information transfer within a simple entity. In fact, from the communication layer perspective, this is not necessary. The most important goal is to identify the inter-entity interconnectivity.

The subsections that follow describe the AID template, present the Level 1 and Level 0 (top level) AID’s.

3.2.2.1 AID Template

As depicted in Figure 3.2-1, each AID shows the two communicating transportation subsystem entities, the interconnection partition (i.e., wireline, wireless, or both), and a characterization of the interconnection. The latter is not a link-specific description, which the AIS provides, but a high-level interpretation in terms of services and operation modes. When not obvious, the choice of operation mode is based on the rationale provided in Table 3.3-1. The interconnect description for each AID provides a data flow, service and operation mode description for each data flow between the two entities. The Data Flow information also provides directionality when more than one data flow exists between the entities, not all of which are in the same direction. If all the data flows are in the same direction, no indication is given and the data flows from the left entity to the one on the right.

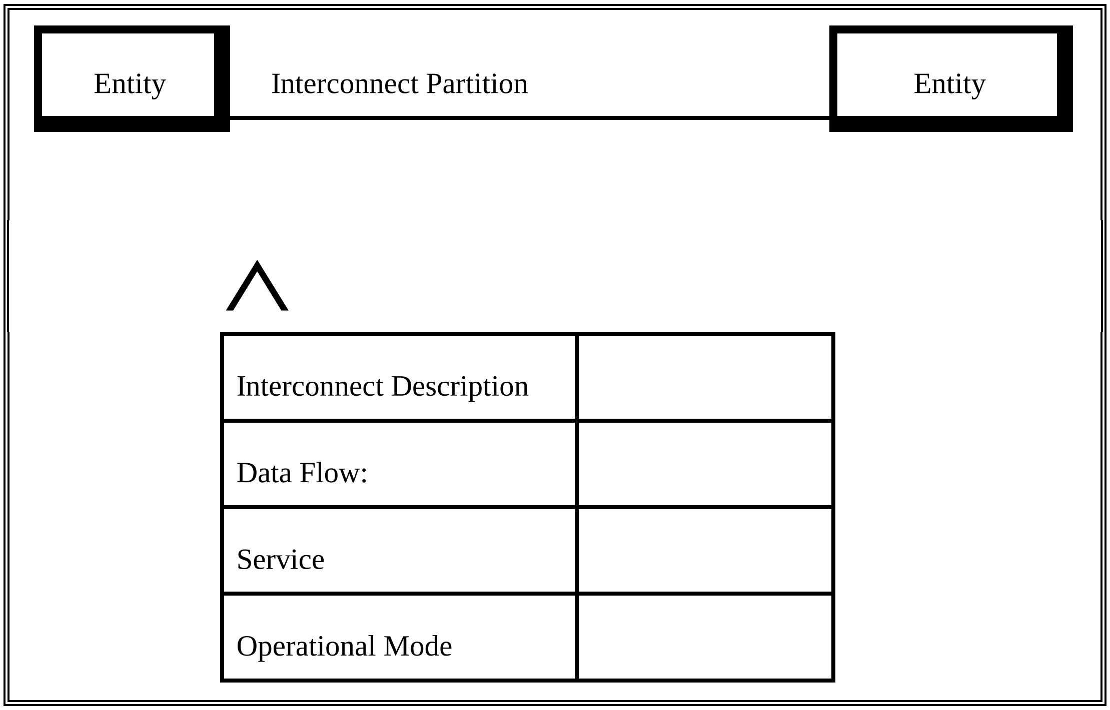
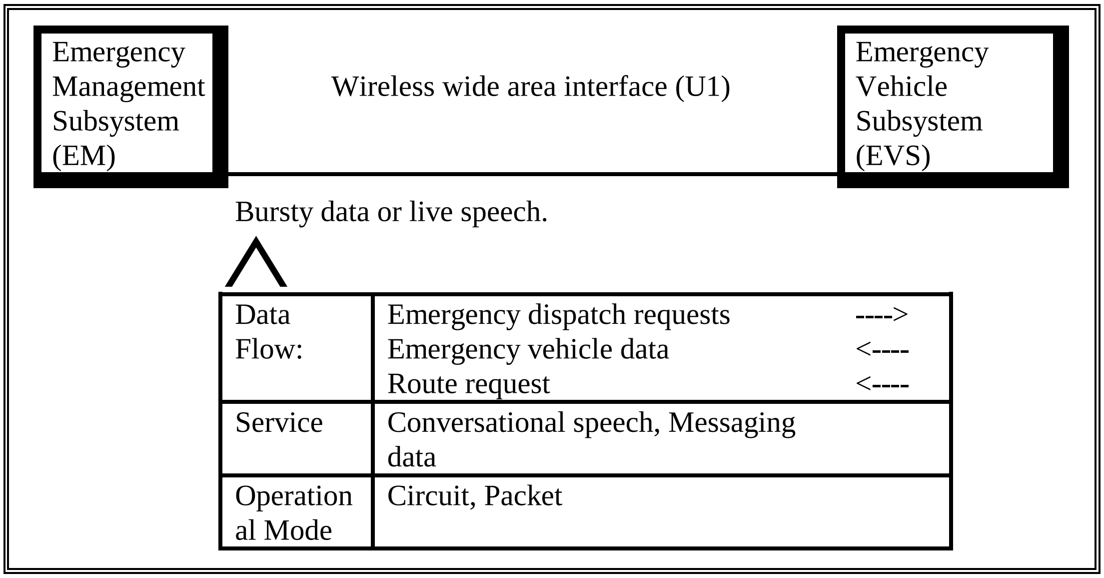


Figure 3.2-1. Template for the Architecture Interconnect Diagram (AID)

3.2.2.2 Level 1 AIDs

Using the AID template and Table A.5-1, Data Flow – Communication Services Mapping Table, the data flows are represented in an Architecture Interconnect Diagram (AID) format. A single example is presented here in Figure 3.2-2, and various others are compiled in Appendix B of the Communications Document.



Data Flow:	Emergency dispatch requests Emergency vehicle data Route request	----> <---- <----
Service	Conversational speech, Messaging data	
Operational Mode	Circuit, Packet	

Figure 3.2.-2 Example of AID Level-1

3.2.2.3 Level 0 AID

The Level 0 Architecture Interconnect Diagram shows all communications connectivity required by the Physical Architecture. Several different versions of these diagrams have been generated during the course of architecture development. Figure 3.2-3 is a comprehensive interconnect diagram that shows all interconnects between subsystems as well as between subsystems and terminators. This results in a figure that has over 150 connections between more than 70 entities. The figure has been filtered to remove human interconnects and physical/environmental interconnects since these are not the focus of the communications layer. Figure 3.2-4 shows only the interconnects between subsystems in the architecture which are the primary focus of the communications analysis. This figure is a percolation to a top level of all the detailed, level 1 AID’s. It presents all the interfaces between the physical subsystem entities, capturing the wireline (w) or wireless (u_1, u_2, or u_3) nature of the interfaces in the ITS architecture. As such, it is a comprehensive, albeit not complete, representation of the ITS communication architecture. More detailed variations can be easily derived from it. For example, Figure 3.2-5 shows the data flows using the U1b wide area wireless broadcast “sub-interface”. Figure 3.2-6 shows the subset that uses either of U1t (two-way wide area wireless) or U1b (wide area wireless broadcast). Note that U1b does not imply a certain technology – FM subcarrier, paging, messaging data networks are possible implementations; they all tend to use a broadcast protocol in the forward, i.e.; fixed to mobile, direction.

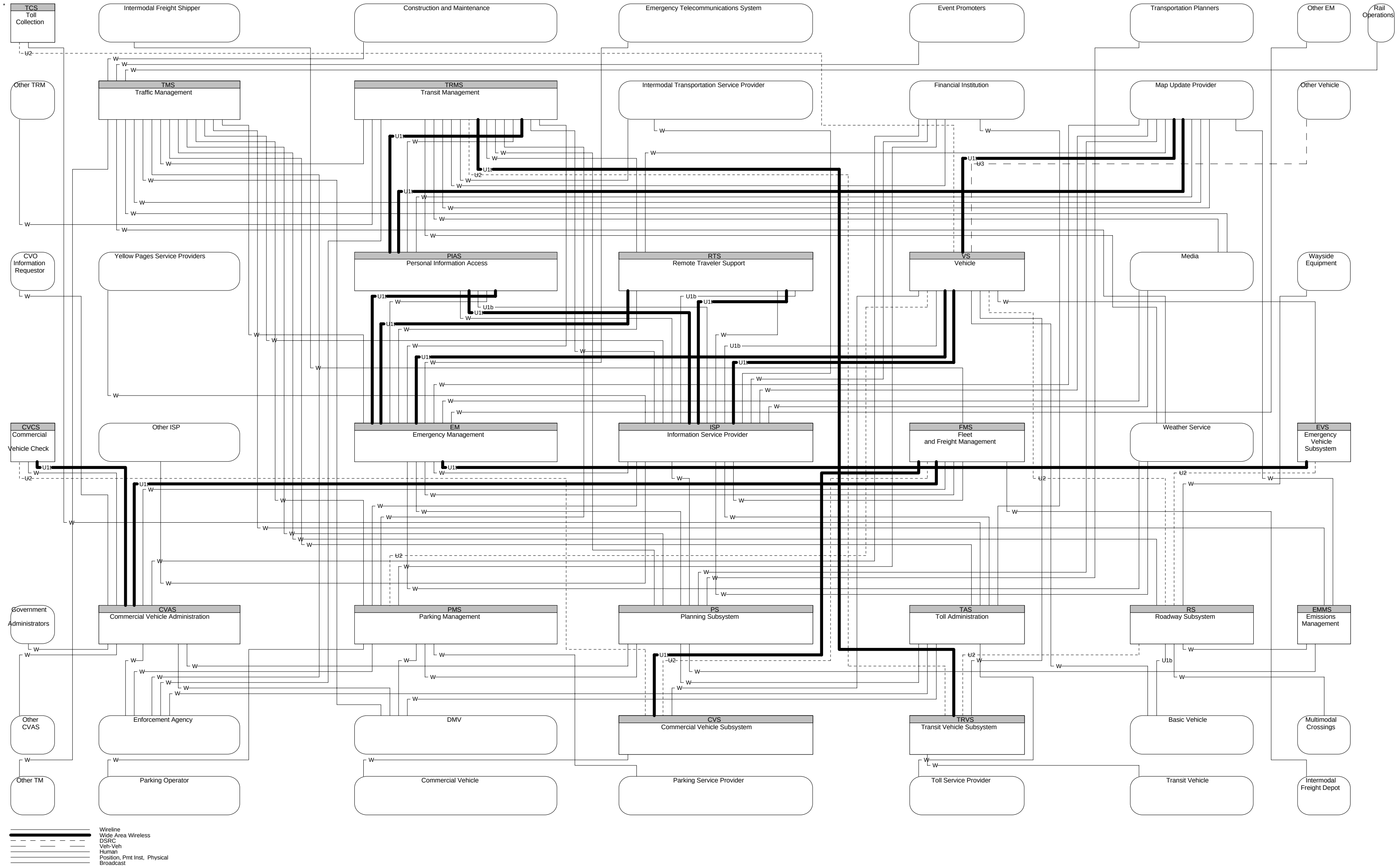


Figure 3.2-3 Level 0 Architecture Interconnect Diagram for the National ITS Architecture
(Interconnects Between Subsystems & Between Subsystems And Terminators)

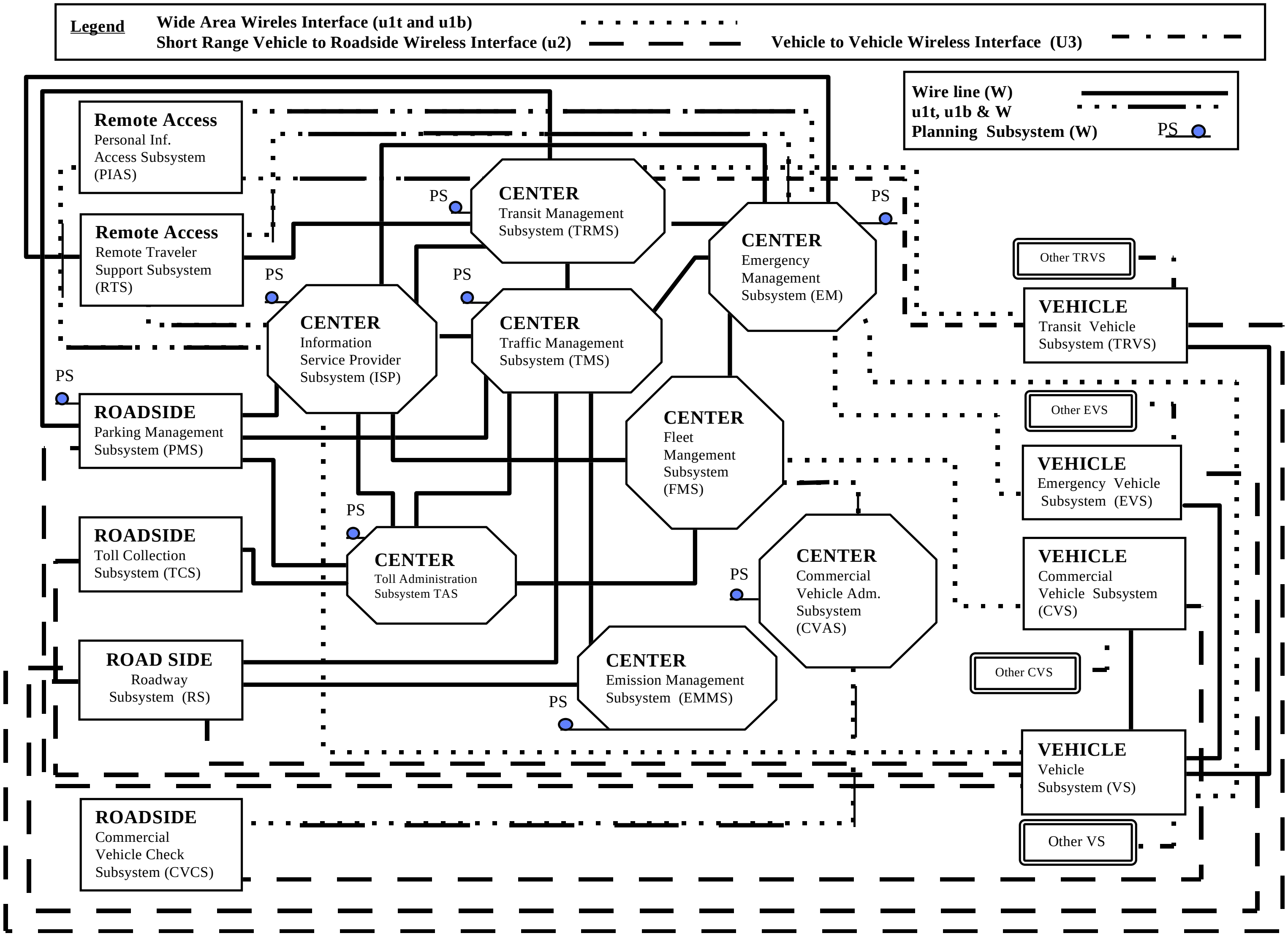


Figure 3.2-4 Level 0 Architecture Interconnect Diagram for the National ITS Architecture
(Interconnects Between Subsystems Only)

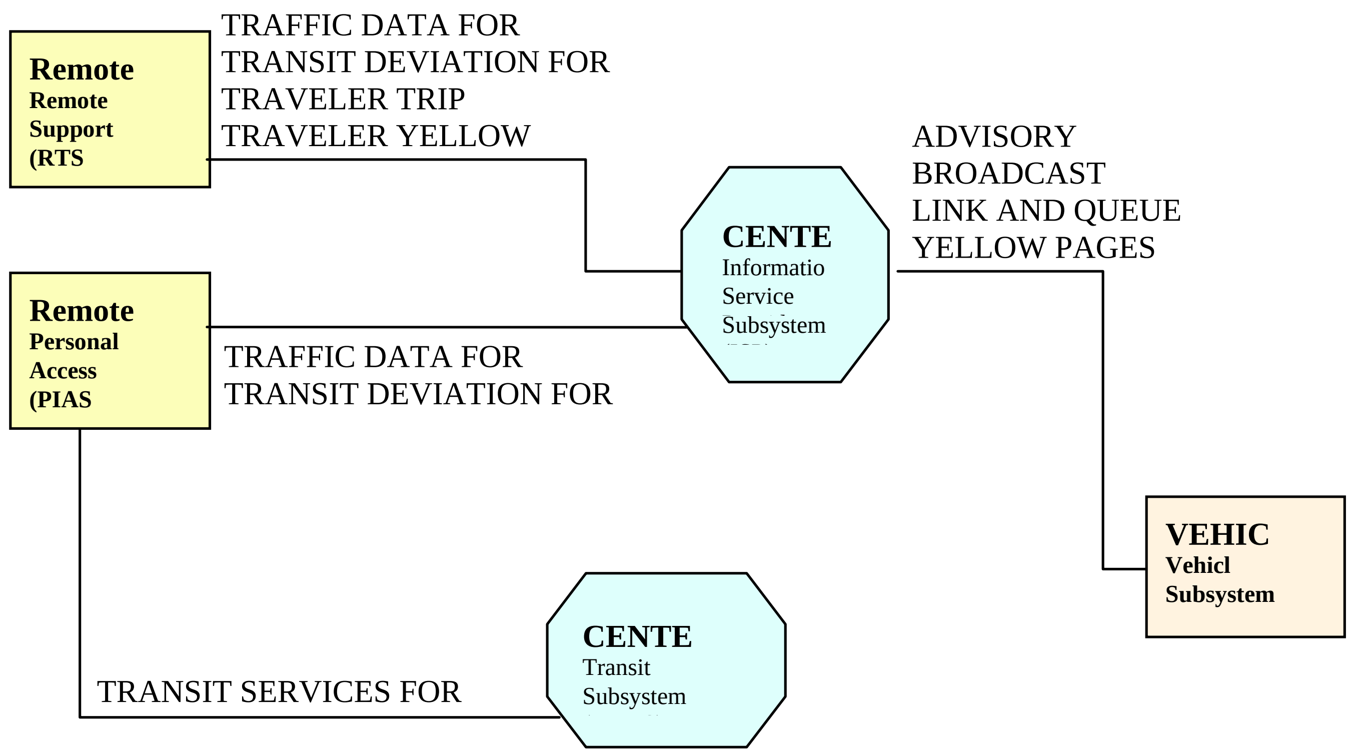


Figure 3.2-5 Level 0 Architecture Interconnect Diagram for the National ITS Architecture
(Subset showing U1b data flows)

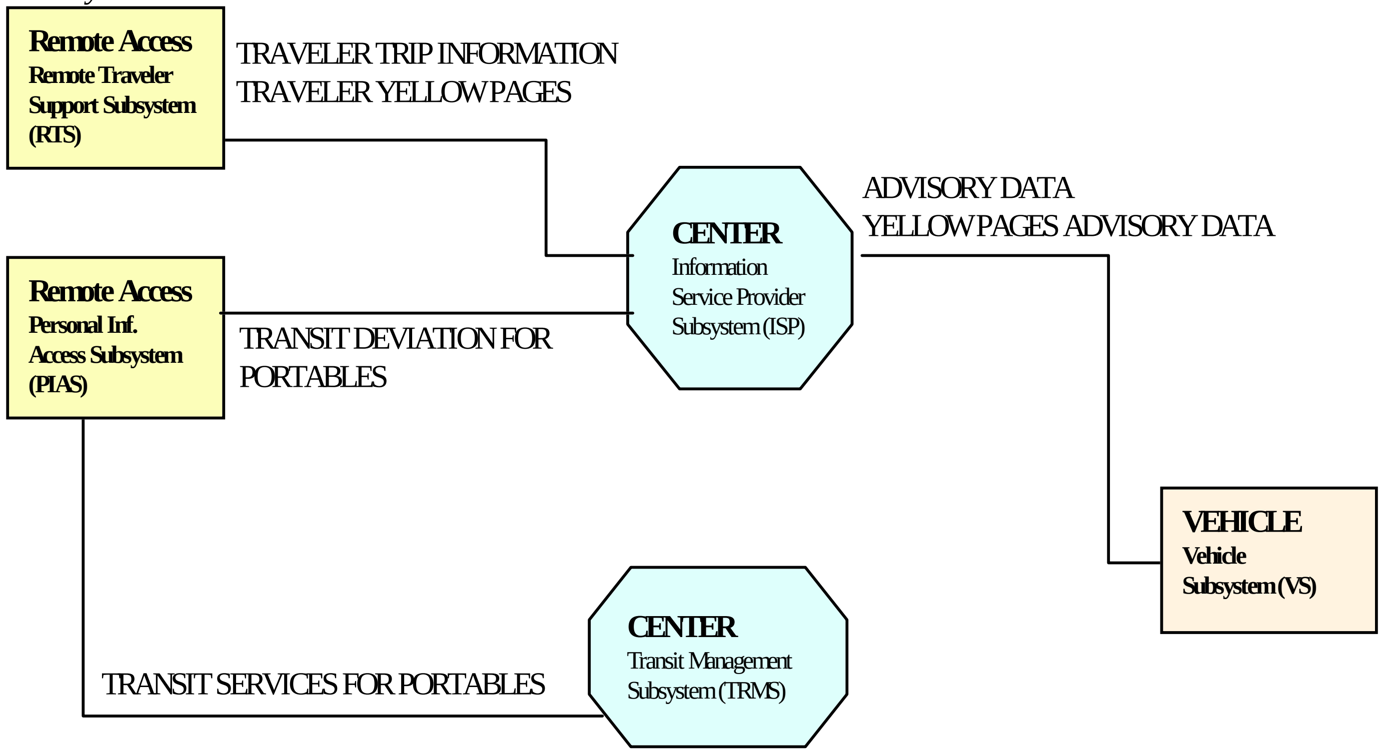


Figure 3.2-6 Level 0 Architecture Interconnect Diagram for the National ITS Architecture;

(Subset showing data flows using either U1t or U1b)

3.2.3 Architecture Renditions

The next step in the communications architecture design process is the development of the communication Architecture Renditions, as depicted in Figure 3.1-1. Combining elements from the Generic Communication Model (Figure 3.1-2) and the ITS Communication Network Reference Model (Figure 3.1-4) provides a more detailed view of the flow of information between two users. This information includes communication services and operational modes (i.e., circuit switched, packet switched, etc.). The architecture renditions are essentially examples of how to provide connections between users based on the communications network reference model and the evaluations of classes of feasible implementations.

Two levels of renditions are generated. A Level 1 rendition is generated for each of the possible interconnections between services. The Level 0 Rendition (the top level) shows the full connectivity between users over multiple links. The details of the renditions, how they are generated, and those that apply to the different interconnections in the architecture are provided in Appendix C of the Communications Document; an example of a Level-1 rendition and the Level-0 will be provided here to support the subsequent task of AIS generation.

3.2.3.1 Level 1 Rendition

Figure 3.2-6 depicts level 1 renditions for the wide-area wireless communication link (u1) through switched networks. This figure depicts interconnection between tetherless users or tetherless and stationary users, utilizing two distinct classes of wide-area wireless technologies. Several technologies or systems can fit within each rendition. For example, CDPD, RAM, ARDIS and so on are possibilities for implementing the packet-switched wireless data network (shown on the right-hand-side of the diagram in Figure 3.2-7.

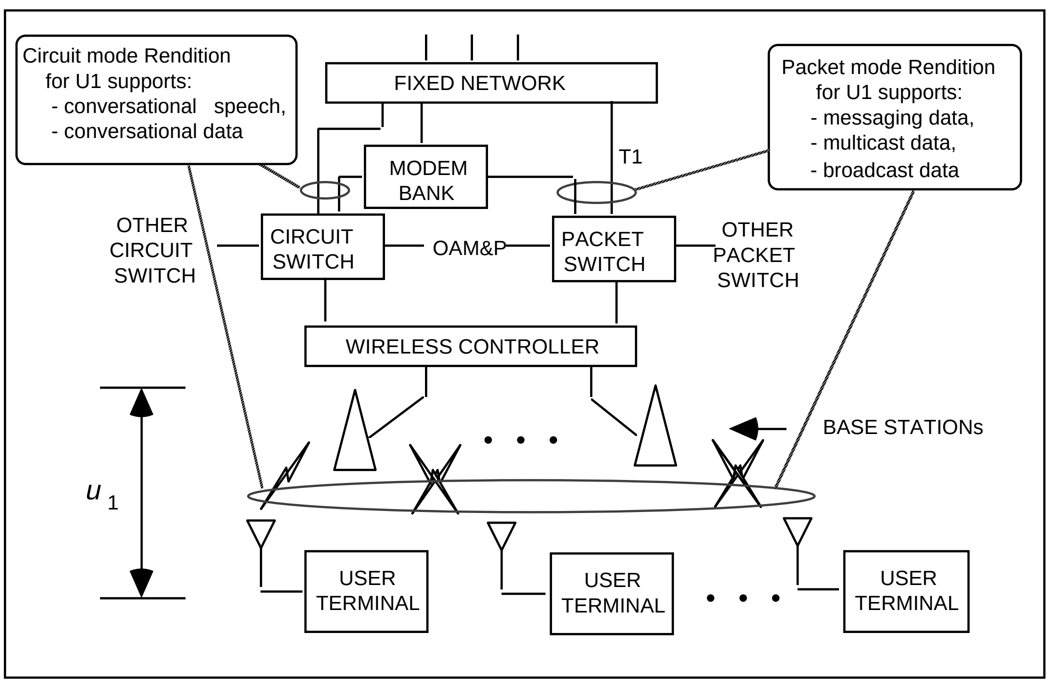


Figure 3.2-7 Rendition 1 — Wide-Area Wireless (u1) Link Through Switched Networks

3.2.3.2 Level 0 Rendition

Figure 3.2-8 illustrates the Level 0 rendition. It represents a composition of all the renditions to reflect the combined needs of the architecture. This rendition shows a user communicating to another user, central office or a base station over various communication links such as u1, u2, u3 and w. Again, the details of this mapping are provided in Appendix C.1.1 of the Communications Document.

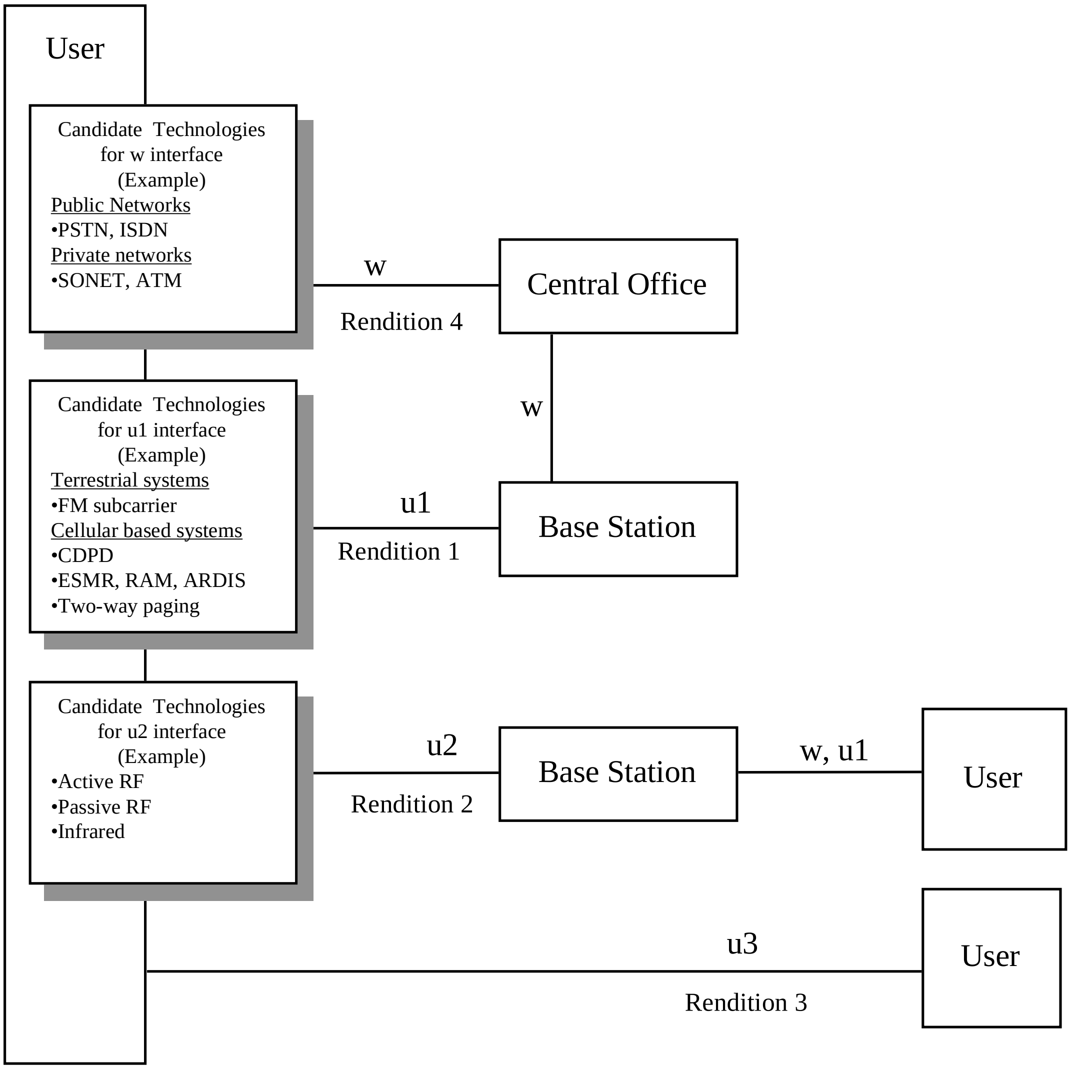


Figure 3.2-8 Level 0 Rendition

3.2.4 Architecture Interconnect Specifications

The Architecture Interconnect Specifications (AISs) are now developed from the technology assessment and refined by combining the renditions with applicable technologies and evaluating the achievable performance (Figure 3.0-1). This involves mapping the applicable communication technologies to the renditions.

To facilitate the mapping of the communication technologies to the renditions, the candidate wireless and wireline technologies are surveyed. The candidate technologies are further assessed and their performance is evaluated from the National ITS Architecture standpoint. For example, the assessment includes: short range and wide area, one-way and two-way wireless data communication. Systems analyzed include terrestrial networks (e.g., cellular, ESMR), FM broadcast, and satellite systems for mobile and fixed services. The details of this survey are presented in Appendix D of the Communications Document, with the assessment results summarized in Chapter 7 of the Communications Document.

The results of this assessment are used in identifying the candidate technologies to support level 0 and level 1 renditions. The results of this mapping are summarized below.

It is apparent from the matrices provided the Technology Assessment Section (Section 7.5 of the Communications Document) that for the foreseeable future, wireless data networks (such as CDPD, RAM, etc.) form the class of communication systems most suitable to interactive wide area wireless ITS links (u1t). The infrastructure is already largely available (short, in some areas, of adding the appliqué equipment). Service costs are already low, and equipment costs are coming down. Coverage nationally is excellent with the possible shortcoming that it may not for some tine be available in rural and remote areas. Yet with the advent of innovative solutions like circuit-switched CDPD, which utilizes the AMPS cellular infrastructure in a manner transparent to a CDPD subscriber, this problem would be largely mitigated. In any event, for ITS users how insist on uninterrupted coverage in remote areas, holes in the coverage of terrestrial cell-based systems can be supplemented by satellite communication systems.

Figure 3.2-9 depicts the use of CDPD in a u1 communication architecture rendition to create an example of Architecture Interconnect Specification.

According to internal market research and analysis, users are concerned with two overshadowing factors: cost and quality. The Architecture Development Team believes that the market will determine the winning technologies which will gain wide scale acceptance.

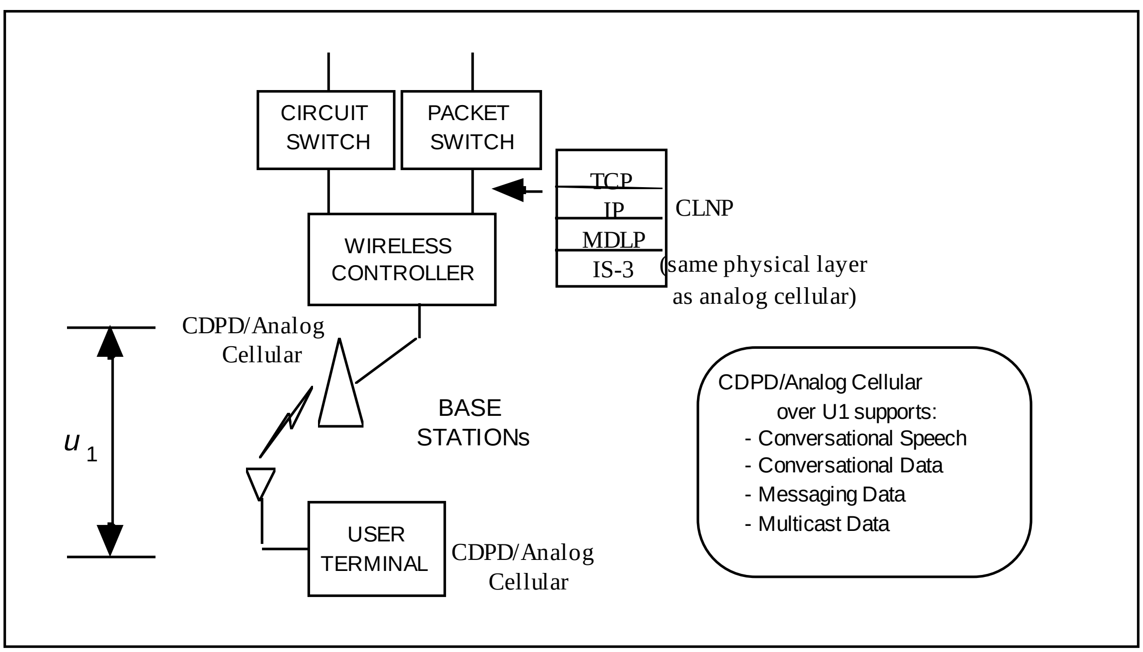


Figure 3.2-9 AIS Example Using CDPD for Wide Area Wireless (u1)

Communications Layer

Table 3.2-1 provides an illustration of candidate technologies for the different wireless data flows within the context of the communication layer of the ITS Architecture. In addition to wireless data networks, for wide area ITS data flows, the short range wireless interface u_2 comprises a distinct set of communication services and supporting radio technologies. In a real world ITS implementation, the system designer makes use of the technology assessments of Chapter 7 of the Communications Document, and the ITS architecture communication renditions presented above, to select the specific communication systems/technologies most appropriate for the deployment at hand.

Table 3.2-1 Examples of Candidate Technologies for Wireless Data Flows

wide-area wireless interface (messaging services; bursty data transfers)		
Source	**Architecture Flow**	**Destination**
CVAS	**Electronic credentials**	FMS
CVAS	**Safety information**	CVCS
CVS	**Driver & vehicle information**	FMS
CVS	**On board vehicle data**	FMS
EM	**emergency dispatch requests**	EVS
EM	**emergency acknowledge**	VS
EM	**emergency acknowledge**	RTS
EM	**emergency acknowledge**	PIAS
EVS	**Emergency vehicle driver status update**	EM
EVS	**Emergency vehicle driver input**	EM
EVS	**Emergency vehicle dispatch acknowledge**	EM
FMS	**fleet to driver update**	CVS
PIAS	**Demand responsive transit request**	TRMS
PIAS	**Traveler information request**	ISP
PIAS	**Emergency notification**	EM
RTS	**Emergency notification**	EM
TRMS	**Demand responsive transit request**	PIAS
TRMS	**Request for vehicle measures**	TRVS
TRMS	**Route assignment**	TRVS
TRVS	**Emergency notification**	TRMS
TRVS	**Vehicle probe data**	TRMS
TRVS	**Traveler information request**	TRMS
VS	**vehicle probe data**	ISP
VS	**emergency notification**	EM
VS	**Traveler information request**	ISP
VS	**map update request**	X23

U1 - Circuit Switched data (Messaging; larger data transactions, e.g., compressed image)		
Source	**Architecture Flow**	**Destination**
ISP	**Traveler information**	PIAS
ISP	**Traveler information**	VS
X23 MAP Update Provider	**Map updates**	PIAS
X23 MAP Update Provider	**Map updates**	VS

U1 - Circuit Switched Voice (live voice interaction; early implementations)		
Source	**Architecture Flow**	**Destination**
EM	**Assigned route**	EVS
EM	**Hazmat information**	EVS
EM	**Emergency dispatch requests**	EVS
EM	**Emergency acknowledge**	RTS
EVS	**Emergency vehicle driver status update**	EM

EVS	**Emergency vehicle driver input**	EM
RTS	**Emergency notification**	EM
VS	**Emergency notification**	EM

U1 - FM Subcarrier Broadcast Services (Broadcast of free services and services that require subscription; e.g., traveler information,)		
source	**Architecture Flow**	**destination**
ISP	**Broadcast information**	PIAS
ISP	**traveler information**	VS
ISP	**broadcast information**	VS
ISP	**broadcast information**	RTS

U1 - Multicast Services (Distribution services that require subscription; e.g., map updates)		
source	**Architecture Flow**	**destination**
ISP	**traveler information**	PIAS
ISP	**traveler information**	VS
ISP	**broadcast information**	VS
ISP	**broadcast information**	RTS
X23 MAP update provider	**map updates**	PIAS
X23 MAP update provider	**map updates**	VS

U2 - RF beacon for close-proximity wireless communication between vehicle and roadside		
Source	Architecture Flow	Destination
CVS	**screening data**	CVCS
CVS	**on board safety data**	CVCS
CVCS	**safety inspection record**	CVS
CVCS	**Pull in message**	CVS
EVS	**Emergency vehicle preemption request**	RS
PMS	**Tag update**	VS
RS	**AHS control data**	VS
RS	**vehicle signage data**	VS
RS	**intersection status**	VS
TCS	**Request tag data**	VS
TRVS	**signal priority request**	RS
VS	**Tag data**	TCS
VS	**AHS vehicle data**	RS
VS	**Tag data**	PMS

Appendix A

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Basic Vehicle	Vehicle	vehicle measures	W	Broadcast data	product	I	R
Commercial Vehicle	Commercial Vehicle Check	CVO weight and presence	P	Physical Interface	NA		
Commercial Vehicle	Commercial Vehicle Subsystem	vehicle measures	W	Broadcast data	product	I	R
Commercial Vehicle Administration	Commercial Vehicle Check	credentials information	W,U1t	Conversational data, Messaging data	regional	04	
Commercial Vehicle Administration	Commercial Vehicle Check	CVO database update	W	Conversational data, Messaging data	regional	04	
Commercial Vehicle Administration	Commercial Vehicle Check	international border crossing data	W	Messaging data	regional	04	
Commercial Vehicle Administration	Commercial Vehicle Check	safety information	W,U1t	Conversational data, Messaging data	regional	04	
Commercial Vehicle Administration	CVO Information Requestor	credentials and safety information response	W	Messaging data	national	04	
Commercial Vehicle Administration	DMV	license request	W	Messaging data	national	04	P
Commercial Vehicle Administration	Enforcement Agency	request for information on violators	W	Messaging data	national	04	P
Commercial Vehicle Administration	Enforcement Agency	violation notification	W	Messaging data	regional	04	P
Commercial Vehicle Administration	Financial Institution	payment request	W	Conversational data, Messaging data	national	04	F
Commercial Vehicle Administration	Fleet and Freight Management	activity reports	W	Messaging data	national	04	
Commercial Vehicle Administration	Fleet and Freight Management	compliance review report	W	Messaging data	national	04	
Commercial Vehicle Administration	Fleet and Freight Management	electronic credentials	W,U1t	Messaging data	national	04	
Commercial Vehicle Administration	Government Administrators	tax-credentials-fees request	W	Messaging data	national	04	
Commercial Vehicle Administration	Other CVAS	credentials and safety information request	W	Messaging data	national	04	
Commercial Vehicle Administration	Other CVAS	CVAS information exchange	W	Messaging data	national	04	
Commercial Vehicle Administration	Planning Subsystem	operational data	W	Messaging data	regional	04	
Commercial Vehicle Check	Commercial Vehicle Administration	citation data	W	Conversational data, Messaging data	regional	04	P
Commercial Vehicle Check	Commercial Vehicle Administration	credentials information request	W	Conversational data, Messaging data	regional	04	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Commercial Vehicle Check	Commercial Vehicle Administration	international border crossing data update	W	Conversational data, Messaging data	regional	04	
Commercial Vehicle Check	Commercial Vehicle Administration	roadside log update	W	Messaging data	regional	04	
Commercial Vehicle Check	Commercial Vehicle Administration	safety information request	W	Conversational data, Messaging data	regional	04	
Commercial Vehicle Check	Commercial Vehicle Driver	CVO Pull in Message	H	Human Interface	regional	H	
Commercial Vehicle Check	Commercial Vehicle Subsystem	border clearance event record	U2	Conversational Data	national	01	T
Commercial Vehicle Check	Commercial Vehicle Subsystem	clearance event record	U2	Conversational data	national	01	T
Commercial Vehicle Check	Commercial Vehicle Subsystem	electronic clearance request	U2	Conversational data	national	01	T
Commercial Vehicle Check	Commercial Vehicle Subsystem	lock tag data request	U2	Conversational Data	national	01	T
Commercial Vehicle Check	Commercial Vehicle Subsystem	on-board safety request	U2	Conversational data	national	01	T
Commercial Vehicle Check	Commercial Vehicle Subsystem	pass/pull-in	U2	Conversational data	national	01	T,R
Commercial Vehicle Check	Commercial Vehicle Subsystem	safety inspection record	U2	Conversational Data	national	01	T
Commercial Vehicle Check	Commercial Vehicle Subsystem	screening request	U2	Conversational data	national	01	T
Commercial Vehicle Check	CVO Inspector	CVO inspector information	H	Human Interface	product	H	
Commercial Vehicle Driver	Commercial Vehicle Subsystem	CVO driver initialization	H	Human Interface	product	H	
Commercial Vehicle Manager	Fleet and Freight Management	fleet manager inquiry	H	Human Interface	product	H	
Commercial Vehicle Subsystem	Commercial Vehicle	lock tag data request	W	Conversational Data	product		
Commercial Vehicle Subsystem	Commercial Vehicle Check	electronic clearance data	U2	Conversational data	national	01	T
Commercial Vehicle Subsystem	Commercial Vehicle Check	lock tag data	U2	Conversational Data	national	01	T
Commercial Vehicle Subsystem	Commercial Vehicle Check	on board safety data	U2	Conversational data	national	01	T
Commercial Vehicle Subsystem	Commercial Vehicle Check	screening data	U2	Conversational data	national	01	T,R
Commercial Vehicle Subsystem	Commercial Vehicle Driver	alerts, Messages	H	Human Interface	product	H	
Commercial Vehicle Subsystem	Commercial Vehicle Driver	CVO Pull in Message	H	Messaging data	product	H	
Commercial Vehicle Subsystem	Commercial Vehicle Driver	log information	H	Human Interface	product	H	
Commercial Vehicle Subsystem	Fleet and Freight Management	driver and vehicle information	U1t	Messaging data, location data	none	P	
Commercial Vehicle Subsystem	Fleet and Freight Management	on board vehicle data	U1t,U2	Messaging data	none	P	
Commercial Vehicle Subsystem	Vehicle	commercial vehicle data	W	Messaging data	product	I	
Construction and Maintenance	Traffic Management	equipment maintenance status	W	Messaging data	product		
Construction and Maintenance	Traffic Management	maintenance resource response	W	Conversational data, Messaging data	product		
Construction and Maintenance	Traffic Management	work zone status	W	Messaging data	product		
CVO Information Requestor	Commercial Vehicle Administration	credentials and safety information request	W	Messaging data	national	04	
CVO Inspector	Commercial Vehicle Check	CVC override mode	H	Conversational data, Messaging data	product	H	T
CVO Inspector	Commercial Vehicle Check	CVO inspector input	H	Human Interface	product	H	
DMV	Commercial Vehicle Administration	registration	W	Messaging data	national	04	P
DMV	Parking Management	registration	W	Messaging data	national		
DMV	Toll Administration	registration	W	Messaging data	national		P
DMV	Traffic Management	registration	W	Messaging data	national	06	P
Driver	Vehicle	driver inputs	H	Human Interface	product	H	
Driver	Vehicle	request for service	H	Human Interface	product	H	
Emergency Management	Emergency System Operator	emergency operations status	H	Human Interface	product	H	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Emergency Management	Emergency Telecommunications System	incident notification response	W	Conversational data conversational speec	regional	09	
Emergency Management	Emergency Vehicle Subsystem	emergency dispatch requests	U1t	Conversational speech, Messaging data	regional		E
Emergency Management	Emergency Vehicle Subsystem	incident command information	U1t	Conversational speech, Messaging data	regional		E
Emergency Management	Emergency Vehicle Subsystem	suggested route	U1t	Conversational speech, Messaging data	regional		E
Emergency Management	Fleet and Freight Management	Hazmat information request	W	Conversational data, Messaging data	national	09	E
Emergency Management	Information Service Provider	emergency vehicle route request	W	Conversational speech, Messaging data	regional	09	E
Emergency Management	Information Service Provider	incident information	W	Conversational speech, Messaging data	regional	09	
Emergency Management	Map Update Provider	map update request	W	Messaging data	national	02	
Emergency Management	Media	incident information for media	W	Messaging data	product		
Emergency Management	Other EM	incident report	W	Conversational data, Messaging data	regional	09	E
Emergency Management	Other EM	incident response coordination	W	Conversational data, Messaging data	regional	09	E
Emergency Management	Personal Information Access	emergency acknowledge	W,U1t	Conversational data, Messaging data	national	05	
Emergency Management	Planning Subsystem	operational data	W	Conversational data, Messaging data	regional	09	
Emergency Management	Remote Traveler Support	emergency acknowledge	W,U1t	Conversational speech, Messaging data	national	05	
Emergency Management	Traffic Management	emergency traffic control request	W	Conversational data, conversational speec	regional	08,09	E
Emergency Management	Traffic Management	incident information	W	Conversational data, Messaging data	regional	09	
Emergency Management	Traffic Management	incident response status	W	Conversational data, Messaging data	regional	09	
Emergency Management	Traffic Management	remote surveillance control	W	Conversational data, Messaging data	regional	09	
Emergency Management	Traffic Management	resource request	W	Conversational data, Messaging data	regional	09	
Emergency Management	Transit Management	transit emergency coordination data	W	Conversational data, Messaging data	regional	09,05	E
Emergency Management	Vehicle	emergency acknowledge	U1t	Conversational data conversational speec	national	05	
Emergency Personnel	Emergency Vehicle Subsystem	emergency personnel inputs	H	Human Interface	product	H	E
Emergency System Operator	Emergency Management	emergency operations request	H	Human Interface	product	H	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Emergency Telecommunications System	Emergency Management	incident notification	W	Conversational data, Conversational speech, Location data	regional	09	
Emergency Vehicle Subsystem	Emergency Management	emergency dispatch response	U1t	Conversational speech, Messaging data	regional		E
Emergency Vehicle Subsystem	Emergency Management	emergency vehicle tracking data	U1t	Conversational speech, Messaging data	regional		E
Emergency Vehicle Subsystem	Emergency Management	incident command request	U1t	Conversational speech, Messaging data	regional		E
Emergency Vehicle Subsystem	Emergency Management	incident status	U1t	Conversational speech, Messaging data	regional		E
Emergency Vehicle Subsystem	Emergency Personnel	dispatch information	H	Human Interface	product	H	E
Emergency Vehicle Subsystem	Emergency Personnel	incident command information presentation	H	Human Interface	product	H	E
Emergency Vehicle Subsystem	Roadway Subsystem	local signal preemption request	U2	Conversational data	regional	08,01	T,E
Emissions Management	Map Update Provider	map update request	W	Messaging data	national	02	
Emissions Management	Planning Subsystem	operational data	W	Messaging data	regional		
Emissions Management	Roadway Subsystem	vehicle pollution criteria	W	Messaging data	product	07	
Emissions Management	Traffic Management	widearea statistical pollution information	W	Messaging data	product	07	
Emissions Management	Traffic Operations Personnel	pollution data display	H	Human Interface	product	H	
Enforcement Agency	Commercial Vehicle Administration	information on violators	W	Messaging	regional	04	P
Environment	Emissions Management	pollution data	P	Physical Interface	NA		
Environment	Roadway Subsystem	pollution data	P	Physical Interface	NA		
Event Promoters	Traffic Management	event plans	W	Messaging data, Multicast data	regional	06	
Financial Institution	Commercial Vehicle Administration	transaction status	W	Conversational data, Messaging data	national	04	F
Financial Institution	Information Service Provider	transaction status	W	Conversational data, Messaging data	national	E	F
Financial Institution	Parking Management	transaction status	W	Conversational data, Messaging data	national	E	F
Financial Institution	Toll Administration	transaction status	W	Conversational data, Messaging data	national	E	F
Financial Institution	Transit Management	transaction status	W	Conversational data, Messaging data	national	E	F
Fleet and Freight Management	Commercial Vehicle Administration	credential application	W	Conversational data, Messaging data	national	04	
Fleet and Freight Management	Commercial Vehicle Administration	information request	W	Conversational data, Messaging data	national	04	
Fleet and Freight Management	Commercial Vehicle Administration	tax filing, audit data	w	Messaging data	national	04	P
Fleet and Freight Management	Commercial Vehicle Manager	fleet status	H	Human Interface	product	H	
Fleet and Freight Management	Commercial Vehicle Subsystem	fleet to driver update	U1t	Messaging data	none	P	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Fleet and Freight Management	Emergency Management	Hazmat information	W	Conversational data, Messaging data	national	09,05	
Fleet and Freight Management	Information Service Provider	route request	W	Conversational data, Messaging data	none	10	P
Fleet and Freight Management	Intermodal Freight Depot	intermod CVO coord	W	Messaging data	national	04	
Fleet and Freight Management	Intermodal Freight Shipper	intermod CVO coord	W	Messaging data	regional	04	
Fleet and Freight Management	Payment Instrument	request for payment	S	Conversational Data	national		
Government Administrators	Commercial Vehicle Administration	regulations	W	Messaging data, Multicast data	national	04	
Information Service Provider	Emergency Management	emergency vehicle route	W	Conversational speech, Messaging data	regional	09	
Information Service Provider	Emergency Management	incident information request	W	Conversational speech, Messaging data	regional	09	
Information Service Provider	Financial Institution	payment request	W	Conversational data, Messaging data	national	E	
Information Service Provider	Fleet and Freight Management	route plan	W	Messaging data	none	10	P
Information Service Provider	Intermodal Transportation Service Provider	intermodal information	W	Messaging data	regional	10	
Information Service Provider	ISP Operator	ISP operating parameters	H	Human Interface	product	H	
Information Service Provider	Map Update Provider	map update request	W	Messaging data	national	02	
Information Service Provider	Media	traveler information for media	W	Messaging data	product	10	
Information Service Provider	Other ISP	ISP coordination	W	Messaging data	national	10	
Information Service Provider	Parking Management	parking lot data request	W	Messaging data	regional	10	P
Information Service Provider	Parking Management	parking reservations request	W	Messaging data	regional	10	P
Information Service Provider	Personal Information Access	broadcast information	W,U1b	Messaging data,Broadcast data, Multicast	national	03	
Information Service Provider	Personal Information Access	traveler information	W,U1t	Broadcast data,Multicast data	national	03	P
Information Service Provider	Personal Information Access	trip plan	W,U1t	Conversational data, Messaging data	national	03	P
Information Service Provider	Planning Subsystem	aggregate travel data	W	Messaging data	regional	10	
Information Service Provider	Remote Traveler Support	broadcast information	W,U1b	Messaging data,Broadcast data, Multicast	product	10	
Information Service Provider	Remote Traveler Support	traveler information	W,U1t	Broadcast data,Multicast data	product	10	P
Information Service Provider	Remote Traveler Support	trip plan	W	Conversational Data	product	10	P
Information Service Provider	Toll Administration	toll data request	W	Messaging data	regional	10	
Information Service Provider	Traffic Management	fare and price information	W	Messaging data	regional	06	
Information Service Provider	Traffic Management	logged special vehicle route	W	Conversational data, Messaging data	regional	06	P
Information Service Provider	Traffic Management	request for traffic information	W	Messaging data	regional	06	
Information Service Provider	Traffic Management	road network use	W	Messaging data	regional	06	
Information Service Provider	Transit Management	demand responsive transit request	W	Messaging data	regional	10	P

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Information Service Provider	Transit Management	selected routes	W	Conversational data, Messaging data	regional	10	P
Information Service Provider	Transit Management	transit information request	W	Messaging data	regional	10	
Information Service Provider	Vehicle	broadcast information	U1b	Messaging data,Broadcast data, Multicast	national	03	
Information Service Provider	Vehicle	traveler information	U1t,U1b	Messaging data,Broadcast data, Multicast	national	03	P
Information Service Provider	Vehicle	trip plan	U1t	Conversational data, Messaging data	national	03	P
Information Service Provider	Yellow Pages Service Providers	provider profile confirm	W	Messaging data	national	10	P
Information Service Provider	Yellow Pages Service Providers	travel service request	W	Messaging data	national	10	P
Intermodal Freight Depot	Fleet and Freight Management	intermod CVO coord	W	Messaging data	national	04	
Intermodal Freight Shipper	Fleet and Freight Management	intermod CVO coord	W	Messaging data	regional	04	
Intermodal Transportation Service Provider	Information Service Provider	intermodal information	W	Messaging data	regional	10	
Intermodal Transportation Service Provider	Transit Management	intermodal information	W	Messaging data	regional		
ISP Operator	Information Service Provider	ISP operating parameter updates	H	Human Interface	product	H	
Location Data Source	Personal Information Access	position fix	L	Broadcast Data	product	02	
Location Data Source	Vehicle	position fix	L	Broadcast Data	product	02	
Map Update Provider	Emergency Management	map updates	W	Messaging data, Multicast data	national	02	
Map Update Provider	Emissions Management	map updates	W	Messaging data	national	02	
Map Update Provider	Information Service Provider	map updates	W	Messaging data, Multicast data	national	02	
Map Update Provider	Personal Information Access	map updates	W,U1t	Messaging data, Multicast data	national	02	
Map Update Provider	Planning Subsystem	map updates	W	Messaging data,Broadcast data, Multicast	national	02	
Map Update Provider	Remote Traveler Support	map updates	W	Messaging data	national	02	
Map Update Provider	Traffic Management	map updates	W	Messaging data, Multicast data	national	02	
Map Update Provider	Transit Management	map updates	W	Messaging data, Multicast data	national	02	
Map Update Provider	Vehicle	map updates	U1t	Messaging data, Multicast data	national	02	
Media	Emergency Management	media information request	W	Conversational Data, Messaging data	product		
Media	Information Service Provider	external reports	W	Messaging data, Multicast data	product	10	
Media	Information Service Provider	media information request	W	Conversational data, Messaging data	product	10	
Media	Traffic Management	external reports	W	Messaging data, Multicast data	product	06	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Media	Traffic Management	media information request	W	Conversational Data, Messaging data	product	06	
Media	Transit Management	media information request	W	Conversational Data, Messaging data	product		
Multimodal Crossings	Roadway Subsystem	multimodal crossing status	W	Conversational data, Messaging data	national		R
Other CVAS	Commercial Vehicle Administration	credentials and safety information response	W	Messaging data	national	04	
Other CVAS	Commercial Vehicle Administration	CVAS information exchange	W	Messaging data	national	04	
Other EM	Emergency Management	incident report	W	Conversational data, Messaging data	regional	09	E
Other EM	Emergency Management	incident response coordination	W	Conversational data, Messaging data	regional	09	E
Other ISP	Information Service Provider	ISP coordination	W	Messaging data	national	10	
Other TM	Traffic Management	traffic control coordination	W	Messaging data	regional	06	
Other TM	Traffic Management	traffic information coordination	W	Messaging data	regional	06	
Other TRM	Transit Management	TRMS coord	W	Messaging data	regional		
Other Vehicle	Vehicle	vehicle to vehicle coordination	U3	Conversational data	national	A	T,R
Parking Management	DMV	license request	W	Messaging data	national		
Parking Management	Driver	transaction status	H	Human Interface	product	H	
Parking Management	Enforcement Agency	violation notification	W	Messaging data	regional		
Parking Management	Financial Institution	payment request	W	Conversational data, Messaging data	national	E	
Parking Management	Information Service Provider	parking availability	W	Messaging data	regional	10	
Parking Management	Information Service Provider	parking lot reservation confirmation	W	Messaging data	regional	10	
Parking Management	Parking Operator	parking status	W	Messaging data	product		
Parking Management	Parking Service Provider	parking availability	W	Messaging data	product		
Parking Management	Planning Subsystem	operational data	W	Messaging data	regional		
Parking Management	Traffic Management	demand management response	W	Messaging data	regional	06	
Parking Management	Traffic Management	parking availability	W	Messaging data	regional	06	
Parking Management	Transit Management	transit parking coordination	W	Messaging data	regional		
Parking Management	Vehicle	request tag data	U2	Conversational data	national	01	T,R
Parking Management	Vehicle	tag update	U2	Conversational data	national	01	T,R
Parking Operator	Parking Management	parking instructions	H	Human Interface	product	H	
Parking Service Provider	Parking Management	request for performance data	W	Messaging data	product		
Payment Instrument	Fleet and Freight Management	payment	S	Conversational Data	national		
Payment Instrument	Personal Information Access	payment	S	Conversational Data	national		
Payment Instrument	Remote Traveler Support	Payment	S	Conversational Data	national		F
Payment Instrument	Transit Vehicle Subsystem	payment	S	Conversational Data	national		F
Payment Instrument	Vehicle	payment	S	Conversational Data	national		F
Pedestrians	Roadway Subsystem	crossing call	H	Human Interface	national	H	
Personal Information Access	Emergency Management	emergency notification	U1t	Conversational data, Messaging data	national	05	E
Personal Information Access	Information Service Provider	traveler profile	W,U1t	Conversational data, Messaging data	national	03	P
Personal Information Access	Information Service Provider	traveler request	W,U1t	Conversational data, Messaging data	national	03	P

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Personal Information Access	Information Service Provider	trip confirmation	W,U1t	Conversational Data, Messaging data	national	03	P
Personal Information Access	Information Service Provider	trip request	W,U1t	Conversational Data, Messaging data	national	03	P
Personal Information Access	Information Service Provider	yellow pages request	W,U1t	Conversational Data, Messaging data	national	03	P
Personal Information Access	Map Update Provider	map update request	W,U1t	Messaging data	national	02	
Personal Information Access	Payment Instrument	request for payment	S	Conversational Data	national		
Personal Information Access	Transit Management	transit information user request	W,U1t	Messaging data	national		
Personal Information Access	Traveler	traveler interface updates	H	Human Interface	product	H	P
Planning Subsystem	Map Update Provider	map update request	W	Messaging data	national	02	
Planning Subsystem	Traffic Management	planning data	W	Messaging data	regional	06	
Planning Subsystem	Transportation Planners	planning data	W	Messaging data	regional		
Potential Obstacles	Vehicle	physical presence	P	Physical Interface	NA		
Rail Operations	Traffic Management	railroad advisories	W	Conversational data, Messaging data	national	12	
Rail Operations	Traffic Management	railroad schedules	W	Conversational data, Messaging data	national	12	
Remote Traveler Support	Emergency Management	emergency notification	W,U1t	Conversational speech, Messaging data, location data	national	05	E
Remote Traveler Support	Information Service Provider	traveler request	W	Messaging data	product	10	P
Remote Traveler Support	Information Service Provider	trip confirmation	W	Messaging data	product	10	P
Remote Traveler Support	Information Service Provider	trip request	W	Conversational Data	product	10	P
Remote Traveler Support	Information Service Provider	yellow pages request	W	Conversational data, Messaging Data	product	10	P
Remote Traveler Support	Map Update Provider	map update request	W	Messaging data	national	02	
Remote Traveler Support	Payment Instrument	request for payment	S	Conversational Data	national		F
Remote Traveler Support	Transit Management	emergency notification	W	Conversational data, Messaging Data	product	05,11	E
Remote Traveler Support	Transit Management	secure area surveillance data	W	Conversational data, Messaging Data	product	11	E,T
Remote Traveler Support	Transit Management	transit fare payment requests	W	Conversational data, Messaging data	product	11	F
Remote Traveler Support	Transit Management	transit information user request	W	Messaging data	product	11	P
Remote Traveler Support	Transit User	transit user fare status	H	Human Interface	product	H	
Remote Traveler Support	Transit User	transit user outputs	H	Human Interface	product	H	
Remote Traveler Support	Traveler	traveler interface updates	H	Human Interface	product	H	P
Roadway	Vehicle	roadway characteristics	P	Physical Interface	NA		
Roadway Environment	Roadway Subsystem	weather conditions	P	Physical Interface	NA		
Roadway Environment	Vehicle	weather conditions	P	Physical Interface	NA		
Roadway Subsystem	Basic Vehicle	broadcast advisories	U1b	Conversational speech	national		
Roadway Subsystem	Driver	driver information	H	Human Interface	national	12	T, R
Roadway Subsystem	Emissions Management	pollution data	W	Messaging data	product	07	
Roadway Subsystem	Multimodal Crossings	highway control status	W	Conversational data, Messaging data	national		R
Roadway Subsystem	Pedestrians	crossing permission	H	Human Interface	national	H	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Roadway Subsystem	Traffic Management	AHS status	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	emissions data	W	Messaging data	product		
Roadway Subsystem	Traffic Management	environmental conditions	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	fault reports	W	Conversational data, Messaging data	product	07	
Roadway Subsystem	Traffic Management	freeway control status	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	hov data	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	hri status	W	Conversational data, Messaging data	product	12	R, S
Roadway Subsystem	Traffic Management	incident data	W	Messaging data	product	07	T, R
Roadway Subsystem	Traffic Management	intersection blockage notification	W	Messaging data	product	12	T, R
Roadway Subsystem	Traffic Management	request for right-of-way	W	Conversational data, Messaging data	product	08,07	R
Roadway Subsystem	Traffic Management	reversible lane status	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	roadway information system status	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	signal control status	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	traffic flow	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	traffic images	W	Messaging data	product	07	T
Roadway Subsystem	Traffic Management	vehicle probe data	W	Messaging data	product	07	
Roadway Subsystem	Vehicle	AHS control data	U2	Messaging data	national	01	T, R
Roadway Subsystem	Vehicle	intersection status	U2	Messaging data	national	01	T, R
Roadway Subsystem	Vehicle	request tag data	U2	Messaging data	national	01	T
Roadway Subsystem	Vehicle	vehicle signage data	U2	Messaging data	national	01	T
Roadway Subsystem	Wayside Equipment	hri status	W	Conversational data, Messaging data	product	12	R, S
Roadway Subsystem	Wayside Equipment	intersection blockage notification	W	Messaging data	product	12	T, R
Secure Area Environment	Remote Traveler Support	secure area characteristics	P	Physical Interface	NA		
Toll Administration	DMV	license request	W	Messaging data	national		
Toll Administration	Enforcement Agency	violation notification	W	Messaging data	regional		
Toll Administration	Financial Institution	payment request	W	Messaging data	national	E	
Toll Administration	Information Service Provider	probe data	W	Messaging data	regional	06	
Toll Administration	Information Service Provider	toll data	W	Messaging data	regional	10	
Toll Administration	Planning Subsystem	operational data	W	Messaging data	regional		
Toll Administration	Toll Collection	toll instructions	W	Messaging data	regional		
Toll Administration	Toll Operator	toll transaction reports	H	Human Interface	product	H	
Toll Administration	Toll Service Provider	toll revenues and summary reports	W	Messaging data	product		
Toll Administration	Traffic Management	demand management response	W	Messaging data	regional	06	
Toll Administration	Traffic Management	probe data	W	Messaging data	regional	06	
Toll Collection	Driver	transaction status	H	Human Interface	national	H	
Toll Collection	Toll Administration	toll transactions	W	Messaging data	regional		
Toll Collection	Vehicle	request tag data	U2	Conversational data	national	01	T,R
Toll Collection	Vehicle	tag update	U2	Conversational Data	national	01	T,R
Toll Operator	Toll Administration	toll operator requests	H	Human Interface	product	H	
Toll Service Provider	Toll Administration	toll fees	H	Human Interface	product	H	
Traffic	Roadway Subsystem	traffic characteristics	P	Physical Interface	NA		
Traffic Management	Construction and Maintenance	closure coordination	W	Conversational data, Messaging data	product	06	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Traffic Management	Construction and Maintenance	maintenance resource request	W	Conversational data, Messaging data	product	06	
Traffic Management	Construction and Maintenance	traffic equipment status	W	Conversational data, Messaging data	product	06	
Traffic Management	DMV	license request	W	Messaging data	national	06	
Traffic Management	Emergency Management	current network conditions	W	Messaging data	regional		
Traffic Management	Emergency Management	emergency traffic control response	W	Conversational data, conversational speech	regional		
Traffic Management	Emergency Management	incident information request	W	Messaging data	regional	09	E
Traffic Management	Emergency Management	incident notification	W	Messaging data	regional	09	E
Traffic Management	Emergency Management	resource deployment status	W	Messaging data	regional	09	
Traffic Management	Emissions Management	pollution state data request	W	Messaging data	product	07	
Traffic Management	Enforcement Agency	violation notification	W	Messaging data	regional	06	
Traffic Management	Event Promoters	event confirmation	W	Messaging data	regional	06	
Traffic Management	Information Service Provider	request fare and price information	W	Messaging data	regional	06	
Traffic Management	Information Service Provider	traffic information	W	Messaging data	regional	06	
Traffic Management	Map Update Provider	map update request	W	Messaging data	national	02	
Traffic Management	Media	traffic information for media	W	Messaging data	product	06	
Traffic Management	Other TM	traffic control coordination	W	Messaging data	regional	06	
Traffic Management	Other TM	traffic information coordination	W	Messaging data	regional	06	
Traffic Management	Parking Management	demand management request	W	Messaging data	regional	06	
Traffic Management	Parking Management	parking instructions	W	Messaging data	regional	06	
Traffic Management	Planning Subsystem	operational data	W	Messaging data	regional	06	
Traffic Management	Rail Operations	hri advisories	W	Conversational Data, Messaging data	national	12	T,R
Traffic Management	Roadway Subsystem	AHS control information	W	Messaging data	product	07	
Traffic Management	Roadway Subsystem	freeway control data	W	Messaging data	product	07	T,S
Traffic Management	Roadway Subsystem	hri control data	W	Messaging data	product	12	R, S
Traffic Management	Roadway Subsystem	hri request	W	Conversational data, Messaging data	product	12	R, S
Traffic Management	Roadway Subsystem	roadway information system data	W	Messaging data	product	07	S
Traffic Management	Roadway Subsystem	sensor and surveillance control	W	Messaging data	product	07	T,S
Traffic Management	Roadway Subsystem	signal control data	W	Messaging data	product	07	T,S
Traffic Management	Toll Administration	demand management request	W	Messaging data	regional	06	
Traffic Management	Traffic Operations Personnel	traffic operations data	H	Human Interface	product	H	
Traffic Management	Transit Management	demand management request	W	Messaging data	regional	06	
Traffic Management	Transit Management	request transit information	W	Conversational data, Messaging data	regional	06	
Traffic Management	Transit Management	traffic control priority status	W	Conversational data, Messaging data	regional	06	
Traffic Management	Transit Management	traffic information for transit	W	Messaging data	regional	06	
Traffic Operations Personnel	Emissions Management	pollution data parameters	H	Human Interface	product	H	
Traffic Operations Personnel	Traffic Management	traffic operations requests	H	Human Interface	product	H	
Transit Driver	Transit Management	transit driver availability	H	Human Interface	product	H	
Transit Driver	Transit Vehicle Subsystem	transit driver inputs	H	Human Interface	product	H	
Transit Fleet Manager	Transit Management	transit fleet manager inputs	H	Human Interface	product	H	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Transit Maintenance Personnel	Transit Management	maintenance status	H	Human Interface	product	H	
Transit Management	Emergency Management	transit emergency data	W	Conversational data, Messaging data	regional	05,09	E
Transit Management	Enforcement Agency	violation notification	W	Messaging data	regional		
Transit Management	Financial Institution	payment request	W	Conversational data, Messaging data	national	E	
Transit Management	Information Service Provider	demand responsive transit plan	W	Conversational data, Messaging data	regional	10	P
Transit Management	Information Service Provider	transit and fare schedules	W	Messaging data	regional	10	
Transit Management	Information Service Provider	transit incident information	W	Messaging data	regional	10	
Transit Management	Information Service Provider	transit request confirmation	W	Messaging data	regional	10	P
Transit Management	Intermodal Transportation Service Provider	intermodal information	W	Messaging data	regional		
Transit Management	Map Update Provider	map update request	W	Messaging data	national	02	
Transit Management	Media	transit incidents for media	W	Messaging data	product		
Transit Management	Media	transit information for media	W	Messaging data	product		
Transit Management	Other TRM	TRMS coord	W	Messaging data	regional		
Transit Management	Parking Management	transit parking lot response	W	Messaging data,Broadcast data, Multicast	regional		
Transit Management	Personal Information Access	personal transit information	W,U1t	Messaging data	national		
Transit Management	Planning Subsystem	operational data	W	Messaging data	regional		
Transit Management	Remote Traveler Support	emergency acknowledge	W	Conversational data, Messaging data	product	05,11	
Transit Management	Remote Traveler Support	secure area monitoring support	W	Conversational data, Messaging data	product	11	E,T
Transit Management	Remote Traveler Support	transit fare payment responses	W	Conversational data, Messaging data	product	11	F
Transit Management	Remote Traveler Support	transit traveler information	W	Messaging data, Multicast data	product	11	
Transit Management	Traffic Management	demand management response	W	Messaging data	regional	06	
Transit Management	Traffic Management	traffic control priority request	W	Messaging data	regional	08,06	
Transit Management	Traffic Management	transit system data	W	Messaging data	regional	06	
Transit Management	Transit Driver	route assignment	H	Human Interface	product	H	
Transit Management	Transit Fleet Manager	transit operations planning data	H	Human Interface	product	H	
Transit Management	Transit Maintenance Personnel	transit work schedule	H	Human Interface	product	H	
Transit Management	Transit System Operators	transit operator display	H	Human Interface	product	H	
Transit Management	Transit Vehicle Subsystem	bad tag list	U1t	Messaging data	product	11	
Transit Management	Transit Vehicle Subsystem	driver instructions	U1t	Messaging data	product	11	
Transit Management	Transit Vehicle Subsystem	emergency acknowledge	U1t	Conversational data, Messaging data	product	05,11	
Transit Management	Transit Vehicle Subsystem	fare management information	U1t	Messaging data	product	11	
Transit Management	Transit Vehicle Subsystem	request for vehicle measures	U1t,U2	Messaging data	product	11	
Transit Management	Transit Vehicle Subsystem	transit schedule information	U1t	Messaging data	product	11	
Transit Management	Transit Vehicle Subsystem	traveler information	U1t	Messaging data	product	11	P
Transit System Operators	Transit Management	transit operator management data	H	Human Interface	product	H	
Transit User	Remote Traveler Support	transit user inputs	H	Human Interface	product	H	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Transit User	Transit Vehicle Subsystem	emergency notification	H	Human Interface	product	H	
Transit User	Transit Vehicle Subsystem	transit user inputs	H	Human Interface	product	H	
Transit Vehicle	Transit Vehicle Subsystem	vehicle measures	W	Broadcast data	product	I	R
Transit Vehicle Subsystem	Payment Instrument	request for payment	S	Conversational Data	national		F
Transit Vehicle Subsystem	Roadway Subsystem	local signal priority request	U2	Conversational data	regional	01,08	T
Transit Vehicle Subsystem	Transit Driver	transit driver display	H	Human Interface	product	H	
Transit Vehicle Subsystem	Transit Management	emergency notification	U1t	Messaging data	product	11,05	E
Transit Vehicle Subsystem	Transit Management	fare and payment status	U1t,U2	Conversational data, Messaging data	product	11	F,T
Transit Vehicle Subsystem	Transit Management	request for bad tag list	U1t,U2	Conversational data, Messaging data	product	11	F,T
Transit Vehicle Subsystem	Transit Management	transit vehicle conditions	U1t,U2	Messaging data	product	11	
Transit Vehicle Subsystem	Transit Management	transit vehicle location data	U1t,U2	Conversational data, Messaging data, location data	product	11	
Transit Vehicle Subsystem	Transit Management	transit vehicle passenger and use data	U1t,U2	Conversational data, Messaging data	product	11	
Transit Vehicle Subsystem	Transit Management	transit vehicle schedule performance	U1t,U2	Conversational data, Messaging data	product	11	
Transit Vehicle Subsystem	Transit Management	traveler request	U1t	Conversational data, Messaging data	product	11	P
Transit Vehicle Subsystem	Transit User	transit user fare status	H	Human Interface	product	H	
Transit Vehicle Subsystem	Transit User	transit user outputs	H	Human Interface	product	H	
Transit Vehicle Subsystem	Vehicle	traveler advisory request	W	Messaging data	product		
Transportation Planners	Planning Subsystem	planning data	W	Messaging data	regional		
Traveler	Personal Information Access	traveler request	H	Human Interface	product	H	
Traveler	Remote Traveler Support	traveler request	H	Human Interface	product	H	
Vehicle	Basic Vehicle	vehicle control	W	Conversational Data	product	I	R
Vehicle	Commercial Vehicle Subsystem	commercial vehicle data request	W	Messaging data	product	I	
Vehicle	Driver	driver updates	H	Human Interface	product	H	
Vehicle	Driver	transaction status	H	Human Interface	product	H	
Vehicle	Emergency Management	emergency notification	U1t	Conversational speech, Messaging data, location data	national	05	E
Vehicle	Emergency Vehicle Subsystem	vehicle location	W	Broadcast data	product		
Vehicle	Information Service Provider	traveler profile	U1t	Conversational data, Messaging data	national	03	P
Vehicle	Information Service Provider	traveler request	U1t	Conversational data, Messaging data	national	03	P
Vehicle	Information Service Provider	trip confirmation	U1t	Conversational data, Messaging data	national	03	P
Vehicle	Information Service Provider	trip request	U1t	Conversational data, Messaging data	national	03	P
Vehicle	Information Service Provider	vehicle probe data	U1t	Messaging data, location data	national	03	P
Vehicle	Information Service Provider	yellow pages request	U1t	Conversational data, Messaging data	national	03	P
Vehicle	Map Update Provider	map update request	U1t	Messaging data	national	02	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Vehicle	Other Vehicle	vehicle to vehicle coordination	U3	Conversational data	national	A	T,R
Vehicle	Parking Management	tag data	U2	Conversational data	national	01	T,P
Vehicle	Payment Instrument	request for payment	S	Conversational Data	national		F
Vehicle	Roadway Subsystem	AHS vehicle data	U2	Conversational data	national	01	T,R
Vehicle	Roadway Subsystem	vehicle probe data	U2	Messaging Data	national	01	T,P
Vehicle	Toll Collection	tag data	U2	Conversational data	national	01	T,P
Vehicle	Transit Vehicle Subsystem	vehicle location	W	Broadcast data	product	I	
Vehicle Characteristics	Parking Management	vehicle characteristics	P	Physical Interface	NA		
Vehicle Characteristics	Roadway Subsystem	vehicle characteristics	P	Physical Interface	NA		
Vehicle Characteristics	Toll Collection	vehicle characteristics	P	Physical Interface	NA		
Wayside Equipment	Roadway Subsystem	arriving train information	W	Messaging data	product	12	T, R
Wayside Equipment	Roadway Subsystem	track status	W	Messaging data	product	12	T,R
Weather Service	Emergency Management	weather information	W	Messaging data,Broadcast data, Multicast	national		
Weather Service	Information Service Provider	weather information	W	Messaging data,Broadcast data, Multicast	national	10	
Weather Service	Traffic Management	weather information	W	Messaging data,Broadcast data, Multicast	national	06	
Weather Service	Transit Management	weather information	W	Messaging data,Broadcast data, Multicast	national		
Yellow Pages Service Providers	Information Service Provider	provider profile data	W	Messaging data	national	10	
Yellow Pages Service Providers	Information Service Provider	travel service info	W	Messaging data	national	10	

*** Interconnect Types as defined in Table A.2 (Communications Document Chapter 3)**
Communication Services Types in Table A.3 (Communications Document Chapter 3)
Interoperability Types in Table A.4 (Standards Development Plan Chapter 1)
Standards Packages in Table A.5 (Standards Requirements Document)
Special Constraints in Table A.6 (Physical Architecture Chapter 2.23)

Table A.2 Interconnect Types

Interconnect	Interconnect Name	Interconnect Description
H	Human Interface	Can be either a user interface to the system, an operator interface, or a driver.
L	Position Location	Interface between position location equipment and the source for indicating position location. This could be either information from a terrestrial source, GPS, FM subcarrier, Dead Reckoning etc
P	Physical Interface	This is an interface which senses some physical characteristic or causes some action that is not represented using standard communications technology (e.g. observing an obstacle)
S	Payment Instrument	This interface is between the card which is carried by the traveler and which contains the account number of stored value and the an object which accepts this information. The non-card interface could be a reader at a kiosk or in a vehicle. In the latter case, the reader in the vehicle forwards the information to the infrastructure.
U1t	Wide Area Wireless	Wide area 2-way communication capable of communication between a mobile traveler or vehicle and the infrastructure from any location.
U1b	Wide Area Broadcast	Wide area broadcast information in which the mobile traveler or vehicle can receive information from any location.
U2	Vehicle-to-Roadside	Short range vehicle to roadside (e.g. beacon). The interface contains information regarding which mobile entity is communicating.
U3	Vehicle-to-Vehicle	Primarily AHS type communications yet to be defined
W	Wireline	Wireline system interconnect which includes fixed to fixed communication capabilities. May include wide area wireless capabilities for transportable devices such as CMS, and may include short hop wireless connections to wireline subsystems from distributed assets such as signal and sensors. Includes normal telephone and public and private fiber-optic links.

Table A.3 Communications Services Types

Service Type	Description
Conversational	Implies the use of a two-way connection established before information exchange begins and terminated when exchange is completed
Messaging	Works more like electronic mail being exchanged between users without establishing a dedicated path between two sites.
Broadcast	Messages sent to all users
Multi-cast	Messages sent to only a subset of users

Table A.4 Interoperability Types

Interoperability	Description
National	Interfaces to the mobile subsystems (Vehicle Subsystems, Personal Information Access Subsystems) in the architecture support national interoperability since the same mobile subsystem should be able to roam the nation and use the local infrastructure to support ITS services. National interoperability is specified for all interfaces to mobile subsystems except where both the mobile subsystem and interfacing infrastructure are owned and operated by the same user. Examples of these include the Information Service Provider to Personal Information Access Subsystem, Toll Collection Subsystem to Vehicle Subsystem, and the Commercial Vehicle Subsystem to Commercial Vehicle Check Subsystem.
Regional	Interfaces connecting subsystems that may be operated by different agencies (interfaces that can span jurisdictional and/or regional boundaries) can be standardized to facilitate the sharing of information between agencies. National standards mitigate issues that may arise as boundaries change and new requirements for information sharing develop over time. Regional interoperability is specified where the underlying coordination issues are regional, rather than national, in scope. For instance, there is no real requirement for a Traffic Management Subsystem in California to be able to communicate and coordinate with a Traffic Management Subsystem in New York. Two different regional dialects for Traffic Management Subsystem communications could be implemented in the two geographically isolated subsystems, without significant impact to national interoperability goals. Examples of these include the Traffic Management Subsystem to Transit Management Subsystem, Traffic Management Subsystem to Information Service Provider, and Traffic Management Subsystem to Traffic Management Subsystem.
Product	Interfaces between subsystems that are operated and maintained by a single stakeholder (e.g. company or agency) do not require standardization to achieve national interoperability. The data formats and communications mechanisms that are used for these interfaces are largely transparent to the remainder of the architecture. In some cases, national standards are still very beneficial (and hence still attainable through the consensus standard process) since they may consolidate a market to achieve economy of scale efficiencies (e.g. Traffic Management Subsystem to Roadway Subsystem). Such standards may also support an optional level of interoperability by enabling various cooperative control options to be implemented based on regional preference.
None	In other cases, the sheer range of application-specific interfaces precludes efficient national standardization and no standard is suggested. For instance, a national standard is not recommended for the interface between the Fleet Management and Commercial Vehicle subsystems since the nature of the interface is so dependent on fleet type. From the National Architecture perspective, standardization for these interfaces is not suggested. Examples include the Fleet Management Subsystem to Commercial Vehicle Subsystem.

Table A.5 Standards Packages

Number	Requirement Package Name
01	Dedicate Short Range Communications (DSRC)
02	Digital Map Data Exchange and Location Referencing
03	Information Service Provider Wireless Interfaces
04	Inter-Center Data Exchange for Commercial Vehicle Operations
05	Personal and HAZMAT Maydays
06	Traffic Management Subsystems to Other Centers (Except EM)
07	Traffic Management Subsystems to Roadway Devices and Emissions Sensing/Management
08	Signal Priority for Transit and Emergency Vehicles
09	Emergency Management to Other Centers
10	Information Service Provider to Other Centers (except EM and TMS)
11	Transit Management to Transit Vehicle
12	Highway Rail Intersection
A	AHS Standards
E	Existing Standards
I	Internal and probably proprietary
P	Proprietary Standards
H	Human Interfaces

Table A.6 Special Constraints

Constraint Abbreviation	Constraint Name	Description
R	Reliability	Failure of the communication medium may result in severe accident. This communication channel may require redundant paths or extra attention paid to potential failure modes. For wireline cases, this may indicate alternate phone or other connections are required. For wireless cases (for AHS applications), special attention will be paid to the transmitters, receivers, and potential interference for these connections
F	Financial Security	Data contains financial information and must be protected accordingly. This data is specifically called out between the user's card and the infrastructure and between the infrastructure and financial institutions. Protections currently exist for the latter. Financial data transmitted over the air must be recognized as private data with an additional reliability requirement. Financial data may exist between other subsystems as part of normal messaging. It is assumed that such data will be treated with the same constraints as the interfaces specifically identified
P	Personal Privacy	Data contains personal information. Traveler requests and traveler location are private and should be protected. Subsystems aggregate these data and forward specific data with the traveler's permission.
E	Emergency Priority	Communication channel requires priority in emergencies. These data channels require that they be operational even when there is an emergency which might place other loads on the interface. A private communication channel or frequency may be required to satisfy the requirement.
T	Performance (Timing)	Timing is critical. Timing for most ITS communication services is based on the response to a request for data. Because of this, common communication media designed to handle voice data will likely support these requirements. The beacon interface has special requirements of identifying the vehicle as well as exchanging information before the vehicle gets out of range. This is more of a problem with vehicles traveling at speed. The architecture constrains such time critical access to data such that the data is available at the beacon site. This obviates the need for explicit specification of other timing information to support data transfer over a short range beacon.

3. COMMUNICATIONS LAYER

The overall ITS physical architecture consists of three layers, the Transportation Layer, the Communication Layer – which is presented in this section and has wireless and wireline components – and the Institutional Layer. This section presents an overview of the Communication Layer and is divided into two main sections: Communication Architecture (Section 3.1), and Communication Layer linkage to the Transportation Layer (Section 3.2).

The Communication Architecture Section (Section 3.1) presents a generic communication model which illustrates the basic relationship between the ITS Physical Architecture's Transportation and Communication Layers. This generic communication model, which should not be confused with the ITS communication network reference model, is based on the International Standards Organization's (ISO) Open Systems Interconnection (OSI) model. The ISO OSI model consists of seven layers: application, presentation, session, transport, network, data link, and physical layer. In general, the application, presentation, and session layers are supported by the Transportation Layer while the transport, network, data link and physical layers are supported by the Communications Layer.

The Communication Architecture Section (Section 3.1) also provides definitions of the various components that make up the communication layer. Some of these components include: communication services, communication logical functions, communication functional entities, and communication network reference model. The communication network reference model is the primary ITS communication model.

The communication architectures for commercial communication systems such as Personal Communication Services (PCS), Group Special Mobile (GSM), TIA-IS-41, Cellular Digital Packet Data (CDPD), to name a few, use communication network reference models. A network reference model is used to identify physical equipment that perform communication functions, and is used to identify reference interfaces between these physical equipment (standards are usually written for these reference interfaces). The ITS network reference model is based on, and presents extensions of, several reference models that were developed for the above mentioned standard communication systems. The model provides a structure that shows how various communication technologies can implement the ITS Architecture Interconnect Diagrams (AIDs), which are presented later in the Communication Layer Linkage Section (Section 3.2).

The Communication Layer Linkage Section also identifies the relationship between the Transportation Layer and Communication Layer definitions. This is accomplished through the following steps:

1. Mapping the communication services to the data flows identified in the Transportation Layer.

2. Generating the Architecture Interconnect Diagrams (AIDs) which define the interconnections between transportation subsystems and modules defined in the Transportation Layer.

3. Identifying the Architecture Renditions (ARs) which are examples, based on the network reference model, of how to provide communication connections between users defined in the Transportation Layer.

4. Mapping of the AIDs to the AR’s (each AR stays one level above technology specification, and comprises a family of systems with similar attributes, e.g., wireless packet data networks).

5. Identifying the Architecture Interconnect Specifications (AISs) which are examples of specific systems to implement an applicable communication technology to a particular rendition, for example, the use of CDPD for cellular wide-area wireless data communication.

To summarize, the Communication Layer Linkage Section presents the communication services/data flow mapping, AIDs, ARs, AID/AR mapping, and AISs.

In general, the Communication architecture for ITS will have two components: one wireless and one wireline. All Transportation Layer entities requiring information transfer are supported by one, or both, of these components. In most cases, the wireless component merely provides a tetherless user, usually one in a vehicle, with access to fixed (or wireline) network resources. The wireless portion will be manifested in three different ways:

- Wide-area wireless infrastructure supporting wide-area information transfer (many data flows). For example, the direct use of existing and emerging mobile wireless systems.

- Short range wireless infrastructure for short-range information transfer (also many data flows, but limited to specific applications), similar to systems used for electronic toll collection.

- Dedicated wireless system handling high data rate, low probability of error, fairly short range, Advanced Highway Systems related (AHS-related) data flows, such as vehicle to vehicle transceiver radio systems.

Because of the variances in the ITS user service requirements (from a communication perspective), it is clear from a cursory examination that the user services do not share a common information transfer capability. Specifically, ITS user services like electronic toll collection demand communication needs that can only be met by dedicated infrastructures for technical and feasibility, notwithstanding institutional, reasons. The ITS user services information transfer needs are supported by a sample deployment of the communication network reference model described in Section 3.1.4. Implementation candidates are identified as a result of a broad, balanced communication technology assessment task. After examining the assessment results for these candidates, an ITS implement or service provider can decide on the mix of communication technologies that are best suited to the implementation scenario at hand.

The wireline portion can be manifested in many different ways, most implementation dependent. Note that in defining the Communication Layer, no assumptions have been made regarding media type.

The process of developing the communications layer (architecture) is illustrated in Figure 3.0-1, and starts from the data flows in the transportation layer. In the following sections, the reader is referred to this figure at each step of the design process description.

The upper left block in Figure 3.0-1 shows the mapping of the identified data flows to communication services. The data flows are derived from the Architecture Flow Diagram (AFD) provided in the Physical Architecture document, which is used to specify which transportation subsystems communicate directly with each other. The communication services are described in Section 3.2.1 in terms of flow response and capabilities (they should not be confused with the ITS user services, which from a communication standpoint, are applications, as will be discussed shortly.) The mapping provides one or more communication services for each of the data flows between transportation entities. The Architecture Interconnect Diagrams (AIDs) encapsulate the type of partition between each of the transportation layer subsystems, as wireline or wireless, accompanied by a description of the communication service and operation mode for all the data flows between each pair of entities.

In parallel, a Network Reference Model (lower left block in Figure 3.0-1) is derived from models for standard commercial communication systems to fit ITS needs. This communication model is then used, in combination with feasibility and cost constraints, to develop renditions, or examples, of how to realize the required communication services. These renditions are based on the communication interface type, and are at one level above specific technology.

As shown in Figure 3.0-1, the MAP AID-AR block is done in an abstract way, identifying which data flows are supported by each rendition. At the same time, the results of the Technology Assessment are used to develop Architecture Interconnect Specifications, which identify and assess specific features of the technology important to interconnecting the transportation layer entities. The AIS involves further specification of the renditions, and completes the description of the ITS Communications Architecture. To illustrate, mobile wireless packet data networks are considered a rendition. Several technologies, like CDPD, RAM, and so on, are specific technologies that belong to this rendition that could be used in the implementation. The AIS Section here is maintained brief, and includes a few examples of the results of the communication technology assessment (from an ITS architecture standpoint) which is presented in Chapter 7 and Appendix D of the Communications Document. The AIS leads to technology recommendations, to be interpreted as implementation examples of the communication elements in the ITS architecture. In a real-world ITS implementation, this last step would be performed by the communication system designer.

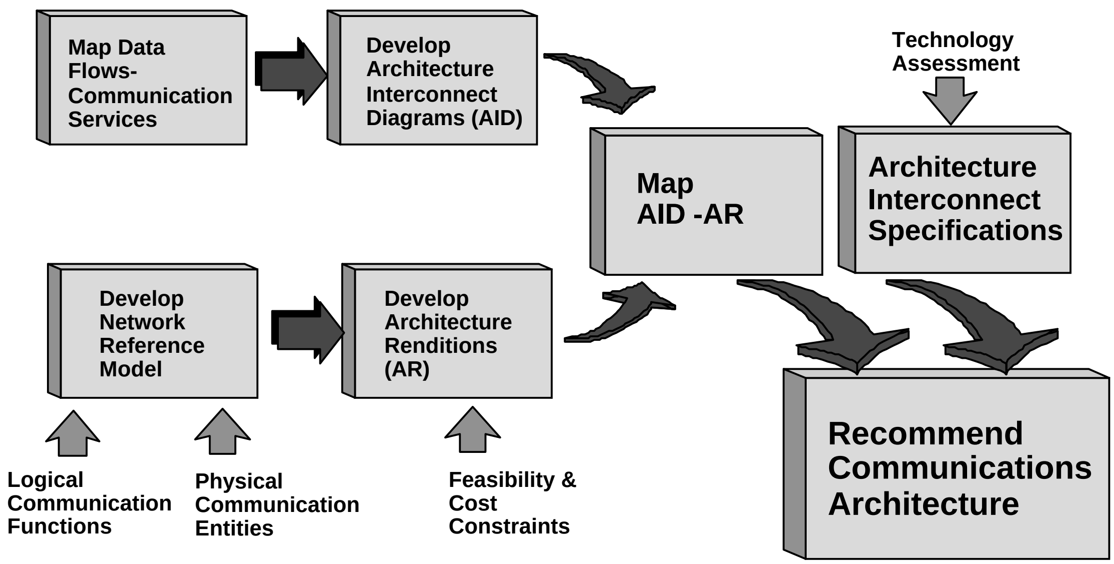


Figure 3.0–1 Communications Architecture Design Process

3.1 Communication Architecture

The generic communication hierarchical model presented in Figure 3.1-1 shows the relationship between the Transportation and Communication Layers. Each data user can be one entity in the Transportation Layer (e.g., the Information Service Provider Subsystem or Personal Vehicle Subsystem in an information exchange). The user does not care about and should not be concerned with the specifics of this information transfer layer. In fact, the Communication Layer can be viewed as plumbing that carries information from one user to another.

The complex makeup of the network is usually defined by system architectures developed to meet specific requirements, performance objectives, and socio-economic drivers. In the absence of crisp specifications and because of the jurisdictional-independence of this particular architecture, the end framework precludes the design of low level implementation details. However, to properly evaluate the communication architecture candidates, select technologies and detailed designs are recommended in an evaluatory design (see the later chapters of the Communications Document.)

The generic hierarchical communication model shown in Figure 3.1-1 follows the Open Systems Interconnection model which organizes the communication network in a highly structured format to reduce its overall design complexity. This model is structured as a series of layers each with the function of providing certain services to the layer above and capable of conversing with the corresponding layer at the other end of the link. Thus the high level layers (e.g. ITS application) are shielded from the actual implementation details of the communication services. Different networks can use layers different from the OSI model, such as the IBM SNA (Systems Network Architecture). When different protocols are used in different networks, an inter working function must provide the conversion between the protocols at the various levels.

The lowest layer in the OSI model is the physical layer (layer 1), which provides the transmission of bits over wires or radio links . Layer 2 is the data link layer, and is concerned with making the link appear to the receiver as bit error-free as possible by implementing error detection and correction (EDAC) coding schemes in the transceiver; one example is the use of a cyclic redundancy code (CRC) to a block or frame of the data and when the data passes the CRC check at the receiver, the returned acknowledgment indicates whether re-transmission is needed. Layer 3 is the network layer, which controls the operation of the network, where the key issue is routing packets, which is also used to generate billing information for the communications service provider; billing is tied to IP addresses. Layer 4 is the transport layer, which mediates between the session layer and the network layer, providing end-to-end accounting for all the data at the receiving end, and isolates the system from the changing physical technologies. Layer 5 is the session layer, which allows users on different machines to establish communications, or sessions, between them, involving ordinary data transport but with enhanced services such as remote log-in or file transfer. Layer 6, the presentation layer, performs syntax and semantic operations on the information transmitted between the users, such as encoding data in a standard way, or compressing or encrypting that data. Layer 7 is the application layer, which provides commonly used protocols for such tasks as terminal emulation, file transfer, electronic mail and remote job entry. (Note that for many ITS applications, layers 5 and 6 are absorbed into the application layer, layer 7.)

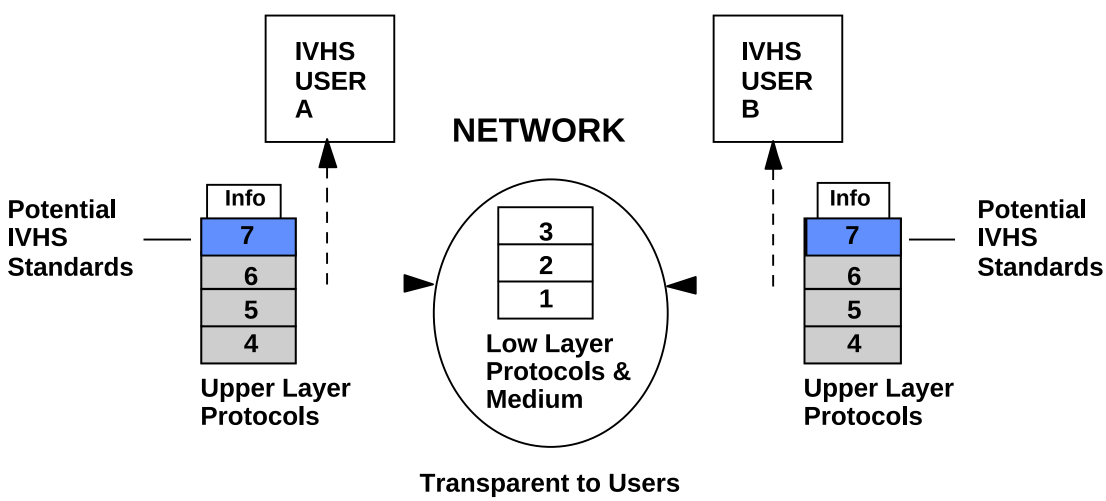


Figure 3.1-1 Generic Hierarchical Communication Model

From the Communication Layer perspective, the term "services" is defined according to communications governing bodies (*e.g.*, ITU, TIA, etc.), and should be used with care. That is, when describing a communications architecture, one should not refer to Route Guidance or Pre-trip Planning as services. Rather, they are applications in need of a communication service. Elaborating more along these lines, ITS appears to the Communication Layer as a collection of applications with markedly different communication requirements. Thus the service provided by the communication model is characterized more by 1) the application's directionality requirements (*e.g.*, one-way or two-way) for information transport, 2) whether it is between mobile elements, mobile and stationary elements or stationary elements, 3) the amounts of data to be transported, and 4) the urgency rather than the precise description as Route Guidance or Pre-trip Planning.

The next section identifies various communication services to which the Transportation Layer data flows can be matched. This matching process will assign broad generic communication services to the data flows without specifying a particular technology.

3.1.1 Communication Services

The communication services define the exchange of information between two points and are independent of media and application (i.e., ITS user service). In essence, they are a specified set of user-information transfer capabilities provided by the communication layer to a user in the transportation layer. Figure 3.1-2 illustrates the hierarchy of communication services, the detailed of these is given in Appendix A-1 in the Communications Document. In what follows a brief description of the services is presented.

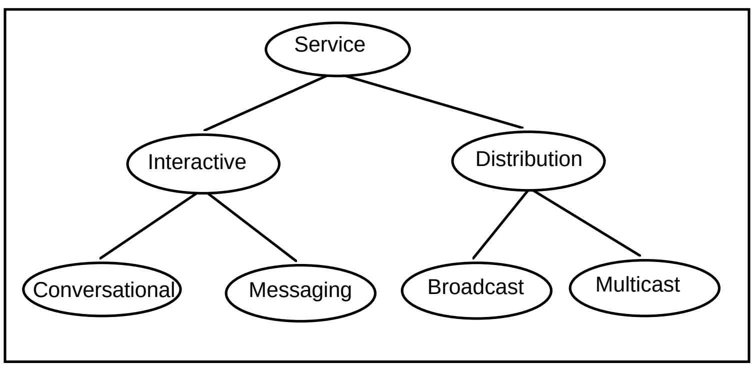


Figure 3.1-2 Communication Services Hierarchy

Communication services consist of two broad categories, interactive and distribution. Interactive services allow the user to exchange data with other users or providers in real or near real time, asking for service or information and receiving it in the time it takes to communicate or look up the information. Distribution services allow the user to send the same message to multiple other users.

Interactive services may be either conversational or messaging. Conversational implies the use of a two-way connection established before information exchange begins and terminated when the exchange is completed. Messaging, on the other hand, works more like electronic mail being exchanged between users. The messages are exchanged without establishing a dedicated path between the two sites. Each message is addressed and placed on the network for transmission, intermixed with messages from other users. The communications community labels this mode of communication a “datagram” service.

Distribution services may be either <u>broadcast</u> or <u>multicast</u> and may be used over wireline and/or wireless communication links. Broadcast messages are those sent to all users while multicast messages are sent only to a subset of users. Multicast differs from broadcast in its use of a designated address for all users and user groups. Examples of broadcast information might include current weather or road conditions, whereas multicast information might be information sent to all drivers working for a specific company. A changing group membership could be the set of users traveling between two locations or with a certain destination, for which unique information must be transmitted. The services that can be supported using circuit or packet connection mode include voice, video, image and data. (see Appendix A-1 of the Communications Document for a complete description.)

Not shown in the Figure 3.1-2 are location services. These fall in two categories: (1) the services that do not use the communication network (i.e., GPS, and stand alone terrestrial systems); (2) location services that use the network for providing the service (e.g., cellular based systems). In the latter case, the location services fall under the interactive services. The service will be rendered by a service provider in response to a request for information or help.

3.1.2 Logical Communication Functions

Based on the objectives of the communication architecture, a list of logical functions to support the ITS system communication requirements are identified. The primary logical communication functions can be confined to: wireless and wireline access, switching, routing, registration authentication, inter working, validation, billing, and operations (see Appendix A-2 of the Communications Document for a detailed description.).

3.1.3 Functional Entities

The functional entities that make up the communication layer were derived from existing and emerging infrastructure specifications and standards (*e.g.*, TIA, ITU, Bellcore, ANSI). These basic building blocks form the foundation of a generic communication system. As with the transportation layer, each functional entity consists of one or more logical functions. These entities include: 1) user device, 2) user profile module, 3) switch, 4) wireless controller, 5) wireless base station, 6) inter working platform, 7) profile data base, and 8) wireline network. The detail description of these functional entities is presented in Appendix A-3 of the Communications Document.

3.1.4 Communication Network Reference Model

As shown previously in Figure 3.0-1, the communication architecture design process consists of several steps. The previous sections listed the communication logical functions and physical entities. The architecture design process now starts on the lower leg of Figure 3.0-1 with the development of the Communication Network Reference Model. This model provides an architecture or structure that shows how various communication technologies can implement the Architecture Interconnect Diagrams developed in the next section.

The network reference model for ITS is depicted in Figure 3.1-3, and is a generic abstraction which builds upon several reference models developed for standard commercial systems. Boxes represent the various physical equipment (with descriptive uppercase letters) that perform the communication functions. Identified by lowercase letters (s, v, u_1, u_2, u_3) are the interfaces important to ITS. They are described in the following paragraphs.

Communications Layer

The most important reference point is the wireless interface (*u*) connecting the WBS and the wireless transceiver. To meet the objectives of the national ITS Architecture it will be necessary in some cases that the air interface become standard. The wireless portion of the architecture is manifested in 3 different ways. The *u* interface is realized in three ways: u_1, u_2, u_3, with each interface corresponding to one of the wireless manifestations, as follows:

- u_1 defines the wide area wireless air-link with one of a set of base stations providing connections to mobile or mobile or untethered users. It is typified by the current cellular telephone and data networks or the larger cells of Specialized Mobile Radio for two way communication, as well as paging and broadcast systems.

- u_2 defines the short-range air-link used for close-proximity (less than 50–100 feet) transmissions between a mobile user and a base station, typified by transfers of vehicle identification numbers at toll booths; and

- u_3 addresses the vehicle-vehicle (AHS-type) air-link, for high data rate, burst, usually line-of-sight transmission with high reliability between vehicles, where standards are in their infancy.

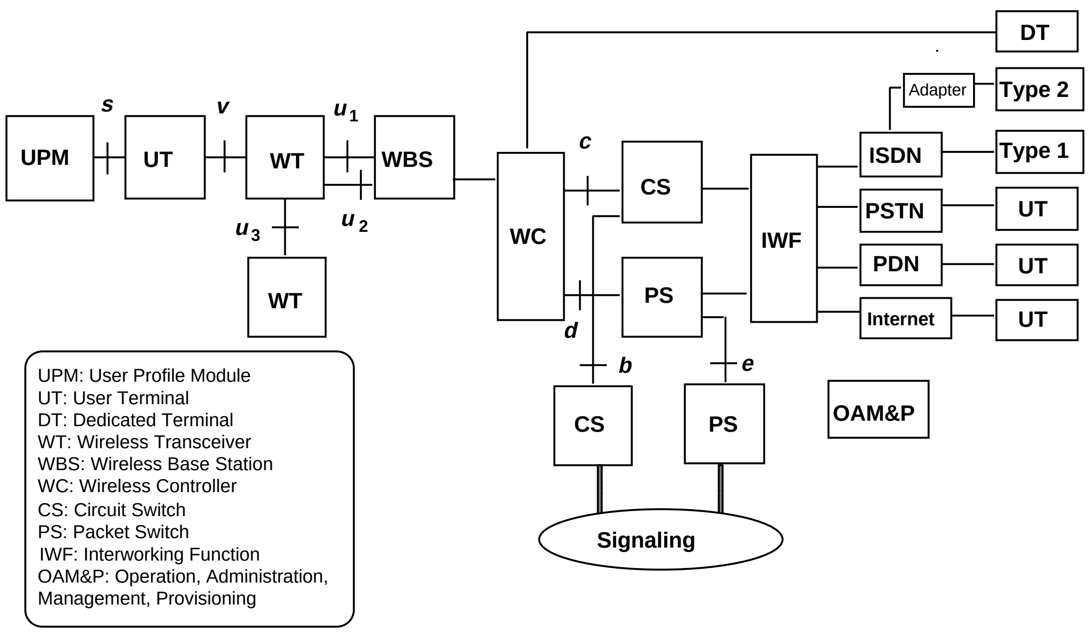


Figure 3.1.-3 Network Reference Model for the Communications Layer

The National ITS architecture provides for implementation flexibility. Various of the data flows in the Architecture can be carried over multiple of these interfaces, and the final choices would be made by the local implementers. This flexibility is depicted conceptually in Figure 3.1.-4.

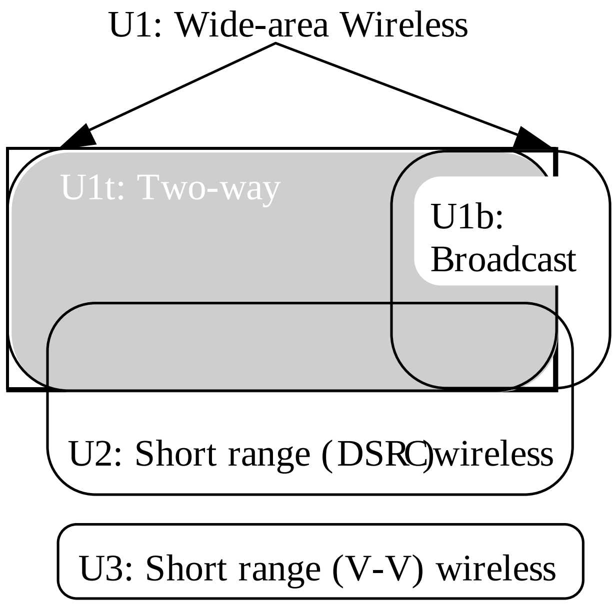


Figure 3.1-4 Implementation Flexibility

Since the wireline segment encompasses standard wireline configurations, the ITS-critical elements from a standards perspective are those comprising the wireless portion on the left side of Figure 3.1-3. The wireless portion consists of the User Profile Module (UPM), the User Terminal (UT), the Wireless Transceiver (WT) and the Wireless Base Station (WBS). The connections through the Dedicated Terminal and various User Terminals are shown in the column of boxes on the right. The equipment in the center is the existing public telecommunications services, so the details are transparent to ITS, which is a major benefit to the ITS community. *All management, operations, expansion, and improvement costs are shared with the wider set of all telecommunications users.*

This is an important point to jurisdictions and agencies who prefer to procure and trench their own network along the right-of-way. Whereas a financial sensitivity analysis may point to a private solution, it frequently does not fully consider the large and sustained Operation, Administration, Management , and Provisioning (OAM&P) fees that the agency will have to pay the telecommunications vendor during the system's life cycle.

Appendix A-4 of the Communications Document presents a detailed description of the wireline side of the above network reference model, in addition to a more thorough treatment for the required interfaces, such as switches, controllers, and terminals. This appendix also presents the network entities, interfaces, and signaling plane, and includes a discussion on circuit connection and data packet transmission.

3.2 Communication Layer Linkage

This Communication Layer Linkage Section further identifies the relationship between the Transportation Layer and Communication Layer definitions. This is accomplished by mapping the communication services to the data flows identified in the Transportation Layer, generating the Architecture Interconnect Diagrams (AIDs), identifying the Architecture Renditions (ARs), mapping the AIDs to the ARs, finally identifying the Architecture Interconnect Specifications (AISs) (based on the technology assessment).

3.2.1 Mapping Communication Services to Data Flows

Mapping of the communication services to the data flows establishes the first link between the transportation layer and the communication layer, and this initial link depends on the completion of two technical architecture milestones. First, the message sizes and data transfer requirements are broadly identified. Second, the physical architecture that allocates logical functions (see Logical Architecture Document) to subsystems necessitates a partitioning exercise, which defines the data flows that require communication. This mapping is an iterative procedure, calibrated by feedback from the logical and physical architectures (and in turn the ITS stakeholders) by retracing the steps shown in Figure 3.0-1.

Appendix A-5 of the Communications Document details the mapping process. It also depicts the assigned communication service for each data flow with the corresponding rationale.

3.2.2 Architecture Interconnect Diagrams

As denoted in Figure 3.1-1, this section presents the development of the Architecture Interconnect Diagrams (AIDs). These diagrams show the subsystem-to-subsystem communication interfaces of all transportation subsystem entities (defined in the transportation part of the Physical Architecture). The diagrams identify the communication mode and partition, either wireline or one of three types of wireless connection, as well as documenting the rationale for of these choices when needed for clarification. The diagrams identify the requirements, developed from the physical relationships of the various subsystem entities, but do not force any specific communication technology to be used. The information contained in the AIDs can be traced to the information provided in the Data Flow-Communications Service Mapping Table (Appendix A-5 of the Communications Document).

A template is used to illustrate the interconnections between entities and between modules and is described next. At this stage in the physical architecture, no AIDs are defined for inter-module information transfer within a simple entity. In fact, from the communication layer perspective, this is not necessary. The most important goal is to identify the inter-entity interconnectivity.

The subsections that follow describe the AID template, present the Level 1 and Level 0 (top level) AID’s.

3.2.2.1 AID Template

As depicted in Figure 3.2-1, each AID shows the two communicating transportation subsystem entities, the interconnection partition (i.e., wireline, wireless, or both), and a characterization of the interconnection. The latter is not a link-specific description, which the AIS provides, but a high-level interpretation in terms of services and operation modes. When not obvious, the choice of operation mode is based on the rationale provided in Table 3.3-1. The interconnect description for each AID provides a data flow, service and operation mode description for each data flow between the two entities. The Data Flow information also provides directionality when more than one data flow exists between the entities, not all of which are in the same direction. If all the data flows are in the same direction, no indication is given and the data flows from the left entity to the one on the right.

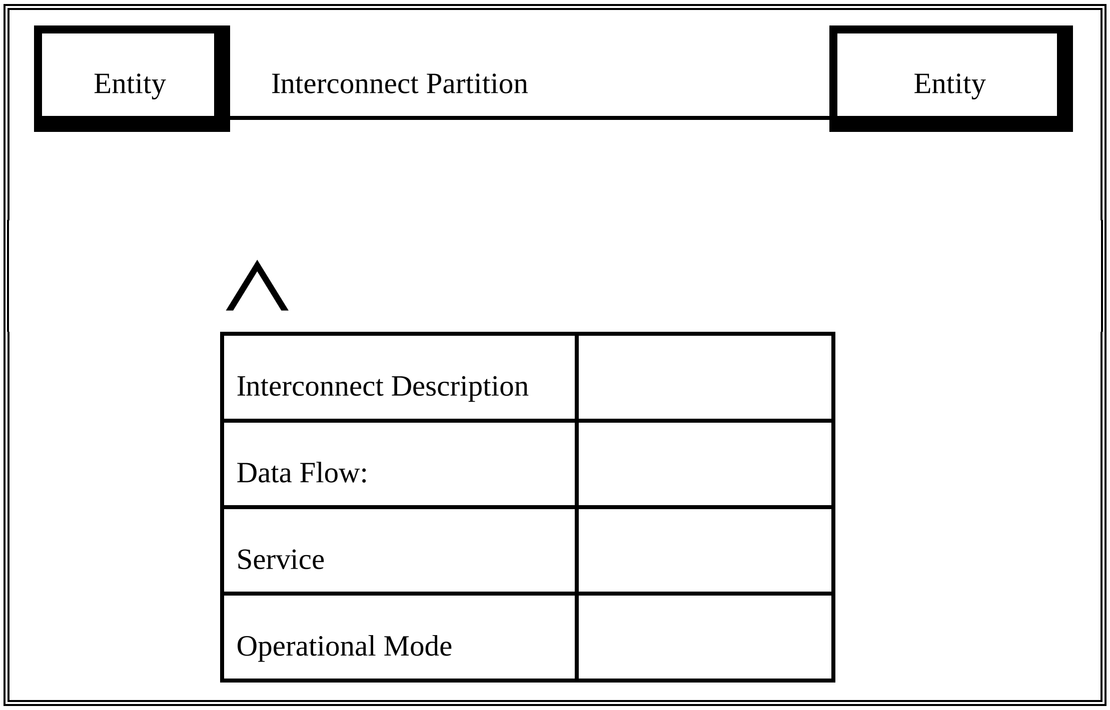


Figure 3.2-1. Template for the Architecture Interconnect Diagram (AID)

3.2.2.2 Level 1 AIDs

Using the AID template and Table A.5-1, Data Flow – Communication Services Mapping Table, the data flows are represented in an Architecture Interconnect Diagram (AID) format. A single example is presented here in Figure 3.2-2, and various others are compiled in Appendix B of the Communications Document.

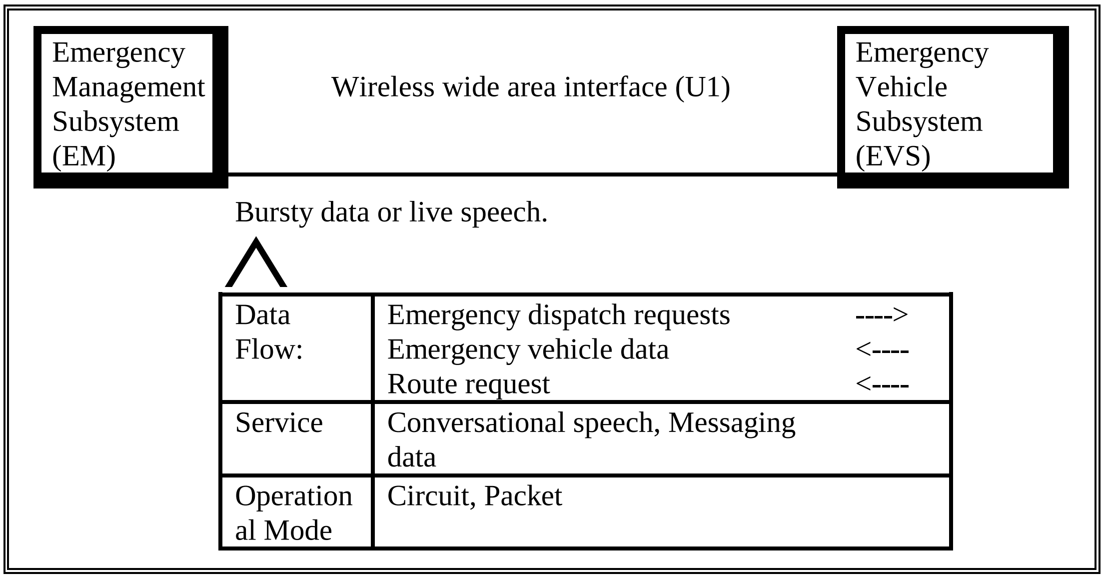


Figure 3.2.-2 Example of AID Level-1

3.2.2.3 Level 0 AID

The Level 0 Architecture Interconnect Diagram shows all communications connectivity required by the Physical Architecture. Several different versions of these diagrams have been generated during the course of architecture development. Figure 3.2-3 is a comprehensive interconnect diagram that shows all interconnects between subsystems as well as between subsystems and terminators. This results in a figure that has over 150 connections between more than 70 entities. The figure has been filtered to remove human interconnects and physical/environmental interconnects since these are not the focus of the communications layer. Figure 3.2-4 shows only the interconnects between subsystems in the architecture which are the primary focus of the communications analysis. This figure is a percolation to a top level of all the detailed, level 1 AID's. It presents all the interfaces between the physical subsystem entities, capturing the wireline (w) or wireless (u_1, u_2, or u_3) nature of the interfaces in the ITS architecture. As such, it is a comprehensive, albeit not complete, representation of the ITS communication architecture. More detailed variations can be easily derived from it. For example, Figure 3.2-5 shows the data flows using the U1b wide area wireless broadcast "sub-interface". Figure 3.2-6 shows the subset that uses either of U1t (two-way wide area wireless) or U1b (wide area wireless broadcast). Note that U1b does not imply a certain technology – FM subcarrier, paging, messaging data networks are possible implementations; they all tend to use a broadcast protocol in the forward, i.e.; fixed to mobile, direction.

Communications Layer

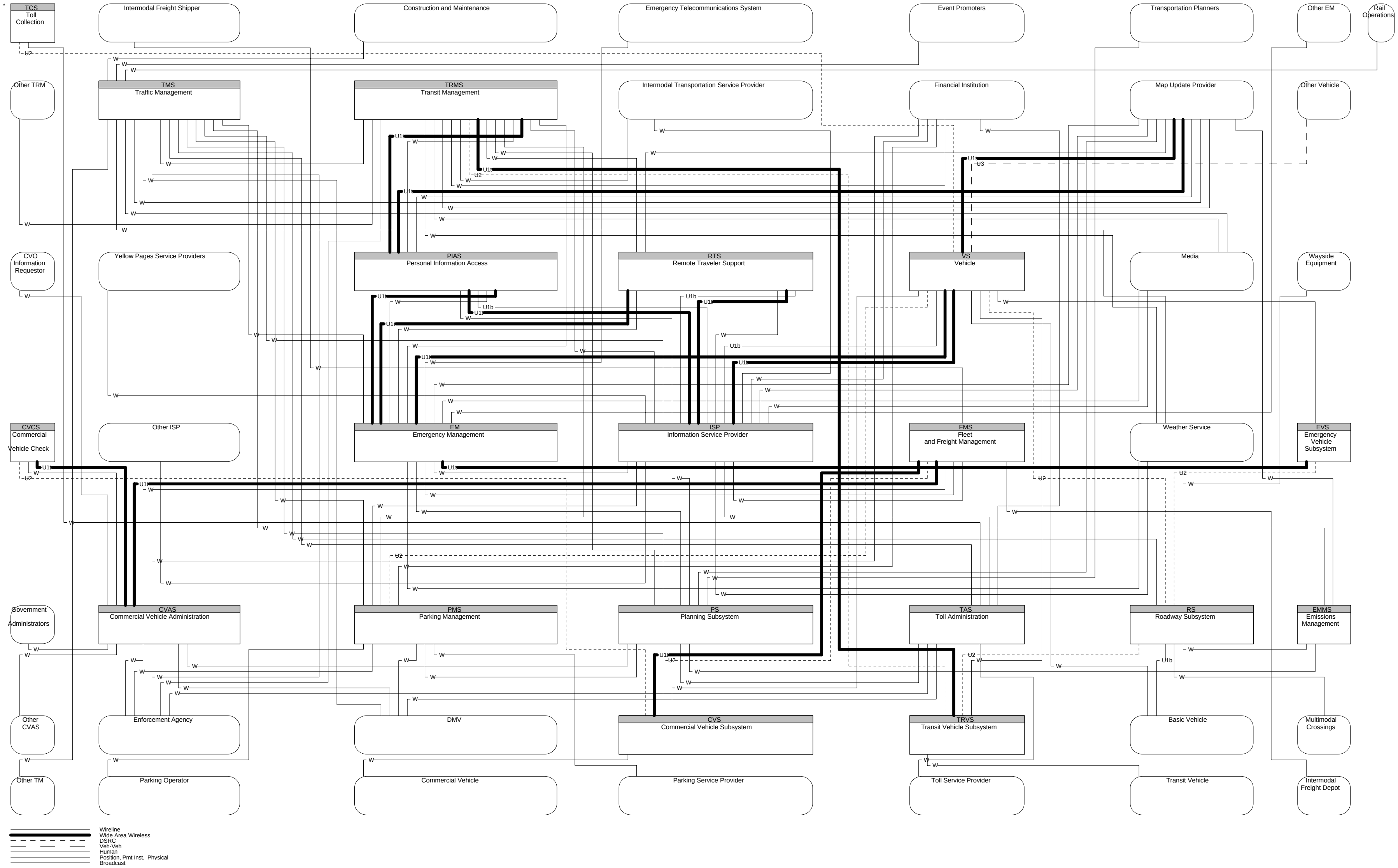


Figure 3.2-3 Level 0 Architecture Interconnect Diagram for the National ITS Architecture
(Interconnects Between Subsystems & Between Subsystems And Terminators)

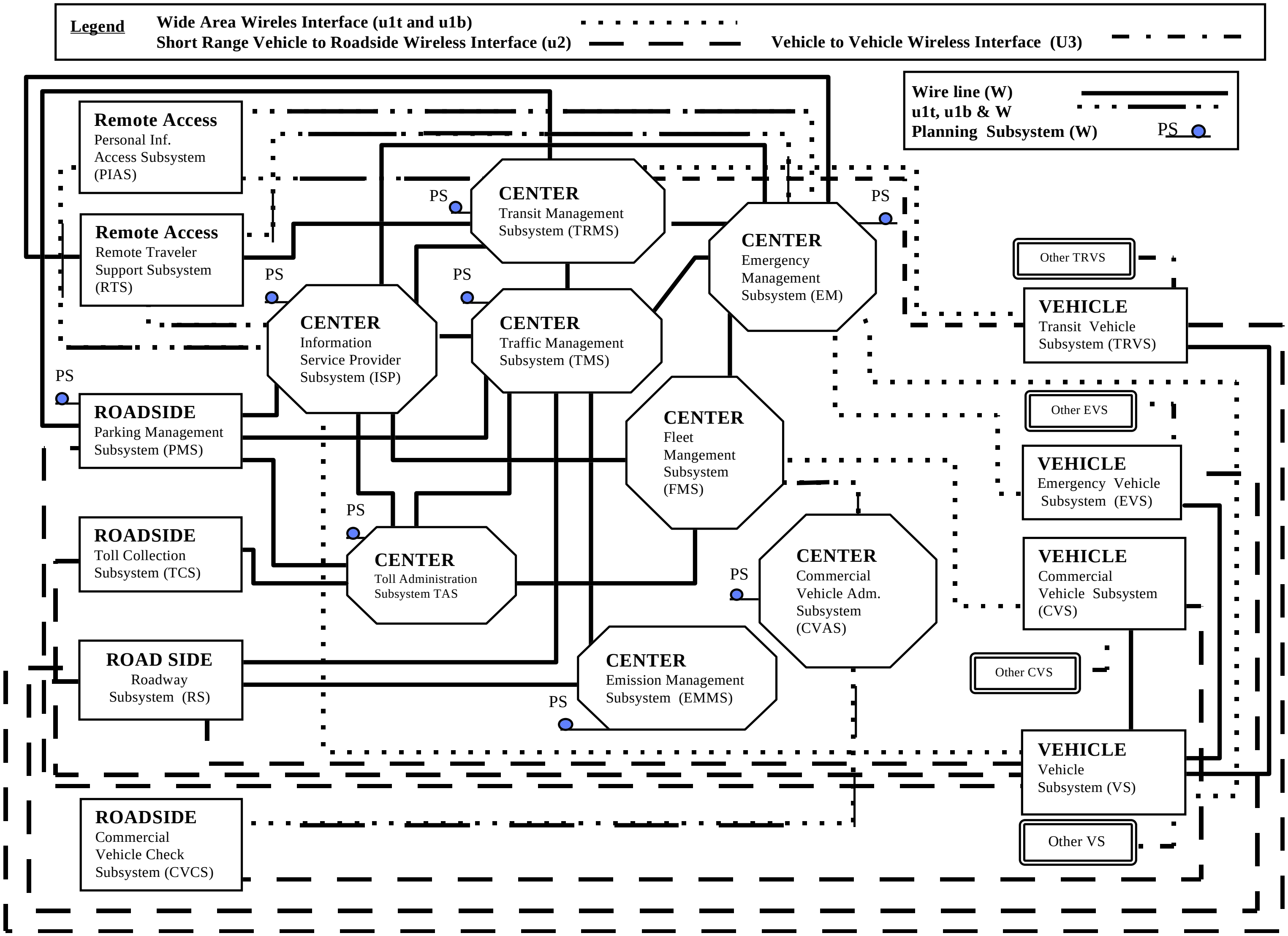


Figure 3.2-4 Level 0 Architecture Interconnect Diagram for the National ITS Architecture
(Interconnects Between Subsystems Only)

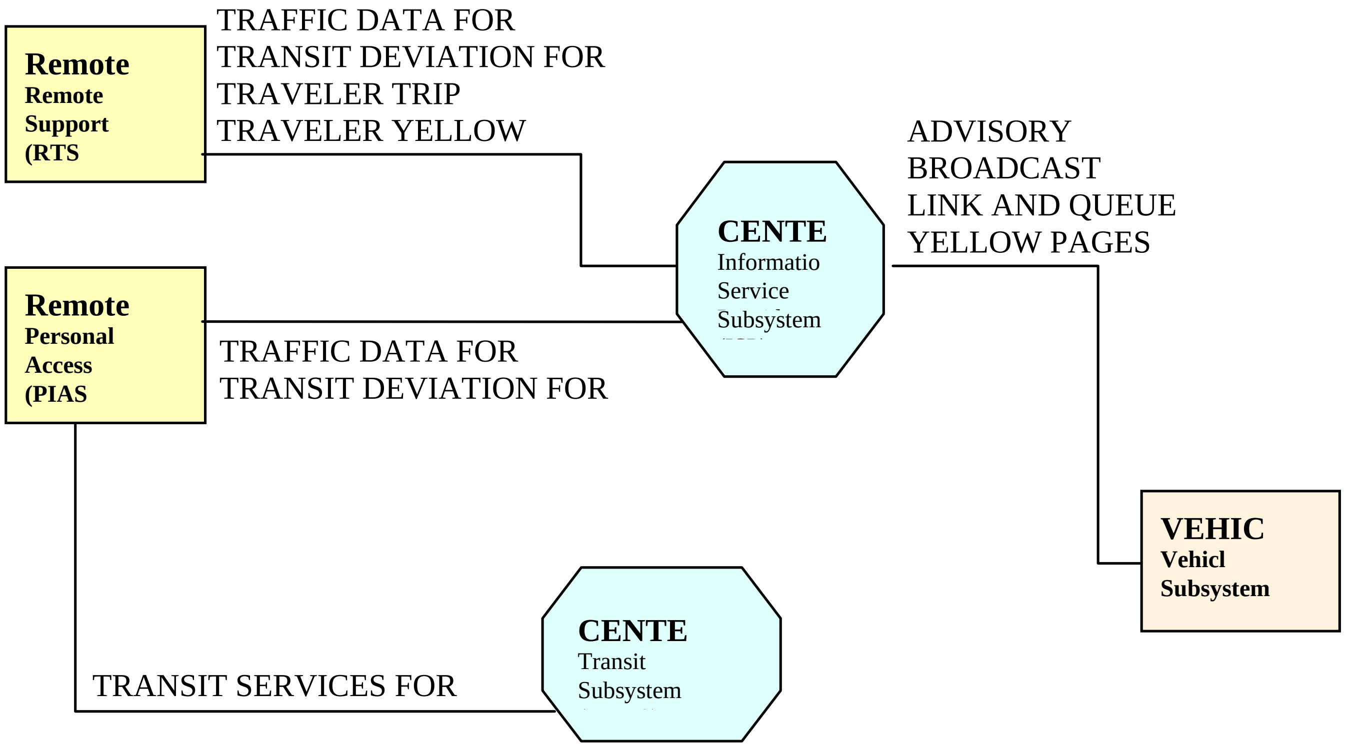


Figure 3.2-5 Level 0 Architecture Interconnect Diagram for the National ITS Architecture
(Subset showing U1b data flows)

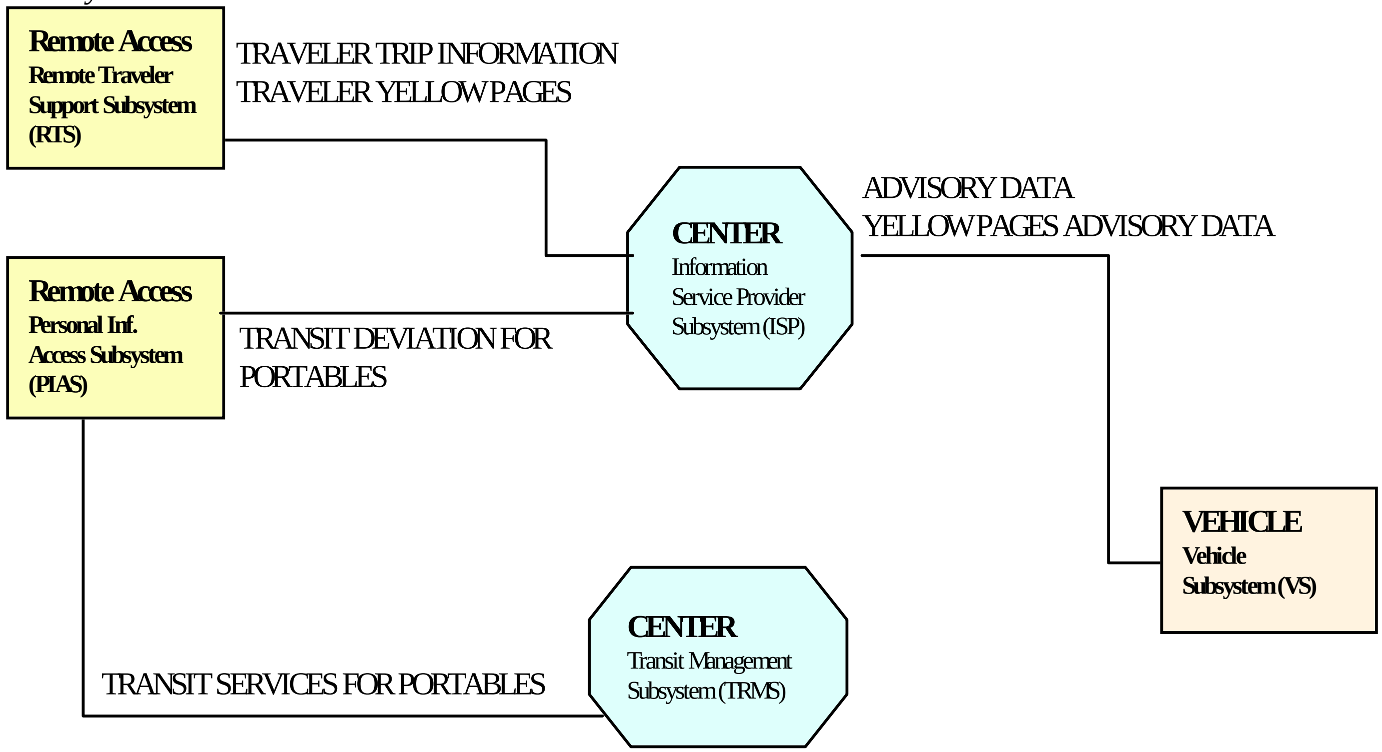


Figure 3.2-6 Level 0 Architecture Interconnect Diagram for the National ITS Architecture;

(Subset showing data flows using either U1t or U1b)

3.2.3 Architecture Renditions

The next step in the communications architecture design process is the development of the communication Architecture Renditions, as depicted in Figure 3.1-1. Combining elements from the Generic Communication Model (Figure 3.1-2) and the ITS Communication Network Reference Model (Figure 3.1-4) provides a more detailed view of the flow of information between two users. This information includes communication services and operational modes (i.e., circuit switched, packet switched, etc.). The architecture renditions are essentially examples of how to provide connections between users based on the communications network reference model and the evaluations of classes of feasible implementations.

Two levels of renditions are generated. A Level 1 rendition is generated for each of the possible interconnections between services. The Level 0 Rendition (the top level) shows the full connectivity between users over multiple links. The details of the renditions, how they are generated, and those that apply to the different interconnections in the architecture are provided in Appendix C of the Communications Document; an example of a Level-1 rendition and the Level-0 will be provided here to support the subsequent task of AIS generation.

3.2.3.1 Level 1 Rendition

Figure 3.2-6 depicts level 1 renditions for the wide-area wireless communication link (u1) through switched networks. This figure depicts interconnection between tetherless users or tetherless and stationary users, utilizing two distinct classes of wide-area wireless technologies. Several technologies or systems can fit within each rendition. For example, CDPD, RAM, ARDIS and so on are possibilities for implementing the packet-switched wireless data network (shown on the right-hand-side of the diagram in Figure 3.2-7.

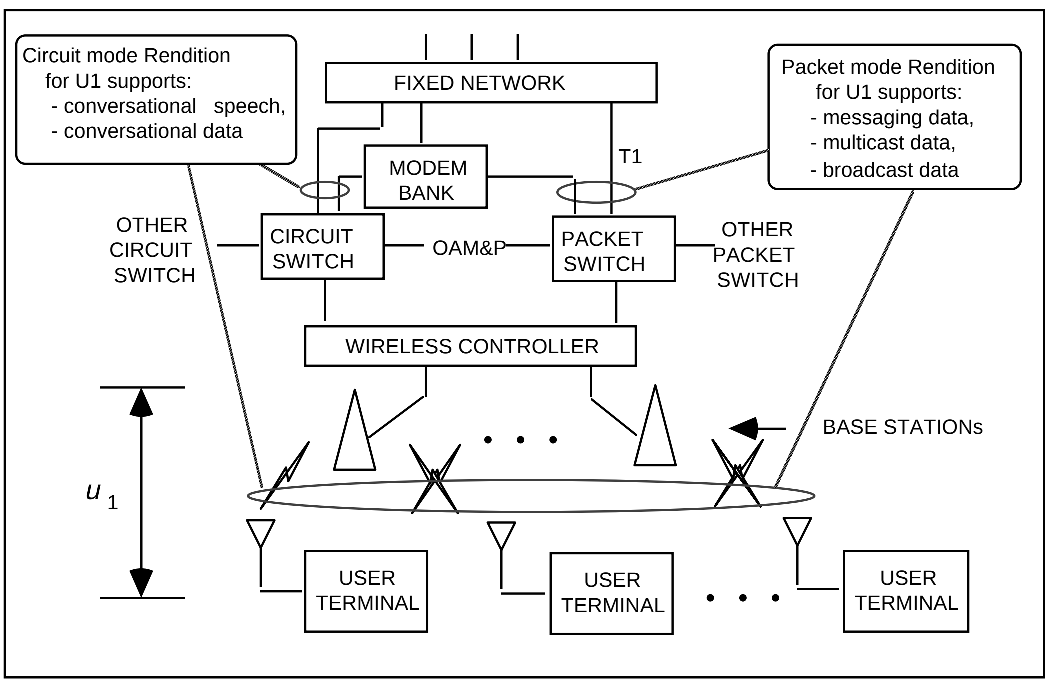


Figure 3.2-7 Rendition 1 — Wide-Area Wireless (u1) Link Through Switched Networks

3.2.3.2 Level 0 Rendition

Figure 3.2-8 illustrates the Level 0 rendition. It represents a composition of all the renditions to reflect the combined needs of the architecture. This rendition shows a user communicating to another user, central office or a base station over various communication links such as u1, u2, u3 and w. Again, the details of this mapping are provided in Appendix C.1.1 of the Communications Document.

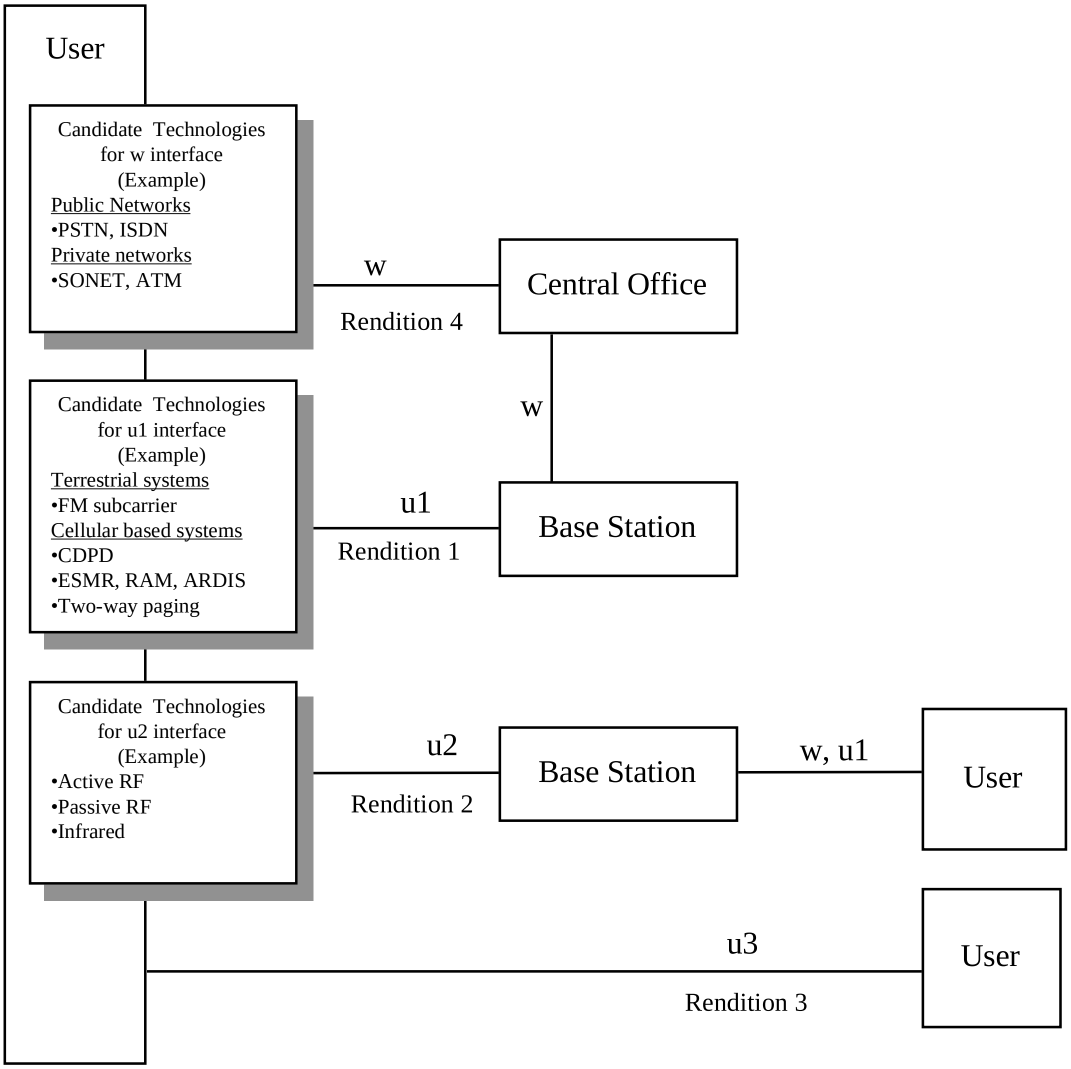


Figure 3.2-8 Level 0 Rendition

3.2.4 Architecture Interconnect Specifications

The Architecture Interconnect Specifications (AISs) are now developed from the technology assessment and refined by combining the renditions with applicable technologies and evaluating the achievable performance (Figure 3.0-1). This involves mapping the applicable communication technologies to the renditions.

To facilitate the mapping of the communication technologies to the renditions, the candidate wireless and wireline technologies are surveyed. The candidate technologies are further assessed and their performance is evaluated from the National ITS Architecture standpoint. For example, the assessment includes: short range and wide area, one-way and two-way wireless data communication. Systems analyzed include terrestrial networks (e.g., cellular, ESMR), FM broadcast, and satellite systems for mobile and fixed services. The details of this survey are presented in Appendix D of the Communications Document, with the assessment results summarized in Chapter 7 of the Communications Document.

The results of this assessment are used in identifying the candidate technologies to support level 0 and level 1 renditions. The results of this mapping are summarized below.

It is apparent from the matrices provided the Technology Assessment Section (Section 7.5 of the Communications Document) that for the foreseeable future, wireless data networks (such as CDPD, RAM, etc.) form the class of communication systems most suitable to interactive wide area wireless ITS links (u1t). The infrastructure is already largely available (short, in some areas, of adding the appliqué equipment). Service costs are already low, and equipment costs are coming down. Coverage nationally is excellent with the possible shortcoming that it may not for some tine be available in rural and remote areas. Yet with the advent of innovative solutions like circuit-switched CDPD, which utilizes the AMPS cellular infrastructure in a manner transparent to a CDPD subscriber, this problem would be largely mitigated. In any event, for ITS users how insist on uninterrupted coverage in remote areas, holes in the coverage of terrestrial cell-based systems can be supplemented by satellite communication systems.

Figure 3.2-9 depicts the use of CDPD in a u1 communication architecture rendition to create an example of Architecture Interconnect Specification.

According to internal market research and analysis, users are concerned with two overshadowing factors: cost and quality. The Architecture Development Team believes that the market will determine the winning technologies which will gain wide scale acceptance.

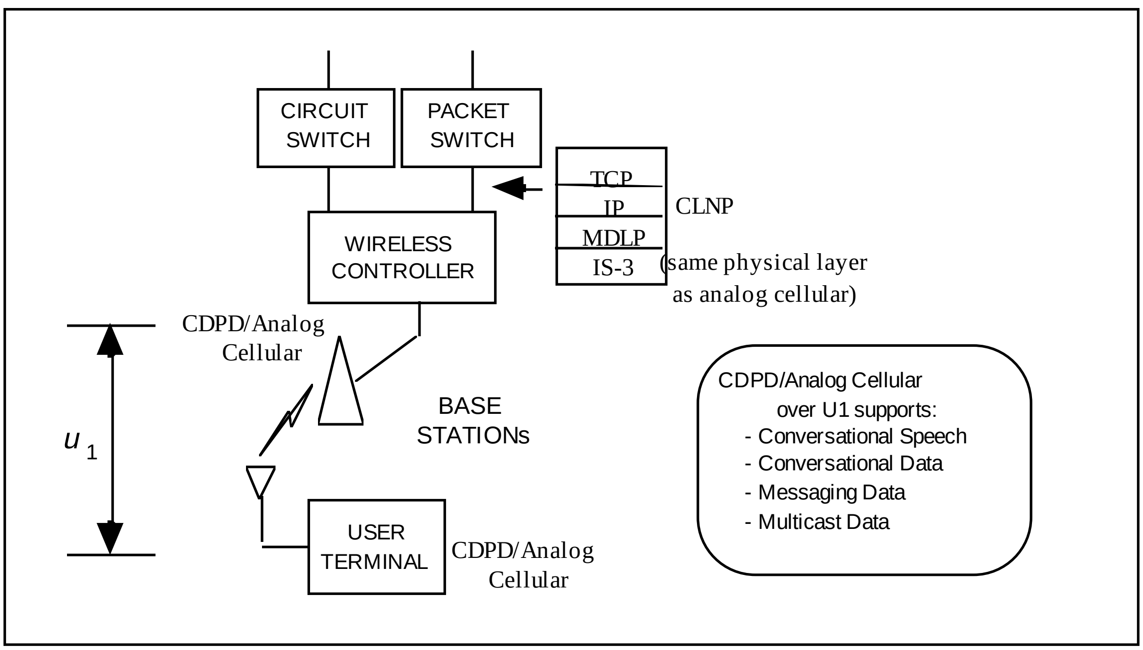


Figure 3.2-9 AIS Example Using CDPD for Wide Area Wireless (u1)

Communications Layer

Table 3.2-1 provides an illustration of candidate technologies for the different wireless data flows within the context of the communication layer of the ITS Architecture. In addition to wireless data networks, for wide area ITS data flows, the short range wireless interface u_2 comprises a distinct set of communication services and supporting radio technologies. In a real world ITS implementation, the system designer makes use of the technology assessments of Chapter 7 of the Communications Document, and the ITS architecture communication renditions presented above, to select the specific communication systems/technologies most appropriate for the deployment at hand.

Table 3.2-1 Examples of Candidate Technologies for Wireless Data Flows

wide-area wireless interface (messaging services; bursty data transfers)		
Source	**Architecture Flow**	**Destination**
CVAS	**Electronic credentials**	FMS
CVAS	**Safety information**	CVCS
CVS	**Driver & vehicle information**	FMS
CVS	**On board vehicle data**	FMS
EM	**emergency dispatch requests**	EVS
EM	**emergency acknowledge**	VS
EM	**emergency acknowledge**	RTS
EM	**emergency acknowledge**	PIAS
EVS	**Emergency vehicle driver status update**	EM
EVS	**Emergency vehicle driver input**	EM
EVS	**Emergency vehicle dispatch acknowledge**	EM
FMS	**fleet to driver update**	CVS
PIAS	**Demand responsive transit request**	TRMS
PIAS	**Traveler information request**	ISP
PIAS	**Emergency notification**	EM
RTS	**Emergency notification**	EM
TRMS	**Demand responsive transit request**	PIAS
TRMS	**Request for vehicle measures**	TRVS
TRMS	**Route assignment**	TRVS
TRVS	**Emergency notification**	TRMS
TRVS	**Vehicle probe data**	TRMS
TRVS	**Traveler information request**	TRMS
VS	**vehicle probe data**	ISP
VS	**emergency notification**	EM
VS	**Traveler information request**	ISP
VS	**map update request**	X23

U1 - Circuit Switched data (Messaging; larger data transactions, e.g., compressed image)		
Source	**Architecture Flow**	**Destination**
ISP	**Traveler information**	PIAS
ISP	**Traveler information**	VS
X23 MAP Update Provider	**Map updates**	PIAS
X23 MAP Update Provider	**Map updates**	VS

U1 - Circuit Switched Voice (live voice interaction; early implementations)		
Source	**Architecture Flow**	**Destination**
EM	**Assigned route**	EVS
EM	**Hazmat information**	EVS
EM	**Emergency dispatch requests**	EVS
EM	**Emergency acknowledge**	RTS
EVS	**Emergency vehicle driver status update**	EM

EVS	**Emergency vehicle driver input**	EM
RTS	**Emergency notification**	EM
VS	**Emergency notification**	EM

U1 - FM Subcarrier Broadcast Services (Broadcast of free services and services that require subscription; e.g., traveler information,)		
source	**Architecture Flow**	**destination**
ISP	**Broadcast information**	PIAS
ISP	**traveler information**	VS
ISP	**broadcast information**	VS
ISP	**broadcast information**	RTS

U1 - Multicast Services (Distribution services that require subscription; e.g., map updates)		
source	**Architecture Flow**	**destination**
ISP	**traveler information**	PIAS
ISP	**traveler information**	VS
ISP	**broadcast information**	VS
ISP	**broadcast information**	RTS
X23 MAP update provider	**map updates**	PIAS
X23 MAP update provider	**map updates**	VS

U2 - RF beacon for close-proximity wireless communication between vehicle and roadside		
Source	Architecture Flow	Destination
CVS	**screening data**	CVCS
CVS	**on board safety data**	CVCS
CVCS	**safety inspection record**	CVS
CVCS	**Pull in message**	CVS
EVS	**Emergency vehicle preemption request**	RS
PMS	**Tag update**	VS
RS	**AHS control data**	VS
RS	**vehicle signage data**	VS
RS	**intersection status**	VS
TCS	**Request tag data**	VS
TRVS	**signal priority request**	RS
VS	**Tag data**	TCS
VS	**AHS vehicle data**	RS
VS	**Tag data**	PMS

Appendix A

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Basic Vehicle	Vehicle	vehicle measures	W	Broadcast data	product	I	R
Commercial Vehicle	Commercial Vehicle Check	CVO weight and presence	P	Physical Interface	NA		
Commercial Vehicle	Commercial Vehicle Subsystem	vehicle measures	W	Broadcast data	product	I	R
Commercial Vehicle Administration	Commercial Vehicle Check	credentials information	W,U1t	Conversational data, Messaging data	regional	04	
Commercial Vehicle Administration	Commercial Vehicle Check	CVO database update	W	Conversational data, Messaging data	regional	04	
Commercial Vehicle Administration	Commercial Vehicle Check	international border crossing data	W	Messaging data	regional	04	
Commercial Vehicle Administration	Commercial Vehicle Check	safety information	W,U1t	Conversational data, Messaging data	regional	04	
Commercial Vehicle Administration	CVO Information Requestor	credentials and safety information response	W	Messaging data	national	04	
Commercial Vehicle Administration	DMV	license request	W	Messaging data	national	04	P
Commercial Vehicle Administration	Enforcement Agency	request for information on violators	W	Messaging data	national	04	P
Commercial Vehicle Administration	Enforcement Agency	violation notification	W	Messaging data	regional	04	P
Commercial Vehicle Administration	Financial Institution	payment request	W	Conversational data, Messaging data	national	04	F
Commercial Vehicle Administration	Fleet and Freight Management	activity reports	W	Messaging data	national	04	
Commercial Vehicle Administration	Fleet and Freight Management	compliance review report	W	Messaging data	national	04	
Commercial Vehicle Administration	Fleet and Freight Management	electronic credentials	W,U1t	Messaging data	national	04	
Commercial Vehicle Administration	Government Administrators	tax-credentials-fees request	W	Messaging data	national	04	
Commercial Vehicle Administration	Other CVAS	credentials and safety information request	W	Messaging data	national	04	
Commercial Vehicle Administration	Other CVAS	CVAS information exchange	W	Messaging data	national	04	
Commercial Vehicle Administration	Planning Subsystem	operational data	W	Messaging data	regional	04	
Commercial Vehicle Check	Commercial Vehicle Administration	citation data	W	Conversational data, Messaging data	regional	04	P
Commercial Vehicle Check	Commercial Vehicle Administration	credentials information request	W	Conversational data, Messaging data	regional	04	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Commercial Vehicle Check	Commercial Vehicle Administration	international border crossing data update	W	Conversational data, Messaging data	regional	04	
Commercial Vehicle Check	Commercial Vehicle Administration	roadside log update	W	Messaging data	regional	04	
Commercial Vehicle Check	Commercial Vehicle Administration	safety information request	W	Conversational data, Messaging data	regional	04	
Commercial Vehicle Check	Commercial Vehicle Driver	CVO Pull in Message	H	Human Interface	regional	H	
Commercial Vehicle Check	Commercial Vehicle Subsystem	border clearance event record	U2	Conversational Data	national	01	T
Commercial Vehicle Check	Commercial Vehicle Subsystem	clearance event record	U2	Conversational data	national	01	T
Commercial Vehicle Check	Commercial Vehicle Subsystem	electronic clearance request	U2	Conversational data	national	01	T
Commercial Vehicle Check	Commercial Vehicle Subsystem	lock tag data request	U2	Conversational Data	national	01	T
Commercial Vehicle Check	Commercial Vehicle Subsystem	on-board safety request	U2	Conversational data	national	01	T
Commercial Vehicle Check	Commercial Vehicle Subsystem	pass/pull-in	U2	Conversational data	national	01	T,R
Commercial Vehicle Check	Commercial Vehicle Subsystem	safety inspection record	U2	Conversational Data	national	01	T
Commercial Vehicle Check	Commercial Vehicle Subsystem	screening request	U2	Conversational data	national	01	T
Commercial Vehicle Check	CVO Inspector	CVO inspector information	H	Human Interface	product	H	
Commercial Vehicle Driver	Commercial Vehicle Subsystem	CVO driver initialization	H	Human Interface	product	H	
Commercial Vehicle Manager	Fleet and Freight Management	fleet manager inquiry	H	Human Interface	product	H	
Commercial Vehicle Subsystem	Commercial Vehicle	lock tag data request	W	Conversational Data	product		
Commercial Vehicle Subsystem	Commercial Vehicle Check	electronic clearance data	U2	Conversational data	national	01	T
Commercial Vehicle Subsystem	Commercial Vehicle Check	lock tag data	U2	Conversational Data	national	01	T
Commercial Vehicle Subsystem	Commercial Vehicle Check	on board safety data	U2	Conversational data	national	01	T
Commercial Vehicle Subsystem	Commercial Vehicle Check	screening data	U2	Conversational data	national	01	T,R
Commercial Vehicle Subsystem	Commercial Vehicle Driver	alerts, Messages	H	Human Interface	product	H	
Commercial Vehicle Subsystem	Commercial Vehicle Driver	CVO Pull in Message	H	Messaging data	product	H	
Commercial Vehicle Subsystem	Commercial Vehicle Driver	log information	H	Human Interface	product	H	
Commercial Vehicle Subsystem	Fleet and Freight Management	driver and vehicle information	U1t	Messaging data, location data	none	P	
Commercial Vehicle Subsystem	Fleet and Freight Management	on board vehicle data	U1t,U2	Messaging data	none	P	
Commercial Vehicle Subsystem	Vehicle	commercial vehicle data	W	Messaging data	product	I	
Construction and Maintenance	Traffic Management	equipment maintenance status	W	Messaging data	product		
Construction and Maintenance	Traffic Management	maintenance resource response	W	Conversational data, Messaging data	product		
Construction and Maintenance	Traffic Management	work zone status	W	Messaging data	product		
CVO Information Requestor	Commercial Vehicle Administration	credentials and safety information request	W	Messaging data	national	04	
CVO Inspector	Commercial Vehicle Check	CVC override mode	H	Conversational data, Messaging data	product	H	T
CVO Inspector	Commercial Vehicle Check	CVO inspector input	H	Human Interface	product	H	
DMV	Commercial Vehicle Administration	registration	W	Messaging data	national	04	P
DMV	Parking Management	registration	W	Messaging data	national		
DMV	Toll Administration	registration	W	Messaging data	national		P
DMV	Traffic Management	registration	W	Messaging data	national	06	P
Driver	Vehicle	driver inputs	H	Human Interface	product	H	
Driver	Vehicle	request for service	H	Human Interface	product	H	
Emergency Management	Emergency System Operator	emergency operations status	H	Human Interface	product	H	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Emergency Management	Emergency Telecommunications System	incident notification response	W	Conversational data conversational speec	regional	09	
Emergency Management	Emergency Vehicle Subsystem	emergency dispatch requests	U1t	Conversational speech, Messaging data	regional		E
Emergency Management	Emergency Vehicle Subsystem	incident command information	U1t	Conversational speech, Messaging data	regional		E
Emergency Management	Emergency Vehicle Subsystem	suggested route	U1t	Conversational speech, Messaging data	regional		E
Emergency Management	Fleet and Freight Management	Hazmat information request	W	Conversational data, Messaging data	national	09	E
Emergency Management	Information Service Provider	emergency vehicle route request	W	Conversational speech, Messaging data	regional	09	E
Emergency Management	Information Service Provider	incident information	W	Conversational speech, Messaging data	regional	09	
Emergency Management	Map Update Provider	map update request	W	Messaging data	national	02	
Emergency Management	Media	incident information for media	W	Messaging data	product		
Emergency Management	Other EM	incident report	W	Conversational data, Messaging data	regional	09	E
Emergency Management	Other EM	incident response coordination	W	Conversational data, Messaging data	regional	09	E
Emergency Management	Personal Information Access	emergency acknowledge	W,U1t	Conversational data, Messaging data	national	05	
Emergency Management	Planning Subsystem	operational data	W	Conversational data, Messaging data	regional	09	
Emergency Management	Remote Traveler Support	emergency acknowledge	W,U1t	Conversational speech, Messaging data	national	05	
Emergency Management	Traffic Management	emergency traffic control request	W	Conversational data, conversational speec	regional	08,09	E
Emergency Management	Traffic Management	incident information	W	Conversational data, Messaging data	regional	09	
Emergency Management	Traffic Management	incident response status	W	Conversational data, Messaging data	regional	09	
Emergency Management	Traffic Management	remote surveillance control	W	Conversational data, Messaging data	regional	09	
Emergency Management	Traffic Management	resource request	W	Conversational data, Messaging data	regional	09	
Emergency Management	Transit Management	transit emergency coordination data	W	Conversational data, Messaging data	regional	09,05	E
Emergency Management	Vehicle	emergency acknowledge	U1t	Conversational data conversational speec	national	05	
Emergency Personnel	Emergency Vehicle Subsystem	emergency personnel inputs	H	Human Interface	product	H	E
Emergency System Operator	Emergency Management	emergency operations request	H	Human Interface	product	H	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Emergency Telecommunications System	Emergency Management	incident notification	W	Conversational data, Conversational speech, Location data	regional	09	
Emergency Vehicle Subsystem	Emergency Management	emergency dispatch response	U1t	Conversational speech, Messaging data	regional		E
Emergency Vehicle Subsystem	Emergency Management	emergency vehicle tracking data	U1t	Conversational speech, Messaging data	regional		E
Emergency Vehicle Subsystem	Emergency Management	incident command request	U1t	Conversational speech, Messaging data	regional		E
Emergency Vehicle Subsystem	Emergency Management	incident status	U1t	Conversational speech, Messaging data	regional		E
Emergency Vehicle Subsystem	Emergency Personnel	dispatch information	H	Human Interface	product	H	E
Emergency Vehicle Subsystem	Emergency Personnel	incident command information presentation	H	Human Interface	product	H	E
Emergency Vehicle Subsystem	Roadway Subsystem	local signal preemption request	U2	Conversational data	regional	08,01	T,E
Emissions Management	Map Update Provider	map update request	W	Messaging data	national	02	
Emissions Management	Planning Subsystem	operational data	W	Messaging data	regional		
Emissions Management	Roadway Subsystem	vehicle pollution criteria	W	Messaging data	product	07	
Emissions Management	Traffic Management	widearea statistical pollution information	W	Messaging data	product	07	
Emissions Management	Traffic Operations Personnel	pollution data display	H	Human Interface	product	H	
Enforcement Agency	Commercial Vehicle Administration	information on violators	W	Messaging	regional	04	P
Environment	Emissions Management	pollution data	P	Physical Interface	NA		
Environment	Roadway Subsystem	pollution data	P	Physical Interface	NA		
Event Promoters	Traffic Management	event plans	W	Messaging data, Multicast data	regional	06	
Financial Institution	Commercial Vehicle Administration	transaction status	W	Conversational data, Messaging data	national	04	F
Financial Institution	Information Service Provider	transaction status	W	Conversational data, Messaging data	national	E	F
Financial Institution	Parking Management	transaction status	W	Conversational data, Messaging data	national	E	F
Financial Institution	Toll Administration	transaction status	W	Conversational data, Messaging data	national	E	F
Financial Institution	Transit Management	transaction status	W	Conversational data, Messaging data	national	E	F
Fleet and Freight Management	Commercial Vehicle Administration	credential application	W	Conversational data, Messaging data	national	04	
Fleet and Freight Management	Commercial Vehicle Administration	information request	W	Conversational data, Messaging data	national	04	
Fleet and Freight Management	Commercial Vehicle Administration	tax filing, audit data	w	Messaging data	national	04	P
Fleet and Freight Management	Commercial Vehicle Manager	fleet status	H	Human Interface	product	H	
Fleet and Freight Management	Commercial Vehicle Subsystem	fleet to driver update	U1t	Messaging data	none	P	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Fleet and Freight Management	Emergency Management	Hazmat information	W	Conversational data, Messaging data	national	09,05	
Fleet and Freight Management	Information Service Provider	route request	W	Conversational data, Messaging data	none	10	P
Fleet and Freight Management	Intermodal Freight Depot	intermod CVO coord	W	Messaging data	national	04	
Fleet and Freight Management	Intermodal Freight Shipper	intermod CVO coord	W	Messaging data	regional	04	
Fleet and Freight Management	Payment Instrument	request for payment	S	Conversational Data	national		
Government Administrators	Commercial Vehicle Administration	regulations	W	Messaging data, Multicast data	national	04	
Information Service Provider	Emergency Management	emergency vehicle route	W	Conversational speech, Messaging data	regional	09	
Information Service Provider	Emergency Management	incident information request	W	Conversational speech, Messaging data	regional	09	
Information Service Provider	Financial Institution	payment request	W	Conversational data, Messaging data	national	E	
Information Service Provider	Fleet and Freight Management	route plan	W	Messaging data	none	10	P
Information Service Provider	Intermodal Transportation Service Provider	intermodal information	W	Messaging data	regional	10	
Information Service Provider	ISP Operator	ISP operating parameters	H	Human Interface	product	H	
Information Service Provider	Map Update Provider	map update request	W	Messaging data	national	02	
Information Service Provider	Media	traveler information for media	W	Messaging data	product	10	
Information Service Provider	Other ISP	ISP coordination	W	Messaging data	national	10	
Information Service Provider	Parking Management	parking lot data request	W	Messaging data	regional	10	P
Information Service Provider	Parking Management	parking reservations request	W	Messaging data	regional	10	P
Information Service Provider	Personal Information Access	broadcast information	W,U1b	Messaging data,Broadcast data, Multicast	national	03	
Information Service Provider	Personal Information Access	traveler information	W,U1t	Broadcast data,Multicast data	national	03	P
Information Service Provider	Personal Information Access	trip plan	W,U1t	Conversational data, Messaging data	national	03	P
Information Service Provider	Planning Subsystem	aggregate travel data	W	Messaging data	regional	10	
Information Service Provider	Remote Traveler Support	broadcast information	W,U1b	Messaging data,Broadcast data, Multicast	product	10	
Information Service Provider	Remote Traveler Support	traveler information	W,U1t	Broadcast data,Multicast data	product	10	P
Information Service Provider	Remote Traveler Support	trip plan	W	Conversational Data	product	10	P
Information Service Provider	Toll Administration	toll data request	W	Messaging data	regional	10	
Information Service Provider	Traffic Management	fare and price information	W	Messaging data	regional	06	
Information Service Provider	Traffic Management	logged special vehicle route	W	Conversational data, Messaging data	regional	06	P
Information Service Provider	Traffic Management	request for traffic information	W	Messaging data	regional	06	
Information Service Provider	Traffic Management	road network use	W	Messaging data	regional	06	
Information Service Provider	Transit Management	demand responsive transit request	W	Messaging data	regional	10	P

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Information Service Provider	Transit Management	selected routes	W	Conversational data, Messaging data	regional	10	P
Information Service Provider	Transit Management	transit information request	W	Messaging data	regional	10	
Information Service Provider	Vehicle	broadcast information	U1b	Messaging data,Broadcast data, Multicast	national	03	
Information Service Provider	Vehicle	traveler information	U1t,U1b	Messaging data,Broadcast data, Multicast	national	03	P
Information Service Provider	Vehicle	trip plan	U1t	Conversational data, Messaging data	national	03	P
Information Service Provider	Yellow Pages Service Providers	provider profile confirm	W	Messaging data	national	10	P
Information Service Provider	Yellow Pages Service Providers	travel service request	W	Messaging data	national	10	P
Intermodal Freight Depot	Fleet and Freight Management	intermod CVO coord	W	Messaging data	national	04	
Intermodal Freight Shipper	Fleet and Freight Management	intermod CVO coord	W	Messaging data	regional	04	
Intermodal Transportation Service Provider	Information Service Provider	intermodal information	W	Messaging data	regional	10	
Intermodal Transportation Service Provider	Transit Management	intermodal information	W	Messaging data	regional		
ISP Operator	Information Service Provider	ISP operating parameter updates	H	Human Interface	product	H	
Location Data Source	Personal Information Access	position fix	L	Broadcast Data	product	02	
Location Data Source	Vehicle	position fix	L	Broadcast Data	product	02	
Map Update Provider	Emergency Management	map updates	W	Messaging data, Multicast data	national	02	
Map Update Provider	Emissions Management	map updates	W	Messaging data	national	02	
Map Update Provider	Information Service Provider	map updates	W	Messaging data, Multicast data	national	02	
Map Update Provider	Personal Information Access	map updates	W,U1t	Messaging data, Multicast data	national	02	
Map Update Provider	Planning Subsystem	map updates	W	Messaging data,Broadcast data, Multicast	national	02	
Map Update Provider	Remote Traveler Support	map updates	W	Messaging data	national	02	
Map Update Provider	Traffic Management	map updates	W	Messaging data, Multicast data	national	02	
Map Update Provider	Transit Management	map updates	W	Messaging data, Multicast data	national	02	
Map Update Provider	Vehicle	map updates	U1t	Messaging data, Multicast data	national	02	
Media	Emergency Management	media information request	W	Conversational Data, Messaging data	product		
Media	Information Service Provider	external reports	W	Messaging data, Multicast data	product	10	
Media	Information Service Provider	media information request	W	Conversational data, Messaging data	product	10	
Media	Traffic Management	external reports	W	Messaging data, Multicast data	product	06	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Media	Traffic Management	media information request	W	Conversational Data, Messaging data	product	06	
Media	Transit Management	media information request	W	Conversational Data, Messaging data	product		
Multimodal Crossings	Roadway Subsystem	multimodal crossing status	W	Conversational data, Messaging data	national		R
Other CVAS	Commercial Vehicle Administration	credentials and safety information response	W	Messaging data	national	04	
Other CVAS	Commercial Vehicle Administration	CVAS information exchange	W	Messaging data	national	04	
Other EM	Emergency Management	incident report	W	Conversational data, Messaging data	regional	09	E
Other EM	Emergency Management	incident response coordination	W	Conversational data, Messaging data	regional	09	E
Other ISP	Information Service Provider	ISP coordination	W	Messaging data	national	10	
Other TM	Traffic Management	traffic control coordination	W	Messaging data	regional	06	
Other TM	Traffic Management	traffic information coordination	W	Messaging data	regional	06	
Other TRM	Transit Management	TRMS coord	W	Messaging data	regional		
Other Vehicle	Vehicle	vehicle to vehicle coordination	U3	Conversational data	national	A	T,R
Parking Management	DMV	license request	W	Messaging data	national		
Parking Management	Driver	transaction status	H	Human Interface	product	H	
Parking Management	Enforcement Agency	violation notification	W	Messaging data	regional		
Parking Management	Financial Institution	payment request	W	Conversational data, Messaging data	national	E	
Parking Management	Information Service Provider	parking availability	W	Messaging data	regional	10	
Parking Management	Information Service Provider	parking lot reservation confirmation	W	Messaging data	regional	10	
Parking Management	Parking Operator	parking status	W	Messaging data	product		
Parking Management	Parking Service Provider	parking availability	W	Messaging data	product		
Parking Management	Planning Subsystem	operational data	W	Messaging data	regional		
Parking Management	Traffic Management	demand management response	W	Messaging data	regional	06	
Parking Management	Traffic Management	parking availability	W	Messaging data	regional	06	
Parking Management	Transit Management	transit parking coordination	W	Messaging data	regional		
Parking Management	Vehicle	request tag data	U2	Conversational data	national	01	T,R
Parking Management	Vehicle	tag update	U2	Conversational data	national	01	T,R
Parking Operator	Parking Management	parking instructions	H	Human Interface	product	H	
Parking Service Provider	Parking Management	request for performance data	W	Messaging data	product		
Payment Instrument	Fleet and Freight Management	payment	S	Conversational Data	national		
Payment Instrument	Personal Information Access	payment	S	Conversational Data	national		
Payment Instrument	Remote Traveler Support	Payment	S	Conversational Data	national		F
Payment Instrument	Transit Vehicle Subsystem	payment	S	Conversational Data	national		F
Payment Instrument	Vehicle	payment	S	Conversational Data	national		F
Pedestrians	Roadway Subsystem	crossing call	H	Human Interface	national	H	
Personal Information Access	Emergency Management	emergency notification	U1t	Conversational data, Messaging data	national	05	E
Personal Information Access	Information Service Provider	traveler profile	W,U1t	Conversational data, Messaging data	national	03	P
Personal Information Access	Information Service Provider	traveler request	W,U1t	Conversational data, Messaging data	national	03	P

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Personal Information Access	Information Service Provider	trip confirmation	W,U1t	Conversational Data, Messaging data	national	03	P
Personal Information Access	Information Service Provider	trip request	W,U1t	Conversational Data, Messaging data	national	03	P
Personal Information Access	Information Service Provider	yellow pages request	W,U1t	Conversational Data, Messaging data	national	03	P
Personal Information Access	Map Update Provider	map update request	W,U1t	Messaging data	national	02	
Personal Information Access	Payment Instrument	request for payment	S	Conversational Data	national		
Personal Information Access	Transit Management	transit information user request	W,U1t	Messaging data	national		
Personal Information Access	Traveler	traveler interface updates	H	Human Interface	product	H	P
Planning Subsystem	Map Update Provider	map update request	W	Messaging data	national	02	
Planning Subsystem	Traffic Management	planning data	W	Messaging data	regional	06	
Planning Subsystem	Transportation Planners	planning data	W	Messaging data	regional		
Potential Obstacles	Vehicle	physical presence	P	Physical Interface	NA		
Rail Operations	Traffic Management	railroad advisories	W	Conversational data, Messaging data	national	12	
Rail Operations	Traffic Management	railroad schedules	W	Conversational data, Messaging data	national	12	
Remote Traveler Support	Emergency Management	emergency notification	W,U1t	Conversational speech, Messaging data, location data	national	05	E
Remote Traveler Support	Information Service Provider	traveler request	W	Messaging data	product	10	P
Remote Traveler Support	Information Service Provider	trip confirmation	W	Messaging data	product	10	P
Remote Traveler Support	Information Service Provider	trip request	W	Conversational Data	product	10	P
Remote Traveler Support	Information Service Provider	yellow pages request	W	Conversational data, Messaging Data	product	10	P
Remote Traveler Support	Map Update Provider	map update request	W	Messaging data	national	02	
Remote Traveler Support	Payment Instrument	request for payment	S	Conversational Data	national		F
Remote Traveler Support	Transit Management	emergency notification	W	Conversational data, Messaging Data	product	05,11	E
Remote Traveler Support	Transit Management	secure area surveillance data	W	Conversational data, Messaging Data	product	11	E,T
Remote Traveler Support	Transit Management	transit fare payment requests	W	Conversational data, Messaging data	product	11	F
Remote Traveler Support	Transit Management	transit information user request	W	Messaging data	product	11	P
Remote Traveler Support	Transit User	transit user fare status	H	Human Interface	product	H	
Remote Traveler Support	Transit User	transit user outputs	H	Human Interface	product	H	
Remote Traveler Support	Traveler	traveler interface updates	H	Human Interface	product	H	P
Roadway	Vehicle	roadway characteristics	P	Physical Interface	NA		
Roadway Environment	Roadway Subsystem	weather conditions	P	Physical Interface	NA		
Roadway Environment	Vehicle	weather conditions	P	Physical Interface	NA		
Roadway Subsystem	Basic Vehicle	broadcast advisories	U1b	Conversational speech	national		
Roadway Subsystem	Driver	driver information	H	Human Interface	national	12	T, R
Roadway Subsystem	Emissions Management	pollution data	W	Messaging data	product	07	
Roadway Subsystem	Multimodal Crossings	highway control status	W	Conversational data, Messaging data	national		R
Roadway Subsystem	Pedestrians	crossing permission	H	Human Interface	national	H	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Roadway Subsystem	Traffic Management	AHS status	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	emissions data	W	Messaging data	product		
Roadway Subsystem	Traffic Management	environmental conditions	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	fault reports	W	Conversational data, Messaging data	product	07	
Roadway Subsystem	Traffic Management	freeway control status	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	hov data	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	hri status	W	Conversational data, Messaging data	product	12	R, S
Roadway Subsystem	Traffic Management	incident data	W	Messaging data	product	07	T, R
Roadway Subsystem	Traffic Management	intersection blockage notification	W	Messaging data	product	12	T, R
Roadway Subsystem	Traffic Management	request for right-of-way	W	Conversational data, Messaging data	product	08,07	R
Roadway Subsystem	Traffic Management	reversible lane status	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	roadway information system status	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	signal control status	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	traffic flow	W	Messaging data	product	07	
Roadway Subsystem	Traffic Management	traffic images	W	Messaging data	product	07	T
Roadway Subsystem	Traffic Management	vehicle probe data	W	Messaging data	product	07	
Roadway Subsystem	Vehicle	AHS control data	U2	Messaging data	national	01	T, R
Roadway Subsystem	Vehicle	intersection status	U2	Messaging data	national	01	T, R
Roadway Subsystem	Vehicle	request tag data	U2	Messaging data	national	01	T
Roadway Subsystem	Vehicle	vehicle signage data	U2	Messaging data	national	01	T
Roadway Subsystem	Wayside Equipment	hri status	W	Conversational data, Messaging data	product	12	R, S
Roadway Subsystem	Wayside Equipment	intersection blockage notification	W	Messaging data	product	12	T, R
Secure Area Environment	Remote Traveler Support	secure area characteristics	P	Physical Interface	NA		
Toll Administration	DMV	license request	W	Messaging data	national		
Toll Administration	Enforcement Agency	violation notification	W	Messaging data	regional		
Toll Administration	Financial Institution	payment request	W	Messaging data	national	E	
Toll Administration	Information Service Provider	probe data	W	Messaging data	regional	06	
Toll Administration	Information Service Provider	toll data	W	Messaging data	regional	10	
Toll Administration	Planning Subsystem	operational data	W	Messaging data	regional		
Toll Administration	Toll Collection	toll instructions	W	Messaging data	regional		
Toll Administration	Toll Operator	toll transaction reports	H	Human Interface	product	H	
Toll Administration	Toll Service Provider	toll revenues and summary reports	W	Messaging data	product		
Toll Administration	Traffic Management	demand management response	W	Messaging data	regional	06	
Toll Administration	Traffic Management	probe data	W	Messaging data	regional	06	
Toll Collection	Driver	transaction status	H	Human Interface	national	H	
Toll Collection	Toll Administration	toll transactions	W	Messaging data	regional		
Toll Collection	Vehicle	request tag data	U2	Conversational data	national	01	T,R
Toll Collection	Vehicle	tag update	U2	Conversational Data	national	01	T,R
Toll Operator	Toll Administration	toll operator requests	H	Human Interface	product	H	
Toll Service Provider	Toll Administration	toll fees	H	Human Interface	product	H	
Traffic	Roadway Subsystem	traffic characteristics	P	Physical Interface	NA		
Traffic Management	Construction and Maintenance	closure coordination	W	Conversational data, Messaging data	product	06	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Traffic Management	Construction and Maintenance	maintenance resource request	W	Conversational data, Messaging data	product	06	
Traffic Management	Construction and Maintenance	traffic equipment status	W	Conversational data, Messaging data	product	06	
Traffic Management	DMV	license request	W	Messaging data	national	06	
Traffic Management	Emergency Management	current network conditions	W	Messaging data	regional		
Traffic Management	Emergency Management	emergency traffic control response	W	Conversational data, conversational speech	regional		
Traffic Management	Emergency Management	incident information request	W	Messaging data	regional	09	E
Traffic Management	Emergency Management	incident notification	W	Messaging data	regional	09	E
Traffic Management	Emergency Management	resource deployment status	W	Messaging data	regional	09	
Traffic Management	Emissions Management	pollution state data request	W	Messaging data	product	07	
Traffic Management	Enforcement Agency	violation notification	W	Messaging data	regional	06	
Traffic Management	Event Promoters	event confirmation	W	Messaging data	regional	06	
Traffic Management	Information Service Provider	request fare and price information	W	Messaging data	regional	06	
Traffic Management	Information Service Provider	traffic information	W	Messaging data	regional	06	
Traffic Management	Map Update Provider	map update request	W	Messaging data	national	02	
Traffic Management	Media	traffic information for media	W	Messaging data	product	06	
Traffic Management	Other TM	traffic control coordination	W	Messaging data	regional	06	
Traffic Management	Other TM	traffic information coordination	W	Messaging data	regional	06	
Traffic Management	Parking Management	demand management request	W	Messaging data	regional	06	
Traffic Management	Parking Management	parking instructions	W	Messaging data	regional	06	
Traffic Management	Planning Subsystem	operational data	W	Messaging data	regional	06	
Traffic Management	Rail Operations	hri advisories	W	Conversational Data, Messaging data	national	12	T,R
Traffic Management	Roadway Subsystem	AHS control information	W	Messaging data	product	07	
Traffic Management	Roadway Subsystem	freeway control data	W	Messaging data	product	07	T,S
Traffic Management	Roadway Subsystem	hri control data	W	Messaging data	product	12	R, S
Traffic Management	Roadway Subsystem	hri request	W	Conversational data, Messaging data	product	12	R, S
Traffic Management	Roadway Subsystem	roadway information system data	W	Messaging data	product	07	S
Traffic Management	Roadway Subsystem	sensor and surveillance control	W	Messaging data	product	07	T,S
Traffic Management	Roadway Subsystem	signal control data	W	Messaging data	product	07	T,S
Traffic Management	Toll Administration	demand management request	W	Messaging data	regional	06	
Traffic Management	Traffic Operations Personnel	traffic operations data	H	Human Interface	product	H	
Traffic Management	Transit Management	demand management request	W	Messaging data	regional	06	
Traffic Management	Transit Management	request transit information	W	Conversational data, Messaging data	regional	06	
Traffic Management	Transit Management	traffic control priority status	W	Conversational data, Messaging data	regional	06	
Traffic Management	Transit Management	traffic information for transit	W	Messaging data	regional	06	
Traffic Operations Personnel	Emissions Management	pollution data parameters	H	Human Interface	product	H	
Traffic Operations Personnel	Traffic Management	traffic operations requests	H	Human Interface	product	H	
Transit Driver	Transit Management	transit driver availability	H	Human Interface	product	H	
Transit Driver	Transit Vehicle Subsystem	transit driver inputs	H	Human Interface	product	H	
Transit Fleet Manager	Transit Management	transit fleet manager inputs	H	Human Interface	product	H	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Transit Maintenance Personnel	Transit Management	maintenance status	H	Human Interface	product	H	
Transit Management	Emergency Management	transit emergency data	W	Conversational data, Messaging data	regional	05,09	E
Transit Management	Enforcement Agency	violation notification	W	Messaging data	regional		
Transit Management	Financial Institution	payment request	W	Conversational data, Messaging data	national	E	
Transit Management	Information Service Provider	demand responsive transit plan	W	Conversational data, Messaging data	regional	10	P
Transit Management	Information Service Provider	transit and fare schedules	W	Messaging data	regional	10	
Transit Management	Information Service Provider	transit incident information	W	Messaging data	regional	10	
Transit Management	Information Service Provider	transit request confirmation	W	Messaging data	regional	10	P
Transit Management	Intermodal Transportation Service Provider	intermodal information	W	Messaging data	regional		
Transit Management	Map Update Provider	map update request	W	Messaging data	national	02	
Transit Management	Media	transit incidents for media	W	Messaging data	product		
Transit Management	Media	transit information for media	W	Messaging data	product		
Transit Management	Other TRM	TRMS coord	W	Messaging data	regional		
Transit Management	Parking Management	transit parking lot response	W	Messaging data,Broadcast data, Multicast	regional		
Transit Management	Personal Information Access	personal transit information	W,U1t	Messaging data	national		
Transit Management	Planning Subsystem	operational data	W	Messaging data	regional		
Transit Management	Remote Traveler Support	emergency acknowledge	W	Conversational data, Messaging data	product	05,11	
Transit Management	Remote Traveler Support	secure area monitoring support	W	Conversational data, Messaging data	product	11	E,T
Transit Management	Remote Traveler Support	transit fare payment responses	W	Conversational data, Messaging data	product	11	F
Transit Management	Remote Traveler Support	transit traveler information	W	Messaging data, Multicast data	product	11	
Transit Management	Traffic Management	demand management response	W	Messaging data	regional	06	
Transit Management	Traffic Management	traffic control priority request	W	Messaging data	regional	08,06	
Transit Management	Traffic Management	transit system data	W	Messaging data	regional	06	
Transit Management	Transit Driver	route assignment	H	Human Interface	product	H	
Transit Management	Transit Fleet Manager	transit operations planning data	H	Human Interface	product	H	
Transit Management	Transit Maintenance Personnel	transit work schedule	H	Human Interface	product	H	
Transit Management	Transit System Operators	transit operator display	H	Human Interface	product	H	
Transit Management	Transit Vehicle Subsystem	bad tag list	U1t	Messaging data	product	11	
Transit Management	Transit Vehicle Subsystem	driver instructions	U1t	Messaging data	product	11	
Transit Management	Transit Vehicle Subsystem	emergency acknowledge	U1t	Conversational data, Messaging data	product	05,11	
Transit Management	Transit Vehicle Subsystem	fare management information	U1t	Messaging data	product	11	
Transit Management	Transit Vehicle Subsystem	request for vehicle measures	U1t,U2	Messaging data	product	11	
Transit Management	Transit Vehicle Subsystem	transit schedule information	U1t	Messaging data	product	11	
Transit Management	Transit Vehicle Subsystem	traveler information	U1t	Messaging data	product	11	P
Transit System Operators	Transit Management	transit operator management data	H	Human Interface	product	H	
Transit User	Remote Traveler Support	transit user inputs	H	Human Interface	product	H	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Transit User	Transit Vehicle Subsystem	emergency notification	H	Human Interface	product	H	
Transit User	Transit Vehicle Subsystem	transit user inputs	H	Human Interface	product	H	
Transit Vehicle	Transit Vehicle Subsystem	vehicle measures	W	Broadcast data	product	I	R
Transit Vehicle Subsystem	Payment Instrument	request for payment	S	Conversational Data	national		F
Transit Vehicle Subsystem	Roadway Subsystem	local signal priority request	U2	Conversational data	regional	01,08	T
Transit Vehicle Subsystem	Transit Driver	transit driver display	H	Human Interface	product	H	
Transit Vehicle Subsystem	Transit Management	emergency notification	U1t	Messaging data	product	11,05	E
Transit Vehicle Subsystem	Transit Management	fare and payment status	U1t,U2	Conversational data, Messaging data	product	11	F,T
Transit Vehicle Subsystem	Transit Management	request for bad tag list	U1t,U2	Conversational data, Messaging data	product	11	F,T
Transit Vehicle Subsystem	Transit Management	transit vehicle conditions	U1t,U2	Messaging data	product	11	
Transit Vehicle Subsystem	Transit Management	transit vehicle location data	U1t,U2	Conversational data, Messaging data, location data	product	11	
Transit Vehicle Subsystem	Transit Management	transit vehicle passenger and use data	U1t,U2	Conversational data, Messaging data	product	11	
Transit Vehicle Subsystem	Transit Management	transit vehicle schedule performance	U1t,U2	Conversational data, Messaging data	product	11	
Transit Vehicle Subsystem	Transit Management	traveler request	U1t	Conversational data, Messaging data	product	11	P
Transit Vehicle Subsystem	Transit User	transit user fare status	H	Human Interface	product	H	
Transit Vehicle Subsystem	Transit User	transit user outputs	H	Human Interface	product	H	
Transit Vehicle Subsystem	Vehicle	traveler advisory request	W	Messaging data	product		
Transportation Planners	Planning Subsystem	planning data	W	Messaging data	regional		
Traveler	Personal Information Access	traveler request	H	Human Interface	product	H	
Traveler	Remote Traveler Support	traveler request	H	Human Interface	product	H	
Vehicle	Basic Vehicle	vehicle control	W	Conversational Data	product	I	R
Vehicle	Commercial Vehicle Subsystem	commercial vehicle data request	W	Messaging data	product	I	
Vehicle	Driver	driver updates	H	Human Interface	product	H	
Vehicle	Driver	transaction status	H	Human Interface	product	H	
Vehicle	Emergency Management	emergency notification	U1t	Conversational speech, Messaging data, location data	national	05	E
Vehicle	Emergency Vehicle Subsystem	vehicle location	W	Broadcast data	product		
Vehicle	Information Service Provider	traveler profile	U1t	Conversational data, Messaging data	national	03	P
Vehicle	Information Service Provider	traveler request	U1t	Conversational data, Messaging data	national	03	P
Vehicle	Information Service Provider	trip confirmation	U1t	Conversational data, Messaging data	national	03	P
Vehicle	Information Service Provider	trip request	U1t	Conversational data, Messaging data	national	03	P
Vehicle	Information Service Provider	vehicle probe data	U1t	Messaging data, location data	national	03	P
Vehicle	Information Service Provider	yellow pages request	U1t	Conversational data, Messaging data	national	03	P
Vehicle	Map Update Provider	map update request	U1t	Messaging data	national	02	

Table A.1 Physical Architecture Flows - Interconnects - Interoperability Requirements (continued)

Source	Destination	Architecture Flow	Interconnects*	Communication Service*	Interoperability*	Standards Package	Special Constraints*
Vehicle	Other Vehicle	vehicle to vehicle coordination	U3	Conversational data	national	A	T,R
Vehicle	Parking Management	tag data	U2	Conversational data	national	01	T,P
Vehicle	Payment Instrument	request for payment	S	Conversational Data	national		F
Vehicle	Roadway Subsystem	AHS vehicle data	U2	Conversational data	national	01	T,R
Vehicle	Roadway Subsystem	vehicle probe data	U2	Messaging Data	national	01	T,P
Vehicle	Toll Collection	tag data	U2	Conversational data	national	01	T,P
Vehicle	Transit Vehicle Subsystem	vehicle location	W	Broadcast data	product	I	
Vehicle Characteristics	Parking Management	vehicle characteristics	P	Physical Interface	NA		
Vehicle Characteristics	Roadway Subsystem	vehicle characteristics	P	Physical Interface	NA		
Vehicle Characteristics	Toll Collection	vehicle characteristics	P	Physical Interface	NA		
Wayside Equipment	Roadway Subsystem	arriving train information	W	Messaging data	product	12	T, R
Wayside Equipment	Roadway Subsystem	track status	W	Messaging data	product	12	T,R
Weather Service	Emergency Management	weather information	W	Messaging data,Broadcast data, Multicast	national		
Weather Service	Information Service Provider	weather information	W	Messaging data,Broadcast data, Multicast	national	10	
Weather Service	Traffic Management	weather information	W	Messaging data,Broadcast data, Multicast	national	06	
Weather Service	Transit Management	weather information	W	Messaging data,Broadcast data, Multicast	national		
Yellow Pages Service Providers	Information Service Provider	provider profile data	W	Messaging data	national	10	
Yellow Pages Service Providers	Information Service Provider	travel service info	W	Messaging data	national	10	

*** Interconnect Types as defined in Table A.2 (Communications Document Chapter 3)**
Communication Services Types in Table A.3 (Communications Document Chapter 3)
Interoperability Types in Table A.4 (Standards Development Plan Chapter 1)
Standards Packages in Table A.5 (Standards Requirements Document)
Special Constraints in Table A.6 (Physical Architecture Chapter 2.23)

Table A.2 Interconnect Types

Interconnect	Interconnect Name	Interconnect Description
H	Human Interface	Can be either a user interface to the system, an operator interface, or a driver.
L	Position Location	Interface between position location equipment and the source for indicating position location. This could be either information from a terrestrial source, GPS, FM subcarrier, Dead Reckoning etc
P	Physical Interface	This is an interface which senses some physical characteristic or causes some action that is not represented using standard communications technology (e.g. observing an obstacle)
S	Payment Instrument	This interface is between the card which is carried by the traveler and which contains the account number of stored value and the an object which accepts this information. The non-card interface could be a reader at a kiosk or in a vehicle. In the latter case, the reader in the vehicle forwards the information to the infrastructure.
U1t	Wide Area Wireless	Wide area 2-way communication capable of communication between a mobile traveler or vehicle and the infrastructure from any location.
U1b	Wide Area Broadcast	Wide area broadcast information in which the mobile traveler or vehicle can receive information from any location.
U2	Vehicle-to-Roadside	Short range vehicle to roadside (e.g. beacon). The interface contains information regarding which mobile entity is communicating.
U3	Vehicle-to-Vehicle	Primarily AHS type communications yet to be defined
W	Wireline	Wireline system interconnect which includes fixed to fixed communication capabilities. May include wide area wireless capabilities for transportable devices such as CMS, and may include short hop wireless connections to wireline subsystems from distributed assets such as signal and sensors. Includes normal telephone and public and private fiber-optic links.

Table A.3 Communications Services Types

Service Type	Description
Conversational	Implies the use of a two-way connection established before information exchange begins and terminated when exchange is completed
Messaging	Works more like electronic mail being exchanged between users without establishing a dedicated path between two sites.
Broadcast	Messages sent to all users
Multi-cast	Messages sent to only a subset of users

Table A.4 Interoperability Types

Interoperability	Description
National	Interfaces to the mobile subsystems (Vehicle Subsystems, Personal Information Access Subsystems) in the architecture support national interoperability since the same mobile subsystem should be able to roam the nation and use the local infrastructure to support ITS services. National interoperability is specified for all interfaces to mobile subsystems except where both the mobile subsystem and interfacing infrastructure are owned and operated by the same user. Examples of these include the Information Service Provider to Personal Information Access Subsystem, Toll Collection Subsystem to Vehicle Subsystem, and the Commercial Vehicle Subsystem to Commercial Vehicle Check Subsystem.
Regional	Interfaces connecting subsystems that may be operated by different agencies (interfaces that can span jurisdictional and/or regional boundaries) can be standardized to facilitate the sharing of information between agencies. National standards mitigate issues that may arise as boundaries change and new requirements for information sharing develop over time. Regional interoperability is specified where the underlying coordination issues are regional, rather than national, in scope. For instance, there is no real requirement for a Traffic Management Subsystem in California to be able to communicate and coordinate with a Traffic Management Subsystem in New York. Two different regional dialects for Traffic Management Subsystem communications could be implemented in the two geographically isolated subsystems, without significant impact to national interoperability goals. Examples of these include the Traffic Management Subsystem to Transit Management Subsystem, Traffic Management Subsystem to Information Service Provider, and Traffic Management Subsystem to Traffic Management Subsystem.
Product	Interfaces between subsystems that are operated and maintained by a single stakeholder (e.g. company or agency) do not require standardization to achieve national interoperability. The data formats and communications mechanisms that are used for these interfaces are largely transparent to the remainder of the architecture. In some cases, national standards are still very beneficial (and hence still attainable through the consensus standard process) since they may consolidate a market to achieve economy of scale efficiencies (e.g. Traffic Management Subsystem to Roadway Subsystem). Such standards may also support an optional level of interoperability by enabling various cooperative control options to be implemented based on regional preference.
None	In other cases, the sheer range of application-specific interfaces precludes efficient national standardization and no standard is suggested. For instance, a national standard is not recommended for the interface between the Fleet Management and Commercial Vehicle subsystems since the nature of the interface is so dependent on fleet type. From the National Architecture perspective, standardization for these interfaces is not suggested. Examples include the Fleet Management Subsystem to Commercial Vehicle Subsystem.

Table A.5 Standards Packages

Number	Requirement Package Name
01	Dedicate Short Range Communications (DSRC)
02	Digital Map Data Exchange and Location Referencing
03	Information Service Provider Wireless Interfaces
04	Inter-Center Data Exchange for Commercial Vehicle Operations
05	Personal and HAZMAT Maydays
06	Traffic Management Subsystems to Other Centers (Except EM)
07	Traffic Management Subsystems to Roadway Devices and Emissions Sensing/Management
08	Signal Priority for Transit and Emergency Vehicles
09	Emergency Management to Other Centers
10	Information Service Provider to Other Centers (except EM and TMS)
11	Transit Management to Transit Vehicle
12	Highway Rail Intersection
A	AHS Standards
E	Existing Standards
I	Internal and probably proprietary
P	Proprietary Standards
H	Human Interfaces

Table A.6 Special Constraints

Constraint Abbreviation	Constraint Name	Description
R	Reliability	Failure of the communication medium may result in severe accident. This communication channel may require redundant paths or extra attention paid to potential failure modes. For wireline cases, this may indicate alternate phone or other connections are required. For wireless cases (for AHS applications), special attention will be paid to the transmitters, receivers, and potential interference for these connections
F	Financial Security	Data contains financial information and must be protected accordingly. This data is specifically called out between the user's card and the infrastructure and between the infrastructure and financial institutions. Protections currently exist for the latter. Financial data transmitted over the air must be recognized as private data with an additional reliability requirement. Financial data may exist between other subsystems as part of normal messaging. It is assumed that such data will be treated with the same constraints as the interfaces specifically identified
P	Personal Privacy	Data contains personal information. Traveler requests and traveler location are private and should be protected. Subsystems aggregate these data and forward specific data with the traveler's permission.
E	Emergency Priority	Communication channel requires priority in emergencies. These data channels require that they be operational even when there is an emergency which might place other loads on the interface. A private communication channel or frequency may be required to satisfy the requirement.
T	Performance (Timing)	Timing is critical. Timing for most ITS communication services is based on the response to a request for data. Because of this, common communication media designed to handle voice data will likely support these requirements. The beacon interface has special requirements of identifying the vehicle as well as exchanging information before the vehicle gets out of range. This is more of a problem with vehicles traveling at speed. The architecture constrains such time critical access to data such that the data is available at the beacon site. This obviates the need for explicit specification of other timing information to support data transfer over a short range beacon.