

# **KAJIAN PELAN INDUK SISTEM PENGANGKUTAN PINTAR**

## **Development of ITS System Architecture for Malaysia**

### **Technical Note No. 2**

### **Volume I – Description and Data Flow Diagrams**

October 2006

# TABLE OF CONTENTS

## Logical Architecture Framework Volume I - Description and Data Flow Diagrams

1	LOGICAL ARCHITECTURE PROCESS MODEL.....	1
1.1	Introduction.....	1
1.2	Methodology.....	2
1.3	Document Organization.....	3
1.4	Functional Specifications.....	3
1.5	Data Flow Naming Convention.....	4
2	GRAPHICAL DATA FLOW MODEL.....	5
2.1	Introduction.....	5
2.2	Data Flow Diagrams.....	5

# 1 LOGICAL ARCHITECTURE PROCESS MODEL

## 1.1 Introduction

The Logical Architecture is based on a high level Computer Aided Systems Engineering (CASE) model of the functional requirements for the flow of data and control through the Intelligent Transportation System (ITS). Structured Data Flow Diagrams and Specifications are presented in this document to illustrate the decomposition of the User Services and User Sub-Services Requirements as well as specific strategic requirements to processes and information flow. The Logical Architecture CASE model further depicts the input (source) terminators and output (sink) terminators of the entire ITS and defines the information flow into and out of the system.

The Logical Architecture is first illustrated by showing the interactions between the processes with the external inputs (sources) and outputs (sinks) by the top level Data Flow Diagram (DFD 0). These bubbles are then decomposed in lower level of Data Flow Diagrams (DFDs). At the lowest level of this decomposition are the actual Process Specifications (P-Specs).

A structured English specification is composed for each P-Spec to define how the P-Spec output data flows are constructed from its input data flows. It is the P-Specs that represent the sources (and sinks) of data flows inside the ITS. Technically, a “data store” can also source and sink flows, however all stores in the Architecture are tightly coupled to a “store controlling” process, which can be considered logically to be the source and sink for data to/from the store.

## 1.2 Methodology

The Structure Analysis methodology was used to develop the Malaysian ITS Architecture. This approach was originally set forth by Tom DeMarco and it is based on functional decomposition of complex systems into single-function tasks and sub-tasks.

The main source of methodology adopted for the Logical Architecture has been based on the work of Hatley-Pirbhai (H-P). H-P is an extension of structure analysis which uses DFDs to show the data flow between the functional elements that make up the architecture. There are three types of functional elements: Terminators, Process Specifications (P-Specs) and Stores.

Terminators provide the sources and sinks for external information that flows to and from the Architecture. Terminators appear on the top level DFD. A departure from the H-P methodology but it is meant to enhance the reader's ability to understand the relationships among the top level processes and the external world. The P-Specs are provided as structured English descriptions that use "will" statements to describe the way in which they transform input information into desired outputs. Stores are internal place holders for data and may appear on any DFD.

On a DFD, anything other than stores and data flows is shown as a circular object or bubble. A bubble may be a P-Spec or may further decompose into a lower level DFD.

### **1.3 Document Organization**

This document is comprised of three volumes. This document, volume I, describes the Logical Architecture through DFDs. It provides a roadmap to finding information on the components of the Logical Architecture. Volume II of the Logical Architecture provides a complete listing of the P-Specs. Contained in each P-Spec are the associated Subsystems, Deployment Packages and data flows. Volume III provides the complete Data Dictionary model which contains the definitions of all the data flows.

### **1.4 Functional Specifications**

The Functional Specification is expressed in two ways: First, in graphical form as DFDs, and second, in more detailed textual form as P-Specs and data flow descriptions.

In general, the DFDs provide a convenient roadmap of the structure of the requirements and are shown by the Figures in section 2.2 of this document. They consist of lines showing data flows and “bubbles” that represent either a P-Spec, or a lower level DFD. The highest level DFD (DFD 0) shows the least detailed functionality needed to meet the User Service Requirements. Each of its nine bubbles is itself a DFD which provides a first level decomposition of the high level functionality. The bubbles in these nine DFDs will themselves decompose into further lower level DFDs, or P-Specs.

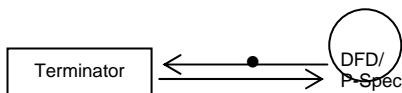
The P-Specs provide the essence of the requirements in that they describe how data that flows into the Logical Architecture is transformed either for use elsewhere, or for output from the Architecture. The P-Specs stand alone in that they completely and rigorously capture the functional requirements of the Architecture. As

noted above, a full textual description of the functionality provided by each P-Spec is provided separately in Volume II of this document.

The descriptions of the data flows are contained in the Data Dictionary Entries (DDEs), which like the DFDs and P-Specs, form a hierarchical structure. In this instance, the hierarchical structure enables one data flow to be comprised of several other data flows, each of which may decompose into other data flows until the primitive or lowest level data flows are reached. The description of each data flow is provided in Volume III of this document.

## 1.5 Data Flow Naming Convention

In order to identify the many data flows with terminators external to the ITS (i.e. those flows originate or terminate on the DFD 0), the following naming convention has been adopted:



To\_TerminatorName and From\_TerminatorName – data flows towards and from a terminator.

Because the messages flowing to and from terminators are generally compound messages that will split into individual data flows at lower levels, the following additional convention has been adopted:

- txx\_data\_name and fxx\_data\_name – individual data flows towards and from a terminator.

Note that these data flows always begin with a lowercase “t” or “f”, and that “xx” consists of one or more letter that indicate a unique abbreviation for the terminator name. The “data\_name” text string is a free form expression to describe the particular data flow. The actual

definition or any individual terminator data flow can be seen in the DDE in Volume III. The following example illustrates both naming conventions mentioned above.

The data flow from terminator Travellers on DFD 0 “From\_Traveller” comprises of the following individual data flows which originate from the terminator Traveller:

From\_Traveller comprises of:

- ft-extra\_trip\_data
- + ft\_guidance\_data
- + ft\_guidance\_map\_update\_request
- + ft-guidance\_request
- + ft-guidance\_route\_accepted
- + ft-remote\_emergency\_request
- + ft-personal\_emergency\_request
- + ft-personal\_extra\_trip\_data
- + ft-personal\_map\_display\_update\_request
- + ft-personal\_trip\_planning\_requests
- + ft-trip\_planning\_requests

## **2 GRAPHICAL DATA FLOW MODEL**

### **2.1 Introduction**

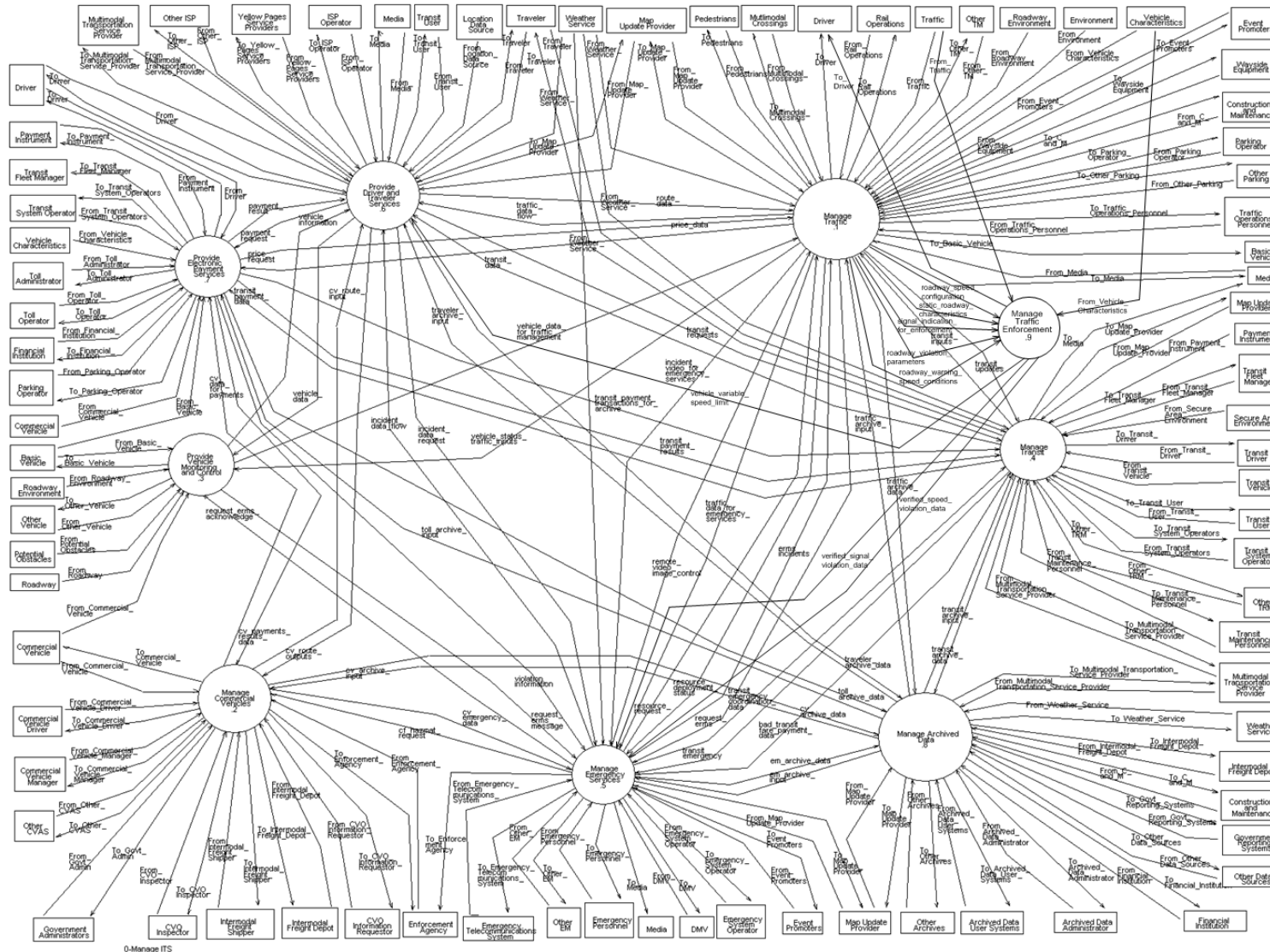
This section contains a complete set of the Data Flow Diagrams (DFDs) for the Malaysian ITS Logical Architecture. A full list of the DFDs and the associated P-Specs is presented in Appendix A.

### **2.2 Data Flow Diagrams**

# Development of ITS System Architecture for Malaysia

## Technical Note No. 2

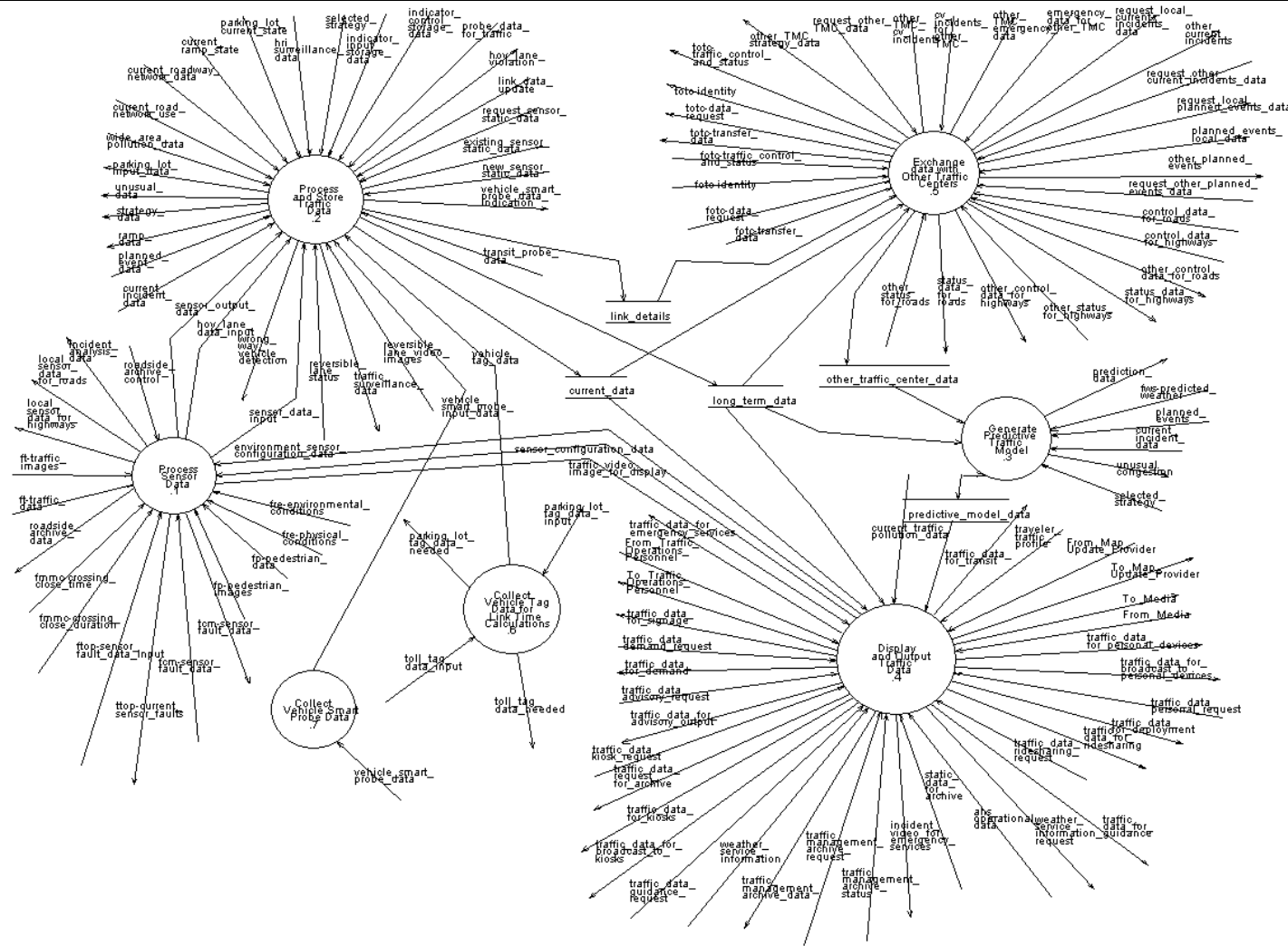
### Logical Architecture Framework - Volume I



# Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

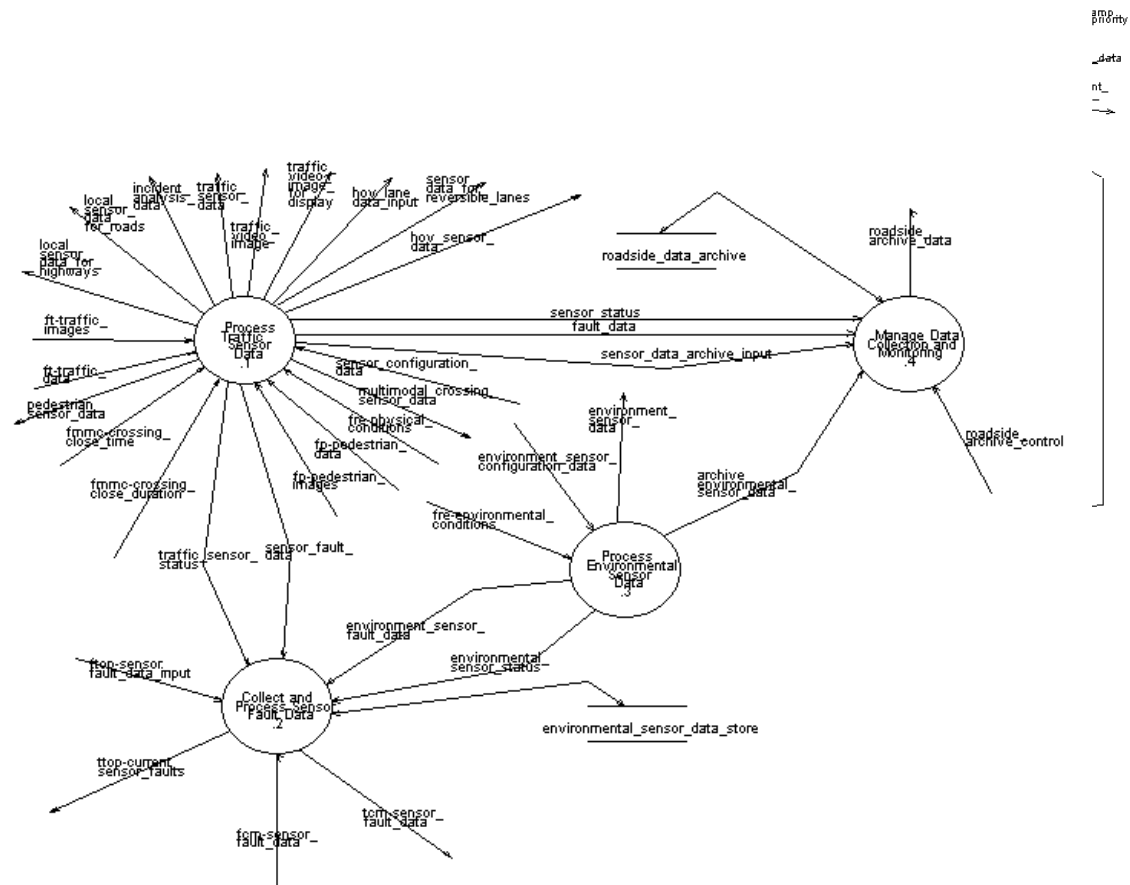


1.1-Provide Traffic Surveillance

## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

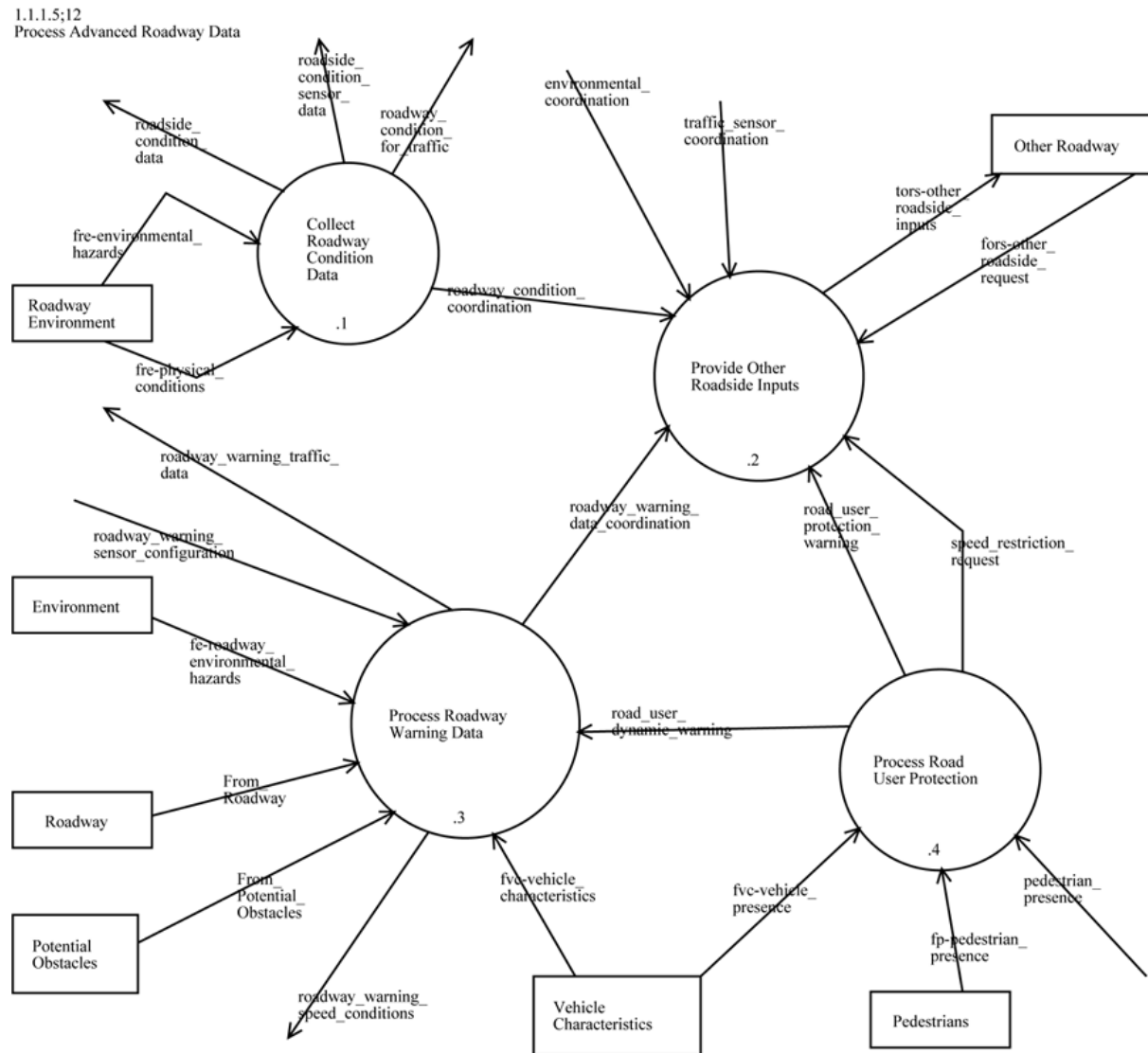


1.1.1-Process Sensor Data

Development of ITS System Architecture for Malaysia

Technical Note No. 2

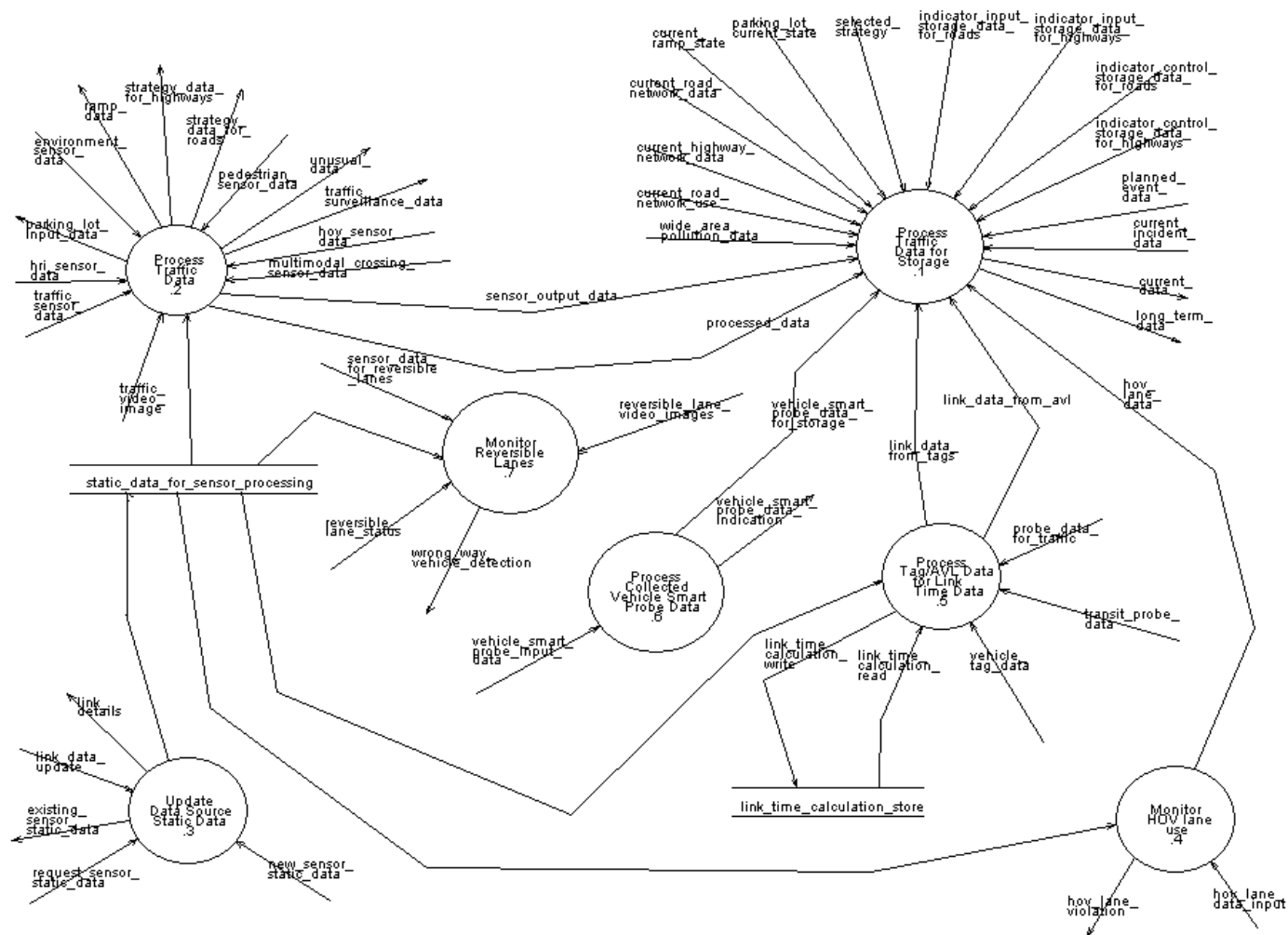
Logical Architecture Framework - Volume I



## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

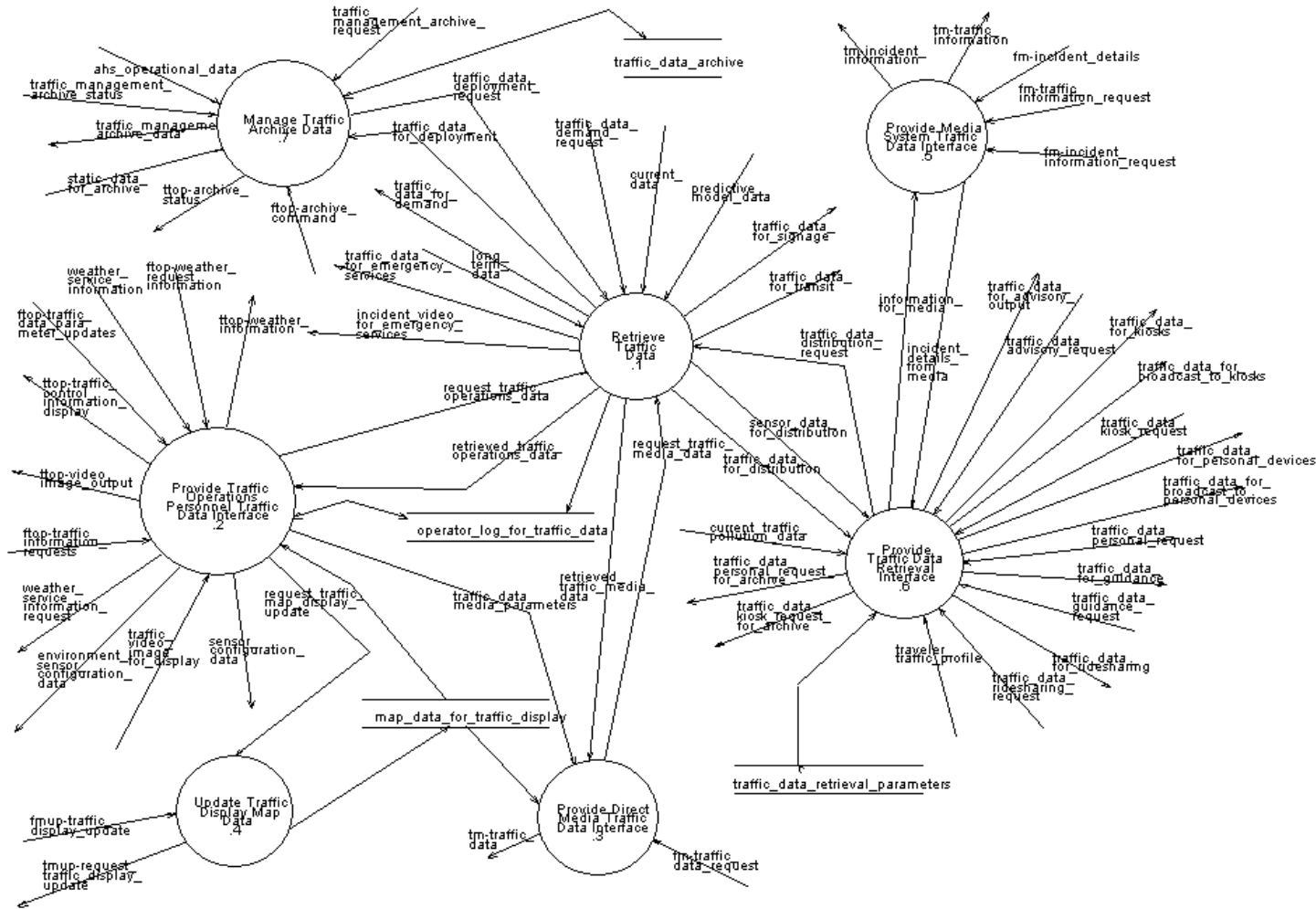


1.1.2-Process and Store Traffic Data

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

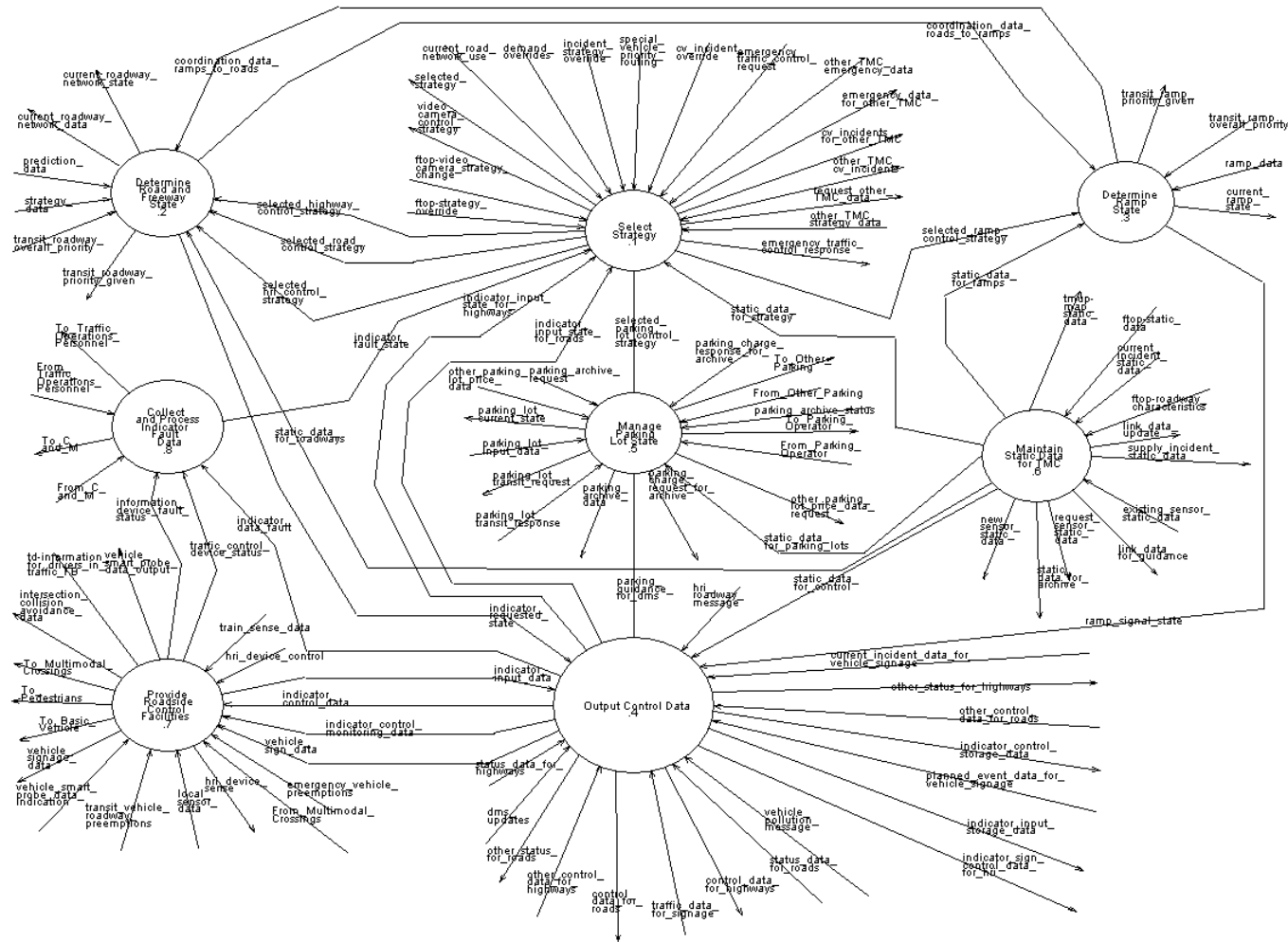


1.1.4-Display and Output Traffic Data

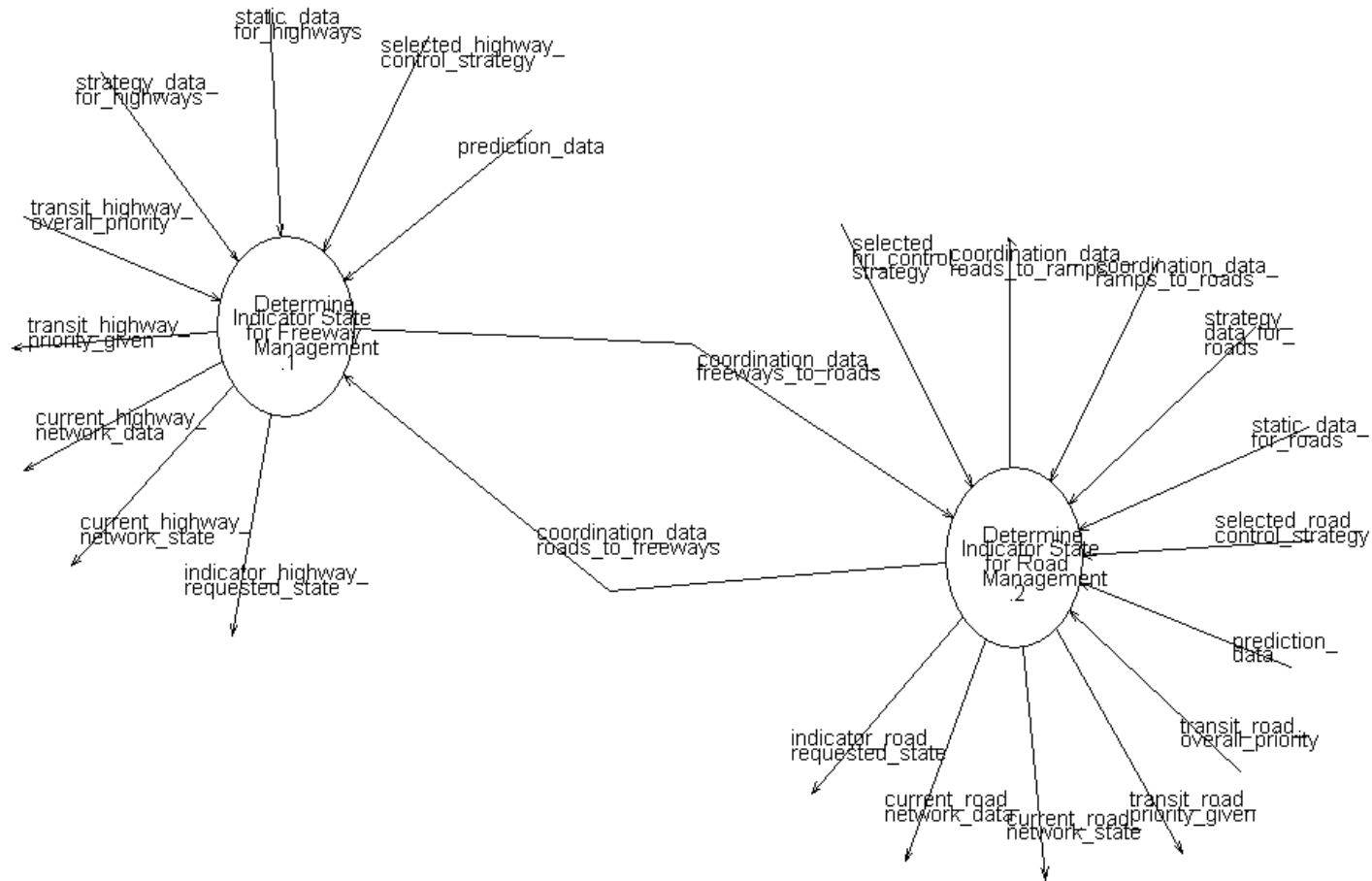
## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



1.2. Provide Device Control



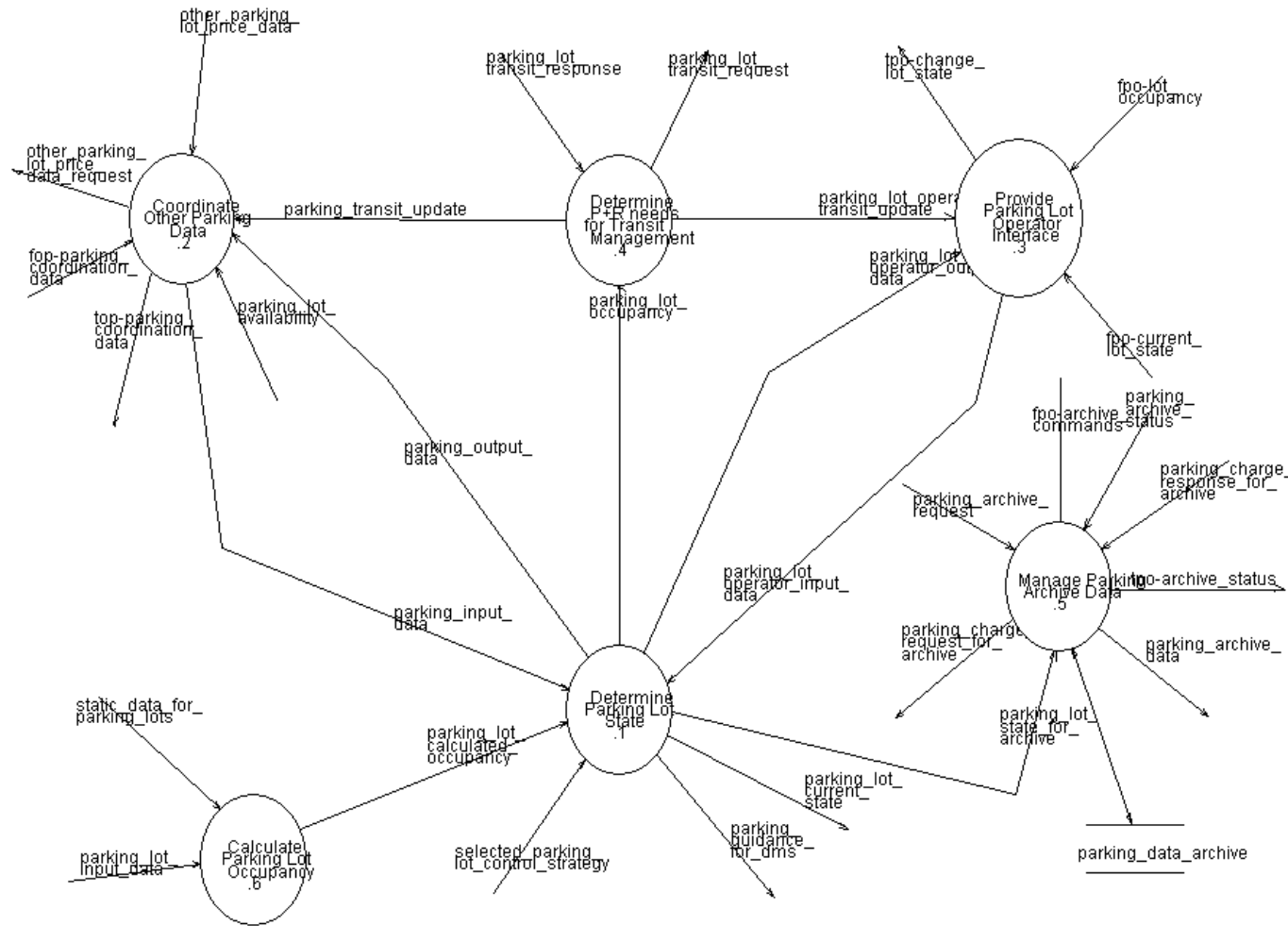
1.2.2-Determine Road and Freeway State



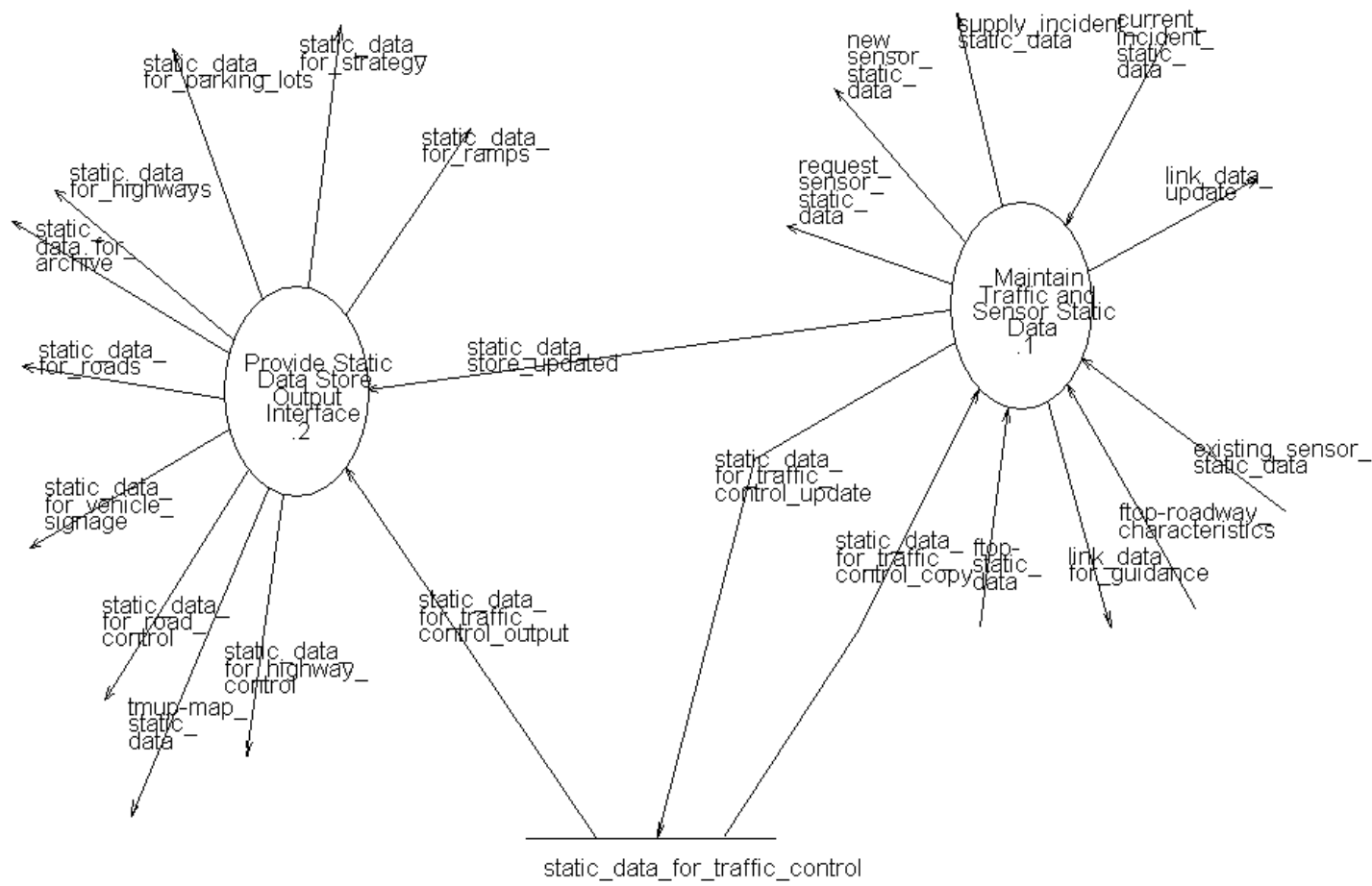
Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



1.2.5- Manage Parking Lot State



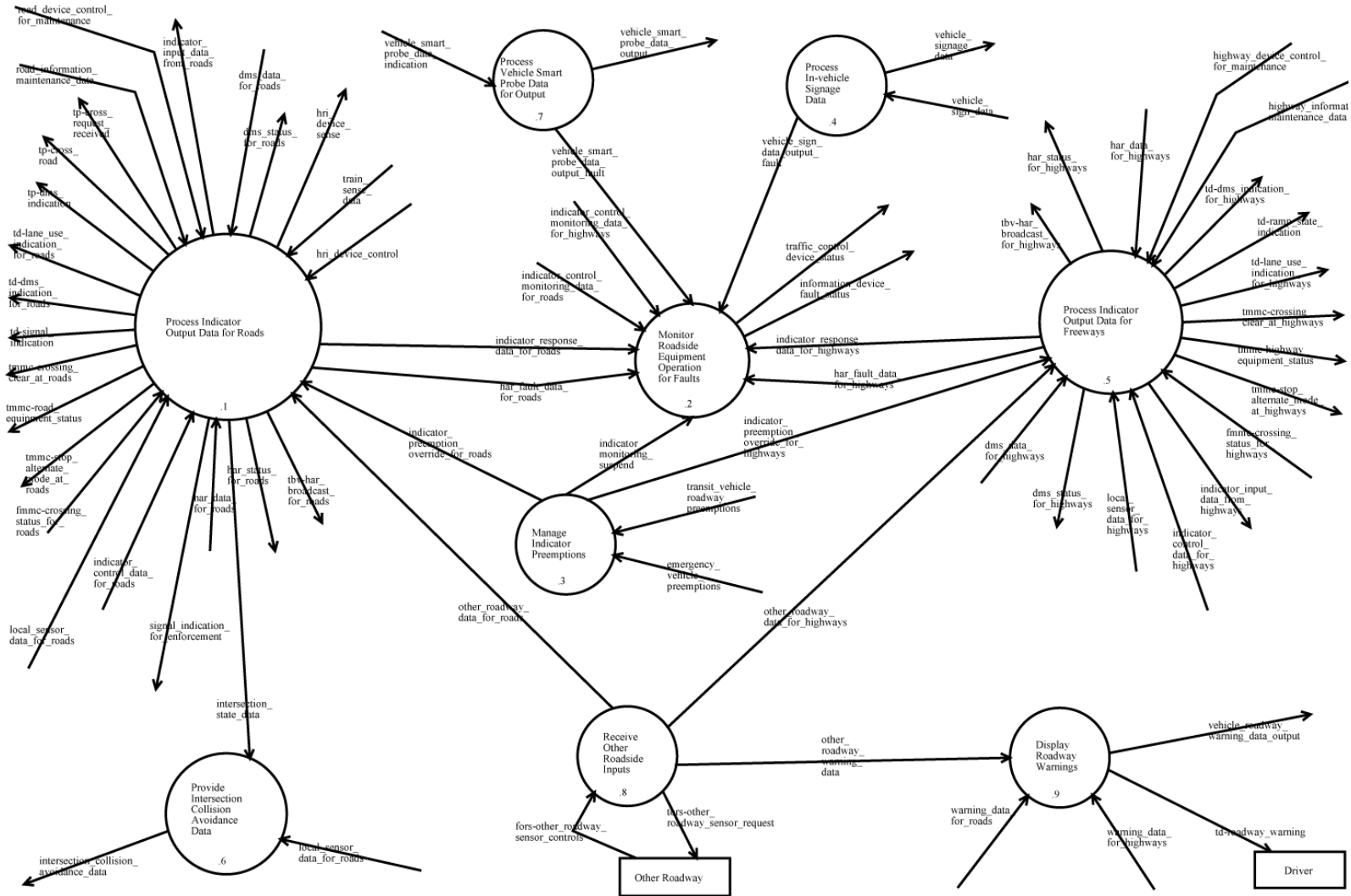
1.2.6-Maintain Static Data for TMC

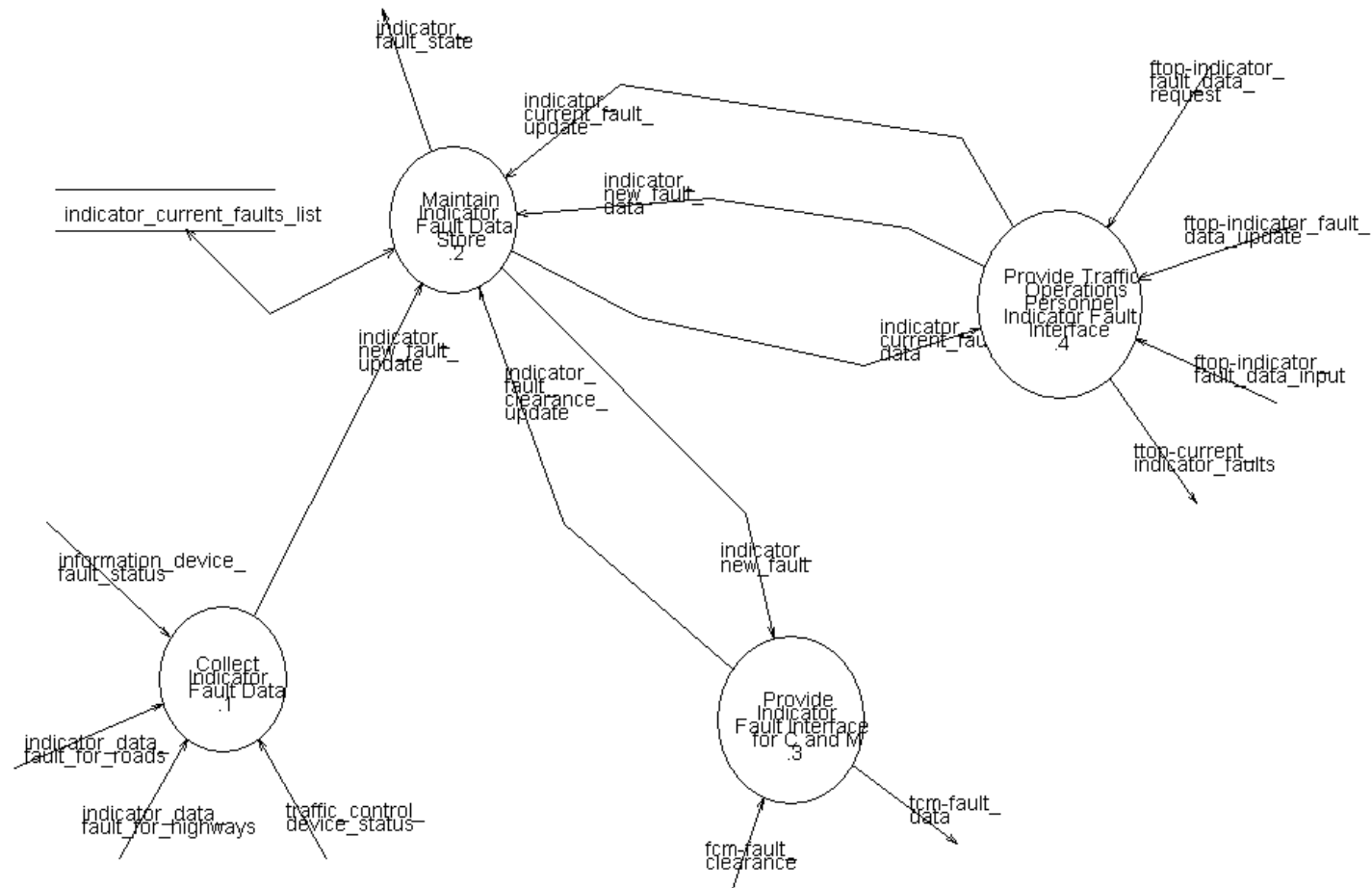
# Development of ITS System Architecture for Malaysia

## Technical Note No. 2

### Logical Architecture Framework - Volume I

1.2.7.7  
Provide Roadside Control Facilities



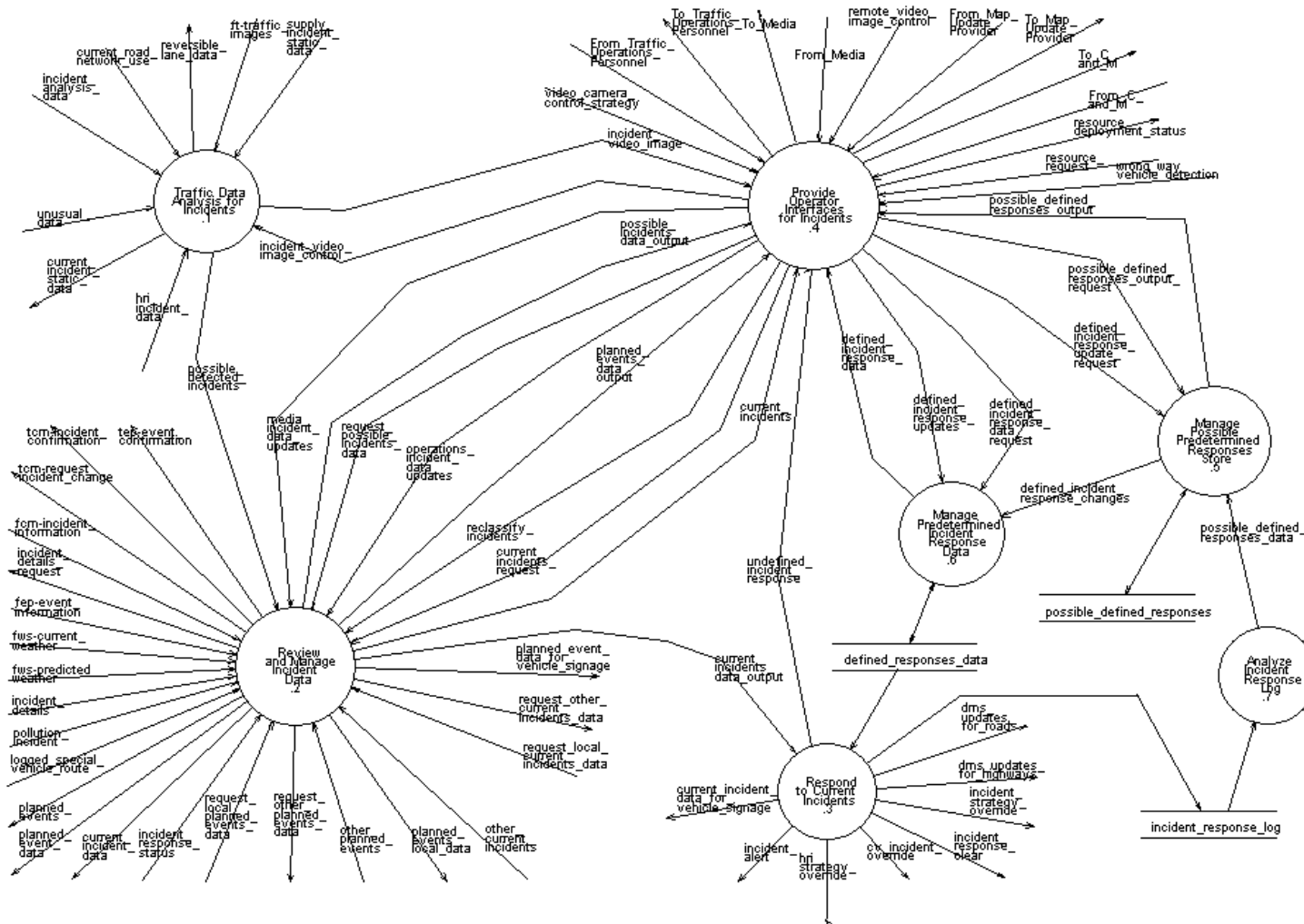


1.2.8-Collect and Process Indicator Fault Data

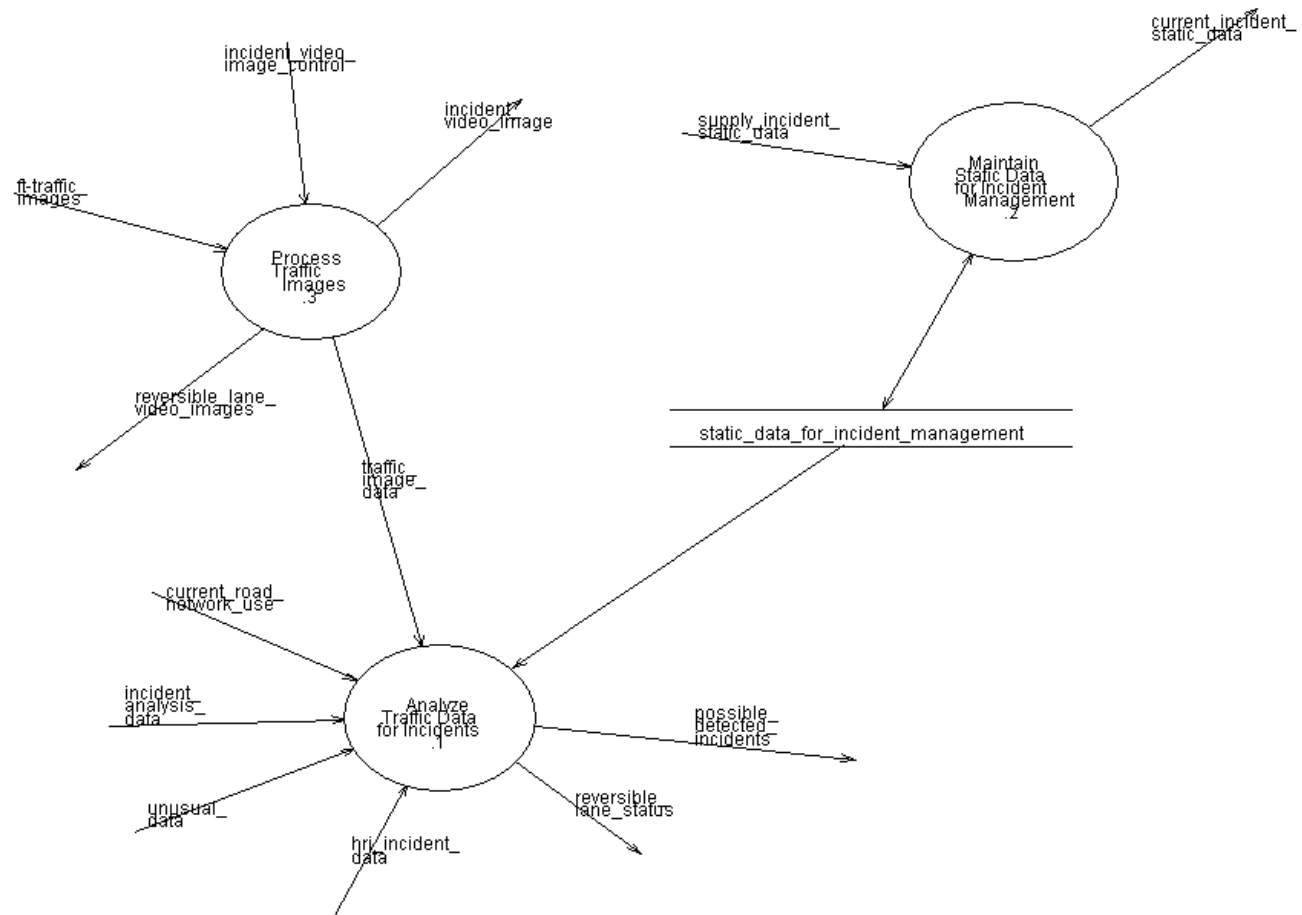
Development of ITS System Architecture for Malaysia

Technical Note No. 2

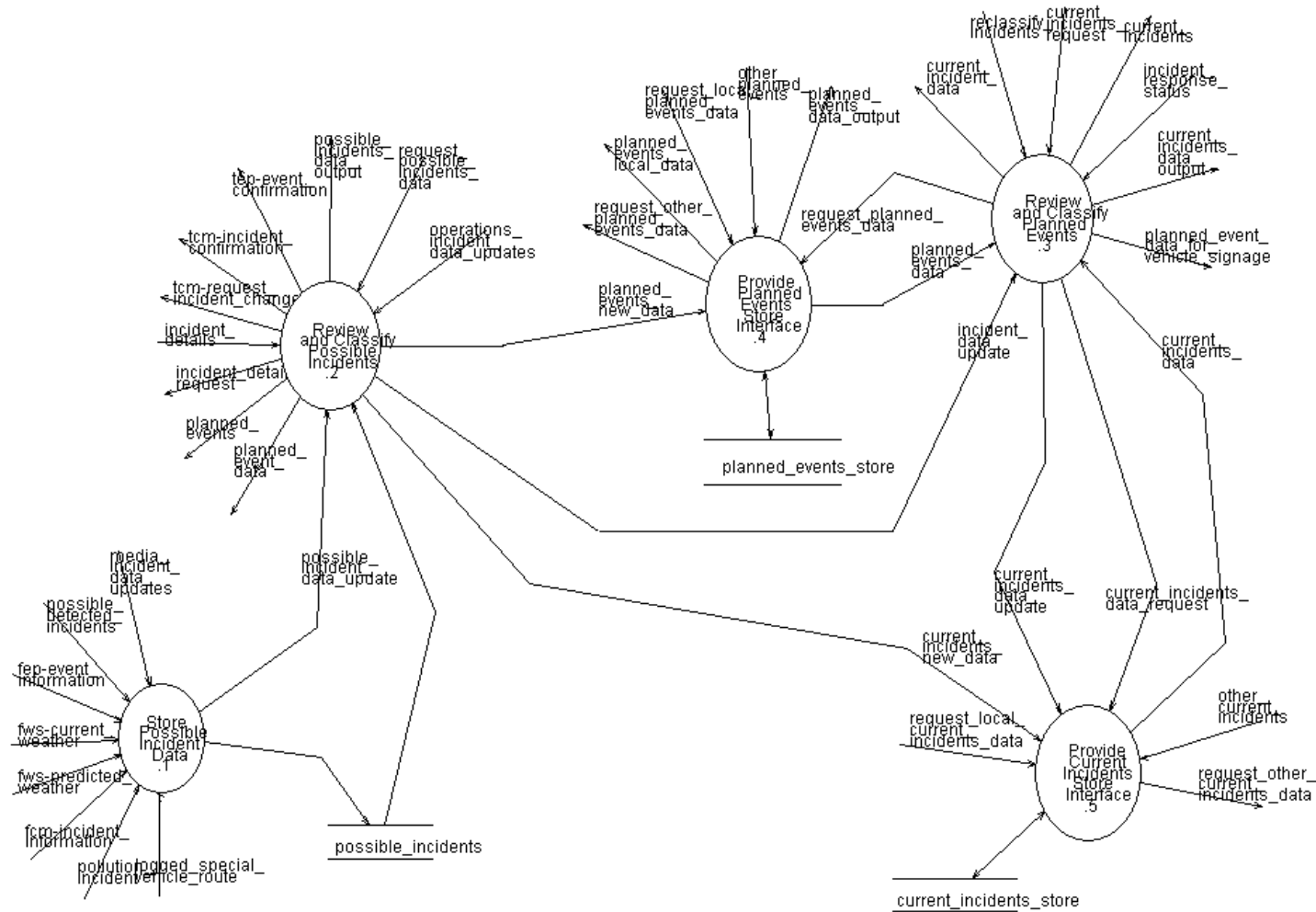
Logical Architecture Framework - Volume I



1.3- Manage Incidents



1.3.1-Traffic Data Analysis for Incidents



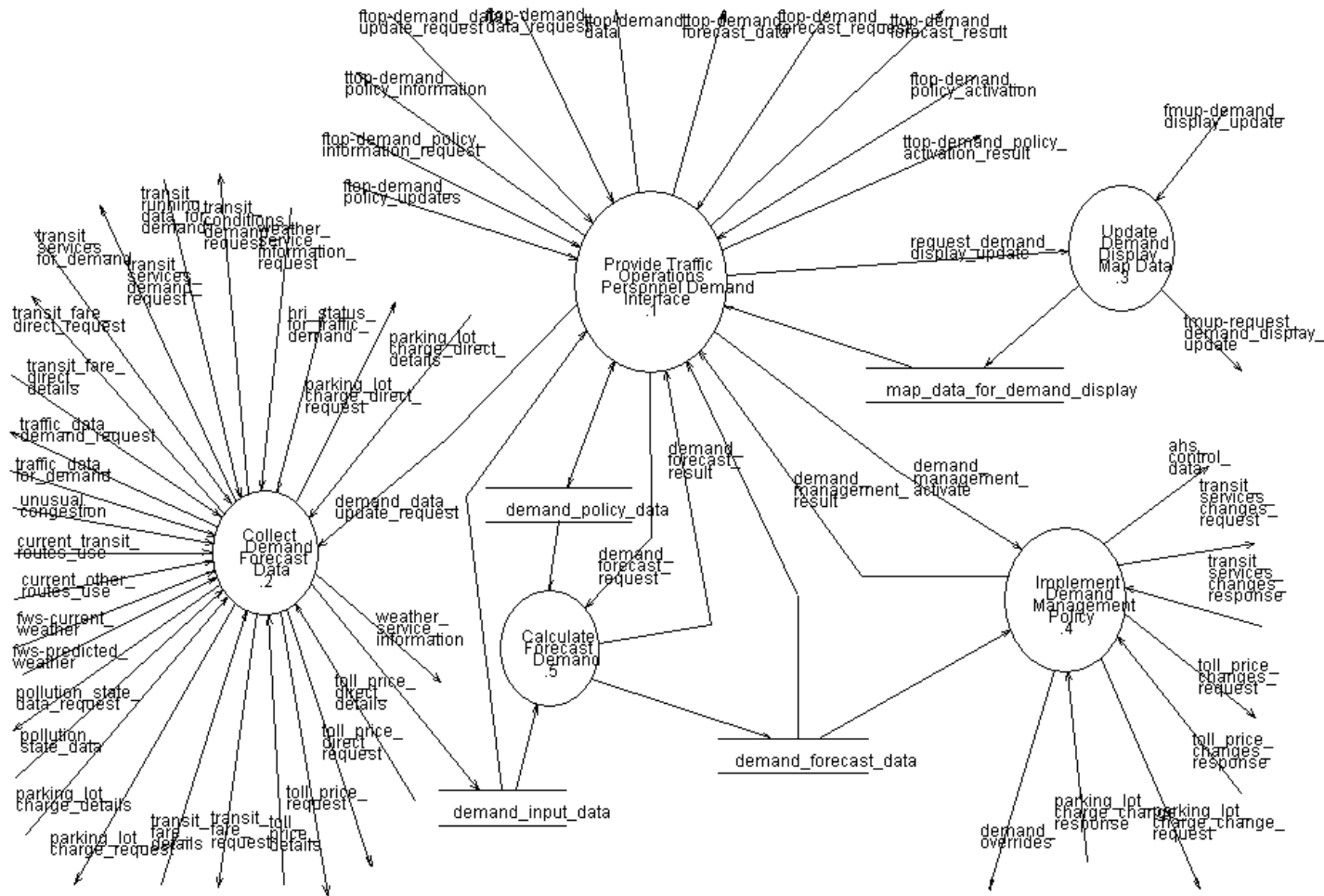
1.3.2-Review and Manage Incident Data



Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

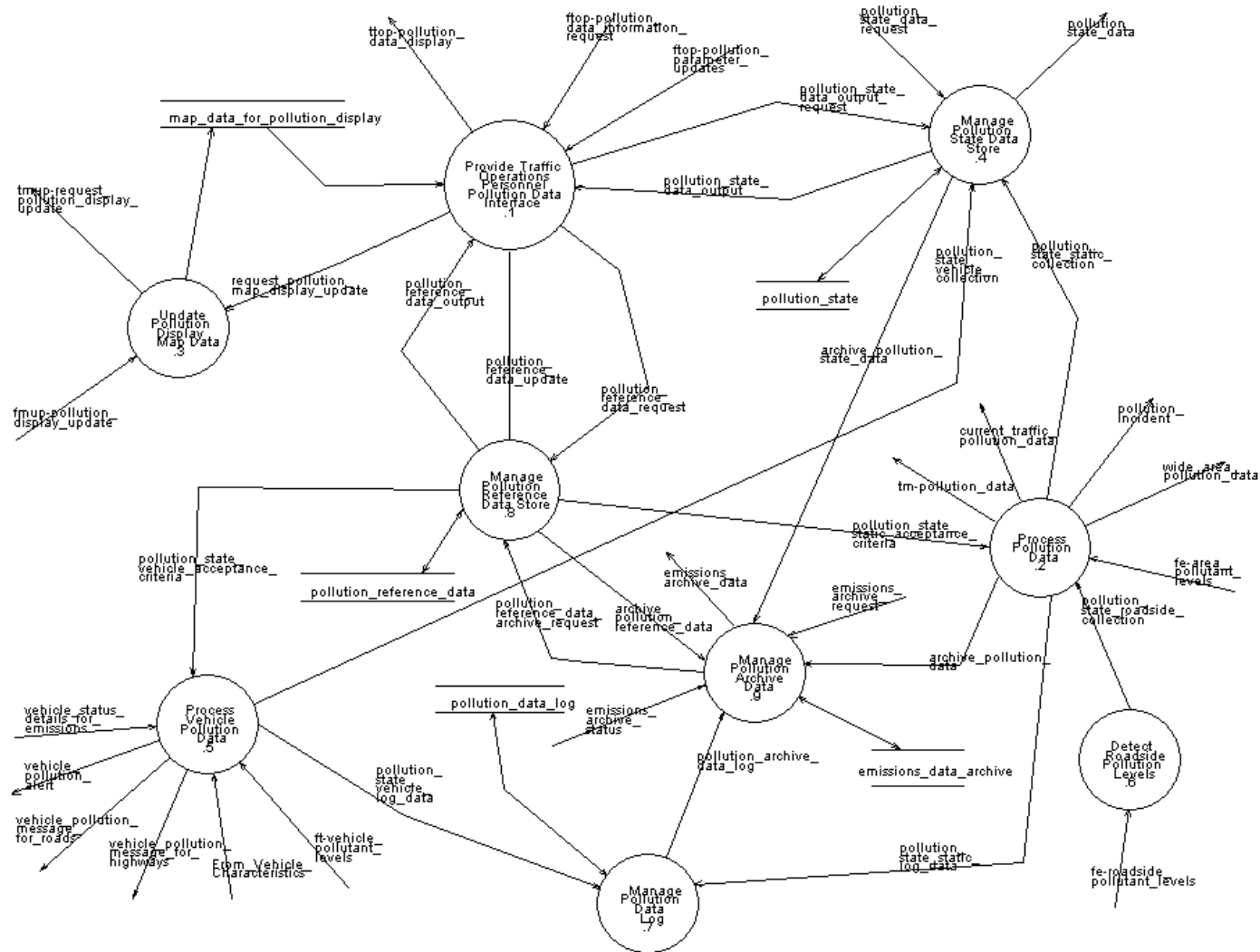


1.4-Manage Travel Demand

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

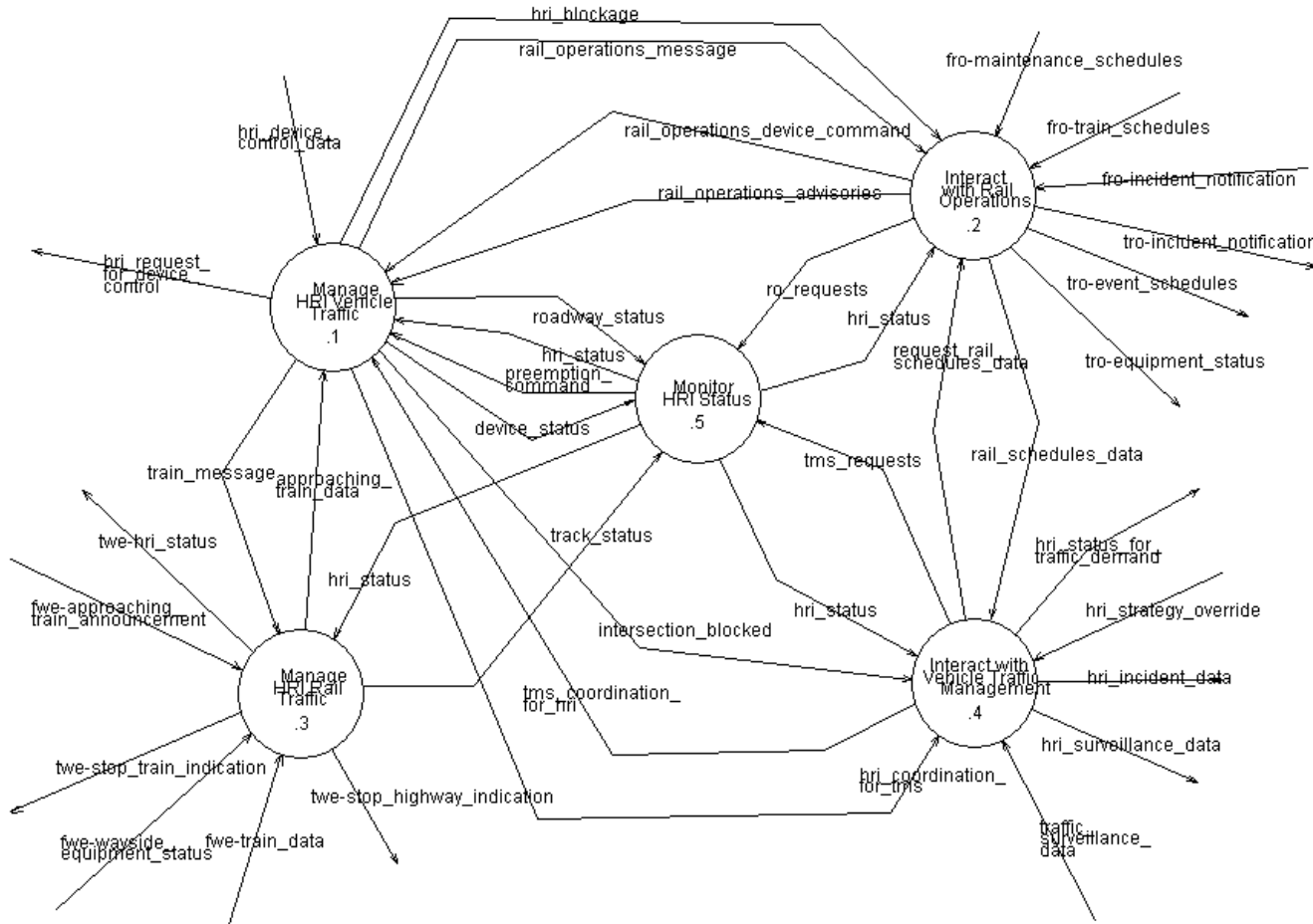


1.5-Manage Emissions

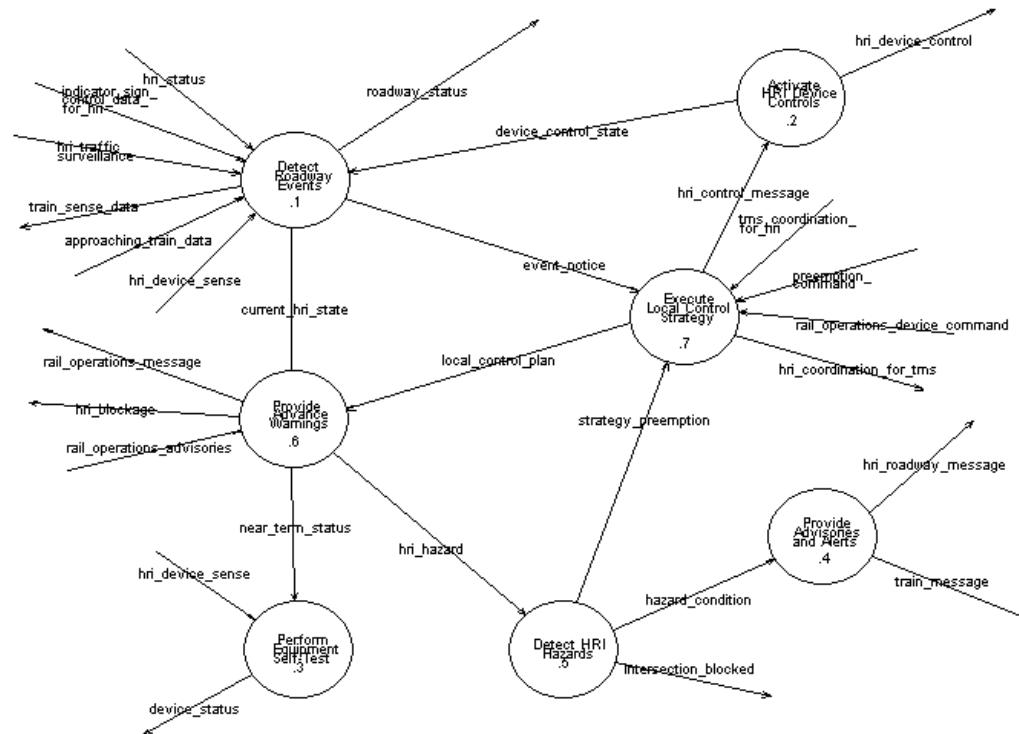
Development of ITS System Architecture for Malaysia

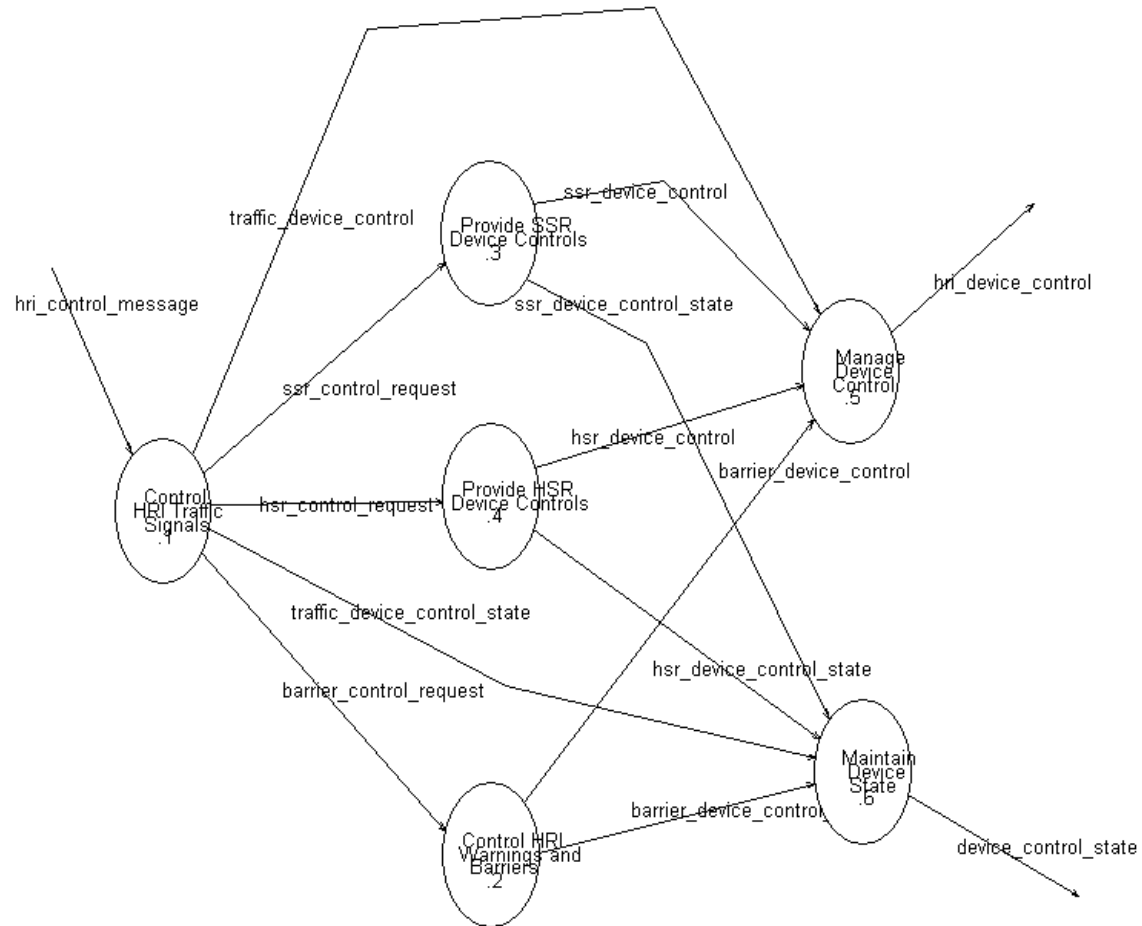
Technical Note No. 2

Logical Architecture Framework - Volume I

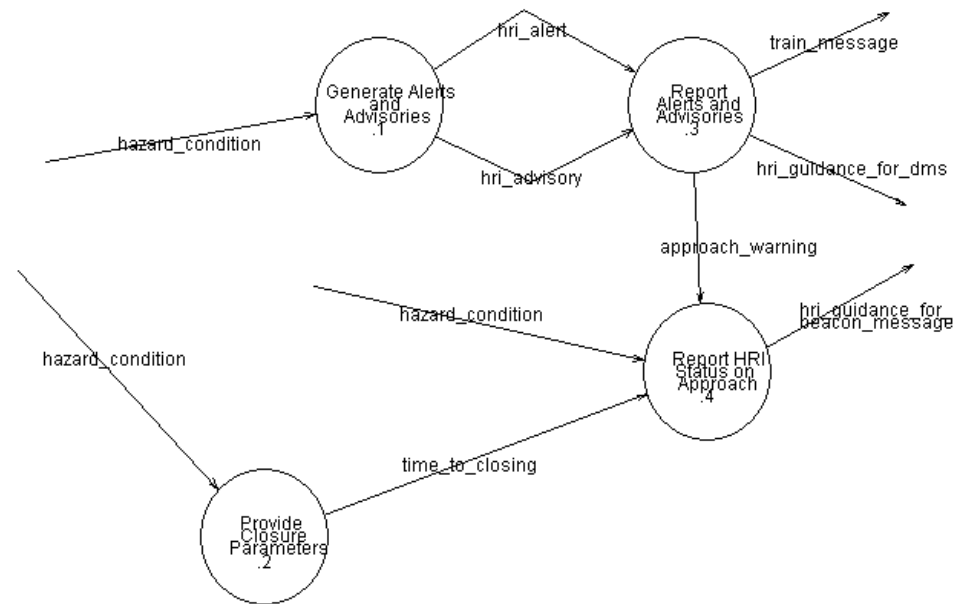


1.6-Manage Highway Rail Intersections

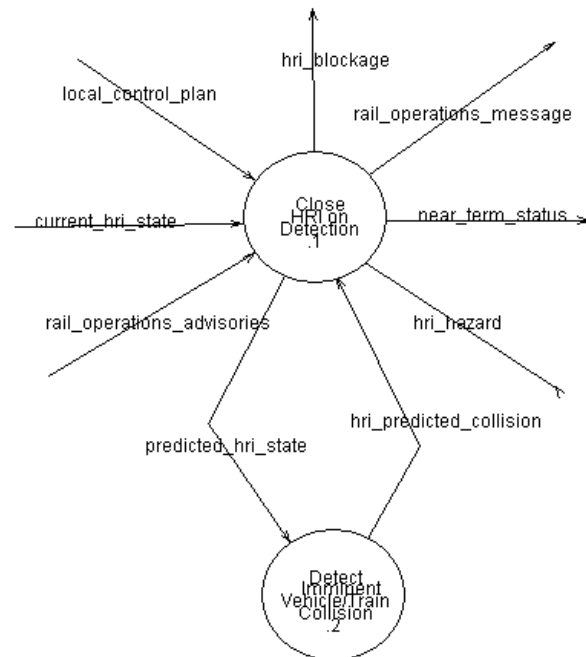




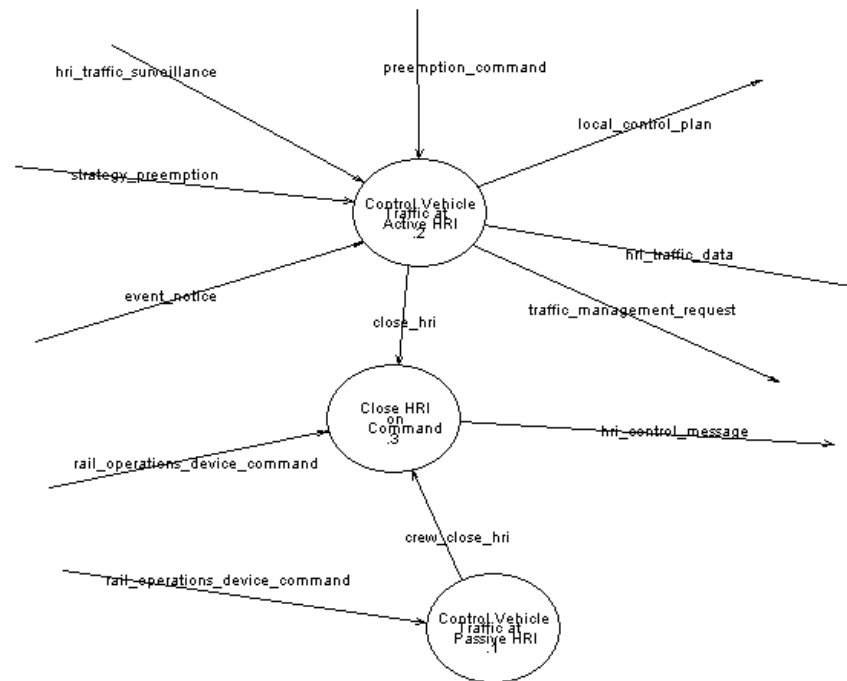
1.6.1.2-Activate HRI Device Controls



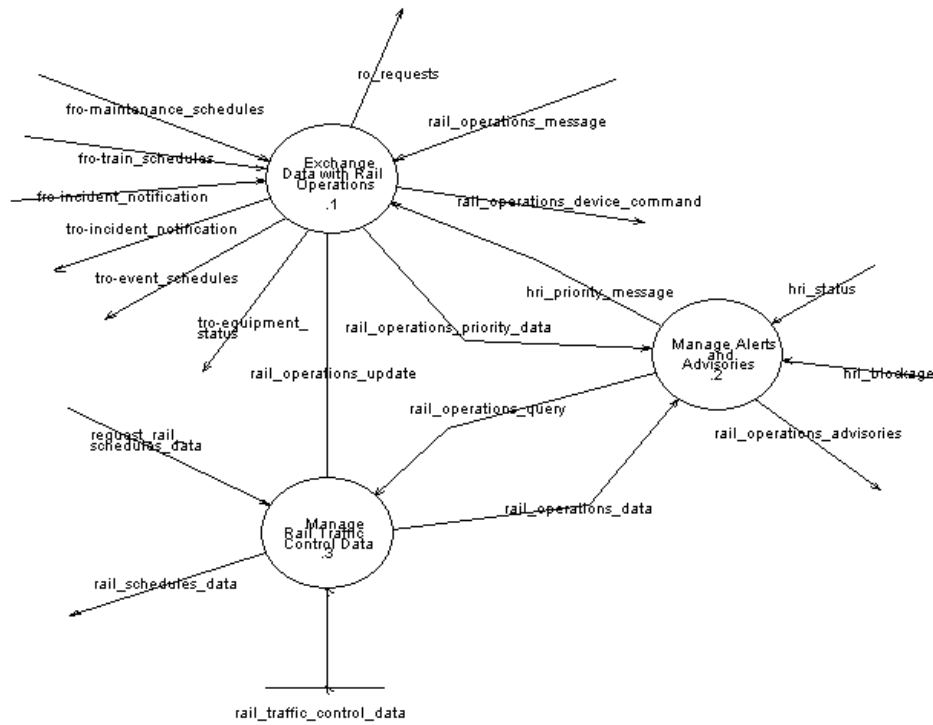
1.6.1.4-Provide Advisories and Alerts



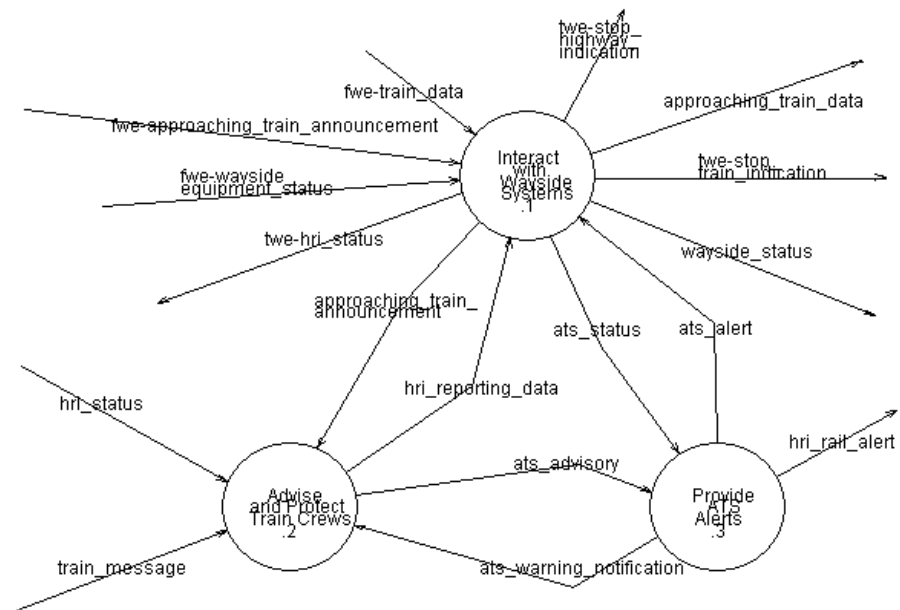
1.6.1.6-Provide Advance Warnings



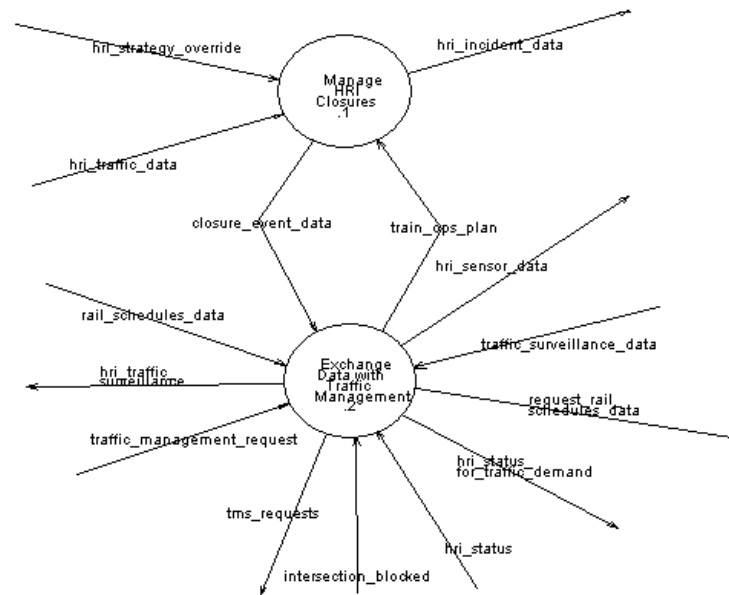
1.6.1.7-Execute Local Control Strategy



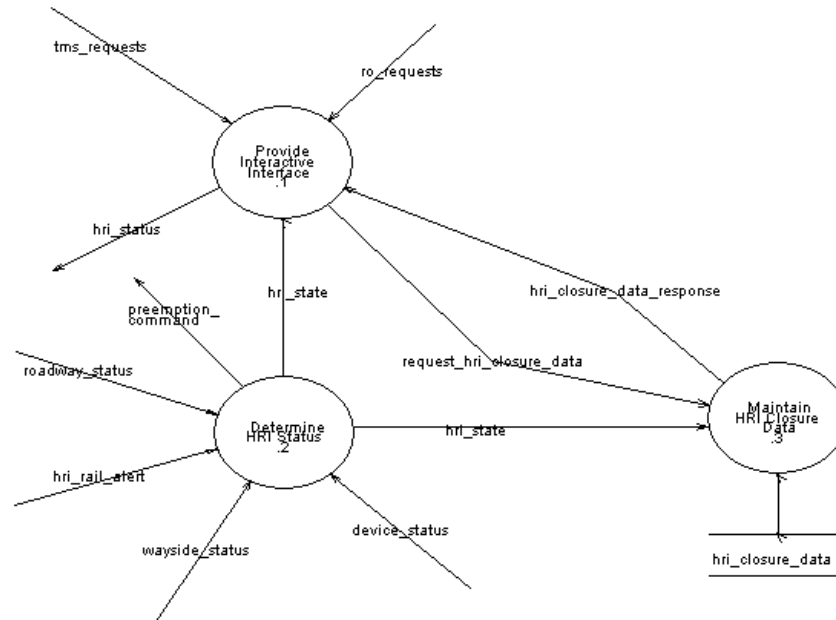
1.6.2-Interact with Rail Operations



1.6.3-Manage HRI Rail Traffic



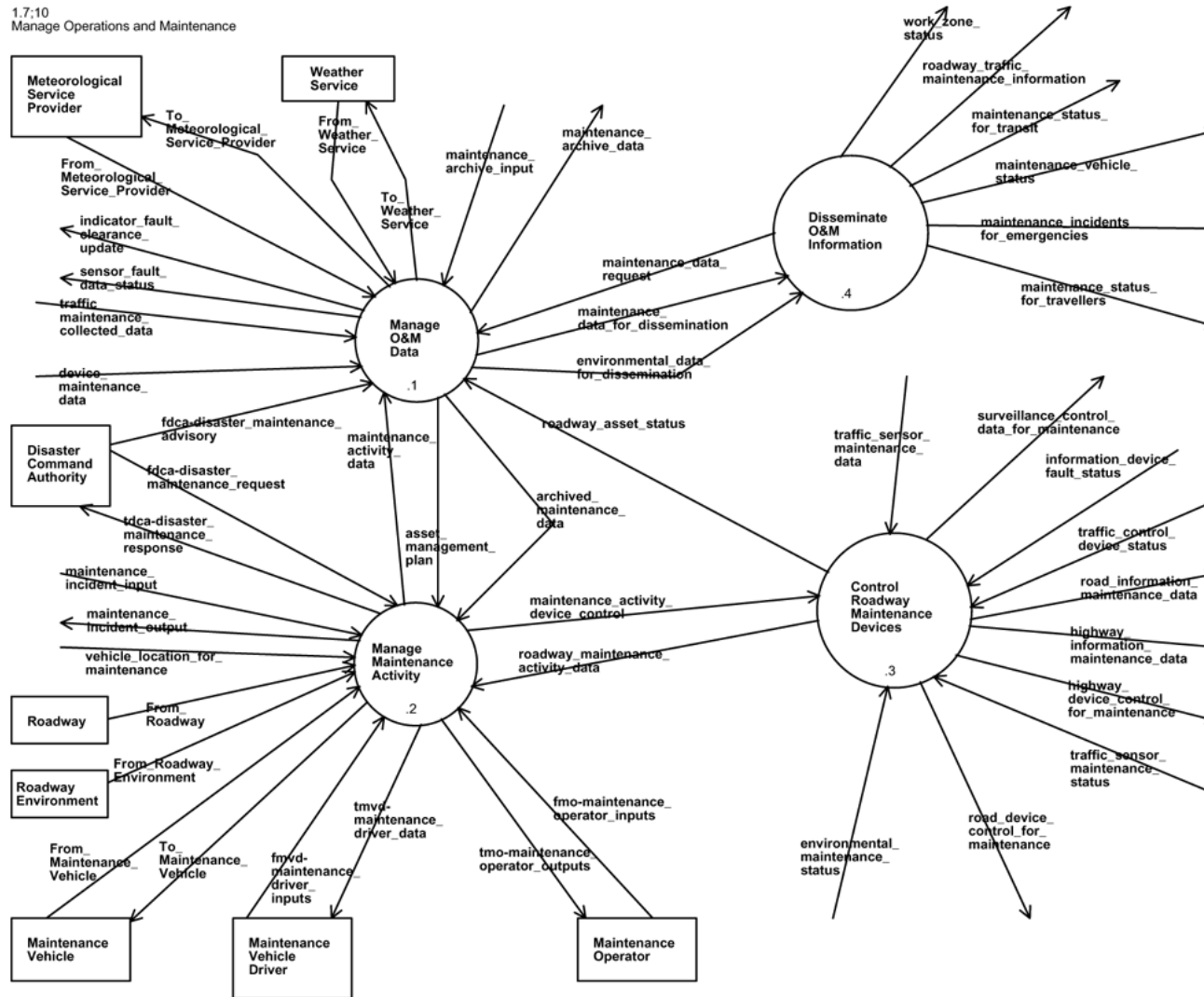
1.6.4 Interact with Vehicle Traffic Management



Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

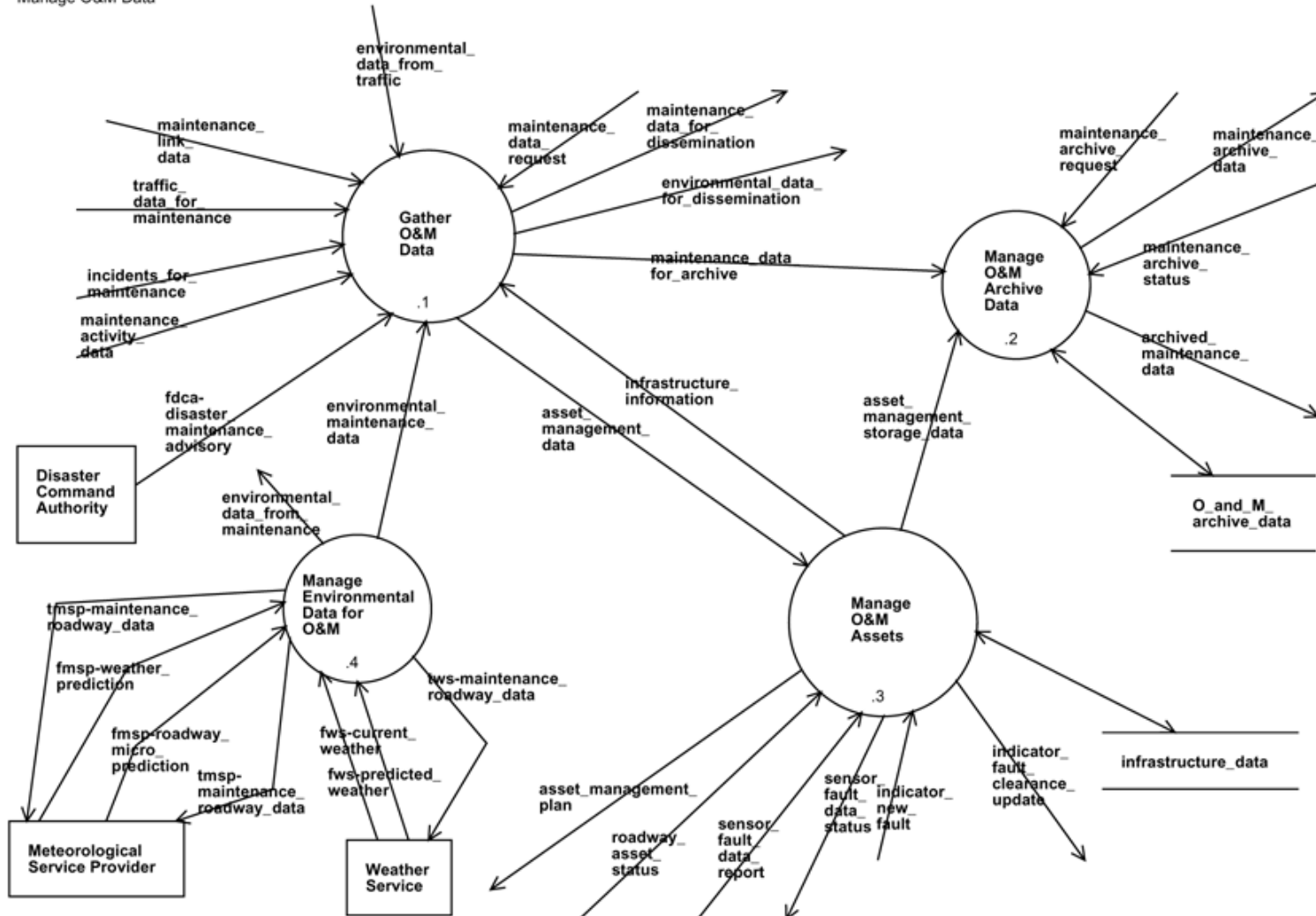


Development of ITS System Architecture for Malaysia

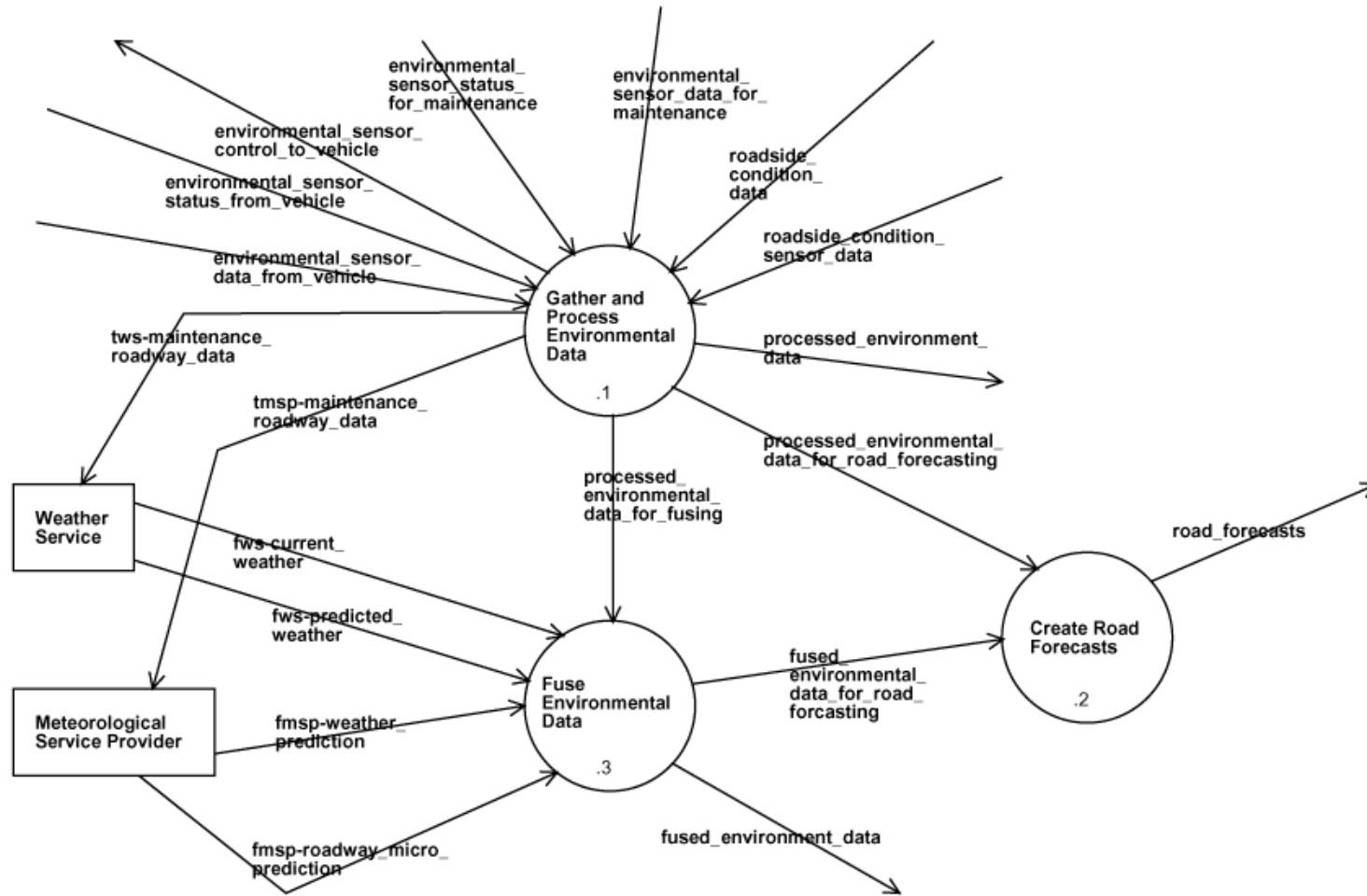
Technical Note No. 2

Logical Architecture Framework - Volume I

1.7.1;10  
Manage O&M Data



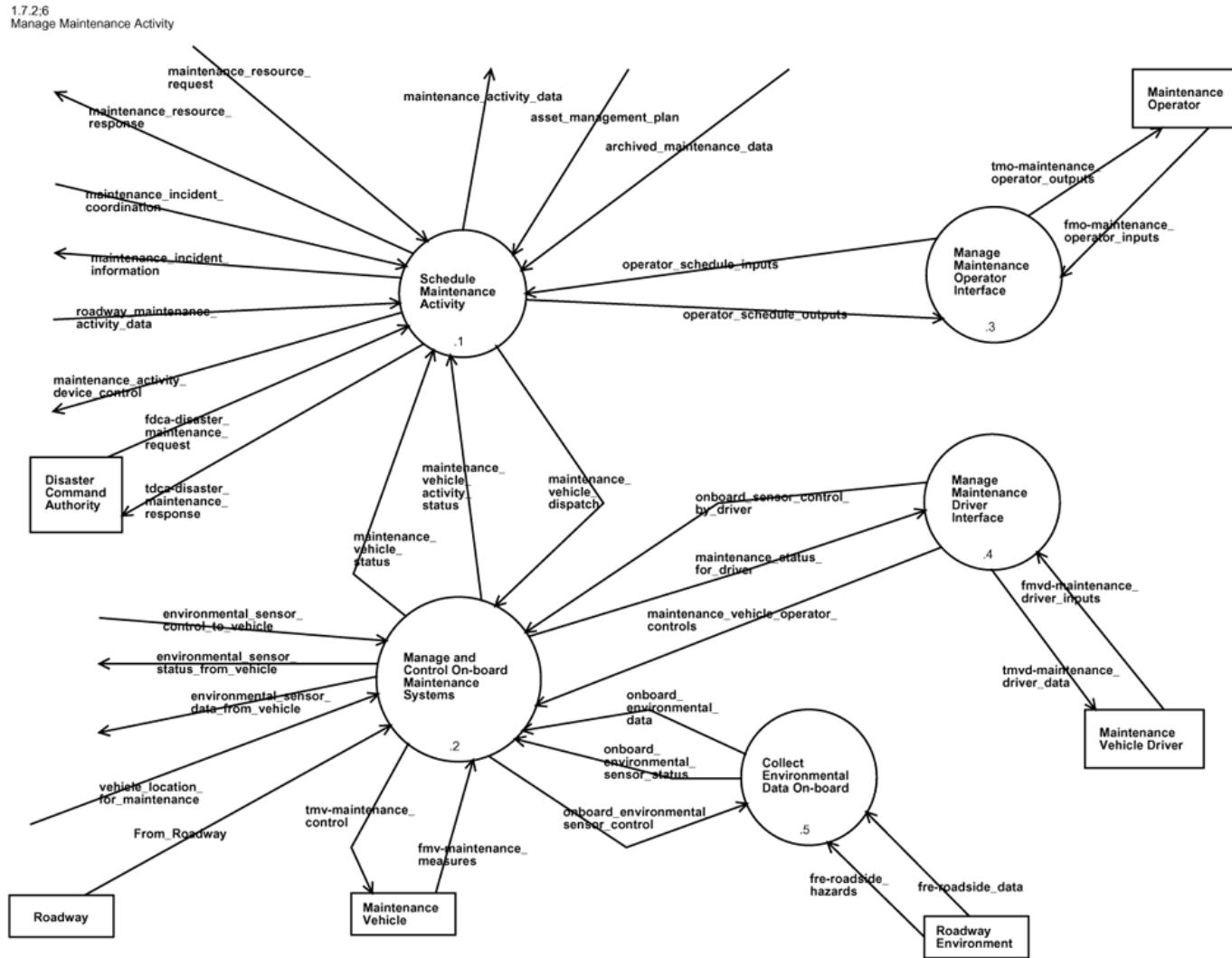
1.7.1.4;5  
Manage Environmental Data for O&M



Development of ITS System Architecture for Malaysia

Technical Note No. 2

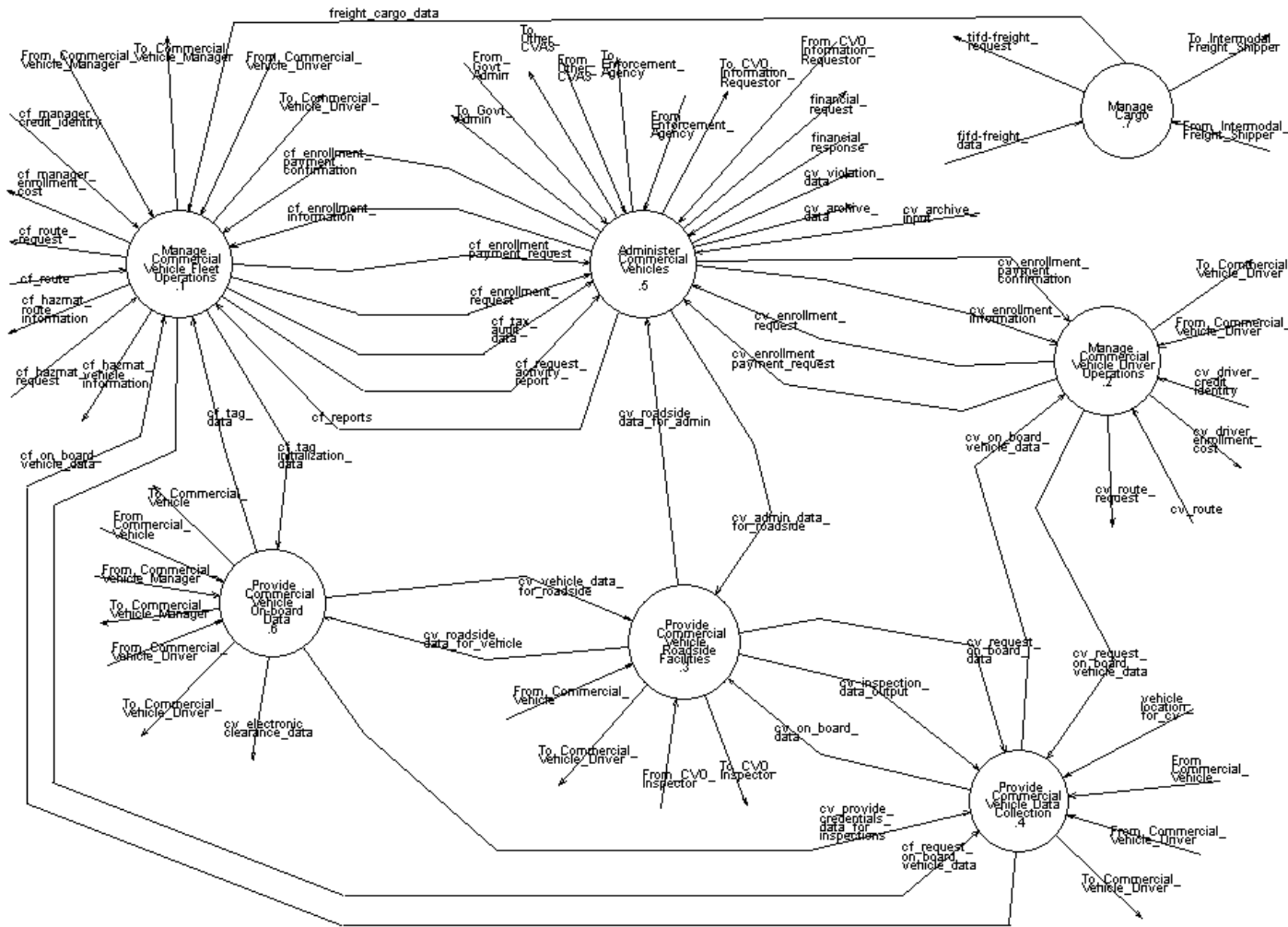
Logical Architecture Framework - Volume I



## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



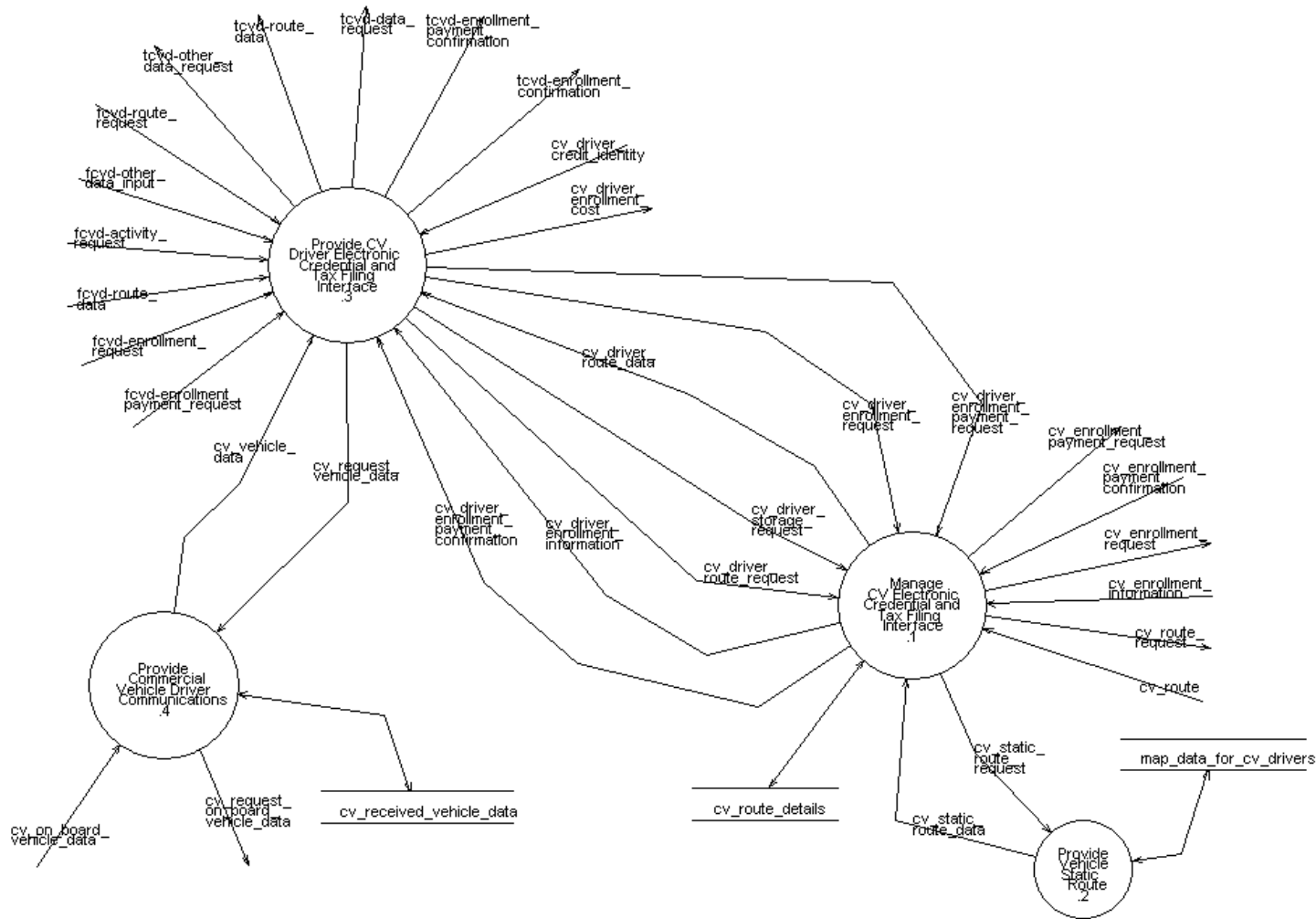
2-Manage Commercial Vehicles



Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

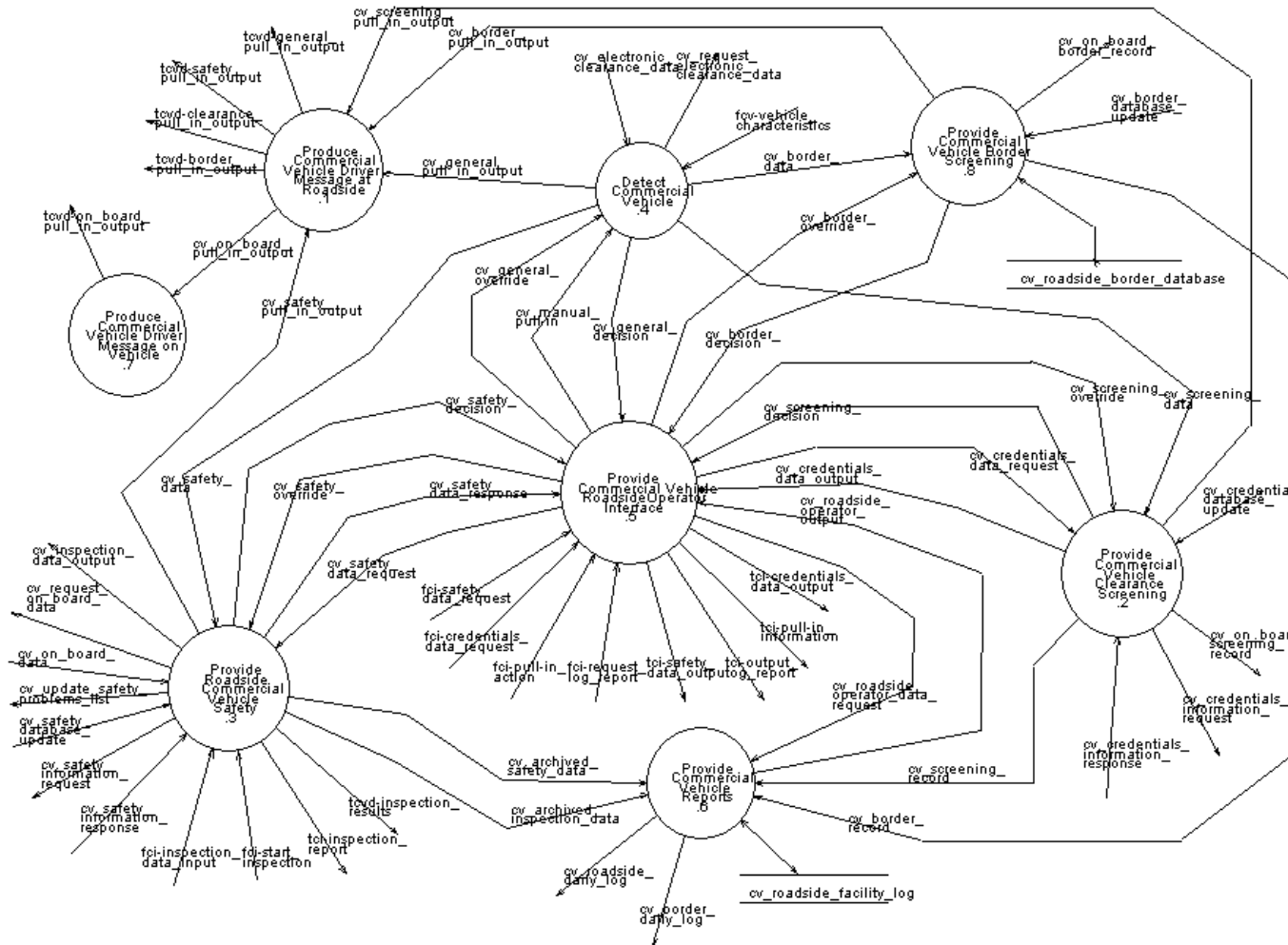


2.2-Manage Commercial Vehicle Driver Operations

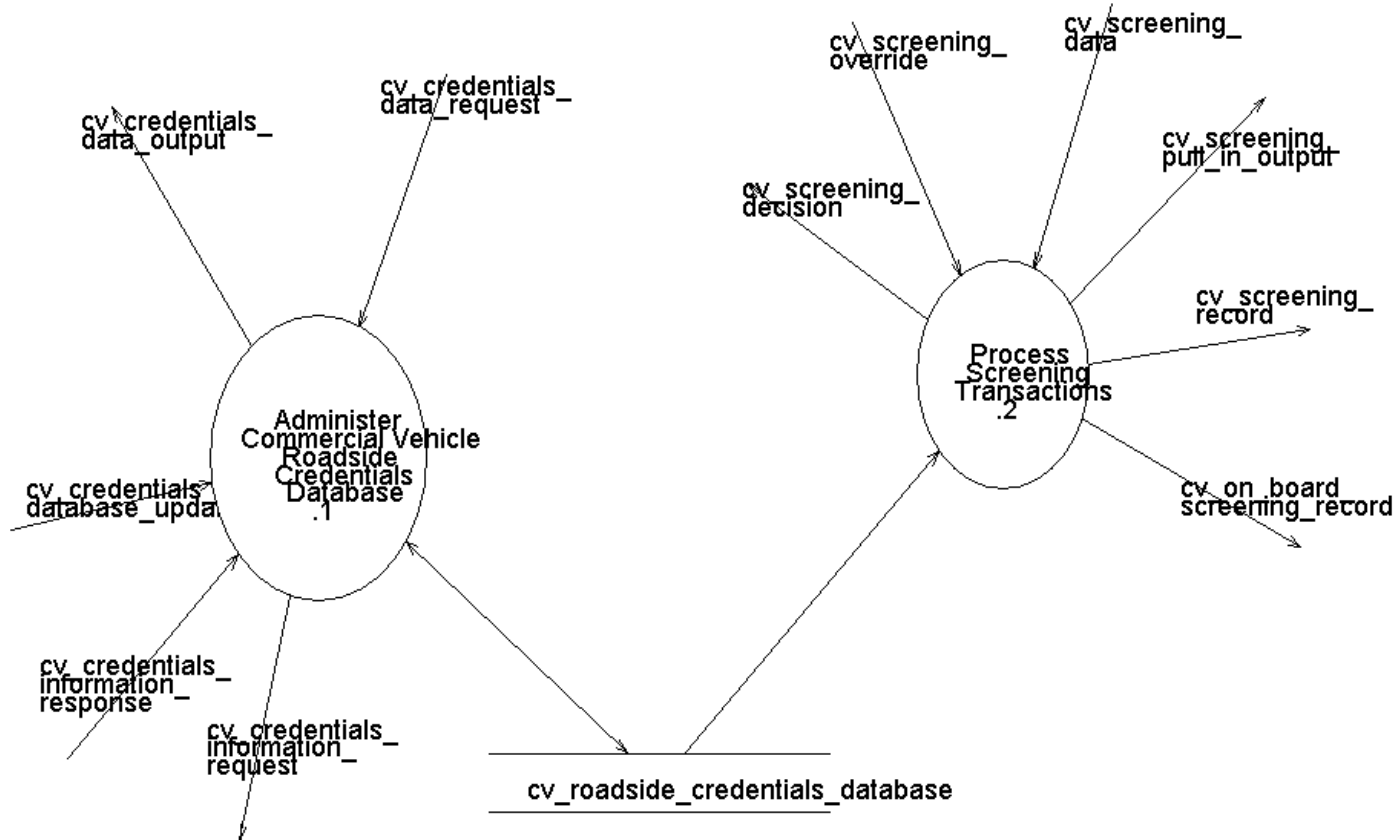
## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



2.3-Provide Commercial Vehicle Roadside Facilities

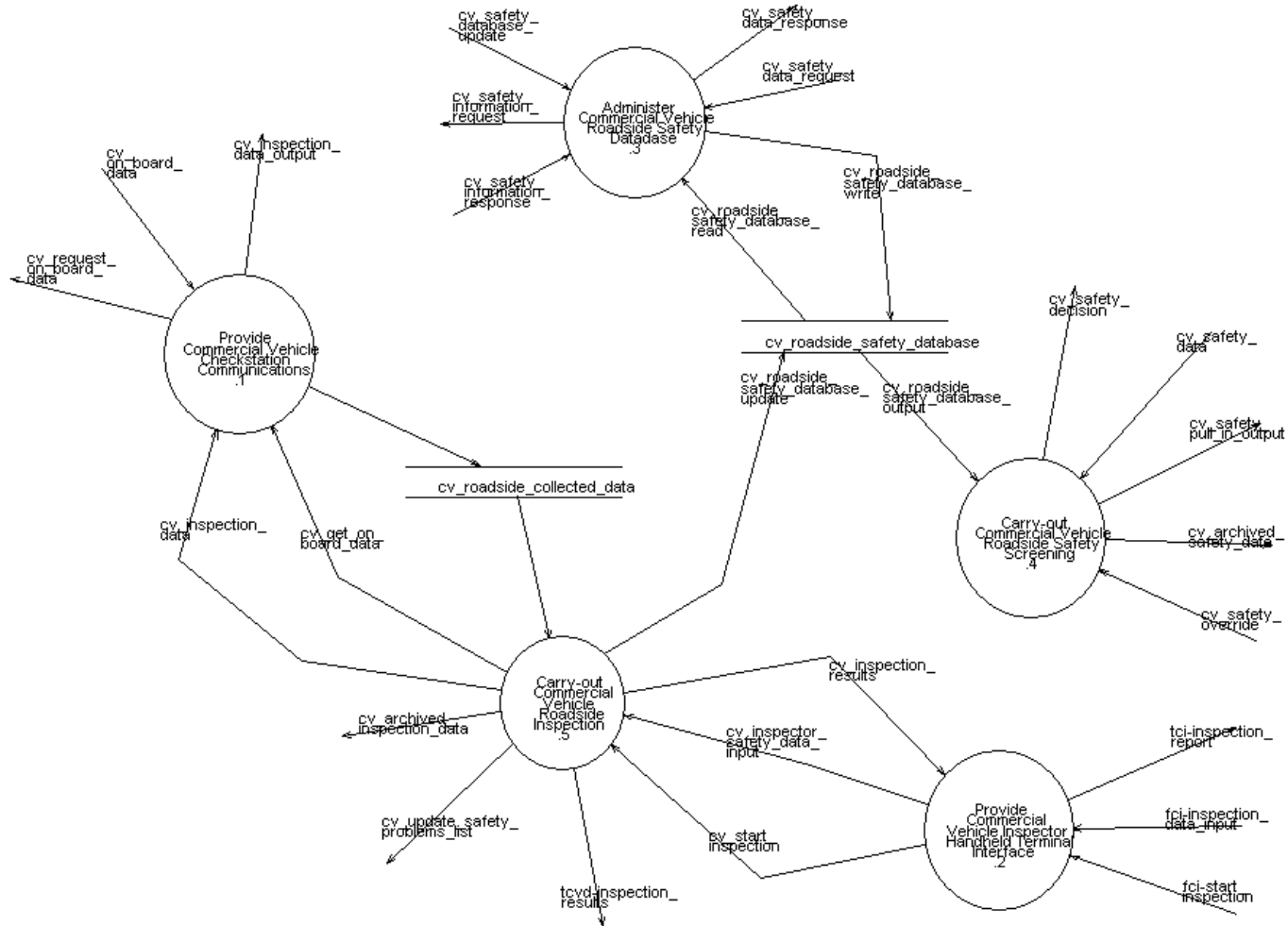


2.3.2-Provide Commercial Vehicle Clearance Screening

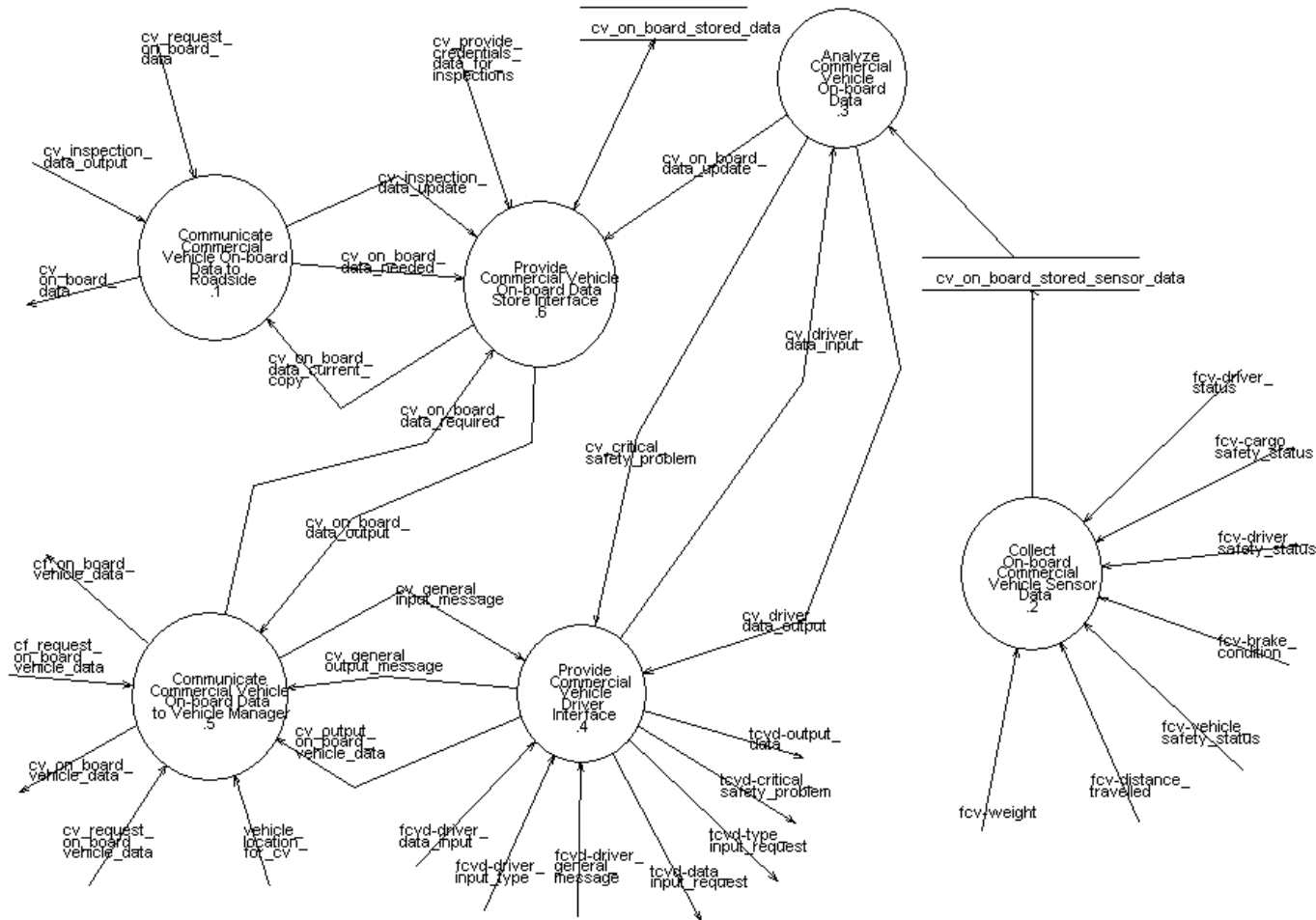
## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



2.3.3-Provide Roadside Commercial Vehicle Safety

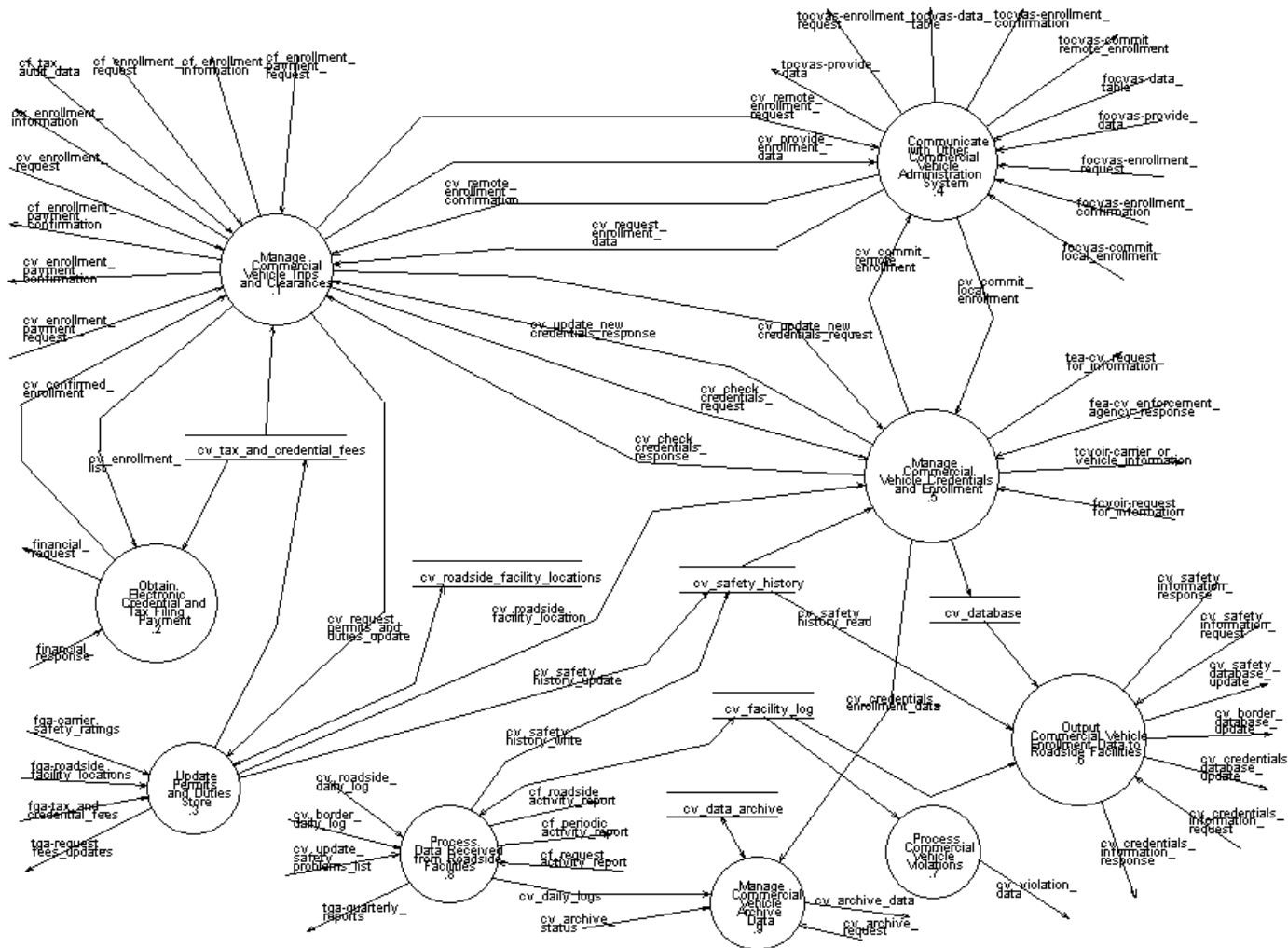


2.4-Provide Commercial Vehicle Data Collection

## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

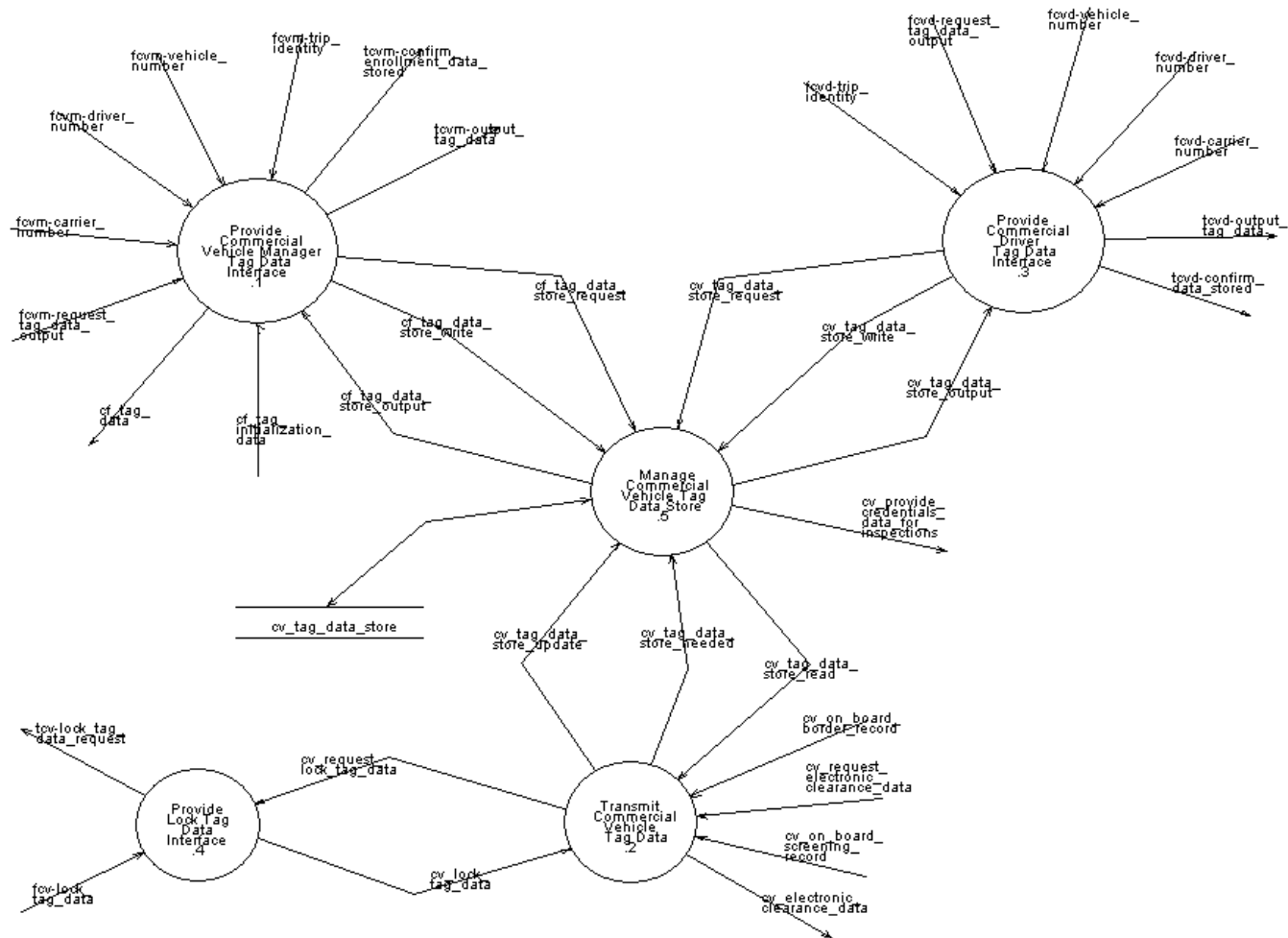


2.5-Administer Commercial Vehicles

## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



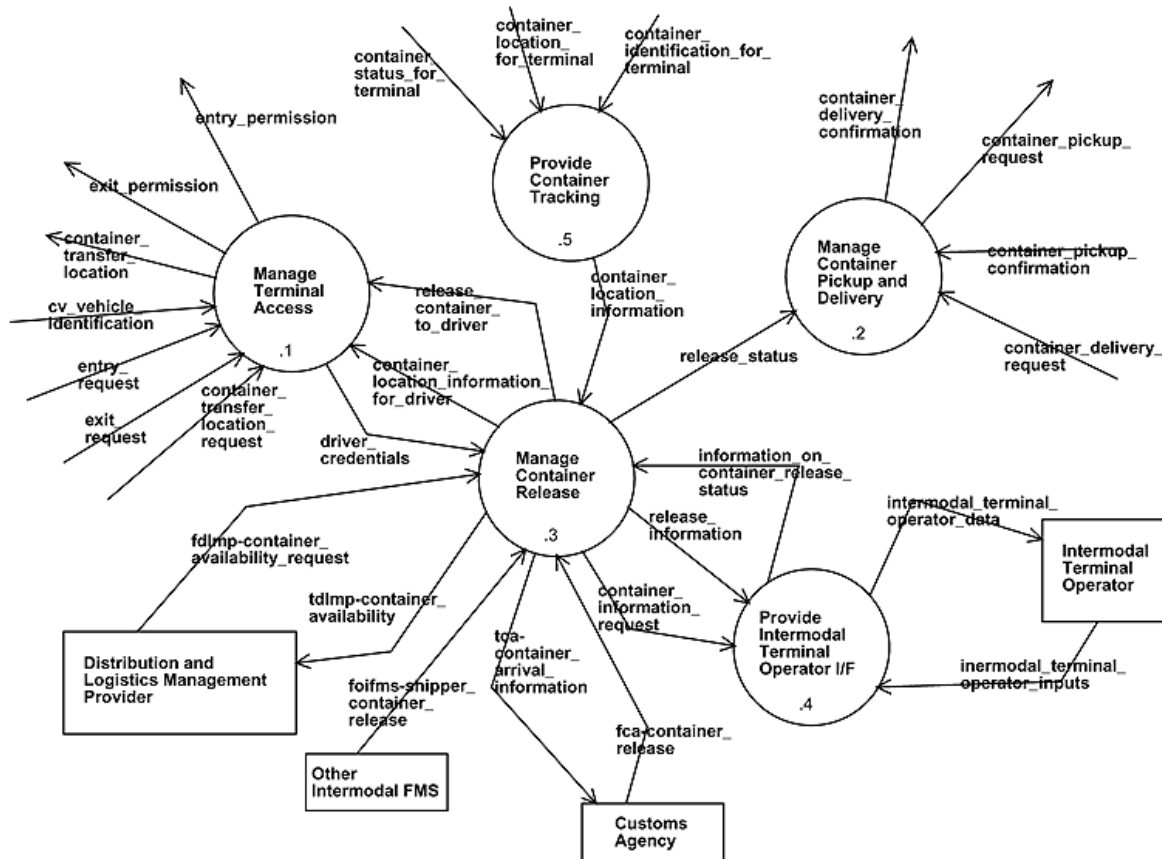
2.6-Provide Commercial Vehicle On-board Data

## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

2.7.1;7  
Manage Intermodal Terminal Interface

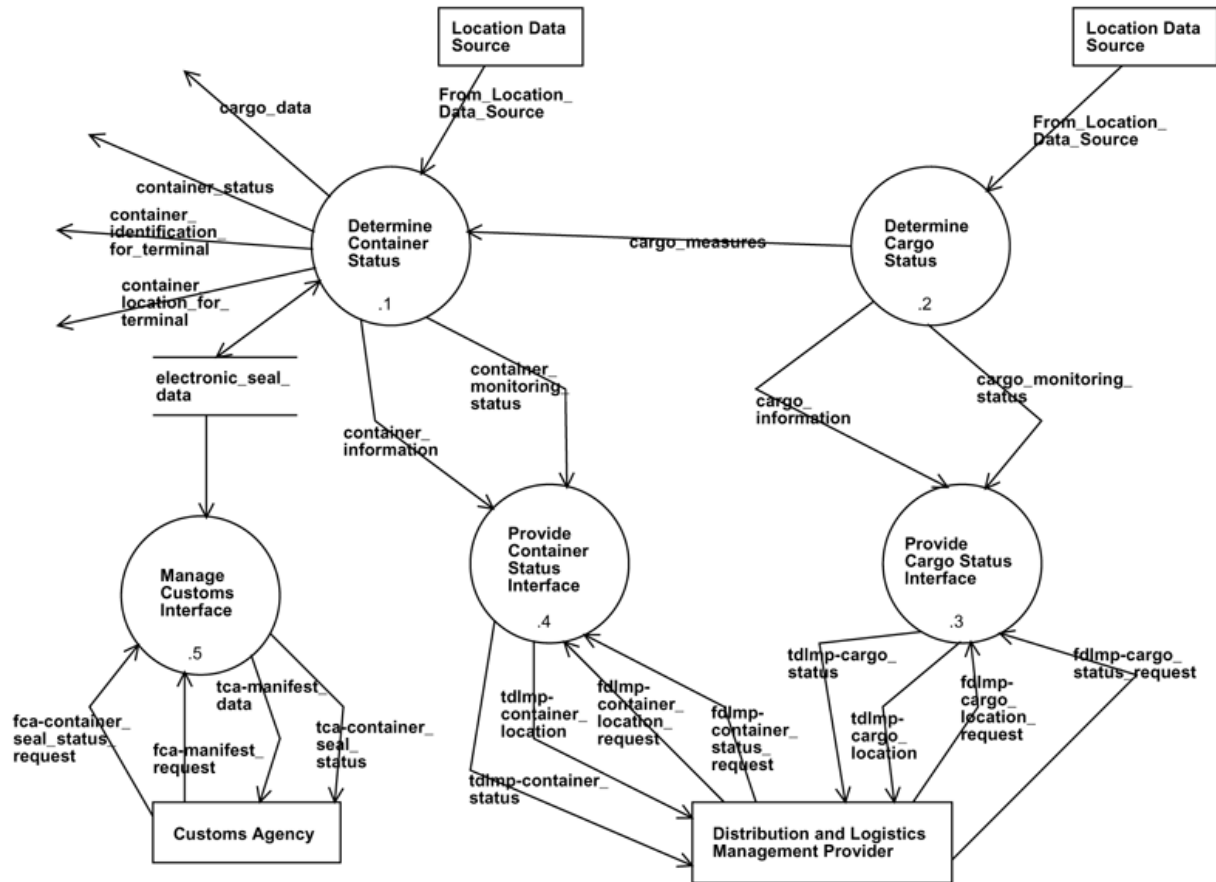


Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

2.7.2;8  
Manage Intermodal Container

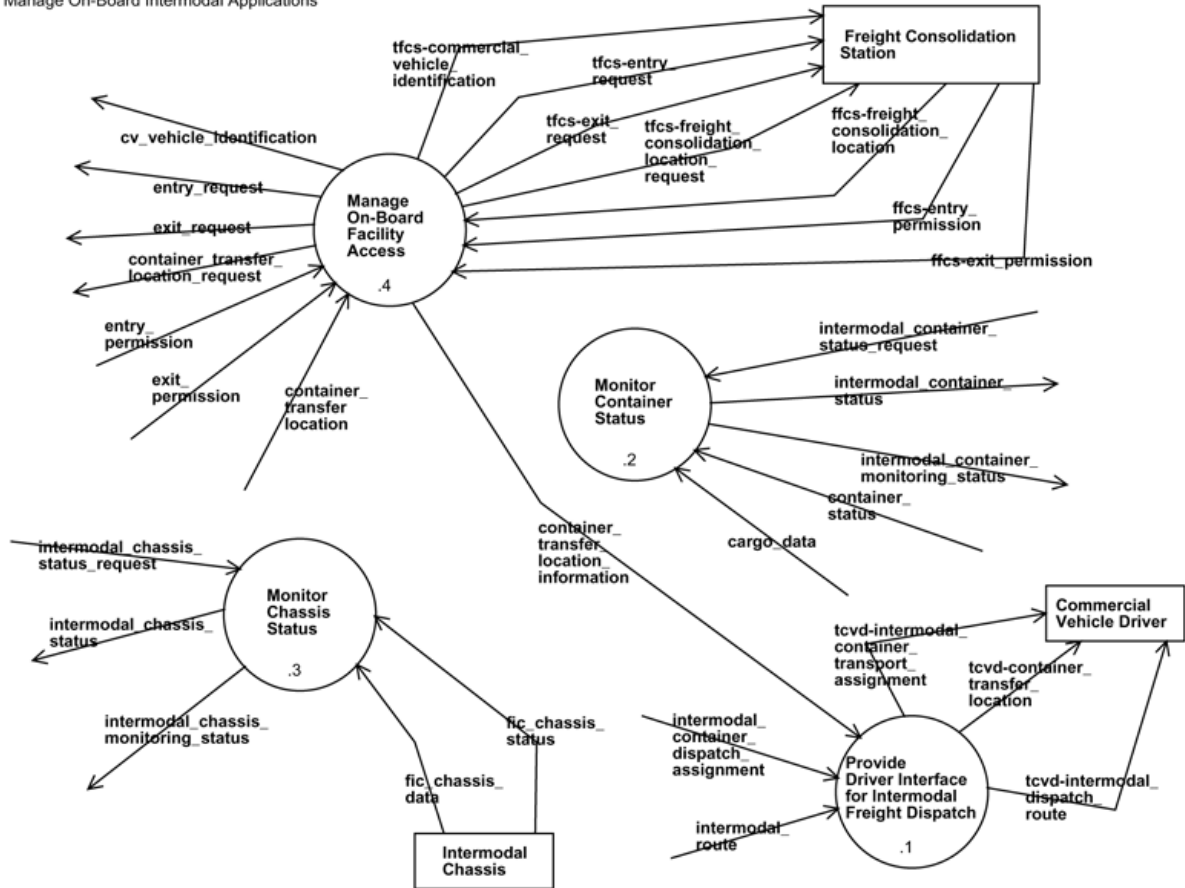


Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

2.7.3;5  
Manage On-Board Intermodal Applications

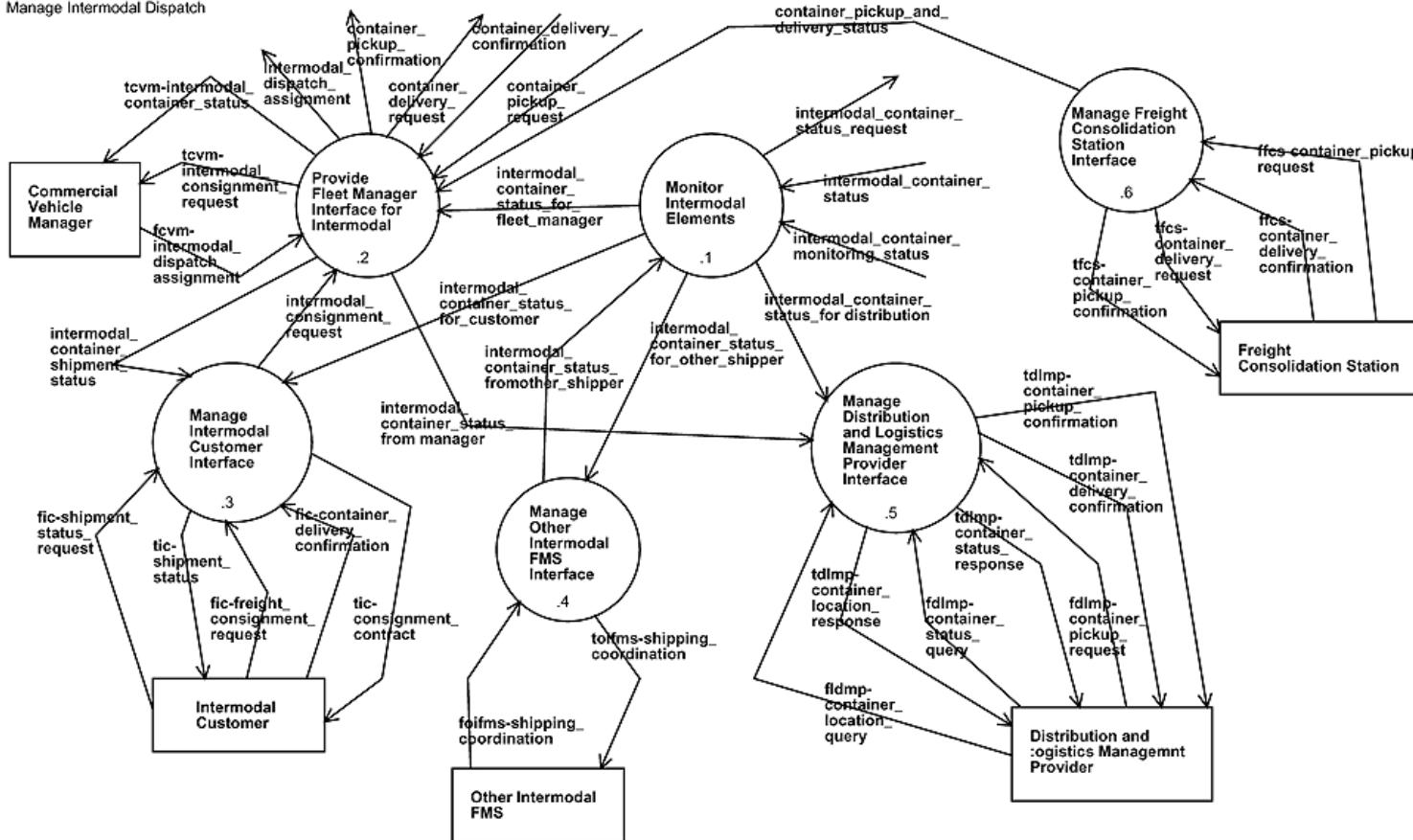


Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

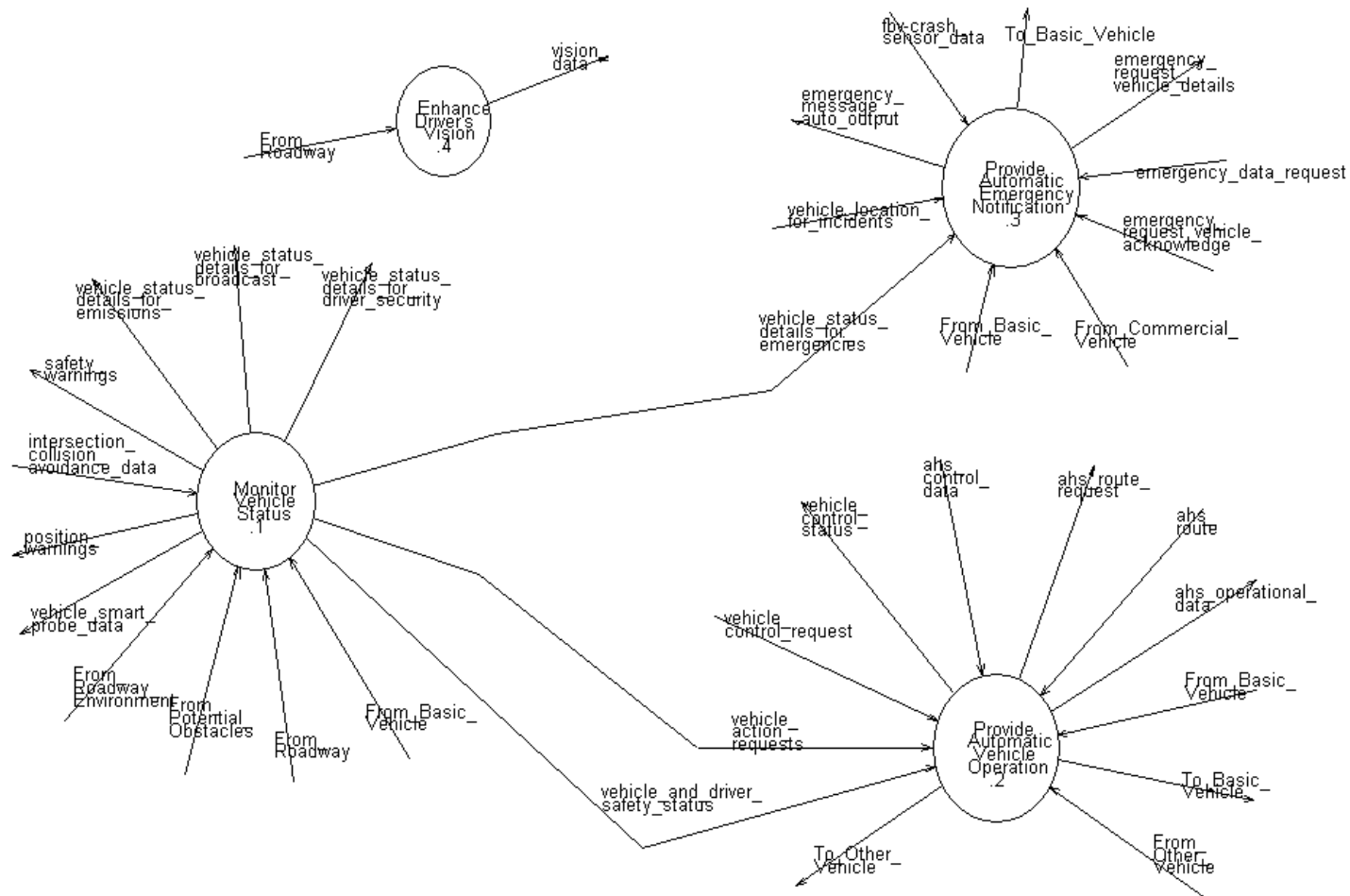
2.7.4.6  
Manage Intermodal Dispatch



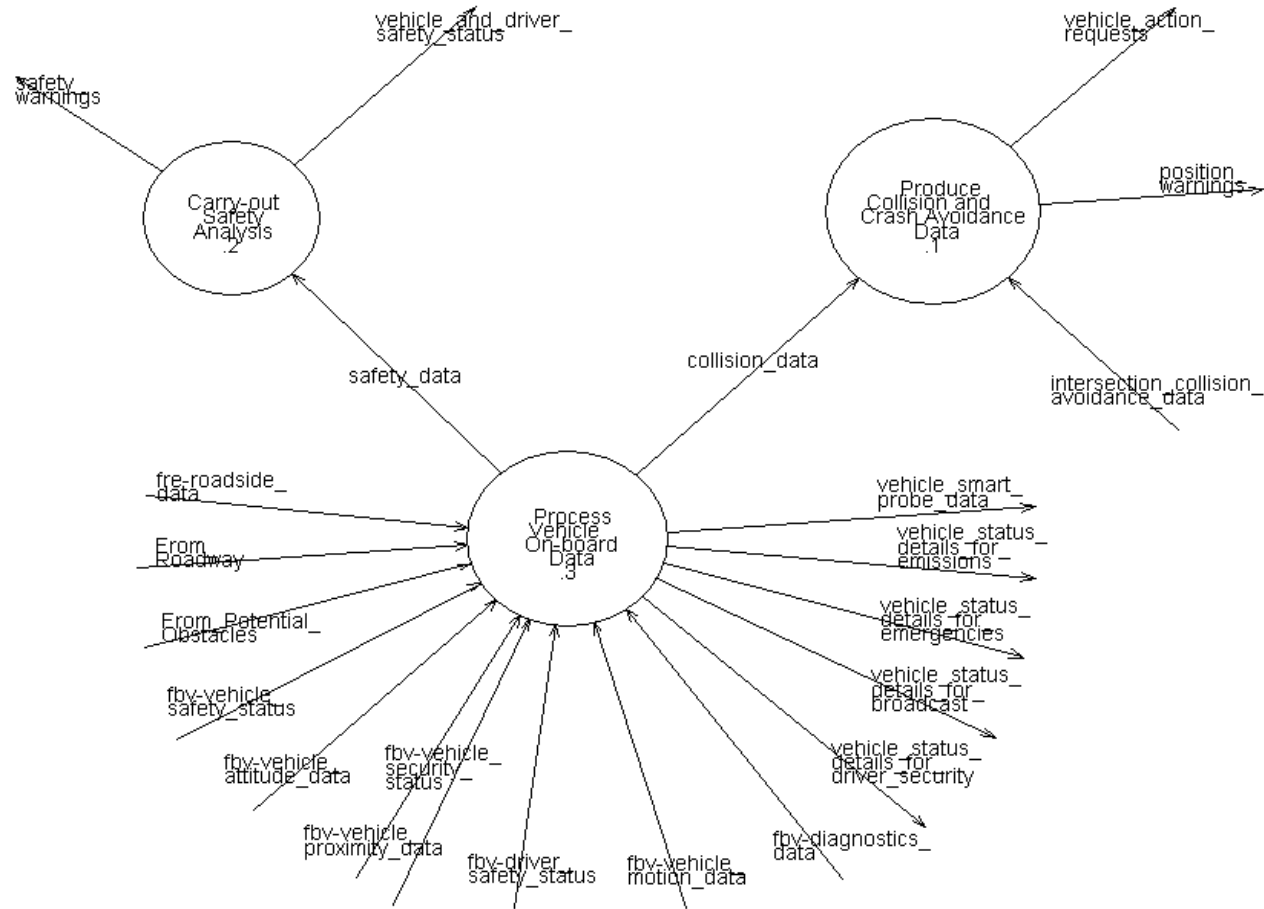
Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



3-Provide Vehicle Monitoring and Control

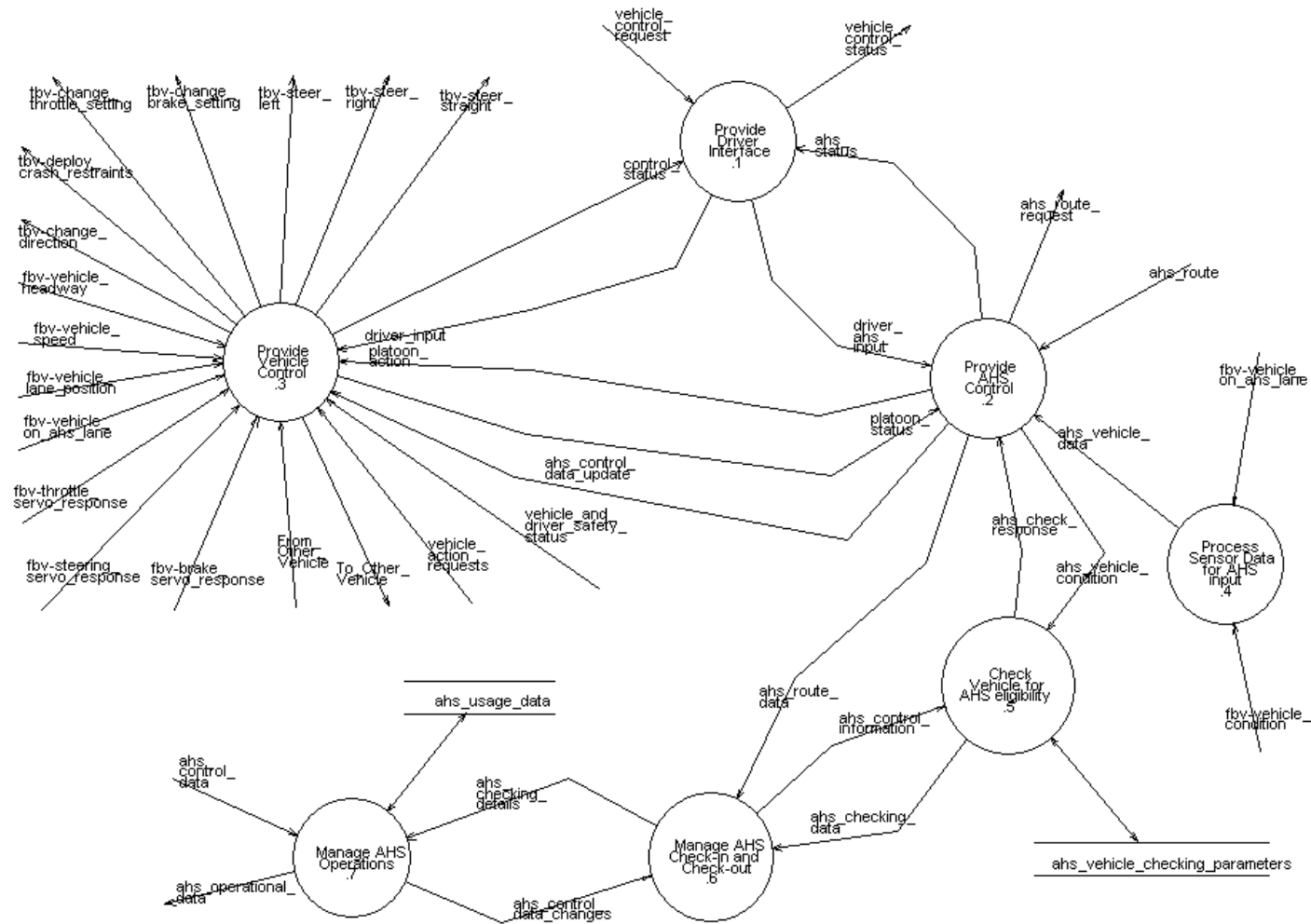


3.1-Monitor Vehicle Status

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

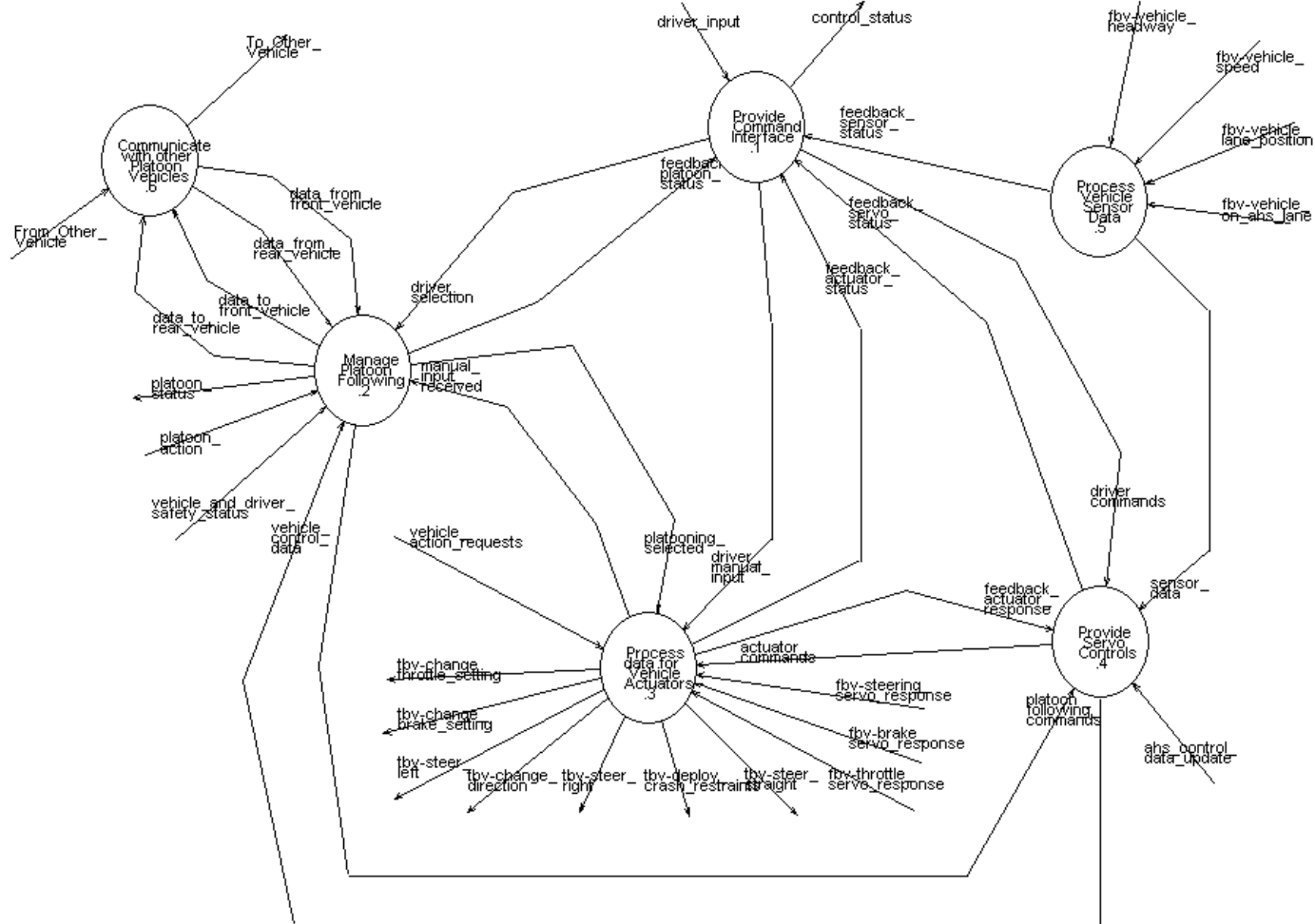


3.2-Provide Automatic Vehicle Operation

## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

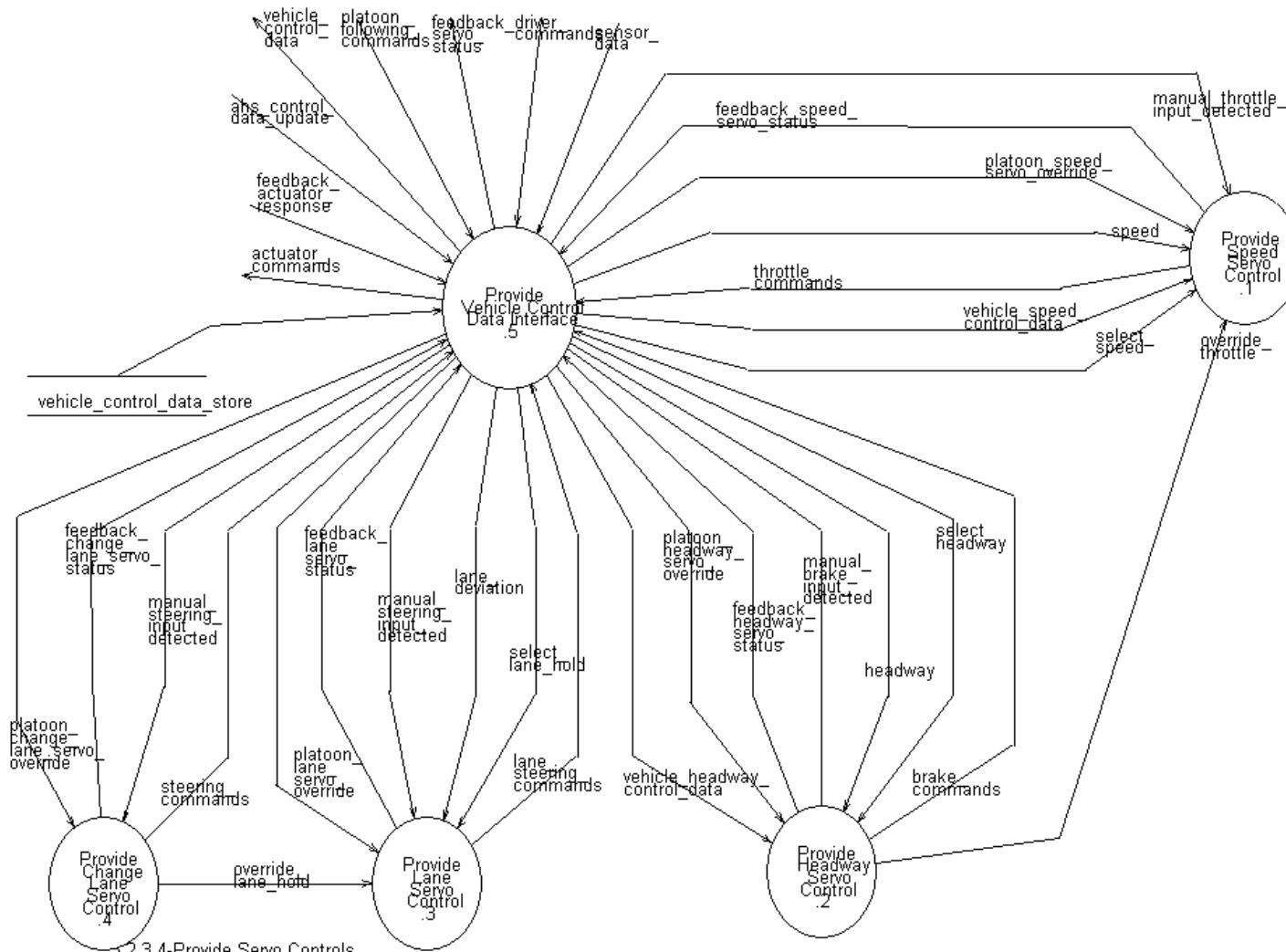


3.2.3-Provide Vehicle Control

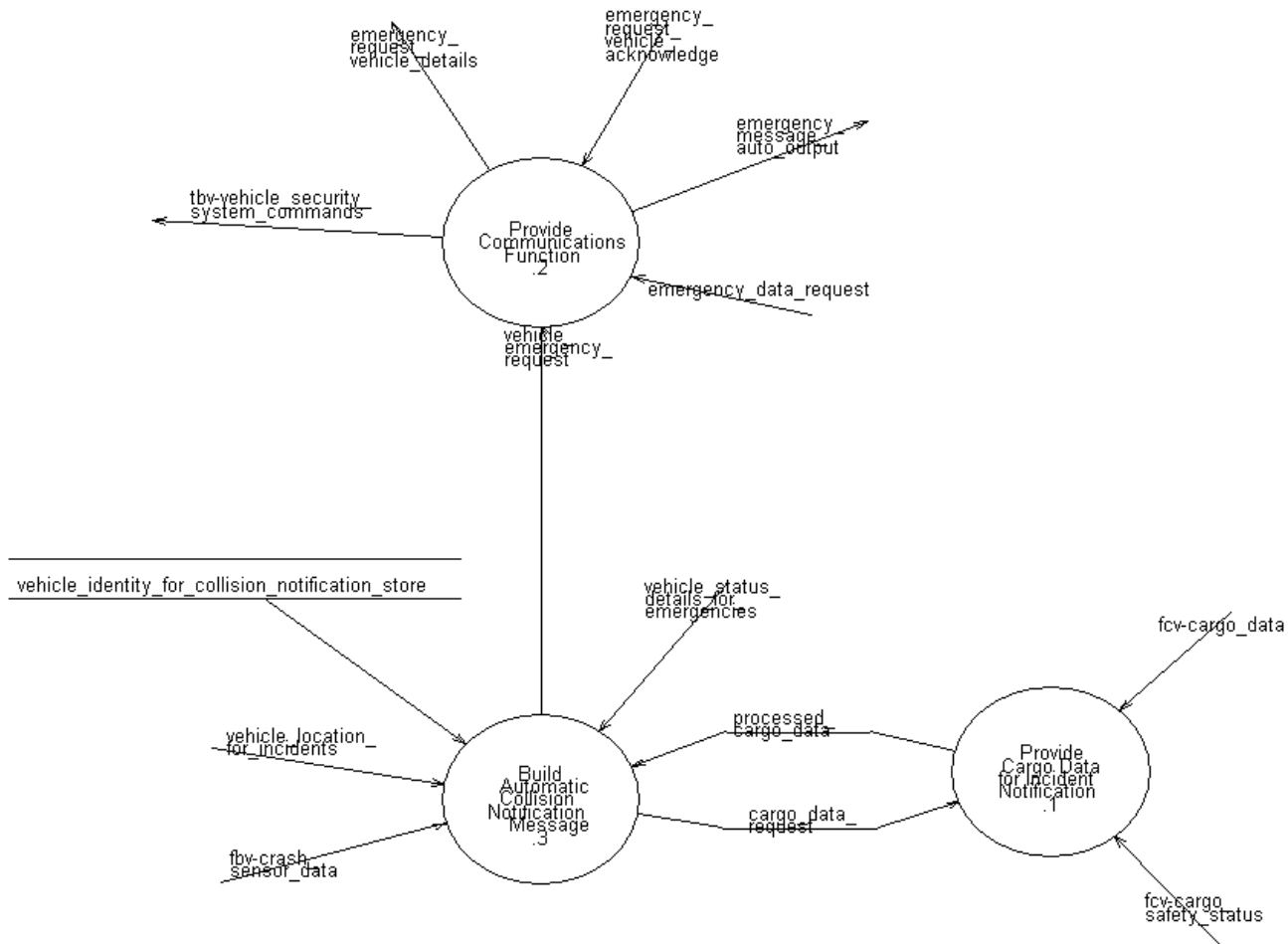
Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



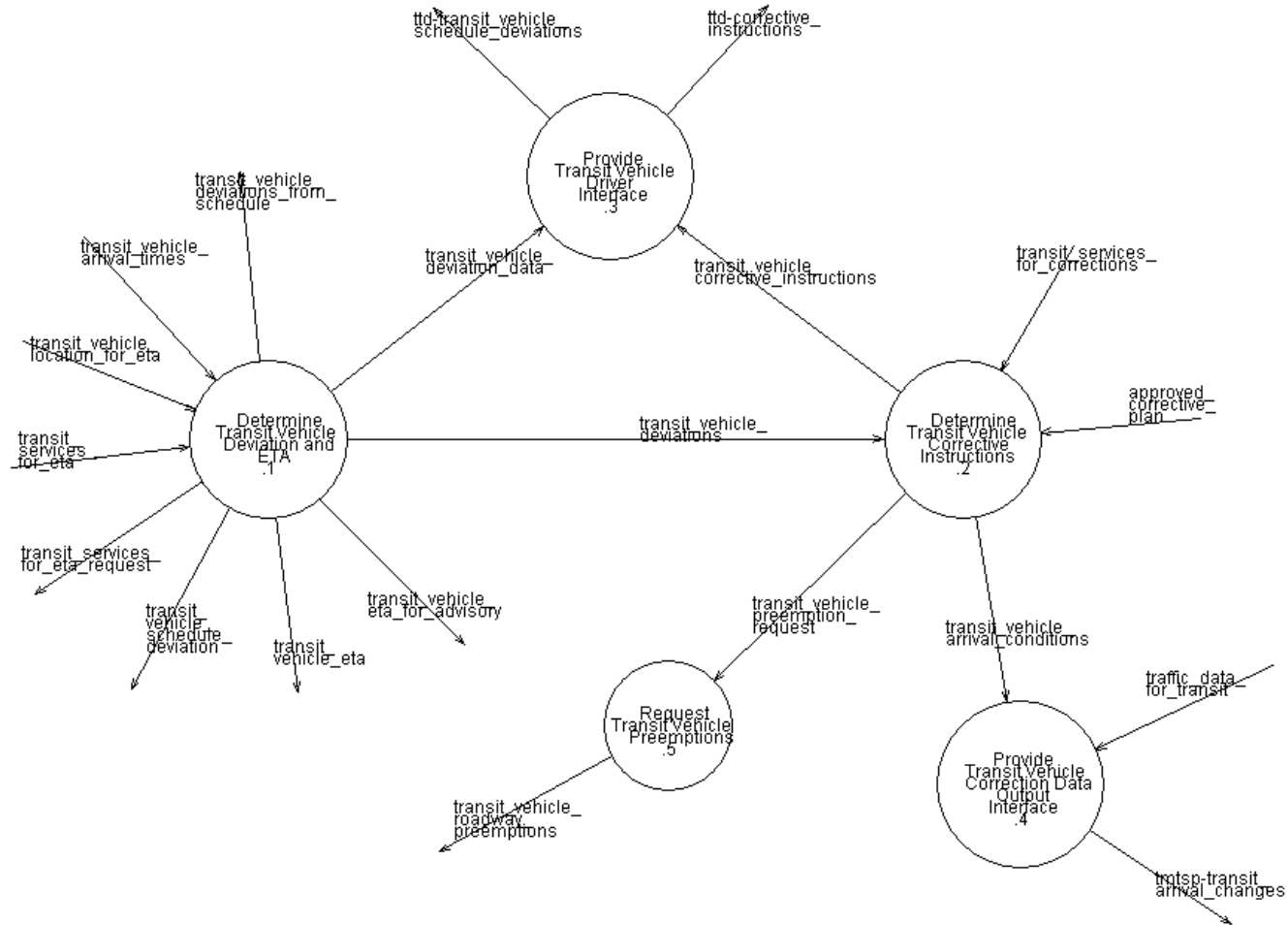
5.2.3.4-Provide Servo Controls



3.3-Provide Automatic Emergency Notification

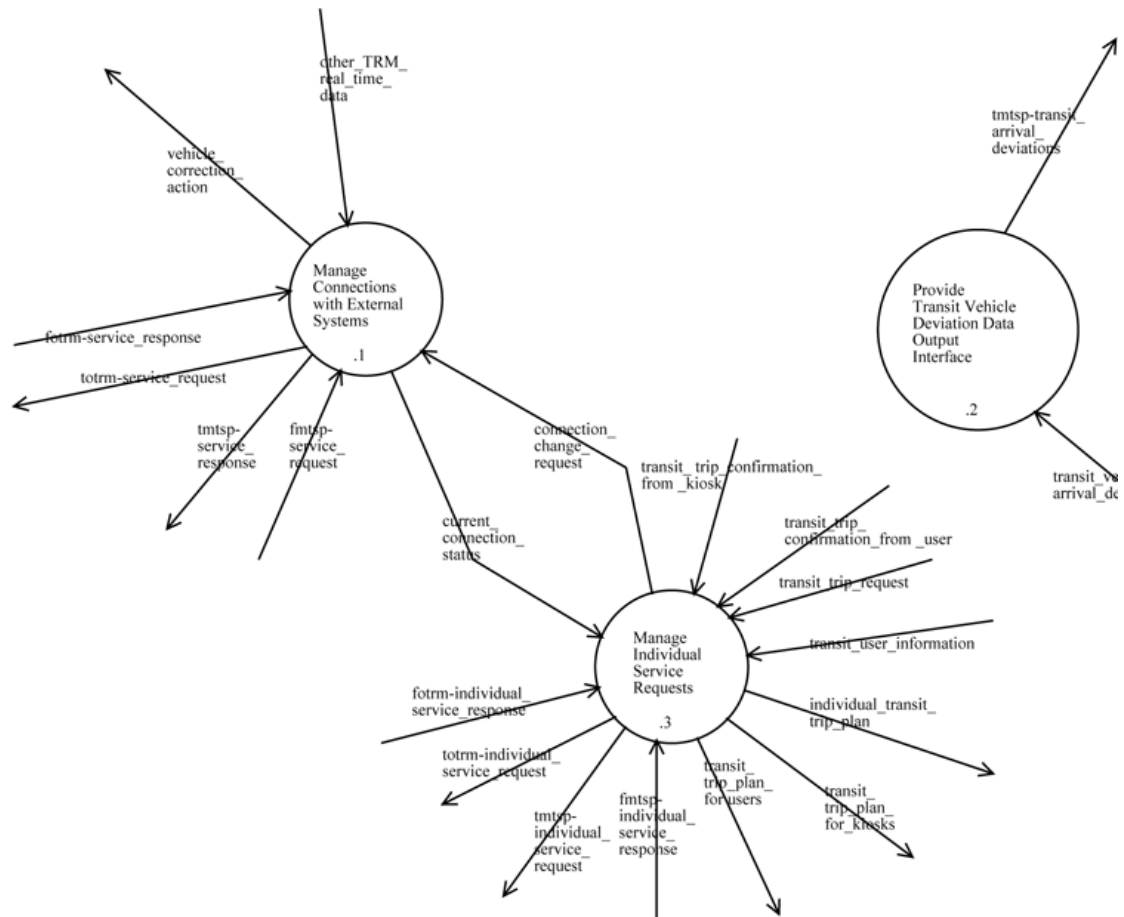






4.1.2-Determine Transit Vehicle Deviation and Corrections

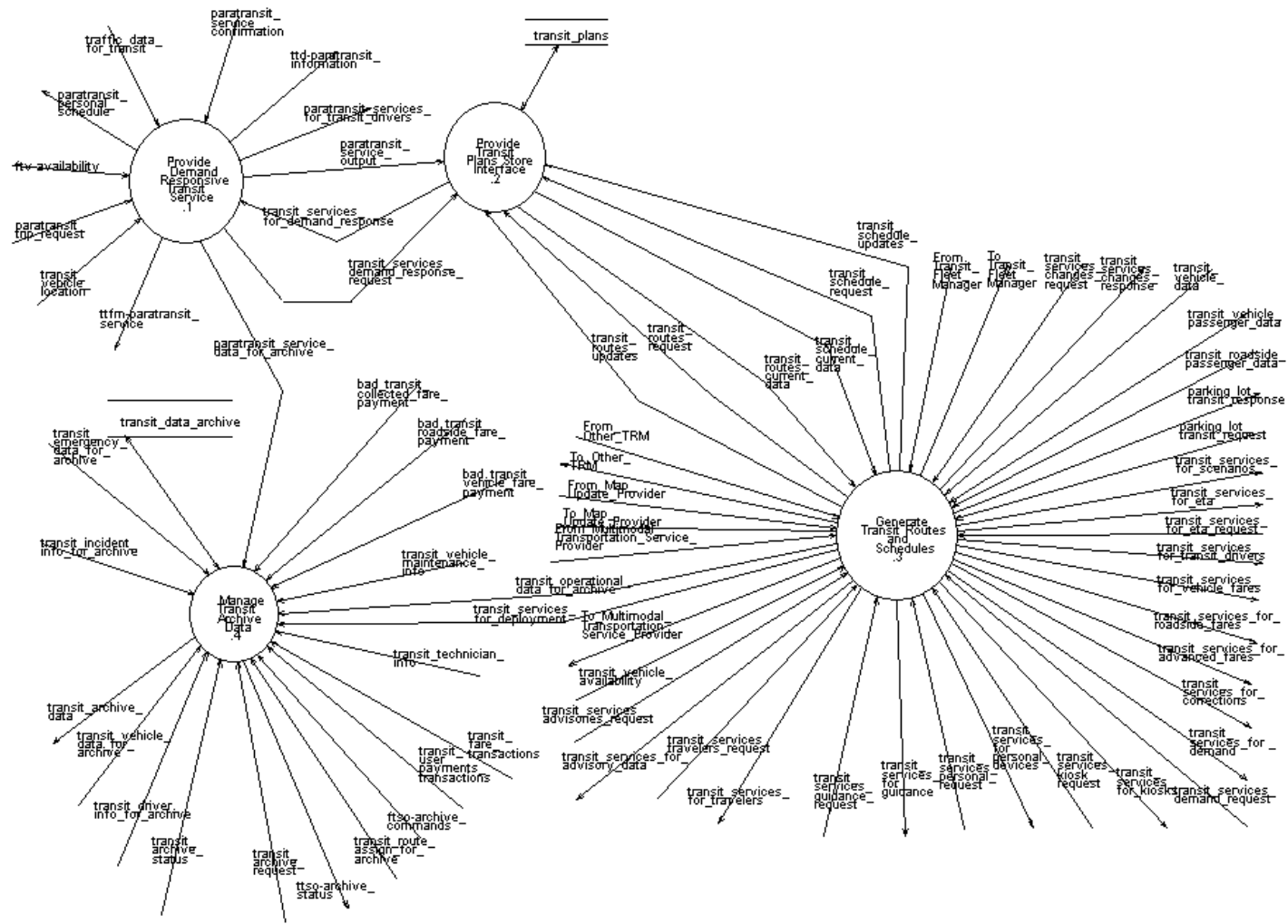
4.1.7:3  
Manage Connection Protection



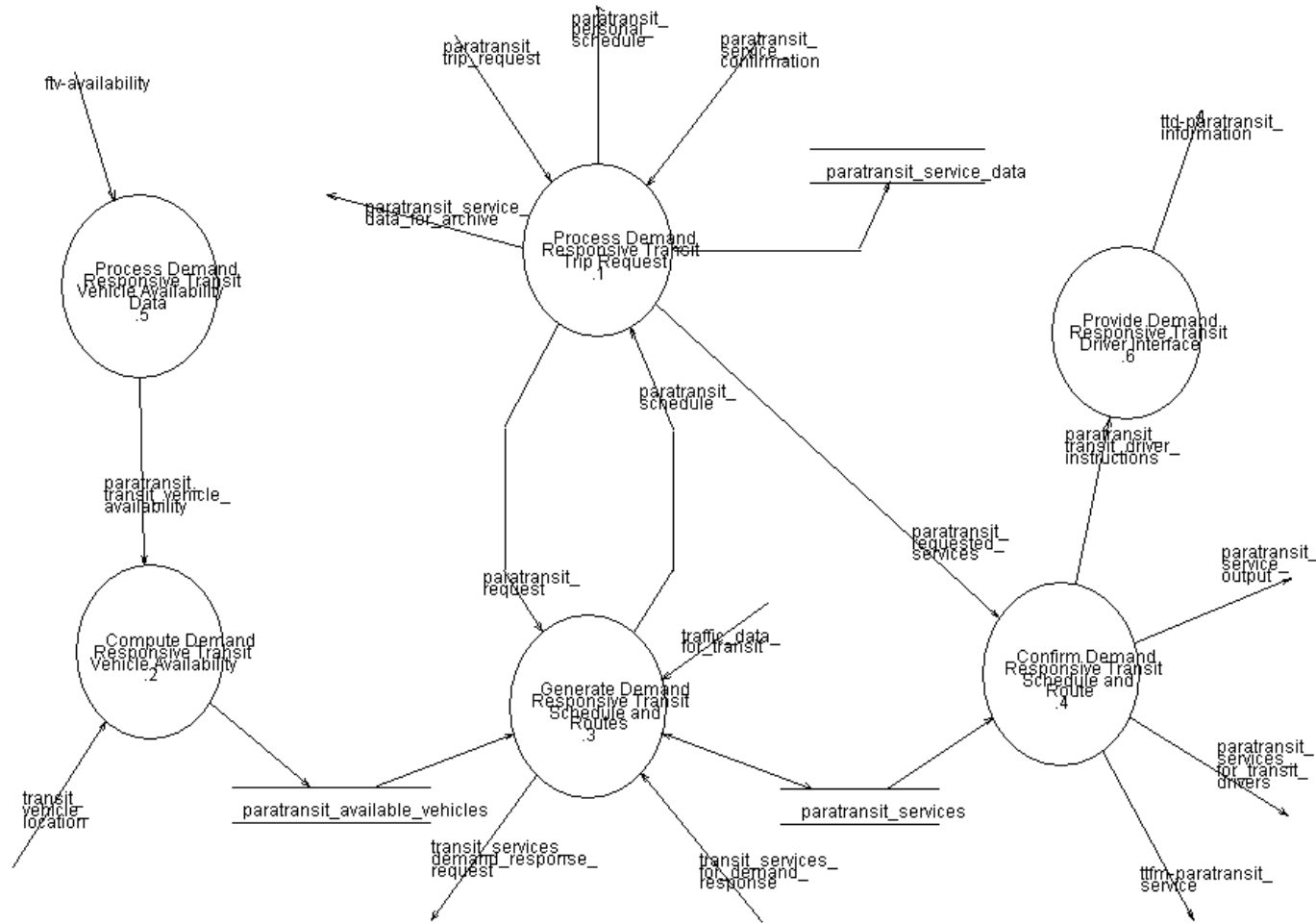
## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



4.2-Plan and Schedule Transit Services

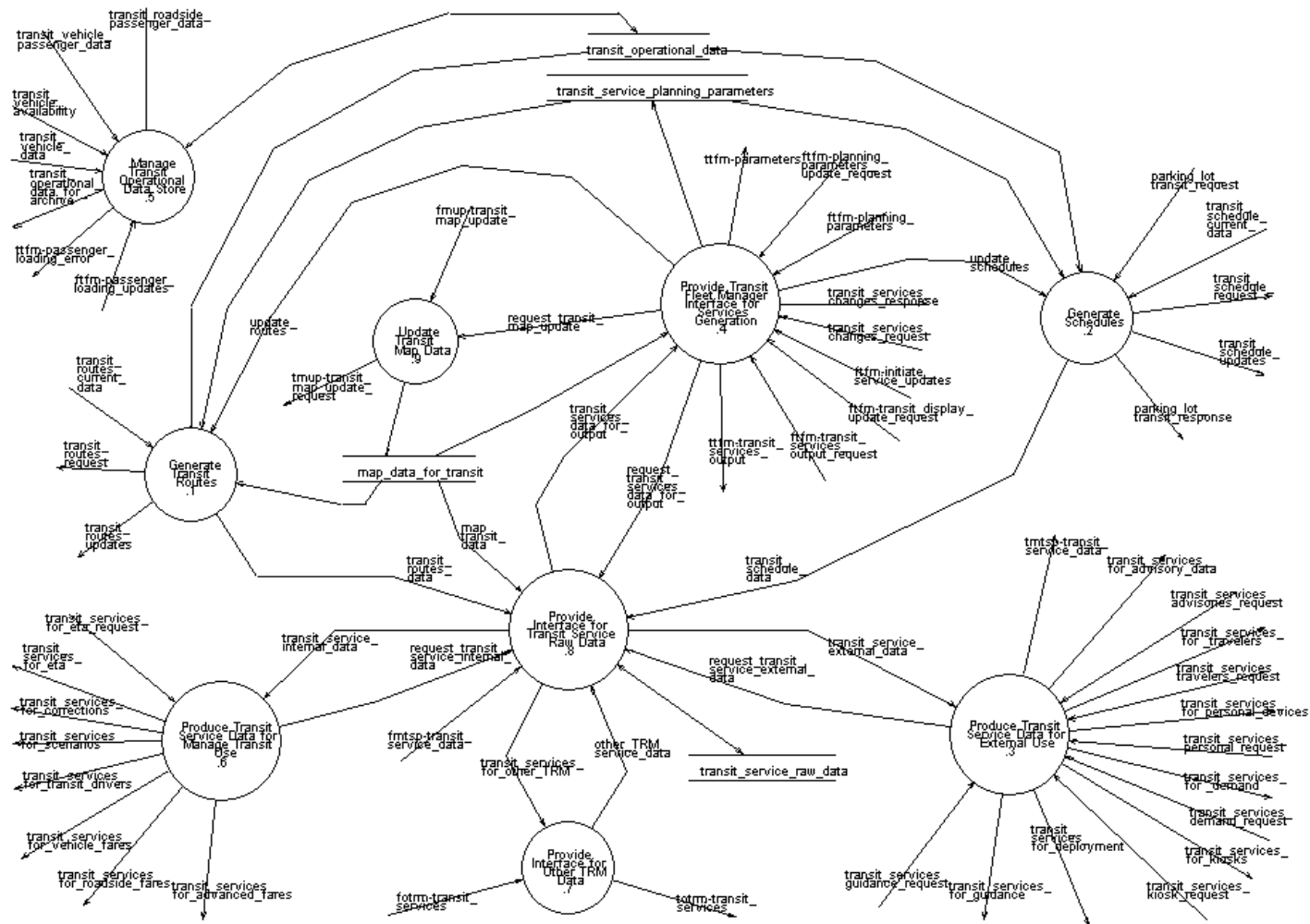


4.2.1-Provide Demand Responsive Transit Service

## Development of ITS System Architecture for Malaysia

### Technical Note No. 2

#### Logical Architecture Framework - Volume I

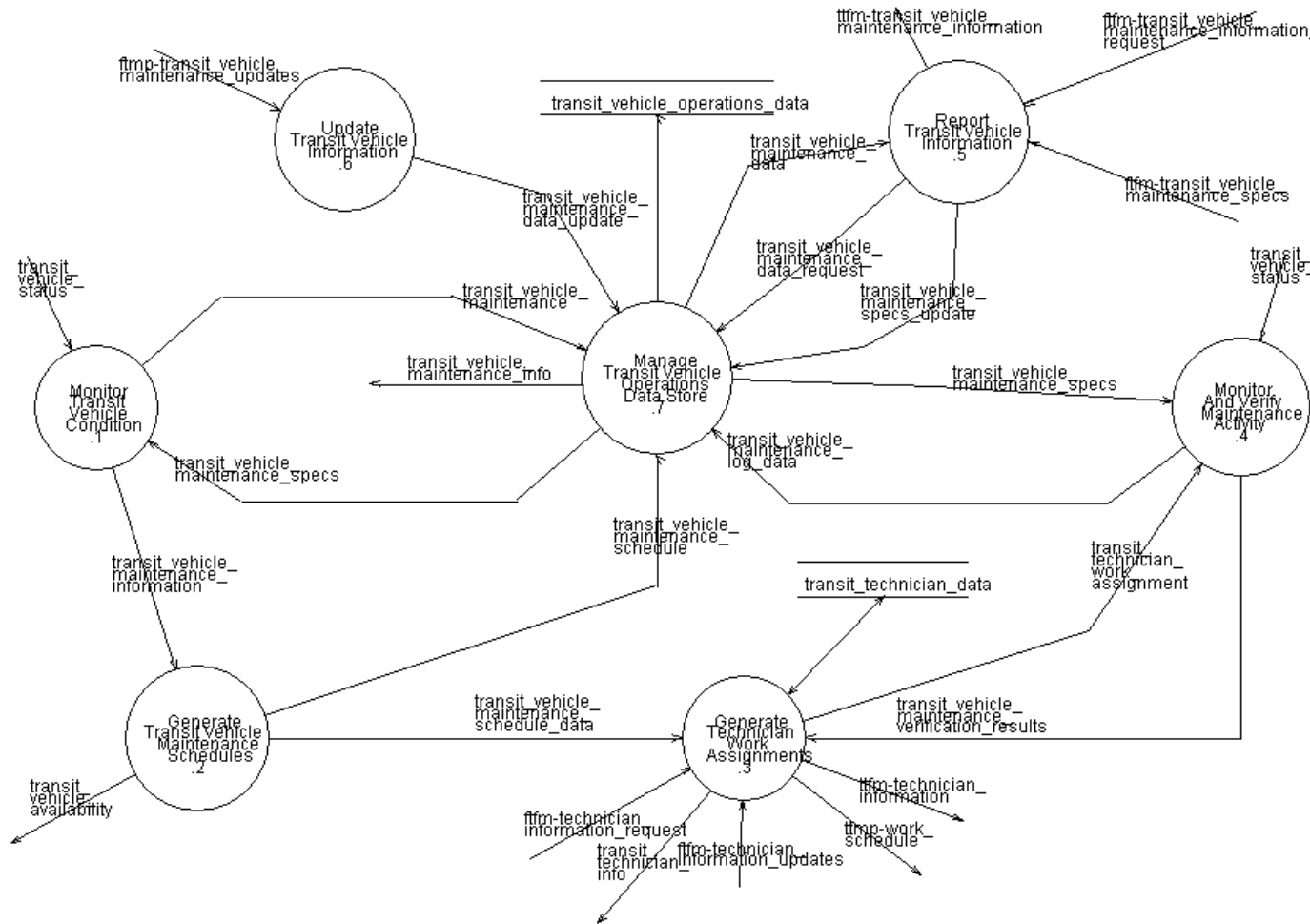


4.2.3-Generate Transit Routes and Schedules

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

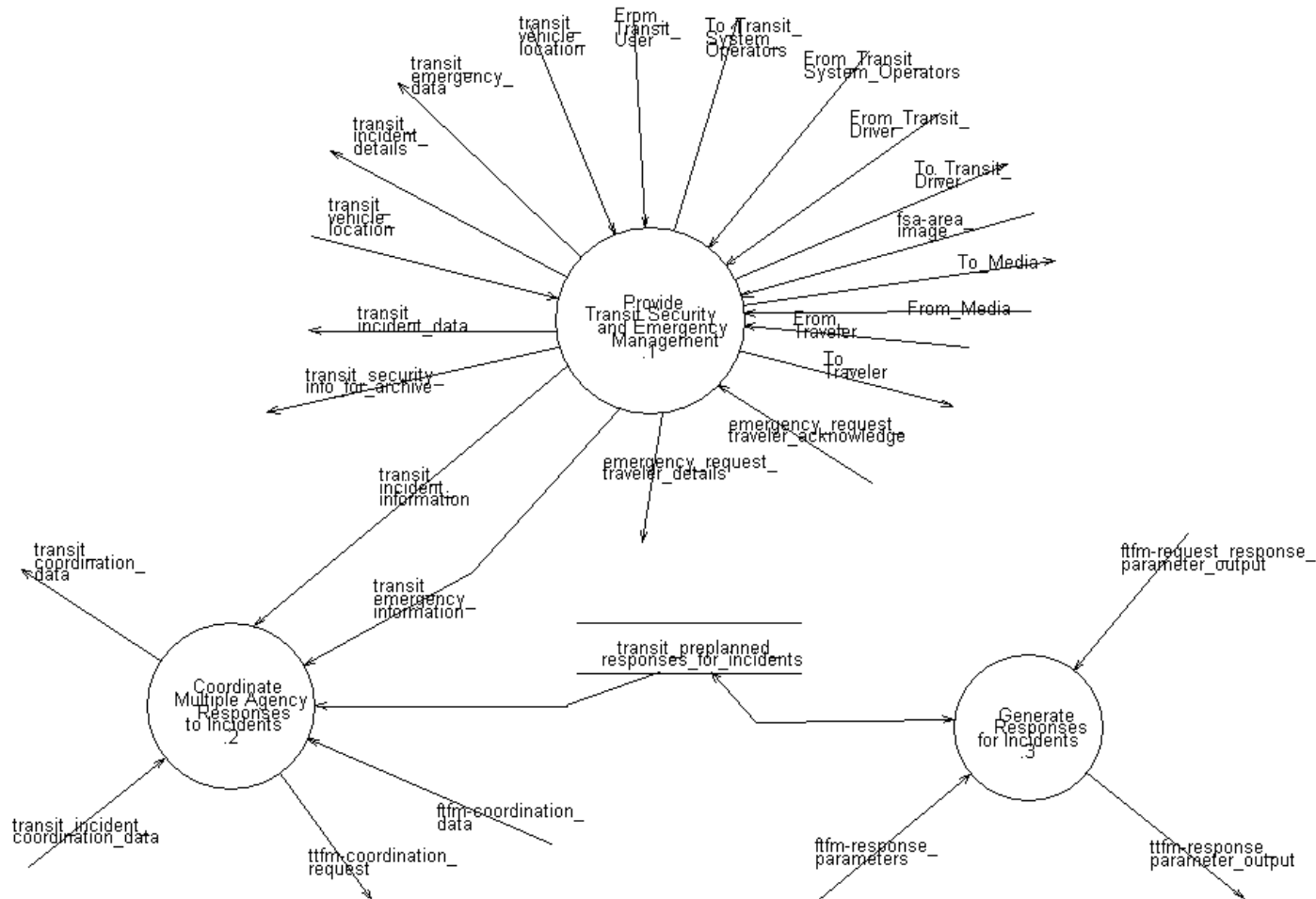


4.3-Schedule Transit Vehicle Maintenance

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

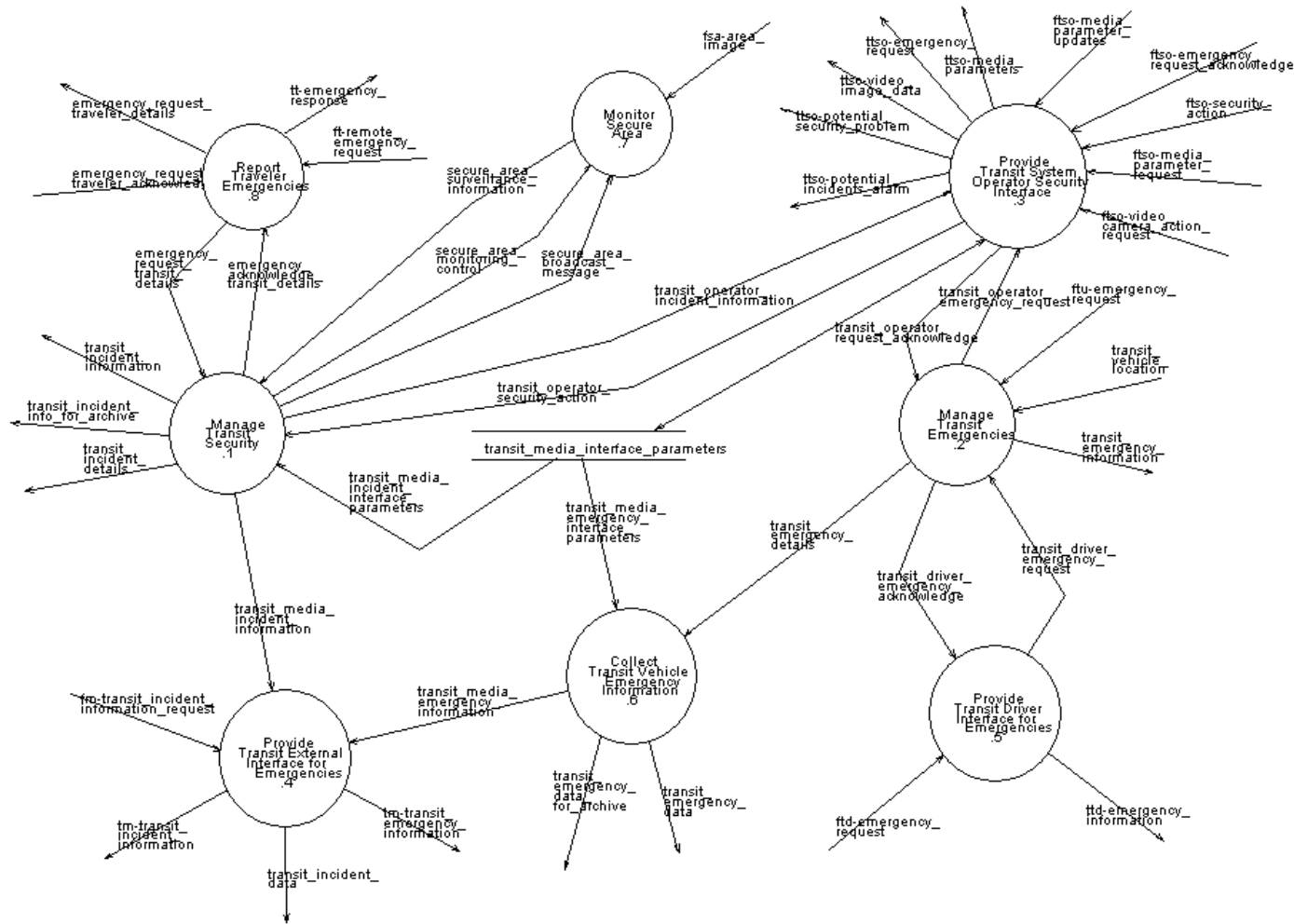


4.4-Support Security and Coordination

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

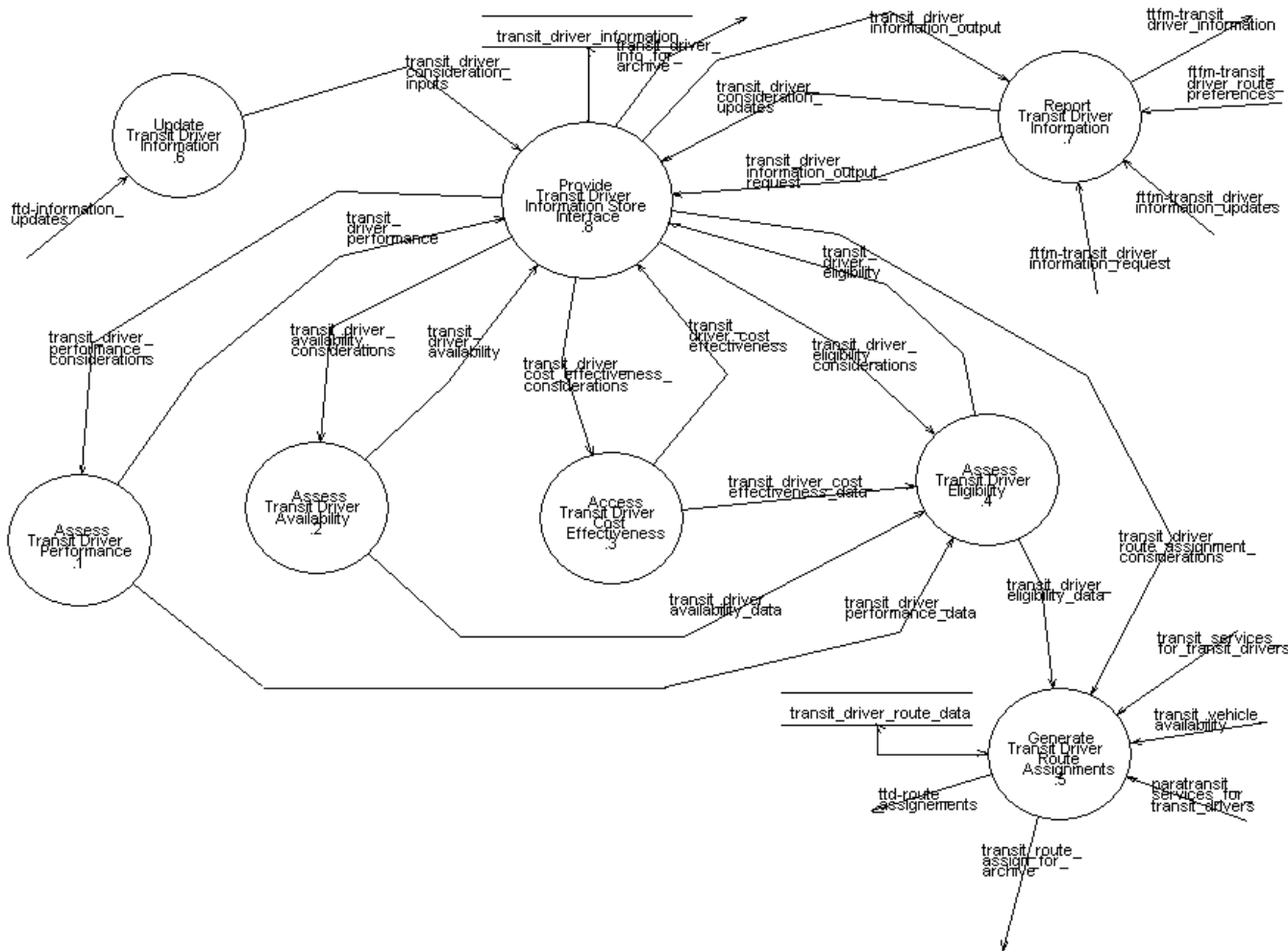


4.4.1-Provide Transit Security and Emergency Management

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

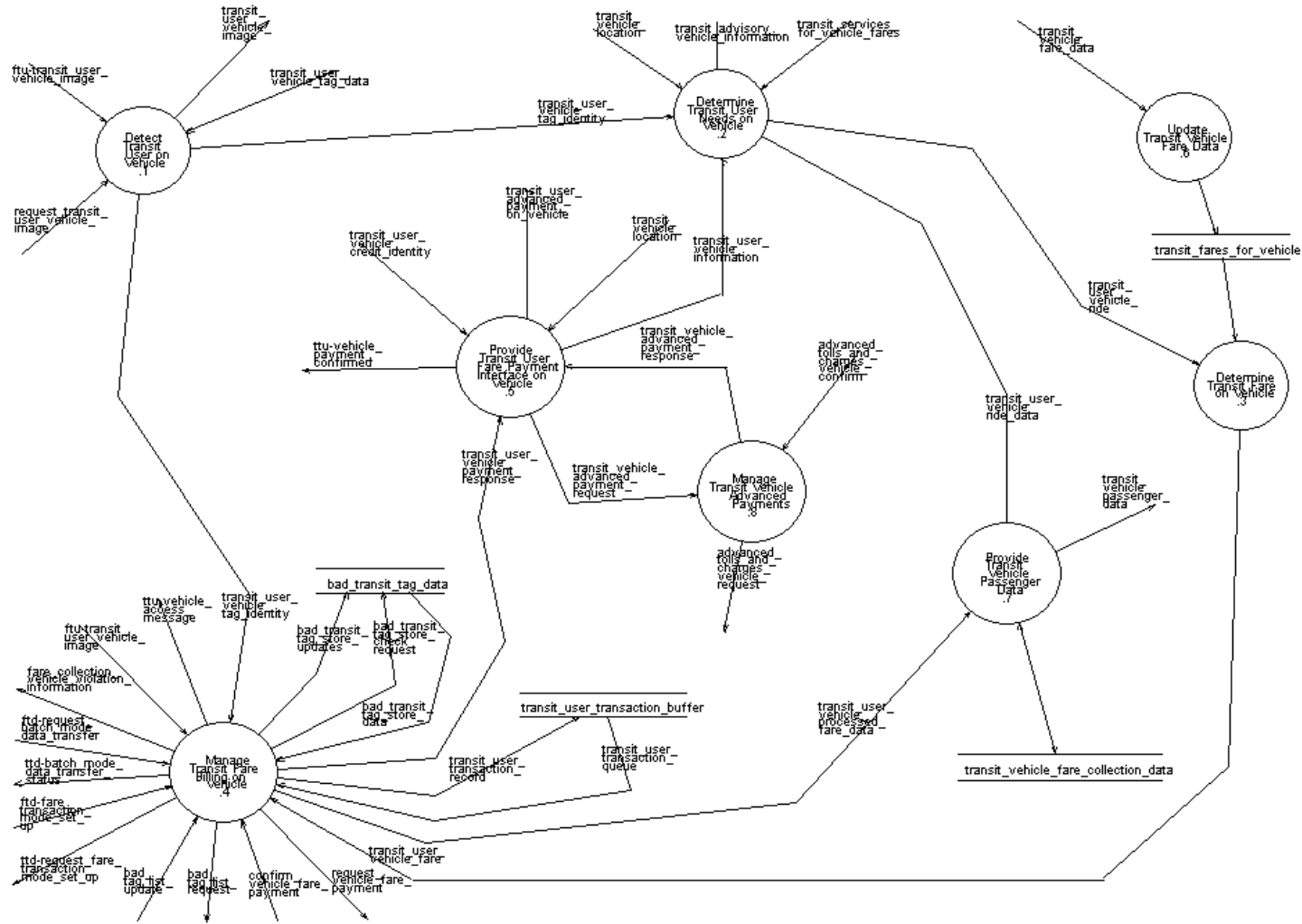


4.5-Generate Transit Driver Schedules

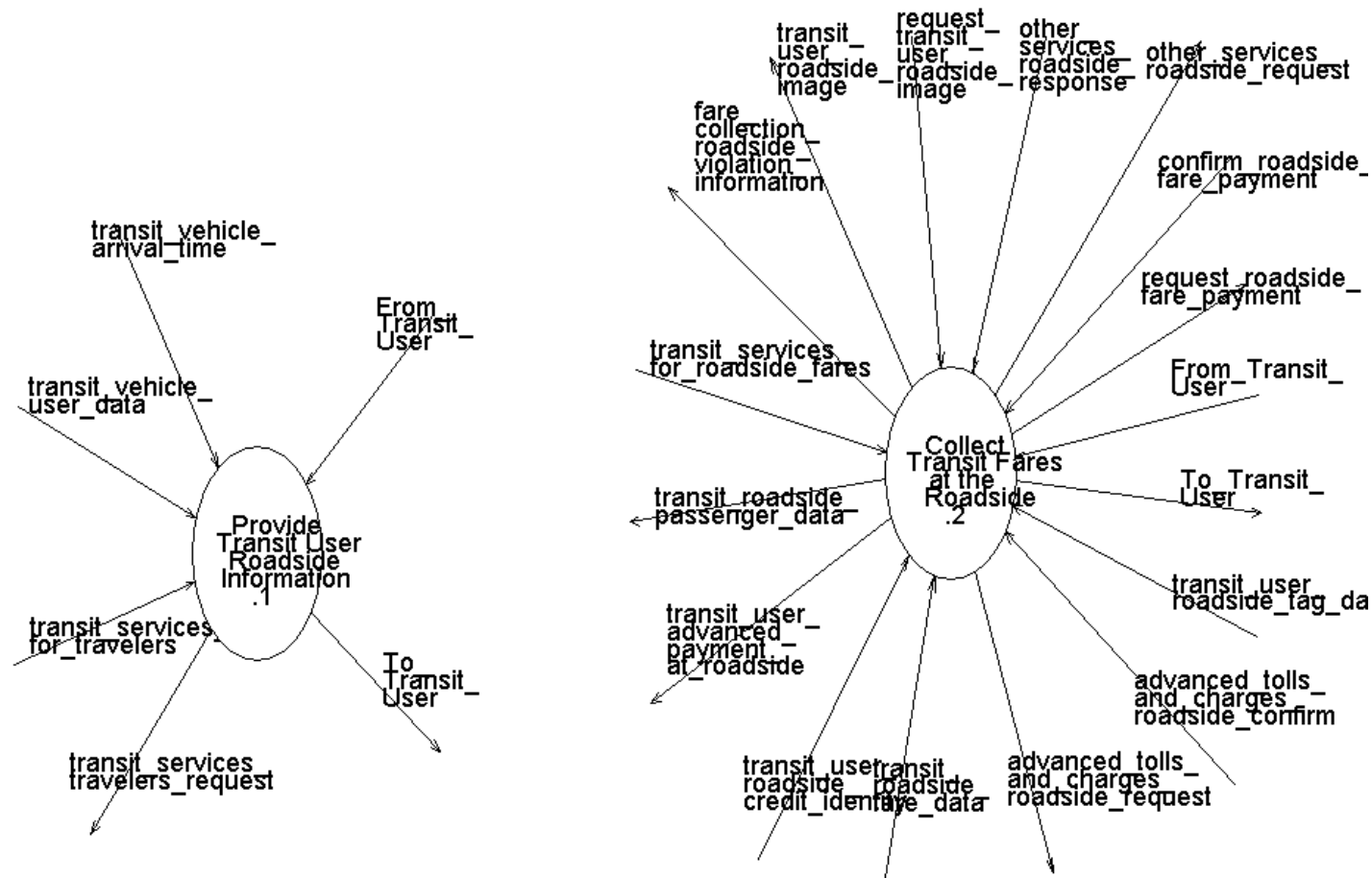
Development of ITS System Architecture for Malaysia

Technical Note No. 2

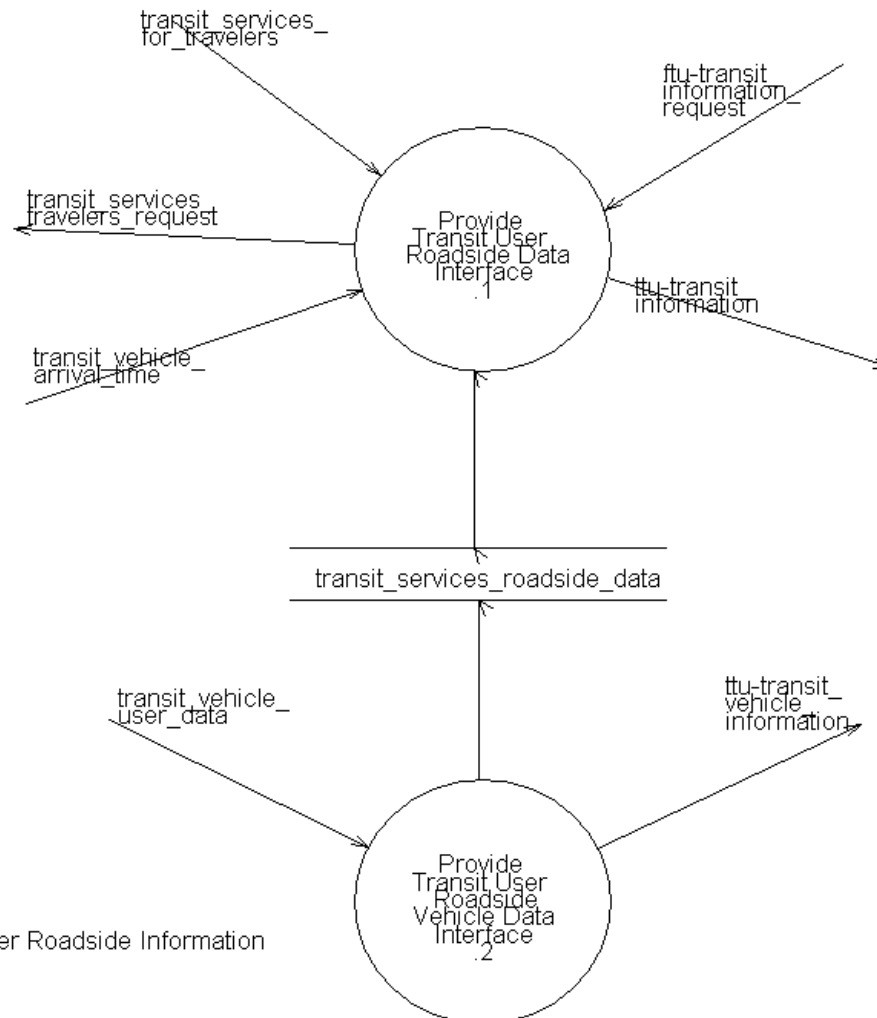
Logical Architecture Framework - Volume I



4.6-Collect Transit Fares in the Vehicle



4.7-Provide Transit User Roadside Facilities

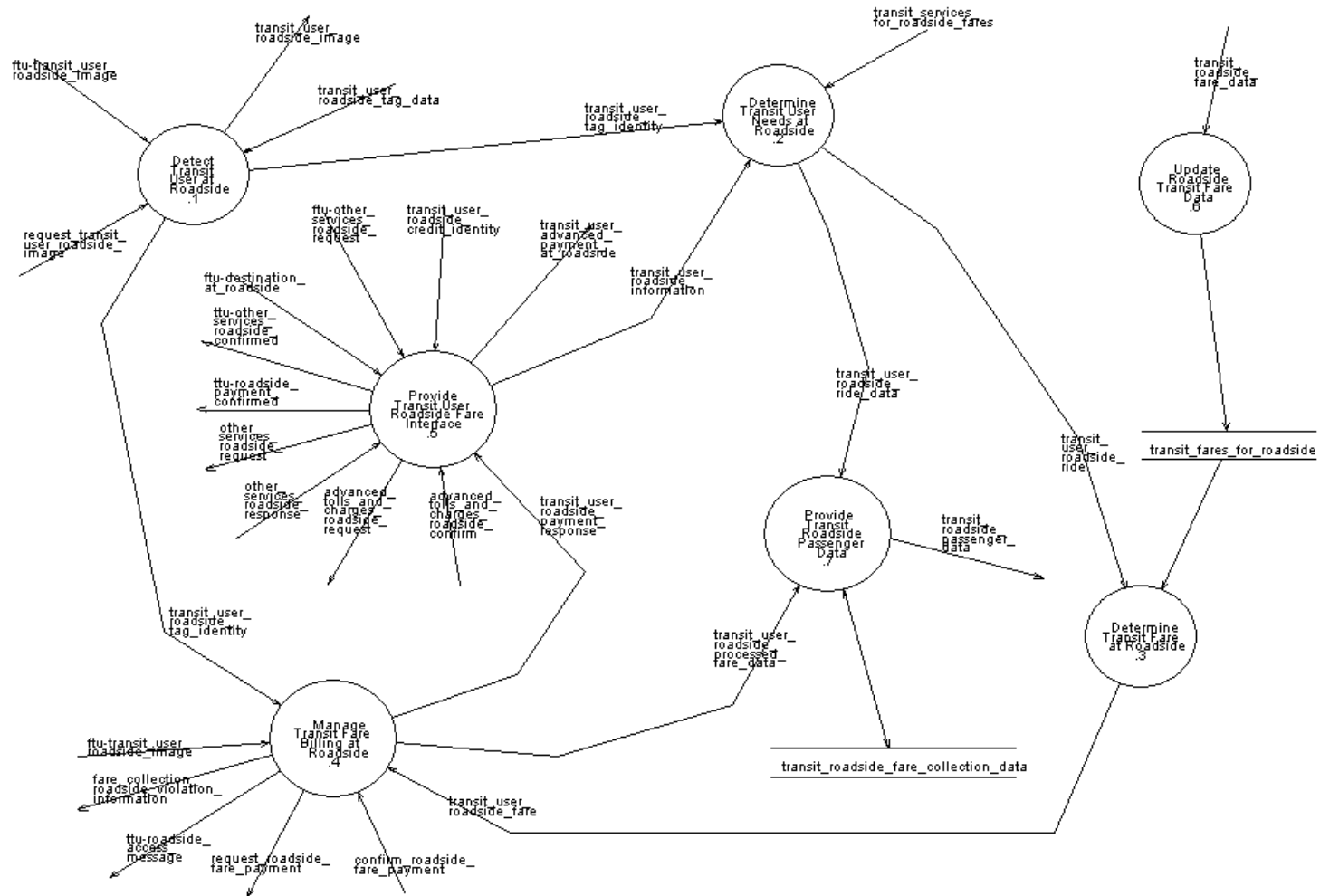


4.7.1-Provide Transit User Roadside Information

## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



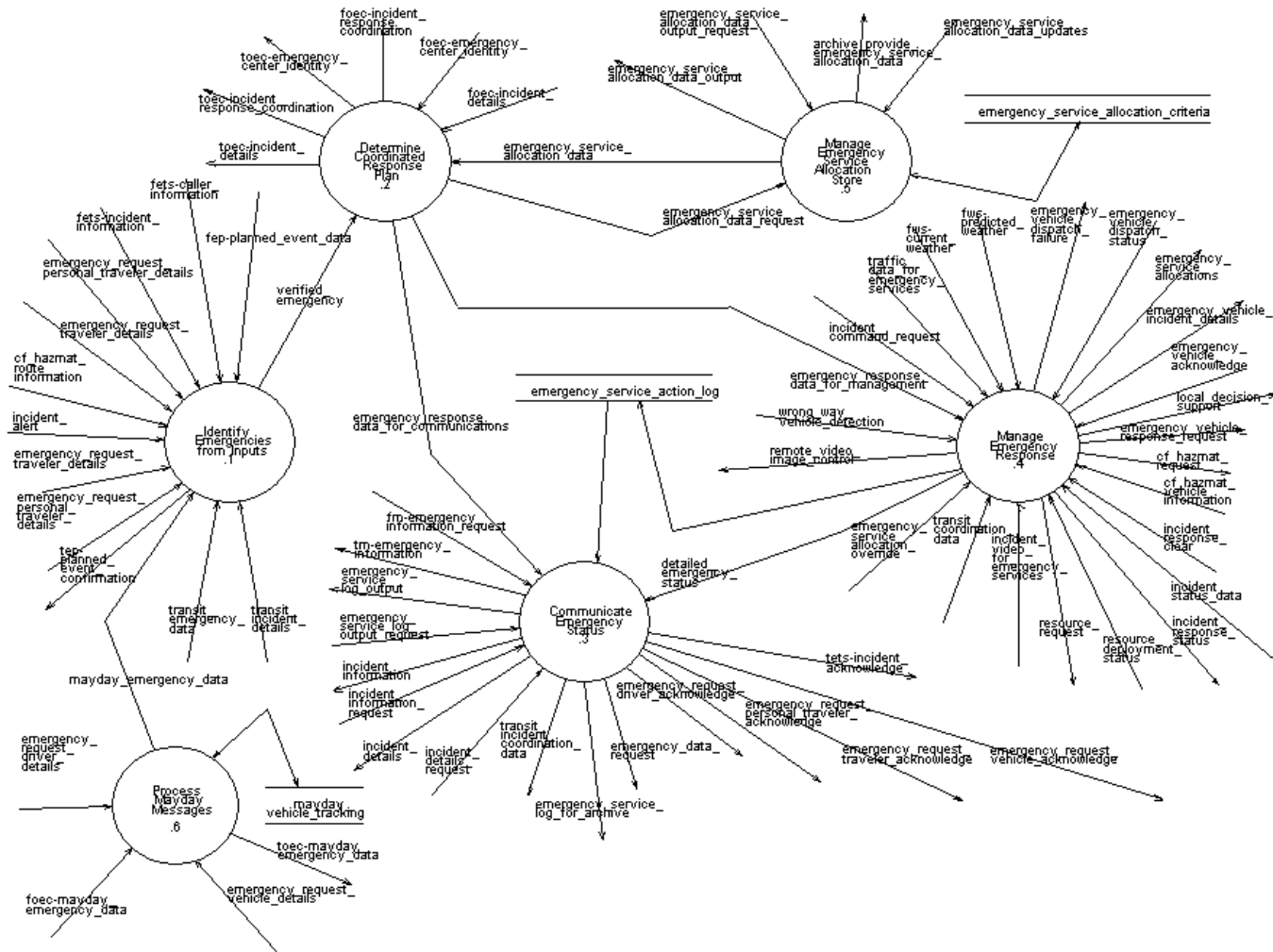
4.7.2-Collect Transit Fares at the Roadside



Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

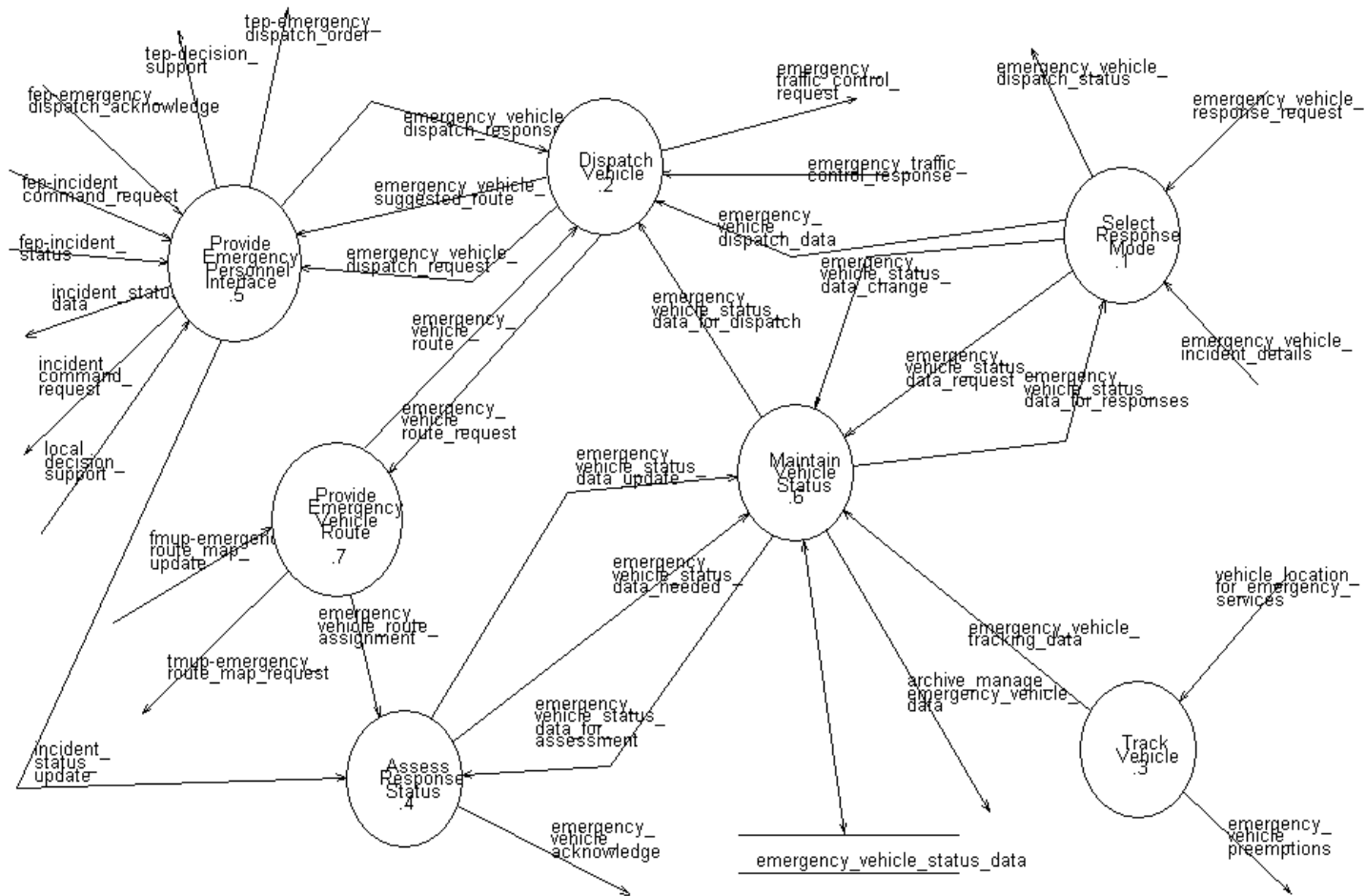


5.1-Provide Emergency Service Allocation

## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

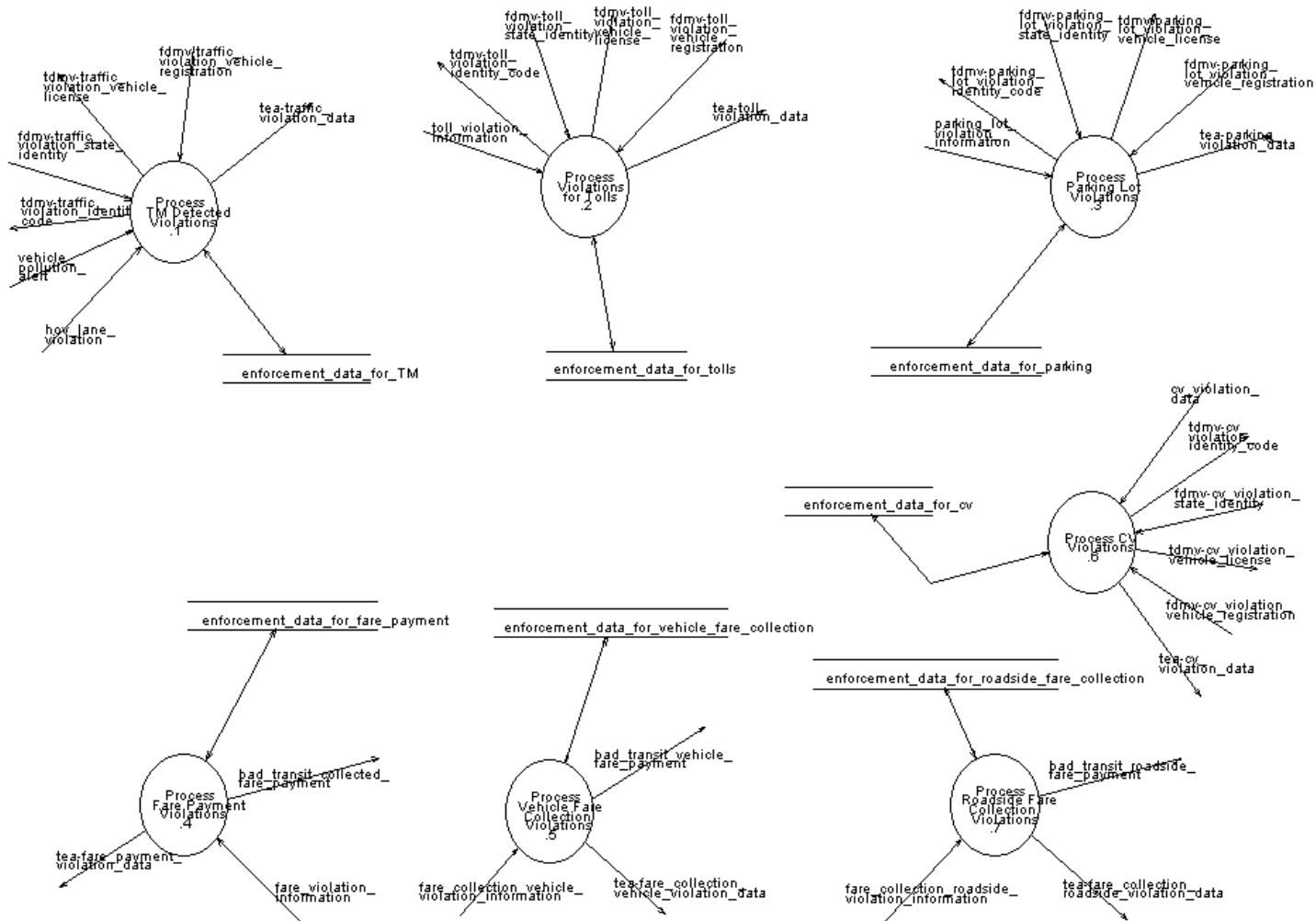


5.3-Manage Emergency Vehicles

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

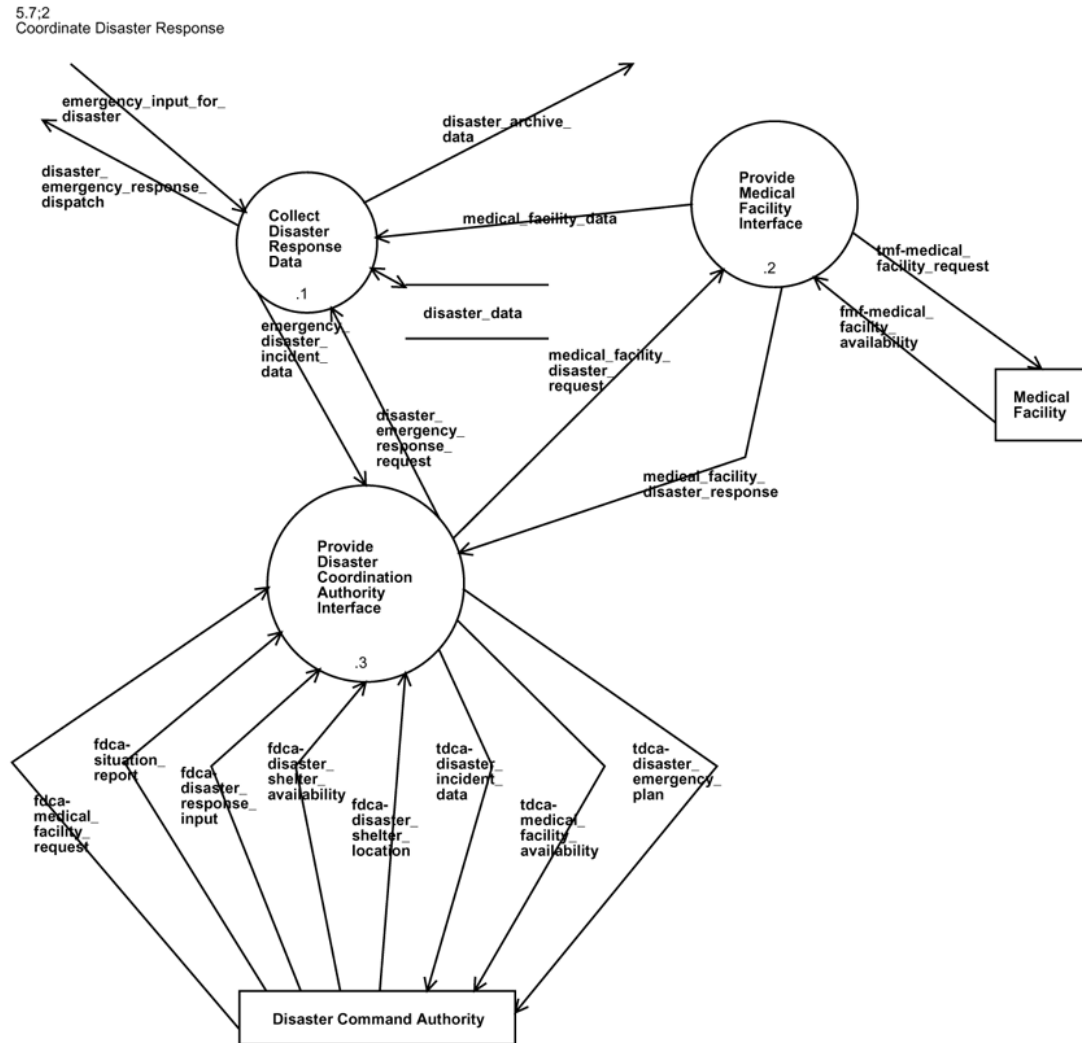


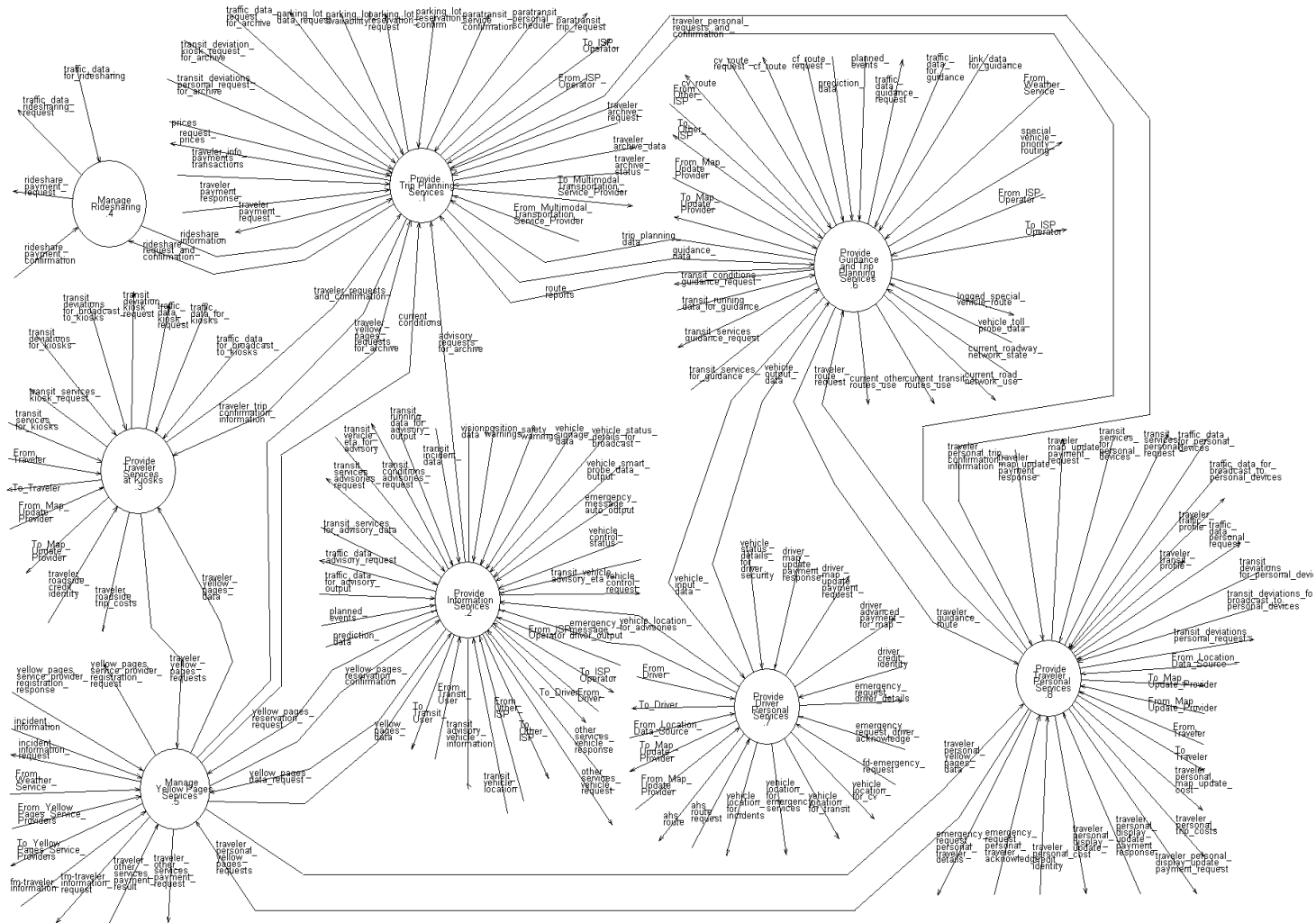
5.4-Provide Law Enforcement Allocation

## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



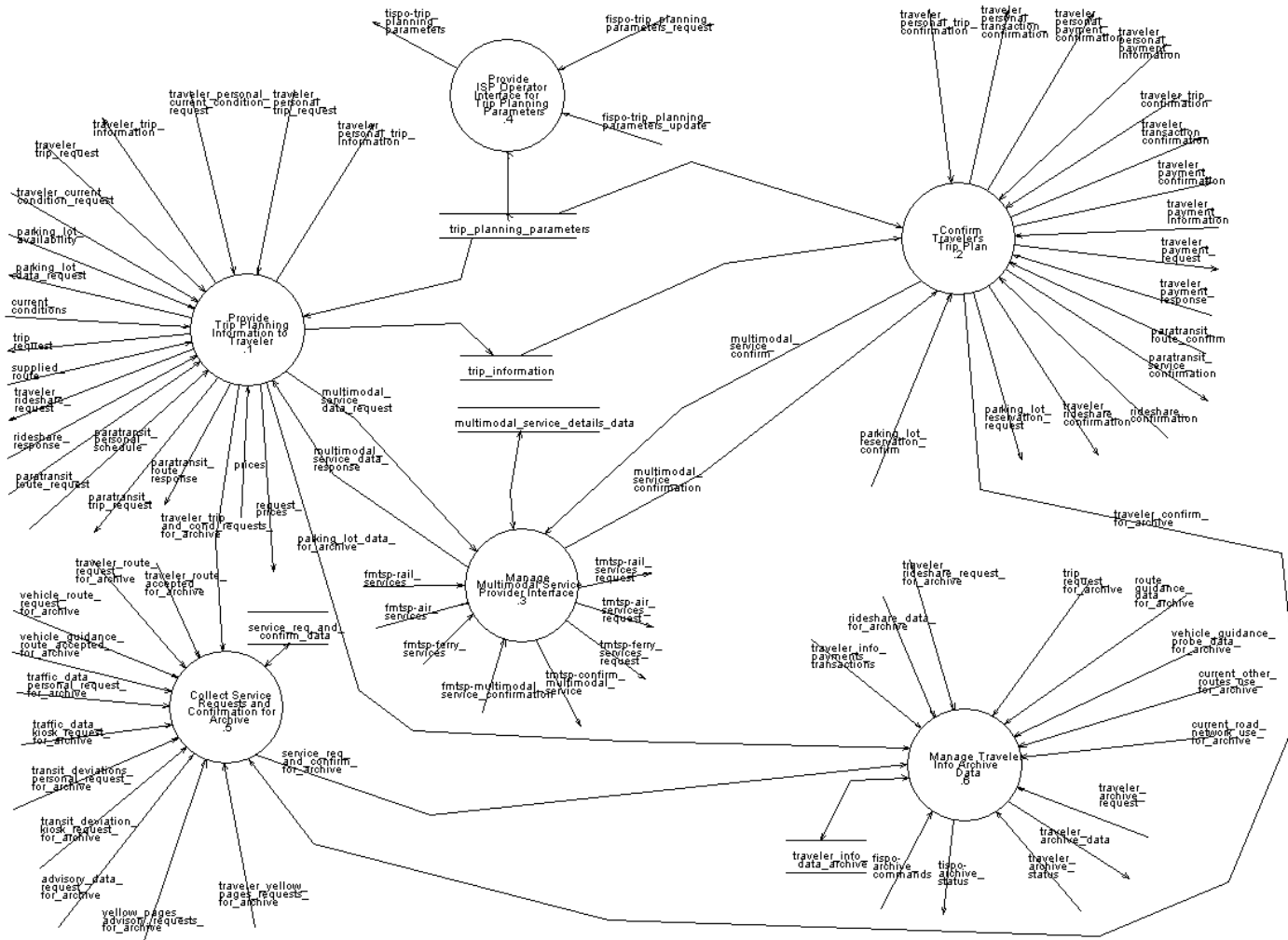


6-Provide Driver and Traveler Services

# Development of ITS System Architecture for Malaysia

## Technical Note No. 2

### Logical Architecture Framework - Volume I

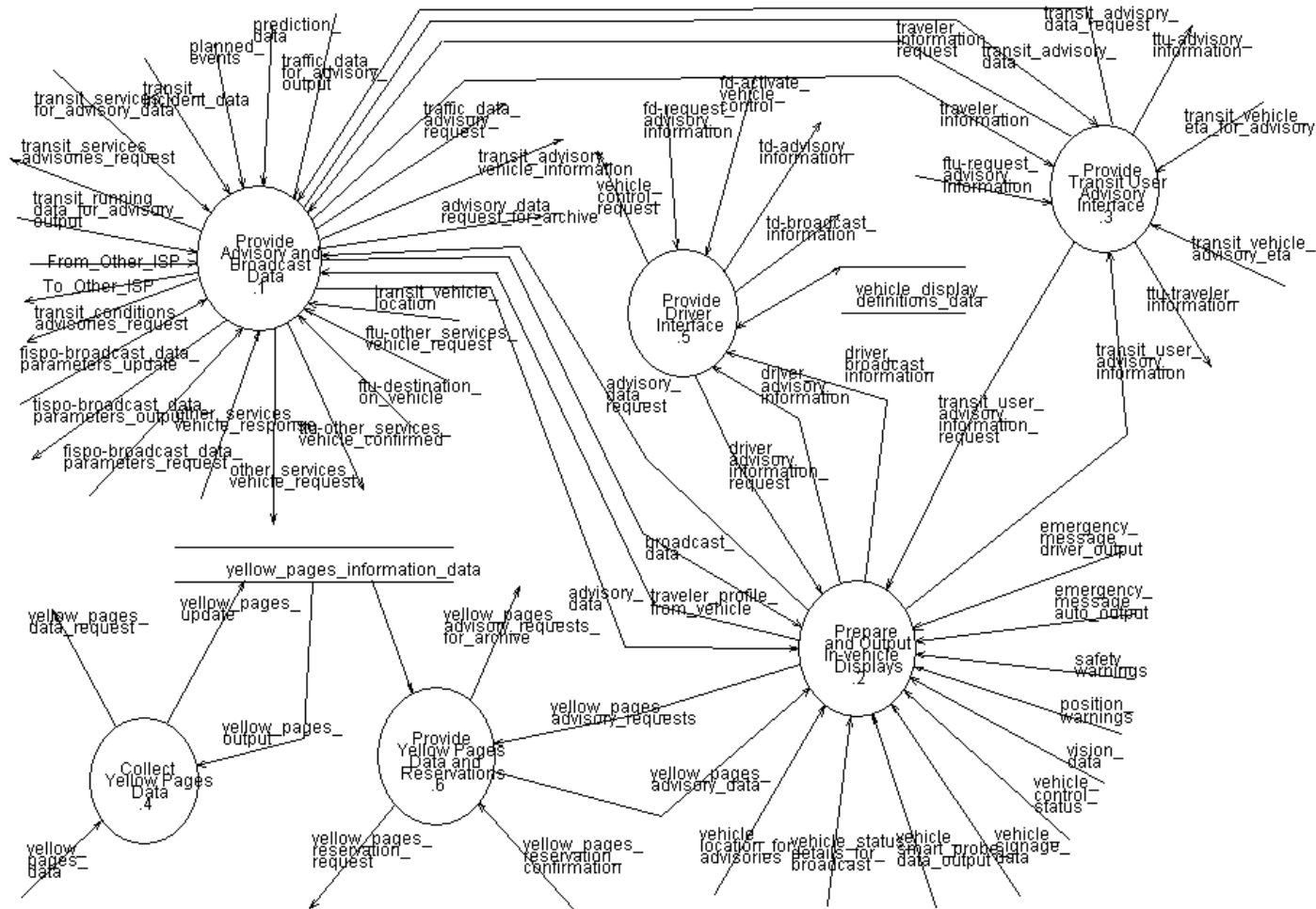


6.1-Provide Trip Planning Services

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

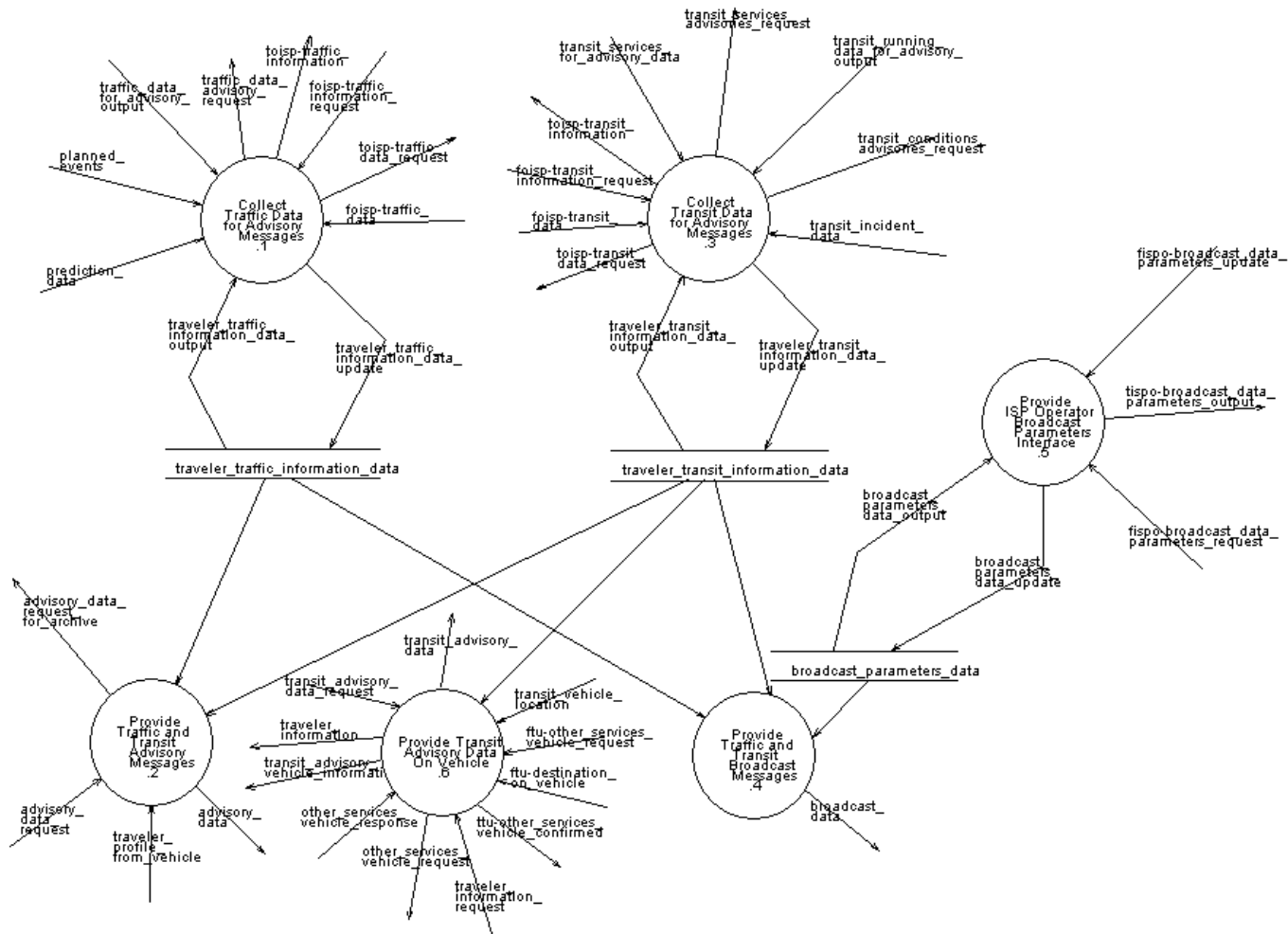


6.2-Provide Information Services

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

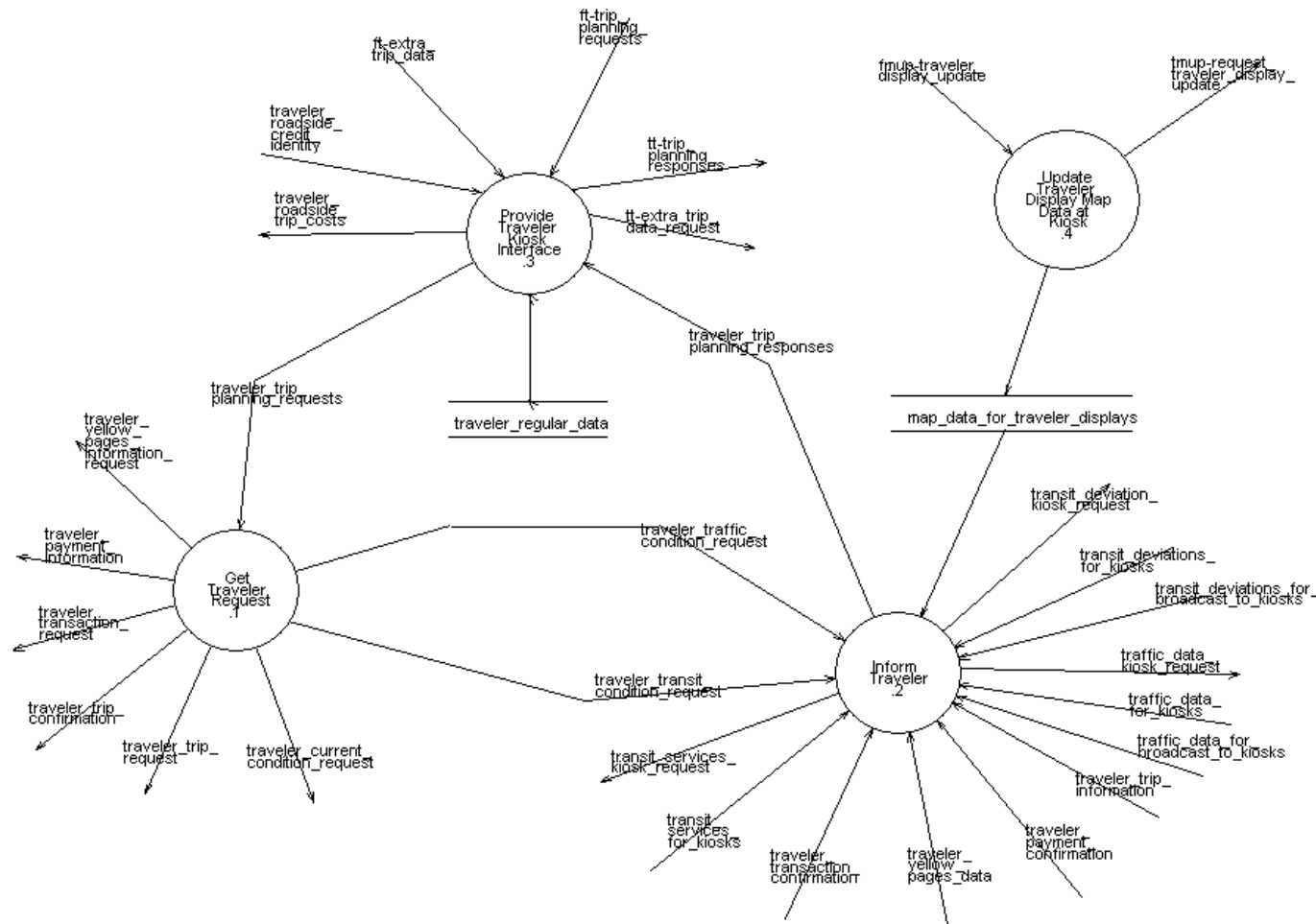


6.2.1-Provide Advisory and Broadcast Data

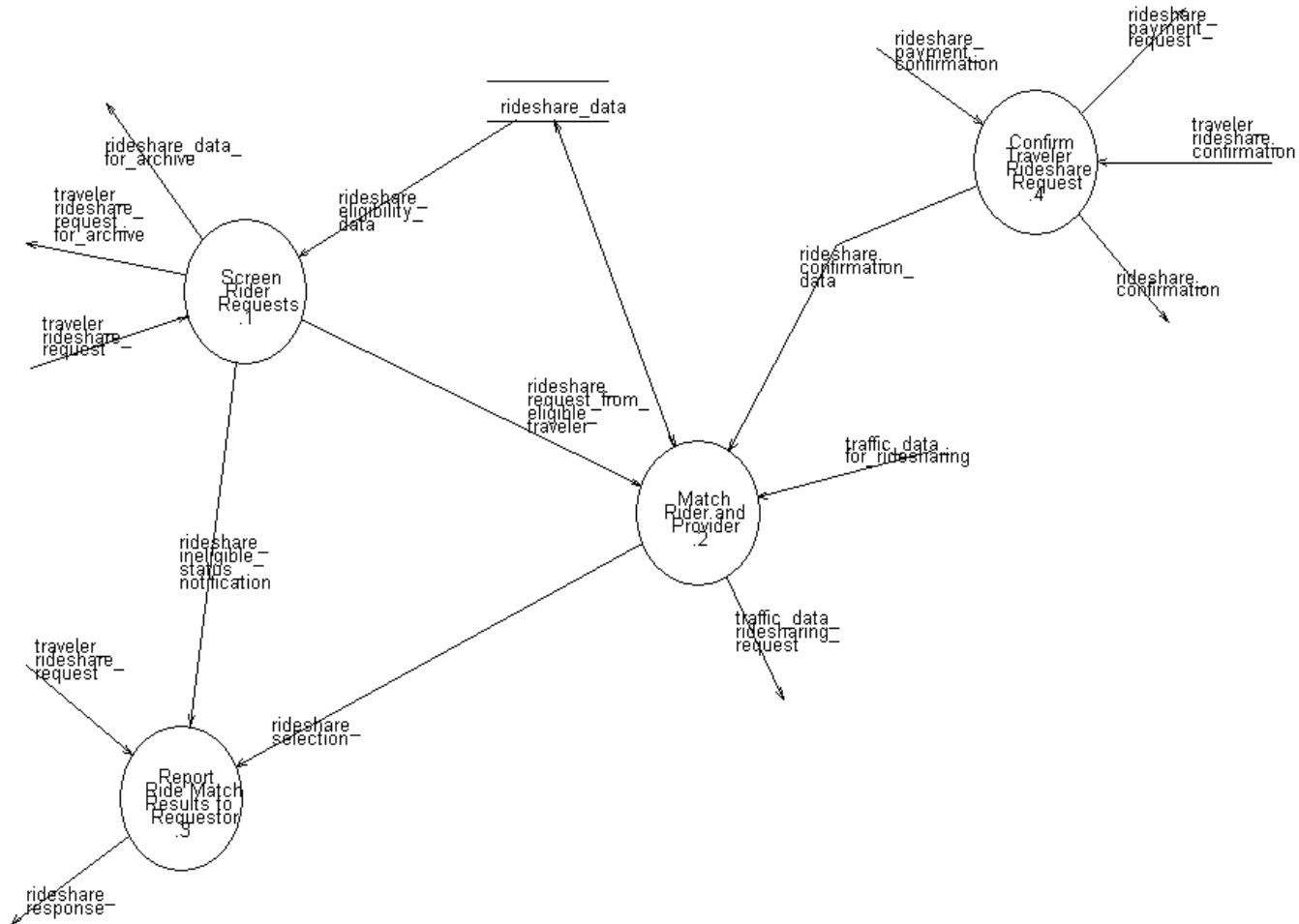
Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



6.3-Provide Traveler Services at Kiosks

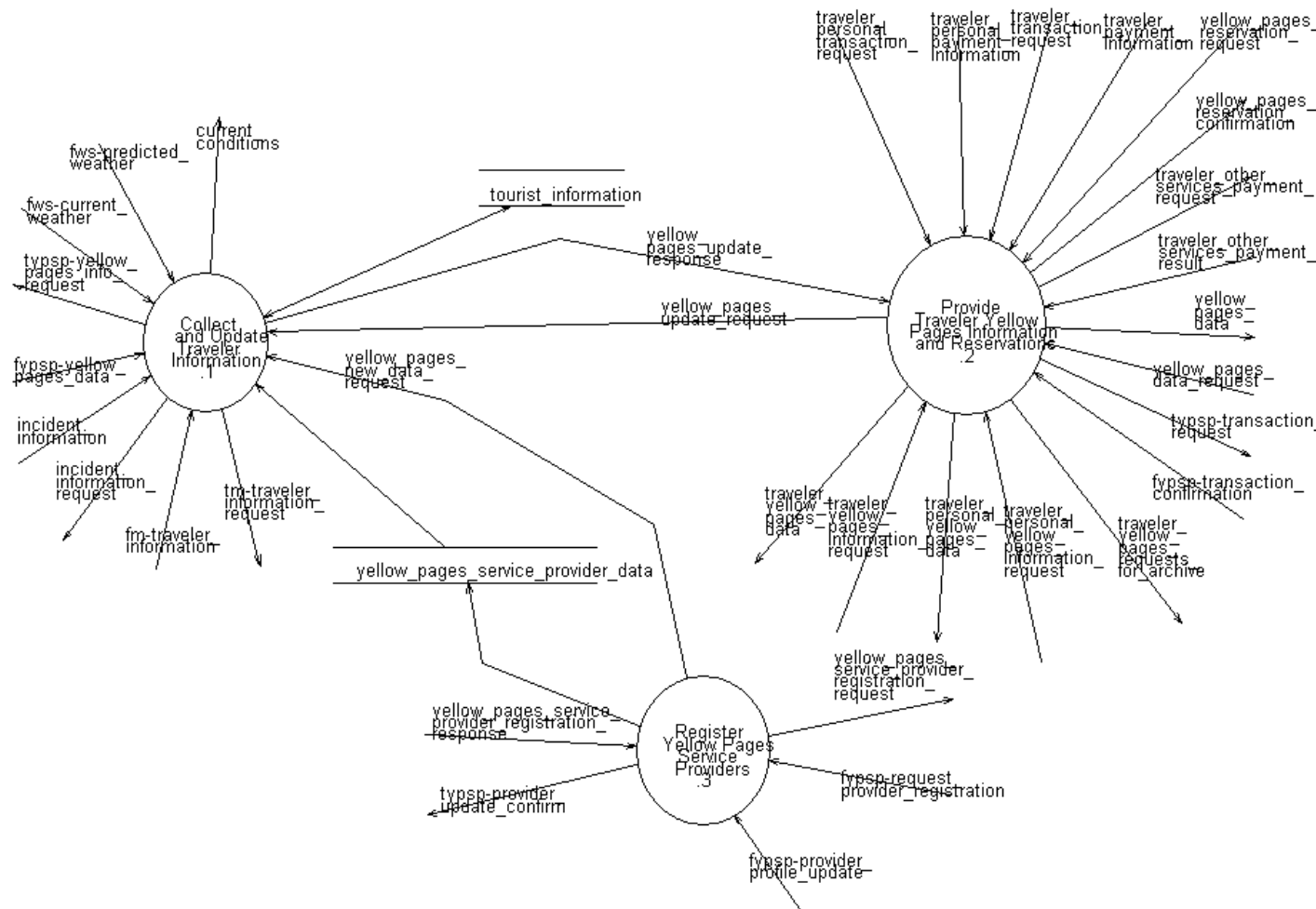


6.4- Manage Ridesharing

## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

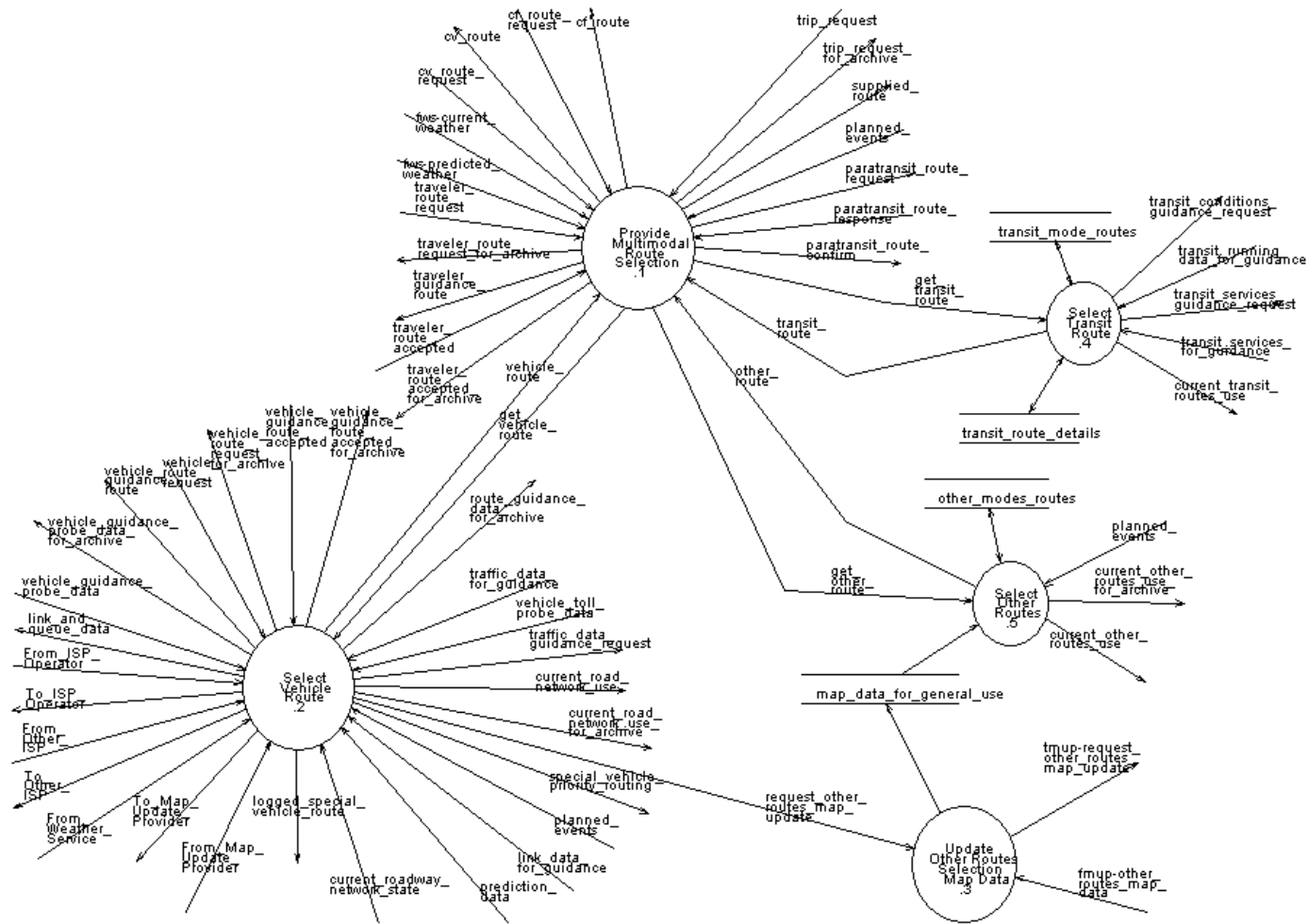


6.5-Manage Yellow Pages Services

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

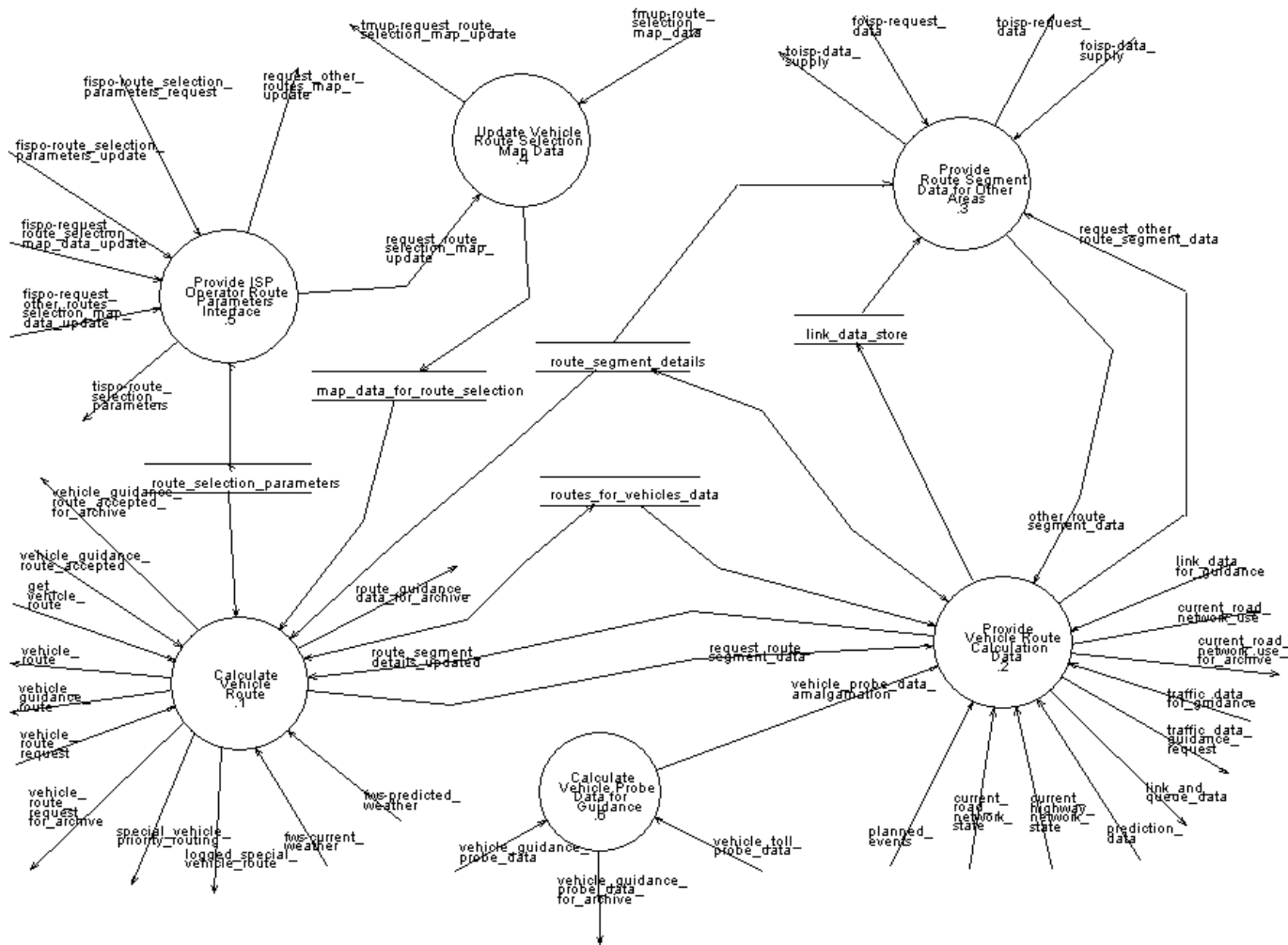


6.6-Provide Guidance and Trip Planning Services

## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

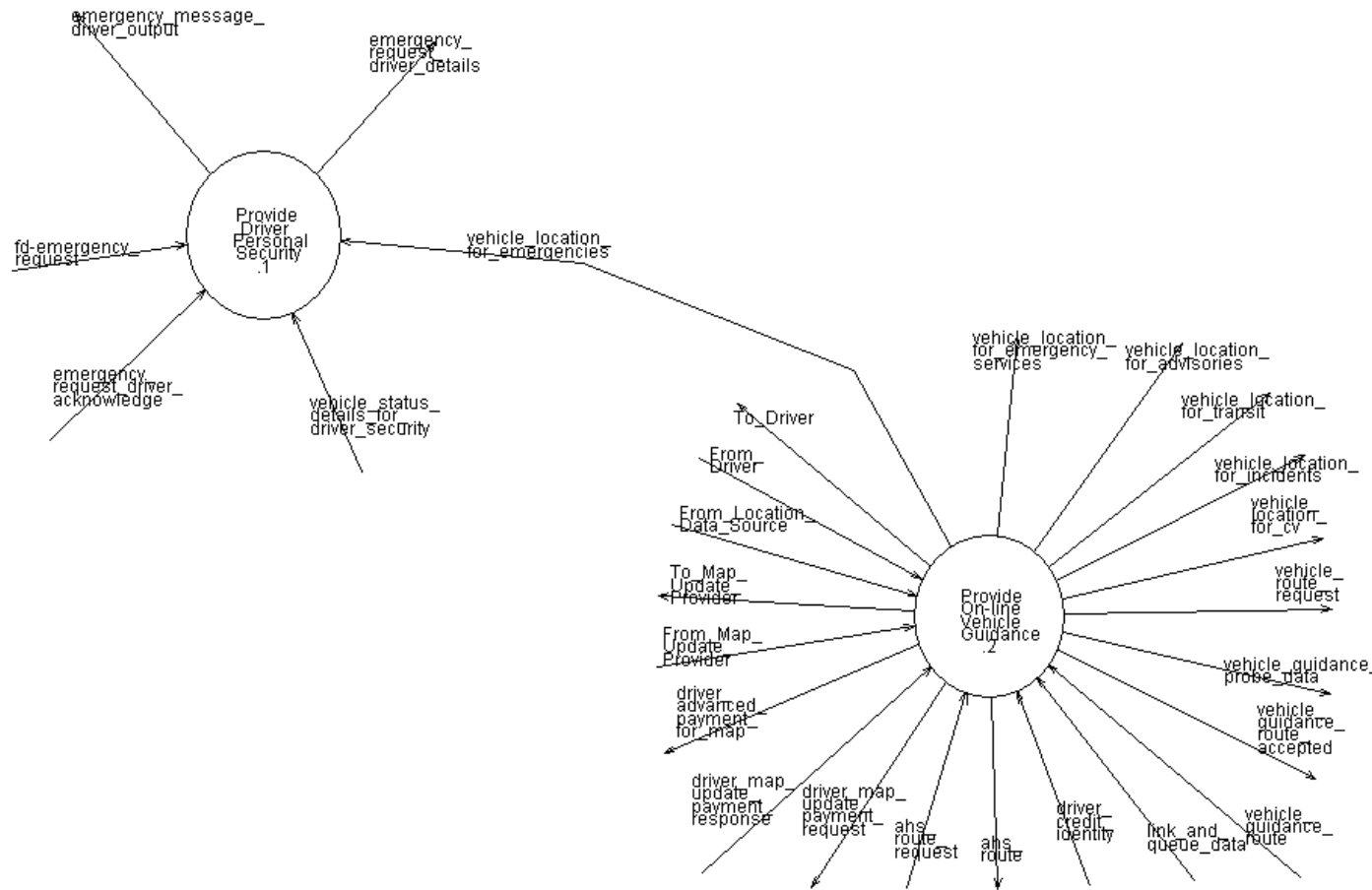


6.6.2-Select Vehicle Route

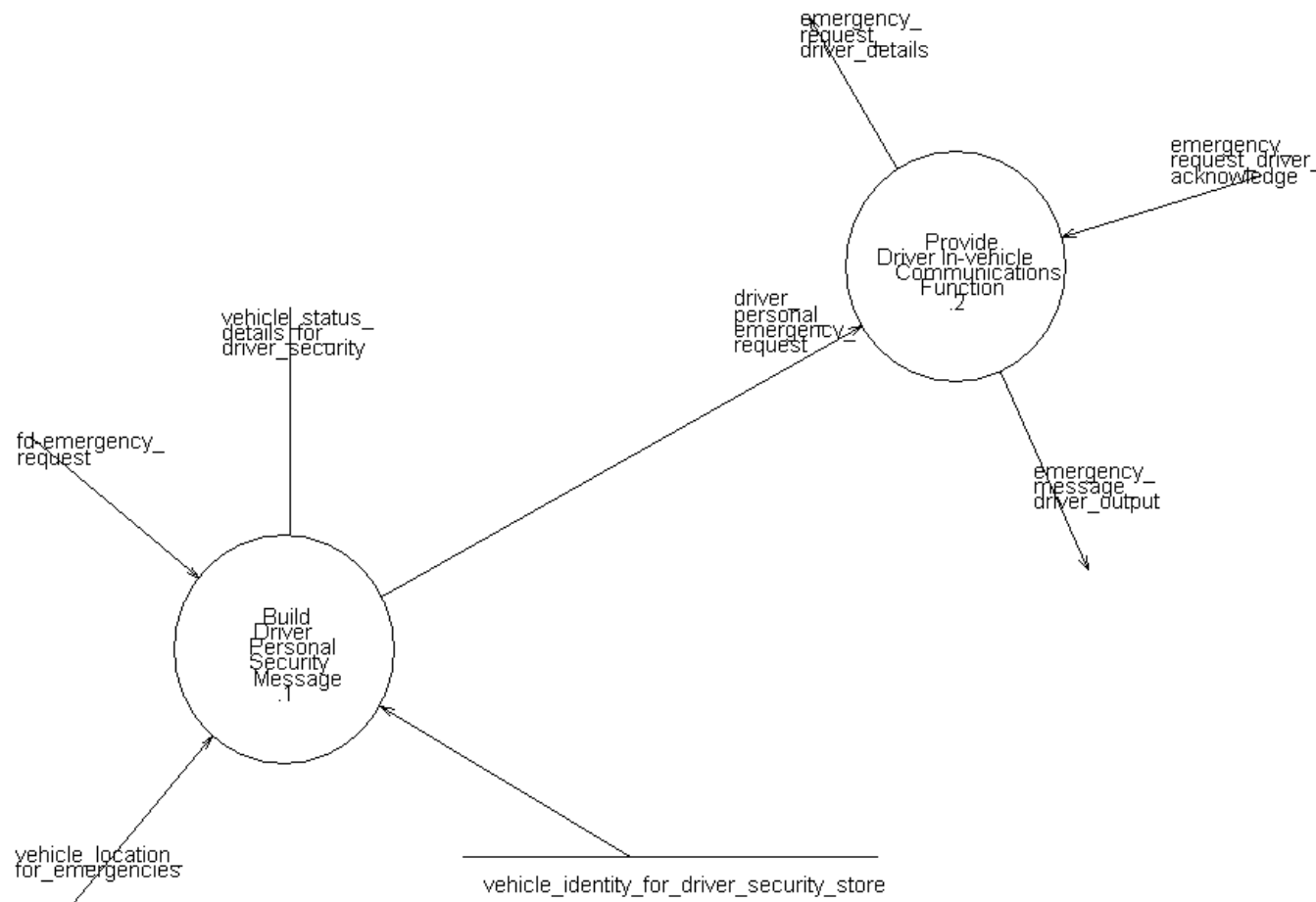
Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

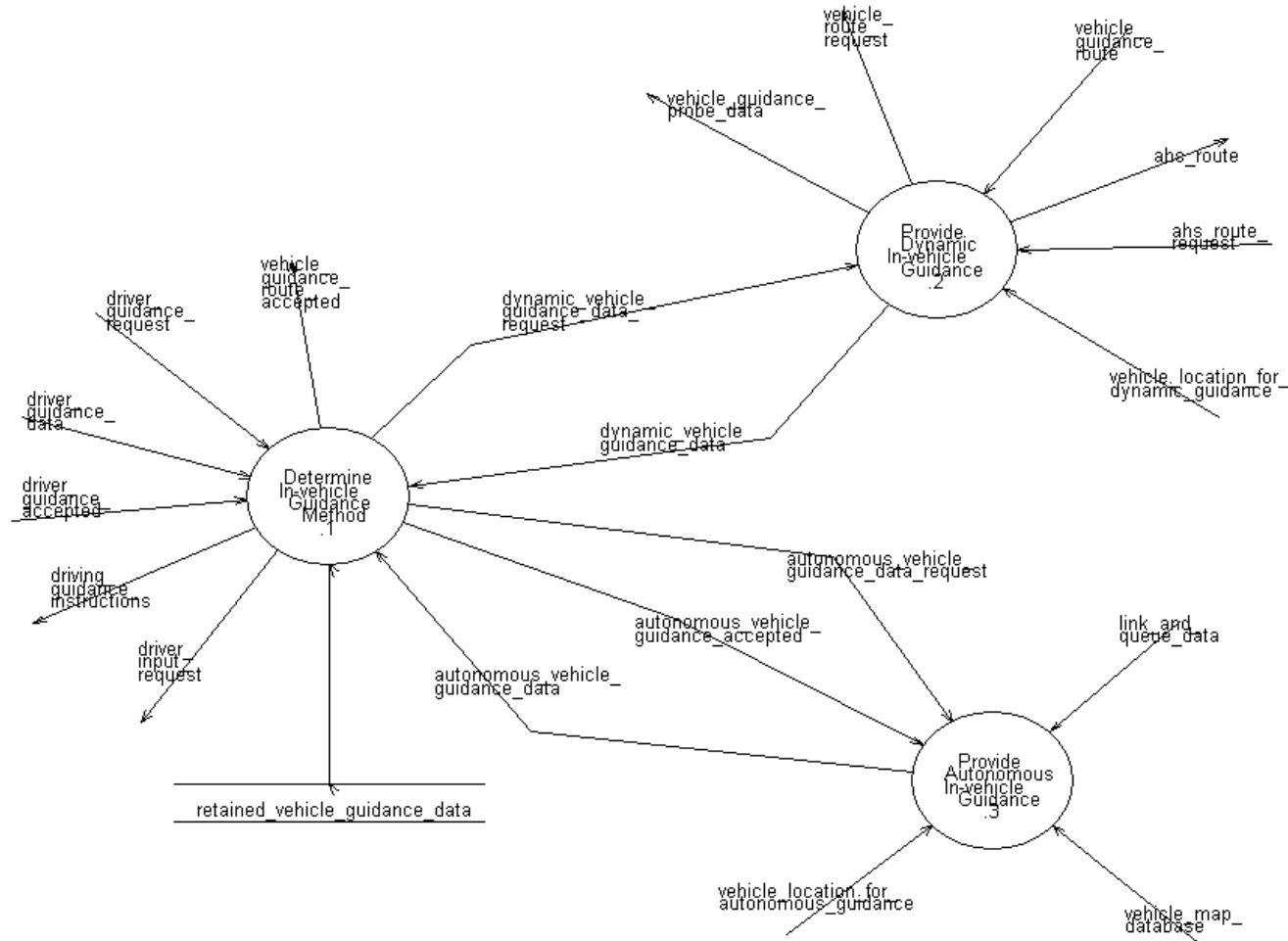


6.7-Provide Driver Personal Services



6.7.1-Provide Driver Personal Security



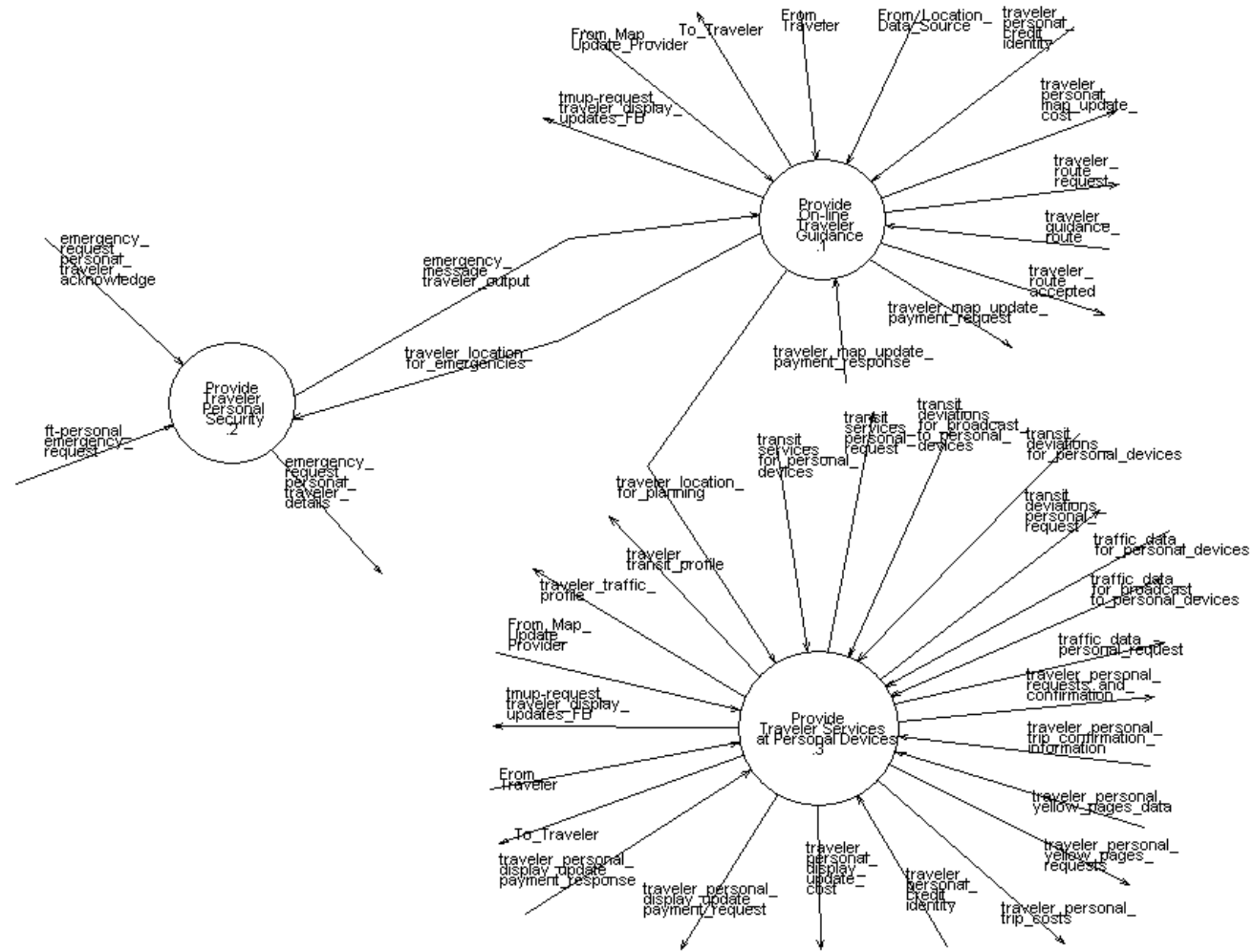


6.7.2.1-Provide Vehicle Guidance

## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

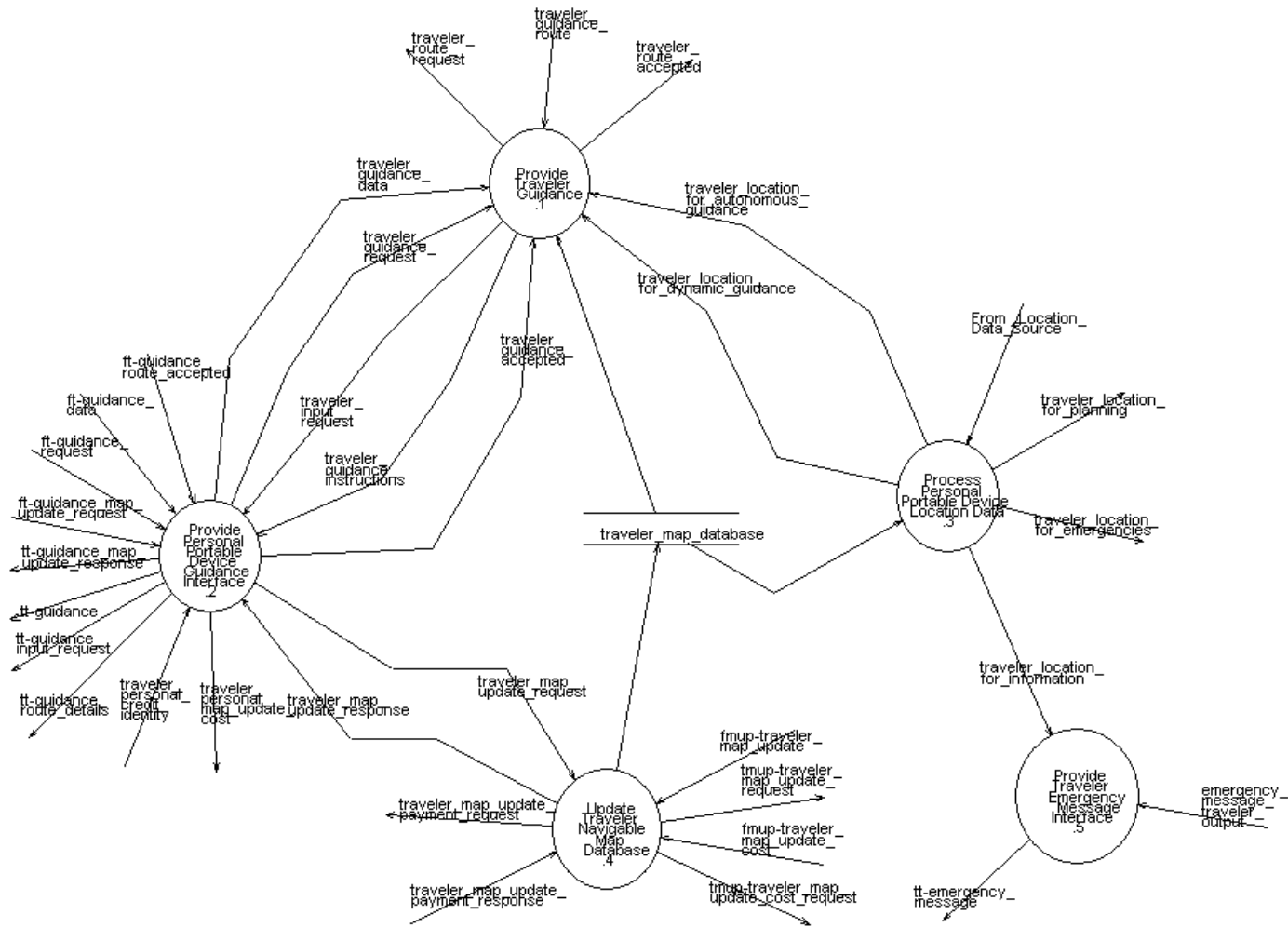


6.8-Provide Traveler Personal Services

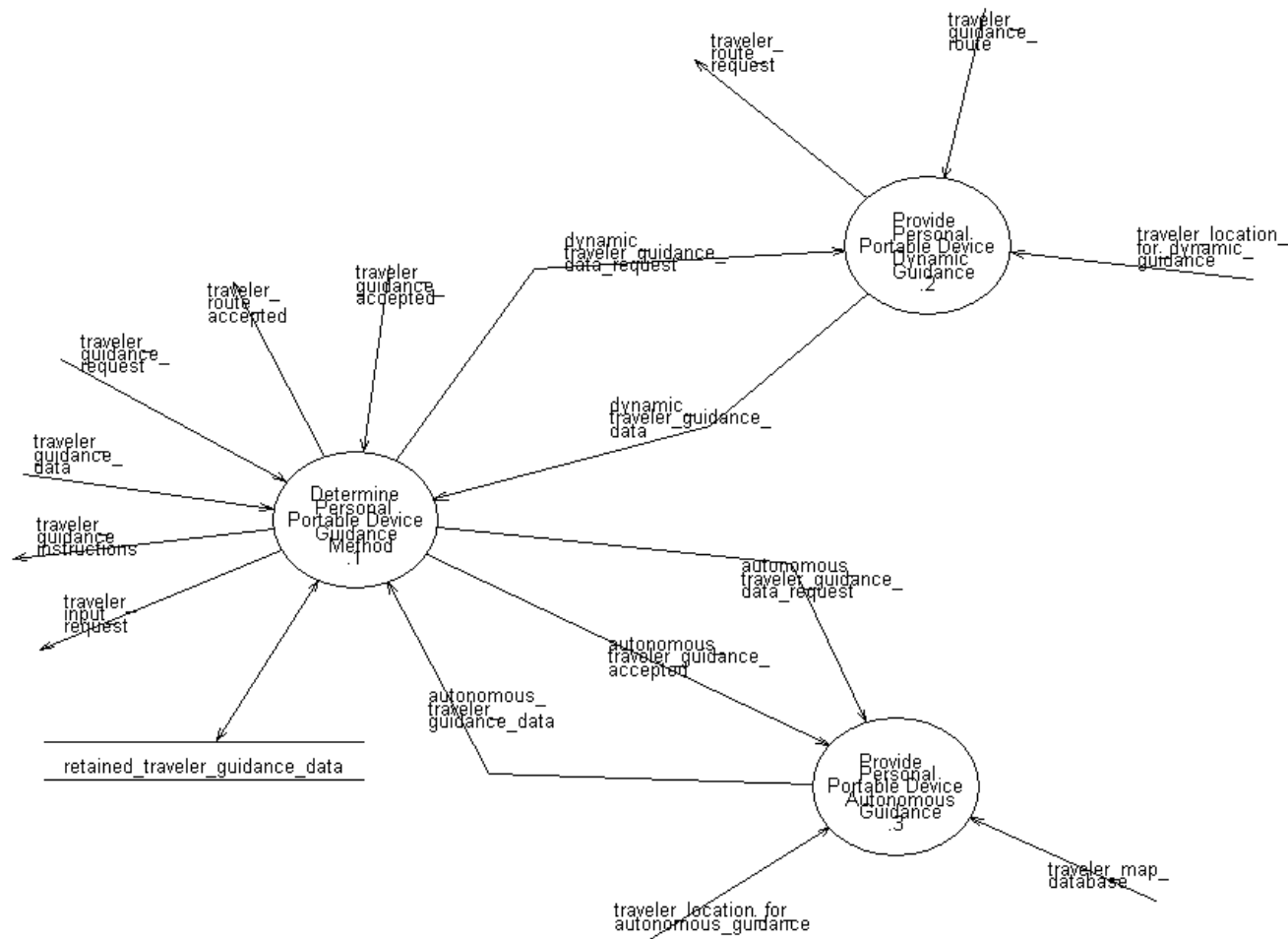
## Development of ITS System Architecture for Malaysia

Technical Note No. 2

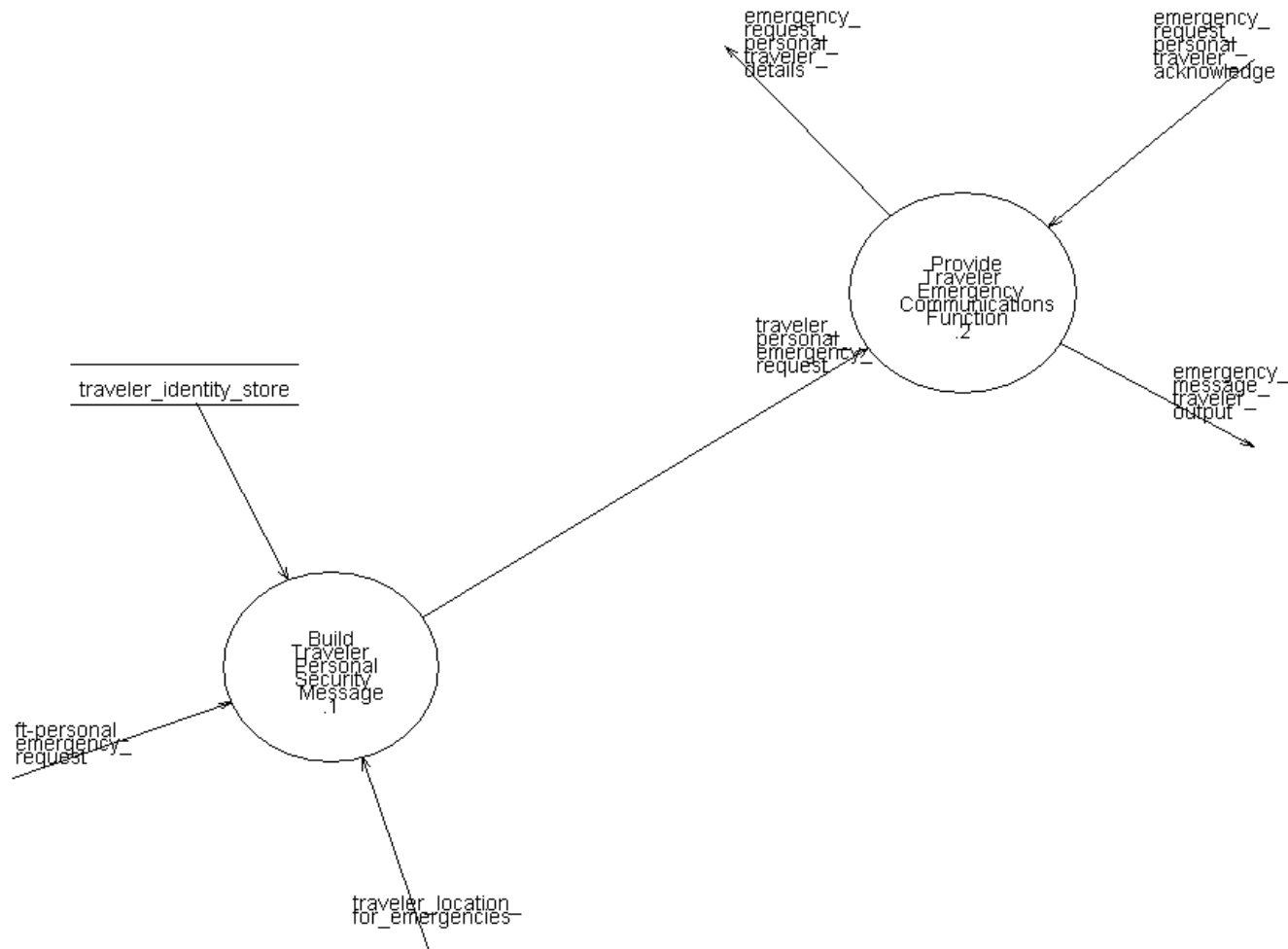
Logical Architecture Framework - Volume I



6.8.1-Provide On-line Traveler Guidance



6.8.1.1-Provide Traveler Guidance

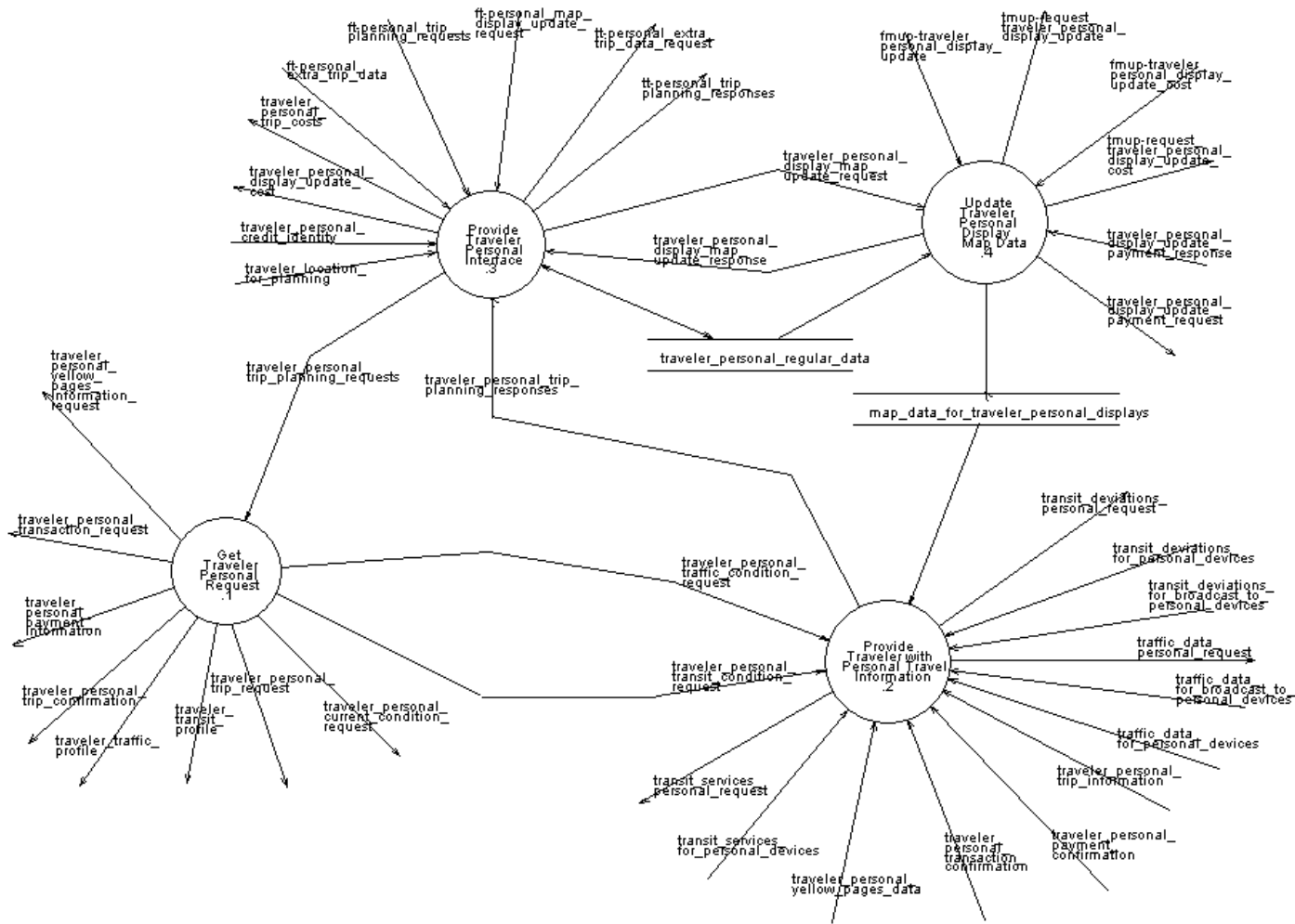


6.8.2-Provide Traveler Personal Security

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

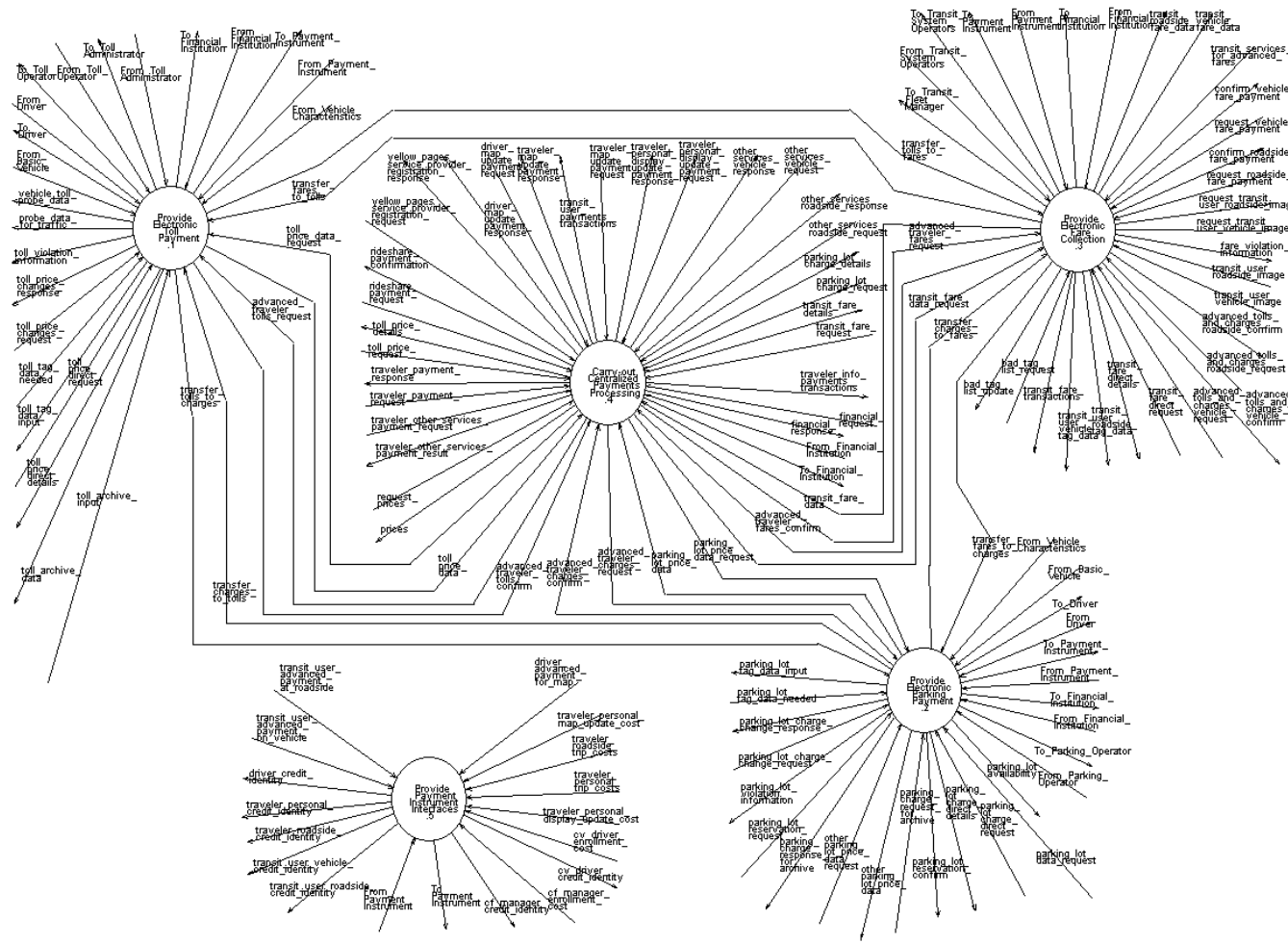


6.8.3-Provide Traveler Services at Personal Devices

# Development of ITS System Architecture for Malaysia

## Technical Note No. 2

### Logical Architecture Framework - Volume I

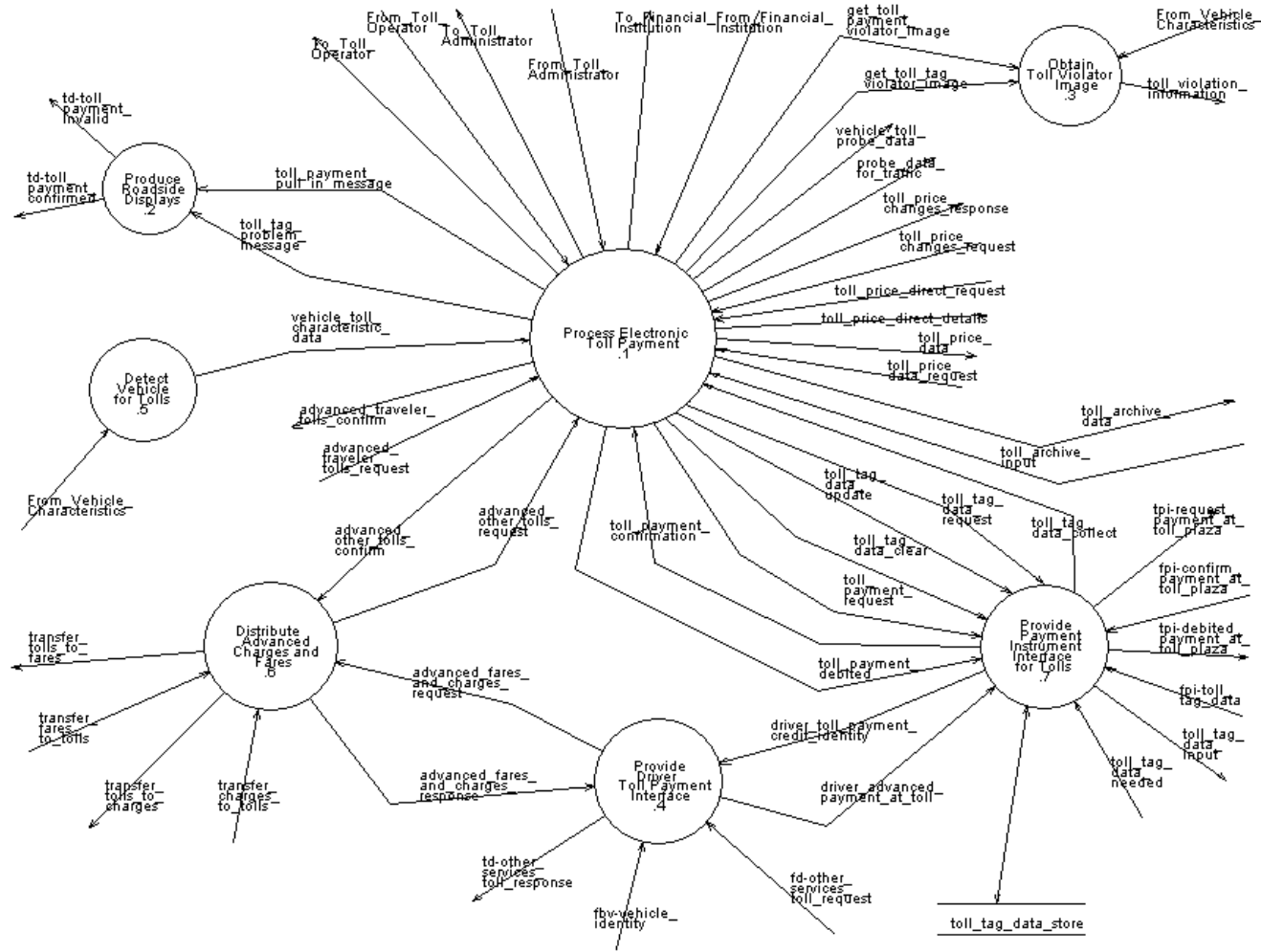


7- Provide Electronic Payment Services

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

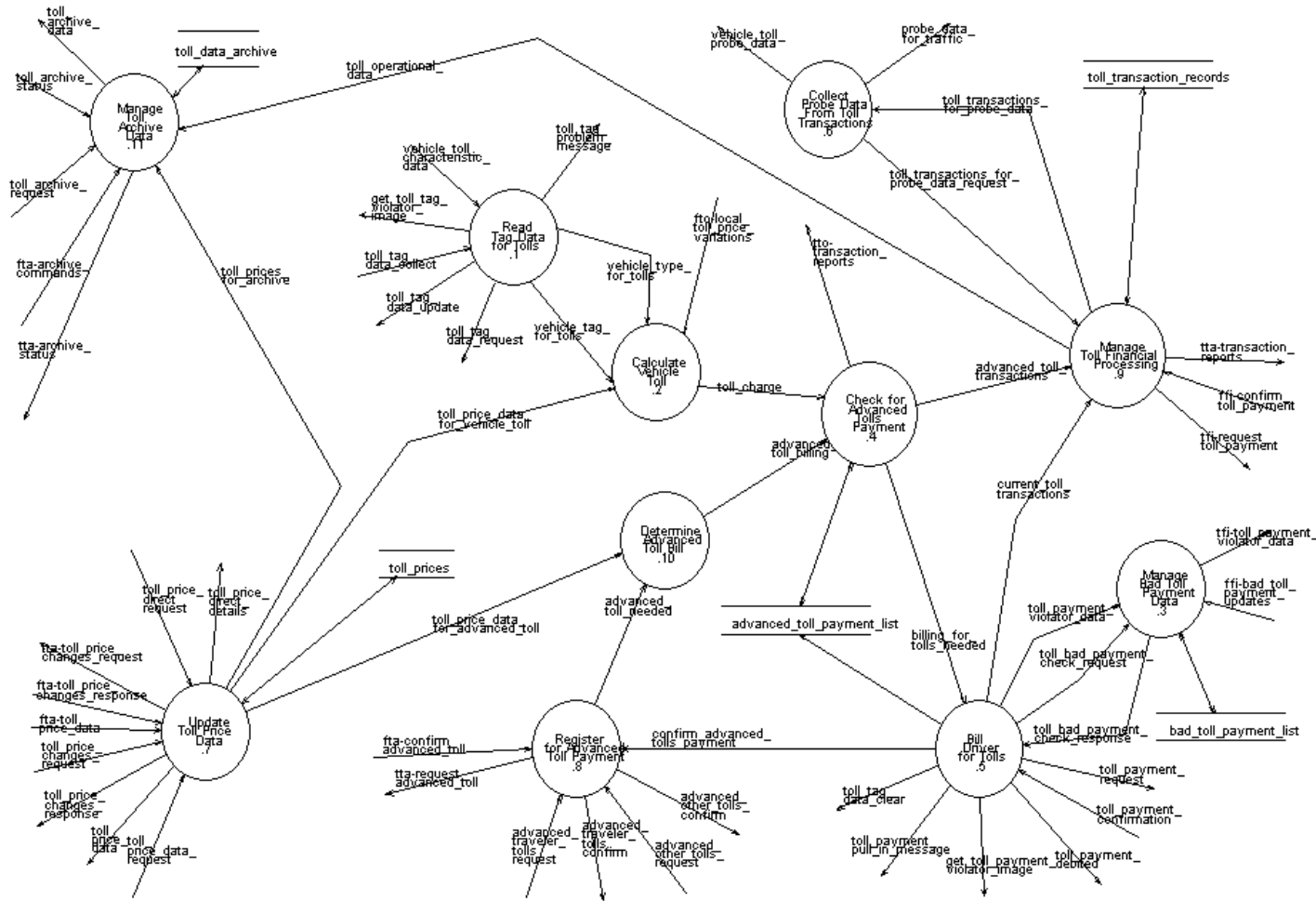


7.1-Provide Electronic Toll Payment

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

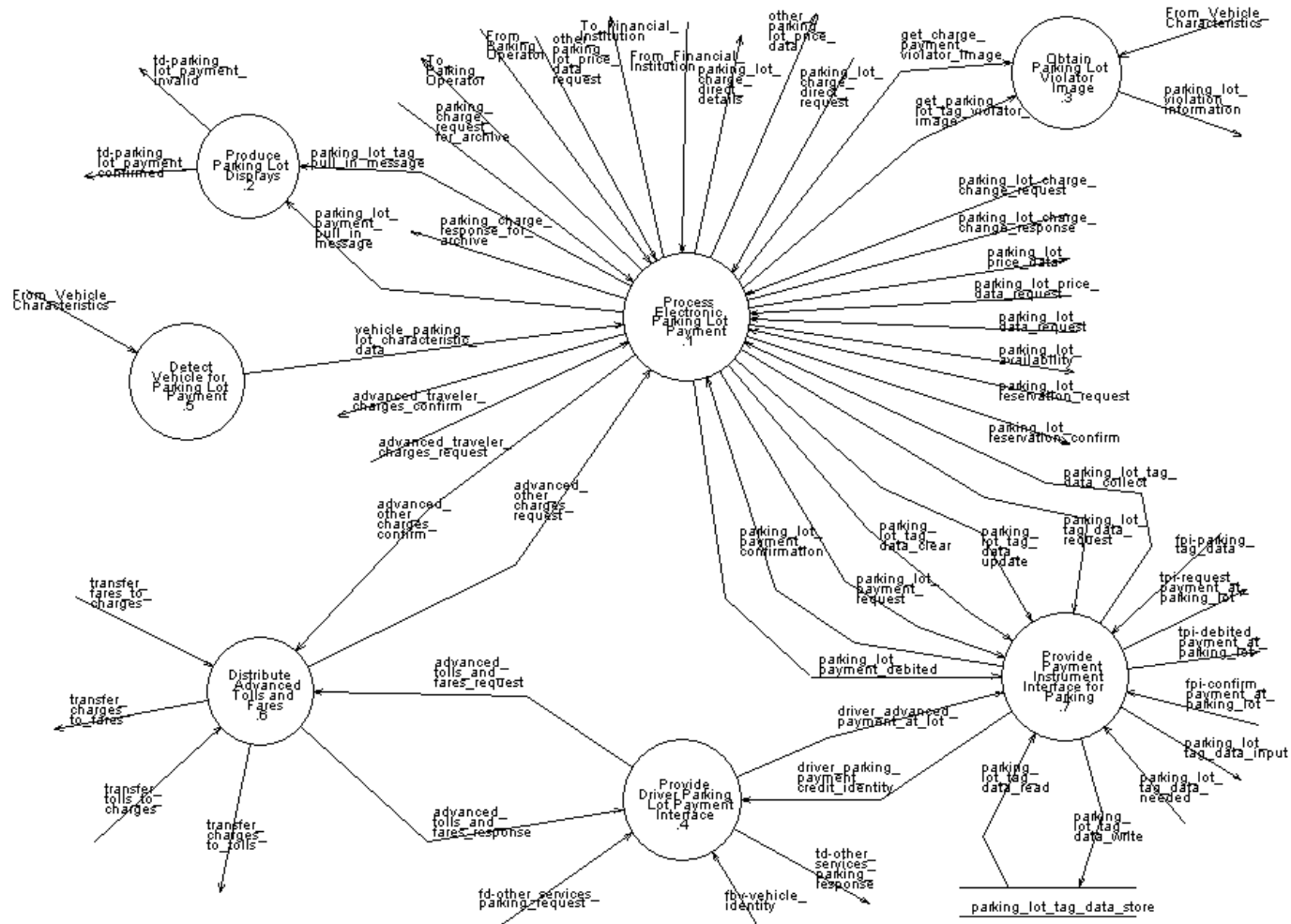


7.1.1-Process Electronic Toll Payment

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

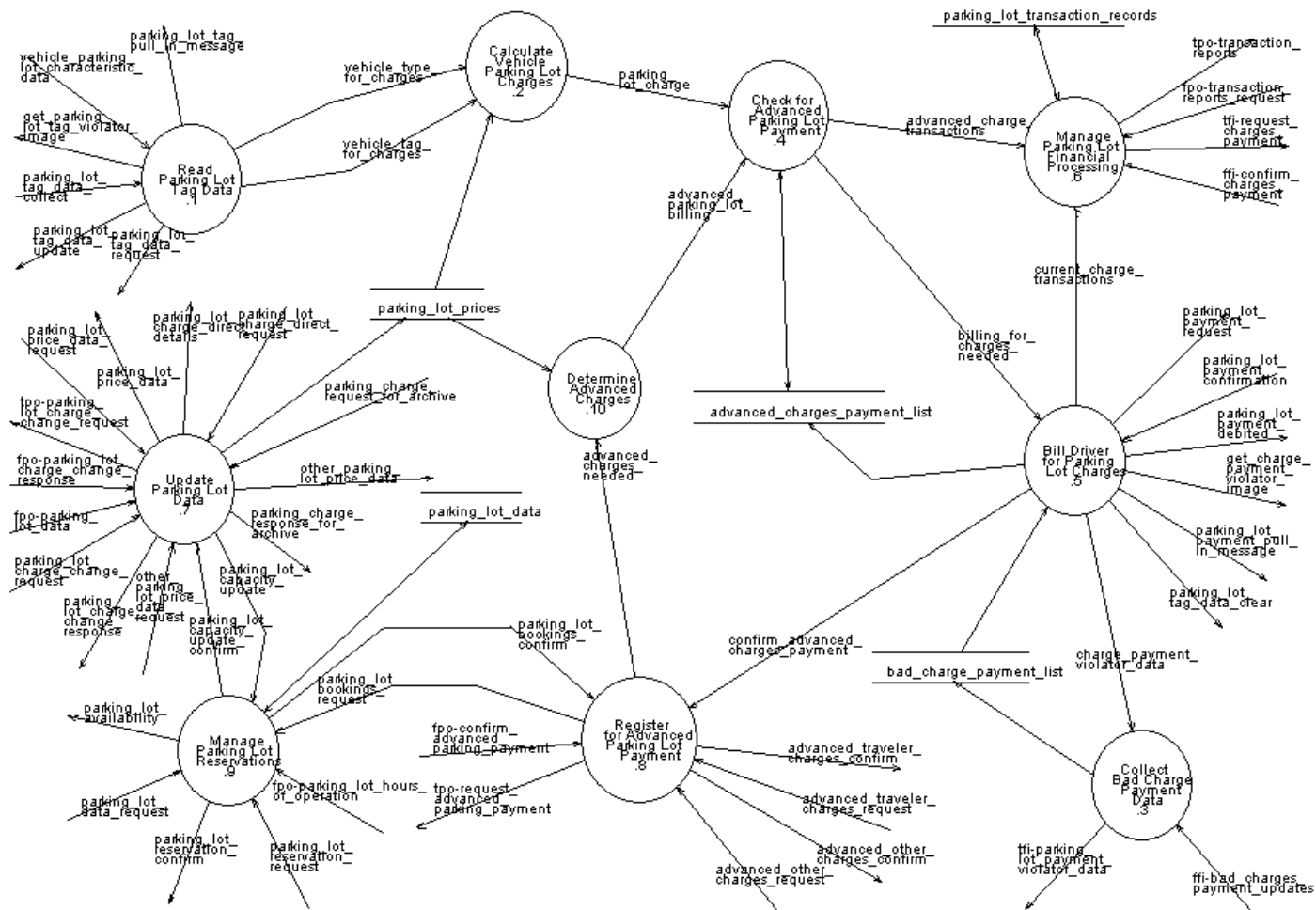


7.2-Provide Electronic Parking Payment

## Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

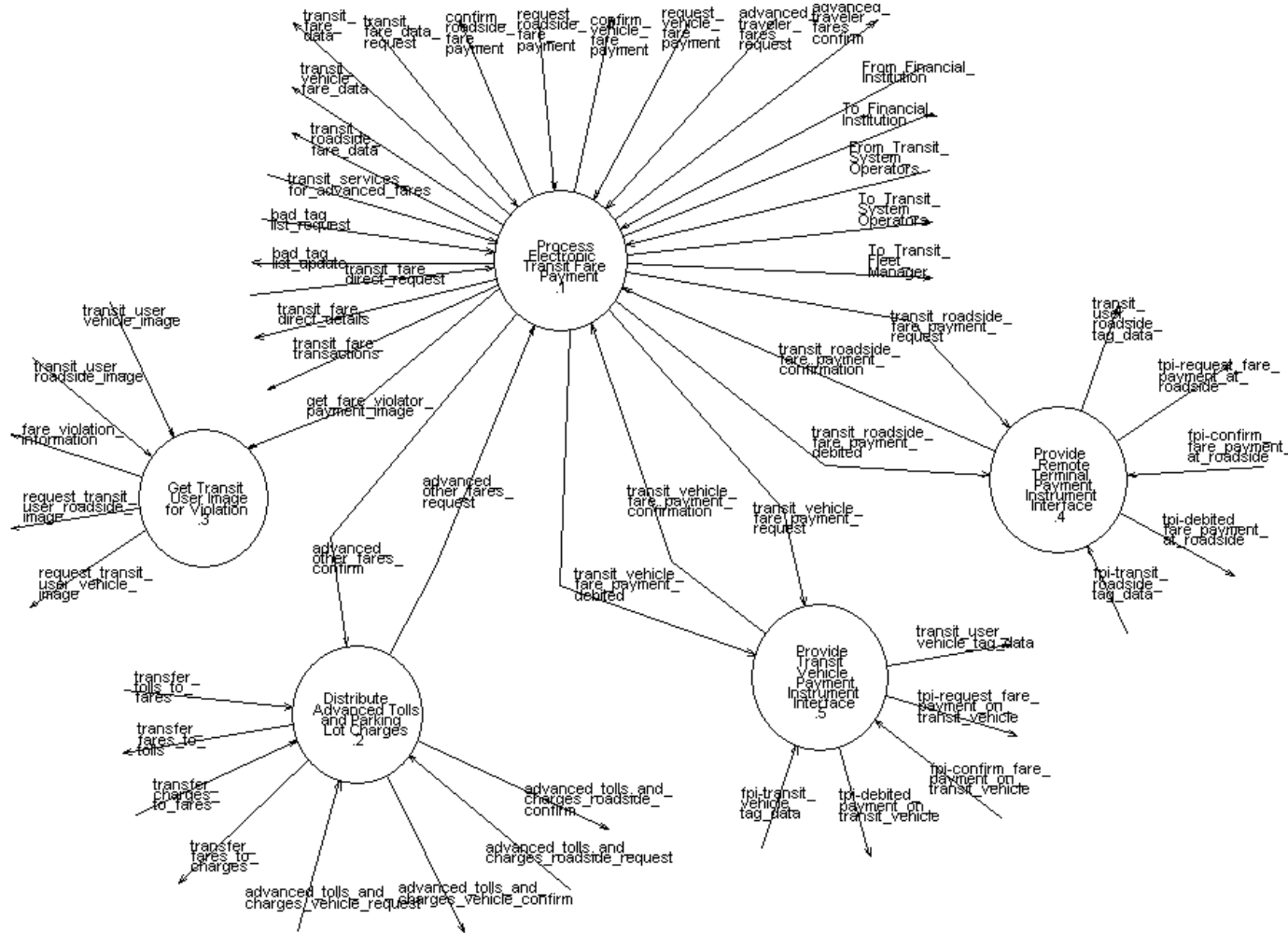


7.2.1-Process Electronic Parking Lot Payment

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

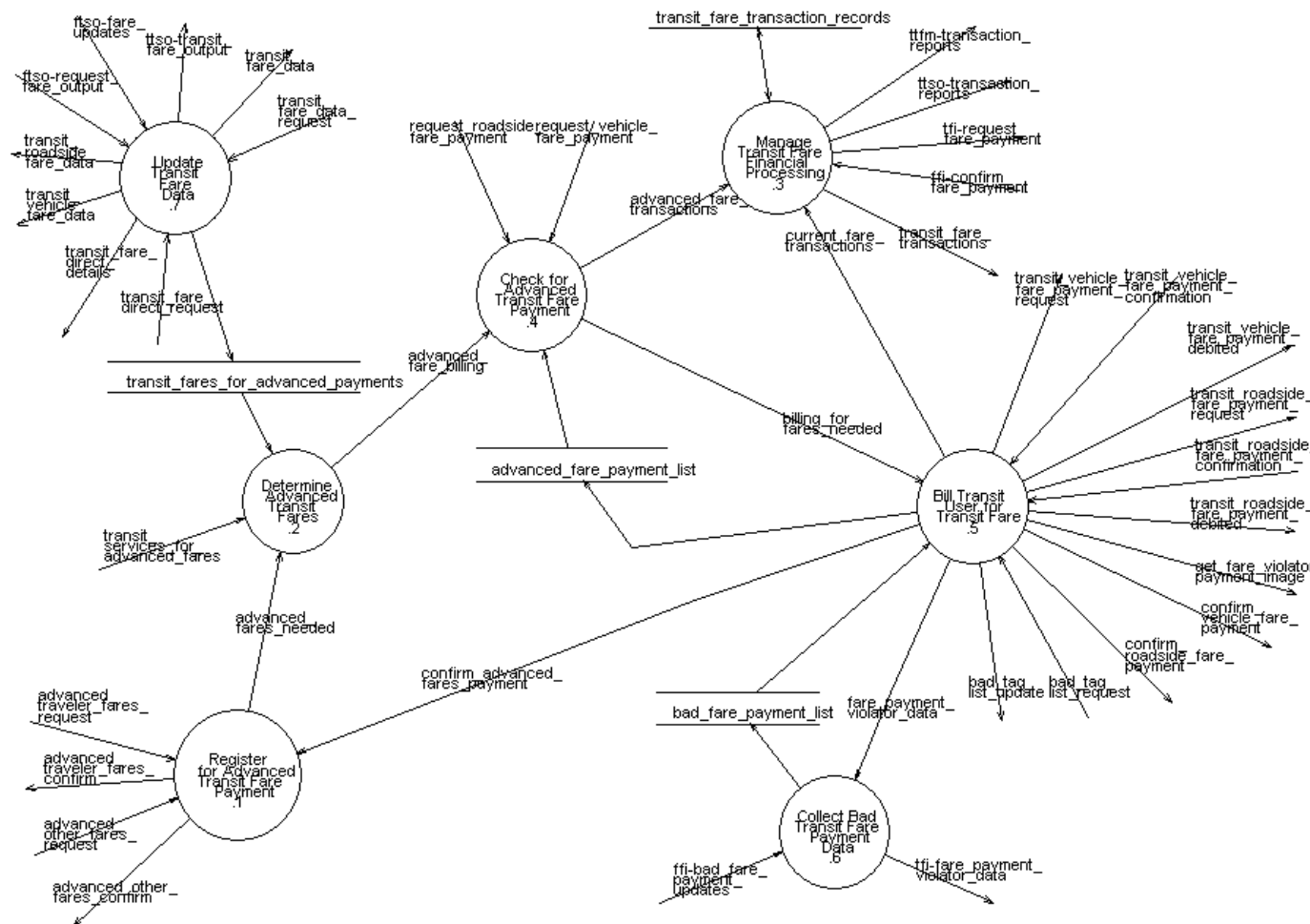


7.3-Provide Electronic Fare Collection

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

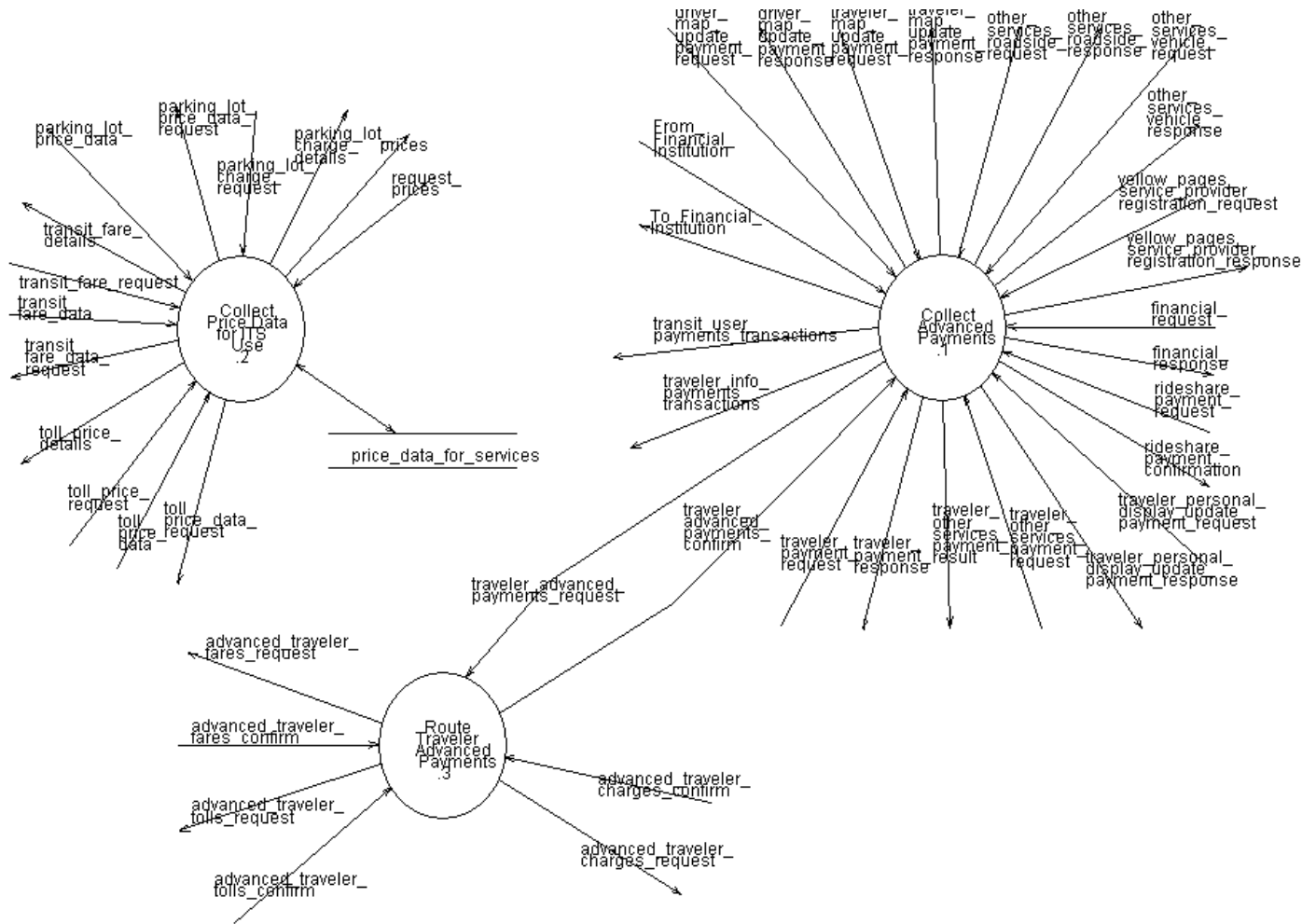


7.3.1-Process Electronic Transit Fare Payment

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

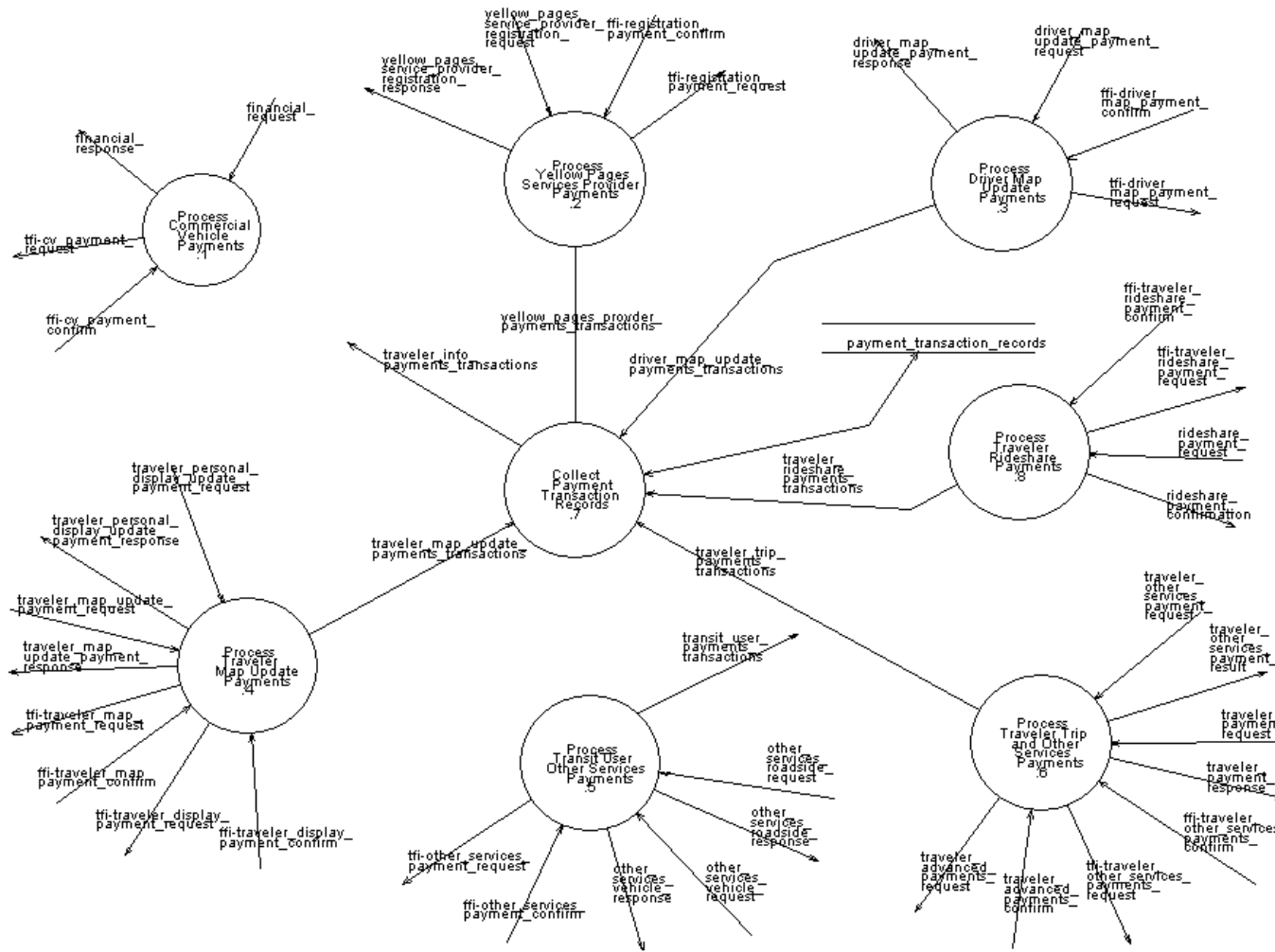


7.4-Carry-out Centralized Payments Processing

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

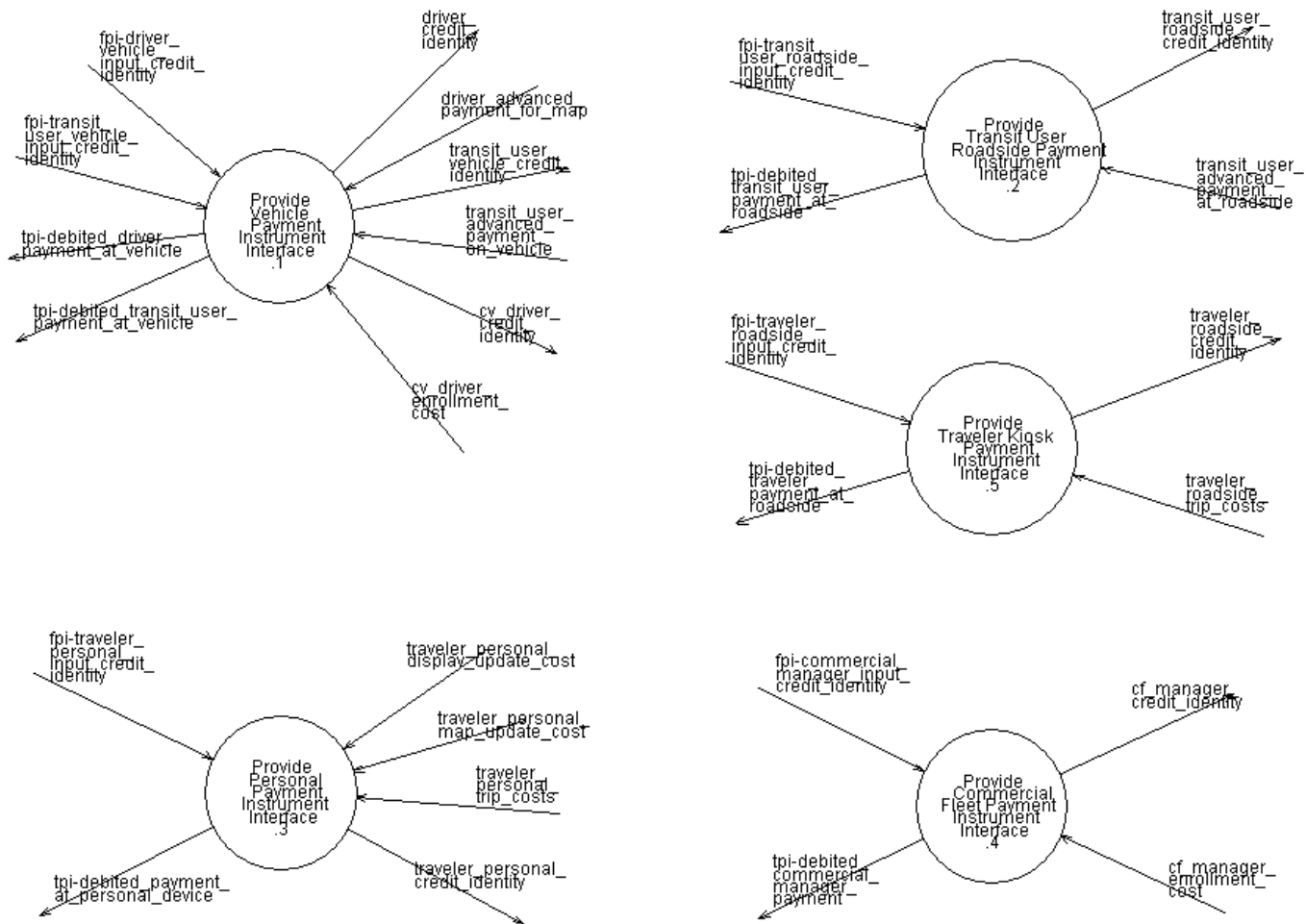


7.4.1-Collect Advanced Payments

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I

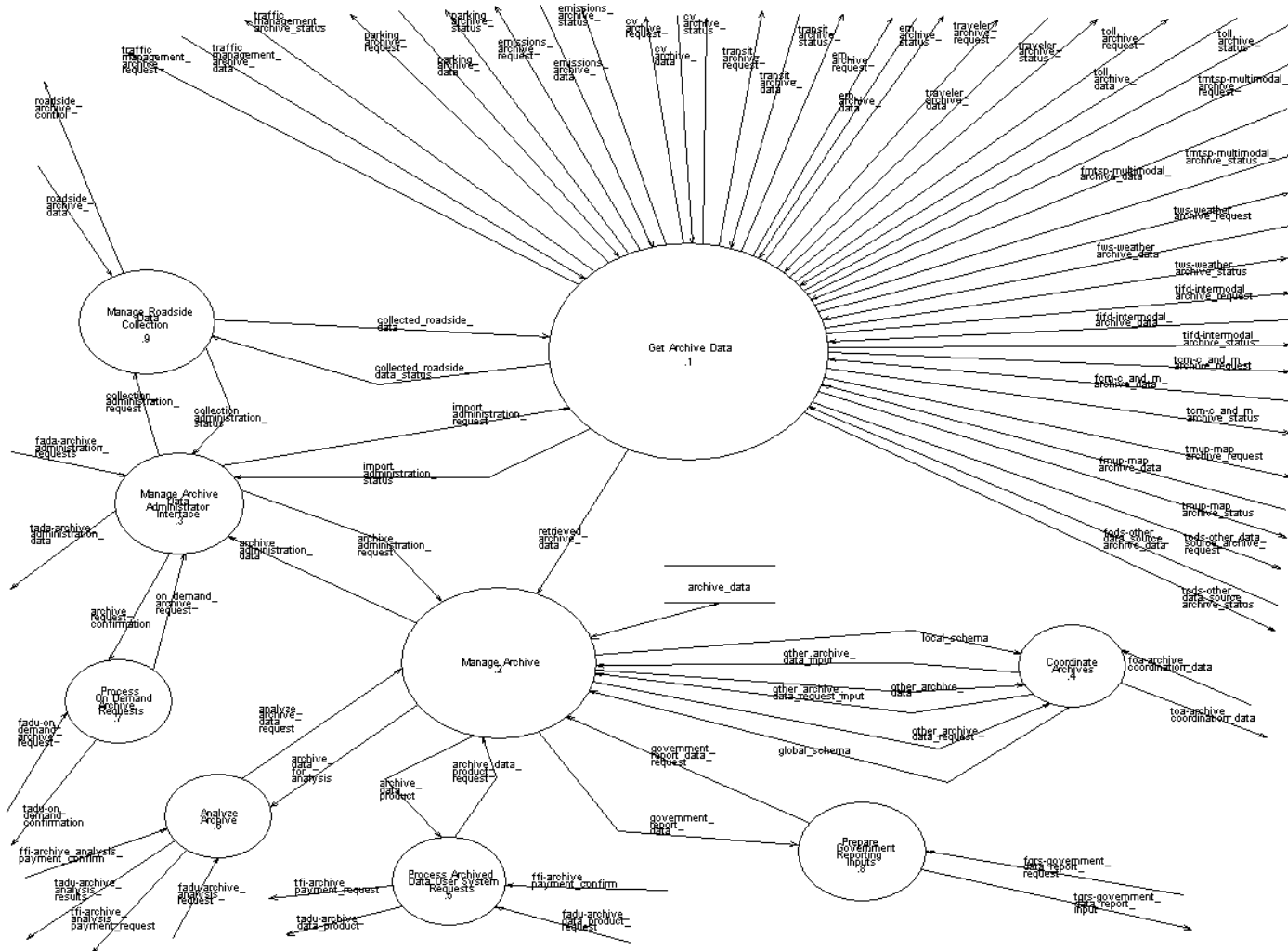


7.5-Provide Payment Instrument Interfaces

Development of ITS System Architecture for Malaysia

Technical Note No. 2

Logical Architecture Framework - Volume I



8-Manage Archived Data

9- Manage Traffic Enforcement

