European ITS Framework Architecture

-

Physical Architecture

D3.2 - Issue 1

August 2000

This public report has been produced by the KAREN (Keystone Architecture Required for European Networks) project, as part of the 4th Framework Programme - Telematics Application Programme – road sector.

KAREN partners contributing to this report are:

SIEMENS TRAFFIC CONTROLS LIMITED
UNIVERSITY OF LEEDS
ALCATEL AUSTRIA
ALCATEL SPACE INDUSTRIES
TNO
FIAT CRF
MIZAR

Document control sheet

Activity name: KAREN

Work area: Framework Architecture Development - WP3

Document title: Physical Architecture

Document number: D3.2

Electronic reference:

Main author(s) or editor(s): Richard Bossom

Other author(s): Simon Chambers, Rino Ciancaglini, Alessandro Coda, Jean-Luc Durand, Gino Franco, Peter Jesty, Markus Szvetits.

Dissemination level[1]: Public usage

Version history:

Version number	**Date**	**Editor**	**Summary of changes**
Issue 1	August 2000	R.A.P. Bossom	Final Public Issue

Approval:

	Name	**Date**
Prepared	*Richard Bossom*	*July 2000*
Reviewed	*Gino Franco*	*August 2000*
Authorised	*Jan Willem Tierolf*	*August 2000*

Circulation:

Recipient	**Date of submission**
CEC	August 2000

[1] This is either: Restricted (to the programme, to the activity partners) or for Public usage

Table of Contents

EXECUTIVE SUMMARY 1

1. Introduction 2

1.1 Outline 2

1.2 Where the document fits in the Architecture Documentation 2

1.3 Definition of a Physical Architecture 2

1.4 Overview of the Document Structure 3

1.5 List of Abbreviations 3

2. Physical Elements and definition of terms 6

2.1 Introduction 6

2.2 Physical Elements 6

2.2.1 Introduction 6

2.2.2 Systems 6

2.2.3 Sub-systems 7

2.2.4 Modules 8

2.2.5 Physical Data Flows 8

2.2.6 Terminator Data Flows 8

2.3 Other Entities 9

3. Conventions used for Document creation 10

3.1 Introduction 10

3.2 Numbering Convention 10

3.3 Detailed Document Structure 10

3.4 System Specifications 11

3.4.1 Introductions 11

3.4.2 Specification Structure 12

4. Methodology for the creation of Physical Systems 14

4.1 Introduction 14

4.2 Why create a Physical System ? 14

4.3 Pre-requisites for the creation of a Physical System 15

4.4 Method for the creation of a Physical System 15

4.5 Aids to the creation of a Physical System 19

4.5.1 Introduction 19

4.5.2 Sub-systems and Modules Database 19

4.5.3 Physical Data Flows Database 21

4.5.4 Documentation 21

4.6 Adding extra User Needs 21

4.7 Adding extra Functionality and Terminators 22

4.7.1 Introduction 22

4.7.2 Adding new Terminators 23

4.7.3 Adding new Functionality 23

4.7.4 Using the Functional Architecture Database 24

5. Methodology for the creation of an Architecture 26

5.1 Introduction 26

5.2 Architectures 26

5.3 Pre-requisites for creating an Architecture 27

5.4 Method for the creation of an Architecture 27

5.5 Adding new Terminators and Functionality 29

5.6 Modifying the Physical Architecture Database 30

5.7 Use of Architecture Development Tools 30

6. Physical Architecture links to the outside World 32

6.1 Introduction 32

6.2 Definition of Terminator 32

6.3 KAREN Terminator Definitions 32

6.4 Context Diagram 42

6.5 Table of Acronyms 43

6.6 Use of Terminators by Functional Area 45

7. Summary of Key Issues from all Systems 48

7.1 Introduction 48

7.2 P1 - Integrated Urban Traffic and Public Transport Management System 48

7.3 P2 – RDS-TMC Italian System 49

7.4 P3 – Urban Traffic Control and Public Transport Priority System 49

7.5 P4 – Telematics Technologies for Transport and Traffic in Torino System 50

7.6 P10 - Electronic Cash Transaction System 51

7.7 P11 - Automatic Road Tolling System 52

7.8 P 22 – Hazardous Goods Management System 52

7.9 P 30 - Urban Traffic Management System 52

7.10 P31 - Inter-urban Traffic Management System 53

7.11 P50 - Advanced Driving Assistance System 53

7.12 P60 - Traveller Assistance and Route Guidance System 53

7.13 P 70 - Law Violation and Vehicle Detection System 54

7.14 P81 - Distributed Freight Management System 54

8. Preliminary Safety Analysis 56

8.1 Introduction 56

8.2 Modelling P30 - Urban Traffic Management System 56

8.3 Hazard identification 58

8.3.1 ‘What If?’ analysis 59

8.4 Safety issues 65

8.4.1 System boundary and safety objectives 65

8.4.2 Safety requirements 65

8.5 Conclusion 67

9. References 69

Appendix 1 P30 I-O Matrix 71

Figures

Figure 1 European ITS Framework Architecture Context Diagram 43

Figure 2 Structure of a PASSPORT Diagram..57

Figure 3 Structure of a PASSPORT Spreadsheet for a KAREN System.................................57

Tables

Table 1 Database Table format for Functions in "example System" Development 20

Table 2 Database Table format for Data Flows in "example System" Development 21

Table 3 European ITS Framework Architecture Terminator Definitions 32

Table 4 Terminator Acronyms 44

Table 5 Use of Terminators by Functional Areas 46

Table 6 Definition of Controllability Categories 58

Table 7 “What if ?” Analysis Results Table..60

Table 8 Relationship between Controllability Categories and SIL 65

Table 9 P60 I/O Matrix – Part 1 72

Table 10 P60 I/O Matrix – Part 2 74

Table 11 P60 I/O Matrix – Part 3 76

Table 12 P60 I/O Matrix – Part 4 78

Table 13 P60 I/O Matrix – Part 5 80

Executive Summary

This Document provides a description of the Physical Architecture which forms part of the European ITS Framework Architecture. A Physical Architecture defines and describes how the functionality created in the European ITS Functional Architecture can be grouped to form Systems that can be produced. These Systems will use components that are produced from hardware, software, or a mixture of the two. Through their inclusion of parts of the Functional Architecture, these Systems will of course be able to satisfy some or all of the requirements of the European ITS User Needs. These User Needs are provided in a separate document produced by another part of the KAREN Project.

This Document considers the Physical Architecture as a series of “example Systems”. This is approach has been adopted because there are many ways in which a Physical Architecture can be produced from a Functional Architecture. The purpose of these “example Systems” is to illustrate examples of Physical Systems that could be produced to fulfil some of the European ITS User Needs. The description of each “example System” is contained in a separate Annex (Annex 1) to the Main Document for ease of access and reading. It includes details of the parts of the Functional Architecture (Functions, Data Flows and Data Stores) that are included in each “example System”, as well as providing information on what they can provide. An example of a System Specification that can be used to procure equipment to implement a Physical System is included in another separate Annex (Annex 2) to the Main Document.

A description of the methodology used to create the Physical Systems and that which could be used to create Architectures that are based on the European ITS Framework Architecture are discussed and described. Templates for the diagrams and parts of the descriptions are provided in a separate Annex (2). There is also advice on how to make changes to the Architecture to accommodate a different scope of ITS services that a System or Architecture must provide.

The final part of each “example System” description includes a section on “Key Issues”. These are issues that need to be addressed and resolved to enable each System to be successfully deployed and implemented. They are highlighted grouped in a single Chapter in the Main Document for consideration by other work within the KAREN Project.

A Preliminary Safety Analysis (PSA) has been included for one of the “example Systems”. It is intended that this should be used as a model if similar analyses are required of other “example Systems”.

The way in which this Document should be used and its role within the European ITS Framework Architecture will be described in the Overview Document. This will be produced separately.

1. Introduction

1.1 Outline

This Document is part of the set of deliverables produced by the KAREN Project to describe the European ITS Framework Architecture. This particular document (D3.2) provides a description of the Physical Architecture that has been developed by the Project Team. This Architecture has been produced in the form of "example Systems". These are intended to show examples of the physical Systems that can be developed using components from the Functional Architecture. It was not considered to be in the best interests of European ITS deployment to develop a single "definitive System" as the European ITS Physical Architecture, because there is no one way in which this can or should be done. The background to the development of the European ITS Framework Architecture as a whole is provided in the European ITS Framework Architecture Overview document (D3.6).

1.2 Where the document fits in the Architecture Documentation

The Document is one of a set of six documents by the KAREN Project to describe the complete European ITS Framework Architecture. The other documents in the set are as follows:

D3.1 European ITS Functional Architecture
D3.2 European ITS Physical Architecture - this document
D3.3 European ITS Communications Architecture
D3.4 European ITS Cost Benefits Report
D3.5 European ITS Deployment Study Report (internal use only, but included in D4.2)
D3.6 European ITS Framework Architecture Overview
D3.7 European ITS Models for ITS deployment

All of the above documents that have references in this Document are shown in its last Chapter. The Overview Document provides an introduction to the whole European ITS Framework Architecture and guidance on its use.

1.3 Definition of a Physical Architecture

A Physical Architecture defines and describes the way in which the constituents of the Functional Architecture can be brought together into groups to form physical entities. The main characteristics of these entities are firstly that they provide one or more services required by the User Needs, and secondly that they can be created. This creation process may involve physical things such as roadside structures and various forms of equipment, non-physical things such as software, or a combination of the two. In the European ITS Physical Architecture these physical entities are called "example Systems". A definition of an "example System" is provided in Chapter 2.

The Physical Architecture forms part of the European ITS Framework Architecture and therefore shares its characteristics. These characteristics are described in the Overview Document - see Deliverable D 3.6.

1.4 Overview of the Document Structure

This Document is divided into two main parts – the Main Documents and the Annexes. The Main Document comprises nine Chapters. After this Chapter, the others contain the following:

Chapter 2 – a description of the components associated with a Physical Architecture;

Chapter 3 – the conventions that have been used in the creation of the "example Systems" that are part of the European ITS Physical Architecture;

Chapter 4 – the methodology that should be used for creating an "example System" that is a Physical Systems;

Chapter 5 – the methodology that should be used for creating "example Systems" that is an Architecture;

Chapter 6 – the links that the Physical Architecture has with the outside World;

Chapter 7 – a collection of all the Key Issues identified in the development of the "example Systems";

Chapter 8 – a Preliminary Safety Analysis (PSA) of one of the "example Systems" which is intended to be used as a model for similar analyses that may be required of other "example Systems";

Chapter 9 – references.

Two Annexes are provided for this Main Document. One contains all of the "example Systems" that have been created and included in the Physical Architecture. The second Annex contains lists of Functions and Data Stores, a sample System Specification and templates for use in creating new "example Systems". The Executive Summaries for each of the Annexes will be found at the end of this (Main) Document.

The Appendix to this Document contains a table of data that was used to produce the PSA that is described in Chapter 8.

1.5 List of Abbreviations

The following abbreviations may be used in this (Main) Document and in either or both of the Annexes.

ACI	Automobile Club d'Italia (Italian automobile club)
ADAS	Advanced Driving Assistance System
AISCAT	Associazione Italiana Società Concessionarie Autostrade e Trafori (Italian National association of Motorway Operators)

ANAS	Ente Nazionale per le Strade (Italian road national administration)
ATP	Actual Trip Preference - used in several Systems
DATEX	DAta EXchange - the name given to a specification and activities aimed at providing and maintaining a standard for the exchange of data. For more information please go to the DATEX Web Site at: http://www.datex.org.
DFD	Data Flow Diagram - see Chapter 2 of KAREN Functional Architecture Deliverable Document for a detailed definition.
DGPS	Differential GPS
DSRC	Dedicated Short Range Communications
EDI	Electronic Data Interchange
EU	European Union
FMEA	Failure Mode and Effects Analysis
HMI	Human Machine Interaction
FFM	Freight and Fleet Management
GPS	Global Positioning System - the satellite network that is run by the US Department of Defense, that enables an object's position to be determined within an accuracy of down to +/- 30 metres.
GSM	Global System for Mobile - a World-wide based standard for mobile telephone communications.
GTP	General Trip Preference - used in several Systems
HGV	Heavy Goods Vehicle
ILOC	Intersection LOCation - a referencing standard
ITS	Intelligent Transport Systems
IVD	In-Vehicle Device - used on the travel guidance System
KAREN	Keystone Architecture Required for European Networks
LLOC	Link LOCation - a referencing standard
MFO	Multi-Functional Outstation - used in the Urban Traffic Control and Public Transport Priority System
MLOC	Movement LOCation - a referencing standard
OBU	On-Board Unit

PASSPORT	Promotion and Assessment of System Safety and Procurement of Operable and Reliable road transport Telematics
P+R	Park and Ride
PC	Personal Computer
PSA	Preliminary Safety Analysis
PT	Public Transport
PTM	Public Transport Management
RDS-TMC	Radio Data System - Traffic Message Channel. Note that these abbreviations are also used on their own, and take their individual meanings.
SIL	Safety Integrity Level
TA	Traveller Assistant - used on the travel guidance System
TARG	Traveller Assistance and Route Guidance - name for “example System”
TCC	Traffic Control Centre
TIC	Traffic Information Centre
TICS	Transport Information and Control Systems
TITOS	Torino ITS 2000 Open Showcase
TSC	Traveller Support Centre
VPS	Vehicle Positioning System - used in the Automatic Road Tolling System

2. Physical Elements and definition of terms

2.1 Introduction

This Chapter describes the physical elements that are used to create each “example System” and their relationship to the Functional Architecture. During the course of this Chapter definitions are also provided of the various terms used in the “example Systems” and their descriptions.

2.2 Physical Elements

2.2.1 Introduction

As noted in the introductory Chapter, a Physical Architecture is the result of grouping together the constituents of the Functional Architecture into physical entities. In the European ITS Framework Architecture, these are called “example Systems”. In general terms a System provides a means of implementing the Functional Architecture to provide Services and may also serve a few particular User Needs that only relate to physical things. The term “example System” is used by the KAREN Project because each System represents an example of how the European ITS Functional Architecture can be used to serve User Needs in a particular group, or groups.

2.2.2 Systems

It is possible to create Systems for a Physical Architecture in many different ways using the constituents of the Functional Architecture. To try and cover all the possibilities would require a very large (and probably un-manageable) document. Thus the approach adopted by the KAREN Project has been to develop a series of “example Systems”. These show how the Functional Architecture can be used to create an example of a particular type, or application, of a System. Therefore they do not provide an exhaustive list of all the Systems that can be created from the Functional Architecture. However it is intended that the “example Systems” in this Document will represent what can be created for those areas of interest to ITS deployment in Europe. Note that in some parts of the rest of this Document, the term “example System” has been shortened to System.

Each “example System” will have a unique name and be numbered. It will also have its own Context Diagram. This Diagram may exclude some of the terminators in the overall European ITS Framework Architecture Context Diagram if they are not required. The overall Context Diagram and all of its Terminators are described in Chapter 4 of this Document. In the description of each “example System” the reasons why any of the full set of Terminators have been omitted will be explained. Each “example System” will normally be complete, though it may have a small number of links to other “example Systems” to provide additional features.

2.2.3 Sub-systems

All "example Systems" will consist of two or more Sub-systems. A Sub-system performs one or more defined tasks and may be offered as a commercial product. Each Sub-system will consist of one or more parts of the Functional Architecture (Functions and Data Stores), and may require to communicate with other Sub-systems and with one or more terminators in order to work. These communications will be provided using Physical Data Flows.

Within the European ITS Physical Architecture each Sub-system will include all of the parts of the Functional Architecture that will exist in the same physical location. Each Sub-system will have its own name plus a number derived from that of its parent "example System" - see section 2.2.6 that follows. The name of each Sub-system can be free form text, ending with the word "Sub-system". The free form text for each Sub-system name should be unique within each "example System".

The number of Sub-systems in a particular "example System" will depend on the number of locations that it covers. The work in the KAREN Project has defined the following as possible locations for Sub-systems.

Central - the place that is used by parts of a System to collect, collate and manage the storage of traffic data, toll payments, freight shipping orders, and/or the generation of traffic management measures, or fleet management instructions, with or without human intervention, e.g. TCC, or TIC, or Freight and Fleet Management Centre.

Roadside - the place that is used by parts of a System for the detection of traffic, vehicles and pedestrians, or the collection of tolls, and/or the generation of traffic management measures, and/or the provision of information and commands to drivers and/or pedestrians.

Vehicle - a device that is capable of moving through the road network and carrying one or more people (bicycles, motorcycles, cars, Public Transport Vehicles) and/or goods (vans and any other form of road going freight carrying vehicle) in which parts of System can be installed during manufacture or can be added to later.

Personal Device - a device in which part of the System can be installed so that it can be easily used (and possibly carried) by Travellers as one of their personal possessions;

Freight Device - a device in which part of the System can be installed so that it is an integral part of a freight carrying unit, e.g. freight container, trailer, or vehicle body;

Kiosk - a device usually located in a public place, into which part of the System can be installed to enable Travellers to have limited and controlled access to some of its facilities.

These locations should be regarded as "generic". They have been chosen because they are unique and are directly relevant to the deployment of ITS. This means that there may be more than one Sub-system in the same overall location. For example, there may be more than one Sub-system within a "Central" location because they can be in different physical places, and the characteristics of data flows between them need to be highlighted. However a Sub-system may not cover two or more different "generic" locations.

Sub-systems that provide the same or similar Services, but in different locations will be treated as two separate Sub-systems, with different names and numbers, e.g. when they contain functionality that may be in either the vehicle or carried by a traveller as a personal device. In this case two different products may have to be developed because the physical requirements such as size, weight and tolerance of ambient conditions will be different.

2.2.4 Modules

A Sub-system may be split into two or more Modules. Each Module will have exactly the same properties as a Sub-system (vis-à-vis its functionality and ability to be a commercial product) and will have its own separate physically identity. The main difference between Modules and Sub-systems (and one reason for their use) is that each Module is more likely to contain functionality from a single Area of the Functional Architecture. Another reason for using Modules is to create physical components that contain a grouping of functionality that is more logical from a manufacturing or physical design view point. Modules will communicate with each other (either within the same Sub-system, or with Modules in other Sub-systems) using Physical Data Flows.

The identity of a Module will be similar to that for a Sub-system, with its number being derived from the "parent" Sub-system - see section 2.2.6 that follows. The name of each Module can be free form text, ending with the word "Module". The free form text for each Module name should be unique within each Sub-system.

2.2.5 Physical Data Flows

Physical Data Flows provide the communication links within each "example System". They enable data to be sent between Sub-systems, between Modules, between Sub-systems and Modules, as well as to and from Terminators. They may consist of one or more Functional Data Flows. Each Physical Data Flow is given a name using the following format:

abcd *free form text*

In this format "abcd" are the initials of the name given to the "example System" to which the Physical Data Flow belongs. The "*free form text*" will be unique within each "example System" and will provide an indication of what the Physical Data Flow contains.

Note that when a Sub-system contains Modules it may be necessary to split the Physical Data Flows to/from that Sub-system into two or more constituent Physical Data Flows to/from the Modules. In this situation the constituent Physical Data Flows are defined as separate entities with their descriptions containing the identities of the Functional Data Flows that they include. The inter Sub-System Physical Data Flow descriptions will identify their constituent Physical Data Flows but will not include the identities of the Functional Data Flows.

2.2.6 Terminator Data Flows

Terminator Data Flows provide the links between the Sub-systems and Modules and the outside World that is represented by the Terminators. Those that have been defined for the European ITS Framework Architecture are described in Chapter 4 of this Document and are the same as those for the Functional Architecture – see Chapter 4 of the Functional Architecture Document (D 3.1).

The definition and naming of the Data Flows should follow part of the convention used for the Terminator Data Flows in the Functional Architecture. The whole convention is described in section 2.3.4 of Annex 2 of the Functional Architecture Document (D 3.1). Only those parts for the Top Level and Low Level Terminator Data Flows should be followed in the Physical Architecture.

The Top Level Terminator Data Flows are those that appear in the Context Diagram. That for the whole European ITS Architecture will be found in Chapter 4 of this Document. There is also a specific one for each "example System" in section 2 of its description. The naming convention that has been used for these Data Flows is:

To Terminator Name
From Terminator Name

Note that only the "generic" Terminators names have been used. These appear in the left hand column of the table of Terminator definitions in Chapter 4 of this Document. The identification of the separate Data Flows to any "actors" in the Terminator is made in the Low Level Terminator Data Flow names.

Low Level Terminator Data Flows provide the final link between the Sub-systems and Modules, and the Terminators. They will be found in the Sub-system and Sub-system Diagrams – see those for the "example Systems" in Annex 1. Their naming convention is as follows:

"t(Terminator Initials) - *free text name*" and
"f(Terminator Initials) - *free text name*".

Note that "t" stands for "to" and "f" stands for "from". The Terminator Initials for each terminator and its actors will be found in the "Table of Acronyms". This is shown at the end of Chapter 4 of this Document.

2.3 Other Entities

The other types of entities that make up a Physical Architecture all into two groups. These comprise Functional Architecture constituents and the Terminators.

The constituents of the Functional Architecture are its Functions, Data Stores and Data Flows. The format and properties of these are described in Chapter 2 of the European ITS Functional Architecture Document – see reference (a) in Chapter 9.

The Terminators for the Physical Architecture will be the same as those for the Functional Architecture. This is because the Physical Architecture is derived from the Functional Architecture, it must therefore contain interfaces to the same Terminators. These Terminators must also have the same "functionality" and hence the same definitions. These definitions have been included in Chapter 4 of this Document, but are a repeat of what is in Chapter 4 of the European Functional Architecture Document.

3. Conventions used for Document creation

3.1 Introduction

This section of Chapter 2 describes the conventions that have been used in the creation of this Document (D 3.2), including Annex 1. They cover the numbering convention used to identify the “example Systems” and the structure of the description of each “example System”. A template for the description structure will be found in Annex 2 of this Document.

3.2 Numbering Convention

As mentioned in the previous sections, the three main elements of the European ITS Physical Architecture (“example System”, Sub-system and Module) have been given numbers. These numbers have been applied according to the following scheme:

System - Pi
Sub-system - Pi.j
Module - Pi.j.k

Sub-systems and Modules may have the same names, but must have different numbers. These numbers are used throughout this Document. The highest level in the numbering hierarchy is that for the “example Systems”. The allocation of these numbers to individual “example Systems” has been organised into groups. Each group covers a particular area of ITS. They are numbered as follows:

P1-P9	- Integrated Traffic Management
P10-P19	- Electronic Fee Collection (no technology)
P20-P29	- Safety and Emergency
P30-P39	- Traffic Management
P50-P59	- Vehicle (not FFM or PTM)
P60-P69	- Traveller Assistance and Route Guidance
P70-P79	- Law Enforcement
P80-P89	- Freight and Fleet Management

The “example Systems” included in each of the above groups are covered by separate Chapters in Annex 1 of this Document. Some of the Integrated Traffic Management “example Systems” (P1 - P9) include facilities that could have been described as separate “Systems”. They have been integrated to provide a more realistic set of “example Systems”. It also reduces the size of this Document from that needed to describe “examples” of each type of System separately.

3.3 Detailed Document Structure

As already noted, this Document has been organised into three parts. The second part (Annex 1), contains Chapters that cover each of the various types of “example System” that have been included in the European ITS Physical Architecture. Within each Chapter one or more “example Systems” may be described. The description of each “example System” uses the same structure. This is as follows:

Overview	- what the "example System" does;
System Context	- includes reasons for omitting terminators from the overall Context Diagram;
Sub-systems	- description of each Sub-system and its constituents;
Sub-Systems and Functions	- for those Sub-systems without Modules, a table showing the Functions that they contain, including the User Needs that they serve, plus a table showing any Data Stores used by the Functions;
Modules	- only provided for those Sub-systems containing Modules and describes each Module including its purpose;
Modules and Functions	- for those Sub-systems with Modules, a table showing the Functions that the Module(s) contain, including the User Needs that they serve, plus a table showing any Data Stores used by the Functions;
Physical Data Flows	- a description of each Physical Data Flow, including a list of its constituent Functional Data Flows;
Key Issues	- things that will affect in some way the deployment of the "example System".

(For ease of reference a list of all the Functions used in the "example Systems", together with their Overview descriptions and User Needs is provided in Annex 2 of this Document.)

All the Key Issues from each "example System" in Annex 1 have been collected together to form separate sections in Chapter 5. This is intended to make it easier for these Issues to be used by other groups within the KAREN Project, or by other organisations.

Note that only one table of Functions will be provided for each "example System". Thus those "example Systems" having some Sub-systems with Modules and some without, will have one table of Functions. This will appear in the section on Modules and Functions and will contain details of the Functions included in all Sub-systems, including those that do not contain any Modules. The same principles will apply to tables of Data Stores.

A template for the above Document structure is provided in Annex 2 of this Document. It is intended for use by those who wish to create their own "example System" using the methodology described in Chapter 3 of this Document.

3.4 System Specifications

3.4.1 Introductions

The purpose of the European ITS Physical Architecture, has been to show how the European ITS Functional Architecture can be used to create "physical" Systems. The end point of this work has been the highlighting of "Key Issues" that will need to be considered if such a System were implemented. In practice, the actual end point would need to be a System Specification.

The System Specification should enable the procurement of the System and its Sub-systems. It should describe what the System shall do and the requirements that it shall meet. There will be section(s) describing and specifying each Sub-system including the Modules that each contains. Additionally, the Specification must also include various other things such as how and when the System shall be tested, what documentation shall be provided, etc.

3.4.2 Specification Structure

A System Specification therefore needs to be structured in such a way that all of the above elements are included. It should therefore conform to the following template:

(1) System Name

(2) Overview Description: a textual overview of what the ITS services that the System is going to deliver. This should be written in “shall” language.

(3) Component Parts: A list of the Sub-systems with their names.

(4) Sub-system Descriptions: a set of descriptions for each Sub-system to include the following.

   (a) Overview: a general description of the ITS services that the Sub-system will deliver. Again this should be written in “shall” language.

   (b) List of Modules: covers names only.

   (c) List of the European ITS User Needs with which the Sub-system shall comply, but taken from Groups 2-10 only. This should be the numerical identification of each User Need that is served by the Sub-system.

   (d) List of the European ITS User Needs in Group 1 with which the Sub-system must comply. Note that not all the categories and individual User Needs in this Group will apply to each Sub-system. They should be selected on a “per Sub-system” basis to ensure that they are entirely relevant.

   (e) Other Requirements: these should cover, but not be limited to the following areas, Test Specifications, Acceptance Tests (Factory and/or Site), Installation and Commissioning, Documentation, etc. These Requirements can contain more detail than is in any of the User Needs. They can also address areas that are outside the User Needs, such as precise details of Traveller interfaces, sources of geographic data, etc. Note that it is unlikely that any User Needs from Group 1.1 (Architectural Properties) will be required by any Sub-system or Module.

(5) System Requirements: these are requirements that must be met by the System as a whole and are in addition to those that must be met by each of the Sub-systems. They should cover, but not be limited to the following areas, Test Specifications, Acceptance Tests (Factory and/or Site), Installation and Commissioning, Documentation, etc. Note that is possible for the overall System requirements and those of the individual Sub-systems to cover the same areas. This is because there may be different parts of these areas that are specific to some Sub-systems, and to the System as a whole.

In some cases it may be necessary to provide separate Specifications for each Module in a Sub-system. This is most likely to be necessary when the complete set of User Needs served by the Sub-system come from different Groups, or it is intended to source the complete supply of one or Modules from several suppliers. For example, a Sub-system that covers both traffic and public transport management will need to have the Central, Roadside and Vehicle Modules described separately. As a minimum, the scope of these separate (Module) descriptions must include separate lists of the User Needs (covering both Groups 2-10 and 1) served by each Module.

The above System Specification should not be taken as the complete list of contents for a System Procurement Document. Instead it should be included in the complete documentation that is assembled for System procurement. This is because the Specification does not attempt to cover aspects of procurement such as Legal Requirements, Payment Terms, milestones for the completion of certain activities, Penalty Clauses, etc. These have not been included as they are outside the scope of the KAREN Project. Also different standards and requirements for these parts of procurement documents will exist in each Nation, State, Province, Department, etc.

An example of a System Specification using the contents and scope described above has been included in Chapter 4 of Annex 2 to this Deliverable Document. Only one example has been produced, because it is all that is needed to cover the main points concerning the creation of System Specifications.

4. Methodology for the creation of Physical Systems

4.1 Introduction

This Chapter describes the process that can be followed to create a Physical System such as the "example Systems" in the European ITS Physical Architecture – see Annex 1 to this Document. These have been provided so that users of the European ITS Framework Architecture can see "worked examples" of how to create their own "example Systems" if none of those in this Document exactly fulfils their requirements.

Note that the term "example System" has been used since it will not necessarily be the definitive way that the Functional Architecture can be used to create something physical. It is possible to create many different Physical Systems from the European ITS Framework Architecture, and each one may fulfil the same or a different range of European ITS User Needs.

4.2 Why create a Physical System ?

A Physical System is created to provide a physical implementation of the European ITS Functional Architecture. It will satisfy some or all of the European ITS User Needs contained in Groups 2 to 10 – see reference (b) in Chapter 9. The development of such a System will be made because the selected European ITS User Needs can provide the ITS services that will be required from a particular System e.g. to serve the Incident Management group of User Needs.

The scope of any Physical System will be dictated by the scope of the User Needs that it is being required to serve. It is however possible to apply the methodology in this Chapter to any size of System. It can be anything from a System to provide traffic information to drivers using a small number of signs, to a large System to manage traffic and Public Transport in a metropolitan area. The only thing that will change is the scope of the work involved in the System creation.

Creating a large System should also be seen as a way to identify some of the issues that may arise from its implementation. These issues may be related to such things as the integration of the component parts, and organisation of the operation of the completed System by the Authorities charged with its management. The creation of a large System should enable those owning and managing the System to see the "Big Picture" that will be created when it is put into operation.

It is possible for new (additional) User Needs and functionality to be added to those in the European ITS Framework Architecture. For a Physical System, these should be small in number and scope. Typically they will be minor expansions of what has been provided. The way in which these additions can be made is discussed in later sections of this Chapter.

If a large number of new (additional) User Needs are to be added, and they alter (extend) the scope of the ITS services, then it may be more prudent to consider the creation of a new Architecture. This is discussed and described in the next Chapter.

4.3 Pre-requisites for the creation of a Physical System

The creation of a Physical System has several pre-requisites. These are as follows:

(1) a list (or an idea) of the European ITS User Needs that must be served;

(2) the description for each of the Low Level Functions, all of the Data Flows and all of the Data Stores that make up the European ITS Functional Architecture ;

(3) Trace Tables of User Needs and Functions.

Everything in these three items will be found in the Appendix F (spreadsheet) to the European ITS User Needs Document (1), Annex 1 of the European ITS Functional Architecture Document (2) and Annex 1 of the European ITS Framework Architecture Overview Document (3). References for these Documents will be found in Chapter 9 of this Document.

Note that it is not necessary to include any of the High Level Functions in (2) above. This is because it may be necessary only to use some (and not all) of the Low Level Functions that they contain.

4.4 Method for the creation of a Physical System

This method can be used to create a Physical System. The creation is achieved using the following steps.

1. Assign a name to the proposed System. It should not be necessary to use the "Pi" convention described in Chapter 3 of this Document for the System identification. The alternative that is suggested is that it be replaced by the initial letters of the chosen System name. Thus for example, a System for urban traffic management could have the identification "utm", whilst a System for the "CETE de Lyon" could have the identification "CdL".

2. Go through the European ITS User Needs – see Appendix F of reference (b) in Chapter 9, and select those Needs that are to be served by the proposed "example System" i.e. the Physical System. These Needs should reflect the policies and goals of the proposed System. All the selected Needs need not to come from the same Group or Category. There can be some needs from several categories.

3. It may be that not all the policies and goals of the proposed System can be satisfied by the European ITS User Needs. For example they may not contain enough detail, or there may be certain things that are particular to the proposed System. To capture these specific policies and goals accurately, it will be necessary to some create new User Needs. This should be done now, and the mechanism for doing this is discussed in more detail in section 4.6.

4. Get all the stakeholders for the proposed Physical System to review the selection of User Needs to make sure that they agree with all the services that will be provided. As part of this process, produce one or more of the Models listed below. They can be used to ensure that all stakeholders understand the proposed Physical System in the same way.

(a) Enterprise Model – shows the structure of the relationships that exist between organisations, persons, services and/or functionality.

(b) Primary Process Model – shows the processes that will take place, and their relationship with the environment outside the system, e.g. users.

(c) Layered Reference Model – shows a division of the system functionality into dependant layers.

(d) Conceptual Model - a high-level description of the relationship between the main functions and, possibly, their physical locations.

Examples of Models for ITS deployment and their use will be found in reference (f) of Chapter 9. Working with the some or all of the above Models will give a better understanding of what services the User Needs will provide. They will also help to determine what should go into the System in the following Steps.

5. From the Trace Tables – see Annex 1 of reference (c) in Chapter 9, select the Low Level Functions that fulfil the User Needs and Models identified in Steps 3 and 4 above. This can be achieved by studying the Table in Annex 1 that shows the Functions that serve each User Need. Also look through the Data Flow Diagrams in the Main Document of reference (a) and identify the Data Flows that link the Functions and Data Stores. This information can also be found in the detailed Functional descriptions in Annex 1 of the Document in reference (a).

6. If new User Needs were created in Step 3, then it may be necessary to provide new Functions to serve them. Even if there are no new User Needs, the Functions selected in Step 5 need to be analysed to make sure that no extra functionality is being included that will not be needed. The reasons for this and the process for creating new functionality will be discussed and described in section 4.7.

7. Select the physical locations that will need to be used by the Functions in the Physical System. There is a choice of six "generic" locations and these are described in section 2.2.3 of this Document.

8. Allocate each of the Functions selected in Steps 5 and 6 to one of the locations identified in Step 7. The Functions at each location will be a separate Sub-system which should be given a number and a name. It is suggested that the numbering convention similar to that used for the "example Systems" in this Document be followed - see section 2.2.7. The generalised form of the numbering convention for each Sub-system should be as follows:

   "System Identification".n

   If the convention described in Step 1 is followed then examples of Sub-system numbers would be "UTM1", "UTM2" etc., or "CdL1", "CdL2" etc. The Sub-system name can be any "free form" text, but it should end with the word "Sub-system".

9. If any of the Sub-systems contain more that one Function, decide whether to divide the Sub-system into two or more Modules. Using Modules enables parts of a Sub-system to be provided separately. This has some advantages, including ease of maintenance, and improving the ability to source the Modules from more than one supplier. The

main disadvantage is that it may be difficult to get any one supplier to take overall System responsibility. As an aid to the decision on whether or not to use Modules, the following can be used as a guide.

(a) If the Functions in one location are from different Functional Areas then they should be in separate Modules. For example, there should be separate Modules for Functions from Provide Electronic Payment Facilities Area (1) and Mange Traffic Area (3) in a road based Toll Collection System.

(b) If the Functions in one location are from different High Level Functions within the same Functional Area, they may need to be in separate Modules. For example, separate Modules should be used for Functions from the Incident Management and Traffic Management High Level Functions in the Manage Traffic Area (3)

Each Module that is created should be given a number and a name. The number should follow the convention used for Module numbers in the “example Systems” and described in section 2.2.7 of this Document. If the convention described in Steps 1 and 8 is followed then examples of Module numbers would be “UTM1.1”, “UTM1.2” etc., or “CdL1.1”, “CdL1.2” etc. The Module name can be any “free form” text, but it should end with the word “Module”.

10. Produce a list of the Functions in each Sub-system. If any of the Sub-systems include Modules, then there should be a list of Functions for each of the Modules and not for the Sub-system(s) of which they are a part. It is recommended that the lists should include the User Needs served by each Function. The list should be in the form of a table, such as those that have been produced for each “example System” – see Annex 1 of this Document. Templates for this table will be found at the end of Annex 2.

11. Identify any Data Stores in the Functional Architecture that need to be included in each Sub-system (and Module). They can be identified by looking at Annex 3 of the Functional Architecture Document (D 3.1). Find the Chapters that contain the descriptions of the Data Stores in each of the Functional Areas from which the Functions identified in Steps 5 and 6 have been selected. The following points should be noted.

(a) The numbering convention for Functions and Data Stores is such that those in the same Area will start with the same digit. Thus Data Stores in Area 2 will be numbered D 2.x.

(b) The opening sections of each Chapter of Data Store descriptions in Annex 3 for the Functional Architecture Document (D 3.1) contains the relationship between the Data Stores and Functions, i.e. the identity of the Function(s) that use each Data Store.

(c) Some Data Stores may be shared between different Functional Areas. For example, one of the Data Stores in the Provide Electronic Payment Facilities Area (1) is shared with Functions in the Provide Support for Law Enforcement Area (7). Again this information will be found at the start of each Chapter in Annex 3 of the Functional Architecture Document (D 3.1).

Include the Data Stores in the list of Functions produced by Step 10 – a separate table can be used for the Data Stores as shown in some of the "example System" descriptions in Annex 1 of this Document.

12. Identify the Terminators with which the System has links. This can be done by identifying the Terminators that link to the Functional Areas to which the Functions identified in Steps 5 and 6 belong. As an aid to this process, a table showing the Terminators used by all the Functional Areas is included at the end of Chapter 4 of the Functional Architecture Document – see reference (a) in Chapter 9 and in Chapter 6 of this Document. Using this table, the Terminators used by each of the Sub-systems and Modules can be identified. It must be remembered that the interfaces for each Terminator will be part of the Functions included in the Sub-systems and Modules.

13. Draw diagram(s) to show the Sub-systems (and Modules within each Sub-system). The diagrams should be similar to those produced for each "example System" in Annex 1 of this Document. They should comprise the following.

    (a) Context Diagram: this shows the links (Data Flows) that the System has with the outside World as represented by the Terminators.

    (b) System Diagram: this shows the Sub-systems, the Data Flows between them, and between each Sub-system and the Terminators.

    (c) Sub-system Diagram: this shows the Modules in a Sub-system and the Data Flows between them. It should also show the Data Flows to and from Terminators, although it is not necessary to show the Terminators themselves as in the other two Diagrams.

    Each Diagram should show the flow of data between the Sub-systems and Modules as single directional lines with an arrow at the destination end. The lines will represent Physical Data Flows which are defined using the method in the next Step. Advice on how to create these Diagrams is provided in the next section of this Chapter.

14. Each Physical Data Flow that appears on any of the Diagrams in the previous Step should be given a unique name and description. It is suggested that the name should follow the convention that has been used for the Data Flows in the "example Systems" described in Annex 1 of this Document. This convention is described in section 2.2.5.

15. Complete the description of each Physical Data Flow by listing its constituent Functional Data Flows. These will be the Data Flows that link the Functions in the different Modules and Sub-systems and they will have been identified in Step 5. Note that the Functional Data Flows that are between Functions within a Sub-system or Module, need not be identified as they do not appear in any of the Diagrams identified in Step 10 above.

16. The Data Flows to and from the Terminators in the Diagrams identified in Step 13 should also be given names. The naming convention should be the same as that used for the Terminator Data Flows in the "example Systems" and described in section 2.2.6 of this Document.

17. Produce a description of each Sub-system (and Module). It is recommended that the starting point for this description is the Overview description for each of the constituent Functions.

18. If it is intended to proceed with the implementation of the System, then a System Specification must be produced. The format of such a Specification is discussed in Chapter 3, and an example is shown in Annex 2 of this Document. It will also be necessary to carry out an analysis of the communications infrastructure requirements for the System. The methodology described in the European ITS Communications Architecture Deliverable Document – see reference (e) in Chapter 9 should be used. Some examples of the using this methodology are provided in this Document with tables of values for communications bandwidth shown in Annex 1.

19. In cases where the System will replace all or part of an existing System (or Systems), a Deployment Strategy will need to be produced. This and other deployment issues are discussed in the European ITS Deployment Approach and Scenarios Document – see reference (d) in Chapter 9.

The Physical System that has been created will need to be documented. It is recommended that this is done as the above Steps are carried out. A template for use in the creation the documentation has been provided and is discussed at the end of the next section (4.5) of this Chapter. Use of this template will also enable the descriptions and Diagrams produced as a result of the above Steps to be included in the documentation as and when they are available.

4.5 Aids to the creation of a Physical System

4.5.1 Introduction

As an aid to using the method described in the previous section, a set of Microsoft® Access Databases are available to prospective users of the European ITS Framework Architecture. They will be provided on the European ITS Framework Architecture CD-ROM with the current (Issue 1, August 2000) Functions and Data Flows already included.

The Databases are designed to be used for the allocation of Functions and Data Stores to Sub-systems and Modules and for the development of Physical Architecture Data Flows. Ultimately they will contain the information about the make-up of the Sub-systems, Modules and Physical Data Flows. Some templates for the Diagrams are also provided.

4.5.2 Sub-systems and Modules Database

The Sub-systems and Modules Database contains a table which includes details of all of the Low Level Functions in the European ITS Functional Architecture. In addition, extra columns are included so that the details of the Sub-systems and Modules for each Function can be entered. The actual structure of the Database is illustrated by the table that is shown on the next page.

Note that the table is only provided for illustrative purposes only. A template for a table of this format, used in the documentation of the “example Systems” in Annex 1 of this

Document is provided in Annex 2. The use of this table in the documentation is discussed at the end of this section (4.5) and also in Annex 2.

Table 1 Database Table format for Functions in "example System" Development

Sub-System			Module		Function		
Number	Name	Location	Number	Name	Number	Name	User Needs

The columns for the Function will already be filled in with the details of all the Low Level Functions in the European ITS Functional Architecture. All other columns will be left blank for use in Steps 5and 6 of the previous section.

Data can be entered into the “Forms” facility provided by Microsoft® Access and included in the Database. Users of the European ITS Framework Architecture should use the “Forms” to enter details of the Sub-systems and Modules in their particular Physical System. This should start with the selection of the Functions in each physical locations as described in Steps 4 and 5 of the previous section. Names should then be given to the Sub-system for each location by entering them (and the Sub-system numbers) in the rows for each location.

Next, the entries for the un-used Functions should be removed from the Database. If Modules are needed for the reasons described in Step 6, then their numbers and names should be entered in the rows for the Functions that they will contain. Note that Modules cannot belong to more than one Sub-system, but that there can be several similar Modules in each of the Sub-systems

A basic “Report” will also be included in the Database to enable the above table to be printed. However this may need to be customised to suit individual users’ requirements. It may also need to be converted into a Table. Templates for Sub-systems with and without Modules are provided in Annex 2 of this Document.

A similar Database is provided for Data Stores. It has a table like that shown above, except that Data Store information from the Functional Architecture only comprises Number and Name. Again a basic “Report” is also included in the Database to enable the table to be printed. However this may need to be customised to suit individual users’ requirements. It may also need to be converted into a Table and a template in the correct format is also provided in Annex 2 of this Document.

Illustrations of the way in which the Function tables should appear when printed are contained in the descriptions of each of the “example Systems” provided in Annex 1 of this Document. Some descriptions also contain illustrations of the Data Store tables.

4.5.3 Physical Data Flows Database

The Database for the Physical Data Flows has a table with a completely different structure to that shown above. It contains columns for details of the Physical Data Flows plus their constituent Data Flows and is illustrated by the table on the next page.

As with the previous table note that it is only provided for illustrative purposes only. Unlike the previous table however, it is not actually used in the documentation of the "example Systems" in Annex 1 to this Document.

Table 2 Database Table format for Data Flows in "example System" Development

Physical Data Flow		Constituents	
Name	**Description**	**Physical Data Flows**	**Functional Data Flows**

None of the columns in this table contain pre-defined information about the Functional Data Flows. In this case the users have to provide all of the information themselves. The source of information about Functional Data Flows is Annex 2 of the European ITS Functional Architecture Deliverable Document - see reference (a) in Chapter 9. Note that if a Physical Data Flow contains constituent Physical Data Flows, then each of these constituents should also be defined as separate entries. The Functional Data Flows should only be shown for Physical Data Flows that have no constituents. Both methods of data entry described for the Sub-systems and Modules Database are available for this Database.

4.5.4 Documentation

It is strongly recommended that the documentation of any new Physical System follows the structure that has been provided in this Document. This should ensure consistency with this Document and that nothing important has been missed. A template of the structure used for the description of each "example System" in Annex 1 of this Document, can be found in Annex 2.

Templates for the Diagrams described in Step 10 of the previous section in this Chapter are also provided in Annex 2 of this Document. They have been created using the VISIO® drawing package. This package will be needed to modify these templates for use in the creation of Physical Systems.

4.6 Adding extra User Needs

During the course of creating some Physical Systems, it may become necessary to add or create new User Needs. The requirement for any new User Needs will appear during the course of work on Step 2 in section 4.4. It may occur for one or more of the following reasons:

(a) the scope of the ITS services that the System will provide is wider than those in the European ITS User Needs;

(b) the European ITS User Needs have the correct scope, but do not have enough detail.

Either of the above can be accomplished by adapting the European ITS Framework Architecture User Needs. They have been structured in such a way that additions can be made to some parts without invalidating the remainder. Thus any additions can be accomplished within their existing numbering scheme. These additions may require extra Groups, or Categories, or individual Needs, or a combination of all three.

Extra Groups can be added starting with Group 11. Any new Group should follow the same structure as found in the other Groups, although in the first instance it may only contain one Category. Thus the first actual User Need will have the number, 11.1.0.1. Similarly, extra Categories and other levels of User Need can be added within each Group of User Needs. When adding a new Category, it should take the next available number in the sequence that already exists within the Group. A new User Need should take the next available number within the Category. More information is provided about this in the European ITS User Needs Document – see reference (b) in Chapter 9.

Note that it is very important that any new User Needs do not have the same numbers as any existing User Needs, even if they are not being included in a particular Physical System. This is to avoid confusion between new and unused User Needs, which may in fact be used by other Systems and Architectures. In other words the new User Needs should have new numbers within the existing numbering scheme.

4.7 Adding extra Functionality and Terminators

4.7.1 Introduction

During the course of creating some Physical Systems, it may become necessary to add or create new functionality. The requirement for any new functionality will appear during the course of work on Steps 5 and 6 in section 4.4. It may be for one or more of the following reasons:

(a) extra User Needs have been added, as described in the previous section;

(b) some of the Functions in the European ITS Functional Architecture need to be split up so that only part of their overall functionality is included in a Sub-system or Module.

Both of these can be achieved by adapting the European ITS Framework Architecture. The Architecture has been designed to enable additions to be made to some parts without invalidating the remainder. Additions may be in the form of extra (new) functionality and new Terminators. The ways in which these additions can be achieved are described in a following section.

Note that (b) above should not be a result of the addition of any new User Needs. Instead it is a recognition that during the development of the European ITS Functional Architecture, some assumptions were made about the level of detail (sometimes called the Functional Decomposition) that would be required. For many Physical Systems what exists will be sufficient, however for some, it may be necessary to split an existing Function into two or more Low Level Functions. The purpose of such as exercise will be to prevent the Physical System including un-necessary (unused) functionality.

4.7.2 Adding new Terminators

As a result of adding new User Needs it may be necessary to add a new Terminator. It is most strongly recommended that any new Terminator should be an actor within one of the existing Terminators. This should minimise the scope of the changes, since the Top Level (Context) Terminator Data Flows will remain the same. The Medium Level Terminator Data Flows need only be changed if the new functionality is in an Area that does not currently have access to the Terminator to which the extra actor has been added. When a new actor is added to an existing Terminator, the Table of Acronyms (see Chapter 4 in reference (a) of Chapter 9 and Chapter 6 of this Document) must be modified to include the Data Flow identification for the new actor.

The result of adding a new Terminator will be that changes to the functionality will be required. This can be achieved by the work described in the next section. Note that in addition to the work described in the next section, all the DFD's for the Functional Area containing the interface to the new Terminator must also be modified to include the new Terminator Data Flows.

4.7.3 Adding new Functionality

Adding new functionality to the European ITS Functional Architecture can be accomplished within the existing hierarchy of Functional Areas, High Level Functions and Low Level Functions. The method used depends on how much functionality is being added, and may therefore be one of the following.

(a) Modifying an existing Function: change the details such as Overview description, input/output Data Flows, Functional Requirements and User Needs by editing the information in Annex 1 of the European ITS Functional Architecture – see reference (a) in Chapter 9. If any new Data Flows are added, include their (new) descriptions in Annex 2 and modify the DFD(s) in which the Function appears to show the new Data Flow. These Data Flows will also need to be added to the definitions of the Physical Data Flows.

(b) Adding a new Function: use an existing Function description in Annex 1 as a template to create the description of the new Function. Its number should be the next available number in the sequence for the High Level Function or Functional Area to which it will belong. When complete, add the description to those in Annex 1. If a new Low Level Function is being added, modify the description of the High Level Function to which it belongs in Annex 1. Modify the relevant DFD(s) to include the new Function. Add descriptions of the (new) Data Flows that link the new Function to other Functions to Annex 2. If new Data Stores are needed, then add them to the DFD(s) and their descriptions to Annex 3. The new Function must now be added to the list of those in the Sub-system or Module. Any new Data Flows will become constituents of Physical Data Flows if they are cross Sub-system or Module boundaries.

(c) Splitting an existing Function: the process steps to be followed to split an existing Function are more complex than those described in (a) and (b) above. They have therefore been separated as follows.

(1) Use the description in Annex 1 of the Function that is to be split to create the descriptions of the new Functions that are to be its constituents. The existing Data Flows, Functional Requirements and User Needs will need to be split between the new Functions. A new Overview description will have to be written for each Function. The numbers for the new Functions must be derived from that of the Function that is being split. For example, if Function F 3.1.1.3 is being split, the (new) constituent Functions will be numbered F 3.1.1.3.1, F 3.1.1.3.2, etc.

(2) Modify the DFD in which the existing Function appears so that a “(d)” appears underneath its name. This is used to indicate that it is a High Level Function for which a DFD exists.

(3) Create the DFD for the High Level Function taking care to ensure that the Data Flows that are linked to the Function in the existing DFD appear in the new DFD. Note that new Data Flows may be needed to link the two Functions together. The existing Data Flows may also have to be split up into (new) constituent Data Flows and allocated to the new Functions.

(4) Add any new Data Flow descriptions to Annex 2, adding the new Data Flows as constituents of the existing Data Flows.

(5) Modify the Hierarchy Diagram for the Functional Area in Annex 2 to show the new Functions and where they fit into the hierarchy.

The new Functions must now be added to the list of those in the Sub-system or Module. Any new Data Flows will become constituents of Physical Data Flows if they are cross Sub-system or Module boundaries.

In all of the above, the current conventions for Data Flow names must be followed when creating new Data Flows. These are described in Chapter 2 of Annex 2.

Note that it is very important that any new Functions, Data Flows, or Data Stores do not have the same numbers or names as any existing ones, even if they are not being included in a particular Physical System. This is to avoid confusion between those that are new and those that are unused, as they may in fact be used by other Systems and Architectures. In other words the new Functions, Data Flows, or Data Stores should have new numbers and names within the existing numbering scheme.

4.7.4 Using the Functional Architecture Database

There is an alternative to using the textual descriptions of Functions, Data Flows and Data Stores in Annexes 1, 2 and 3 of the European ITS Functional Architecture Document (D 3.1), as templates for any additions. This is to use the European ITS Functional Architecture Database.

This Database is provided as one of the items on the European ITS Framework Architecture CD-ROM. It is a Microsoft® Access Database and so should be available to most users. It contains several tables that are used to hold the data for High Level Functions, Low Level Function, all types of Data Flows and Data Stores.

Also included in the Database are “Forms” for each of the tables. These will enable the descriptions of new Functions, Data Flows and Data Stores to be added in a structured way without the need to worry about the layout of the text. It will also prevent duplicate numbers and/or names being used for Functions, Data Flows and Data Stores.

The Database also includes “Reports” for some of the tables. These are intended to enable the Function, Data Flow and Data Store descriptions to be created as part of a Microsoft® WORD document.

The structure and use of the Database are described in more detail by part of Chapter 15 in the European ITS Functional Architecture Main Document – see reference (a) in Chapter 9. This should be read and understood before any changes are made.

5. Methodology for the creation of an Architecture

5.1 Introduction

This Chapter describes the process that should be followed to create an “example System” that is an Architecture in its own right. It will not contain a physical implementation of the functionality in the European ITS Functional Architecture, as it will have its own Physical Architecture.

The main reason for creating an Architecture as opposed to a Physical System are that the User Needs that are to be served, and hence the ITS services that are to be provided, are very different in scope. Such an Architecture should be designed to form the basis of the development of its own “example Systems” and this may be another reason for developing a new Architecture.

The initial processes used to create an Architecture are the same as those for a Physical System. It is only when the functionality has been selected from the European ITS Functional Architecture that the differences occur. They arise because the creation of a new Architecture will lead to the creation of its own Physical Architecture. From this its own Physical Systems can be created using the methodology described in Chapter 4.

5.2 Architectures

It is possible to create other (different) Architectures from the European ITS User Needs and Functional Architecture. The result of such as exercise will be a different version of the European ITS Framework Architecture. Such an Architecture will be designed to fulfil a particular requirement for the provision of ITS services. Typically these requirements will have been set by organisations at a level where they control ITS Nationally, or within a State, Province, Department, District, County, City, etc.

It is possible to create a different version of the European ITS Framework Architecture in one of two ways. These are as follows.

(1) A sub-set of the European ITS Framework Architecture. This Architecture will use some Groups of the European ITS User Needs and therefore only contain some parts of the overall Architecture. For example, it can be used to create an Architecture from which Systems providing Traffic and Public Transport Management can be created. Some of these Systems may include facilities for the Management of Incidents or Demand Responsive Public Transport, others may not.

(2) A super-set of the European ITS Framework Architecture. This Architecture will again use some Groups of the European ITS User Needs and therefore only contain some parts of the overall Architecture. However it will also contain additional User Needs that deliver a particular service, or services. For example there might be a need to provide services that are related more to a particular type of transport, e.g. cable cars, or waterborne ferries that are only found in selected parts of Europe. The links to Other Modes that are provided by the European ITS Framework Architecture may be

too generalised to make it possible to integrate these modes of travel in such services as Trip Planning, Fee Collection and Demand Management.

In both of the above examples, the final selection of European ITS User Needs that is included in the Architecture must always include those in Group 1. In the first example it may be possible to exclude some individual User Needs, but this should only be done after careful consideration. All the User Needs in Group 1.1 (Architecture Properties) must always be included in both types of Architecture.

5.3 Pre-requisites for creating an Architecture

The creation of each Architecture has several pre-requisites. These are as follows:

(1) a list (or an idea) of the European ITS User Needs that must be served;

(2) the description for each of the Low Level Functions, all of the Data Flows and all of the Data Stores that make up the European ITS Functional Architecture;

(3) Trace Tables of User Needs and Functions.

Everything in these three items will be found in the Appendix F (spreadsheet) to the European ITS User Needs Document (1), Annex 1 of the European ITS Functional Architecture Document (2) and Annex 1 of the European ITS Framework Architecture Overview Document (3). References for these Documents will be found in Chapter 9 of this Document.

Note that it is not necessary to include any of the High Level Functions in (2) above. This is because it may be necessary only to use some (and not all) of the Low Level Functions that they contain.

5.4 Method for the creation of an Architecture

In this method the starting point is the desire to create an "example System" that is an Architecture. In view of this, it is essential that the actions described in the European ITS Deployment Report are followed – see reference (d) in Chapter 9. The actual Architecture creation process is described in the following steps.

1. Assign a name to the proposed Architecture. It should not be necessary to employ the "Pi" convention used for the "example Systems" in Annex 1 of this Document and described in section 3.2.

2. Go through the European ITS User Needs – see Appendix F of reference (b) in Chapter 9, and select those Needs that are to be served by the proposed Architecture. These Needs should reflect the policies and goals of the proposed Architecture. All the selected Needs need not to come from the same Group or Category. There can be some needs from several categories. Get all the stakeholders of the proposed Architecture to review the selection of User Needs to make sure that they agree with all the services that will be provided.

3. At this point there should be a review of the selected User Needs to ensure that they will meet the ITS goals of the Nation, State, District, Province or County. If there are

gaps, i.e. goals that are not served by the selected User Needs, then new ones must be created and added to those selected in Steps 2 and 3. The method for doing this is described in section 4.6

4. It is strongly recommended that the complete set of User Needs produced by Steps 2 and 3 are reviewed by all the relevant stakeholders. This and other similar actions that should be taken to develop an Architecture are described in the European ITS Deployment Report – see reference (d) in Chapter 9.

5. As an aid to the process in Step 4, produce one or more of the Models listed below. Use them to help all the stakeholders to look at and understand the proposed Architecture in the same way.

   (a) Enterprise Model – shows the structure of the relationships that exist between organisations, persons, services and/or Functions.

   (b) Primary Process Model – shows the processes that will take place, and their relationship with the environment outside the system, e.g. users.

   (c) Layered Reference Model – shows a division of the Architecture functionality into dependant layers.

   (d) Conceptual Model - a high-level description of the relationship between the main functions and, possibly, their physical locations.

   Examples of Models for ITS deployment and their use will be found in reference (f) of Chapter 9. Working with the some or all of the above Models will give a better understanding of what services the User Needs will provide. They will also help to determine what should go into the Architecture in the following Steps.

6. From the Trace Tables – see Annex 1 of reference (c) in Chapter 9, select the Low Level Functions that fulfil the User Needs identified in Steps 2 and 3 above. This can be achieved by studying the Table the Annex that shows the Functions that serve each User Need.

7. Identify the High Level Functions to which each of the Functions selected in Step 6 belong. This can be done by looking at Hierarchy Diagrams for each Functional Area that are included in Annex 1 of the Functional Architecture Document – see reference (a) in Chapter 9. Follow the hierarchy upwards so that all the functionality up to and including the Functional Area are identified.

8. If any new User Needs were created in Steps 2 and 3, then new functionality and (possibly) Terminators will be needed to serve them. Even if there are no new User Needs, the Functions selected in Step 6 need to be analysed to make sure that no extra functionality is being included that will not be needed. The reasons for this and the process for creating new functionality will be discussed and described in section 4.7. The way in which new Terminators can be added to the Architecture is also discussed in section 4.7.

9. From this point on, the following actions are designed to create information that is a sub-set of that produced for the European ITS Framework Architecture. Therefore the following Documents and other sets of information must be created:

(a) Architecture User Needs List: this should contain all of the European ITS User Needs identified in Step 2, plus those produced in Step 3. The draft version of this List must be used for the stakeholder review identified in Step 4. The final (definitive) version will be the end result of Step 4.

(b) Functional Architecture Document: this should be a "customised" version of the European ITS Functional Architecture Document – see reference (a) in Chapter 9. The European Document should be edited to remove the High Level Functions (and possibly Functional Areas) not identified in Step 6. The Annexes for the European Document should also have the un-used Functions, Data Flows and Data Stores removed. The Hierarchy Diagrams in Annex 1 should either be deleted (unused Functional Areas) or modified to show the structure of the High and Low Level Functions that the Architecture contains. Note that new Diagrams (DFD's) may be required as a result of the work required in Step 7.

(c) Physical Architecture Document: this should be produced in a similar form to this Document. The most important part will be the equivalent of the "example Systems" in Annex 1. It is recommended that the range of "example Systems" be restricted to a small number, but this will depend upon the scope of the User Needs that have been agreed in Step 4. Another possibility is to follow the Steps for the creation of an "example System" that is capable of physical implementation. This is described in Chapter 4.

10. The need to produce Architecture Documents similar to others of those produced for the European ITS Framework Architecture will depend on the scope of the Architecture and the ITS goals that are being served. If the practicality of developing Systems from the Architecture is the main consideration, then a communications analysis will be required. The methodology in the European ITS Communications Architecture Document should be followed to perform this analysis – see reference (e) of Chapter 9. The Document also includes examples of the data needed for such analysis and these will be found in its Annex 1.

The scope and need to perform other activities as part of the creation of the Architecture is covered in the European ITS Deployment Report – see reference (d) in Chapter 9. This should be studied in some detail before embarking on the creation of an Architecture.

5.5 Adding new Terminators and Functionality

The structure of the functionality in the European ITS Functional Architecture is such that new Functions (both High and Low Level) can be added without affecting the numbering of the retained (or unused) Functions. In some cases, the scope of the new functionality that is required may lead to the creation of a new Functional Area, in which case its number needs to be 9 or above.

The actual processes by which new Terminators and functionality can be added to the existing European ITS Functional Architecture is described in section 4.7. Although this methodology is in the Chapter covering the creation of Physical Systems, it can equally be applied to the creation of an Architecture. For Architecture creation it is very strongly recommended that the Functional Architecture Database is used to make the additions. This will provide better control of the changes and will enable the documentation required in Step 9(a) of the previous section to be produced. Its use is discussed in section 4.7.4.

As has been noted before, it is very important that any new Functions, Data Flows, or Data Stores do not have the same numbers or names as any existing ones. This should be true even if some of the existing ones are not being included in the new Architecture. This is to avoid confusion between those that are new and those that are unused, as they may in fact be used by other Systems and Architectures. In other words the new Functions, Data Flows, or Data Stores should always have new numbers and names within the existing numbering scheme. The conventions for Data Flow names are described by Chapter 2 in Annex 2 of the European ITS Functional Architecture – see reference (a) in Chapter 9.

5.6 Modifying the Physical Architecture Database

Following the work in the previous section, the Physical Architecture Database described in part of section 4.6 will need to be modified. This will enable it to be used in the creation of "example Systems" that are Physical Systems from the new Architecture. This process would actually use the methodology that has been described in Chapter 4. The modified version of the table of Functions in Sub-systems and Modules can be produced by the following the Steps shown below:

(a) make a copy of the table of Low Level Functions from the new Functional Architecture Database that should have been created by Step 8 and the comments in section 5.5;

(b) modify the design of the table to remove all the fields except those for the Function Number, Name, Overview and User Needs list;

(c) modify the design of the table by adding new fields for the Sub-system Number, Sub-system Name, Sub-system Location, Module Number and Module Name.

A similar set of Steps will need to be followed to create the table of the Physical Data Flows and their Constituents. The starting point will be the tables of Functional Data Flows and the Data Flows between Functional Areas.

5.7 Use of Architecture Development Tools

The KAREN Project that produced the European ITS Framework Architecture deliberately decided not to use any specific Architecture development tools. The reasoning behind that decision are discussed in the first part of Chapter 2 in the European ITS Functional Architecture Main Document – see reference (a) in Chapter 9.

The KAREN Project team recognises that some Architecture developments may wish to change this policy and use a specific Architecture development tool. This may be because those involved in the development are already familiar with a particular tool because it is (or has been used) for other similar developments. Such reasoning will probably apply to Architectures being developed at National, State, Province, District and City levels.

To use most of the Architecture development tools that are currently available, it will be necessary for them to import the data from the European ITS Framework Architecture. It is recommended that this should be achieved using the User Needs spreadsheet and the Functional Architecture Database. Both of these are available from the European ITS Framework Architecture CD-ROM, Microsoft® Excel and Access respectively.

It will also be necessary to include data about the Terminators, and Functional Areas. Both of these will be found in the European ITS Functional Architecture Main Document – see Chapters 4 and 5 respectively.

As part of the data import, or when it is completed, the resulting Architecture that the tool can produce will need to be checked. How this is done will depend on the facilities provided by the tool itself. It is strongly recommended that whatever and however the checks are made, the data in the European ITS Functional Architecture Database is taken as being the “master”. This is particularly important if the checking process involves the use of the Data Flows Diagrams (DFD’s) in the European ITS Functional Architecture.

6. Physical Architecture links to the outside World

6.1 Introduction

This section provides the definition and description of the terminators for the European ITS Framework Architecture. They are illustrated using the Context Diagram that shows the links between the terminators and the Architecture. Finally a table of acronyms is provided that will be used where necessary throughout the rest of this Document and its Annex.

6.2 Definition of Terminator

A terminator is the link between the Framework Architecture and the outside World. It provides a definition of what the Architecture expects the outside World to do, the data it is expected to provide, and the data to be provided to it by the Architecture. A terminator may be a human entity, a system, or a physical entity from which data can be obtained such as the atmosphere, or road surface. Both human entities and systems may be part of Organisations or Public Authorities that contribute in some way to the provision of ITS services. A rigorous definition is provided for each terminator, written in "shall" language. This is to make it easier to verify that the terminators do fulfil the roles that the Architecture expects. The terminators provide links to both the Functional and Physical Architectures so that the same terminator definitions will be found in the Physical Architecture document.

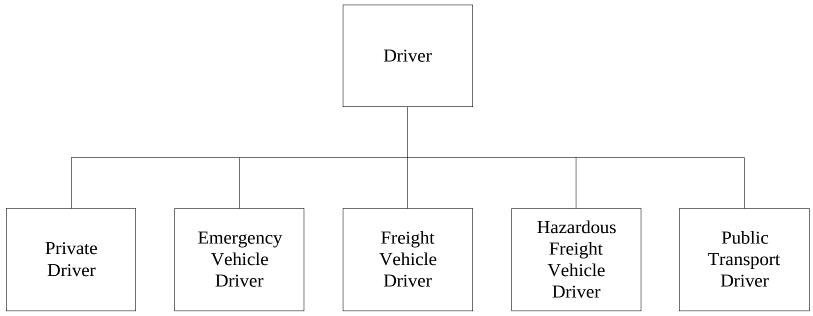
6.3 KAREN Terminator Definitions

There are twenty terminators for the European ITS Framework Architecture. They are defined in the following table which provides their names and definitions. In every entry, the term "the System", with a capital 'S', means the implementation of the KAREN Architecture as a System.

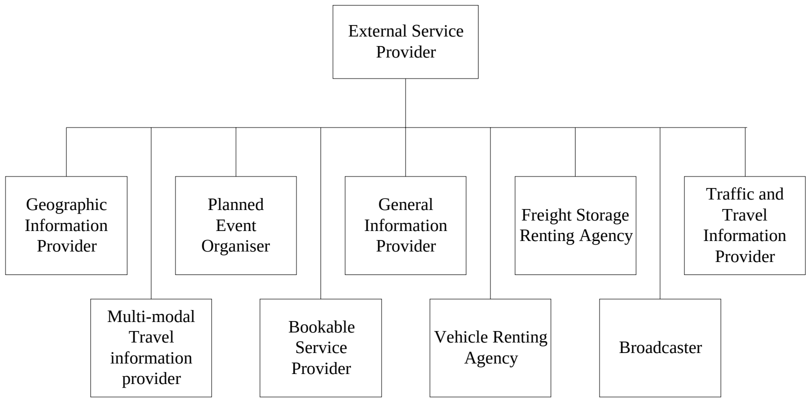
Table 3 European ITS Framework Architecture Terminator Definitions

Terminator Name	Terminator Description
Ambient Environment	This terminator shall represent the operational setting in which road-related ITS services interface and operate. It shall consist of weather effects such as snow, rain, fog, pollution effects such as dust, smoke, and man-made electromagnetic effects. This terminator is a physical entity from which data can be obtained. In this case the data shall be obtained through monitoring by appropriate functionality within the System. The data provided by this functionality shall enable Travellers to be informed about adverse conditions. The monitoring shall also enable Authorities and System Operators to choose relevant management strategies to minimise any adverse effects on the use of the road network.

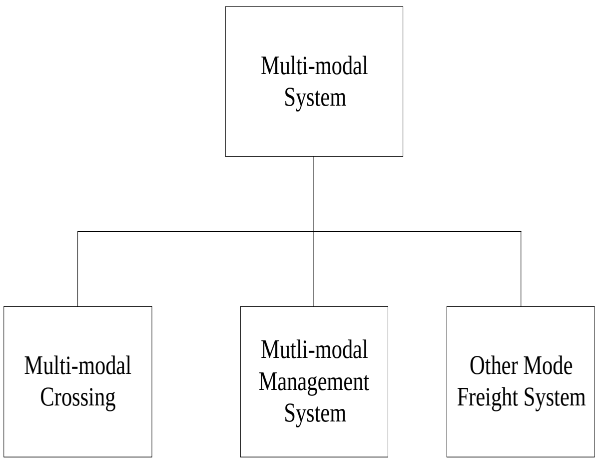
Terminator Name	Terminator Description
Bridge/Tunnel Infrastructure	This terminator shall represent the physical conditions of bridges and tunnels that are part of the road network. It shall represent either the conditions themselves or systems that can detect these conditions. In the first case an analogue input shall be provided from the terminator, and in the second case the input shall contain data. In either case, the conditions that are detected or about which data is provided, shall comprise such things as the status of the bridge or tunnel infrastructure, atmospheric pollution levels on the bridge or in the tunnel, fire (tunnels only) and weather conditions (bridges only). When these conditions are detected by analogue inputs, they must be monitored by appropriate functionality within the System to detect adverse conditions that could affect travel conditions. If data inputs are provided then they shall be capable of being interpreted by the System to determine if and when any adverse conditions prevail..
Consignor/Consignee	This terminator shall represent human or physical entities that need freight (goods) to be transported from one place to place. When the transport is being arranged, the freight (goods) may be referred to as a "consignment". The terminator shall consist of the following two actors. • Freight Shippers - the Sender/Recipient of goods and the owner of details regarding the goods. It shall interface with the System so that good may be prepared and accepted for transport; • Principal - an individual or more often an organisation that is the originator of a freight request. The actor may, after a period of negotiation, establish a contract for a freight service with a freight haulage company. After successful delivery of the Consignment the actor pays the company. It shall be possible for either or both of these actors to arrange for goods to be transported as a consignment, using facilities provided by the System.
Driver	This terminator shall represent the human entity that controls a vehicle. Such a human entity shall be called a Driver. The Driver shall be able to operate any type of power driven vehicle that is licensed to operate on the road network. The terminator shall originate Driver requests to, and receive driver information from the System. Such requests and information shall be that which is common to all Drivers, plus special requests and information for Drivers of particular types of vehicle.

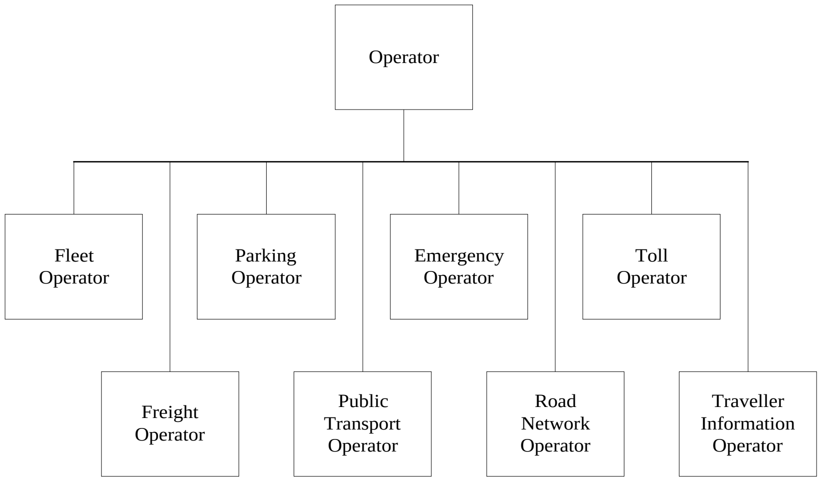
Terminator Name	Terminator Description
Driver (continued)	The actors generalised in the Driver terminator shall comprise those for vehicles such as Private Cars, Freight, Public Transport and Emergency Services. They are shown in the diagram below. 
Emergency Systems	This terminator shall represent systems that are designed for and used by Emergency Services as part of their operations. In this context the term “Emergency Services” shall include organisations that are responsible for services such as fire, police, ambulance and vehicle recovery. The Emergency Systems shall be able to co-ordinate the activities of individual Services. They shall dispatch and control the activities of the vehicles and personnel belonging to a particular Service when they attend incidents. The Emergency Systems shall be given information by the System about emergencies that its functionality has detected. In return the Emergency Systems shall provide reports on progress in dealing with the emergency to enable traffic and travel management strategies to be updated. The Systems shall also provide details of emergency situations affecting road transportation that are reported directly to them, such as through a Mayday call from a vehicle.
External Service Provider	This terminator shall include two types of actors who interface with the System. They shall be responsible for providing information about services that are outside the System and for the output of information provided by the System. The first type of actor in this terminator shall comprise the providers of information used by the System. It shall include information provided as a result of requests from the System. The individual actors included in this type and what they provide shall comprise the following: • Bookable Service Provider - a provider of information about such services that can be booked. These shall comprise but not be limited to things such as accommodation, plus leisure and sports activities, events, etc.

Terminator Name	Terminator Description
External Service Provider (continued)	• Freight Storage Renting Agency - an organisation from which is shall be possible to rent a freight storage area for individual cargo units. Rental shall be performed by a freight operator during the process of synchronising multi-mode transport or of providing the transport service to the consignor in case there is no storage area available at the destination. • General Information Provider - a provider of information about such services as garages, shops, banks, post offices, places of interest, tourist sites, town and city plans, etc.; • Geographic Information Provider - a provider of digitised map data that shall be for use in vehicles and where ever information or data output is to be shown against the background of a map; • Multi-modal Travel Information Provider - a provider of travel information for non-road transport modes (rail, waterborne and air), including details of multi-modal exchange facilities; • Planned Event Organiser - an organiser of external events that may have an impact on the travel conditions on the road network, such as football matches, parades, etc.; • Vehicle Renting Agency - an organisation from which it shall be possible to hire a vehicle for part of a trip. The definition of a vehicle shall comprise but not be limited to a car, coach (for parties), bicycle, taxi, aeroplane, train, or boat. The second type of actor shall comprise the providers of traffic and travel information based on data that originates from within the System. The following two actors fit into this category. • Broadcaster - a provider of traffic and travel information to travellers. It shall be possible for the broadcast mechanism to be through “live” radio (interrupting other programmes) or through other means, such as the Internet and wireless technologies. The information shall be freely available either as a public service or through sponsorship. • Traffic and Travel Information Provider - a provider of a subscription service through which travellers can obtain traffic and travel information. The style, presentation, availability and content of the information shall be the responsibility of each of the above two actors. In some cases the information may be based on data obtained from other sources.

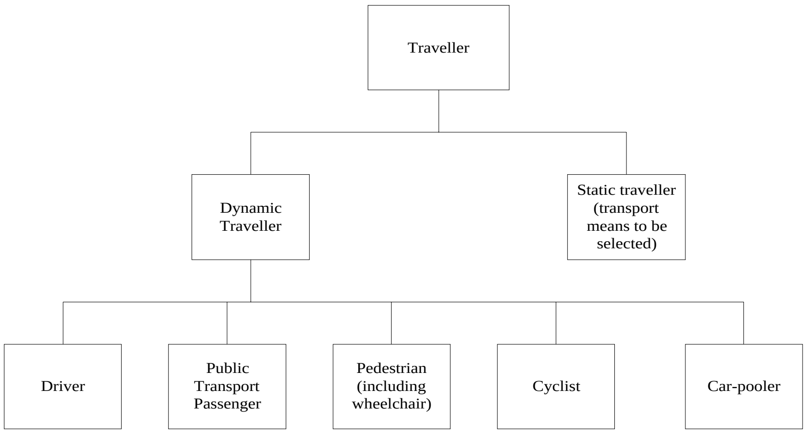
Terminator Name	Terminator Description
External Service Provider (continued)	The last two actors shall each use their own infrastructure(s) to provide the information to Travellers. Neither actor shall be able to input data directly to the System beyond requests for traffic and travel data that they will use as the basis of the information that they output. All the different actors of the two types that are included in the External Service Provider terminator and defined on the previous page are shown in the diagram below. 
Financial Clearinghouse	This terminator shall represent the organisation(s) that process all electronic fund transfer requests originated from the System. These organisations shall enable the transfer of funds from the user of the System (i.e. a Traveller) to the provider of a service. Typically such transfers shall occur when a Traveller pays for services, such as tolls, or parking. These transfers shall also occur if at the same time, the Traveller pays for other services, such as accommodation.
Freight Equipment	This terminator shall represent a freight transport unit that is unable to move by itself. It shall be possible for the unit to be used to carry different cargoes on successive trips, or for it to be moved with no cargo, i.e. the unit shall be reusable. In the context of the European ITS Framework Architecture it shall be possible for the unit to be a trailer, semi-trailer, swap body, container, or pallet. The unit shall therefore need the services of a road tractor unit to be moved through the road network, or to be placed on a road vehicle. Its physical design shall be such that it can also be moved to and from other modes of transport, e.g. heavy rail, water, air. It shall be possible to fit devices to the unit that enable it to be tracked, its condition to be monitored and for its location to be determined. Each of these functions shall be possible directly, or by using communications with the road tractor unit or unit from another mode of transport.

Terminator Name	Terminator Description
Law Enforcement Agency	This terminator shall represent an Authority taking the necessary measures or actions to achieve compliance with laws, rules and regulations for the management of road traffic. If any violations of laws are detected by the System, the terminator shall provide sufficient data for the Authority to identify and initiate prosecution of the offenders. This data that is provided by the System shall as a minimum comprise such things as, visual image, vehicle identity, location, time, date, nature of violation.
Location Data Source	This terminator shall represent external entities that provide position information to the System. Typically this information shall be provided continuously without requests being made by any of the receiving Functions within the System. The information shall be used by these Functions in the determination of the position of vehicles and travellers within the road network managed by the System. Positional information is useful when for example, the location of a Public Transport vehicle is needed to determine the need for and timing of priority at junctions controlled by the System.
Maintenance Organisation	This terminator shall represent a human entities or Systems that are part of organisations able to carry out work to build and/or maintain a road network, and/or can carry out maintenance on equipment that is part of the System. It shall be possible for the terminator to exchange data with the System in two ways. Firstly by providing information to the System about the time, place and duration of planned maintenance work. Secondly by receiving requests from the System for maintenance work to be performed. Maintenance activities shall include any repairs required to all types of System equipment including that at the roadside, or Public Transport Vehicles and equipment. It shall also be possible for data about the status and completion of maintenance activities to be sent by the terminator to the System
Multi-Modal System	This terminator shall represent any systems that manage the transportation of Travellers and Freight by modes that are other than those that use the road network. These modes shall comprise but not be limited to things such as heavy rail, water (canals, and other in-land waterways), ferry operations, and air services. All of them shall either have a direct interface with the road network (e.g. a crossing), or shall be available for use as part of a trip that can be planned and implemented by a Traveller, or be used for the transport of freight.

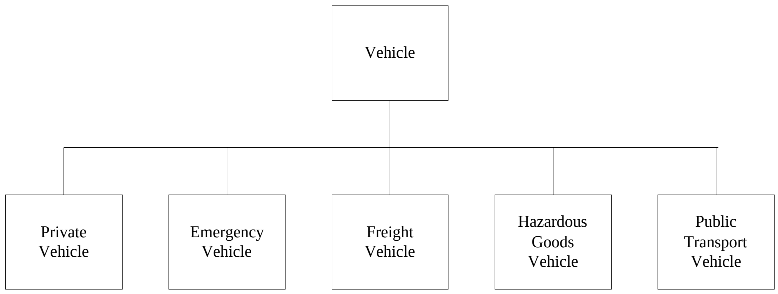
Terminator Name	Terminator Description
Multi-Modal System (continued)	This terminator has been subdivided into the following actors. • Mutli-Modal Crossing - shall provide the input from a non-road based transportation system that has a physical interference with a road-crossing. This input shall enable the System to generate traffic control strategies that can grant temporary priority to the non-road traffic. Examples of actors that are included in this terminator are heavy rail systems, river bridges, etc. The System shall be able to send data requesting that the physical interfaces remain open to road traffic to enable the passage of emergency vehicles, or vehicles with hazardous goods. This may cause interruption to the other mode, e.g. a train may have to stop and wait for a road crossing to open. • Multi-modal Management System - shall provide the link to other non-road information or control systems that may need to exchange information with the System. Access to these systems shall be used to enable trip planning, or to exchange information about incidents that have occurred. It shall be possible for these incidents may be in the network controlled by either the System or the Related Multi-modal System. • Other Mode Freight System - shall provide the link to systems that are responsible for the conveyance of freight using modes of transport other than road, e.g. water, air, and rail. These systems shall exchange data with the System to enable the synchronisation between the use of the different modes in order to maximise the efficiency of freight transport, e.g. to reduce the waiting time at modal interchanges. The actors included in this terminator and described above are illustrated in the diagram shown below. 

Terminator Name	Terminator Description
Operator (continued)	This terminator shall comprise a diverse set of human entities who can perform privileged interactions with the System, thereby contributing to the way in which it operates. It shall be possible for this contribution to include the planning, monitoring, controlling and the evaluation of the System operation. The scope of the human entities (actors) included in this terminator is shown in the diagram below. It shall be possible for some or all of them to be combined into a lesser number of entities for particular System implementations. Thus for example, it shall be possible for the Freight Operator and the Fleet Operator to be the same human entity if required by a particular implementation. In the diagram below showing the various actors, the Road Network Operator actor includes both the Traffic Operator who is the user of the System and the System Operator who is its guardian. The Traffic Operator will use the System to manage traffic, whilst the System Operator will control the way in which the System manages traffic and the static data that it uses. 
Related Road System	This terminator shall represent a link to other instances of Systems that have been produced using the European ITS Framework Architecture. Typically these Systems shall be located in TCC's or TIC's that either serve other geographic areas, or are part of other organisations serving the same geographic area. This terminator shall enable traffic and travel information as well as traffic flows and control strategies to be exchanged with these other Systems. It shall be possible for data to and from these other Systems to be requested by either System, or for it to be exchanged between the Systems at regular intervals.

Terminator Name	Terminator Description
Road Pavement	This terminator shall represent road-surfacing material whose status is monitored by the System. The data that is collected shall enable the System to decide what maintenance operations are necessary to ensure that the surface causes no hazard to vehicles, pedestrians, as well as those in wheelchairs, or have imperfect sight. The different status of the road surfacing that can be measured shall include but not be limited to conditions such as ice, flood water, landslides, etc.
Traffic	This terminator shall represent the movement of vehicles along a route. Traffic shall depict the vehicle population from which traffic surveillance information is collected and upon which traffic management measures are applied. It shall be possible for the movement of vehicles shown by this terminator to be manifest to the System in a number of forms. These shall include but not be limited to such things as video, laser or infra-red images, magnetic signature or any other way in which the presence of a vehicle can be determined.
Transport Planner	This terminator shall represent the human entities and/or systems that are responsible for planning changes to the structure of the road transportation network managed by the System. It shall be possible for them to use information gathered by the System and to provide input and guidance to enable the System to produce strategies that can be implemented to optimise transport network use. This optimisation may be required for incident management, or to influence the demands for particular modes of road transport so that particular transport policies may be implemented.
Traveller	This terminator shall represent as an actor any human entity who uses (or is about to use) a transportation service provided by the System. Such a human entity shall be called a Traveller. It shall be possible for the service provided by the System to be available to a Traveller across all road related modes of travel. These shall comprise, Public Transport, private car, cycling and walking. Travellers shall also be able to use other modes of transport through the interfaces provided by the System to other terminators. The Traveller shall be able to interact with the System to get travel information and to plan a trip. When doing this, the Traveller is assumed to be static and is thus the Static Traveller actor. If the Traveller embarks on a trip he/she becomes a Dynamic Traveller. Such a Traveller may be a Driver, a Public Transport passenger, a Pedestrian, a Cyclist, or a Car Pooler. It is possible for a Dynamic Traveller to switch between any of these actors depending on the mode(s) used between the origin and destination of the trip.

<table>
<tr><th>Terminator Name</th><th>Terminator Description</th></tr>
<tr><td>Traveller
(continued)</td><td>The actors that are included in the Traveller Terminator are shown in the diagram below.

Note: the Driver actor is further decomposed by Vehicle type as a separate Terminator – see previous definition.</td></tr>
<tr><td>Vehicle</td><td>This terminator shall represent the vehicle in terms of parameters connected with its operation and devices or systems that can be used to change its operation through the action of the System. The parameters shall be monitored by sensors within the System and shall comprise but not be limited to the physical movement, orientation, location and condition of the vehicle, plus the condition of the driver. For Public Transport vehicles the number of its passengers and their requests for fare payment and other services shall be included, whilst the condition of the cargo shall be included for Freight Vehicles.

The in-vehicle devices and systems represented by this terminator shall either be those that are standard fittings or be those needed to enable specific other services to be provided. Examples of such devices shall comprise but not be limited to, engine management, automatic braking systems, power assisted steering, cruise control, climate control, entertainment, air bag deployment, central locking and vehicle transmission operation.

The individual actors that have been included in this terminator are illustrated in the diagram shown below.
</td></tr>
</table>

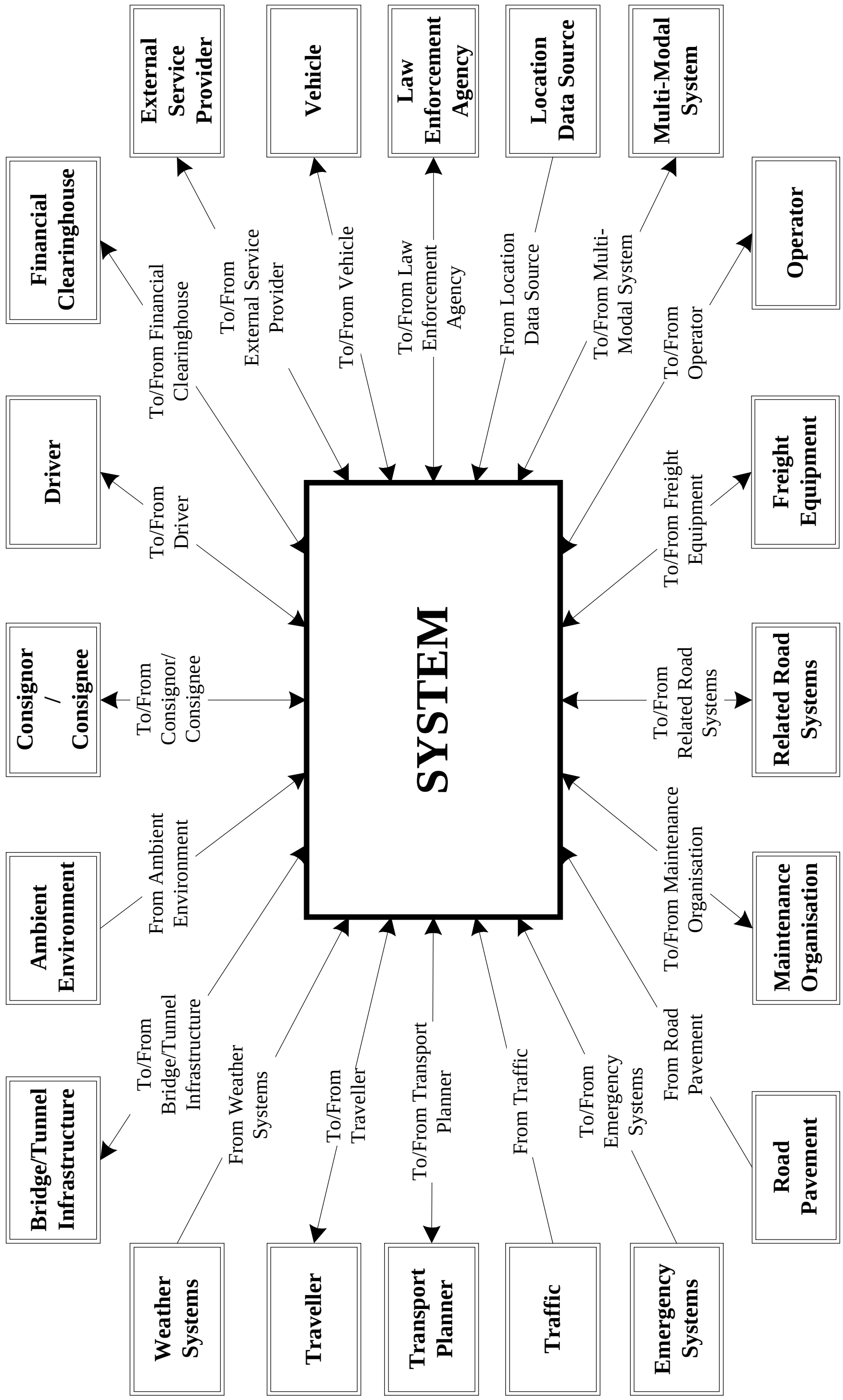
Terminator Name	Terminator Description
Weather Systems	This terminator shall represent the system or human entity that provides general area weather information and weather forecasts to the System. The information that is provided shall comprise things such as temperature, fog, rain and wind (direction and strength), whilst the forecasts shall predict changes in these conditions. It shall be possible for both types of information to be provided on a regular basis or on request from the System. The System shall also be able to send information back to the terminator. This information shall cover weather conditions that have been determined by functionality in the System.

The above table is important because it defines the functionality in each of the Terminators and also sets the System boundary. This boundary defines what is “inside” and “outside” the System. It has been set so that all the things that the System needs to provide the ITS services required by the User Needs are within the System. Thus for example, Vehicle detectors and other roadside devices are “inside” the System. However so far as the Vehicle is concerned, only those devices needed to implement ITS services are included “inside” the System. This means that the systems for such things as automatic cruise control, engine management, traction control, brake control, air conditioning, and air-bag deployment that are currently fitted as standard to most Vehicles are “outside” the System.

6.4 Context Diagram

The connections between the twenty terminators defined in the previous section and the System are shown by the Context Diagram in Figure 1 on the next page. Each terminator is linked to the System by a terminator Data Flow. Those shown in the Context Diagram are the Top Level, or Context Terminator Data Flows. The definitions for each of the Data Flows on this diagram (including a list of their constituent Data Flows) will be found in part of Annex 2 to this Document.

Figure 1 European ITS Framework Architecture Context Diagram



6.5 Table of Acronyms

A set of acronyms has been created for each of the terminators and is shown by Table 2 below. They utilise the initial letter or a set of letters that represents the terminator name. The set includes acronyms for the actors identified as being part of terminators. These acronyms are in two parts, the first being the acronym for the terminator to which the actor belongs. The objective is to provide consistency in the use of names across all the Data

Flows Diagrams (DFD's) and within the databases for Function, Data Flow and Data Store definitions.

Table 4 Terminator Acronyms

Terminator Name	Actor (sub-terminator) Name	Acronyms
Ambient Environment		ae
Bridge/Tunnel Infrastructure		bti
Consignor/Consignee		cc
	Freight Shipper	cc.fs
	Principal	cc.p
Driver		d
	Emergency Vehicle Driver	d.e
	Freight Vehicle Driver	d.fvd
	Hazardous Freight Vehicle Driver	d.hfvd
	Private Driver	d.pr
	Public Transport Driver	d.ptd
Emergency Systems		es
External Service Provider		esp
	Bookable Service Provider	esp.bsp
	Geographic Information Provider	esp.g
	General Information Provider	esp.gip
	Multi-Modal Travel Information Provider	esp.mmtip
	Planned Event Organiser	esp.peo
	Vehicle Renting Agency	esp.vra
	Freight Storage Renting Agency	esp.fsra
	Broadcaster	esp.b
	Traffic and Travel Information Provider	esp.ttip
Financial Clearinghouse		fc
Freight Equipment		fe
Law Enforcement Agency		lea
Location Data Source		lds
Maintenance Organisation		mo
Multi-Modal System		mms
	Multi-Modal Crossing	mms.mmc
	Multi-Modal Management System	mms.mmms
	Other Mode Freight System	mms.omfs
Operator		o

Terminator Name	Actor (sub-terminator) Name	Acronyms
	Emergency Operator	o.eo
	Fleet Operator	o.flo
	Freight Operator	o.fro
	Parking Operator	o.po
	Public Transport Operator	o.pto
	Road Network Operator	o.rno
	Toll Operator	o.to
	Traveller Information Operator	o.tio
Related Road System		rrs
Road Pavement		rp
Traffic		trfc
Transport Planner		tp
Traveller		t
	Car-Pooler	t.cp
	Cyclist	t.c
	Dynamic Traveller	dt
	Driver	*(see above)*
	Public Transport Passenger	t.ptp
	Pedestrian	t.p
	Static Traveller	st
Vehicle		v
	Emergency Vehicle	v.ev
	Freight Vehicle	v.fv
	Hazardous Freight Vehicle	v.hfv
	Private Vehicle	v.p
	Public Transport Vehicle	v.ptv
Weather Systems		ws

These acronyms are used in the three Annexes to this Main Document. They are also used in the European ITS Functional Architecture – see reference (a) in Chapter 9.

6.6 Use of Terminators by Functional Area

Not all of the Terminators will be used by every Functional Area. This is because some Terminators are specific to the functionality in each of the Areas. The distribution of the use of the Terminators between the Areas is shown by the table below an on the following pages.

Table 5 Use of Terminators by Functional Areas

Terminator Name	Functional Area		Flow Direction
	Name	No.	
Ambient Environment	Manage Traffic	3	I
Bridge/Tunnel Infrastructure	Manage Traffic	3	I/O
Consignor/Consignee	Manage Freight and Fleet Operations	8	I/O
Driver	Manage Freight and Fleet Operations	8	I/O
	Manage Public Transport Operations	4	I/O
	Manage Traffic	3	I/O
	Provide Advanced Driver Assistance Systems	5	I/O
	Provide Electronic Payment Facilities	1	I/O
	Provide Security and Emergency Facilities	2	I/O
	Provide Support for Law Enforcement	7	I/O
	Provide Traveller Journey Assistance	6	O
Emergency Systems	Provide Security and Emergency Facilities	2	I/O
External Service Provider	Manage Freight and Fleet Operations	8	I/O
	Manage Public Transport Operations	4	I
	Manage Traffic	3	I/O
	Provide Advanced Driver Assistance Systems	5	I/O
	Provide Electronic Payment Facilities	1	I/O
	Provide Security and Emergency Facilities	2	I/O
Financial Clearinghouse	Provide Electronic Payment Facilities	1	I/O
Freight Equipment	Manage Freight and Fleet Operations	8	I/O
Law Enforcement Agency	Manage Freight and Fleet Operations	8	I/O
	Provide Support for Law Enforcement	7	I/O
Location Data Source	Manage Freight and Fleet Operations	8	I
	Provide Advanced Driver Assistance Systems	5	I
	Provide Security and Emergency Facilities	2	I
	Provide Traveller Journey Assistance	6	I
Maintenance Organisation	Manage Public Transport Operations	4	I/O
	Manage Traffic	3	I/O
	Provide Advanced Driver Assistance Systems	5	I
Multi-Modal System	Manage Freight and Fleet Operations	8	I/O
	Manage Public Transport Operations	4	I/O
	Manage Traffic	3	I/O

Terminator Name	Functional Area		Flow Direction
	Name	No.	
Operator	Manage Freight and Fleet Operations	8	I/O
	Manage Public Transport Operations	4	I/O
	Manage Traffic	3	I/O
	Provide Electronic Payment Facilities	1	I/O
	Provide Security and Emergency Facilities	2	I/O
	Provide Traveller Journey Assistance	6	I/O
Related Road System	Manage Public Transport Operations	4	I/O
	Manage Traffic	3	I/O
	Provide Advanced Driver Assistance Systems	5	I
Road Pavement	Manage Public Transport Operations	4	I
	Manage Traffic	3	I
	Provide Advanced Driver Assistance Systems	5	I/O
Traffic	Manage Traffic	3	I
	Provide Advanced Driver Assistance Systems	5	I
Transport Planner	Manage Traffic	3	I/O
Traveller	Manage Public Transport Operations	4	I/O
	Manage Traffic	3	I/O
	Provide Electronic Payment Facilities	1	I/O
	Provide Security and Emergency Facilities	2	I/O
	Provide Traveller Journey Assistance	6	I/O
Vehicle	Manage Freight and Fleet Operations	8	I/O
	Manage Public Transport Operations	4	I/O
	Manage Traffic	3	I
	Provide Advanced Driver Assistance Systems	5	I/O
	Provide Electronic Payment Facilities	1	I
Weather Systems	Manage Traffic	3	I
	Provide Advanced Driver Assistance Systems	5	O

The symbol “I” means that data is sent from the Terminator to the Functional Area, whilst the “O” symbol means that the data is sent from the Functional Area to the Terminator. The “I/O” symbol means that data is sent in both directions.

Descriptions (including lists of constituent Data Flows) of the main Data Flows that link the Terminators to the System and their descriptions will be found in Annex 2 of the European ITS Functional Architecture Main Document – see reference (a) in Chapter 9.

7. Summary of Key Issues from all Systems

7.1 Introduction

The following is a list of the Key Issues highlighted for each of the "example Systems" included in the Physical Architecture. They are included here to provide a single source of reference as they may of relevance to the European ITS Deployment Study - see D 3.5, and to other KAREN Project Deliverables.

7.2 P1 - Integrated Urban Traffic and Public Transport Management System

1. Choice of a suitable location referencing to be used in the urban domain. Currently there are not standards available for referring data as detailed as those available for the urban area (e.g. travel time, traffic flow and density for each link, vehicle turning percentages at intersections, traffic light timings, etc.). The only location referencing currently used and recommended, for example in the DATEX protocol, is the one used for the RDS-TMC. It is unsuitable for urban environment because it describes points in the network only and does not describe links/arcs nor vehicle movements in the network. In addition to these limitations, it implies the creation and maintenance of an external coded database of locations. An emerging standard solution seems to be the ILOC (Intersection LOCation) location referencing which includes the whole description of locations and can be completed with LLOC (Link LOCation) and MLOC (Movement LOCation) for the description of links and movement of the graph. Currently the state of development and verification of this method is limited and refinement and consolidation work is still needed before being fully operational.

2. Protocol to be used for data exchange. The selection of an harmonised communication interface to share and exchange data and services is now naturally oriented to the DATEX solution, although it may not suffice all the requirements generated by urban environment based applications and services (e.g. DATEX does not include yet messages for data such as the turning percentages, travel time and vehicle position for public transport, etc.), alternative suitable communication interfaces should be looked at for specific class of data. Mobile agent technology provides, for example, a promising solution but consolidated standards are not available and specific transport related application layers may be needed.

3. Organisational structure for sharing data and infrastructure. The feasibility of an actual "real life" environment, where data and infrastructures are shared, requires suitable technical solutions but also a solid organisational structure, which is usually more critical to set-up. Separation of responsibility must be clear to involved actors and guaranteed quality and reliability of shared data and services must me given. The model provided by the "DATEX interchange agreement" can be sufficient for the pure data exchange relationship, it should be used as basis for a more comprehensive needs.

7.3 P2 – RDS-TMC Italian System

1. The Italian RDS-TMC service requires that the given traffic information is reliable and provided as quickly as possible. The efficacy and efficiency of this service is obtained through the direct connection to the information centres belonging to institutions/organisations like Road Police, Carabinieri and highway responsible companies. On the one hand, this gives the sureness and legitimacy of the given information. On the other, it reduces the delay between when the event happens and when the information is provided. This second characteristic is improved by the reception of other traffic data on different communication ways (telematics, telephone, telex, fax,…) and proprietary formats.

2. The System is capable of providing its validated information to other External Service Providers who are going to use the reliable traffic information within value added services.

3. Organisational issues are critical because of the number of actors involved, responsibility involved, ownership of data. The criticality is mainly related to the potential impact of information transmitted on the business of single actors. Consequently, on top of technical solutions, interchange agreements have to be established between the actors involved to make the service actually feasible.

4. Data exchange solution must take into account that the service requires two-way exchange of data with correspondent Traffic Information Services operating in neighbouring countries. In particular messages and geographical location references used must be language independent.

5. Specific solutions must be adopted to allow travellers receivers to select among three level of information details: National, Regional and Local. In the Italian implementation the same messages (i.e. including all level of information) are broadcasted over the whole country, selection capability is demanded to the receivers.

6. Specific encryption solutions may be required to have a basic level of service freely available to public and a one or more levels of services reserved to subscribers.

7. Solutions for updating and changing, when needed, the reference map and location codes must be found to make the service actually feasible. The RDS bandwidth cannot allow the transmission of these data, the Italian solution make use of CD-ROM periodically issued under the responsibility of a unique authority.

7.4 P3 – Urban Traffic Control and Public Transport Priority System

1. The System has an open architecture. This gives the opportunity of exchanging data and information with other systems on the base of an agreed format. The openness to potential integration is extremely important for the given example. In fact the described System aims at providing a response to two fundamental requirements of wide-area traffic control. On the one hand it provides significant improvements in private vehicle mobility in all traffic conditions. On the other, it gives priority to

selected public transport vehicles at traffic light controlled intersections. Consequently, the integration for co-operation with an AVM (Automatic Vehicle Monitoring) system is needed or alternatively obtained with special roadside detectors. In this example the AVM provides the selected public transport vehicles forecasted arrivals at the controlled intersections, and these data represent an input for the local control system to give the required priority. In addition to interfacing directly with AVM, it is also important to notice that this System can integrate with related road systems, like a Town Supervisor or other traffic control systems, emergency systems, in order to actuate a urban road network control strategy which considers also other influence factors like maximum priorities, pollution emission levels, city-wide demand management schemes, etc.

2. The System is designed to improve urban level conditions by the application of fully automated control algorithms. Its physical architecture is hierarchical and decentralised, the optimal reference control strategy is determined at the higher level on the basis of area traffic predictions, traffic light control is finally optimised and actuated at the local level amending the reference control strategy according to the actual traffic conditions occurring at the intersections. This type of architecture requires a communication network among intersections and the connection of some nodes (e.g. one third) of this network to the control centre. The control approach is known as “goal co-ordination” and is typical of the hierarchical control systems: the upper level is in charge of determining the criteria to be adopted in the control law, while the lower level is in charge of fitting these criteria in a fast reaction feedback loop on the estimated state of the local system.

3. The communication network at the intersection level has the MFO as nodes. The MFO are the multifunctional outstations with the following peculiar functionality and benefits. First of all they have the capability of “observing” the intersection status and actuate the needed local optimum control actions. At the same time, they enable the exchange of application data and information between the local infrastructures (e.g. VMS panel, bus-stop displays, beacons, car park panels) and the related systems at the central level. As benefit, this shared communication network solution offers considerable savings on operating costs if compared to an alternative solution connecting directly each single MFO to the operations centres.

4. The open architecture is assured at the Central Traffic Control level. In fact, here the Data Exchange Interface module is responsible for the properly translation of input information into the internal proprietary formats and location referencing to be elaborated and/or routed to the local MFO’s. The data are received and distributed over the communication protocol TCP/IP.

7.5 P4 – Telematics Technologies for Transport and Traffic in Torino System

1. The System is integrated. In fact, it represents a co-operative environment where operators and systems can all be part of a global ITS (Intelligent Transport System) and benefit from tangible added value, if compared with non-integrated solutions. On the other hand integration must not generate additional problems during the system life-time. These targets respond to the following requirements:

- Build on existing systems
- Allow data sharing by adding network capabilities
- Favour the implementation of new high-level function for overall co-ordination

2. The System can host additional Sub-systems adopting the same rules without requiring any modification to the others. In fact, the integrated Sub-systems keep their own choices for data modelling, but agree on a common "language" for exchanging data with other systems. Consequently a minimum set of standards are defined (i.e. data dictionary, location reference and message format) and the impact is minimised on Sub-systems by providing proper interfaces to exchange data with the others.
3. The System is characterised by shared communication networks at all levels (between Sub-systems, from Sub-systems to roadside MFO, between MFO). At the road level, sensors, beacons and information displays share the same communication network minimising the duplication of hard-wiring. This solution allows considerable savings in both installation and operation costs, by installing multifunctional outstations on strategic locations on the transport network and connecting as many devices as possible to each of these outstations. Each outstation then provides the management of a shared communication link to the control centre or to other outstations. Thus outstations act as nodes of a shared communication network where messages/data are transmitted between devices and control centres regardless the application which they belong to.

7.6 P10 - Electronic Cash Transaction System

1. The Electronic Cash Transaction System (P10) is based on the Conceptual Model of DD ENV ISO 14904 "Road Transport and Traffic Telematics - Automatic Fee Collection - Interface Specification for Clearing between Operators". This, or a similar model, will have to be agreed throughout a given geographical area in order to satisfy User Need 4.1.2.2.

2. The User's Electronic Travel Pass (P10.2), together with its interface, is fundamental to the correct operation of this System:
   - It must be capable of mass production;
   - It must be robust;
   - The method, and rate, of data transfer will dictate how it may be used, e.g. in circuit or contact-less;

3. For seamless travel throughout the EU:
   - The User's Electronic Travel Pass (P10.2) will have to function in the same way at all locations;
   - There will have to be very many, easily accessible, Collection Agent Sub-systems (10.1);
   - The fv-detect data flow must be understood by all Charge Customer Modules (P10.3.1);

- It will probably be necessary to have many co-operating Clearing Sub-systems (10.4);
- The Traveller and Driver interface with the Charge Customer Modules (P10.3.1) must be understood by all travellers and drivers (see also User Need 6.1.2.8)

7.7 P11 - Automatic Road Tolling System

1. For seamless travel throughout the EU:
   - The On Board Unit Sub-System (P11.2) will have to function in the same way at all locations;
   - There may be many Contract Management Sub-systems (P11.1);
   - There may be many Toll Management Sub-systems (P11.5);
   - It will probably be necessary to have many co-operating Toll Clearing Sub-systems (P11.5);
   - The efc_from_OBU and efc_to_OBU data flows must be understood by all On Board Unit Sub-Systems (P11.2);
   - The handling of vehicles whose On Board Unit Sub-System is defective, or not present, has not been developed in this System. A consistent approach will need to be taken that is likely to have an affect on the fv_vehicle_ID data flow.

7.8 P 22 – Hazardous Goods Management System

1. The definition of a consistent classification of hazardous goods is required to guarantee that the correct information is available in case of an emergency.

2. The response time of the System must be nearly real-time to reduce the interference time of Emergency Services to a minimum.

7.9 P 30 - Urban Traffic Management System

1. The successful operation of the System depends at least in part on its ability to accurately detect the traffic conditions in the road network that it is serving, particularly in those parts that form “bottle-necks” or are crucial to the rest of the network. Deployment of this detection can be costly, both in capital for initial installation, and in revenue for on-going maintenance and operation.

2. Most Systems are judged to be operating successfully if they are seen to manage the traffic using the road network in the most efficient way. What is judged as being “efficient” will depend on the traffic policy goals of the Government Organisation(s) that regulates the use of the road network. However it will also depend on the techniques that are available to achieve efficient traffic management. There are several of these techniques all have varying degrees of success in different road network configurations. It is therefore important that any System is capable of implementing the most appropriate technique in each part of the road network.

3. Most traffic management techniques rely on the detection of traffic conditions to trigger or modify their implementation. However in many cases, particularly when

congestion is rising, the detection is often too late for the implementation of any traffic management technique to achieve any worth-while benefits. It is therefore necessary for cheap and reliable congestion prediction techniques to be developed that can lead to the advanced implementation, thereby achieving better results in terms of more efficient operation of the road network.

4. The total cost of operation of the System will depend in part on the cost of the infrastructure used to communicate data between the central traffic management and roadside Sub-systems. The type of infrastructure used, and the ability for it to be shared with other users will be crucial to the size of these costs.

7.10 P31 - Inter-urban Traffic Management System

1. The successful operation of the System depends at least in part on its ability to accurately detect the traffic conditions in the inter-road network that it is serving. Given the large geographic area sometimes served by inter-urban road networks, deployment of this detection can be costly, both in capital for initial installation, and in revenue for on-going maintenance and operation.

2. Most Systems are judged to be operating successfully if they are seen to manage the traffic using the road network in the most efficient way. For inter-urban networks, this generally means keeping the traffic flowing in such a way that the maximum flow rate is obtained with minimum disruption. One of the major problems appears to be the prediction of flow breakdown (congestion) due to both traffic volumes and incidents. Work is needed to refine and enhance these techniques to give better advanced warning of congestion without increasing the concentration of detection. There is also a need to look more closely at ways of determining diversionary routes with enough warning to enable drivers to take action.

3. The total cost of operation of the System will depend in part on the cost of the infrastructure used to communicate data between the central traffic management and roadside Sub-systems. The type of infrastructure used, and the ability for it to be shared with other users will be crucial to the size of these costs.

7.11 P50 - Advanced Driving Assistance System

1. The Area of Vehicle Systems has the particularity of having, by definition, all the sub-systems and modules located inside the vehicle. This is typical of this area only as, in general, systems for Intelligent Transport have a more distributed architecture and consequently a wider allocation of the sub-system, functions and modules in different places.

7.12 P60 - Traveller Assistance and Route Guidance System

1. Message formats: The communication with external sources of information will primarily be done via internet. This however necessitates message formats to be defined for questions and answers. A general format is required, for each type of internet site (e.g. hotels, event information) a more specific message will be needed

and maybe each individual site will foster its own dedicated information set. This in total requires reference models and a reference architecture, to accommodate future extensions.

2. TA functionality: What is described above is the most advanced version of the TA, which also includes information from previous trips. Less advanced solutions are possible, the lowest one being to work without a TA alone. However, if trip planning is to succeed in the proposed manner, repeated and cumbersome re-entry of the same data has to be avoided, especially to prevent errors to be made by the traveller in a boring machine communication process.

3. User orientation: What is offered is a technical solution, which needs to take the human, the traveller, at the centre. What the traveller wants is pivotal, the technical solution is secondary. Unfortunately, the user needs are not very specific in this respect and cannot be used as guidance; a seamless user involvement is mandatory to make it really useable and useful.

7.13 P 70 - Law Violation and Vehicle Detection System

1. The system has to follow the legal restrictions covering laws on detection, image capturing of the driver and other relevant issues. The co-operation with all kinds of executive power (Law Enforcement Agency, Police, ..) is absolutely necessary while planing and implementing the system.

2. Due to the fact that this systems is containing high-sensitive data it is necessary to ensure high security standards and very low fault tolerance.

7.14 P81 - Distributed Freight Management System

1. Freight Service Provider: It is easy to imagine that with the widespread acceptance of the Internet and the increasing ease of data communications will become more widespread. For example, a potential customer for freight services (“Principal” actor within the Consignor/Consignee terminator in the “example” System) having no experience with contacting Haulage Companies, would prefer to deal with a broker that offers freight services. Such a broker could offer different freight services via a Web Site, the site, for example, having the following characteristics:

   - a tariff table
   - a Secured Sockets Layer for secured payment of freight services
   - traffic information for the principal routes in Europe
   - a simple route planner

2. New communication protocols: The mobile phone Operators Harmonisation Group has agreed to apply third generation (3G) wireless communication systems to allow EDI, video transmission, data exchange via Universal Mobile Telephone Services. This would bring cost benefits by allowing competing products to provide communications between Fleet Management Sub-system and the Driver/Vehicle (in

the Vehicle to Terminal interface and the Terminal to Central Fleet Manager interface).

3. Hazardous Goods, or more generally "Non-standard Goods": What becomes apparent when designing physical systems is the desire to make a system "realistic", or to make it adaptable to the real world where market forces predominate. This is especially true for the road transport domain where competition is high. For any operational planning system, the important things to identify at the start of process are the constraints imposed by the inherent nature of the system and the constraints imposed by the real world. In the case of this system (though the argument could equally well apply other types of route planning systems, route monitoring systems, emergency-response systems etc.) the physical manifestations of these constraints are:

   - infrastructure dimensions: road widths, bridge heights, tunnel diameters etc.
   - trailer sizes
   - cargo dimensions (though Container sizes are standardised)

   Recommendations for standard ways of describing these constraints should be developed as soon as possible. This will enable improvements to be made in the *interoperability* between Systems.

4. Container tagging: Consider a park of freight Containers; studies have shown that a company can expect a turnover of between 15 and 20% of its containers through loss or theft. A cheap electronic tag could be attached to a container to enable low-efficiency tracking of containers (e.g. with one-way communication based on a wireless infrastructure. A satellite surveillance would be costly and unreliable due to masking, since the tag must have line-of-sight communications with a satellite to send a message).

8. Preliminary Safety Analysis

8.1 Introduction

The objective of a Preliminary Safety Analysis (PSA), sometimes called a Preliminary Hazard Analysis, is to classify all the safety hazards that can be identified from the current description of the system. A hazard is "a physical situation with a potential for human injury" [IEC 61508], and they can be found by performing a 'What If?' analysis on a model of the system. Each hazard should then be assigned a Controllability Category to indicate the degree of risk associated with it [UTMC22 2000a].

It is therefore necessary to produce a model of the system so that the possible interactions between it and its environment (i.e. everything that might affect, or be affected by, the system) can be readily identified. A model that has been specifically designed to act as a target for the PSA of transport telematic systems is the PASSPORT Diagram [PASSPORT 1995], and this is an example of its application.

Whilst the specification used as the basis for this example PSA is the "example System" P30 Urban Traffic Management System – see Annex 1, it should be remembered that almost all ITS are safety-related to some degree. Therefore a PSA should always be performed, even if it only demonstrates that there are no safety hazards.

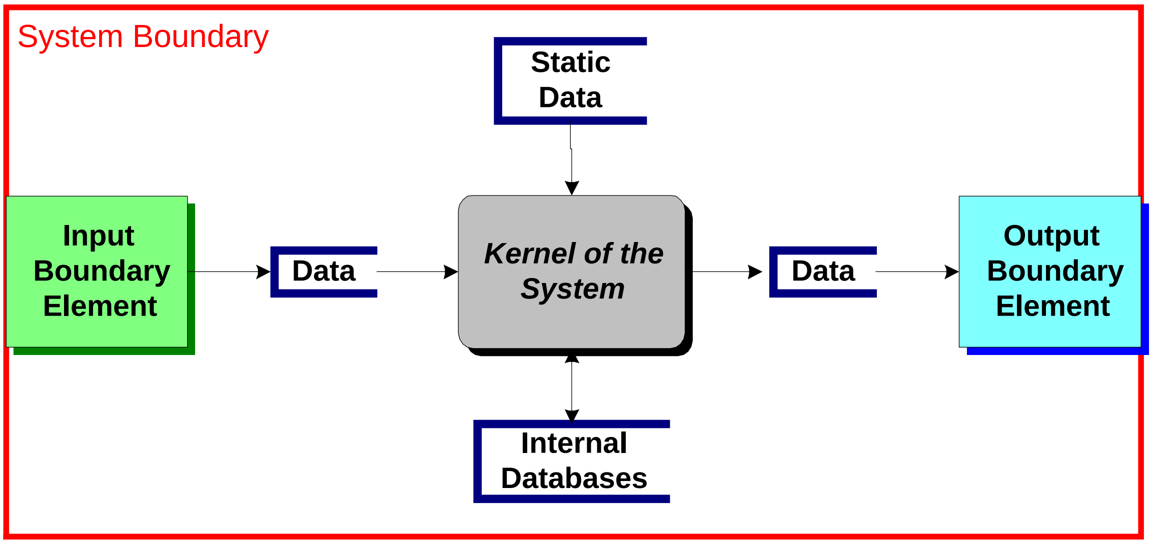
Many of the hazards identified in the PSA of P30 that follows, can only be stated in rather general terms. This is due to the high-level definitions of the various Functions being used by P30. Thus a developer of a real Urban Traffic Management System should repeat this exercise with the actual, and more detailed, definitions of the Functions that will be used. This will enable the hazards to be more clearly defined and, in particular, the degree of risk associated with them.

8.2 Modelling P30 - Urban Traffic Management System

The "example System" P30 has a number of Terminators, i.e. the 'things' that are outside of the System and which provide data for the System and/or receive data from the system. A PASSPORT Diagram does not model the Terminators per se, but the Boundary Elements which are that part of the System. These interact with, or provide interfaces to, each of the Terminators. However, since P30 has a very large number of Boundary Elements and it is not possible to amalgamate them into a small enough number to be able to draw a proper PASSPORT Diagram, an alternative approach is necessary. This was developed initially for the UTMC programme in the UK [UTMC22 2000b]

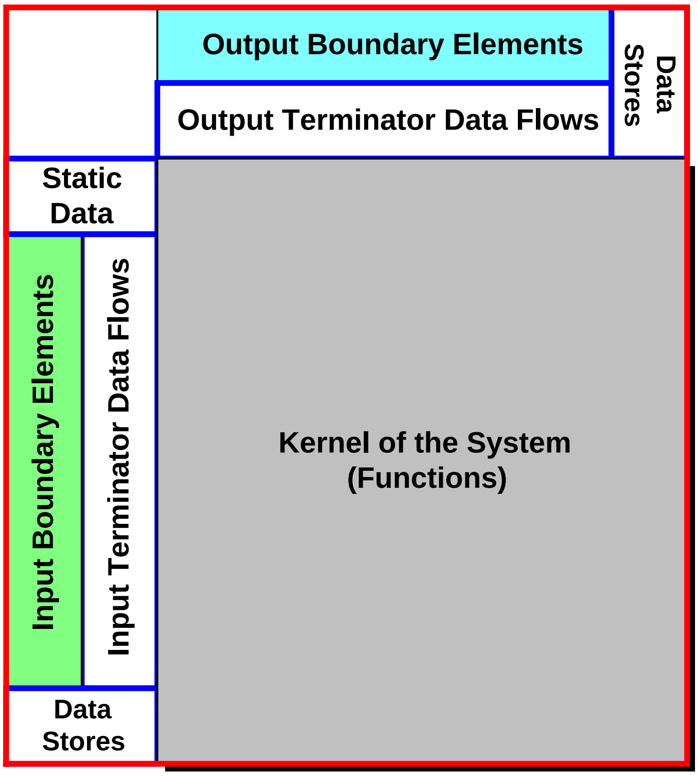
A normal PASSPORT Diagram has the structure shown in the Figure on the next page. The Input Boundary Elements, and the data that they produce, are placed on the left hand side, whilst the Output Boundary Elements, and the data that goes to them, are placed on the right hand side. At the top are any sets of data needed by the System, but that are unlikely to change frequently; and at the bottom are any sets of varying data that are normally stored together. The Kernel of the System contains everything else necessary to perform the function(s) of the System.

Figure 2 Structure of a PASSPORT Diagram



An alternative approach is to present the same information using a spreadsheet whose structure is shown in **Error! Reference source not found.**. In fact more information can be displayed in the spreadsheet than on the normal PASSPORT Diagram, which has to use supporting text, in particular it is possible to include in the Kernel all the functions that use and/or create the data flows.

Figure 3 Structure of a PASSPORT Spreadsheet for a KAREN System



The PASSPORT Spreadsheet for P30 is given in Appendix 1 to the Main Document. It contains all the Functions and Terminator Data Flows of that System. Those Functions with both input and output Terminator Data Flows are located in the corresponding part(s) of the Kernel. Three additional Boundary Elements have been introduced, namely Central, Roadside and Vehicle, to cater for those Functions with only one, or even no Terminator Data Flows. They relate to the Locations specified in P30. There is no specific Static Data for P30.

8.3 Hazard identification

The definition of a hazard that has been used for this PSA is "a physical situation with a potential for human injury" [IEC 61508], and so damage to property, fauna, flora atmosphere, economy, security etc. have not been considered. There are two basic types of hazard, those that are a natural consequence of the way that the System has been specified, and those that may occur as a result of some sort of random or systematic failure during operation.

The Controllability Categories that may be used to classify the degree of seriousness of each hazard are defined in the Table below, and hence the Safety Integrity Level (SIL) of the equipment that will be associated with that hazard (see Section 8.4.2). The proposal for their use is described more fully in [UTMC22 2000 and PASSPORT 1995]. However due to the high-level nature of the Functional Specifications, the Controllability Categories chosen for the hazards that have been identified in the following sections can only be considered to be an approximation. Their value must be confirmed for each specific deployment.

Table 6 Definition of Controllability Categories

Controllability Categories	Definition
Uncontrollable	This relates to failures whose effects are not controllable by the road user(s), and which are most likely to lead to extremely severe outcomes. The outcome cannot be influenced by a human response.
Difficult to Control	This relates to failures whose effects are not normally controllable by the road user(s) but could, under favourable circumstances, be influenced by a mature human response. They are likely to lead to very severe outcomes.
Debilitating	This relates to failures whose effects are usually controllable by a sensible human response and, whilst there is a reduction in the safety margin, can usually be expected to lead to outcomes which are at worst severe.
Distracting	This relates to failures which produce operational limitations, but a normal human response will limit the outcome to no worse than minor.

Controllability Categories	Definition
Nuisance Only	This relates to failures where safety is not normally considered to be affected, and where customer satisfaction is the main consideration.

It should also be noted that not all traffic safety hazards will automatically result in an accident. Most are "conflicts" between one or more vehicles which are resolved by the drivers braking and/or taking avoiding action. It is only when these measures fail that an accident will occur. However, since the difference between "conflicts" and accidents may only be a small fraction of a second, or a few millimetres, they are still undesirable, and measures should be taken to avoid them whenever this is possible.

8.3.1 'What If?' analysis

The only way that a System can produce a hazard is through its outputs. A 'What If?' Analysis is a systematic process in which each output is considered in turn in order to identify possible hazards. It is an informal version of Failure Mode and Effects Analysis (FMEA) in which each output Data Flow is considered in turn, along with the Function that produced it, and consideration is given to its possible failure modes.

With some systems it is also possible to consider each of the inputs in order to gauge their effect on the outputs.

The table on the following pages shows the results of the "What If ?" Analysis for P30. As its Functions are defined at a fairly high level and are technology independent, they have only been considered implicitly during this Analysis. For each failure mode, the resulting hazard(s) have been identified, and a controllability category assigned. However, due to the high-level nature of the Function descriptions it has only been possible to give a range of values for the controllability category. This will need to be assigned exactly for each specific deployment.

The numbers in the column labelled 'O/P' relate to the Output Boundary Elements listed in Appendix 1 to this Main Document.

Table 7 “What if ?” Analysis Results Table

O/P	‘What If?’ Scenario	Resulting Hazard	Controllability	Comments
1.1	Incorrect, or no, information received by the emergency services driver about the request for priority (F2.1.3).	The driver may make an incorrect decision, and the emergency vehicles may be delayed unnecessarily	Up to Difficult to Control	This depends on the actual functionality implemented.
1.2	The offsets of a set of traffic signals between a series of junctions is not suitable for the road network, either because they were set up wrongly or, say, because of a fault in the communications (F3.1.1.5.5).	There may be an increase in collisions and/or pedestrian accidents downstream.	Up to Debilitating	It is assumed that the traffic signal controller controls the cycle phases and stages of the lights, such that the basic safety at each junction is maintained. The ITS may, however, vary the offsets between junctions, e.g. the SCOOT principle.
1.3	The offsets of a set of traffic signals between a series of junctions is not suitable for the traffic conditions (F3.1.1.5.5).	There may be an increase in collisions and/or pedestrian accidents downstream.	Up to Debilitating	As for 1.2
1.4	The traffic signals may be set in an unusual, inconsistent or complex manner (F3.1.1.5.5).	Non-local drivers (at least) will be confused and may try to cross a junction when it is not safe to do so.	Up to Debilitating	As for 1.2
1.5	Failure to provide a ‘green wave’ to emergency vehicles (F3.1.1.5.5).	Emergency vehicles may be delayed unnecessarily.	Up to Debilitating	

O/P	'What If?' Scenario	Resulting Hazard	Controllability	Comments
1.6	Failure to display a traffic control command, e.g. to stop because of an incident, or to reduce the speed limit (F3.1.1.5.5).	The drivers will be unaware of the need to obey the command.	a) Nuisance Only b) Distracting	The driver should be alert. If drivers have become used to relying on such messages, then its absence will be taken to mean 'all clear'.
1.7	A set of VMS display inconsistent traffic control commands, e.g. Controlled Motorway, Tidal Flow (F3.1.1.5.5).	Drivers will, at best, become confused, and drive erratically; at worst may be placed into dangerous situations.	Up to Difficult to Control	
1.8	Failure to display advisory information (F3.1.1.5.5).	Drivers will be unaware of the information.	Nuisance Only	Assuming the information is not safety-related.
1.9	Failure to display traffic management information (F3.1.1.5.5).	There may be an increase in traffic congestion and driver frustration.	Up to Distracting	
1.10	Incorrect traffic management information transmitted (F3.1.1.5.5).	There may be an increase in traffic congestion and driver frustration.	Up to Distracting	
1.11	A set of VMS display inconsistent traffic management information (F3.1.1.5.5).	Drivers will become confused and there may be an increase in traffic congestion and driver frustration.	Up to Distracting	
1.12	Incorrect route information transmitted, e.g. to a car park (F3.1.1.5.5).	The driver uses an inappropriate route.	Up to Distracting	Assuming that the fault only affects a few vehicles in the flow of traffic.

O/P	'What If?' Scenario	Resulting Hazard	Controllability	Comments
1.13	Incorrect information about the conditions of the road network transmitted (F3.1.1.5.5).	The driver chooses an inappropriate route.	Up to Distracting	
1.14	Incorrect weights of lorries measured (too low) (F7.1.1).	Damage may be done to certain bridges.	Up to Debilitating	This depends on the actual functionality implemented.
1.15	Failure to request for drivers records (F7.1.1).	Law Enforcement Agency will not be able to prosecute that driver.	Nuisance Only	
2.1	Incorrect information received by the External Service Providers (F3.1.1.4).	Incorrect advice may be given to drivers. There may be an increase in traffic congestion and driver frustration.	Up to Distracting	
3.1	Incorrect, or no, information received about an offence by the Law Enforcement Agency (F7.3.2).	Law Enforcement Agency will not be able to prosecute that offender.	Nuisance Only	
3.2	Incorrect, or no, information received about "fraud" by the Law Enforcement Agency (F7.4).	Law Enforcement Agency will not be able to prosecute that offender.	Nuisance Only	
4.1	Incorrect, or no, information received by the maintenance organisation about the need for equipment repair (F3.5.3).	Maintenance activities may not get done, especially emergency repairs. This may affect the safety integrity of, say, traffic signals and VMS.	Up to Distracting	Assuming that the road-side controllers put the equipment into the defined safe state.
4.2	Incorrect, or no, information received by the maintenance organisation about the need for road maintenance (F3.5.1/2).	Road repairs may not get done. This may affect the safe passage of vehicles.	Up to Distracting	Driver should aware of the conditions, and should drive according to them.

O/P	'What If?' Scenario	Resulting Hazard	Controllability	Comments
4.3	Incorrect, or no, information received by the maintenance organisation about the need for de-icing (F3.5.4).	De-icing activities may not get done. This may affect the safe passage of vehicles.	Up to Difficult to Control.	If drivers have become used to relying on this function, they will not expect to meet, say, black ice.
5.1	Failure to receive crossing inhibit by the Multi-Modal Crossing (F3.1.1.5.1).	Emergency vehicle may be delayed.	Up to Difficult to Control	
6.1	Incorrect, or no, information received by the Road Network Operator on the current status of the traffic (F3.1.1.5.7).	The operators may choose an incorrect strategy, or be unaware of the need for a change in strategy, for the traffic conditions, which may lead to increased congestion and driver frustration.	Up to Distracting	It is assumed that the traffic signal controllers will continue to provide basic safety at each junction.
6.2	Incorrect, or no, information received by the Road Network Operator on the current status of, or need for, maintenance activities (F3.1.1.5.7).	As for 4.1 and 4.2	As for 4.1 and 4.2	As for 4.1 and 4.2
7.1	Incorrect, or no, information received by Related Road System (F3.1.1.4).	There may be an increase in traffic congestion and driver frustration.	Up to Distracting	
8.1	Incorrect predictions received by the Transport Planner (F3.1.1.3)	An incorrect strategy may be created.	Up to Distracting	It is assumed that the traffic signal controllers will continue to provide basic safety at each junction.

O/P	'What If?' Scenario	Resulting Hazard	Controllability	Comments
9.1	Incorrect, or no, commands provided to Pedestrian (including wheelchair) and/or Cyclist Travellers.	These Travellers may perform an action which will put them into an unsafe situation.	Up to Difficult to Control	It is assumed that these Travellers will be alert to the current situation.

8.4 Safety issues

8.4.1 System boundary and safety objectives

As a consequence of the System boundary defined for the European ITS Framework Architecture, the owners and operators of an Urban Traffic Management System will be responsible for the provision of suitable and safe commands and advice. However, they will be responsible for the way that an individual driver behaves after receiving that information. They are also responsible for maintaining the roads and equipment in a manner such that Drivers and Travellers are not put at risk.

8.4.2 Safety requirements

The high level definition of the European ITS Framework Architecture means that it has not been possible to search for the top-level safety requirements in a systematic manner. However it has been possible to identify certain safety requirements that stand out as a consequence of the hazard identification in Section 8.3.

Each hazard has been assigned a Controllability Category and equipment should be developed to a Safety Integrity Level (SIL) that corresponds to that Controllability Category in order to reduce the probability that the hazard will occur to an acceptable rate – see table below). When an item of equipment has more than one hazard associated with it, then it should normally be developed to the highest corresponding SIL.

Table 8 Relationship between Controllability Categories and SIL

Controllability Category	Safety Integrity Level	Acceptable Occurrence Rate
Uncontrollable	4	Extremely improbable
Difficult to Control	3	Very remote
Debilitating	2	Remote
Distracting	1	Unlikely
Nuisance Only	0	Reasonably possible

There are three different sets of safety requirements to reduce the risk associated with the hazards listed in Section 8.3. These may result from one or more of the following types of fault: (see also [UTMC22 2000a])

- Random - a fault that usually occurs as a result of some degradation process in a material, or when a system is operated outside its limits.
- Systematic - a fault that will always occur in the same manner in a given set of circumstances or, in other words, one that is designed into the system, albeit accidentally.

- Systemic - a major fault that pervades the whole system, or at least a large part of it. This normally occurs because of a misunderstanding and the “fault” is designed into the system deliberately.

8.4.2.1 General safety requirements

The General Safety Requirements are aimed at eliminating possible systemic faults. They are as follows:

1. The detailed System specification shall be unambiguous, in particular with respect to the following:
   (a) the boundary of the System;
   (b) the Terminators that provide the link with the “outside World”;
   (c) the definitions of the Functions that are supported;
   (d) the (data and) messages that may pass between the Functions, and between them and the Terminators.
2. A full safety analysis shall be performed whenever a specific deployment is planned.
3. Whenever two or more Functions are integrated into a system, a full safety analysis shall be performed on their emergent properties.

8.4.2.2 Safety Integrity Requirements

Safety Integrity Requirements are aimed at managing systematic faults, and are measures that should be taken to minimise the possibility of each hazard occurring. They are as follows:

1. All message channels used by a safety-related Function(s) shall be demonstrated to be (sufficiently) robust, e.g. capable of handling a pre-defined number and type of errors in transmission, provide a guaranteed delivery time for each message.
2. The relationship between the System and its users shall be considered during the life-cycle, in particular with respect to the HMI and to the way that traffic may behave (see [UTMC22 2000a]).
3. Equipment components, and the way that they are assembled, shall be chosen so that the required reliability can be achieved (see [UTMC22 2000a]).
4. The processes used to develop safety-related electrical, electronic or programmable electronic equipment shall be suitable for the SIL of the part of the System to which they belong. These may consist of measures to avoid faults and measures to control faults (see [UTMC22 2000a]).

8.4.2.3 Safety Functional Requirements

Safety Functional Requirements are mainly targeted at handling random faults, though some systematic faults may also be ameliorated. Whilst most of them will be identified during the PSA of plans for specific deployments (see Section 8.4.2.1), a few generic safety functional requirements can been identified at this stage:

1. Non-safety-related equipment shall not share the same message channel(s) as safety-related equipment unless (sufficient) independence can be demonstrated.
2. The ‘basic (or primary) safety’ of any safety-related equipment, e.g. traffic signals, shall not be dependant on long distance communications.
3. The messages presented to travellers using one, or more, sets of devices (e.g. a sequence of VMS) and/or technologies (e.g. VMS and RDS/TMC (via a Service Provider)) shall be consistent.
4. The System shall be capable of reconciling conflicting traffic management strategies.

8.5 Conclusion

The objective of this PSA has been to identify and advise on any inherent hazards that are associated with an “example System”. It has been possible to show that a typical Urban Traffic Management System is likely to have a number of safety hazards associated with it whose severity cannot be ignored. However, the high level nature of the European ITS Framework Architecture means that more precise results must be determined for each specific deployment.

The principal hazards identified during the PSA were associated with traffic management and traffic control at junctions. When junction controllers are integrated to provide co-ordinated instructions along a traffic corridor it is possible to disrupt the flow of traffic as well as to make it more coherent. Such disruption will result in an increased number of conflict situations and even accidents. This is additional to the basic safety hazard associated with any individual set of traffic signals, e.g. simultaneous green on conflicting arms of the junction.

The use of VMS can also introduce some hazards. If they cannot be read or understood easily then they will be a distraction to the driver. They should also provide consistent information when a number are being used in an integrated system. The potential severity of the hazard increases when VMS are used to provide commands to drivers rather than advice or information.

It is not possible to provide more than generic safety requirements at this stage, given the large number of possible ways in which such Systems are likely to be created. For example, the setting of the message on a VMS can be a manual operation with a controller choosing from a number of pre-set messages, to a fully automated system where the controller is only informed as to what has been done. Each version will need a PSA to identify the hazards that are specific to that System.

An attempt was also made to assign Controllability Categories to the various hazards, but on most occasions it was only possible to state a range, since the precise value would depend on the exact deployment being proposed.

Probably the most important result of this PSA is that it demonstrates that there are very many potential hazards in an Urban Traffic Management System such as P30. These are over and above those related solely to the basic safety of a road junction controlled by traffic signals. In addition it will be necessary to perform PSA’s on each and every planned deployment in order to identify the precise hazards associated with the desired applications, and hence to confirm the SIL(s) for the equipment that will comprise a particular System.

9. References

(a) European ITS Functional Architecture Deliverable Document (D 3.1), Issue 1, August 2000. It can be downloaded from the European Commission Web Site at: http://www.trentel.org/ and following the links “Transport->Deployment Information->System Architecture->System Architecture Library.

(b) European ITS User Needs Deliverable Document (D 2.02), Issue 1, August 200. This document contains Version 4 of the User Needs spreadsheet. It can be downloaded from the European Commission Web Site at: http://www.trentel.org/ and following the links “Transport->Deployment Information->System Architecture->System Architecture Library.

(c) European ITS Framework Architecture Overview Document (D 3.6), Issue 1, August 2000. It can be downloaded from the European Commission Web Site at: http://www.trentel.org/ and following the links “Transport->Deployment Information->System Architecture->System Architecture Library.

(d) European ITS Deployment Report (D 4.2), Issue 1.1, August 2000. It can be downloaded from the European Commission Web Site at: http://www.trentel.org/ and following the links “Transport->Deployment Information->System Architecture->System Architecture Library.

(e) European ITS Framework Architecture Communications Architecture Document (D 3.3), Issue 1, August 2000. It can be downloaded from the European Commission Web Site at: http://www.trentel.org/ and following the links “Transport->Deployment Information->System Architecture->System Architecture Library.

(f) [IEC 61508]
IEC 61508, *Functional Safety - Safety-Related Systems*, International Electrotechnical Commission, 1999.

(f) [PASSPORT 1995]
PASSPORT, *Framework for Prospective System Safety Analysis Volume 1 - Preliminary Safety Analysis*, EC DRIVE II Project PASSPORT (V2058), Deliverable N° 9a, 1995.

(g) [UTMC22 2000a]
UTMC22, *Framework for the Development and Assessment of Urban Traffic Management and Control Systems*, DETR UTMC Programme, 2000. This can be found through the UTMC Web Site at: http://www.utmc.org.uk.

(h) [UTMC22 2000b]
UTMC22, *Preliminary Safety Analysis of the Urban Traffic Management and Control Systems Architecture*, DETR UTMC Programme, 2000. This can be found through the UTMC Web Site at: http://www.utmc.org.uk.

Annex 1 – Descriptions of "example Systems"

This Document is the first Annex (Annex 1) to the Main Document part of the European ITS Physical Architecture Deliverable Document (D3.2). It provides a description of each of the "example Systems" that have been used to show how the Functional Architecture can be used to create "real" Systems or Architectures. These "example Systems" illustrate examples of products that could be produced to fulfil some of the European ITS User Needs, or Architectures that could be used for further ITS development at National or local level. This approach has been adopted because there are many ways in which a Physical Architecture can be produced from a Functional Architecture.

The description of each "example System" includes details of the parts of the Functional Architecture (Functions, Data Flows and Data Stores) that it includes, as well as providing information on what the System or Architecture can provide. The final part of each description includes a section on "Key Issues". These are "Issues" that have been addressed and resolved to enable each "System" to be successfully deployed and implemented. They are also shown together in a single Chapter in the Main Document.

Annex 2 – Function and Data Store Descriptions

This Document is the second Annex (Annex 2) to the Main Document part of the European ITS Physical Architecture Deliverable Document (D3.2). It contains the descriptions for each of the Functions and Data Stores in the European ITS Functional Architecture, plus templates for use in creating the descriptions of "example Systems".

The Function and Data Store descriptions are intended to provide a "quick and easy reference" to the descriptions that can be used when studying the "example Systems" that are described in the first Annex (Annex 1) to the European ITS Physical Architecture Main Document. Using these descriptions should avoid the need to refer to either Annex 1 or Annex 3 of the European ITS Functional Architecture Deliverable Document (D 3.1). Note that in the case of the Functions, only the Overview and User Needs list are included in this Annex.

The templates cover document structure, Diagrams and Tables that will be needed as part of the descriptions of any "example Systems" that are created in the same way as those in Annex 1 of the European ITS Physical Architecture Main Document. Further details of when and how they should be used will be found in Chapter 4 of the Main Document.

Appendix 1 P30 I-O Matrix

The following pages contain a table of the Input/Output (I/O) Matrix used for the Preliminary Safety Analysis (PSA) that has been performed on one of the “example Systems”. This is the “example System” for Urban Traffic Management – P30.

The table has been produced from a Microsoft® Excel spreadsheet into which the data was originally entered. If a PSA is carried out on any other “example Systems” then either the table on the following pages can be used, or the table can be turned back into an Excel spreadsheet.

Table 9 P60 I/O Matrix – Part 1

	Data Flows - Input	Central	Roadside	Vehicle	I/face to Driver (1)	I/face to External Service Provider (2)
Data Flows - Output					td-plse_record_request/ td-urban_traffic_commands td.e-green_wave_infos +td.e-global_emergency_report	tesp.ttip-urban_traffic_data/ tesp.b-urban_traffic_data
Central		F3.1.1.5.2, F3.1.1.5.9, F7.1.2, F7.3.1				
Roadside			F3.1.1.5.3, F3.1.1.5.4, F3.1.1.5.6			
Vehicle						
I/face from Driver	fd-psle_record/ fd.e-green_wave_request +fd.e-individual_emergency_progress_report				F7.1.1, F2.1.3	
I/face from Law Enforcement Agency	flea-psle_fraud_data_request/ flea-psle_rules	F7.5.1				
I/face from Maintenance Organisation	fmo-update_activity_status	F3.5.6				
I/face from Multi-Modal Systems	fmms.mmc-urban_crossing_request				F3.1.1.5.5	
I/face from Road Network Operator	fo.rno-urban_commands/ fo.rno-maintenance_commands					
I/face from Related Road System	frrs-urban_data_updates					F3.1.1.4

	Data Flows - Input	Central	Roadside	Vehicle	I/face to Driver (1)	I/face to External Service Provider (2)
Data Flows - Output					td-plse_record_request/ td-urban_traffic_commands td.e-green_wave_infos +td.e-global_emergency_report	tesp.ttip-urban_traffic_data/ tesp.b-urban_traffic_data
I/face from Road Pavement	frp-short_term_wearing_state/ frp-long_term_wearing_state/ frp-current_conditions					
Interface from Traffic	ftrfc-urban_traffic_data_flow/ ftrfc-carpark_vehicle_data/ ftrfc-local_traffic_ presence_data/ ftrfc-urban_traffic_identity_data		F3.1.1.1 F3.1.1.2 F3.1.1.5.8		F3.1.1.5.5	
I/face from Transport Planner	ftp-urban_traffic_prediction_commands					
I/face from Traveller	ft-urban_data				F3.1.1.5.5	
I/face from Vehicle	fv_psle_characteristics/ fv.ptv-local_priority_request				F7.1.1 F3.1.1.5.5	
I/face from Weather System	fws-short_term_maintenance_conditions/ fws-long_term_maintenance_conditions/ fws-ice_formation_conditions					
D1.6 Fraud Store	psle_fraud_characteristics/ psle_fraud_history	F7.3.1				
D3.1 Urban Traffic Data Store	mt_read_urban_traffic_data					F3.1.1.4
D3.6 Maintenance Data Store	mt_read_maintenance_data	F3.5.6				
D3.7 Urban Road Static Data Store	mt_urban_static_data_read	F3.1.1.5.9				
D7.1 Rules Store	psle_rules/ plse_fraud_classification	F7.1.2, F7.3.1				

Table 10 P60 I/O Matrix – Part 2

	Data Flows - Input	I/face to Law Enforcement Agency (3)	I/face to Maintenance Organisation (4)	I/face to Multi-Modal System (5)
Data Flows - Output		tlea-psle_prosecution_file/ tlea-psle_fraud_prosecution_file +tlea-psle_fraud_notification	tmo-short_term_activities/ tmo-long_term_activities/ tmo-de-icing_tasks/ tmo-equipment_tasks	tmms.mmc-urban_crossing_inhibit
Central		F7.3.2	F3.5.3	F3.1.1.5.1
Roadside				
Vehicle				
I/face from Driver	fd-psle_record/ fd.e-green_wave_request +fd.e-individual_emergency_progress_report			
I/face from Law Enforcement Agency	flea-psle_fraud_data_request/ flea-psle_rules	F7.4		
I/face from Maintenance Organisation	fmo-update_activity_status			
I/face from Multi-Modal Systems	fmms.mmc-urban_crossing_request			
I/face from Road Network Operator	fo.rno-urban_commands/ fo.rno-maintenance_commands			
I/face from Related Road System	frrs-urban_data_updates			
I/face from Road Pavement	frp-short_term_wearing_state/ frp-long_term_wearing_state/ frp-current_conditions		F3.5.1, F3.5.2, F3.5.4	

	Data Flows - Input	I/face to Law Enforcement Agency (3)	I/face to Maintenance Organisation (4)	I/face to Multi-Modal System (5)
Data Flows - Output		tlea-psle_prosecution_file/ tlea-psle_fraud_prosecution_file +tlea-psle_fraud_notification	tmo-short_term_activities/ tmo-long_term_activities/ tmo-de-icing_tasks/ tmo-equipment_tasks	tmms.mmc-urban_crossing_inhibit
Interface from Traffic	ftrfc-urban_traffic_data_flow/ ftrfc-carpark_vehicle_data/ ftrfc-local_traffic_ presence_data/ ftrfc-urban_traffic_identity_data			
I/face from Transport Planner	ftp-urban_traffic_prediction_commands			
I/face from Traveller	ft-urban_data			
I/face from Vehicle	fv_psle_characteristics/ fv.ptv-local_priority_request			
I/face from Weather System	fws-short_term_maintenance_conditions/ fws-long_term_maintenance_conditions/ fws-ice_formation_conditions		F3.5.1, F3.5.2, F3.5.4	
D1.6 Fraud Store	psle_fraud_characteristics/ psle_fraud_history	F7.3.2		
D3.1 Urban Traffic Data Store	mt_read_urban_traffic_data			
D3.6 Maintenance Data Store	mt_read_maintenance_data			
D3.7 Urban Road Static Data Store	mt_urban_static_data_read			
D7.1 Rules Store	psle_rules/ plse_fraud_classification			

Table 11 P60 I/O Matrix – Part 3

	Data Flows - Input	**I/face to Road Network Operator (6)**	**I/face to Related Road System (7)**	**I/face to Transport Planner (8)**
Data Flows - Output		to.rno-urban_responses/ to.rno-maintenance_ responces	trrs-urban_data_updates	ttp-urban_traffic_prediction_responses
Central				
Roadside				
Vehicle				
I/face from Driver	fd-psle_record/ fd.e-green_wave_request +fd.e-individual_emergency_progress_report			
I/face from Law Enforcement Agency	flea-psle_fraud_data_request/ flea-psle_rules			
I/face from Maintenance Organisation	fmo-update_activity_status			
I/face from Multi-Modal Systems	fmms.mmc-urban_crossing_request			
I/face from Road Network Operator	fo.rno-urban_commands/ fo.rno-maintenance_commands	F3.1.1.5.7, F3.5.5		
I/face from Related Road System	frrs-urban_data_updates		F3.1.1.4	
I/face from Road Pavement	frp-short_term_wearing_state/ frp-long_term_wearing_state/ frp-current_conditions			

	Data Flows - Input	I/face to Road Network Operator (6)	I/face to Related Road System (7)	I/face to Transport Planner (8)
Data Flows - Output		to.rno-urban_responses/ to.rno-maintenance_ responces	trrs-urban_data_updates	ttp-urban_traffic_prediction_responses
Interface from Traffic	ftrfc-urban_traffic_data_flow/ ftrfc-carpark_vehicle_data/ ftrfc-local_traffic_ presence_data/ ftrfc-urban_traffic_identity_data			
I/face from Transport Planner	ftp-urban_traffic_prediction_commands			F3.1.1.3
I/face from Traveller	ft-urban_data			
I/face from Vehicle	fv_psle_characteristics/ fv.ptv-local_priority_request			
I/face from Weather System	fws-short_term_maintenance_conditions/ fws-long_term_maintenance_conditions/ fws-ice_formation_conditions			
D1.6 Fraud Store	psle_fraud_characteristics/ psle_fraud_history			
D3.1 Urban Traffic Data Store	mt_read_urban_traffic_data		F3.1.1.4	
D3.6 Maintenance Data Store	mt_read_maintenance_data			
D3.7 Urban Road Static Data Store	mt_urban_static_data_read			
D7.1 Rules Store	psle_rules/ plse_fraud_classification			

Table 12 P60 I/O Matrix – Part 4

	Data Flows - Input	I/face to Traveller (9)	D1.6 Fraud Store	D3.1 Urban Traffic Data Store
Data Flows - Output		tt-urban_traffic_commands	plse_fraud_notification/ +plse_prosecution_file	mt_load_urban_traffic_data
Central				
Roadside				
Vehicle				
I/face from Driver	fd-psle_record/ fd.e-green_wave_request +fd.e-individual_emergency_progress_report			
I/face from Law Enforcement Agency	flea-psle_fraud_data_request/ flea-psle_rules		F7.4	
I/face from Maintenance Organisation	fmo-update_activity_status			
I/face from Multi-Modal Systems	fmms.mmc-urban_crossing_request	F3.1.1.5.5		
I/face from Road Network Operator	fo.rno-urban_commands/ fo.rno-maintenance_commands			
I/face from Related Road System	frrs-urban_data_updates			F3.1.1.4
I/face from Road Pavement	frp-short_term_wearing_state/ frp-long_term_wearing_state/ frp-current_conditions			
Interface from Traffic	ftrfc-urban_traffic_data_flow/ ftrfc-carpark_vehicle_data/ ftrfc-local_traffic_ presence_data/ ftrfc-urban_traffic_identity_data	F3.1.1.5.5		

	Data Flows - Input	I/face to Traveller (9)	D1.6 Fraud Store	D3.1 Urban Traffic Data Store
Data Flows - Output		tt-urban_traffic_commands	plse_fraud_notification/ +plse_prosecution_file	mt_load_urban_traffic_data
I/face from Transport Planner	ftp-urban_traffic_prediction_commands			
I/face from Traveller	ft-urban_data	F3.1.1.5.5		
I/face from Vehicle	fv_psle_characteristics/ fv.ptv-local_priority_request	F3.1.1.5.5		
I/face from Weather System	fws-short_term_maintenance_conditions/ fws-long_term_maintenance_conditions/ fws-ice_formation_conditions			
D1.6 Fraud Store	psle_fraud_characteristics/ psle_fraud_history			
D3.1 Urban Traffic Data Store	mt_read_urban_traffic_data			
D3.6 Maintenance Data Store	mt_read_maintenance_data			
D3.7 Urban Road Static Data Store	mt_urban_static_data_read			
D7.1 Rules Store	psle_rules/ plse_fraud_classification			

Table 13 P60 I/O Matrix – Part 5

	Data Flows - Input	**D3.6 Maintenance Data Store**	**D3.7 Urban Road Static Data Store**	**D7.1 Rules Store**
Data Flows - Output		mt_load_maintenance_data	mt_urban_static_data_update	psle_rules
Central			F3.1.1.5.9	
Roadside				
Vehicle				
I/face from Driver	fd-psle_record/ fd.e-green_wave_request +fd.e-individual_emergency_progress_report			
I/face from Law Enforcement Agency	flea-psle_fraud_data_request/ flea-psle_rules			F7.5.1
I/face from Maintenance Organisation	fmo-update_activity_status	F3.5.6		
I/face from Multi-Modal Systems	fmms.mmc-urban_crossing_request			
I/face from Road Network Operator	fo.rno-urban_commands/ fo.rno-maintenance_commands			
I/face from Related Road System	frrs-urban_data_updates			
I/face from Road Pavement	frp-short_term_wearing_state/ frp-long_term_wearing_state/ frp-current_conditions			

	Data Flows - Input	D3.6 Maintenance Data Store	D3.7 Urban Road Static Data Store	D7.1 Rules Store
Data Flows - Output		mt_load_maintenance_data	mt_urban_static_data_update	psle_rules
Interface from Traffic	ftrfc-urban_traffic_data_flow/ ftrfc-carpark_vehicle_data/ ftrfc-local_traffic_ presence_data/ ftrfc-urban_traffic_identity_data			
I/face from Transport Planner	ftp-urban_traffic_prediction_commands			
I/face from Traveller	ft-urban_data			
I/face from Vehicle	fv_psle_characteristics/ fv.ptv-local_priority_request			
I/face from Weather System	fws-short_term_maintenance_conditions/ fws-long_term_maintenance_conditions/ fws-ice_formation_conditions			
D1.6 Fraud Store	psle_fraud_characteristics/ psle_fraud_history			
D3.1 Urban Traffic Data Store	mt_read_urban_traffic_data			
D3.6 Maintenance Data Store	mt_read_maintenance_data			
D3.7 Urban Road Static Data Store	mt_urban_static_data_read			
D7.1 Rules Store	psle_rules/ plse_fraud_classification			