European ITS Framework Architecture

-

Overview

D3.6 - 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
RIJKSWATERSTAAT
ERTICO
MIZAR

Document control sheet

Activity name:	KAREN
Work area:	Framework Architecture Development - WP2
Document title:	European ITS Framework Architecture Overview
Document number:	D3.6
Electronic reference:	
Main author(s) or editor(s):	Richard Bossom
Other author(s):	Victor Avontuur, Jean-François Gaillet, Gino Franco and Peter Jesty
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*	*July 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

Figures

Tables

Executive Summary

This Document is provided to act as the “base” Document for the European ITS Framework Architecture Documentation. It fulfils this role in three ways.

Firstly this Document provides a description of the general architecture development process that has been followed by the KAREN Project to produce the European ITS Framework Architecture.

Secondly this Document provides an “overview” of all six of the actual European ITS Framework Architecture Deliverable Documents produced by Work Package 3. It does this by including the Executive Summaries from each of these six Documents. Between them these Documents describe the main parts of the Framework Architecture (Functional, Physical and Communications Architectures), the report on the results of the studies into Cost Benefits issues and a study of the way that Models can be used in ITS. The results of the work on Deployment were produced as an internal report. They have been included in the Deployment Deliverable Document (D 4.2) produced by another part of the KAREN Project.

Thirdly, this Document contains any material that is common to some or all of the other six Documents. It therefore includes sections covering the need for a European Framework Architecture, historical background, the relationship with other Architecture activities and Framework Architecture characteristics. In particular there is a section devoted to a comparison between the US National ITS Architecture and the European ITS Framework Architecture.

The Document also includes an Annex as a separate Document. This Annex provides the Trace Tables showing the relationship between User Needs and Functions. These Tables are used in the Architecture and Physical System creation processes described in Chapters 4 and 5 of the European ITS Physical Architecture Document (D 3.2).

1. Introduction

1.1. Outline

This Document provides acts as the "base Document" for the other European ITS Framework Architecture Deliverable Documents that have been produced by Work Package 3 of the KAREN Project and which are in the public domain. It includes an overview of each Document in the form of its Executive Summary, plus background material on the development of system architecture within Europe and the reasons for the establishment of the KAREN Project. Parts of the Document describe the general system architecture development process used by the KAREN and a comparison with other Architectures.

1.2. Scope

This document is one of the Deliverable Documents that has been provided by the KAREN Project as part of its work to define the European ITS Framework Architecture. It covers all of the six publicly available Framework Architecture Deliverable Documents that have been produced by Work Package 3 within the Project. The other documents in the set are as follows:

D3.1 European ITS Functional Architecture - this document
D3.2 European ITS Physical Architecture
D3.3 European ITS Communications Architecture
D3.4 European ITS Cost Benefits Report
D3.6 European ITS Framework Architecture Overview
D3.7 European ITS Models for ITS deployment

The work on ITS Deployment included in D 3.5, is not included in the above list as the Document was produced for use within the KAREN Project. Its content has been included in the Deployment Study Report (D 4.2) produced by Work Package 4. Full details of Documents referred to in this Document, including sources other than the European ITS Framework Architecture CD-ROM, are provided in the reference section of this Document.

1.3. List of Abbreviations

The following abbreviations are used in this Document and its Annex Document. Each of the other European ITS Framework Architecture Documents will have its own abbreviation section.

DRIVE	Dedicated Road Infrastructure for Vehicle safety in Europe
ITE	Integrated Transport Environment
ITS	Intelligent Transport Systems
KAREN	Keystone Architecture Required for European Networks

SATIN	System Architecture and Traffic control Integration
QUARTET	QUadrilateral Advanced Research on Telematics for Environment and Transport
QUARTEX	QUARTet Extension
RAID	Risk Analysis for ITS Deployment
TARDIS	Traffic and Roads- DRIVE Integrated Systems
TCC	Traffic Control Centre
TIC	Traffic Information Centre
TICS	Transport Information and Control Systems
US(A)	United States (of America)
UML	Unified Modelling Language

2. Background

2.1. The Need for a European ITS Framework Architecture

During the last decade, Europe has gained a significant early lead in the development and deployment of telematics technologies in transport applications. The pace at which this progress has been achieved is impressive and appears to be a direct result of a clear focus on the development of well-defined ‘building blocks’ such as driver information, traffic control, route guidance or public transport information systems.

System engineering has also focused along similar lines, with the development of disparate architectures which represent thorough system engineering approaches to specific and focused applications areas. In some cases, wider system architectures encompassing multiple applications have been developed, although they have been designed to meet local requirements at city and/or regional level rather than European.

Europe is now faced with the challenge of integrating these ‘building blocks’ into a pan-European system architecture that would offer some degree of consistency and synergy across applications. The need for progress and efficiency is great and confronts Europe with a huge challenge. How can the wide-scale deployment of **Transport Telematics Systems,** now called **Intelligent Transport Systems** or **ITS,** in Europe be stimulated and co-ordinated?

The implementation of ITS has to yield benefits to the public sector and end-users, as well as to private industry and the service sector. Another important consideration is the competitiveness of the European ITS industry, both at home and in other parts of the world.

One way to meet these challenges is to require all stakeholders to adhere to an agreed European Framework. This Framework must accommodate national plans and support the various efforts in research, standardisation, deployment and investment. It must also provide a migration plan which incorporates and builds upon existing ‘legacy’ systems.

2.2. Historical Background

2.2.1. Introduction

The basis for an Integrated Transport Environment (ITE) concept was laid down in the EC Framework II DRIVE I project TARDIS (V1018) and was further developed by a number of projects during the EC Framework III DRIVE II Programme.

Whilst both of these programmes were oriented towards the road mode of transport, a number of fundamental activities were undertaken which are also applicable to the other modes now being considered in the EC Framework IV Telematics for Transport programme. An ITE can be considered to be made up of a number of different application areas, and the design of the ITE must enable these different areas to exchange information in a common format, to update their strategies, and to perform co-ordinated control actions on the traffic network. The reasons for doing this are:

- To provide a consistent and up-to-date picture of the traffic and transport network state;
- To control the network state around a common reference optimum using collective control actions (e.g. traffic signals, VMS);
- To disseminate relevant and non-conflicting information to users, operators and authorities.

The common reference optimum referred to above will be dictated by local transport management policies. It will therefore vary at least from country to another, and possibly between regions within a country.

2.2.2. The SATIN Task Force

During DRIVE II the SATIN Task Force was inaugurated on 31 January 1994 in order to extend the work that had begun within the forum of Topic Group 10, which was the system architecture group set up as part of the EC concertation activities during the DRIVE II programme. The work included the following items:

- Recommendation of a methodology suitable for the preparation of transport telematic architectures, including assessment (optimisation) criteria and safety analysis;
- Proposal of a system architecture for the road ITE that would include all transport telematic services.

The work undertaken by SATIN was limited in time, and therefore in scope, taking contributions from a large number of DRIVE II projects. Its goal was to try and form a consensus on the methodological issues by identifying the good architecture practices developed during the DRIVE II programme – see reference 8(i).

In parallel with the collection and synthesis of results, SATIN also developed a methodology to support the development of transport telematic system architectures – see reference 8(b).

2.2.3. QUARTEX

QUARTEX was an extension of the DRIVE II projects QUARTET (V2018) and GERDIEN (V2044) whose aim was to provide guidelines for the practical assessment of road ITE architectures. This work produced the following.

- A methodology for the assessment of architectures – see reference 8(g).
- A software based tool for the analysis of architectures – see reference 8(h).

2.2.4. CONVERGE-SA

The EC Framework IV Telematics for Transport project CONVERGE-SA (TR1101) built on the work of both SATIN and QUARTEX. It produced the following:

- Guidelines for the development and assessment of system architectures – see reference 8(a);
- a system architecture analysis tool – see reference 8(c).

The CONVERGE Guidelines have been used as a source of reference by the KAREN Project for the definition of the methodology that it has used to create the European ITS Framework Architecture. This methodology is very closely related to the general Architecture creation process that is described in Chapter 5 of this Document.

The CONVERGE Guidelines also form the basis for the methodology that is described in the European ITS Physical Architecture Document (D 3.2) for the creation of "example Systems" – see reference 8(m). These "example Systems" may be either Physical Systems that can be produced and implemented, or other Architectures. The Physical Systems are therefore the same as the "ITS Systems" shown by Figure 2 in the Appendix to this Document.

3. The KAREN Project

3.1. Introduction

The KAREN (Keystone Architecture Required for European Networks) Project has created a minimum stable framework necessary for the deployment of working and workable ITS within the European Union until at least 2010. It is the European ITS system architecture effort, requested by the High Level Group on road transport telematics, approved by the European Council of Ministers and funded by DGXIII as part of the 4th Framework Programme. The project began on 1st April 1998, and aimed to deliver an agreed, and promoted, Transport Telematics Framework Architecture which will:

- Define the necessary elements for an open market of ITS products throughout Europe, and the rest of the world, for European ITS industry;
- Be the basis for building consensus on issues that still prevent wide-spread deployment of ITS in Europe, and hence permit all categories of user to purchase cost effective ITS products that will work in the same way throughout Europe;
- Provide a bridge between the ITS community and those creating the current and future technologies that may be used by ITS;
- Be a guide for public investments on the basic infrastructure necessary for the deployment of the ITS services;
- Support the identification of areas where new research and demonstrations are needed.

The KAREN Project has produced the following principal outputs – see references (k) to (q) in Chapter 8:

- A consolidated List of European ITS User Needs (this document);
- A European ITS Architecture Framework (documents D3.x);
- Recommendations for the deployment of the Framework Architecture (documents D4.x).

All of these may be found on either the European ITS Framework Architecture CD-ROM issued by the European Commission or one of the pages in the European Commission Web Site at “http://www.trentel.org/index.htm”.

3.2. The European ITS Framework Architecture

The European ITS Framework Architecture must accommodate national plans and support the various efforts in research, standardisation, deployment and investment. It must also provide a migration plan which incorporates and builds upon existing ‘legacy’ systems. (For a discussion as to what is meant by a Framework Architecture see Appendix A).

A common Framework provides specifications that enable:

- Compatibility of information delivered to end-users through different media;
- Compatibility of equipment with infrastructures, thus enabling seamless travel across Europe;
- A basis for regional, national and European authorities to produce master plans and recommendations to facilitate ITS deployment;
- An open market for services and equipment where compatible sub-systems are offered (no more ad-hoc solutions);
- Economies of scale in equipment manufacture permitting competitive prices and cheaper investments when compatibility is guaranteed;
- A known market place into which producers can supply products with reduced financial risk.

3.3. Relationship with other Architecture activities

3.3.1. International Activities

There are two other principal international (i.e. outside Europe) architecture activities from which information is readily available, namely the US National ITS Architecture and the work being done by ISO/TC204/WG1. It should be noted that work is also going on elsewhere, e.g. Japan, Korea, and Australia. Both the US National Architecture and the ISO/TC204/WG1 Reference Architecture begin with a 'green field' and do not consider any existing systems or equipment. Although they were supposed to be reasonably comprehensive, the US User Service Requirements are now recognised as being not totally applicable outside the USA. The 32 Transport Information and Control System (TICS) Fundamental Services produced by ISO/TC204/WG1 have been based mainly on the US User Service Requirements and, as a result, they tend to be oriented towards US problems and desires.

The US National ITS Architecture and the ISO TICS Reference Architecture are discussed in more detail in Chapter 7 of this Document.

3.3.2. RAID

This is a study on system architecture performed as part the EC 4th Framework Programme from January 1998 until March 1999 by a sub-set of the KAREN consortium. The objective of RAID was to identify the obstacles that might prevent the successful deployment of ITS within the EU, and to recommend possible solutions for overcoming them.

The risks that were considered are those that may hinder, or prevent, the implementation of:

- The European ITS Framework Architecture in general;
- Any particular ITS.

All areas of ITS were considered, using the knowledge and expertise of both the members of the team, and the extensive group of external organisations established by the KAREN

Project. Although not a specific objective, the output of the RAID project serves as supplementary background information to the work of the KAREN Project. The RAID report that describes the principal risks, and their suggested mitigation strategies, is included on the European ITS Framework Architecture CD-ROM that is available from the European Commission.

3.3.3. COMETA

The COMETA project (TR4005) is running concurrently with KAREN and has the objective of developing an open architecture for on-board systems in commercial vehicles. In particular COMETA is defining and designing modular associations of a wide range of on-board functions to support professional transport operations by road, and efficient interfaces with a global transport telematics system. Examples of components considered for the on-board systems include devices for on-board data capturing and data processing, driving assistance, tools for on-board remote diagnostics, mechanisms for data exchange within the vehicle and with the different actors involved in the transport chain, the digital tachograph, devices for electronic tolling etc. Further information can be found about the COMETA Project from reference 8(j), or from the Project's own Web Site at "http://www.cometa-project.com/".

COMETA and KAREN have some members in common, and there is active liaison and co-operation between them. A practical consequence of this liaison and co-operation is that the Architectures are compatible.

4. Framework Architecture Characteristics and Definitions of Terms

4.1. Introduction

Before embarking on an overview of what is in the European ITS Framework Architecture it is important to look at its characteristics and to provide a definition of many of the terms used in Architecture development. These will help the readers of this and the other European ITS Framework Architecture Documents understand what the Architecture is and is not.

4.2. Definition of a Framework Architecture

A Framework Architecture defines and describes what needs to be included in a System that can fulfil the requirements of a set of User Needs. For the European ITS Framework Architecture these are provided in a separate document (see reference 8(k)) produced by another part of the KAREN Project.

A Framework Architecture expresses a System in number of ways. These are provided by the different parts of the Architecture which are as follows.

- Functional Architecture: the functionality needed by the System to fulfil the User Needs.
- Physical Architecture: the way in which the functionality can be implemented as Applications to fulfil the User Needs. These Applications may also fulfil the User Needs in ways that cannot be expressed in functional terms, such as physical characteristics. They represent one way of creating Applications, there may of course be others, but the existence of these will depend on such things as implementation constraints for individual Systems.
- Communications Architecture: the links that enable data to be exchanged between the Applications in the Physical Architecture, and between the Applications and the outside World.

The European ITS Framework Architecture also includes other things such as the Cost Benefit Study and the Deployment Study. These describe the benefits that can be expected to accrue from the deployment of the Architecture, and some of the means by which existing systems can be migrated to conform with the Architecture.

Each of the five parts to an Architecture described above have been produced for the European ITS Architecture. They are each in their own separate Document of which this Document provides an overview. More details of what is in these Documents will be found in Chapter 6 of this Document.

4.3. Framework Architecture Characteristics

In generic terms the European ITS Framework Architecture has some main, or fundamental, characteristics. These can be expressed in the following two ways.

The Framework Architecture **is** all of the following.

- Open:- this means that all suppliers, operators and users will be able to make use of what is in the Architecture. Put another way, the Architecture does not set out to exclude anyone.

- Multi-modal:- this means that the Architecture is designed to apply to all forms of road transport, not just the private car. There are also interfaces to other forms of transport that do not use the road. Examples are heavy rail, sea and air.

- Technology Independent:- the Architecture does not require or promote the use of a particular technology. It does however promote the use of generic solutions for which several technologies are available. For example the Communications Architecture (see reference 8(n)) may promote the use of wireless communications in some of its analysis, but not the exact type. In other words there are no requirements to use particular frequencies, protocols, etc.

The Framework Architecture is **not** any of the following.

- A System or Component Design:- it is not be possible to produce anything (hardware or software) directly from the contents of the Framework Architecture. The definitions and descriptions in the Architecture are at a level that is too high for this to be directly achieved. The concept of "Levels of Architecture" are discussed in some detail in the CONVERGE Architecture Guidelines – see reference 8(a).

- A System Specification:- the Architecture will not be presented in such a way that it can be directly used as the specification for a System, whether it be for some hardware, or some software. However the parts of the Architecture can be used as the starting point for the production of System Specifications for the components of individual Systems. This is discussed in the Physical Architecture Deliverable Document – see reference 8(m).

An appreciation of these characteristics will help the readers of these documents and the users of the Architecture to understand its purpose and its limitations.

4.4. Use of System Architecture Names and Terms

4.4.1. Introduction

There are several System Architecture names and terms that are in use both in Europe and in the rest of the World. Some of them arise from differences in approach to Architecture development, whilst others are different names for similar parts of Architectures. This section of the Document provides a definition of those of the generally used names terms that are and relevant to the European ITS Framework Architecture.

4.4.2. Definitions of System Architecture Names and Terms

The Names and Terms are listed below in alphabetical order and not in any order of importance.

(a) Communications Architecture

This Architecture defines the communication mechanisms that are to be used to link the components of a Physical Architecture – see later definition. For further information see Chapter 5, where its use within the European ITS Framework Architecture is also discussed.

(b) Conceptual Model

In general terms, Models are the way people think about things. When a person has a mental image in their mind about what a System looks like, or the way that a System behaves, then it is a Conceptual Model. More strictly it is any high-level description that is able to:

- represent a certain system issue;
- ease communication and understanding about an issue;
- simplify a subject or an issue;
- present overall relationships;
- define certain constructs.

For further discussion on the use of Models in the ITS environment and examples of some European Models, see reference 8(o).

(c) Context Diagram

This diagram shows the context in which the System that the Architecture represents will exist. The context is the part of the outside World with which the System must interface if it is to operate and deliver the Services and facilities required by the users. The outside World is represented by Terminators – see below.

(d) Control Architecture

This Architecture is used to show the flow of Control Data within an Architecture. Control Data is the data that is used to cause parts of the Architecture to perform functionally. Thus it can be used to activate a Function or Process (see below). Control Data may be sent at the same time and to the same destination as other Data, e.g. traffic flows, video images, etc.

In the European ITS Framework Architecture, the notion of control is captured using “trigger” data flows in the Functional Architecture. These are data flows that can be used by one Function to trigger activity by another Function. Further discussion of this is provided in Chapter 2 of the Functional Architecture Deliverable Document (D 3.1).

(e) Cost Benefits

This are the results of an analysis of the costs of producing something when compared with the benefits that it will provide. For an Architecture the “something” is whatever is produced from it. In the case of the European ITS Framework Architecture this may be either Physical Systems or Architectures. For further information see Chapter 5, where its use within the European ITS Framework Architecture is also discussed

(f) Data

Data are raw values, e.g. numbers or characters, that after processing will define states or conditions. In Architectures they apply to anything that is collected from a source that is outside the System and that is represented by the Architecture. There can be two basic types of data, analogue and digital. The analogue type will usually be processed at the interface to the System to convert it into digital data. Digital data may also be collected directly, e.g. from another system. It may remain “data” inside the Architecture even after processing, analysis and storage.

(g) Data Architecture

This is another name for an Information Architecture – see below.

(h) Data Flow

This is the definition of the way that Data moves from one element to another. The European ITS Framework Architecture has used three main types of Data Flow: Functional, Physical and Terminator. The first and last of these are parts of the Functional Architecture – see below, and are defined in Annex 2 of reference 8(l). The Physical Data Flows are part of the Physical Architecture – see below, and are defined in reference 8(m).

(i) Data Store

This is used to represent a collection of data that can be accessed by one or more Functions. and thus forms part of the Functional Architecture – see below. It may be managed by a single Function, or open to access by several Functions. Its contents can be defined in terms of the identity of the data that it contains. The use of Data Stores within the European ITS Functional Architecture is defined in reference 8(m).

(j) Deployment

This is the process that is used to put something into use. For a Physical System it will be the process that enables it to be put into use in its actual physical location(s). In the case of an Architecture it will mean making it available for use in the development of its own Physical Systems – see later definition. Quite often an Architecture will include a “Deployment Study” which will look at ways in which the Architecture can be deployed in the real World. For a discussion on its use within the European ITS Framework Architecture see Chapter 5.

(k) Function

This is a part of the Functional Architecture, – see below, that does something, i.e. it performs a function. This function may be simple or it may be complex. It almost always involves the processing of data that is received by the Function to produce information, - see below, that the Function outputs. This information will be sent to other Functions, Terminators (see below) or Data Stores (see above). For a definition of their use within the European ITS Functional Architecture see Chapter 2 of reference 8(l).

(l) Functional Architecture

This Architecture provides the definition of the functionality that is needed to provide the ITS services and facilities that are defined in the User Needs – see later definition. It consists of Functions, Data Flows and Data Stores, each of which is defined elsewhere in this section. For further information see Chapter 5, where its use within the European ITS Framework Architecture is also discussed.

(m) Implementation Architecture

This name is given to a derivative of the Physical Architecture. The Implementation Architecture adds operational technical details to the elements described in the Sub-systems and Modules of the Physical Architecture. These details include descriptions of components, languages and communication protocols. Also the products that perform functions and interactions identified by the Physical and Functional Architectures are specified, and extra design details are provided to enable the creation of specific code. The Implementation Architecture is comprised of two main principal elements. The first is the specification of the hardware, operating systems, and communication protocols that are to be used in the System implementation. Secondly a System model is used to identify the specific (commercial) products that implement the Sub-systems and Modules defined in the Physical Architecture. Other elements include an interface model specifying exchange mechanisms for the Sub-system (and Module) interfaces and between the database management systems. Finally, it can also include an information model specifying the different data models.

(n) Information Architecture

This Architecture is used to show the structure of information within an Architecture. It focuses on the nature of the data, how it is manipulated and stored, and their relationships, e.g. using a Data Model. The manipulation and storage of the data within the Architecture will take place in Functions and Data Stores respectively. One of the uses of the Information Architecture can therefore be to establish the maximum storage capacity of the Data Stores. In conjunction with an Organisational Architecture – see below, it can also be used to establish how data is, or can be shared between organisations.

A separate Information Architecture has not been produced as part of the European ITS Framework Architecture. Instead it has been subsumed into the Functional Architecture.

(o) Information

Information is data that has been processed so that it has context and a meaning that is determined by the context. Thus it applies to data that has been processed and ready for sending to things that are outside of the System represented by the Architecture.

(p) ITS System

This is another name for a Physical System – see below. Strictly speaking the "System" is redundant as it is part of "ITS". However "ITS" has come to be used in the same way that "Transport Telematics" and other similar phrases have been used in the past. Therefore the word “System” has been retained. If it were shortened to “IT System”, that would probably have a different meaning, e.g. “Information Technology System”. The term “ITS System” is used in the Appendix to this Document.

(q) Logical Architecture

This name is used, in the US National ITS Architecture for example, to describe what in the European ITS Framework Architecture is called the Functional Architecture – see earlier definition and also Chapter 5.

(r) Organisation(al) Architecture

The name is used for a derivative of the Physical Architecture. It is used to show the organisations that will own, and/or operate, and/or maintain the Sub-systems and Modules in the Physical Architecture. This is very useful for highlighting the relationships between organisations that will have anything to do with the Architecture, and any conflicts that may arise. It can also be used with the Information Architecture – see above, to look at how data will have to be, or could be shard between organisations. In some instances, this Architecture is not required because the same organisation owns and operates all of the Sub-systems and Modules, and is also responsible for arranging any maintenance activity.

(s) Physical Architecture

This Architecture provides a definition (or set of definitions) of the elements that can be produced from the Functional Architecture (see above) in order that physical things (hardware and software) can be produced that will deliver the ITS services and facilities required by the User Needs – see later definition. There may in fact be several definitions of how the elements can be produced from a single Functional Architecture. Therefore the Architecture development may choose to be either proscriptive and provide the one and only acceptable definition, or provide a mechanism that enables different definitions to be produced. Each definition can be called a “Physical System” – see below.

It is also possible to produce a Physical Architecture that represents a sub-set of the original Architecture. In this case its own Physical Architecture can be produced. A discussion on the various forms of Architecture that can be produced will be found in Appendix 1 to this Document.

The team that produced the European ITS Framework Architecture chose to provide a mechanism that enables Physical Systems and/or Architectures to be produced. This is discussed in Chapter 5, and in reference 8(m).

(t) Physical System

This is a System, part of whose high level specification can be produced as part of the Physical Architecture – see above.

(u) Presentation Architecture

This name is used for a derivative of the Communications Architecture. It is used to highlight the need for common standards of presentation of information to users of the Sub-systems and Modules in the Physical Architecture.

Two examples of a Presentation Architecture are shown in Chapter 8 of the Communications Architecture Deliverable Document (D 3.3).

(v) Reference Model

A Conceptual Model (see above) becomes a Reference Model if it turns out to be a fundamental part of either the communication between the participants realising the System, or a basic construct during System realisation. A Reference Model usually exists at a very high level of abstraction, e.g. higher than the European ITS Functional Architecture. It will usually capture both the constructional relationship between the Functions, and the overall behaviour of any System that conforms to it, and hence is very rich in information. For further discussion on the use of Models in the ITS environment and examples of some European Models, see reference 8(o).

(w) Risk

A Risk is something that may happen and will pose a threat to the deployment of the Architecture or Systems produced from it. Usually there will be some assessment of the impact of the Risk and a definition of the mitigation strategy that could be deployed to minimise (mitigate) that impact. Its use within the European ITS Framework Architecture is discussed in Chapter 5.

(x) Terminator

This is the definition of how the System represented by the Architecture, expects the outside World to behave. It is in fact a rigorous definition of the functionality that the outside World is expected to provide for the System to operate and deliver the Services and facilities required by the users. Usually there are several Terminators, one for each part of the outside World. For the European ITS Framework Architecture there are twenty – see Chapter 4 of reference 8(l).

(y) User Needs

These are the statements that define what the users require the System to do. In other words they define the Services and facilities that are to be provided by the Architecture (or Physical Systems derived from it) and the constraints on their provision. For road based ITS the Services should cover all aspects of road travel. There may also be links to other modes, e.g. rail, sea and air, if multi-modal travel facilities are required by the users. The constraints may be called the Characteristics and define such things as the “rules” under which the Architecture (and its Physical Systems) must operate. They may also include constraints on the way in which users can access the Physical Systems, e.g. output must be in both audio and visual forms. The use of the User Needs within the European ITS Framework Architecture is discussed in Chapter 5 and defined in reference 8(k).

(z) User Service Requirements

This name is used in the US National ITS Architecture to describe what in the European ITS Framework Architecture is called the User Needs – see above. It describes what the users require the Architecture to do. This is expressed as definitions of the Services and facilities that are to be provided by the Architecture and the constraints on their provision. For more information about the US Architecture, see Chapter 7 of this Document.

4.4.3. Further Information

If any reader or user of the European ITS Framework Architecture is interested in the further development of a knowledge of System Architecture terminology and Architecture development, there are a large number of sources of information. Some of them are listed in the Bibliography section of the European ITS User Needs Deliverable Document (D 2.2) – see reference 8(k). Others may be found by consultation with libraries, and professional bodies such as the Institute of Electrical Engineers (UK) and the Institution of Electrical and Electronic Engineers (US). The choice of source will depend on the method of Architecture development that is to be followed, and thus other sources may need to be found. It is recommended that those with access to documentation produced by the ISO should consult with its “Glossary of Terms”.

5. Framework Architecture Development Process

5.1. Introduction

The purpose of this Chapter is to describe the general System Architecture development process and relate this to that used by the KAREN Project to develop the Framework Architecture. The process description is intended to help readers of the Documentation and users of the Framework Architecture better understand how each of the Documents relates to the other.

5.2. General Architecture development process description

The general process for System Architecture development is best illustrated by the diagram shown on the next page. It is valid for Architecture that are at Levels 1 to 3 (inclusive) as defined in reference 8(a). The process is described in the following paragraphs, and produces a number of Architecture components. These are identified by "boxes" on the Figure have been highlighted in **bold** in the following text.

The establishment of the **User Needs** is the most important part of the Architecture Development process, since it is the foundation upon which it is built. It involves getting input from current and future stakeholders about what services they would like the Systems derived from the Architecture to provide. The User Needs will cover not only functional requirements, but also many other requirements that the Systems must fulfil. Examples are maintainability, reliability and security. This part of the Architecture development process is best described in the CONVERGE Architecture Guidelines – see reference in the last Chapter.

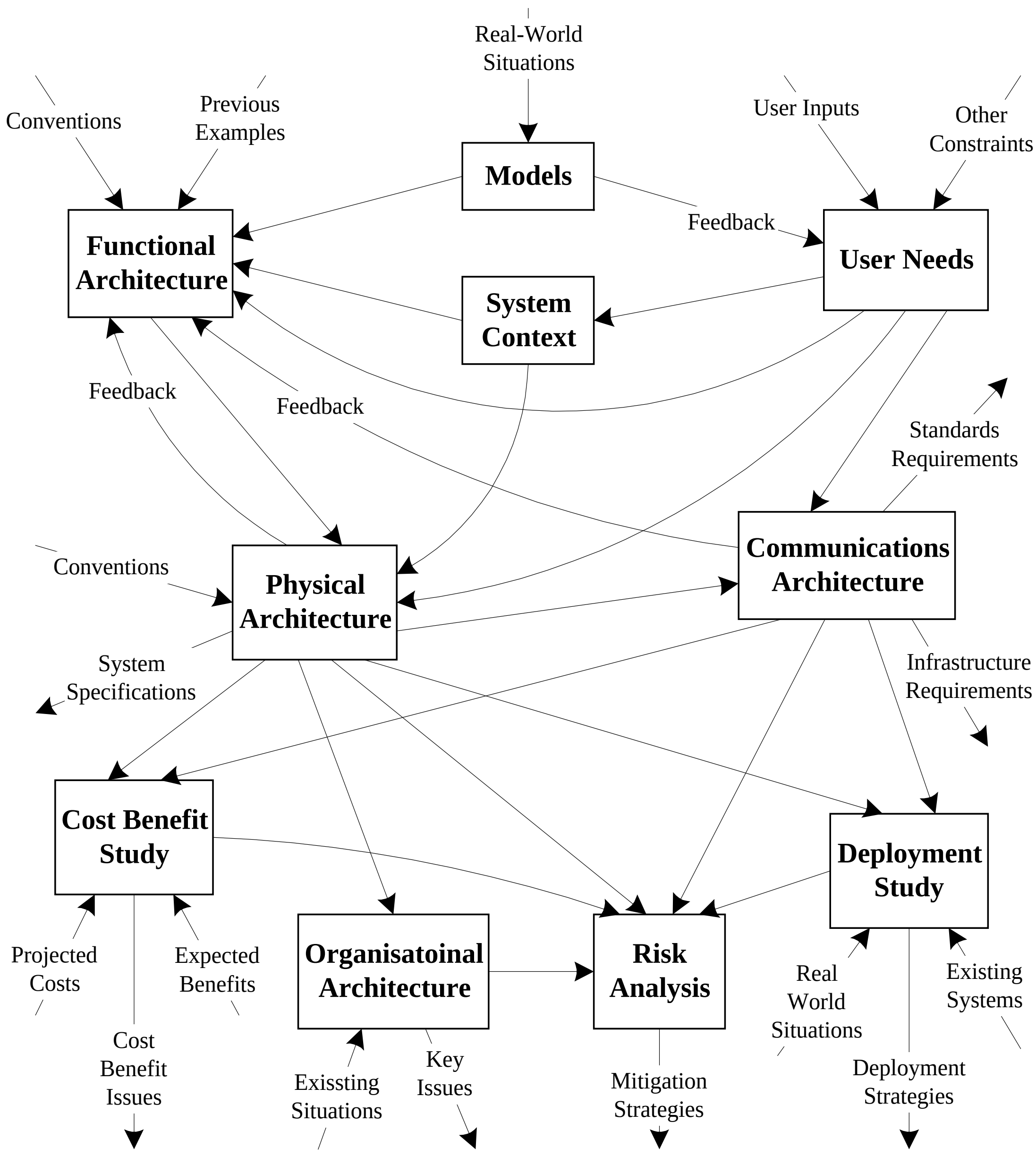
At this point **Models** should be developed. These represent "real World" situations and in some cases may represent scenarios of how the functionality that it is intended shall be provided by the Architecture can be used. There are three types of Models, comprising, Enterprise, Primary Process and Layered Reference. There use is discussed in reference 8(o).

The first step of the actual Architecture development is to develop the **System Context**. This defines how the outside World will relate to the Systems. This relationship is defined through the specifications for Terminators. These describe what data the Systems will expect the outside World to provide and what it must do with the data that it receives from the Systems. The Terminators are portrayed in the **Context Diagram**.

The Terminator descriptions and Context Diagram, together with the User Needs are then used to develop the **Functional Architecture**. This defines the functionality needed to fulfil the User Needs and interface with the outside World through the Terminators. It will also include a definition of the data that the System uses, as input, within itself and as output. This results in an Information Architecture, which can be produced separately from the Functional Architecture.

The next step in the Architecture development is to produce the **Physical Architecture**. This takes the functionality from the Functional Architecture and puts it into groups, called Sub-systems. There must be one Sub-system for each physical location that is to be used by the System. Up to six different generic physical locations should be used. These comprise, Central, Roadside, Vehicle, Personal Device, Freight Device and Kiosk. The Sub-systems may be divided into Modules when the functionality that they contain is complex and/or because it serves a variety of different User Needs. For each Sub-system and/or Module specifications are written based on the description of the functionality that they contain. These specification can then be used as the starting point from which actual Systems and/or their components can be produced. The Physical Architecture must also take account of any User Needs that have physical as opposed to functional requirements.

Figure 1 KAREN Framework Architecture Process Diagram



The **Communications Architecture** is developed from the Physical Architecture and defines the Communications Needs of the Systems. Again it may also include some requirements from the User Needs, where they relate to specific communications requirements. The outputs from the Communications Architecture will be the specification of the communications infrastructure that the Systems require, plus a definition of interfaces for which standards are needed.

Another output from the Physical Architecture is the **Organisational Architecture**. This takes the Sub-systems and allocates them to the Organisations that will own or use them. The reason for creating this Architecture is that it helps to define any issues that may arise because different Organisations own and/or use different parts of a System.

From the Physical and Communications Architectures the **Deployment Study** and **Cost Benefits Study** are produced. The Deployment Study shows how the Systems derived from the Architecture can be deployed. This must specify the time frames involved and how any existing Systems are to be accommodated. The Cost Benefit Study provides a prediction of the likely costs and benefits to be provided from the deployment of the Systems derived from the Architecture.

The **Risk Analysis** is produced from the Physical, Communications and Organisational Architectures, plus the Cost Benefit and Deployment Studies. It defines the Risks generated by all of these Architectures and Studies. Usually they will be categorised in some way according to severity and/or impact. Those in the most severe category will have mitigation strategies developed for them. These will define the actions necessary to reduce (mitigate) or transfer the impact of the Risks.

The lines marked “Feedback” in Figure 1 indicate where the work on a later phase of the Architecture development may cause revisions to the work in earlier Phases. These revisions may be in several forms, but typically will require changes such as the addition of Architecture components such as Functions, Data Stores, Systems, Sub-systems, Modules, plus Functional and Physical Data Flows.

The outputs from the development process appear from several of the components identified above. They can be summarised in the following table.

Table 1 Architecture Componenets and their external outputs

Architecture Component	Output	Description and future use
Physical Architecture	System Specifications	Used to develop Sub-systems and Modules that are used to produce a “real” System. They must be capable of being used in the development process that is considered most appropriate. Sub-systems and Modules may cover hardware, software, or a combination of the two.

Architecture Component	Output	Description and future use
Communications Architecture	Infrastructure Definition	Identifies the infrastructure(s) that will be needed to support the physical communications links required by the System.
Communications Architecture	Standards Definitions	Identifies any standards that need to be developed for interfaces between Sun-systems, or between Modules within Sub-systems
Risk Analysis	Migration Strategies	The steps that are required to minimise or transfer the identified Risks if and when they materialise. Usually confined to the most significant risks.
Deployment Study	Deployment Strategy	A description of how the System will be deployed. It should include migration strategies for existing Systems and interfaces to them.
Organisational Architecture	Key Issues	A description of any key issues relating to how the various organisations that own and/or use the System will work with each other. It may also include an analysis of how data is, or could be, shared amongst the organisations.
Cost Benefit Study	Cost Benefit Issues	Identification of any issues relating to the cost of System deployment and/or the benefits that it will provide. Care must be taken not to confuse these with the Risk Mitigation Strategies described above.

Several more comprehensive descriptions of Architecture development are available from a variety of sources. A list of some of the documents, papers and books are listed in the Bibliography section of the European ITS User Needs Document – see reference in the last Chapter.

5.3. Use of the General Architecture development process by the KAREN Project

The development of the European ITS Framework Architecture by the KAREN Project has followed much of the Architecture development process that has been described in the previous section. The departures that have been made are mainly a result of the "high" level at which the Framework Architecture is defined. The departures from the Architecture components are as follows.

- Information Architecture: not developed separately and included in the Functional Architecture.
- Organisational Architecture: not developed at all. There is no single (or generic) structure for European organisations that are involved in the ownership and/or use of ITS. Thus it was not considered appropriate to develop this Architecture.
- Risk Analysis: this was carried out by the RAID Project. Its results have been documented separately, but have been included on the European ITS Framework Architecture CD-ROM issued by the European Commission.

The departures that the KAREN Project has made from the list of outputs shown in Table 1, in the production of the European ITS Framework Architecture are identified in the following Table.

Table 2 Use of Architecture Componenet Outputs in the European ITS Framework Architecture

Architecture Component	Output	Description and future use
Physical Architecture	System Specifications	A sample System Specification has only been produced for one "example System".
Communications Architecture	Infrastructure Definition	An analysis of some of the "example Systems" in the Physical Architecture has been provided.
Communications Architecture	Standards Definitions	Potential for standards identified by Work Package 4 of the KAREN Project and shown in reference 8(q).
Risk Analysis	Migration Strategies	These are shown in the results produced by the RAID Project.
Deployment Study	Deployment Strategy	Output provided by Work Package 4 of the KAREN Project – see reference 8(p).

Architecture Component	Output	Description and future use
Organisational Architecture	Key Issues	Not appropriate for the European ITS Framework Architecture, as organisational structures vary between European Nations, States, Districts, Provinces, Cities, Counties, etc.
Cost Benefit Study	Cost Benefit Issues	See Executive Summary of D 3.4 in Chapter 6 of this Document.

The European ITS Framework Architecture versions of the main components of the Architecture development process are described in the Documents listed in section 1.2. A copy of the Executive Summary for each Document is provided in Chapter 6.

5.4. Using the Architecture Development Process

If a completely new Architecture is required, that may (or may not) be based on the European ITS Framework Architecture, then the process described in section 5.3 should be followed. It has been used by the KAREN Project and by the teams responsible for the development of the US National ITS Architecture – see Chapter 7 of this Document.

Anyone wishing to develop their own Physical System or Architecture from the European ITS Framework Architecture should follow the steps and processes that are described in the Deployment Strategy produced by Work Package 4 within the KAREN Project - see reference 8(p). However this does not describe the development of an actual Physical System or Architecture. Their development is described in European ITS Physical Architecture Deliverable Document –see reference 8(m). Chapter 4 of that Document covers Physical System development and Chapter 5 covers Architecture development. Both of them follow what is described in section 5.3, but have some detail differences.

6. Framework Architecture Documentation

6.1. Introduction

This chapter covers the structure of the documentation that describes the European ITS Framework Architecture. This documentation that is publicly available has been divided into six parts as follows:

D3.1 - Functional Architecture
D3.2 - Physical Architecture
D3.3 - Communication Architecture
D3.4 - Cost Benefit Study Report
D3.6 - Overview (this Document)
D3.7 – Models of Intelligent Transport Systems

The "missing " Document (D 3.5) was produced for internal (KAREN Project) use only. However its contents have been included in the European ITS Deployment Study (D 4.2) – see reference 8(p). The remaining sections of this Chapter provide overviews of each of the other Deliverable Document. Those Documents that have Annexes as separate Documents will have the Executive Summaries of the Annexes at the end of the Main Document.

6.2. D3.1 - Functional Architecture

This document provides a description of the Functional Architecture which forms part of the European ITS Framework Architecture. A Functional Architecture defines and describes what functionality needs to be included in a System that can fulfil the requirements of the European ITS Framework Architecture User Needs. These User Needs are provided in a separate document produced by another part of the KAREN Project.

This document describes the Functional Architecture in some detail and also covers the methodology used for its development. It shows how the Architecture links to the outside World through terminators and how it has been divided into Functional Areas. The way in which these have been divided into Functions is also included, together with diagrams for all the Areas. These diagrams (called Data Flow Diagrams) show how the Functions relate to each other, to Data Stores and to the terminators through the Data Flows.

Details of the Information Architecture are provided through the description of the Data Flows and Data Stores. These and the descriptions of the Functions are contained in three separate Annexes to the document. The reasons for using these three Annexes are to simplify the study of the Architecture and to avoid producing one large document. An overview of what is in each Functional Area can be obtained from the main part of the document before the detail of the Functions, Data Flows and Data Stores is studied using each of the three Annexes.

This document forms the starting point for the description of the European ITS Framework Architecture. This description will be expanded by further documents that will provide descriptions of the other parts of the Architecture, such as the Physical Architecture and Deployment Study.

6.3. D3.2 - Physical Architecture

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.

6.4. D3.3 - Communication Architecture

This Document provides a description of the Communication Architecture which forms part of the European ITS Framework Architecture. A Communication Architecture defines and describes what kind of communication links needs to be used in a System in order to support its physical data flows. These physical data flows are presented in the European ITS Physical

Architecture provided in a separate document produced by another part of the KAREN Project.

The Communication Architecture is described in some detail by this Document, which also covers the methodology used for its development. The basic concepts of communications are discussed, including general points about the characteristics of links within Systems between different physical locations, and between Systems and the parts of the outside World with which they interface.

The communications requirements of several of the "example Systems" in the European ITS Physical Architecture are described and analysed. Conclusions about the best type of communications to use are also provided.

A brief description of current communications technologies and an overview of the OSI model are included in Annex 1. This is provided as a separate Document for ease of use and reference. The actual data used for the analysis of the "example Systems" featured in the main part of the Document is included in Annex 2.

6.5. D3.4 - Cost Benefit Report

Introduction and Objectives

In order to effectively design and implement any type of framework architecture, it is essential that the costs and benefits involved are clearly understood. In this way, the architecture can be focused on areas where the best value for money can be obtained.

This document (European ITS Framework Architecture Deliverable D3.4) aims to address this need by presenting a Cost Benefit Study into the development and adoption of a framework architecture for Intelligent Transport Systems (ITS).

The objective of this task is to look at the benefits likely to be obtained from the deployment of the Framework Architecture (FA). In particular, it considers the results obtained from European Telematics projects where the integration of different applications in a common infrastructure has demonstrated benefits.

Cost-benefit aspects considered

The costs and benefits considered in this study are those which are additional to the costs and benefits which would occur in the case of ITS implementation without a common Framework Architecture.

This distinction is however difficult to clarify, since results from deployments where there has been a common architecture cannot be directly compared with those relating to deployments without a common architecture, as schemes are affected by so many other factors.

Actors

The actors involved in ITS architecture and deployment cover a wide spectrum, comprising:

- ITS system manufacturers;

- operators providing ITS services (infrastructure and public transport operators and other service providers, such as broadcasters, the electronic and print media, motoring organisations, etc);
- private users (individual members of the public);
- commercial users (businesses which stand to benefit from ITS);
- governments and other public authorities; and
- standardisation bodies.

Actors may be users or providers of ITS, or in some cases both. In the case of standardisation bodies, they are neither ITS users nor providers.

Costs

Regarding costs, the deployment of the Framework Architecture requires investment, such as:

- the investment that has already been dedicated to the European ITS Architecture (in the 4^{th} Framework Programme European projects and in particular KAREN); and
- the investment that will be necessary at the national, or even at the regional level for establishing an EU compliant national or regional FA.

The costs of an ITS Framework Architecture can be classified in two ways:

- the types of costs; and
- by whom they are incurred (these are the actors detailed above).

The types of cost can be classified in terms of three broad groups as follows:

- Architecture development costs (identification of system architecture promoters, planning phase, tendering process, technical development and reviewing the results);
- Costs of exploiting the "paper" results (dissemination/awareness/promotion, training/ education, political acceptance; defining funding access and maintenance of paper results);
- Implementation costs (finding funding resources, migration of existing systems, procurement and tendering processes, building and maintaining ITS state-of-the-art, delivery and acceptance of systems, training on the use of new systems and maintenance of systems).

Benefits

The benefits of deploying a common framework architecture can be considered at different levels:

- benefits obtained through specific ITS applications/services that will use a common framework architecture. In general, these benefits will be obtained by a reduction in implementation costs;
- benefits obtained as a result of the Framework Architecture offering interoperable solutions between different services or different geographical areas;

- benefits obtained as a result of the ITS Framework Architecture facilitating ITS deployment in Europe. This will increase the number of ITS services and reduce implementation times, with each of these services bringing its own costs and benefits.

These benefits can take the following forms:

- reduced barriers to market entry for ITS manufacturers and service providers, leading to increased competition and reducing costs;
- reduced costs for developing national or local architectures (as they can use KAREN as a base);
- reduced costs for developing ITS systems themselves;
- reduced costs for purchasing the equipment;
- reduced operating and maintenance costs;
- reduced costs to users for services (where producers or operators pass on savings to users – although in this case, the financial benefits to operators and suppliers listed above will be reduced);
- reduced risk in developing and implementing systems;
- better knowledge of demand/transport patterns;
- enhanced communication between actors (at a national and international level);
- improved management of research and development and standardisation at a national level;
- interoperability of services (i.e. technical, functional and contractual interoperability);
- more seamless, co-ordinated and consistent services, leading to enhanced quality for the end user;
- increased multimodality; and
- increased market size resulting from lower costs and higher service standards, leading to increased employment.

These benefits are interrelated in that some of them are not final objectives as such, but are prerequisites for achieving other benefits, for example, interoperability leads to more co-ordinated, consistent and seamless services.

Benefits can also be clustered in terms of the following groups:

- technical benefits;
- social benefits;
- legal/institutional benefits;
- financial benefits.

The benefits emanating from a common Framework Architecture are indirect in that they are effectively the third stage of the following sequence:

1. Production of an accepted standard or result (such as the KAREN Architecture).

2. Application of the standard/result by ITS implementers (in the development of local architectures or in system deployment).
3. Benefits resulting from this application, in terms of reduced costs and time for development and implementation and reduced operation and maintenance costs.

Architecture deployment costs in selected EU Member States

National Framework Architecture projects are currently underway in a number of EU Member States. These can be summarised in the following table:

National Framework Architecture Projects in the EU

Country:	Finland	France	Italy	The Netherlands	Sweden
Name of project	TelemArk	ACTIF	ARTISTA	Koepelarchitectuur	OPTIS
Scope	Passenger transport telematics for all modes (road, rail, air, water).	Road, rail and inland waterway modes, plus their interfaces with air and sea transport.	All inland transport modes	Road, rail and inland waterway modes, plus their interfaces with air and sea transport.	Road
Aims/Approach	Adopts a bottom-up approach through workshops involving actors	To improve communication between actors, elaborate prospective visions and act as a tool to address ITS architecture issues. Approach involves case studies	Development of a national architecture for ITS that serves as a reference platform for the implementation of real intermodal services nation-wide. Involves the design of a national intermodal demonstrator	To demonstrate the coherence of various ITS elements, to demonstrate coherence with the European FA, to serve as a basis for further architecture development and to serve as a means to reach agreement on further ITS implementation	Focuses on services and aims to deliver dynamic information
Cost (euros)	0.6m (1998-2000)	2m (1999-2001)	3m (2000-2001)	0.2m for Koepelarchitectuur (1999-2000) plus 2m for sub-architectures	0.1m

Examples of "success stories" from European ITS projects and deployment programmes

The **QUARTET** and **QUARTET Plus** EU demonstration projects involved local authorities, transport operators and the others (including vehicle manufacturers, providers of telecommunications, IT and ITS systems, broadcasters and research organisations) in four cities, expanding to six in the case of QUARTET Plus.

The aim of the QUARTET project was to demonstrate a pilot implementation of integrated ITS applications. The project demonstrated that integration of advanced transport telematics on a large scale is both technically feasible and socially desirable.

QUARTET Plus had the objective of demonstrating a European platform for launching transport informatics in urban areas on an open market by means of comparative evaluations and by operating an information infrastructure, multimodal information and public transport management. A key part of the project with relevance to framework architectures is the workpackage dedicated to the demonstration of IRTE strategies, architectures and tools.

QUARTET and QUARTET Plus implemented a communication and physical architecture which shows a relatively high level of integration in the data exchange facilities between “systems” operating in urban area. A co-operative control scheme was introduced in different sites and key results were demonstrated by means of extensive field trials and surveys.

The high level of integration demonstrated relevant benefits by sharing infrastructure. For example, it is estimated that the costs of purchasing and installing roadside outstations in the Turin application (the 5T system) were recovered by savings on transmission charges over a payback period of one year. Cost benefit analyses also showed relevant benefit for the operators, e.g. efficiency savings have allowed the Turin public transport operator to remove one tram from the service as a result of improved regularity and efficiency.

The **ROMANSE** project in Southampton aimed to realise a vision of an information-rich society based on integrated ITS and to improve travelling conditions in the city by reducing uncertainties and providing the information required for travellers to make more informed travel choices. The project involved the implementation and testing of a range of telematic applications.

The architecture employed involved the linking of these applications to a strategic information system which is a fully integrated and locationally referenced database and can be viewed in real time on a map-based display.

The cost-benefit analysis of the applications was based on the average amount of time that each user would have to save in order to justify the investment. Examples of the time savings necessary to justify the implementations are between half a minute and one minute for the parking guidance system and half a minute to two minutes for the “STOPWATCH” real-time bus information display at bus stops. These benefits did not include benefits to operators (e.g. increased patronage for the bus company). Before and after surveys have also shown a significant increase in the proportion of users rating traffic information in the city as good,

Whilst the monitoring and evaluation has proved to be difficult, awareness of the systems was found to be very high amongst the travelling public and the levels of time savings required to justify the investment and operation are relatively small. Furthermore, the performance of the system as a whole has been analysed as being better than that of any individual product.

The **ENTERPRICE** project had the aim of improving multimodal traffic information. Its core activity was the establishment of Mobility and Traffic Information Centres (“MOTICs”) using an open standard architecture in the Rhine Delta area, Switzerland and Hessen.

Whilst the demonstrations at these sites had different priorities, the basic functionalities of the MOTICs were the same, in terms of collecting and processing traffic data. The architecture of the system was designed in order to fulfil the requirements of the various conditions in each European country.

The research costs for MOTIC within ENTERPRICE were in the order of 200 000 euro and the benefits included services to travellers and fleet operators, a contribution to the sustainable and safe mobility of passengers and goods over a wide area, services to improve multi-modal travel, improved information, a basis for higher safety and comfort at lower cost, and a basis for the development of new products and new markets.

In the case of TIC Nederland, the Dutch national traffic information centre which was established in 1998 as a result of activities within ENTERPRICE, the adoption of architecture standards have had the effects of making it easier for commercial service providers to enter the marketplace and of promoting competition, which has led to improved services to the end user. It has also allowed data exchange between other TICs (in particular, the North Rhine Westphalia TIC in Cologne) by means of Datex.

Electronic data exchange on an international level using the **Datex** format is currently being implemented in traffic centres around Europe and provides these traffic centres with language-independent information from neighbouring regions and countries.

In effect, the Datex architecture is defined as a number of interoperable client and supplier systems. The Datex Memorandum of Understanding (MoU) comprises the latest agreed version of the Datex specifications and the Datex documentation is currently being considered for standardisation by the CEN (European Standards Committee).

The benefits of adopting a communication standard at the Datex level include having a common understanding of the information to be exchanged and its structure, including language independence, and achieving interoperability at functional and technical levels. A further key benefit is a reduction in development costs compared to the situation where centres apply their own ‘standards’ on a bilateral basis.

The implementation of **RDS-TMC** in Europe is another area where a common communication architecture exists. The EU has focused on the deployment of RDS-TMC services as part of the TEN-T implementation programme and its implementation on an international level with a choice of broadcast languages is being planned. The aim is for a continuous, compatible and interoperable service with a key benefit being that the user can use the same certified receiver in any country in Europe to receive a standard quality in the language of his/her choice.

Implementation costs over the past three years have been around 10 million euro, and costs to users of RDS-TMC include the requirement to buy a terminal (or to purchase a car fitted with one).

Electronic Fee Collection (EFC) on motorways is an area where common architectures are being developed. A key requirement of such systems is interoperability, both between different motorway operators within a country and also between countries, to remove the need for motorists to have more than one tag fitted to their vehicle. Currently, interoperability exists on an national basis, such as in France (TIS - Télépéage Inter-Société) and Italy (Telepass). However, current EFC systems are not interoperable between Member States.

The CARDME (Concerted Action for Research on Demand Management in Europe) project involved the development of a non-technical framework architecture for EFC. Its report into the migration of existing systems looked at the costs and benefits related to the introduction of two new tolling services for users: a stop & pay service and an interoperable EFC service. For both services, the main cost is on the set-up of the back office procedures between the different operators for the transfer of claims and settlements between the participating operators. The other main costs are related to information provision to users about the existence of the new service, i.e. signing, brochures and contracts.

The user benefits will be in the time saving by accessing a common level of service across Europe, but will require the purchase and fitting of an electronic tag.

Although operators will face the principal costs at the time of development and introduction of new interoperable EFC services, they will gain longer term benefits such as a reduced need to invest in added capacity at toll plazas to accommodate international traffic, because of the increased capacity of toll plazas. Also, if EFC becomes more interoperability, it also becomes more attractive to users, thereby reducing the number of manual staffed barriers required.

Operators are also providing a value-added service to the motorist in implementing interoperable EFC, especially for international motorists, who will find the motorway both easier to use and will save time queuing, dealing with foreign money or making credit card transactions. Operators could recuperate their costs through modest increases in the toll rate.

Examples of “success stories” from outside the ITS world

GSM (Global System for Mobile communications) is a standard service architecture in the mobile phone sector. It is an architecture which has led to major benefits, in that it allows interoperability of mobile phones on a Europe-wide basis and beyond.

Although there are experiences to be learned from the development of this architecture, direct comparison to the ITS industry is difficult as GSM developed in a specific context where financial issues were predominant. In addition, the mobile phone market is far larger than that for ITS equipment.

The development of a GSM standard has however made a significant contribution to the growth in mobile phone use and competition in this sector is now intense, with prices falling. Service providers now offer a wide range of tariff structures and value-added services to differentiate their products.

The costs of this work are difficult to estimate, and comprise largely of inter-operator co-operation work. Operators do however consider the costs to be worthwhile in terms of the benefits, notably the 200 million GSM users (out of a total of 436 million mobile phones world-wide). Furthermore, growth in GSM ownership has been twice as fast as predicted, with over 40% of the European population owning a mobile phone in 1999 figure, compared to 14% two years previously. This level of success would not have been possible without the standardisation work.

Key benefits transferable to ITS include price pressure due to multi-vendor compatibility at a system level and the fact that customers are more willing to invest as they know that they can develop and realise their investment without being locked into a single supplier.

The **Internet** is another example of a standard architecture outside the ITS domain. This widespread information infrastructure was principally developed in the USA and had a lead time of some 20 years, although its use has exploded in the past two to three years (since the late 1990s).

Its framework architecture is therefore now a World-wide standard, connecting computers made by a variety of different manufacturers and supporting services such as e-mail and the World Wide Web, which have their own sub-architectures. These applications are Internet protocols and use the Internet as a platform.

The Internet is one of the most successful examples of the benefits of sustained investment and commitment to research and development of information infrastructure. Costs were borne initially by the academic world and boosted by a ten year $200 million investment by the US National Science Foundation, but in more recent years, the growth in Internet users has been such that costs can generally be covered commercially.

In ITS, architecture development has been less based in the academic world and more in the industrial and public sectors although, like the Internet, commercial players are coming increasingly to the fore. The other difference is that the formative years of the Internet took place in a non-commercial environment in the USA, where public funding was conditional upon the system being used only for government and research uses. When the service opened up to commercial use, the same basic architecture was used, and was exported from the US world-wide. Although the Internet is now hugely successful, it attracted little interest during its formative years and was effectively producer-led. ITS, on the other hand, is more consumer led and commercially oriented. However, public funding is still necessary in order to develop the architecture as the large number of private players in a competitive environment make this an unfeasible proposition within the commercial sector.

The second major difference is the rate of growth. The benefits of the Internet are clear to most people, whilst those of ITS are less so, and are more of interest to public authorities and road operators than the general public, most of whom are unfamiliar with the concept. So while ITS services are undoubtedly on the increase and their value is being recognised, comparisons with the exponential growth of the use of Internet applications are impossible.

Approaches to ITS Architecture Outside Europe

The **US National Architecture project** for the Federal Department of Transportation represented the first large scale effort in the United States to document the costs and benefits of equipment associated with ITS deployment, and included the following three elements:

- Cost Analysis - this develops a high level cost estimate of implementation expenditures and acts as a costing tool for implementers.
- Performance and Benefits Study - this assesses the technical performance of the US National Architecture on a number of system-level and operational-level criteria. The analysis separates two types of benefits: the benefits of the architecture as a whole and the benefits of particular ITS products

- ITS Programme Assessment/Evaluation - this aims to ensure that the US National ITS Program is effective in meeting the DoT's transportation goals, with emphasis being placed on tracking both programme outputs and programme outcomes. Programme outputs are results-oriented measures that track the progress of a programme for which the benefits are easily understood by the end-user.

In **Japan**, five government bodies jointly finalised the System Architecture for ITS in Japan in 1999. A cost-benefit study of the architecture is being conducted and results are expected at the end of 2000.

Conclusions

This report outlines the costs and benefits of developing, maintaining and implementing an ITS framework architecture. Costs and benefits in this report have been assessed and discussed in a qualitative way rather than a quantitative one, as the lack of experience in an ITS framework architecture means that concrete figures are simply not available at the present time.

Furthermore, these benefits are difficult to measure (due to the lack of an effective "do nothing" or "control" scenario as a benchmark with which to measure the results of implementations), they cannot often quantified in financial terms, they are spread across a very large number of beneficiaries, and it is also difficult to ascertain to what extent the benefits of ITS implementations are enhanced by the adoption of a common architecture or integrated system.

The purpose of this report has therefore been to provide an analysis of the issues and to provide ITS implementers and funders at European, national and regional/local levels with suitable reasons for adopting an approach based on a standard framework. These have been supported by a number of examples on a smaller scale where a common architecture or the integration of separate systems have brought benefits.

Therefore, while the benefits of the KAREN Framework Architecture cannot be quantified (or at least at the present time, when national architecture projects based on KAREN are still underway), KAREN can provide (and indeed is currently providing) benefits to the developers of national, regional and service-oriented architectures by providing a foundation for this architecture, thus speeding implementation and reducing costs.

These architectures will then provide the bulk of the benefits to the end user, whether private or commercial. In addition, the KAREN FA can provide some benefits to the end user in terms of creating a framework in which different applications and different services in Europe can inter-operate. This interoperability not only improves services to the end user but also helps to drive down costs in the longer term for users, operators, manufacturers, implementers and funders alike. Another benefit of the KAREN FA is that the common framework can reduce entry barriers to the market, including barriers between markets in EU Member States, thus contributing to the European Single Market and increasing the competitiveness of the European ITS industry.

6.6. D3.7 - Models of Intelligent Transport Systems

This Document provides an introduction to ITS Architectures and Models, and describes the relationship of the European ITS Framework Architecture to National, Local, Service and System Architectures.

A variety of Models have been developed which present some of the possible ITS services from a variety of viewpoints. Each model is described with an Overview, Scope, Description and short list of the main European ITS User Needs that it satisfies. These can be used to produce an Architecture or System that implements the Model.

Reference Models, e.g. 'Tower Models' capture both the constructional relationship between the functions, and the overall behaviour of any system that conforms to them. A top-level Traffic Control System Pyramid Model is described, followed by the SATIN tower models for Inter-Urban, Urban and In-Vehicle systems, as well as a Traffic Control Cycle Model.

Enterprise Models show the structure of the relationships that exist between organisations, persons, services and/or functions, and models are described for the Transport and Traffic Market, Freight and Fleet Management, Public Transport, Automatic Fee Collection and Institutional and Legal issues respectively.

Primary Process Models describe the way in which a process takes place. Models have been created to describe an ITS and its Environment, the Traveller's Progress, the Vehicle's Progress, the Information Creation Chain, the Information Provision Chain and Emergency Services.

ITS development models have been included. These show how Traffic Control Policy can be created, an ITS can be created, and how an ITS can be improved.

The final chapter identifies five different type of user of these models, namely Decision Makers, End Users, Project Engineers, Infrastructure Operators and Service Providers, and the models that will be of relevance to them.

7. Comparison of the European ITS Framework Architecture with other ITS Architectures

7.1. Introduction

The purpose of this Chapter is to provide a high level comparison between the European ITS Framework Architecture and other ITS Architectures. The main comparison is with the US National ITS Architecture, as its methodology is very similar to that used by the KAREN Project to create the European ITS Framework Architecture. Other Architectures are also considered. These comprise those developed at the international level by the ISO System Architecture Working Group and those developed in Australia. This Chapter is not intended to be critical of any Architecture, but to provide a view of their relationship to the European ITS Framework Architecture.

7.2. US National ITS Architecture

The US National ITS Architecture owes its existence to the Intermodal Surface Transport Efficiency Act (ISTEA) passed by the US Congress in late 1992. From this the US DOT, through the Federal Highway Administration (FHWA) initiated a programme to produce a system architecture for road based transport within the US. The first part of the programme produced the User Service Requirements (USR's). These describe the services that the Architecture must support. The rest of the programme was divided into the following two Phases:

Phase 1 (October 1993 – October 1994): preparation of an initial Architecture by four separate teams working in parallel and in competition for selection to Phase 2.

Phase 2 (February 1995 – June 1996): preparation of the final version of the Architecture by one team made up of two participants selected from the four that participated in Phase 1.

In June 1996, the first definitive version of the US National ITS Architecture was released for use. It was available as a set of documents in both electronic and paper form. At that time, the US DOT contracted the Phase 2 contractors (at a reduced level of effort) to continue maintaining and supporting the US National ITS Architecture, and this work is still in progress. It consists of the following activities:

1. Maintenance of the Architecture. This involves correction of faults and implementing new usability services to the electronic documentation. Most recently this has included development of a software tool ("Turbo Architecture") to assist in the development of deployment-oriented regional ITS architectures based on the US National ITS Architecture.
2. Architecture Enhancements. These have been produced as a result of the creation of new User Services. The first of these was for the Highway-Rail Intersection (HRI) and the Archive Data Management services. Currently attention is being focused on a broad suite of maintenance and rural oriented user services.

3. Delivery of Training Courses. The objective of these courses has been to educate US State officials and private industry in the deployment and use of the Architecture. Recently the US DOT Architecture Team has begun a program to work in short focused engagements with ITS stakeholders in key regions to partially develop regional ITS architectures based on the US National ITS Architecture, using *Turbo Architecture*.

4. Provide support for Architecture use. Aside from ITS deployment, the main use of the Architecture has been to provide the foundation for the creation of ITS standards. These have been based on recommendations from the Architecture development, and in the main cover communications between ITS applications at various locations - central, roadside, vehicle, etc.

Since its first release in June 1996, the Architecture has gone through two major upgrade cycles, so that now (early 2000) it is at Version 3. It has also been ported onto a CD-ROM and is available on-line from a Web Site "http://www.iteris.com/itsarch/". These two distribution channels now provide the only real sensible access to the US National ITS Architecture.

7.3. Comparison between the European and US National ITS Architectures

7.3.1. Introduction

This section provides the actual comparison between the European ITS Framework Architecture and the US National ITS Architecture. The comparison is divided into sections that correspond to the main parts of the two Architectures.

7.3.2. User Requirements

The user requirements for both the US National ITS Architecture and for European ITS Framework Architecture were defined before the main Architecture creation work. In the KAREN Project, this work was done in Work Package 2 (WP2) and were documented separately from the Architecture itself. The user requirements define the services that the Architectures will provide and support. Their structure is basically the same, related requirements being grouped together. In the US Architecture the requirements are in "Bundles" whilst in KAREN they are in "Groups". This is shown in the table below.

Table 3 US Service Bundles and KAREN User Need Groups

US User Service Bundles		KAREN User Needs Groups	
No.	**Title**	**No.**	**Title**
1	Travel And Traffic Management	1	General
2	Public Transportation Management	2	Management Activities
3	Electronic Payment	3	Policing/Enforcing
4	Commercial Vehicle Operations	4	Financial Transactions

US User Service Bundles	
No.	**Title**
5	Emergency Management
6	Advanced Vehicle Safety Systems
7	Information Management

KAREN User Needs Groups	
No.	**Title**
5	Emergency Services
6	Travel Information
7	Traffic Management
8	In-Vehicle Systems
9	Freight and Fleet Operations
10	Public Transport

As will be seen from the table, there is a limited correspondence in the names used by the two Architectures for the grouping of their user requirements. The main difference is that the European requirements have three extra groups (Numbers 1 to 3), although this may decrease to two with planned changes to the US National ITS Architecture.

The actual format of the two sets of user requirements is also similar. An example from each Architecture is shown below.

<u>US National ITS Architecture</u>

1.6.2.2 Traffic Surveillance shall include a Data Collect function to provide the capability to collect data that are needed for determining traffic flow and prediction.

<u>European ITS Framework Architecture</u>

7.1.1.1 The system shall be able to monitor sections of the road network to provide the current traffic conditions (e.g. flows, occupancies, speed and travel times etc.) as real time data.

Any differences between the two sets of user requirements will be due to a number of factors. These include such things as differences in transport policy, legislative procedures, operating practices and language. Also, the use and meaning of terminology varies considerably. More importantly, both sets of user service requirements are architecture neutral and technology neutral.

7.3.3. Context

Each Architecture has a Context Diagram, showing the links between with the outside World in which they each exist. In both cases the outside World is portrayed using “Terminators”, for which definitions are provided. The US National ITS Architecture has 60 “Terminators” whilst the European ITS Framework Architecture has 20. This large difference is because the European Architecture has chosen to use “generic Terminators”, each of which may consist of two or more “actors”. When these are taken into account, the European Architecture has 55 “Terminators”.

7.3.4. Functionality

Both Architectures have chosen to develop the functionality that they contain using the Process Oriented methodology. For the European ITS Framework Architecture the reasons for this choice are given in Chapter 2 of the Functional Architecture Document (D 3.1). In both Architectures the functional analysis is provided through a "top-down" approach, so that users can see as much detail as they want from that which is provided by the Architectures. The main feature of this approach is that it is easy to understand, and can be appreciated by those who are not software or system specialists.

However there are two obvious differences in the way that the two Architectures describe their functionality that they include. The first is that in the US National ITS Architecture the description is provided by the "Logical Architecture", whilst in the European ITS Framework Architecture it is provided by the "Functional Architecture". Although different names have been used, their content is identical. The second main difference is that in the US National ITS Architecture, the functionality is divided into "Processes" whilst in the European ITS Framwork Architecture it is divided into "Functions". The formats and content of the "Processes" and "Functions" are similar and the links between both are shown in Data Flow Diagrams (DFD's).

At the highest level, both Architectures divide the functionality that they contain into Functional Areas. The table below shows the numbers and names of these Areas in the two Architectures. Although the numbers are different, there is a good degree of correspondence in the names of the Areas in the two Architectures.

Table 4 Functional Areas in the US and European Architectures

US Logical Architecture		European Functional Architecture	
No.	**Name**	**No.**	**Name**
1	Manage Traffic	1	Provide Electronic Payment Facilities
2	Manage Commercial Vehicles	2	Provide Safety and Emergency Facilities
3	Provide Vehicle Monitoring and Control	3	Manage Traffic
4	Manage Transit	4	Manage Public Transport Operations
5	Manage Emergency Services	5	Provide Advanced Driver Assistance Systems
6	Provide Driver and Traveller Services	6	Provide Traveller Journey Assistance
7	Provide Electronic Payment Services	7	Provide Support for Law Enforcement
8	Manage Archived Data	8	Manage Freight and Fleet Operations

Within each of the Areas, the functionality is divided into "Processes" and "Functions" as described above. In the US National ITS Architecture there are 482 "Processes" and in the

KAREN Architecture there are 253 “Functions”. Aside from any differences in the scope of the user requirements, the main reason for this difference is the US National ITS Architecture describes its functionality in a much greater level of detail.

Both the “Logical” and “Functional” Architectures include descriptions of the “Stores” that hold data and the “Data Flows” that link “Processes” or “Functions” with each other, with “terminators” and with “Stores”. In the US “Logical” Architecture there are over 2500 “Data Flows”, whilst in the KAREN “Functional” Architecture there are just over 1100. This difference is again a reflection of the difference in functional detail highlighted in the previous paragraph, and also the differing needs of the US and European Communications Architectures – see later section. The US “Data Flows” are also designed to support the much more intensive US standards creation activity that has been going on since the first release of the Architecture in June 1996.

7.3.5. Physical Architectures

The purpose of the Physical Architecture part of the US National ITS Architecture is to provide something that can actually **be** used for ITS deployment. It also provides the source of some of the data used in the US Communications Architecture. The Physical Architecture is divided into 19 Sub-systems covering four classes. These four classes comprise Centre, Roadside, Traveller and Vehicle Sub-systems. The building blocks of the sub-systems are Equipment Packages which group like “Processes” from the “Logical” Architecture of a sub-system together into a package that can be implemented. The grouping also takes into account the particular user services that a specific instance of a Sub-system may need to support and thus the need to accommodate various levels of functionality.

In addition, Market Packages are provided to give a deployment oriented perspective to the US National ITS Architecture. They are tailored to fit - separately or in combination – example real world transport problems and needs. Market packages collect together one or more specific Sub-systems with specific Equipment Package capabilities that must work together to deliver a given transport service. In developing regional ITS architectures based on the National ITS Architecture, the Market Packages generally must be customized for specific applications.

The European ITS Physical Architecture is designed for a different purpose. This is to act as the starting point for the creation of other lower level Architectures and Physical Systems. These may be for implementation at National or Local level, or by ITS manufacturers. The European ITS Architecture itself contains what are called “example Systems”. These show how the lower level Architectures and Physical Systems can be created. In the European ITS Physical Architecture Document (D 3.2), detailed guidance is provided to enable users to create their own “example Systems”. These should be more closely aligned to the particular needs of individual users.

The recent introduction of the idea of National ITS Architecture and regional ITS Architecture layers by USDOT brings the US and KAREN architectures closer together. Although the freedom to allocate Functions and sub-Functions over Sub-systems is, in line with European diversity, it is greater in Karen.

7.3.6. Communications Architecture

Again the US and European versions of the Communications Architectures are designed for different purposes. The US Communications Architecture provides a detailed analysis of the data loads required by the architecture flows in the Physical Architecture. This analysis uses the greater detail contained in the "Function" and "Data Flow" descriptions in the "Logical" Architecture, plus data for generic deployment scenarios, e.g. urban, interurban and rural. It is also used to support the US standards activity that has been on-going since June 1996.

The European Communications Architecture looks at the characteristics of the main "Data Flows" in some of the "example Systems" produced for the Physical Architecture. Some recommendations are made for future standards creation, and advice is provided on the desired characteristics of data links, including those with "Terminators". It is recommended that the methodology used for the analysis is applied to the communications in "example Systems" that are created by user of the European ITS Framework Architecture.

7.3.7. Other Architecture Components

Given the different purposes for the US and European Architectures, it is not surprising to find that other common Architecture components such as Cost Benefits Analysis and Deployment Studies show marked differences. These differences also arise because the two Architectures will be used in the different political, economic and social environments that exist in the US and Europe.

The US National ITS Architecture includes a Risk Analysis Document. This Document presents an analysis of potential critical risks that may delay or prevent the deployment of ITS technologies, and recommends mitigation plans, which will eliminate of reduce these risks to the deployment process. The KAREN Project did not carryout this work directly. It was produced by a separate project called RAID, whose members were part of the KAREN Project team. The RAID Project has produced its own report on the risks in Europe and their mitigation strategies.

7.4. Comparison between European ITS Framework and ISO TICS Reference Architectures

7.4.1. Introduction

The ISO System Architecture is known as the TICS (Traffic Information and Control Systems) Reference Architecture. It is currently (August 2000) in the form of an ISO "Type 2 Report" and has been approved in this form by ISO. The reason for its creation was to provide a platform for the other groups within the transport telematics part of ISO, rather than to act as a "World-wide Standard". However is has been used as the starting point for the creation of an ITS Architecture in Australia – see later section.

Actual comparison with the European ITS Framework Architecture is difficult because the ISO TICS Architecture has been created using a different methodology. The choice made by ISO has been to use the Object Orientated (OO) methodology and to describe its Architecture using Universal Modelling Language (UML). However the following sections provide some comments on how the ISO Architecture relates to the European Architecture.

7.4.2. User Requirements

For the ISO Architecture, the user requirements are defined as the TICS Fundamental Services. As with the European ITS Framework Architecture, they were produced before the start of the Architecture itself, and are described in a separate document within the range of ISO TICS Architecture documentation. The Fundamental Services are divided into 32 Service Groups.

The ISO TICS Fundamental Services were used as one of the main starting points for the definition of the European ITS User Needs. Their relationship is shown in the second of the two spreadsheets (Appendix F) containing the European ITS User Needs and is not discussed further in this Document.

7.4.3. Architecture Definition

The ISO TICS Architecture does not have a Context Diagram because this does not exist as such in the OO methodology. The links to the outside World are shown in the highest level, or Aggregate Use Case diagram. In the ISO TICS Architecture, the outside World is represented by 9 “Actors”, comprising, Conformance Agency, Financial, Information Provider/Consumer, Infrastructure, Location Data Source, Service Enabler, Service Provider, User and Vehicle. These “Actors” all interact with the Architecture and all of them have a hierarchy of other actors underneath them. When the hierarchies are included, the total number of “Actors” with which the Architecture interfaces becomes 33. The ISO TICS Architecture ignores “passive Actors” at the highest level even though they provide input to the Architecture. They are modelled as “Interface Classes” and are physically connected to hardware sensors. Examples of this type of “Actor” are “Traffic” and “Environment”.

The Aggregate Use Case Diagram also shows that at this level there are 8 high level Use Cases. In rough terms these will correspond to the eight high level Functional Areas in both the European ITS Framework and US National ITS Architectures. The list of Use Cases and Functional Areas is shown in the table below.

Table 5 Comparison of the functionality in the ISO, US and EuropeanArchitectures

ISO TICS Reference Architecture	US Logical Architecture		European ITS Functional Architecture	
Name	**No.**	**Name**	**No.**	**Name**
Traveller Information	1	Manage Traffic	1	Provide Electronic Payment Facilities
Traffic Management	2	Manage Commercial Vehicles	2	Provide Safety and Emergency Facilities
Vehicle	3	Provide Vehicle Monitoring and Control	3	Manage Traffic

ISO TICS Reference Architecture	US Logical Architecture		European ITS Functional Architecture	
Name	**No.**	**Name**	**No.**	**Name**
Commercial Vehicle	4	Manage Transit	4	Manage Public Transport Operations
Public Transport	5	Manage Emergency Services	5	Provide Advanced Driver Assistance Systems
Emergency	6	Provide Driver and Traveller Services	6	Provide Traveller Journey Assistance
Electronic Payment	7	Provide Electronic Payment Services	7	Provide Support for Law Enforcement
Safety	8	Manage Archived Data	8	Manage Freight and Fleet Operations

From this point on comparisons between the ISO and European ITS Framework Architectures become difficult and largely meaningless. This is due to the difference in methodologies rather than anything else.

7.4.4. Current Status

During 1998 it was recognised by the US, European and Japanese representatives participating in the ISO Working Group (WG1) responsible for the Architecture that it needed to be improved with regard to its consistency and readability. These and other comments were submitted to WG1 in November 1998, the European ones being submitted through ISO/TC204/WG1/SG7 which is responsible for representing European interests in this area. The comments have resulted in the production of a revised and improved version that is the "Type 2 Report" referred to earlier. A new Work Item for WG1 has also been proposed, to provide some "bridge building function". This would describe the TICS architecture using a Function Oriented approach and look to see if there are any "linkages" between the two approaches.

7.5. Comparison with the Australian Architecture

The Australian ITS Architecture has been created to aid the deployment of ITS in that country. The main reason for its creation appears to be the need to recognise that some Australian environments are very different to those in other Countries. This is largely a reflection of the geography and climate of Australia that requires travel in some urban and most remote areas to only be undertaken when more and/or different information is available to the traveller.

The Architecture itself is very similar to the ISO Architecture described in the previous section. It uses the OO methodology with the same high level Use Cases and "Actors". The main difference is in the emphasis that has been placed on the needs of urban and remote areas. It is particularly seen in areas of functionality such as pre-trip information, on-trip information, on-trip Public Transport information and route guidance.

8. References

(a) CONVERGE, *Guidelines for the Development and Assessment of Intelligent Transport System Architectures*, Framework IV Transport Telematics Project CONVERGE (TR1101), Deliverable 2.3, 1998.

(b) Gaillet J-F (Ed.), *Recommended Methodology for Transport Telematics Architectures*, SATIN Task Force, DRIVE II project CORD (V2056), 1994.

(c) Franco G and Jesty P H, *Architecture Analysis Tool Getting Started Manual*, Framework IV Transport Telematic Project CONVERGE (TR1101), Deliverable DSA4.2, 1997.

(d) ISO TC204 WG1, *Transport Information and Control Systems - Reference Model Architecture(s) for the TICS Sector - Part 1: TICS Fundamental Services*, ISO/TR 14813-1, 1998.

(e) ISO TC204 WG1, *Transport Information and Control Systems - Reference Model Architecture(s) for the TICS Sector - Part 2: Core TICS Reference Architecture*, ISO/PDTR 14813-2, 1998.

(f) Details of the US National ITS Architecture, including documents in PDF format, can be found at the following Web Site: "http://www.iteris.com/itsarch/".

(g) QUARTET, *Final Methodology for Architecture Assessment*, DRIVE II Project QUARTET (V2018), Deliverable Nº 53, 1995.

(h) QUARTET, *Operational Tool for the Analysis of Complex Architectures*, DRIVE II Project QUARTET (V2018), Deliverable Nº 54, 1995.

(i) The documents created by the SATIN Task Force can be found by searching the European Commission Web Site "http://www.trentel.org/index.htm".

(j) COMETA, *Commercial Vehicles On-Board Systems Integrated Architecture(s) Specifications, Risks Analysis and Implementation Time Schedules Hypothesis*, Framework IV Transport Telematics Project COMETA (TR4005), Deliverable 6.1, 2000, which can be found by searching the Project Web Site which can be found at "http://www.cometa-project.com/".

(k) European ITS User Needs (D2.2), Issue 1, August 2000.

(l) European ITS Functional Architecture (D3.1), Issue 1, August 2000.

(m) European ITS Physical Architecture (D3.2), Issue 1, August 2000.

(n) European ITS Communications Architecture (D3.3), Issue 1, August 2000.

(o) European ITS Models of Intelligent Transport Systems (D3.7), Issue 1, August 2000.

(p) European ITS Deployment Approach and Scenarios (D4.2), Issue 1, August 2000.

(q) European ITS Framework of Required Standards (D4.1), Issue 1, August 2000.

Note that all of the European ITS Documents listed above can be found on the European ITS Framework Architecture CD-ROM. They can also be obtained by searching the European Commission Web Site “http://www.trentel.org/index.htm”.

Appendix A The European Architecture Framework

It is important to distinguish between the three different rôles of an “Architecture” in the European scenario. They can be defined as follows:

1. At the level nearest the design there is the system architecture which is of fundamental importance when Systems are created by integrating two or more Sub-systems. The system architecture provides the structure around which a class of systems may be developed. It is the level at which the basis for “Working and Workable Systems” is set up – see reference 8(a).

2. Intermediate level architectures are being defined for EU Member States to create the conditions for a market of compatible and modular solutions, and for establishing the implementation of nation-wide inter-operable solutions and services. Two or more Systems are inter-operable if they can pass data between each other to their mutual benefit, i.e. to provide harmonious and/or complementary functionality: inter-operability includes the technical, operational and organisational aspects.

   National architectures are also the tool for guiding national research initiatives, and for providing the reference document for standards and rules to be used for national applications. They also provide the common terminology for specifying systems, and possibly recommend the standards and the interfaces to be used to achieve compatibility at the European level. Other system architectures may also be defined by the private or public sector independently of any national ITS architecture.

3. The European ITS Framework Architecture is at a “level” such that it can be used as a reference by all ITS Architects and System Designers. It is intended to be the foundation for building the other types of Architecture and can also be used directly for creating Systems. It will enable the Architects and Designers to guarantee compliance at the interfaces of other Systems so that seamless services can be provided to cross-border Travellers, and an open European market of compatible components can be established.

The KAREN Project has had the task of creating the key elements of the European ITS Framework Architecture. Whilst it took account of what exists already, it also had to look to the future and the desires expressed in the List of European ITS User Needs, which may not be fully satisfied by existing systems. Thus, whilst the Framework Architecture concentrates on the functionality and other features that are needed to satisfy the User Needs, it does not “re-invent the wheel” if there is an existing solution.

The diagram in Figure 2 on the next page provides an overview of how the various European Architectures relate to each other. Initially, there will be two main categories of Architecture co-existing:

1) Those that will conform fully to the European ITS Framework Architecture. These will be made up of:

a) National and Local Architectures that have been designed to conform to the Framework Architecture. These may themselves produce Service Architectures that also conform. They will eventually be in the majority;

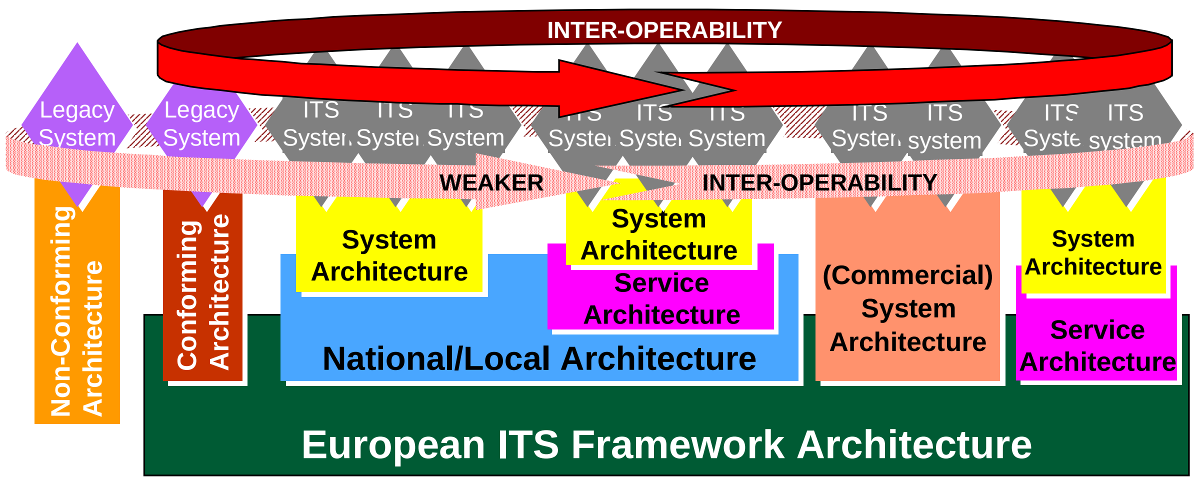
b) Legacy systems that do conform to the European ITS Framework Architecture. Because the Architecture does not "re-invent the wheel", there will be some legacy systems in this category.

c) Service Architectures produced directly from the Architecture.

d) Commercial Architectures produced directly from the Architecture by European industry.

Each of these Architectures (and the European ITS Framework Architecture itself) can be used to create ITS Systems, all of which will conform to the Architecture. By ensuring this conformance, the first major step towards providing System inter-operability will have completed.

Figure 2 Overview of the Relationship Between the Various Architectures



2) Those legacy systems that do not conform completely to the European ITS Framework Architecture. Whilst they may not be fully inter-operable with those Systems that do conform, one can expect there to be a useful degree of mutual functionality especially for those Systems that are not very old.

In order to make legacy systems in group 2 fully inter-operable with those in group 1, there needs to be a migration strategy to inform their owners on how they can become compliant with the European ITS Framework Architecture. This is discussed in the European ITS Deployment Study – see reference 8(p). It should be noted that whilst 'migration' is often assumed to mean 'replace', in practice it can also mean 'enhance' or 'add to'. Both of these are usually less contentious and expensive to implement than completely replacing a System !

The European ITS Framework Architecture also has other rôles, such as providing the bridge between the ITS community and those creating other current and future technologies (e.g. in the telecommunications or banking world). It may also be a catalyst for new research where its functionality requires technologies that do not yet exist.