Cooperative Vehicle-Infrastructure Systems

D.CVIS.3.2	High Level Architecture

SubProject No.	1.2	SubProject Title	CAG
Workpackage No.	WP3	Workpackage Title	Architecture and System Specifications
Task No.	NA	Task Title	NA
Author(s)		Andreas Schmid, Arnor Solberg, Christer Larsson, Erik Olsen, Frank Tuijnman, Gino Franco, Hans Hadderingh, Jean-François Gaillet, Josef Kaltwasser, Knut Evensen, Marcel Konijn, Mick Baggen, Richard Bossom, Siebe Turksma, Thierry Ernst, Zeljko Jeftic	
Dissemination level PU/PP/RE/CO		PU	
File Name		DEL_CVIS_3.2_High_Level_Architecture_v1.0	
Due date		2009-05-11	
Delivery date		2009-05-11	

Abstract	This document defines the High Level Architecture of Cooperative Systems as envisioned by the CVIS project. The main content of the document are five viewpoints describing the architecture considering physical/engineering, functional, information, communication and organisational aspects.

Information Society Technologies	Project supported by European Union DG INFSO IST-2004-2.4.12 eSafety – Cooperative systems for road transport
Project reference	FP6-2004-IST-4-027293-IP
IP Manager	Paul Kompfner, ERTICO – ITS Europe Tel: +32 2 400 0700, E-mail: cvis@mail.ertico.com

Control sheet

Version history			
Version	**Date**	**Main author**	**Summary of changes**
0.01	2006-10-25	Andreas Schmid	TOC proposal
0.02	2006-11-16	Andreas Schmid	BRU meeting results: viewpoints and HLA
0.10	2006-12-20	Andreas Schmid et. al	Update for telephone conference
0.11	2006-12-27	HLA Group	Update for CVIS distribution
0.12-0.13	2007-01-16	Richard Bossom , Frank Tuijnman , Zeljko Jeftic	Updating the entire document Info, Physical and Organisational viewpoints
0.14	2007-01-19	Richard Bossom, Frank Tuijnman, Josef Kaltwasser, Thierry Ernst , Zeljko Jeftic	Revision of the document, mainly Chapter 2
0.15-0.18	2007-02-23	Andreas Schmid, Zeljko Jeftic, Frank Tuijnman, Thierry Ernst, Arnor Solberg, Knut Evensen, Erik Olsen	Revision of the document, especially chapters 2 and 3
1.0	2007-02-28	Zeljko Jeftic	Smaller changes
2.0	2009-05-08	Axel Burkert, Erik Olsen, Zeljko Jeftic, Hannes Stratil, Knut Evensen	Update to Version 2.0 considering annual review comments

	Name	Date
Prepared	Andreas Schmid, Erik Olsen, Frank Tuijnman, Gino Franco, Hans Hadderingh, Jean-François Gaillet, Josef Kaltwasser, Marcel Konijn, Mick Baggen, Richard Bossom, Thierry Ernst, Siebe Turksma, Zeljko Jeftic, Axel Burkert	2007-01-19
Reviewed	Schokker Boudwijn (LogicaCMG) and Paul Mathias (Siemens)	2007-02-26
Authorized	Paul Kompfner	2007-03-01

Circulation	
Recipient	**Date of submission**
V0.11: CAG, WP3 Leaders, DEPN, Test sites for comments	2006-12-27
V0.18: CAG, WP3 Leaders, DEPN, Test sites, Reviewers	2007-02-20
Project partners	2007-02-28
European Commission Project partners European Commission	2007-02-28 2009-05-11 2009-05-11

Table of Contents

Abbreviations and Definitions

Abbreviation	Definition
CAG	Core Architecture Group (horizontal CVIS subproject leading the technical work in the project)
CALM	Continuous Air-interface Long and Medium range
CF&F	Cooperative fleet and freight applications (subproject within CVIS)
CINT	Cooperative inter-urban applications (subproject within CVIS)
COMM	The Communication and networking (subproject within CVIS)
COMO	Cooperative Monitoring (subproject within CVIS)
CVIS	Cooperative Vehicle Infrastructure Systems
CURB	Cooperative urban applications (subproject within CVIS)
DEPN	Deployment Enablers (CVIS subproject investigating non-technical issues)
FOAM	Framework for Open Applications (subproject within CVIS)
FRAME	Common name for European ITS Framework Architecture
HLA	High Level Architecture
HMC	Host Management Centre
IETF	Internet Engineering Task Force
ITS	Intelligent Transport Systems
MDA	Model Driven Architecture
OSGi	Open Services Gateway initiative
OSI	Open Systems Interconnection
POMA	Positioning, maps & local referencing (subproject within CVIS)
RM-ODP	Reference Model – Open Distributed Processing
SAP	Service Access Point
SDK	Service Development Kit
SP	Sub-project
TOGAF	The Open Group Architecture Framework
WP	Work package
RSU	Roadside Unit
V2V	Vehicle to Vehicle (communication)

V2I	Vehicle to Infrastructure (communication)
V2C	Vehicle to Centre (communication)

Executive Summary

CVIS is developing a platform for a Cooperative Systems Network which allows mobile units, roadside units and centres to interact and communicate seamlessly.

It is of core importance for a Cooperative Systems Network to establish connections among the intended communication partners even if they are moving geographically. Hence, the communication technology developed within the CVIS project is based on the global IPv6 standard and integrates mobile and fixed devices through various access networks. The CALM specifications have been chosen as a main communication technology to be validated and demonstrated in CVIS. Basic functions to detect or to search for partner applications are also developed. It is foreseen that most short range communication will be done on a peer-to-peer basis, where every CVIS entity can take a role of either client or server.

CVIS provides a complete platform for ITS applications. This platform provides access to all basic functions such as communication functions, access to sensors and actors, security and remote management of software which is needed to handle the life cycle of applications. Based on JAVA/ OSGi, applications can be deployed in a controlled manner over the (wireless) network. A key component in a cooperative world is Host Management Centre which allows, hence the name, a detailed management of the applications on the CVIS hosts (e.g. load, renew or remove applications).

The presented High Level Architecture is a result of the ongoing specification and architecture work in the CVIS project. The project uses and augments results of various projects such as GST and SEVECOM, and standardisation initiatives such as ETSI TC ITS and CALM. There is also a close relation to other IP projects on Cooperative Systems such as SAFESPOT and COOPERS.

It must be mentioned that closely related to this deliverable is the D.CVIS.3.3 Architecture and System Specifications, which provides detailed information about CVIS overall architecture and its components (e.g. detailed description of component functionalities and interfaces). This High-Level Architecture has been a useful step in developing D.CVIS.3.3, which can be found on www.cvisproject.org

This deliverable in its previous version formed an important input to a common European High Level Architecture for Cooperative Systems. This important task was supported by the EC under the COMeSafety project. During 2008 this common European ITS Architecture was developed and accepted by all relevant European ITS participants, and the resulting document now represents state-of-the-art regarding European ITS Architecture with a special focus on communications architecture.

CVIS played a central role in this work, and we have now fully integrated this common architecture in all aspects of CVIS developments and demonstrations. Hence to avoid duplication and ease maintenance of the various documents, CVIS will reference this European architecture in all relevant chapters of this High Level Architecture.

To visualise the process, the first drawing coming from the original High-Level Architecture document is shown below.

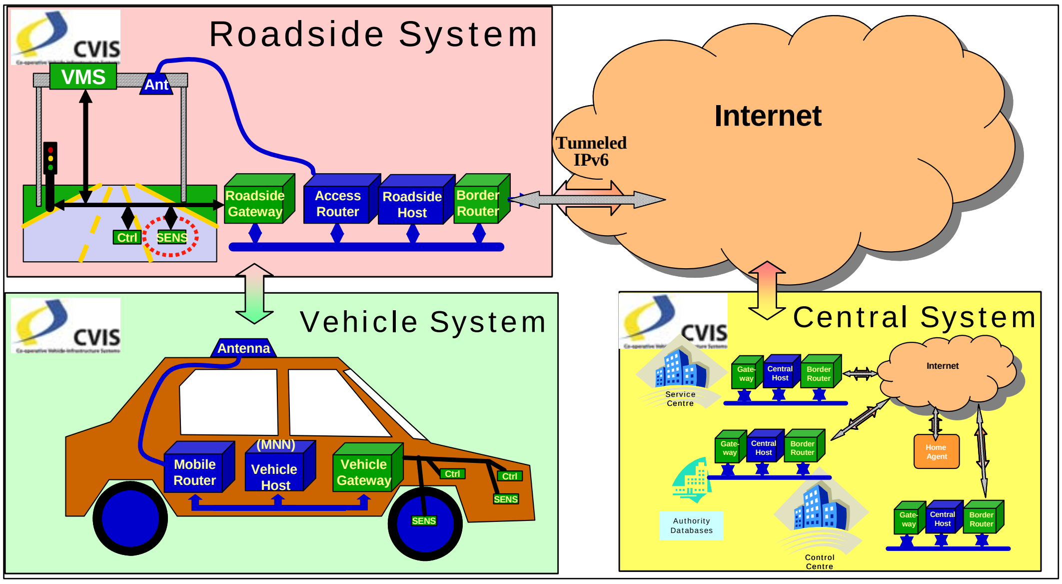

Figure 1: CVIS high level architecture.

In principle, the architecture foresees roadside systems, vehicle systems and central systems connected by an IPv6 network. Each of the systems features a CVIS Host and a CVIS Router. The CVIS host serves as platform for the software active in each system, the router is responsible for the connectivity among the systems. Finally, gateways allow to include legacy data and use them within any CVIS system.

The updated result from COMeSafety shows the same architecture viewpoint, but adds more detail and changes the terminology slightly. See the following figure:

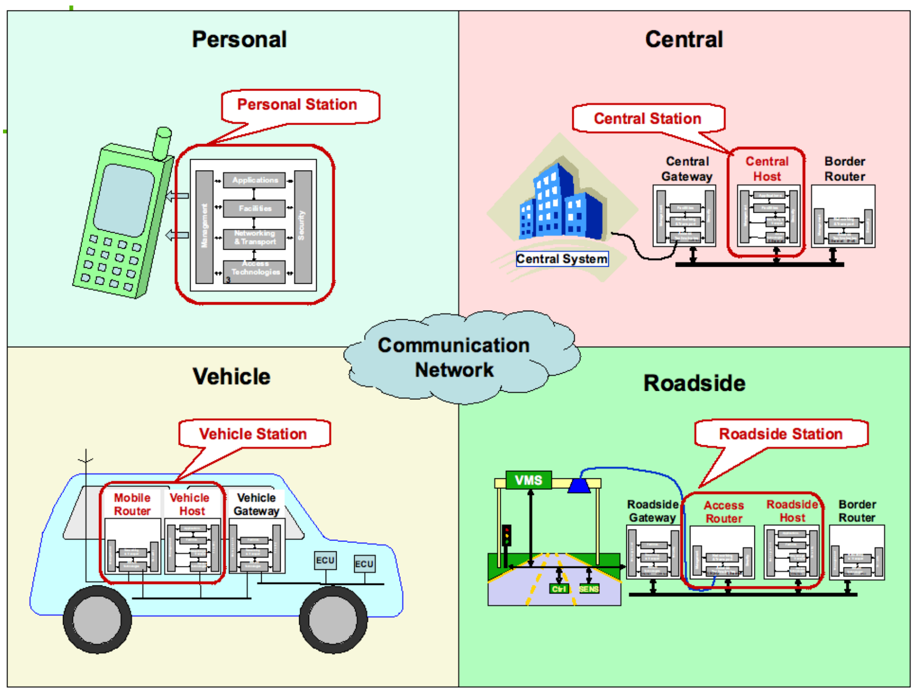

Figure 2: COMeSafety high level architecture.”

Note that the COMeSafety drawing adds important new facets:

- The concept of a Station delimits the scope and responsibility of a COMeSafety defined ITS system inside the wider legacy systems.
  - A Station is indicated with the red frame around the host and router functions. This concept is now widely accepted in CALM and other standards groups. A Station can range from a combined Router and Host software block inside a mobile phone, to multiple physical Hosts and Routers in a vehicle. The important aspect is that all layers in the OSI architecture need to be present in order to have a stand-alone operation which can communicate with other Stations.
  - Stations will form flat network connectivity topologies. This means that all Stations in principle have similar roles and there is no implicit hierarchy. For instance all Stations in vehicles and roadsides have peer-peer connectivity. That means any Vehicle Station or Roadside Station can offer services to be consumed by any other Roadside Stations or Vehicle Stations.
  - A delimited scope means that everything inside a Station is the COMeSafety (and therefore CVIS) responsibility to define and specify, while everything outside is defined by other actors.
- The simplified OSI layered model shown in the following figure (Figure 3) is also a central result from COMeSafety. This concept inside ITS mainly started from the CALM standardisation initiative, and has been the root of the CVIS Core Technology developments. OSI is important because it shows how management and security is connected to the various other communication functions, and the Facilities sublayer that comes from CVIS is now widely considered to be a core function in ITS. The communication starting point is natural since ITS is inherently communications intensive.
  This simplified OSI architecture model is further detailed in this High Level Architecture document.

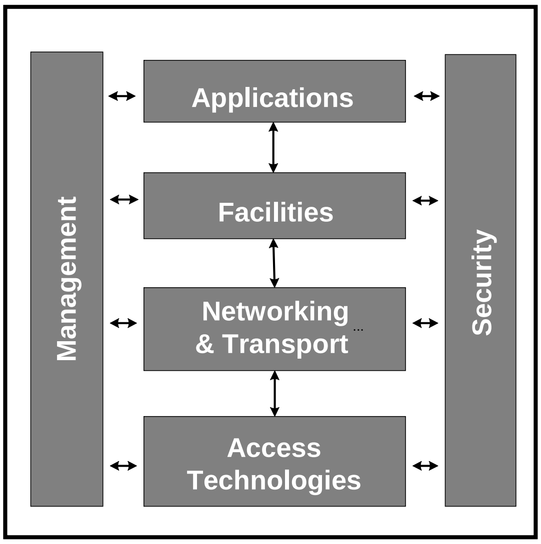

Figure 3: Simplified OSI layered model

The CVIS reference execution platform supports V2V, V2I and V2C. V2V can be accomplished through two different channels. One is direct, ad-hoc (peer-to-peer) communication between two vehicles in a close vicinity of each other, usually for safety, time-critical, purposes. Another way of V2V communication is via IPv6 where two vehicles, which might be in close vicinity or on the other side of the world, are able to communicate with each other as two IPv6 nodes.

Cooperation with projects such as SAFESPOT or HAVE-IT provides additional platform requirements. SAFESPOT is dealing with safety, time-critical, information sharing and has extensive cooperation with CVIS. Common work on use requirements, architectures and implementations between CVIS and SAFESPOT has been done during these 3 years. In addition, joint demonstration will be shown at the CVIS Annual Review Y3 in Helmond.

Considering HAVE-IT project, which is working on highly automated driving, CVIS partner EFKON is providing a communication units to the project based on CVIS work and is creating a link to understanding the work and requirements from this important project. HAVE-IT work on highly automated driving could be supported by cooperative systems in the future.

This document, the CVIS D.3.2 High Level Architecture, describes the key elements of the System Architecture elaborated within the CVIS project. These comprise

- the project idea and approach chosen for the elaboration of the CVIS high level architecture is described in chapter 1
- the high level architecture seen from different viewpoints, namely the
    - physical and engineering viewpoint (this does not refer to the physical hardware design which is described in the “Solution Design” view presented as “Guideline” in chapter 3) providing a high level overview and defining system entities in chapter 2
    - the functional viewpoint which defines the main architectural elements that deliver the CVIS functionality in chapter 2.2
    - the information viewpoint detailing the interchange of information or data between CVIS applications in chapter 2.3
    - the communication viewpoint which considers the communication needs of the system entities defined by the physical viewpoint and the information to be exchanged between entities considering communication requirements in chapter 2.4
    - and the organisational viewpoint explaining the organisational aspects of the future Cooperative Systems world in chapter 2.5

1. Introduction

1.1. Intended audience

The intended audience of this document are all stakeholders interested in the CVIS project and in the Cooperative Systems. Four main audiences can be distinguished:

- System architects and developers in the CVIS project who need to identify how parts developed in the subprojects (SP) are related to other project activities, to check consistencies / overlaps and to identify missing links,
- The European Commission who is supporting the CVIS project,
- The correlated projects in the area of Cooperative Systems,
- External stakeholders who would like to understand the CVIS system architecture.

1.2. Document structure

The document consists of three main chapters. Chapter 1 introduces to whom the document is devoted, why it is created and introduces briefly CVIS. It also explains the approach chosen to elaborate on the issue "High Level Architecture".

Chapter 2 introduces CVIS High Level Architecture elements and high level common ideas about the Cooperative System (e.g. main functions, main entities) through five different Viewpoints of the CVIS Architecture: Physical/Engineering, Functional, Information, Communication and Organisational. For each Viewpoint the most significant issues will be addressed.

Chapter 3 continues with CVIS project specific choices taken for the architecture during the project lifetime.

1.3. Rationale

Reasons and motivation for drafting this deliverable are to achieve the following:

- A common understanding among creators of the CVIS system (e.g. which aspect of the system is discussed at a certain description),
- to support interoperability between all parts of the CVIS system,
- to ensure that the CVIS system parts together deliver the desired benefit to their users,
- to bring important input to the work on creating a common European Architecture for Cooperative Systems.

1.4. What is CVIS?

Vehicles today are generally designed to operate without having possibilities to communicate capabilities with each other and with the road infrastructure. It is up to the driver to perform tasks such as pre-trip planning, on-trip re-planning, incident avoidance and accident mitigation with the help of free-standing support systems and visual information from the infrastructure.

In addition to this, each of the major road transportation stakeholders currently faces major challenges. To meet the ever increasing demand for safer and more efficient transportation of people and goods, radical new approaches to the way we look at the transportation system are necessary.

CVIS is a European Commission sponsored project that aims at enabling flexible, harmonised and open, communication and cooperation between the entities that will lead to the improvement of existing road services and the development of new services. Intelligent Cooperative Systems that are based on vehicle-to-vehicle (V2V) and Vehicle to Infrastructure (V2I) communications increase the "time horizon", the quality and reliability of information available to the drivers about their immediate environment, the other vehicles and road users, enabling improved driving conditions leading to enhanced safety and mobility efficiency. Intelligent Cooperative Systems hold the promise of great improvements both in the efficiency of the transport systems and in the safety of all road users.

Similarly, Cooperative Systems offer increased information about the vehicles, their location and the road conditions to the road operators and infrastructure, allowing optimized and safer use of the available road network, and better response to incidents and hazards.

Challenges of architecture in CVIS

Given what was said so far about the nature of Cooperative Systems, an architecture for CVIS will inevitable differ significantly from the existing architectures for typical in-vehicle or ITS infrastructure systems. The systems in these domains have grown in 'silos' over the last 15+ years, happily ignoring each other. While infrastructure systems treat vehicles as moving metal objects - that can be detected by loops, cameras or radar, but have unpredictable behaviour - in-vehicle systems are typically unaware of the intelligence that could be obtained from the (ITS) infrastructure surrounding them. Obviously, the way these actors treat each other is deeply implemented in their system architectures.

An innovative, genuinely Cooperative System can not be built by retaining these legacy architectures, but needs to discover an entirely new route, where vehicles and infrastructure see each other as equal partners, both holding intelligence that can be added to achieve a whole that is more than the sum of its parts. Therefore, the establishment of a suitable CVIS architecture plays a prominent role in the CVIS work programme. This High Level Architecture illustrates first steps towards this direction, trying to obtain a clear top level view on what the CVIS system is, which components it is made of and how they interact and fit together. This document that sets the scene for the individual architecture work packages (WP3) in the different sub projects, while at the same time providing the glue that ensures that these are following the same approach and provide consistent results that form one single CVIS.

1.5. Requirements towards the Architecture

The requirements towards the CVIS architecture were first defined in WP2 and summarised in the D.CVIS.2.2 Use Cases and Requirements deliverable early in the project. These requirements have been used to develop specifications for the CVIS architecture, in the implementation of the prototype platform and are being used in the validation process.

CVIS Core Architecture Group members believe that these requirements should not be replicated from document to document as that risks creating several versions of the requirements list. We would still like to see D.CVIS.2.2. Use Cases and Requirements, which is available on the www.cvisproject.org, as one common place where the requirements are stated.

1.6. CVIS Architecture Key Ideas

A CVIS based System can be regarded as a peer to peer network of hosts, which are all connected on basis of public IPv6. IPv6 provides support for mobile nodes in an IP network. Still connections to the network can be achieved through any available access networks e.g. 3G, Wireless LAN, IPv4 Service providers.

There are different categories of hosts such as *central hosts* (e.g., control centre and service centre), *roadside hosts* and mobile hosts (e.g. hosts residing in a vehicle or a PDA) as illustrated in Figure 4

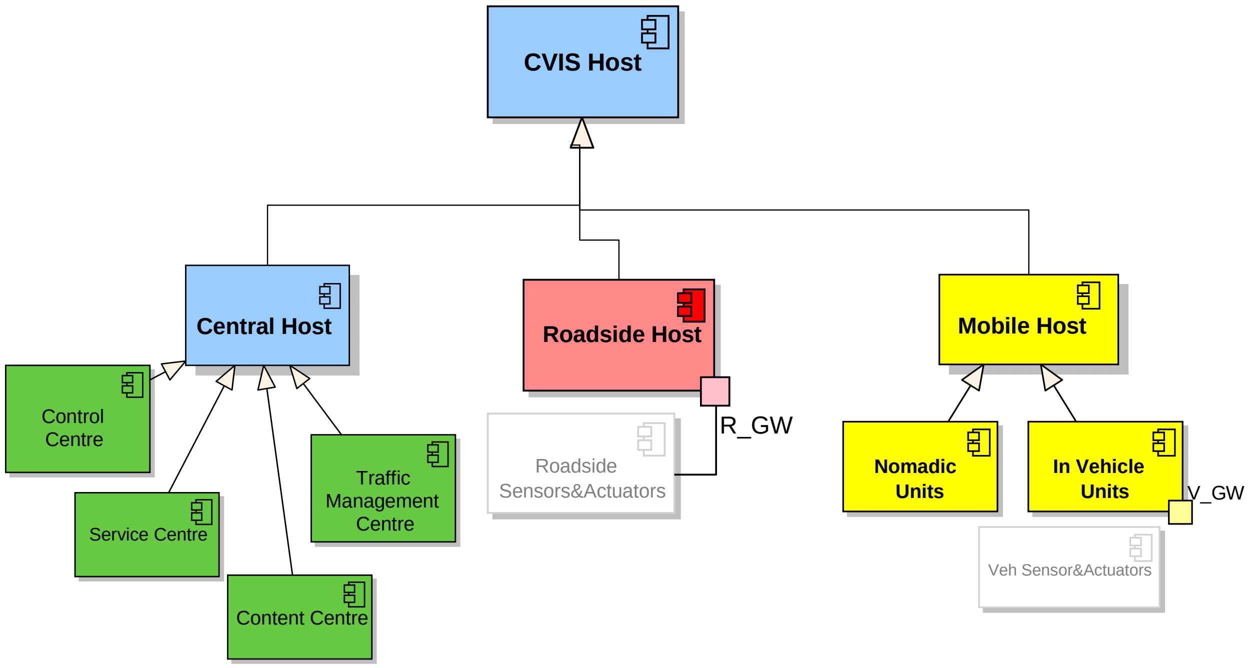

Figure 4: CVIS concept; categories of hosts

CVIS Hosts can adopt various roles. An important aspect of CVIS hosts is that they can play both the role of consumer and supplier, i.e., they can supply or consume information/ services (even at the same time in the same host) as illustrated in Figure 5.

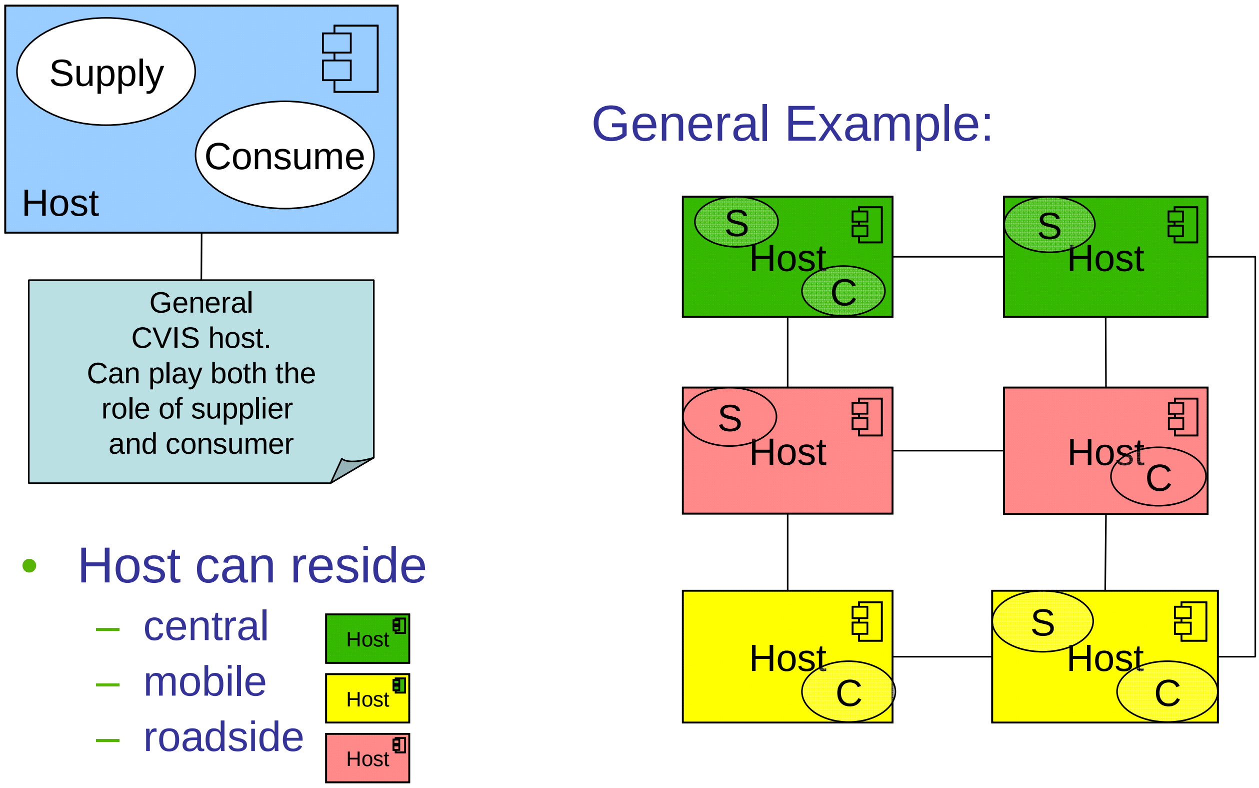

Figure 5: CVIS host concept; Supplier and Consumer

Typically End Users would interact with mobile (nomadic or in-vehicle) hosts which are realised in different hardware devices (e.g. PDA, vehicle integrated). Service data of any kind can be exchanged between all hosts.

During execution the hosts of the different kind will exchange information between each other. There will be an “information flow” between the “Applications” inside the nodes. To enable this inter-operation between hosts for a specific service, communication partners need to have the right applications on their hosts present.

As during lifetime or during the travel of a mobile host it can happen, that a corresponding application is missing, CVIS includes services to add, update or change software applications. This “Remote Management” is one of the key functionalities in the system.

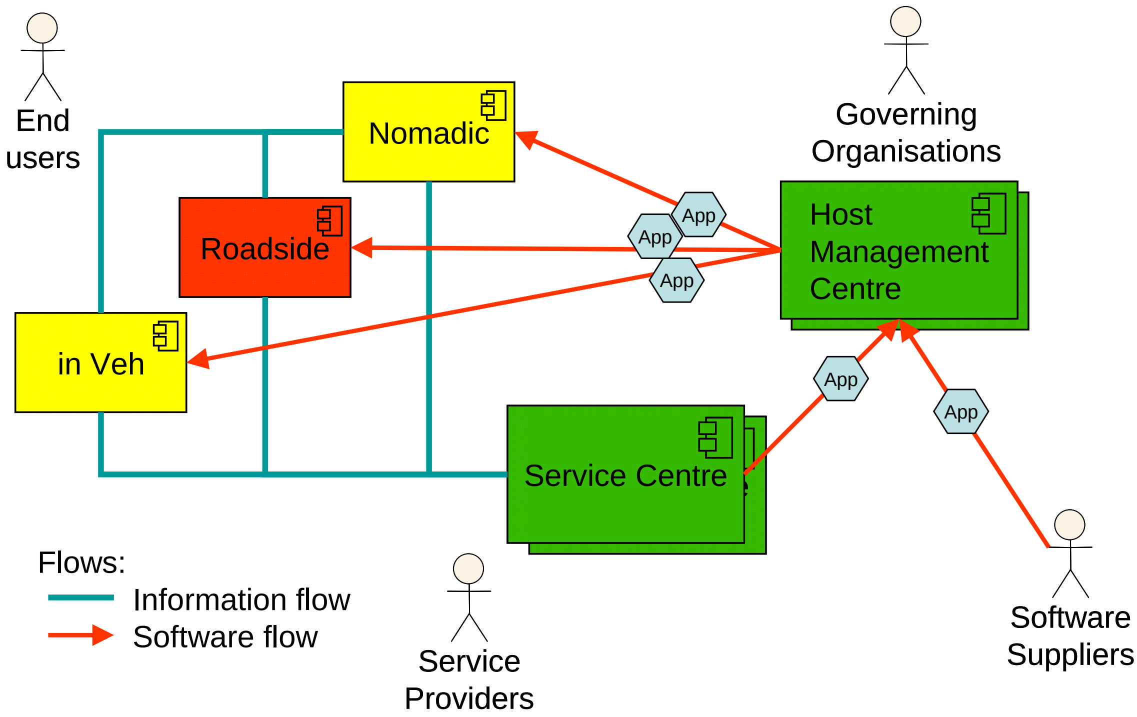

Figure 6: Information and software flows

In order to enable a management of applications (e.g. load, renew, remove) each host that shall be managed must belong to exactly one "Host Management Centre (HMC)". These HMCs take care of all software management issues of its assigned hosts.

HMCs will be run by organisations which want to deploy CVIS hosts in order to allow their target users to consume a service. Details how flexible this can be handled and which further items, functions and organisational patterns are needed will be given in Chapter 2.

1.7. Approach to describe the CVIS System Architecture

This document is part of a series that leads to the full CVIS architecture specification. In this document we present a concise description of the CVIS architecture from a series of different "viewpoints". The viewpoints have been collected from several system architecture approaches such as [FRAME], [RM-ODP], [MDA], [TOGAF]. This document has been deliberately kept short and accessible for an internal and external audience. Its primary focus is on high level design issues. It partly states new design decisions, and partly reiterates old design decisions that were made earlier during the proposal phase. The consequence of aiming both for a concise document and a wide scope is that it does not elaborate issues in detail. Details will be done in the full architecture specification, i.e. in various Architecture documents of the CVIS sub projects.

Below we give an overview of the viewpoints that are used to describe the architecture of CVIS.

1. Physical/Engineering Viewpoint

   The physical/engineering viewpoint presents an overview of the main system entities that are relevant to CVIS.

2. Functional Viewpoint

   The functional viewpoint presents an overview of the system functions and has to depict the system behaviour.

3. Information Viewpoint

   The information viewpoint addresses two topics:

   a. The distinction between information and meta-information, and the use of meta-information during run-time.

   b. A high level taxonomy of the information relevant to CVIS.

4. Communication Viewpoint

   This viewpoint describes the communication technology that will be developed within the CVIS project.

5. Organisational Viewpoint

   This viewpoint presents the organisational aspects of the Cooperative world. It explains why some central bodies are necessary in a Cooperative world.

1.8. The High Level Architecture and the Sub-Projects

The following illustration in Figure 7 presents the relationship between the HLA and the work of the Sub-Projects. It also includes perspectives of the Deployment Enablers (DEPN) sub-project.

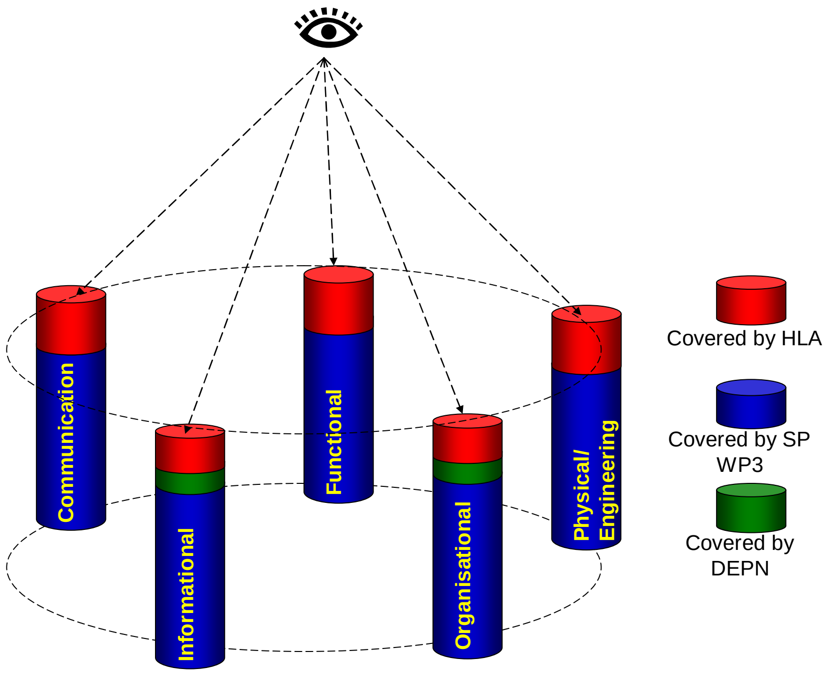

Figure 7: CVIS Architecture Viewpoints

The Diagram illustrates that an architecture description of CVIS is a set of various perspectives ("viewpoints") from which we look at the CVIS System. Each column is representing one viewpoint as an example.

The coloured parts indicate which groups (e.g. sub-projects) in CVIS are expected to elaborate on the various viewpoints in depth. According to the task of a group, the depth of elaboration will have to be determined by each group. The HLA introduces all viewpoints in this document.

CVIS has followed a combined top-down and bottom-up process in it's architecture development. In parallel with a top-down approach of defining use cases and system requirements before architecture definition, CVIS started a bottom-up process by specifying and developing certain core technology functionalites (example: CALM framework).

In addition, as a third dimension CVIS has been closely interacting with standardisation bodies throughout the project duration. The result is a significant contribution to the definition of the European Communications architecture but also update of CVIS architecture and prototype implementation (example: Introduction of Cooperative Awareness Message).

The present High Level Architecture, together with D.CVIS.2.2 Use Cases and Requirements present the top-down approach, while in D.CVIS.3.3 Architecture and System Specifications, the top-down and bottom-up approaches are coming together.

1.9.CVIS in the European domain – relationship to other projects

Currently there is a large number of pan-European research and development activities leading the development within different areas of the Intelligent Transport Systems (ITS). Many of these activities are supported by the European Commission. In order to easily understand how these activities fit together a template has been prepared. It has been filled in by GST, PReVENT, EASIS, APROSYS and AIDE projects. Figure 8 presents CVIS input.

The CVIS addition to the overall picture is the introduction of cooperation with near roadside equipment. Vehicles will be able to communicate to e.g. traffic lights, road signs. Traffic Management Centres will be connected. Vehicle to vehicle communication will be covered through close cooperation with SAFESPOT IP.

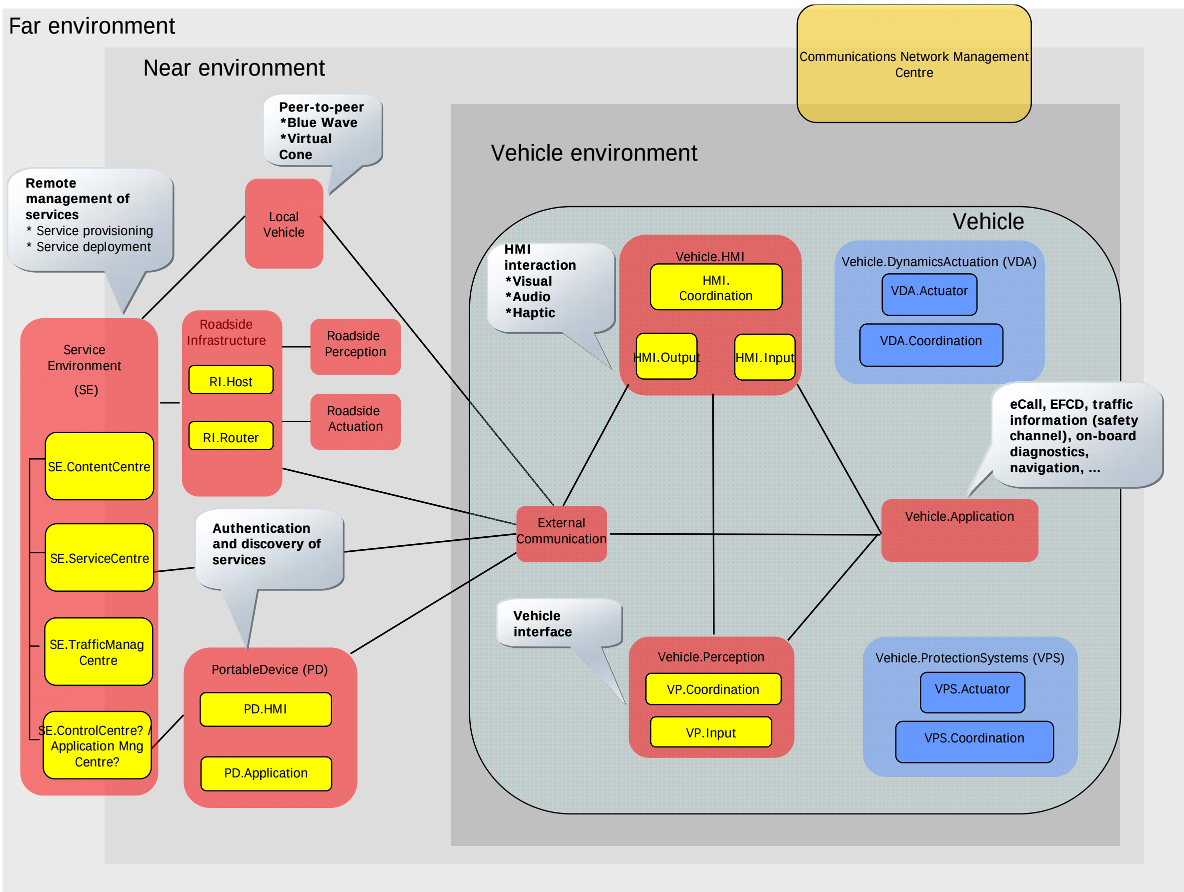

Figure 8: CVIS contribution

Out of the above mentioned projects, CVIS takes further results achieved in GST - an open platform for telematics services while AIDE project guidelines on HMI are followed in CVIS implementations. Due to the nature of the project, very little cooperation has been established with PREVENT, EASYS and APROSYS.

On the other hand very fruitful cooperation has been established with SAFESPOT, COMeSafety, COOPERS, SEVECOM and GEONET projects. These projects have worked on joint communications architecture which was published end of 2008. In this work, the above presented template has not been used, thus the representation of projects, in this template form, with whom we have collaborated the most is not available. However, European Communications Architecture for Cooperative Systems presents much more details of cooperation than this figure. It needs to be mentioned that work has also continued on this common architecture and a version 3.0 is expected during 2009. Please visit www.comesafety.org

2. CVIS High Level Architecture Viewpoints

The CVIS architecture can be seen from different viewpoints, including

- physical and engineering viewpoint (this does not refer to the physical hardware design which is described in the "Solution Design" view presented as "Guideline" in chapter 3) providing a high level overview and defining system entities in chapter 0
- the functional viewpoint which defines the main architectural elements that deliver the CVIS functionality in chapter 2.2
- the information viewpoint detailing the interchange of information or data between CVIS applications in chapter 2.3
- the communication viewpoint which considers the communication needs of the system entities defined by the physical viewpoint and the information to be exchanged between entities considering communication requirements in chapter 2.4
- and the organisational viewpoint explaining the organisational aspects of the future Cooperative Systems world in chapter 2.5

A general overview on the architecture, which is described in the context of the communication viewpoint in chapter 2.4, is shown in the following figure:

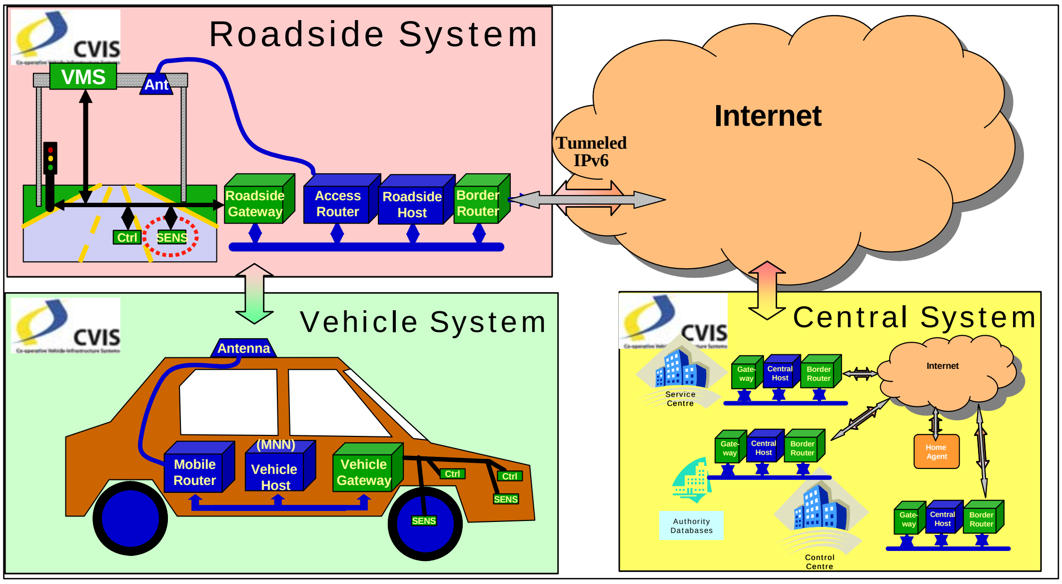

Figure 18 CVIS high level architecture

Please see also document "D.CVIS.3.1 Reference Architecture

The architecture forsees roadside systems, vehicle systems and central systems connected by an IPv6 network. Each of the systems features a CVIS Host and a CVIS Router. The CVIS host serves as platform for the software active in each system, the router is responsible for the connectivity among the systems. Finally, gateways allow to include legacy data and use them within any CVIS system.

The architecture is described from different viewpoints and in more detail in the following chapters.

2.1.Physical / Engineering Viewpoint

In this viewpoint system entities are defined. The system entities still may be of **conceptual** / **logical** nature. Unfortunately this perspective has different names in various domains. Therefore both terms – Physical and Engineering Viewpoint – are kept in the title. "Physical" hardware design shall be described in the "Solution Design" view – which is presented as "Guideline" in chapter 3.

Information and functionality of the system - described in the according viewpoints - can be mapped to the system entities described in this viewpoint. Human actors / users are not explicitly shown here. Due to the level of detail this document also does not illustrate user- or other technical interfaces which will usually be part at each of the illustrated entities.

On a very high level, CVIS can be understood to be composed of entities which are linked by an overall, Internet technology based network (Figure 6). The network provides 'always on' connectivity to any participating entities. This includes mobile units such as in vehicle or nomadic units and stationary units which can be situated at the roadside or in Centres.

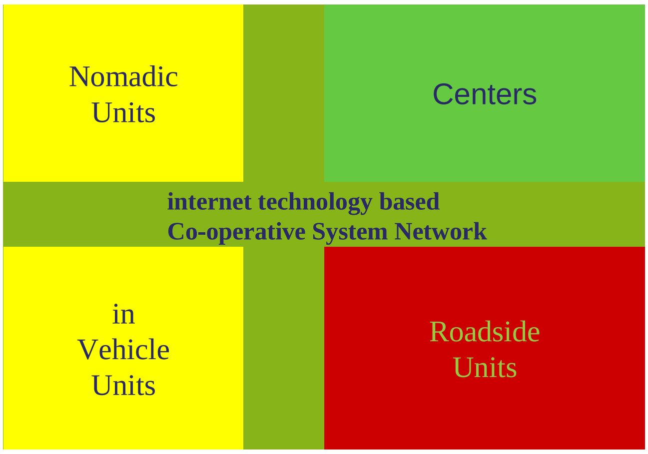

Figure 9: CVIS High level elements Groups

More specific a series of CVIS typical entities can be identified, each of them having a specific role.

Mobile Units

Mobile units are entities that can move around in the (road) network. Mobile units can be of two specific kinds:

In-vehicle units are fixed to a vehicle. These mobile units access the vehicle infrastructure via a vehicle gateway. Thus sensor access or even actuator access (given strict security measures and management of access rights) can be available. The vehicle gateway (V_GW) figures as interface between existing technology and the CVIS world.

Nomadic Units (e.g. PDA, PND) are independent from vehicles. However, it is possible for a Nomadic Unit to take the role of an In-Vehicle Unit, e.g., through a wireless communication channel with a vehicle Gateway.

All types of Mobile Units are ‘always on’ connected to the CVIS network. Thus they can communicate among each other for direct information exchange (e.g. V2V), with road side units (e.g.V2I) or with Centres (e.g. V2C).

Road Side Units

Road Side Units are road operator’s entities. Typically they are fixed and installed as roadside equipment, and are responsible for one or more ITS aspects on one or more road segments.

Road Side Units shall be able to access road side sensors to collect data, and (given strict security measures and management of access rights) to use road side actuators to provide information, advices, warnings or instructions to the traffic participants. The roadside gateway (R_GW) figures as interface between existing technology and the CVIS world.

Exceptions: Also ‘moving’ hosts acting like Roadside Units (e.g. trailer mounted signs, roadwork service vehicles) can act like Roadside Units.

Roadside Units are ‘always on’ connected to the CVIS network. Thus they can communicate among each other for direct information exchange (e.g. I2I), with Mobile Units (e.g.V2I) or with Centres (e.g. I2C).

More detailed aspects of Mobile and Roadside units are described in D.CVIS.3.1 “Reference Architecture”. There, combined with a physical design choice, a more detailed composition of such a “Vehicle-” or “Roadside System” is introduced. Most significant is that each of the “Subsystems” comprises a “Router” and a “Host” Component. The ***CVIS Router*** Component is responsible for connecting devices to the CVIS wireless network. The ***CVIS Host*** Component contains all application components. More details can be found in chapter 3.1.3, “The Technology Sub-Viewpoint”.

Centre Units

Service Centres support different end user services. For each end user service offered by one or more provider there may be one or more service centres. When a user is consuming a service, his ***Mobile Unit*** can connect to a service centre to exchange information needed for the service. In general, a Service Centre is responsible for one or more ITS aspects in one or more Areas.

Service Centres shall get the content, which is needed for a service, from ***Content Centres***. They shall cooperate with ***Traffic Management Centres*** to align their services with the goals of traffic managers.

If new software for a service is released (e.g. with new configurations of for an entirely new service), this is handled by Host Management Centres:

Host Management Centres support the remote management (download, configure…) of applications in ***Mobile Units*** or ***Roadside Units***. These units have the role of a so called *Client System* towards the Host Management Centres. In principle all CVIS hosts have an HMC, however some HMC's may provide very restricted remote management capabilities, for instance control centres owned by an OEM.

In Host Management Centres Software Packages (for applications) are maintained for different clients. Software versions and dependencies between packages are known at the Host Management Centre. They also allow Client Systems to "browse" for software packages, which are compatible with a specific Client's environment (e.g. available interfaces to sensors and actors, available depending packages). Host Management Centres shall also enable to restrict clients to access certified/authorised packages only. If a client downloads a new service in a software bundle, the corresponding Service Centre must be informed to prepare the service provision to the new client.

Host Management Centres may be run by various organisations and can exist independently in unlimited numbers in a CVIS network. Interaction is provided through standardised HMC interaction protocols.

Traffic Management Centres represent entities well known and existing in the ITS world. These Centres are operated by institutions/authorities responsible for managing the traffic on a specific road network in a dedicated area.

Traffic and Violation Control Centre are usually operated by the police forces and include e.g. like digital automated speed control networks already in existence in France today. CVIS allows easy to implement cross border enforcement applications due to its network structure.

Content Centres fulfil the function to provide any (raw/pre-processed) transportation relevant content to other centres. Typically ***Service Centres*** process content from different ***Content Centres*** and create value added services.

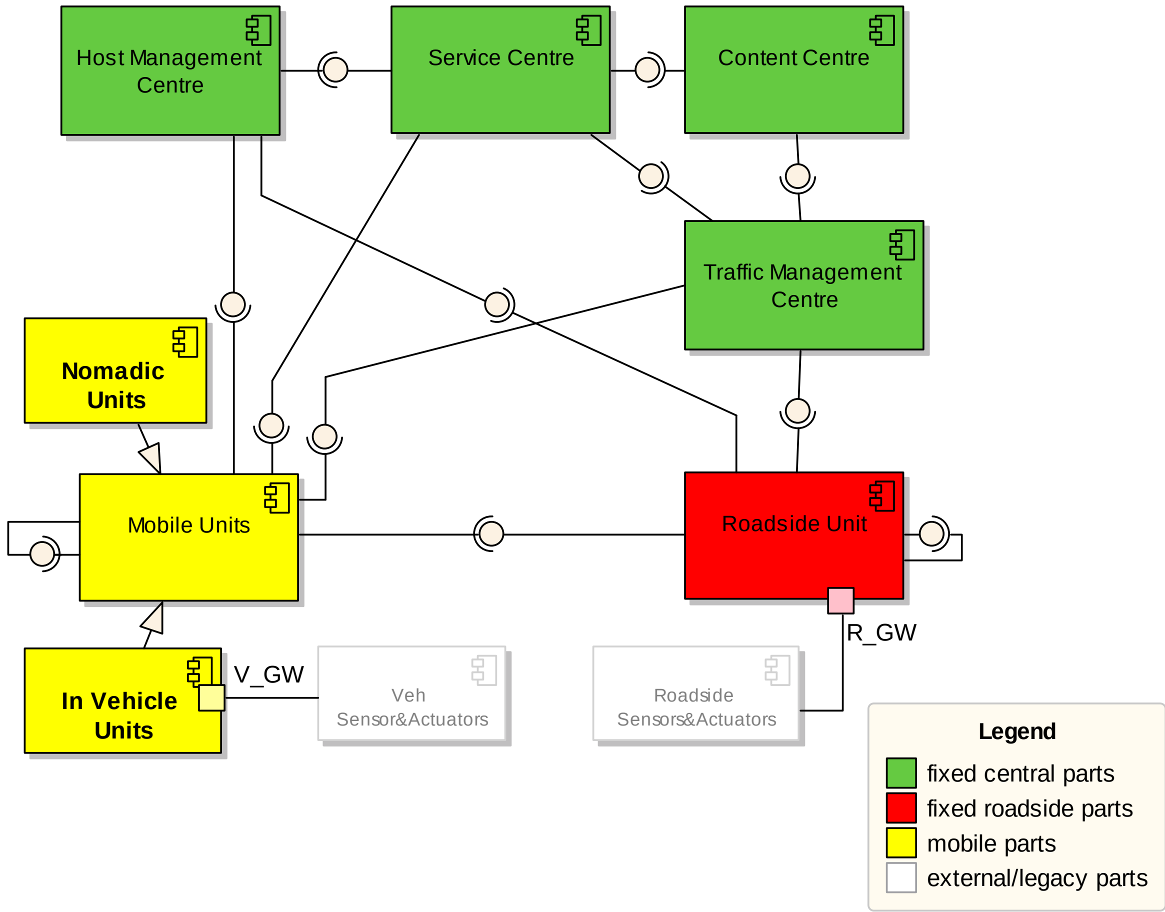

Figure 10: CVIS Entities and their relations as described

CVIS HOST internal structure

The internal structure of a CVIS host can be described as a set of applications.

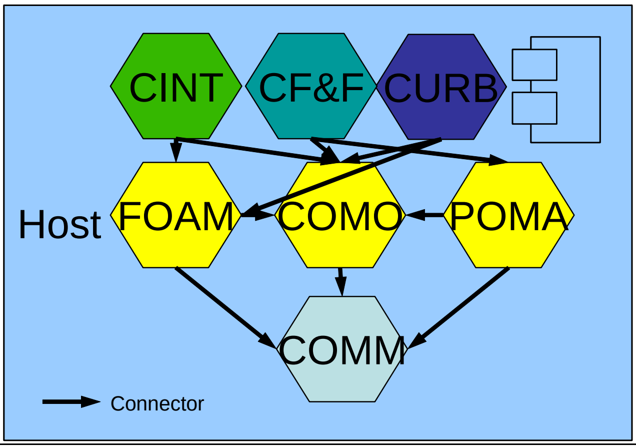	Applications inside the host will have dependencies. High level services will be based on more common ‘middleware’ alike applications. Basic core functions will be the basis for all higher applications.

Figure 11: CVIS Host internal structure

Cooperative inter-urban applications (CINT), Cooperative fleet and freight applications (CF&F) and Cooperative urban applications (CURB) is the application subprojects within CVIS. Framework for Open Applications (FOAM) is the middleware subproject within CVIS. The Cooperative Monitoring (COMO) and Positioning, maps & local referencing (POMA) CVIS subprojects develop common domain services such as monitoring, positioning and mapping. The Communication and networking (COMM) CVIS subproject provides the communication infrastructure (network, antennas, radios etc)

2.2. Functional Viewpoint

The Functional viewpoint defines the main architectural elements that deliver the CVIS functionality. The functional viewpoint documents show the main CVIS system's functional structure - including the key functional elements, their responsibilities, and the interactions between them. Taken together, this demonstrates how the system will perform the required functions as described in WP2 (CVIS Use Cases and System Requirements phase). In this document the high level functional architecture is presented. Details of the different parts and details of the provided interfaces and protocols are described in WP3 (CVIS System Specification and Architecture phase) deliverables.

2.2.1. CVIS overall functional structure and key functional elements

The CVIS overall functional structure and key functional elements are depicted in Figure 12.

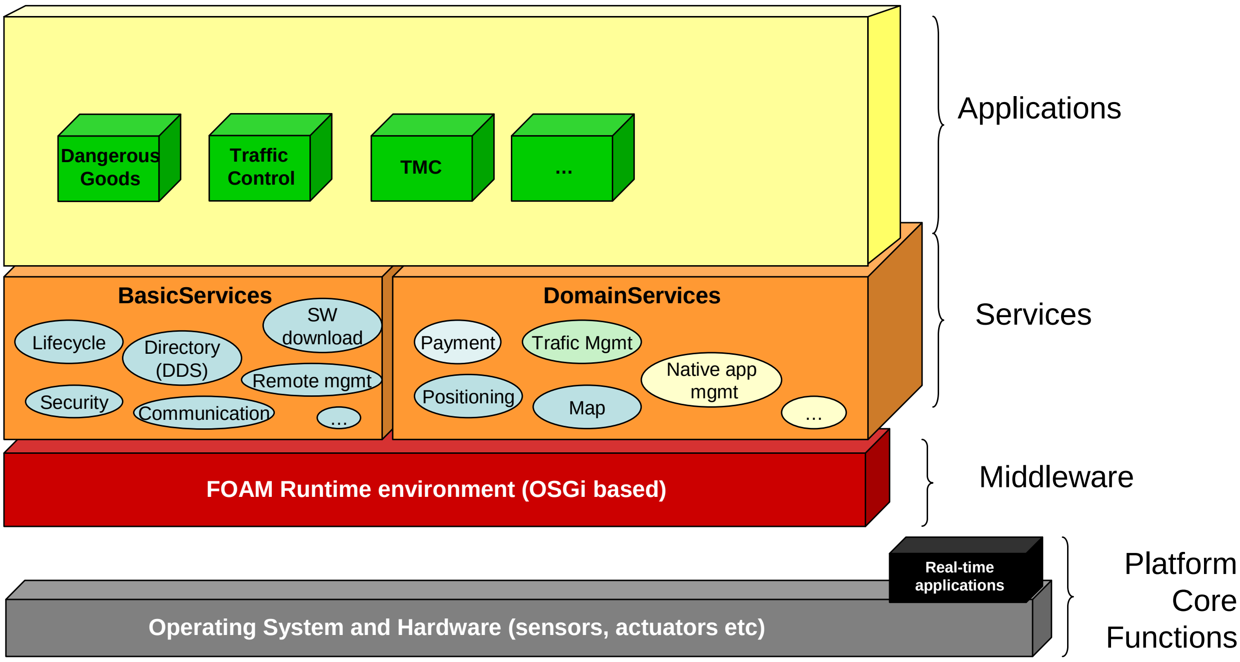

Figure 12: CVIS overall functional structure and key functional elements

The figure shows a layered architecture, where the bottom layer includes the core platform functions provided by the operating system and hardware. This layer includes the sensors actuators etc. CVIS based applications can run directly on top of this layer and use the platform core functions directly. For instance, real time applications where timeliness and delay constraints are critical may require accessing the platform core functions directly (as depicted at the bottom right of the figure). To interoperate with higher level CVIS services and applications, these native applications need to be integrated into the common CVIS environment for instance through bridges or wrappers. The interoperation should preferably be accomplished at the middleware level, however direct communication with particular services and applications can be supported on CVIS platforms.

The CVIS middleware level consists of the OSGi which is a Java-based service platform that can be remotely managed, and FOAM. A set of services are deployed, managed and provided through the middleware. There are two main categories of services:

i) *basic services*, which provide a set of basic CVIS middleware services such as; lifecycle management of services and applications, directory service for service publication and discovery, basic communication services, security services, remote management, and software download to enable dynamic downloading of software on CVIS hosts.

ii) *domain services*, which provide common domain level services such as; services supporting traffic management, positioning, map, and payment (e.g., for dynamic toll road payments).

Applications provide application specific services to end users (note: not all possible applications are shown in the figure above). Examples of CVIS applications are *Dangerous Goods, Traffic control and Traffic management centre* applications. Applications are deployed and managed using basic middleware services. Furthermore, the communication and access to other applications and services are provided through the middleware. From the application viewpoint the CVIS (inter)operation environment is revealed through a set of provided services (both basic and domain services). The services are provided through well defined interfaces and protocols. These interfaces and protocols will be described in detail in the D.3.3 CVIS Specification and Architecture deliverable.

2.2.2.CVIS (inter)operation basic services

CVIS systems are based on a peer to peer architecture where applications and services residing on distributed CVIS hosts interoperate as described in Section 1.5. The Distributed Directory Service (DDS) (categorized as a basic middleware service) have a central role in the CVIS peer to peer interoperation scenario. The DDS is a shared information infrastructure for locating, managing, administrating, and organizing common items and network resources, such as CVIS hosts, services, applications, sensors, actuators etc. A directory service defines the namespace for the network. A namespace in this context is the term that is used to hold one or more objects as named entries. Conversely, each object is uniquely identifiable by its name within its namespace. In CVIS we have identified four different categories or levels of directory service look up scenarios:

i) *Internet look up*. Applications need to locate services, applications and data providers on the Internet, for instance to download the DVD media player of choice.

ii) *Home network look up*. Applications running on a particular CVIS host need to look up services, applications and data available on the *home network*, where the home network refer for instance to the network controlled/owned by a vehicle OEM.

iii) *Local/visiting network look up*. Applications running on a particular CVIS host need to look up services, applications and data available in local networks, for instance services provided by a local traffic management centre.

iv) *Dynamic look up*: An Application component running on a particular CVIS host needs to form a (temporary) collaboration with components running on other CVIS hosts based on a set of specific selection criteria. Examples of specific selection criteria are:

a. Components in vehicles in a particular area;

b. Components in vehicles travelling via a particular junction;

c. Components in vehicles carrying (a particular class of) dangerous goods;

d. Components in road side systems in a particular area;

e. Components in roadside systems along a particular road segment.

The Service Type, representing the capabilities of the Application component, is also part of the selection criteria. These selections will be provided by the Distributed Directory Service, which is a mechanism offered by the CVIS middleware layer. The Distributed Directory Service will be operated by the Host Management Centre. On start up of a particular Application component on its CVIS host, this component will register itself with the DDS Server in the HMC, together with a description of its capabilities (Service type) and its attributes. It will also provide a periodic update of the actual values of these attributes. Typical attributes are location, speed, cargo etc. Another Application component, residing on a different CVIS host, can post a query to the DDS server containing a set of selection criteria, . The DDS will return a collection of references or handles (eg in the form of URIs) to Application components that conform to the selection criteria. The DDS will also return the actual values of the selection attributes for each component.

Please note that it is the responsibility of the Application Component to implement a design-time binding between the Service type and the corresponding API or protocol interface.

(An Application Component is a software component or bundle running on a particular Host. An Application is a group of collaborating Application Components, which are located on one or more Hosts. Different Applications can share (reuse) Application Components.)

A typical CVIS peer to peer interoperation scenario is shown in Figure 13.

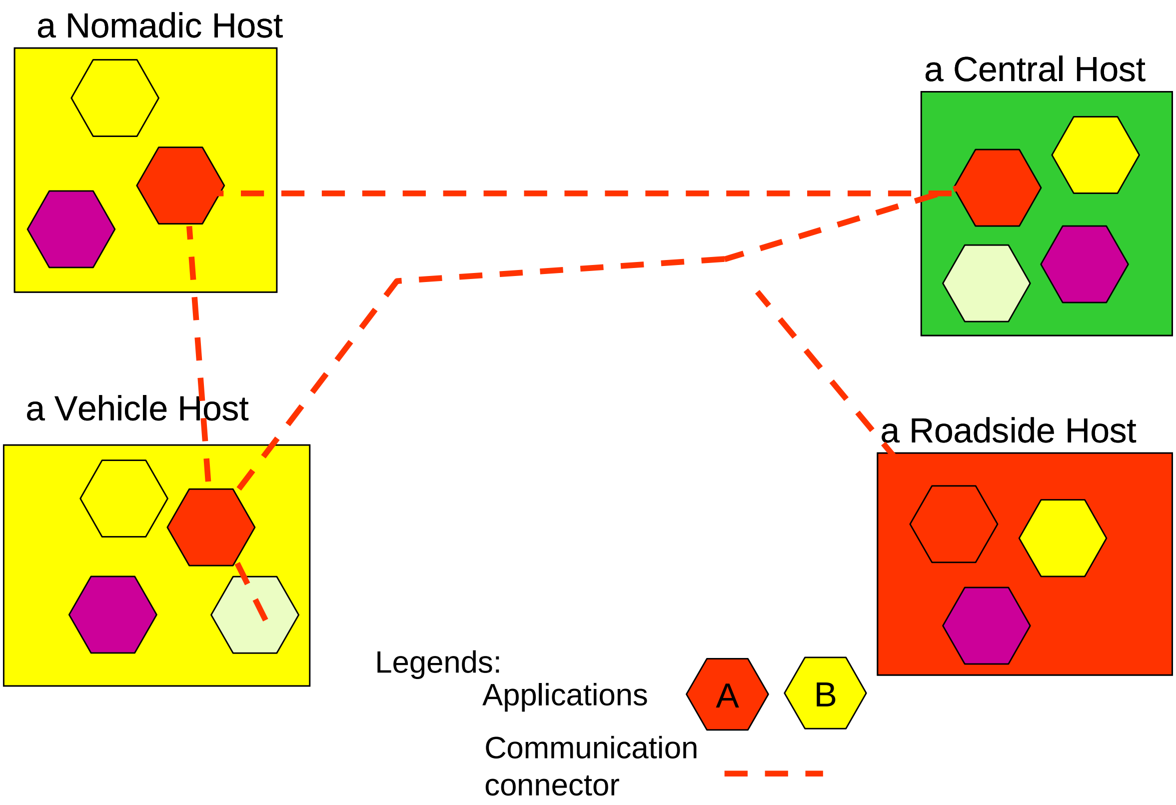

Figure 13: Peer to peer interoperation

The figure depicts that application of the same type residing on different CVIS hosts interoperate. The applications need to find their corresponding partners. The directory service provided by the middleware (i.e., the FOAM/COMM SPs in the CVIS project) offer addressing and contacting mechanisms, while the applications is responsible for identifying who to contact, (i.e., the *who* is processed as part of the application logic). Applications of different types can also interoperate as illustrated with the dotted line between to application types in the vehicle host.

A significant interoperation scenario in CVIS is when a host wants to communicate and collaborate with a CVIS host, but the particular CVIS host is not capable of participating in the interoperation because it lacks the appropriate service or application.

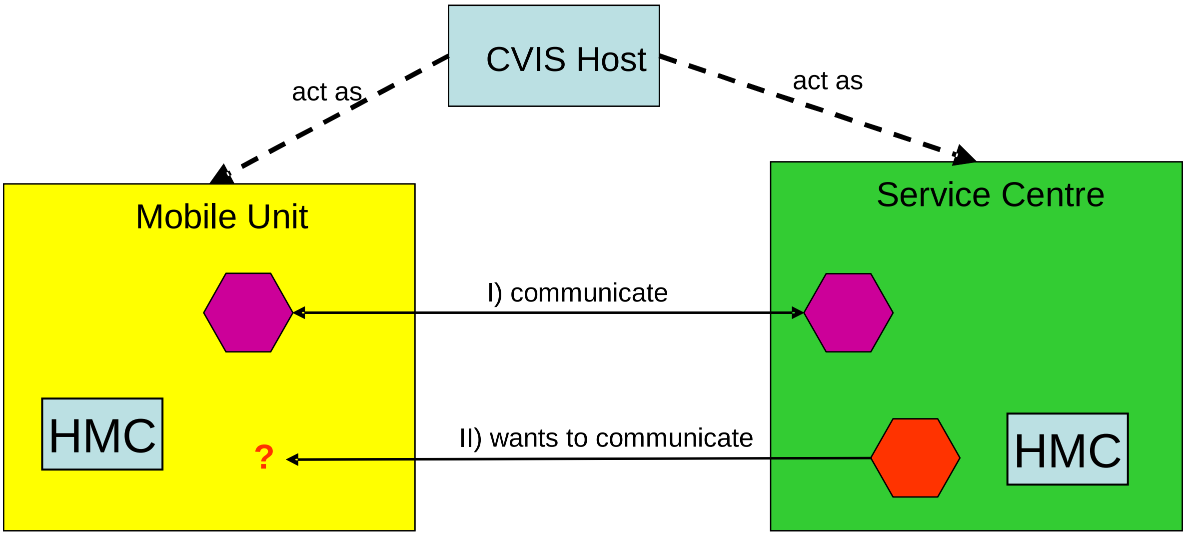

Figure 14: Appropriate application for interoperation is missing

Figure 14 shows that a Service Centre wants to communicate with a Mobile Unit, but the appropriate application is not deployed on the Mobile Unit. In CVIS this situation is resolved through dynamic software downloads. The Host Management Centres (HMC) of the respective hosts (HMC's are introduced in Section 1.5) is involved to accomplish dynamic download. The main responsibility of the HMC in this scenario is to ensure the required level of security. For instance to ensure that software downloads are performed from trusted service providers. This kind of scenario is depicted in Figure 15.

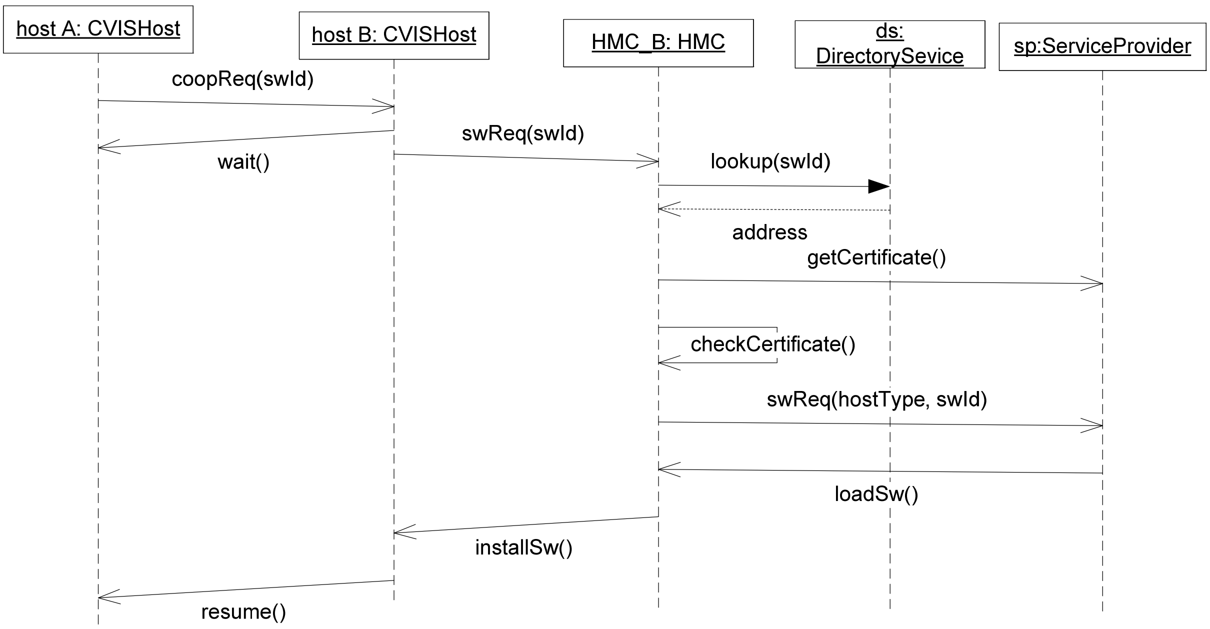

Figure 15: Software download scenario

In this scenario, host A provides a cooperation request to host B. Host B respond with a wait signal since the appropriate software is not installed. Then the HMS of host B takes the action to look up and download the appropriate software from a trusted service provider. In the scenario the HMC checks the certificate of the service provider before the software is downloaded and installed.

2.2.3. Domain services

The definition and description of the ITS domain functionality will be based on the work that has been done in the CVIS WP2 to define the Use Cases and System Requirements. It will also take into account the functionality that has been defined within the European ITS Framework Architecture, or “FRAME” as it is commonly known.

The CVIS cooperative technology has a big impact on most ITS functional areas as defined in FRAME. The applications considered in CVIS give a first indication of this impact. The following paragraphs give an outline of the FRAME Functional Areas that are likely to have an impact on the CVIS Functional Viewpoint.

Area 1 Provide Electronic Payment Facilities

Although there are no applications that directly address electronic payment, the communication and middleware functions of CVIS, (e.g. FOAM) support the introduction of many of the functionalities defined in the FRAME area into CVIS. In a later stage, the alignment between the concepts in the CVIS and Road Charging Interoperability (RCI) projects could be investigated

Area 2 Provide Safety and Emergency Facilities

Safety and emergency facilities are addressed in a number of the CF&F use cases and system requirements. FRAME sub-areas involved are: manage emergencies and manage stolen vehicle notification.

Area 3 Manage Traffic

As most of the CVIS use cases address mobility efficiency, traffic management plays a key role. COMO takes care of the collection and fusion of traffic data, which forms the basis of all traffic management. One of the CURB applications seeks to improve efficiency at an intersection while minimising the environmental impact; this is made possible by the augmented data about approaching vehicles and the direct feedback to road users. CURB and CINT also have applications that aim at the optimisation traffic on the network level. This includes driver awareness, traffic state assessment and demand management aspects.

Area 4 Manage Public Transport Operations

The CVIS cooperative approach is very well suited to public transport operations. The specific use cases defined within the project limit themselves to a dynamic lane usage application where infrastructure dedicated to public transport is dynamically shared with other road users. The functions predict vehicle indicators, monitor infrastructure and confer to vehicles are used.

Area 5 Provide Advanced Diver Assistance Systems

The CVIS communication architecture can form the basis for a lot of advanced driver assistance functionality. Basic FRAME functions that are supplied by CVIS are: collect road infrastructure data, provide traffic regulations, provide vehicle ID, provide vehicle position determination, prepare Floating Car data, provide ISA speed limits and current speed limit.

As the Services that CVIS proposes to provide contain some new concepts and ways of managing the road network and the vehicles using it, additional functionality will have to be defined within the CVIS Functional Viewpoint. This additional functionality will supplement and in some cases replace what is currently defined in FRAME.

2.3. Information Viewpoint

To facilitate interoperation in CVIS systems, interchange of information (data) between CVIS applications and services running on different CVIS hosts is required. CVIS systems are based on a peer to peer architecture where each CVIS host can play both the role of data consumer and data supplier. However, enabling interchange of information is not enough, the players also needs to *understand* the information that are interchanged. Furthermore, protocols for how to look up data suppliers and consumers is needed

In the ITS domain a large number of information models exist. Some of them are standards, some common in all of Europe, some only regional or national, some vendor specific, some are modern from an IT perspective (e.g. Datex-2), some antiquated. Existing standards and upcoming standards will be important sources when defining the information models in CVIS.

When designing cooperation between vehicles and road-side system the CVIS project needs to take into account that full harmonization of information models is not a realistic goal. Therefore the ability to cooperate should not be made dependent on this full harmonization. In addition to striving for such a harmonization CVIS needs also to put in place the mechanisms for systems to cooperate in the absence of single agreed upon universal ITS information model.

The mechanism envisioned within CVIS to achieve this is based upon the explicit and formal representation in a suitable formalism, (such as RDF), of the semantic and structural relationships between information models. Either between two models directly, or through a common agreed upon ontology.

CVIS needs to design systems for an open, heterogeneous and interoperable world, in which systems will interact that:

- are designed and implemented by different vendors,
- range from brand new to 10 years old,
- can be cheap and basic, or laden with add-on features,
- have to deal with different local regulations.

To support interoperability at the information level in this heterogeneous environment, CVIS deployments and their developers need to know about the information models that are used by other systems and their meanings. The CVIS approach to solve this is to support both structural and semantic meta-data, and to impose requirements on the availability of meta-data, both in the run-time and in the design-time environment.

In this section on the information viewpoint of the CVIS high level architecture we address three aspects

- Availability of meta-information (structural and semantic) during run-time (as opposed to design-time),
- a high level taxonomy of the information that appears in CVIS,
- dealing with partially overlapping and conflicting information models of cooperation systems.

2.3.1. Meta-information availability during run-time

Information used to be modelled first and foremost in design documentation, and only implicitly in the actual systems. Historically there has been a trend in IT system design to make the models that describe information explicitly available during run-time as well.

Examples of this trend are:

- relational databases that can be queried to retrieve the names of tables, columns, and various properties of columns such as whether they represent foreign keys (since the early '80's);
- XML messages that contain the field names in each message (since the 90's);
- RDF (Resource Description Framework) to model semantics of information on the Web (since 2000).

The popularity of XML is largely due to the presence of meta-information. Partitioning a message in elements no longer requires a priori knowledge about the message structure. Fields that are unknown can simply be ignored, while the remainder of the message can be processed. This greatly enhances the robustness of distributed heterogeneous systems.

CVIS aims to enable dynamic cooperation across Europe in a continually changing highly heterogeneous environment of vehicle systems, road-side systems and traffic centres. The design of CVIS will have to deal with the unavoidable variation in information models that will be used in different locations and countries and in different versions of systems. To accommodate this it is necessary to provide for intelligence in communication, and this in turn depends on the on-line availability of meta-information. Systems need to be able to query each other about their capabilities in order to determine whether and how to interact, how to interpret information and to monitor the execution of tasks. CVIS systems need to be able to on-line determine which information they can obtain, and what that information means.

These considerations are reflected in the following two high level design decisions for meta-information in CVIS:

1. All services and applications will make available during run-time the meta-information of the information that they can send or receive, and the services that they provide.
2. The published meta-information concerns both the structure and semantics of the information. (XML-Schema preferred candidate for structure, RDF preferred candidate for semantics).

Availability of meta-information enables a number of functions and interactions between systems that are essential for cooperation. Below a short classification is given of the most important types of meta-information related cooperation functions:

Publication	Make available the description of the capability of a service
Discovery:	Locate different services suitable for a given task
Selection:	Choose the most appropriate services among the available ones
Composition:	Combine services to achieve a goal
Mediation:	Solve mismatches (data, protocol, process) among combined services
Execution:	Invoke services following programmatic conventions
Monitoring:	Control the execution process of (a series of) services
Compensation:	Provide transactional support and undo or mitigate unwanted effects
Replacement:	Facilitate the substitution of services by equivalent ones

Table 1: Meta information related cooperation functions

2.3.2.A high level taxonomy of the information in CVIS

Modelling information is done by designing a class hierarchy. This is both a top-down and a bottom-up activity. Here we present a first version of the CVIS top level taxonomy, which will need to be revised and refined during the project. At the highest level we classify objects in the following classes, which describe:

- the IT environment itself (hardware, applications, services, network, gateways, etc.),
- physical objects
    - the (relatively) static part of the external world (roads, roadside equipment network),
    - the dynamic element of the external world (location of vehicles, traffic intensities, accidents),
- the users and organizations involved (drivers, operators, traffic managers)
- the information concepts that define their cooperation, such as contracts, transactions, reservations, notifications, subscriptions, alerts, plans, instructions, signatures, etc…

Figure 16 illustrates these elements in a diagram.

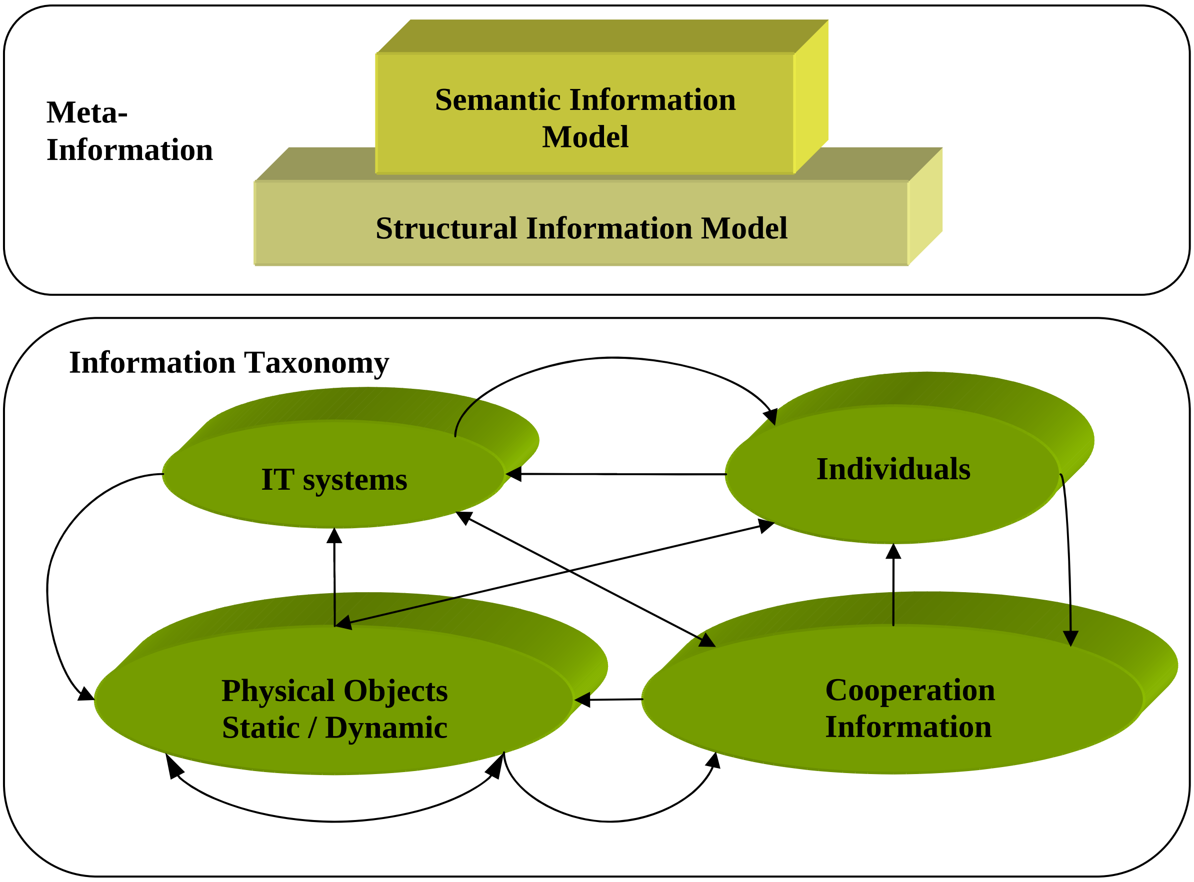

Figure 16: Informational object classes

Physical Objects

This category groups *static* and *dynamic* information about all tangible objects, such as roads, cars, traffic lights, etc. A 'map' is thought of as a visual representation of the position of a collection of physical objects. For the dynamic information, we think of the intensity of the traffic as a property of the road.

IT Systems

CVIS concerns a complex heterogeneous environment with many components. These components include hardware, services (e.g. traffic information services), network elements (gateways, servers, clients, routers), etc.

Individuals and organizations

Within CVIS many individuals and organizations play a role, and in a number of cases information about them has to appear explicitly. A typical example where both sides need to know each other identity is a financial transaction such as for toll and parking. Another situation is incident detection, where typically the authorities like to know who detected the incident (and thereby started the incident management activities).

Cooperation Information

In the ITS domain various forms of cooperation occur. Common patterns are:

- The interaction that starts with a permission request, which can be granted, declined, or declined with an alternative proposal,
- sharing information through pull- or push mechanisms,
- the interaction that starts with an instructions from the road operator, such as to free a lane, adjust speed, or to wait,
- the interaction that relates to paying for and possibly allocating a resource (toll for a road, a bridge, a zone, a parking place).

Classes that model the information used in these patterns will include:

- requests:
  - parking zone reservation requests,
  - green light request,
  - request to use lane (e.g. bus lane),
- permissions:
  - use of parking place, a particular lane, etc.
- contracts
  - payment for usage/allocating resources
- subscriptions
- notifications

2.4. Communication Viewpoint

The Communication Viewpoint considers the communication needs of the system entities in the Physical Viewpoint. Knowing the distribution of functionalities from the Physical Viewpoint and the information to be exchanged between entities, here the required data exchange is treated more in detail. Communication requirements like bandwidth, latency, and communication media types and characteristics such as peer to peer, broadcast, wireless etc. are to be specified.

The CVIS network is based on CALM, IPv6, and CALM Management standards enabling information exchange between geographically distributed mobile and fixed entities.

Information flows seamlessly between producing and consuming applications in distinct entities. Entities are the various processor units embedded in the vehicle, the roadside processor units, and clients and servers located in the Internet. The CALM protocol architecture (Continuous Air-interface Long and Medium range) enables the applications to communicate with each other on a continuous basis using whatever media is available in a particular system environment.

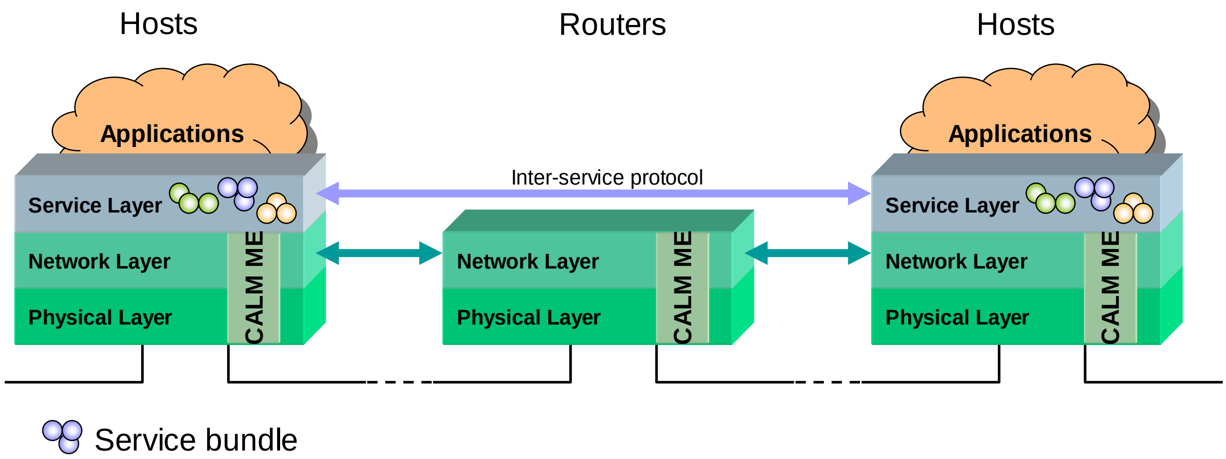

Figure 17: High Level CVIS Communication architecture

The communication design approach followed by CVIS is corresponding to the TCP/IP adaptation of the OSI's (Open Systems Interconnection) separation of actions in distinct layers. Each layer has a specific role. The separation into layers helps the design since each layer can be developed independently one from another. They interact by means of Service Access Points (SAPs), a specification that defines how parameters are passed between layers. In TCP/IP, the physical layer defines the communication mediums. The networking layer defines the path and rules on how processing units communicate with one another. The transport layer is in charge of the actual transmission of data grams, i.e. the emission, the reception and the retransmission if necessary. At the application layer, rules specific to the applications such as voice, video streaming, navigation information, etc. are defined.

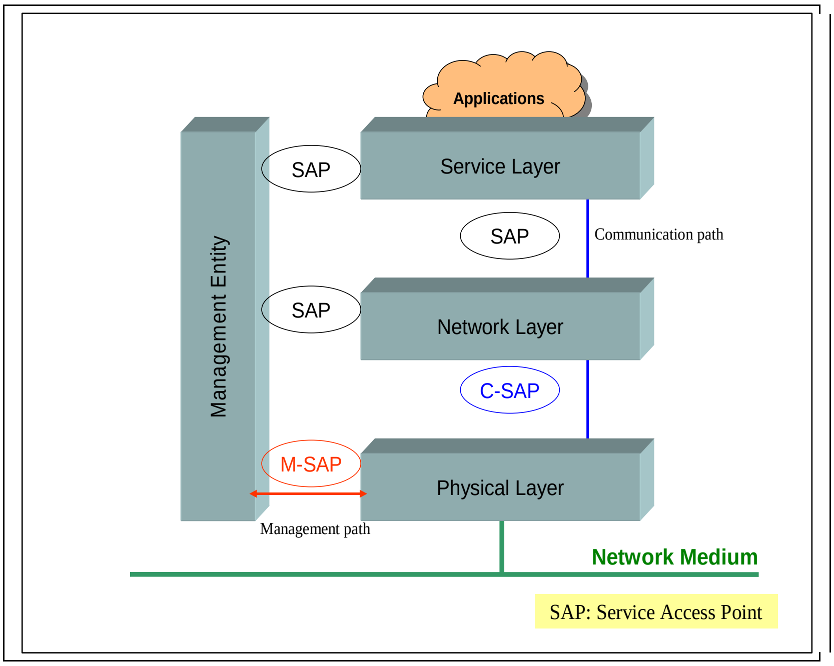

Figure 18: CALM protocol layers and Service Access Points.

CALM is based on IPv6 (Internet Protocol Version 6) which means that it is fully compatible with Internet services, while at the same time not being restricted by the addressing shortcomings of the current IPv4 protocols. CALM also allows non-IPv6 protocols. Important to remember is that the Internet Protocol plays the role of a convergence layer interfacing between all the possible physical medias and all the applications and the means to guarantee proper delivery of data grams between communication peers.

There will be a firewall, controlled by the automotive manufacturer, between the CALM equipment and any critical in-vehicle data bus. This is for the obvious reason that the vehicle manufacturer takes the responsibility for the safety critical aspects of the vehicle, and therefore must be the controller of data flows within the vehicle. To the manufacturer CALM simply provides efficient communications means to the outside world. CALM will support multiple types of applications and it is capable of supporting multiple types of communication media simultaneously. A similar firewall will be provided to protect roadside unit applications and data. (This is described further in the CVIS COMM D3.1 deliverable, Chapter about gateway)

CVIS will look at extending CALM and re-using existing protocols defined at the IETF (Internet Engineering Task Force, http://www.ietf.org). If needed, extensions may be developed and standardization of these extensions would be sought as much as possible within the IETF community for backward compatibility reasons between the CVIS communication architecture and other uses of the TCP/IP protocol suite.

The next figure illustrates different air interface protocols of CALM. CVIS implements the beacon, cellular and navigation communication media.

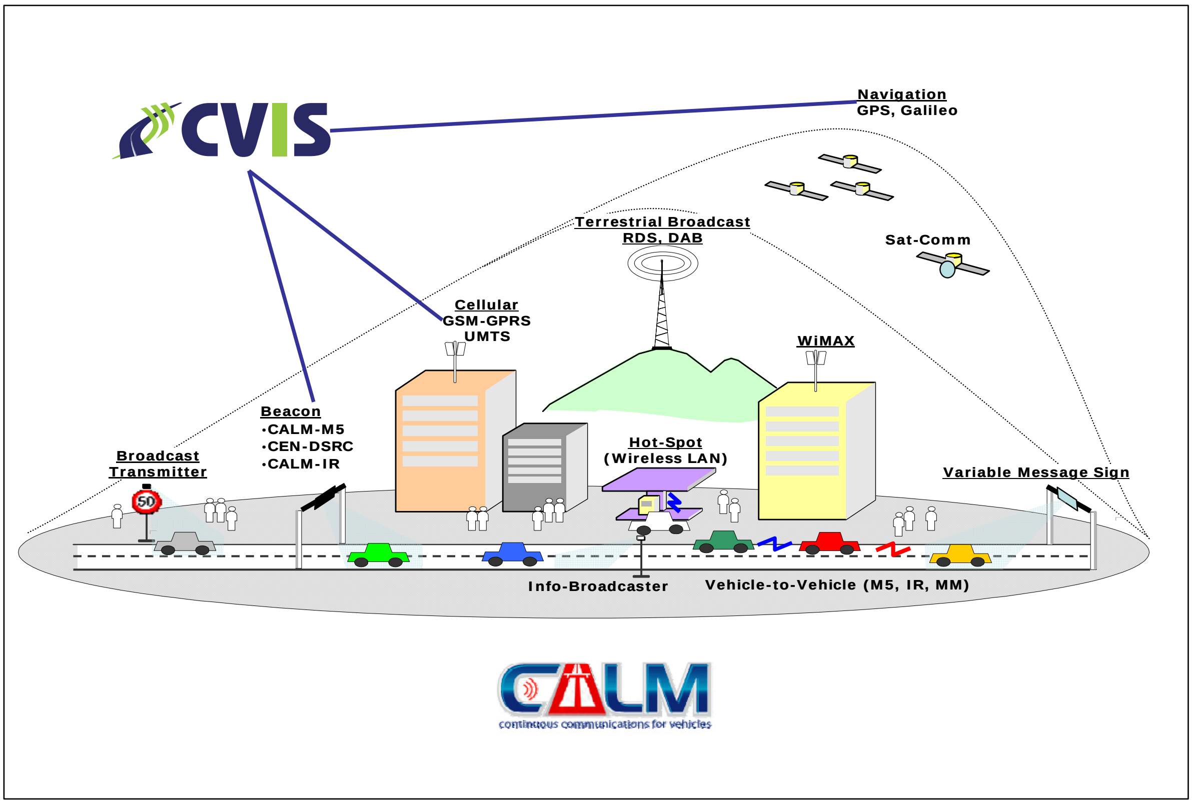

Figure 19: CALM Overview.

CALM includes the following communication modes:

Vehicle-Infrastructure: Multipoint communication parameters are automatically negotiated, and subsequent communication may be initiated by either roadside or vehicle

Infrastructure-Infrastructure: The communication system may also be used to link fixed points where traditional cabling is undesirable.

Vehicle-Vehicle: CVIS supports different communication approaches for vehicle-to-vehicle communication. On one hand, the low latency beaconing approach with the capability to broadcast safety critical data (position, speed, acceleration) to all vehicles in the current communication area. The beaconing messages are also known *Cooperative Awareness Messages (CAM)*. The second communication approach is ad-hoc networking, where vehicles are connected via dedicated communication links. Vehicles can exchange all kind of messages and the messages content is not restricted to safety critical data. This communication approach is the basis for multi-hop communication and geo-networking.

The CVIS architecture and implementation are flexible enough to support both vehicle-to-vehicle communication approaches. E.g., dedicated communication between two (or more) vehicles will be used within the GeoNet project and the CVIS architecture is designed to support easy integration of GeoNet results. The cooperation with GeoNet will be demonstrated by the end of CVIS project.

In addition, while add-hoc networking is limited to a small geographical area, CVIS stations can be seen as nodes on the IPv6 network. This communication mode allows vehicles to communicate with other vehicles regardless of the geographical position (they can on the different sides of the globe).

The next figure shows the high level communication architecture of CVIS. It illustrates the CVIS communication system functionality at a glance. However, when it comes to deploying commercially systems it is important to remember that although CVIS functionality appears in the figure as separate boxes, it will most likely in the future be incorporated into existing computing functions at the vehicle, roadside, or central systems.

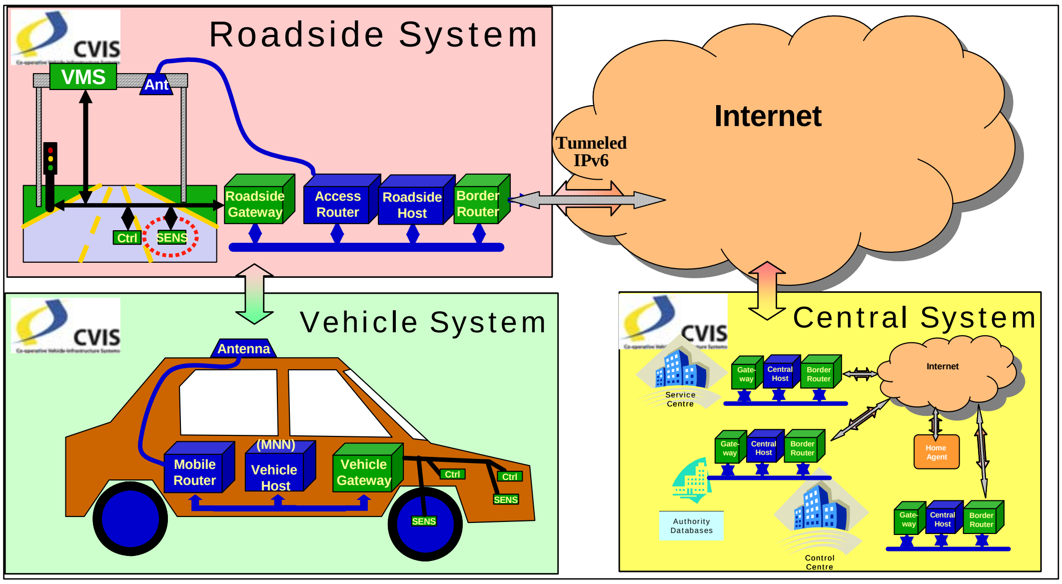

Figure 18 CVIS high level communication system architecture. See also document "D.CVIS.3.1 Reference Architecture"

The CVIS Host (Vehicle Host or Roadside Host) and the CVIS Routers (Mobile Router, Access Router and Border Router) will run an IPv6 stack. On the Roadside System, Roadside Hosts run applications offering CVIS services whereas the Access Router transfers these services to the vehicle and the Border Router connects the Roadside System to the Internet. On the Vehicle System, Vehicle Hosts run applications consuming IPv6 services whereas the role of maintaining the access to the Internet is devoted to the Mobile Router. The Mobile Router will either be connected to an Access Router on the Roadside, or an Access Router from some W-LAN or cellular operator. Servers in the Central System also run IPv6.

CVIS is not going to deploy a separate IPv6 network. It will indeed be part of the global Internet and will make use of any available access network to connect vehicles to the Internet (3G, W-LAN, infrared, etc).

The Roadside System, the Vehicle System and the Central System will be connected over the public Internet either using native IPv6 or will be tunnelled in IPv4 networks. Strong security and VPNs may be established between CVIS entities whenever needed.

Mobility is a key aspect to cooperative systems. There are two main functions that are needed for mobility to function:

- NEMO/MONAMIv6 is needed for mobile networks with mobile hosts "hidden" behind mobile routers in order to maintain Internet connectivity through multiple medias. This is mostly handled by Home Agents and corresponding functions inside the mobile routers
- The other function is related to Service Discovery while moving. Available services

will be discovered dynamically through some mechanism residing at the network layer. Many services may have a very local scope, and may often be time variant or highly time critical. They may be specific to media used or available, and even be offered from mobile entities. Other services have a global Internet scope.

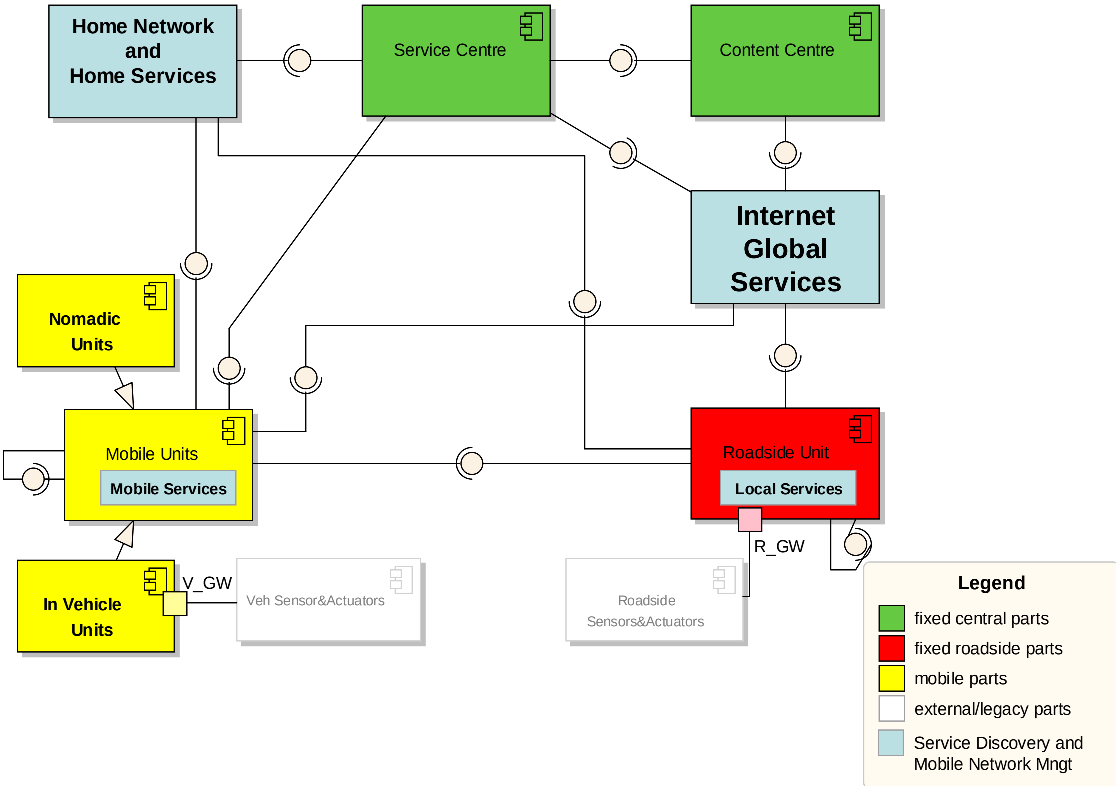

Figure 20: CVIS Entities – Network Management

2.5. Organisational Viewpoint

The Organisational Viewpoint explains the organisational aspects of the future Cooperative Systems world. The Cooperative System deployments should result in new business opportunities and thus new organisational structures. Today most of the ITS deployments are vertically organised. One organisation is responsible for the complete chain from road side sensor up to central system. This has the following consequences:

- Clear situation with respect to ownership and responsibilities of both the information and the data.
- Sub optimal use of the systems and information as they are constrained to be used by units associated with one single organisation only (typically the OEM).
- Often hard geographical boundaries resulting in lack of continuity of services between the service providers.
- Closed systems, each operator has it's own proprietary systems not developed with interoperability as a design goal with proprietary interfaces between the layers.
- It is surely not possible to identify "the" organisational architecture of the Cooperative Systems world. Indeed, it is a basic principle of the CVIS view of Cooperative Systems that the architecture is open and that specific systems can be configured to suit local arrangements. There should be room for new organisations offering services to cooperative users and for new relationships between existing organisations.

2.5.1. Central organisations

Looking at the Organisational viewpoint there is a need for some central organisation/bodies in the Cooperative Systems world. One of the most important tasks for these bodies would be to ensure inter(operability). Such body/bodies would handle following tasks:

- Govern minimum set of basic technology (e.g. Application/Services/Models)
- Govern Cooperative Systems "community" rules (e.g. how different players interact with each other)
- Govern the operation of the Cooperative Systems world (Publish/exchange of data models & interface definitions)
- Govern security (issue Keys, Certificates, etc)

Furthermore, it is important to mention that there will be no special "Cooperative Systems network organisation". This is because the Cooperative Systems will run on global IPv6 network, which itself will have certain governing organisations.

2.5.2. Organisational configurations

The Cooperative Systems network will allow both open and restricted configuration of applications and services. Depending on implementation strategies one organisation might choose to allow open access to its entities, while another organisation might want to restrict the access to their entities, e.g. due to safety and security issues.

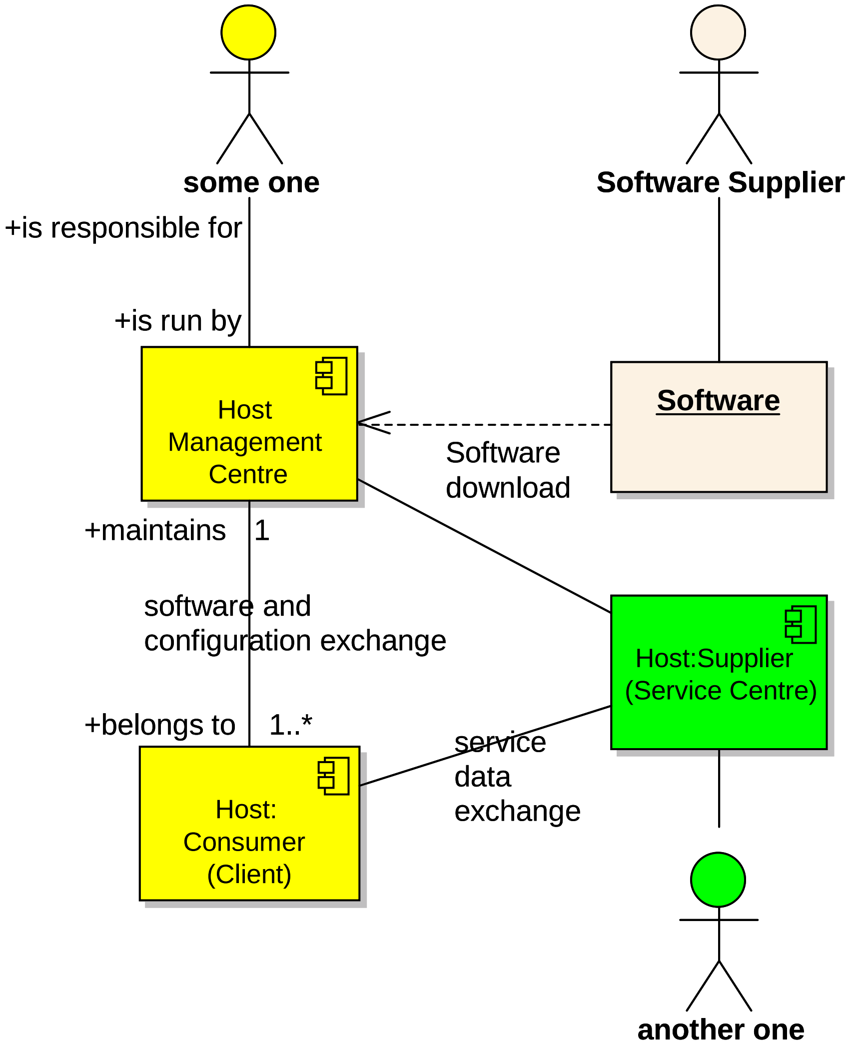

Figure 21: CVIS Organisational example – Service download

Figure 21 presents an example of organisational architecture for service download use case. As mentioned previously, each CVIS host will "belong" to only one Host Management Centre. A service can be downloaded from an "in-house" software supplier. However, it can also be downloaded from an external service centre, as long as that host has agreement with the responsible Host Management Centre. This agreement could be realised in a form of certificate.

Since the Distributed Directory Service is also located at, and operated by the Host Management Centre, the contract/agreement between a Service Centre and a Host Management Centre concerns at least two aspects:

- Deployment and provisioning of particular Service Application components, together with the access rights for each components;
- Access right for usage of the Distributed Directory Service.

This means that there could be an e.g. Volvo Host Management Centre providing services and applications for their customers. However, even other service centres could download services or applications thus when a Volvo driver drives to a new country there could be a service that is automatically downloaded to that Volvo vehicles host computer from a local Service centre.

3. Solution Design Choices and Guidelines

The Solution Design Viewpoint makes specific choices in Hardware, Software and Communication solutions. It can be divided in two Sub-Viewpoints: a ***Software Engineering Sub-Viewpoint*** and a ***Technology Sub-Viewpoint***.

More detailed descriptions about this Viewpoint issues have been published in the CVIS D.3.1 Reference Architecture. Here a selection of most important aspects is provided in summary.

3.1.1. The Software Engineering Sub-Viewpoint

The Software Engineering Sub-Viewpoint selects software development technologies and frameworks. In CVIS major selections are

System Specification

UML 2.1.1 is used as the modelling language for the system, and *Enterprise Architect* 6.5 as the UML design tool.

Software Development

Non-native host applications are developed using JAVA OSGI Service Development Kit (SDK). They will use additional functionality provided by FOAM (Framework for Open Application Management sub-project).

FOAM is an end-to-end application framework and run-time execution environment, connecting in-vehicle systems, roadside infrastructure and back-end infrastructure.

Native applications have to be linked to the rest of the CVIS Architecture via specific Interfaces to the COMM/FOAM layer. Applications can be composed of both – native and JAVA-OSGI parts.

The following figure illustrates, that applications shall be linked by a middleware layer provided by FOAM. Through this middleware layer applications can get access to

- The CVIS communication
- Local sensors and actors
- Other (neighbouring) applications on the same host or on another host.

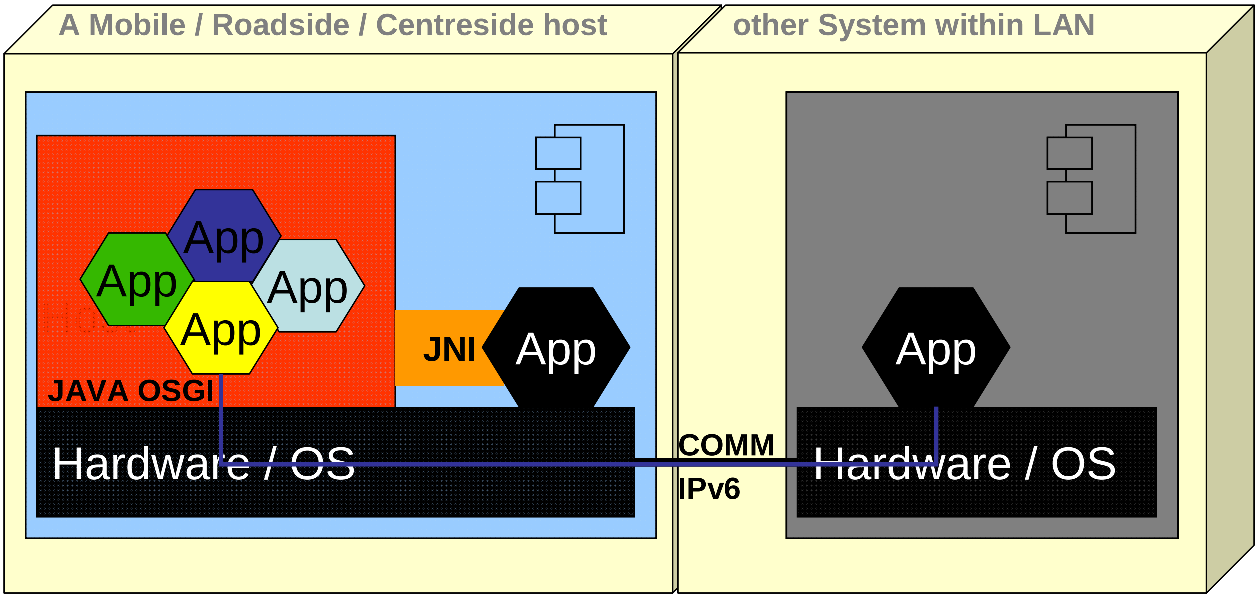

Figure 22: CVIS platform architecture on a host

FOAM summary

FOAM substitutes existing standalone approach and hierarchical approach with direct cooperative management between vehicles as well as between vehicles and local infrastructure, this us due to the growing complexity in transport management systems.

FOAM is an open (i.e. non-proprietary) framework architecture, cf. ETSI open systems strategy, www.etsi.org.

FOAM supports CALM vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I) and infrastructure to-infrastructure (I2I) information and transaction exchange. This is achieved by providing APIs to the communication system developments on one hand and offering the functionality of this development as JAVA OSGI services to any other applications on the other hand.

FOAM supports and assists application development activities in CVIS, and supplies client side run time environments for both, rich java clients as well as embedded native clients.

FOAM implements unified application interfaces for applications, sensors and actuators.

FOAM will create a so-called binding to specific technologies in order to create a fully functional system. The prime candidate for a client binding is Java / OSGi running on top of Unix operating system, most likely NetBSD.

3.1.2. The CVIS Development Stages

The α stage is the development phase where one sub-technology is specified and developed to the stage where it is ready to be integrated into a beta platform. Alpha sub-technologies consist of SW as well as HW system components.

The β stage is the phase where the alpha sub-technologies are integrated into running platforms. There are two platforms: the Router and the Host. Platforms are inherently generic at this stage, i.e. not configured for vehicle, roadside or central usage.

The γ stage is when the configuration of the platforms is done. For the roadside this implies packing router and host platforms into a portable roadside unit, adding local test-site specific services/applications, and qualifying gateways into existing roadside sensor networks. For vehicles this means integrating the beta units into the test vehicles, connecting to vehicle gateways, and adding specific software.

3.1.3. The Technology Sub-Viewpoint

The Technology Sub-Viewpoint chooses the technology for system distribution such as specific hardware and physical interfaces. The highest level of decomposition of the physical infrastructure in CVIS is

- **The CVIS fixed part** - comprising the following devices:
  - Road side equipment (RSE)
  - Local equipment
  - Central equipment

- **The CVIS enabled in-vehicle devices** - comprising the following devices:
  - Human Machine Interface (HMI) layer
  - Data Layer
  - Application Processing layer
  - Bearer independent layer
  - Physical layer

- **The CVIS enabled nomadic devices**
  - PDA's, cell phones, organisers, and other hand held devices that are CVIS enabled

At the CVIS fixed part decomposition level the following platform components are to be used:

The Router platform. The router platform is a platform computer that includes the following air interfaces: CALM M5, DSRC, IR, Galileo/GPS and 2G/3G. This computer will include be fully based on the CALM architecture, and develop a CALM management stack. A router can be further personalized to operate as a mobile router in a vehicle, or an access router in the roadside.

The Host platform. The host platform is a platform computer that includes a set of middleware and services that offers a fast Java virtual machine with a very rich API including positioning, sensor data, HMI, maplets, communications quality of service, remote management of applications and so on.

More details can be found in D.CVIS.3.1 Reference Architecture.

3.2. The Enterprise Security View

In a Cooperative System security and all related issues (e.g. privacy, integrity,…) play a vital role. CVIS has no special sub project for development of respective mechanisms and tools. Therefore exploitation of state of the art technology and links to other projects (e.g. [SEVECOM], [GST-SEC], [GST-CERTECS]) are needed. Organisational considerations and user needs require the core technologies and applications to support and provide basic mechanisms for:

Identification of a single user

To avoid numerous authentications for a user various applications shall base their user authentication to a central "single sign on" service.

Identification of a single application / Version

To enable remote management, software applications and their versions shall have an identifier.

Access control

Only applications with permission shall be able to use functions of other applications. This shall enable to restrict access to services offered to authenticated other applications only.

Anonymous data exchange

To safeguard privacy, any data that allows tracing individual identity and behaviour needs to be avoided.

For example anonymous, random identifiers shall be used for road traffic analysis necessary data acquisition through mobile hosts. Any system management internal data about location (e.g. network locations and addresses of a host) must be used by system management internal processes only.

4. Conclusions

The presented High Level Architecture is a result of the ongoing specification and architecture work in the CVIS project. It presents the CVIS conceptual ideas on how the physical, functional, information, communication and organisational architectures of the Cooperative systems should look like.

Closely related to this deliverable is the D.CVIS.3.3 Architecture and System Specifications, which provides detailed information about CVIS overall architecture and its components (including information such as component functionalities and interfaces). The High-Level Architecture has been a useful step in developing D.CVIS.3.3, which can be found on www.cvisproject.org

The document (together with D.CVIS.3.3) was used in the architecture harmonisation process between the CVIS subprojects. Furthermore it has helped in harmonising the architectures with other Cooperative System IP's e.g. SAFESPOT and COOPERS. It must be mentioned that amendments to this document are expected based on the continued architecture work in the subprojects of the CVIS and the continued harmonisation work together with other Cooperative Systems IP's.

The CVIS architecture provides a solid ground for vehicle-infrastructure and vehicle-vehicle cooperation. Application developers can count on standard functionalities such as the enhanced positioning, life-cycle management and communication capabilities among all CVIS hosts. Functionalities like the host management centre allow safe and reliable software management functionalities which are essential for tomorrow's systems. Software can be supplied on a broad geographical basis, on basis of brand or type or singularly to just one entity. These functionalities are provided to any CVIS entity by standardised interfaces, regardless if it is a vehicle, a roadside units or even a centre. This architecture also provides a significant boost for innovation due the flexibility it provides to future applications bringing Europe one big step closer to seamless services and wide spread use of future ITS systems. It allows to cover the individual needs of each driver with an unlimited range of applications which can and will be tailored to the specific user needs regardless of geographical location and specific requirements. If enough users require an application it will emerge sooner or later - the CVIS architecture and prototype platform support them optimally.

5. References

[RM-ODP]	ISO/IEC 10746-3 Open Distributed Processing - Reference Model http://www.rm-odp.net/
[FRAME]	http://www.frame-online.net
[TOGAF]	The Open Group Architecture Framework http://www.opengroup.org/architecture/togaf8-doc/arch/toc.html
[MDA]	Model Driven Architecture http://www.omg.org/mda/
[CALM HB]	CALM Handbook
[IETF]	http://www.ietf.org
[GST]	http://www.gstproject.org/gst/arch/
[SEVECOM]	http://www.sevecom.org/
CVIS	http://www.cvisproject.org/