C-MobILE

Accelerating C-ITS Mobility Innovation and deployment in Europe

D3.1: Reference Architecture

Status	Final
Main authors	Yanja Daisuren (TU/e) Priyanka Karkhanis (TU/e) Deniz Kadiogullari (htw saar) Manuel Fünfrocken (htw saar)
Work Package	WP3 – Architecture and systems design
Task	T3.1 – Reference architecture
Dissemination level	Public
Issue date	30/11/2017

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 723311

Project coordination

Marcos Pillado Quintas

Applus+ IDIADA

PO Box 20 Santa Oliva

43710, L'Albornar, Tarragona

Spain

Telephone: +34 977 18 93 60

E-mail: marcos.pillado@idiada.com

Revision and history sheet

Version	Date	Authors	Summary of changes	Status
0.1	31/08/2017	Yanja Dajsuren (TU/e)	Deliverable outline	Prepared/ Authorized
0.1	04/10/2017	Yanja Dajsuren (TU/e)	Architecture methodology and framework	Contributed
0.2	09/10/2017	Areti Kotsi (CERTH)	Description of the related architectures and projects for COMPASS 4D, SCOOP@F phase2, InterCor, C-Roads, VRUITS, C-ITS Corridor, Eco-AT, CODECS, AUTOCITS, CITRUS, PROSPECT, XCYCLE, SOLRED	Contributed
0.2	09/10/2017	Manuel Fünfrocken (HTW)	C-MobILE at a glance, Document structure, Intended audience	Contributed
0.3	09/10/2017	Evangelos Mitsakis (CERTH)	Description of the NordicWay architecture	Contributed
0.3	13/10/2017	Yanja Dajsuren (TU/e)	Architecture framework definition	Contributed
0.4	16/10/2017	Manuel Fünfrocken (HTW)	Reviewed	Reviewed
0.4	18/10/2017	Robbin Blokpoel (PT)	Stakeholders and their concerns	Contributed
0.4	18/10/2017	Janki Dodiya (DLR)	Description of the MObiNET architecture	Contributed
0.4	19/10/2017	Geert Vanstraelen (MACQ)	Description of the DITCM, Executive Summary, C-Roads, CITRUS, Talkin Traffic	Contributed
0.4	31/10/2017	Manos Nestoras (TT)	Review of Chapter 1, AUTOCITS and CONVERGE	Reviewed
0.5	13/11/2017	Deniz Kadiogullari (HTW)	Merged separate sections.	Prepared
0.5	14/11/2017	Priyanka Karkhanis (TU/e)	Architectural models for the context, functional, communication, information, physical viewpoints	Prepared / Authorized
0.5	05/11/2017	Yanja Dajsuren (TU/e) Priyanka Karkhanis (TU/e)	Revised architecture framework definitions and models	Contributed
0.5	08/11/2017	Manuel Fünfrocken (HTW)	C-MobILE architecture methodology, C-MobILE architecture concepts, reference architecure concepts, concrete architecture concepts, implementation architecture concepts	Contributed
0.5	08/11/2017	Estibaliz Barañano (KTAS)	Description of the iKoPa and NordicWay	Reviewed
0.5	08/11/2017	Raúl Ferrandez (IDIADA)	Description of the C-The-Difference and Eco-AT and provided Summary	Reviewed
0.6	09/11/2017	Manuel Fünfrocken (HTW)	Description of the CONVERGE	Contributed
0.6	09/11/2017	Vasilis Mizaras (SWARCO)	Description of the US ITS	Reviewed
0.6	09/11/2017	Mark van den Brand (TU/e)	Reviewed and contributed	Reviewed/ Contributed
0.6	09/11/2017	Mark van den Brand (TU/e) Ricard Molins (IDIADA)	Reviewed	Reviewed
0.7	13/11/2017	Priyanka Karkhanis (TU/e)	Revised architectural models for the C-MobILE architectural views	Contributed
0.7	13/11/2017	Pauline Baudens (TOMTOM)	Reviewed Chapter 2 and descriptions of XCYLCE, CITRUS, and MOBiNET	Reviewed / Contributed
0.7	14/11/2017	Jaap Vreeswijk (MAP)	Reviewed of SCOOP@F phase 2, DITCM	Reviewed / Contributed

0.7	14/11/2017	Andre Perpey (GLS)	Reviewed of InterCor, Talking Traffic	Reviewed
0.7	06/11/2017	Charlotte Fléchon (PTV)	Reviewed COMPASS4D, C-Roads, PROSPECT, Chapter 7	Reviewed
0.7	29/11/2017	Rik Eshuis (TU/e)	Reviewed	Reviewed
0.7	29/11/2017	Paul Grefen (TU/e)	Reviewed	Reviewed
0.8	29/11/2017	Oktay Türetken (TU/e) Deniz Kadiogullari (HTW)	Reviewed and contributed	Reviewed / Contributed
0.9	29/11/2017	Geert Vanstraelen (MACQ) Yanja Dajsuren (TU/e)	Updated the executive summary	Reviewed/ Contributed
1.0	29/11/2017	Priyanka Karkhanis (TU/e) Oktay Türetken (TU/e) Yanja Dajsuren (TU/e)	Contributed and reviewed	Contributed / Reviewed
1.0	30/11/2017	Priyanka Karkhanis (TU/e) Deniz Kadiogullari (HTW) Oktay Türetken (TU/e) Yanja Dajsuren (TU/e)	Updated and reviewed	Reviewed / Authorized
1.1	30/11/2017	Marcos Pillado (Applus+ IDIADA)	Review	Approved

Table of Contents

Figures

Tables

Abbreviations

Abbreviation	Definition
3G	3rd generation of mobile telecommunications technology
4G	4th generation of mobile telecommunications technology
Archimate©	An open and independent modelling language for enterprise Architecture
ADAS	Advanced Driver Assisted Systems
AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
ASN.1	Abstract Syntax Notation No.1
AU	Application Unit (in IRS or IVS)
BDD	Block Definition Diagram
BLE	Bluetooth Low Energy
BSA	Basic Set of Applications
BTP	Basic Transport Protocol
B2X/X2B	Business-to-X,X=Business (B), Consumer (C) or Government (G)
BO	Back Office
C2C	Car-to-Car respectively vehicle-to-vehicle Communication; information exchange between vehicles using ETSI G5; the term is now replaced by „V2V“
C2I	Car-to-Infrastructure respectively vehicle-to-infrastructure Communication; information exchange between vehicles and infrastructure using ETSI G5; the term is now replaced by „V2I“
C2X/Car2X	Car-2-X Communication (generic term: X is either infrastructure or car); including communication between vehicles as well as between vehicles and infrastructure; the term is now replaced by „V2x“
CA	Certification Authority
C-AD	Connected-Automated Driving
CACC	Cooperative Adaptive Cruise Control
CAN bus	Controller Area Network bus
CAM	Co-operative Awareness Message
CBA	Cost Benefit Analysis
CBF	Contention Based Forwarding (Geo Broadcast)
CCU	Communication and Control Unit (in IRS or IVS)
CEN	Comité Européen de Normalisation; English: European Committee for Standardization
CIS	Central ITS Subsystem
C-ITS	Cooperative Intelligent Transport Systems
CMD	Cooperative Mobility Device
CP	Communication Provider
CONVERGE	Communication Network Vehicle Road Global Extension
C-MobILE	Accelerating C-ITS Mobility Innovation and depLoyment in Europe
DATEXII	Second generation of a standard for the exchange of traffic related data which can be used by all actors in the traffic sector (www.datex2.eu).
DCC	Distributed Congestion Control
DENM	Decentralized Environmental Notification Message
DIS	Data Interchange Server
DITCM	Dutch ITS Test site for Cooperative Mobility
DP	Data Provider
DoA	Description of action
EC	European Commission
EM	Exploitation Manager
EOBD	European On-Board Diagnostics
ETSI	European Telecommunications Standards Institute
ETSI G5	G5 (Wi-Fi) communication standard for vehicular communication, standardized by ETSI in standard ETSI ES 202 663
ETSI ITS G5	ETSI term for „IEEE 802.11p”
ETSI ITS-G5A	ITS Frequency band 5,875GHz to 5,905GHz dedicated for safety related applications
EV	Electrical Vehicle
FESTA	FESTA (Field opErational teSt supporT Action) is a shared methodological framework to carry out field operation tests (FOTs) and may serve as a handbook for the deployment of C-ITS service bundles and their impact assessment.

FOT	Field Operational Test. Is a study undertaken to evaluate a function, or functions, under normal operating conditions in environments typically encountered by the host vehicle(s) using quasi-experimental methods.
GA	Grant Agreement
GA	General Assembly
GLOSA	Green Light Optimal Speed Advisory
GN	GeoNetworking
GPS	Global Positioning System
HMI	Human Machine Interface
IA	Innovation Action
IBD	Internal Block Diagram
ICS	ITS Central Station
IEEE	Institute of Electrical and Electronics Engineers
IIS	Internet Information System
IM	Innovation Manager
INS	Intersection Safety
IP(v4/6)	Internet Protocol, version 4 or 6
IPTS	Intelligent Pedestrians Traffic Signal
IRP	Intermodal Route Planner
ISO	International Organization for Standardization
IRS	ITS Roadside Station
ITS	Intelligent Transport System
ITS-S	Intelligent Transport System Station
ITSC	Intelligent Transport System Communications
ITS-G5	ITS at 5 GHz frequency band
IVS	ITS Vehicle Station respectively any Mobile ITS Node (e.g. Smartphone)
IVERA	Management protocol for traffic light controllers in the Netherlands
KoM	Kick-off meeting
KPI	Key Performance Indicators
LAN	Local Area Network
LDM	Local Dynamic Map
LHW	Local Hazard Warning
LOS	Level of Service
LTE	Long Term Evolution (4G)
MAC	Media Access Control
MIB	Management Information Base
MNO	Mobile Network Operator
MoM	Minutes of meeting
OBD	On-Board Diagnostics
OBU	On-board Unit
OEM	Original Equipment Manufacturer
OSI	Open Systems Interconnection
PC	Project Coordinator
PHY	Physical layer
PI	Performance Indicators
PID	Personal Information Devices (e.g. smart phone)
PKI	Public Key Infrastructure
PS	Pilot Site
PSL	Pilot Site Leader
PTW	Powered Two Wheelers
PVD	Probe Vehicle Data
pWLAN	Acronym for ITS G5 (IEEE 802.11p)
QoS	Quality of Service
RAS	Roadside Actuation System
RDS - TMC	Radio Data System – Traffic Message Channel
RHW	Road Hazard Warning
RIS	Roadside Intelligent transport Sub-system
RLVW	Red Light Violation Warning
RSS	Roadside Sensor System
RSU	Roadside Unit
RWW	Roadworks Warning
SC	Steering Committee

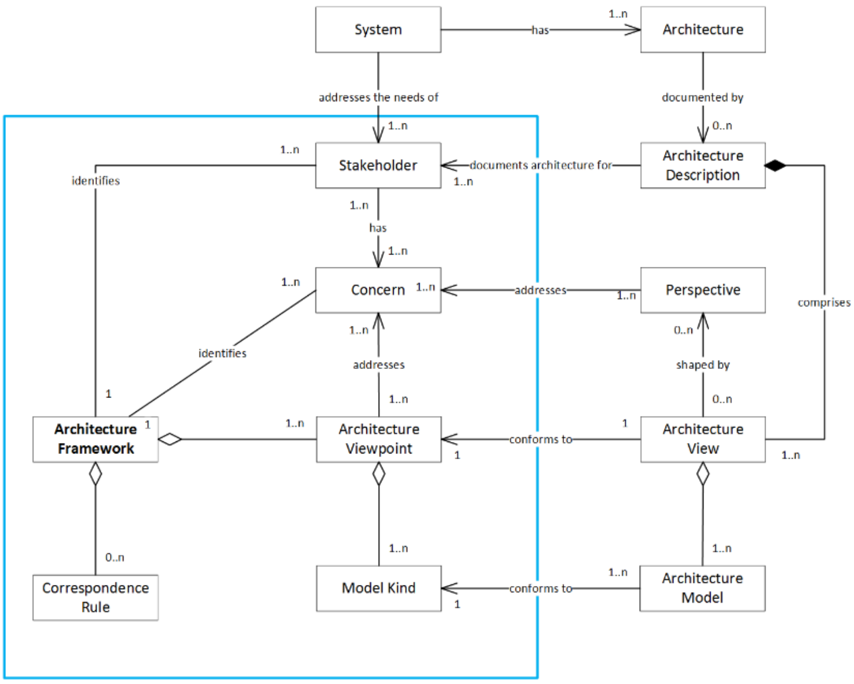
SD	Service Directory
SDM	Strategic Deployment Manager
SDK	Software Development Kit
SEC	Security
SHB	Single - Hop Broadcasting
SP	Service Provider
SPES	Service Provider Exchange System
SPAT	Signal Phase and Timing
SysML	Systems Modelling Language
TC/TCC	Traffic (Control) Centre
TCP	Transmission Control Protocol
TIS	Traffic Information System
TLC	Traffic Light Controller
TMS	Traffic Management System
TMT	Technical Management Team
TOPO	Topology Message
TPC	Transmit power control
TRC	Transmit rate control
TRL	Technical Readiness Level
TS	Technical Specification; usually deployed as „ETSI TS uniqueID“
UC	Use Case
UDP	User Datagram Protocol
UML	Unified Modelling Language
UMTS	Universal Mobile Telecommunications System (3G)
US	User Story
UTC	Coordinated Universal Time
V2I	Vehicle - to - infrastructure
V2V	Vehicle - to - Vehicle
V2X	Vehicle - to - X (generic term: X is either infrastructure or vehicle)
VDP	Visually Disabled Pedestrian
VEE	Vehicle Electrical and Electronic system
VIS	Vehicle Intelligent transport Sub-system
VRU	Vulnerable Road User
VRUITS	improving the safety and mobility of Vulnerable Road Users by ITS applications
WLAN	Wireless LAN
WP	Work Package
WPL	Work Package Leader
WWW	World Wide Web

Executive Summary

In the past years, there has been tremendous progress in the field of intelligent transport systems; several successful cooperative mobility projects and initiatives have proven potential benefits of cooperative systems in increasing both energy efficiency and safety for specific transport modes. However, the large variety of cooperative applications have been designed for different goals, stakeholders or specific settings / environments and have been developed on a silo-based approach and deployed independently from each other, serving however, at a higher level, to similar goals and functionalities for the end-user. Scalability, IT-security, interoperability, decentralization, and operator openness are some of the most important properties that a technical and commercial successful solution must provide.

C-MobILE aims to stimulate / push existing and new pilot sites towards large-scale, real-life C-ITS deployments interoperable across Europe. Well-defined operational procedures will lead to decentralized and dynamic coupling of systems, services and stakeholders across national and organizational borders in an open, but secure C-ITS ecosystem, based on different access technologies, the usage of which is transparent for service providers and seamless and continuous for the end-users across different transport modes, environments and countries.

The C-MobILE deliverable "D3.1 Reference Architecture" defines the **C-ITS architecture framework** which will be used to describe the C-MobILE's C-ITS concrete, implementation, and pilot site architectures as well as the **C-ITS reference architecture** which is adopted from the DITCM reference architecture [48] and captures the essence of C-MobILE vision and eight pilot site architectures. As the result of the architecture analysis and reverse architecting process, we identified that the DITCM reference architecture covers all the C-MobILE pre-selected services except the "Bundle1: urban efficiency" services. Therefore, the C-MobILE reference architecture is adopted from the DITCM reference architecture with the additional changes to enable the Bundle 1 services for urban efficiency. The architecture descriptions have four levels, each one refining the previous one: Reference, Concrete, Implementation, and Pilot Site Architectures. To put the architecture framework and architecture description concepts in context, C-MobILE extends the conceptual model of the ISO/IEC/IEEE 42010 international standard for architecture descriptions of systems and software [12], uses architecture viewpoints of the architecture framework for automotive systems [56] and concept of architecture perspective for shaping the architecture views [14].



As illustrated, an architecture description documents an architecture for the stakeholders using a set of architecture views and models. An architecture view conforms to a viewpoint, which addresses stakeholder concerns and can be shaped by a number of perspectives. The perspective defines concerns that guide architectural decision making to help ensure that the resulting architecture exhibit quality characteristics considered by the perspective. This is essential for the C-ITS architectures as the architectural choices are costly to make after the implementation. In the existing C-ITS reference architectures, security is considered broadly albeit from limited aspects e.g. from the information and communication views.

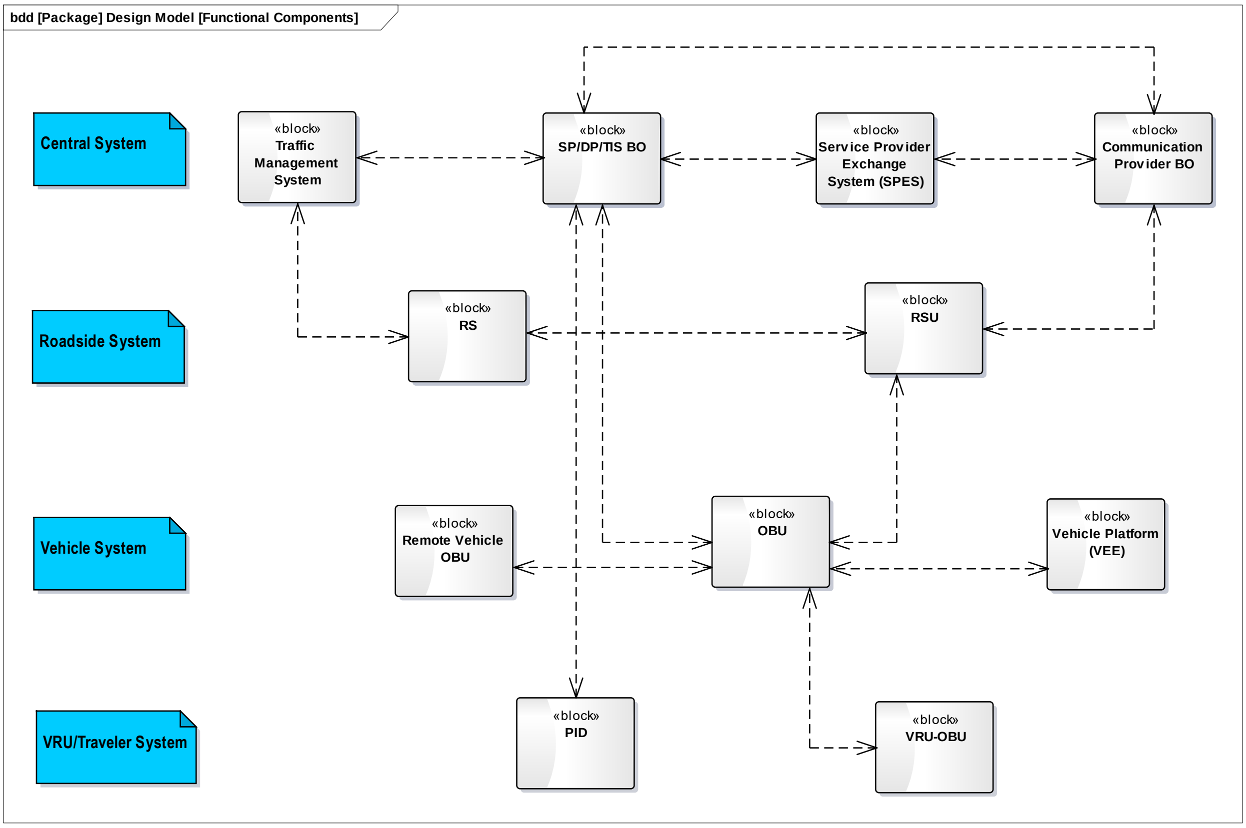
The C-ITS architecture framework specifies stakeholders, their concerns, viewpoints, model kinds, and correspondence rules. In addition, key architecture perspectives related to C-MobILE project will be addressed and given as examples for the concrete and implementation architectures. C-ITS architects can use an architecture framework to represent the C-ITS reference architecture, concrete, implementation and pilot site architectures.

The C-MobILE C-ITS architecture addresses the needs of the following stakeholders:

- **End-Users:** Drivers, Powered Two Wheelers, Cyclists, Pedestrians, Travellers, Fleet Operators, Road Works Operators;
- **Technological Providers:** Original Equipment Manufacturers (OEM), Telecom/Mobile Network Operators, Maps/Navigation and Data Providers, C-ITS Service Providers, Parking Operators or Parking Service Providers, Public Transport Operators;
- **Legal Authorities:** Road Operators and National/Local Authorities, Cities or Municipalities, Policy Advisor, Consultancy, Public-Private Partnership.

The C-ITS architecture framework has six core viewpoints: Context, Functional, Information, Physical, Communication, and Implementation. These viewpoints are defined based on the existing literature and ITS reference architectures. We believe that this set of viewpoints enable structured architectural descriptions for the C-ITS.

The *context* viewpoint describes the relationships, dependencies, and interactions between the system and its environment. The *Functional* viewpoint describes the system's runtime functional elements, their responsibilities, interfaces, and primary interactions. The *information* viewpoint describes how the architecture stores, manages, and distributes data and information. For the *physical* viewpoint, we see that C-MobILE infrastructure must facilitate large-scale deployment. This means that the defined infrastructure must support various technologies. The *communication* viewpoint describes the interfaces and communication protocols between different systems deployed on different hardware environment.



As illustrated, the high level functional model of the C-MobILE reference architecture is described in SysML block definition diagram capturing the main functions of the systems which are categorized into four main type of systems, also known as layers in the DITCM reference architecture [48]. The high level systems will be further refined in the concrete, implementation, and pilot site architectures. The current architectures of the pilot sites have been integrated for the definition of this reference architecture to ensure adaptability of the defined architectures. The high level architectural models in SysML diagrams are provided to the concrete and implementation architects along with this deliverable and supporting analysis materials to avoid architecture erosion and inconsistency issues.

1. Introduction

1.1. C-MobILE at a glance

The C-MobILE (Accelerating C-ITS Mobility Innovation and depLoyment in Europe) vision is a fully safe & efficient road transport without casualties and serious injuries on European roads, in particular in complex urban areas and for Vulnerable Road Users. We envision a congestion-free, sustainable, and economically viable mobility, minimizing the environmental impact of road transport. C-MobILE will set the basis for large-scale deployment in Europe, elevating research pilot sites to deployment locations of sustainable services that are supported by local authorities, using a common approach that ensures interoperability and seamless availability of services towards acceptable end user cost and positive business case for parties in the supply chain.

Eight C-ITS equipped cities and regions are involved in the project, all of which have been research pilot sites for large-scale deployment of sustainable services in the past. This common approach ensures that interoperability and seamless service availability are prioritised and at an acceptable cost for end users.

C-MobILE is engaging with public and private stakeholders, including end-users, to enhance C-ITS services and to establish functioning partnerships beyond the project. It is also carrying out cost-benefit analyses and developing business models, particularly from the end user's perspective, to make sure C-ITS services do their job correctly

1.2. Objectives and Scope

The main objective is to define a C-ITS reference architecture satisfying stakeholders from public and private parties in an EU context. The architecture framework and reference architecture should be used as a basis for developing concrete, implementation and C-MobILE pilot site architectures. This will enable C-MobILE deployment at EU defined eight pilot sites and beyond. The architecture framework and the same modelling approach can enable common language and will be reused for the next deployment projects. Furthermore, organisations can use the architecture to guide their internal development process as it reflects a common understanding of how the (future) ITS landscape will evolve.

From WP3, D3.1 "Reference Architecture" will focus more on an abstract level and use "black box" approach wherever possible. This document describes various systems at a high level in the form of models using SysML, a general purpose modelling language for engineering systems. The SysML Block Definition Diagram and Internal Block Diagram are used to represent the models for the reference architecture at an abstract level providing base level information to architects. However, due to the heterogeneous nature of such interfaces this will not be possible for all interfaces of the architecture. For example, there exists several competing standards for roadside infrastructure to communicate with traffic management centers. C-MobILE reference architecture does neither have the resource nor the intention to change/redefine all those standards. Instead, at high level we highlight the common systems, their interfaces and protocols by considering various existing projects to ensure interoperability.

The objective of the C-ITS architecture framework and reference architecture is to define as a first step in an integrated architecture based on a number of projects. The reference architecture should support the following high-level objectives:

- Consolidate the CONVERGE, MOBiNET, and Dutch C-ITS Reference Architecture (NL-RefArch) into one reference architecture and compare it with other architectures developed (e.g. from Compass4D, CO-GISTICS and SCOOP@F), to identify potential missing concepts.
- Identify a set of system/operational/service patterns based on existing architectures to provide degree of reuse.

The different architectures from the pilot sites will be combined in a single homogeneous, reference architecture in line with the current standards, which will be refined in the other tasks of WP3.

1.3. Document structure

This document is organized as follows:

- **Chapter 2** describes how the architecture was created in the **C-MobILE architecture methodology** process. It describes how the C-ITS architecture framework and reference architecture was created.
- **Chapter 3** lists the **related projects** which had an influence on the architecture created in C-MobILE. It does not only list previous projects, but also ongoing projects and existing reference architectures.
- **Chapter 4** describes the **C-ITS architecture framework** that has been created in C-MobILE. This section describes the architecture framework for the C-ITS domain and the proposed notation in SysML to facilitate common framework and notation method for the C-ITS concrete, implementation, and pilot site architectures. For ease of reading, the definitions of the architecture framework concepts are separately included in the viewpoint chapters.
- **Chapter 5 to 9** highlights the various viewpoints defined by the architecture framework.

- **Chapter 5 Context Viewpoint** describes the relationships, dependencies, and interactions between the system and its environment.
- **Chapter 6 Functional Viewpoint** describes the system's runtime functional elements, their responsibilities, interfaces, and primary interactions.
- **Chapter 7 Communication Viewpoint** describes the communications (e.g. interfaces, communication protocols) between different subsystems deployed on different hardware environment.
- **Chapter 8 Physical Viewpoint** describes the physical environment where the system will be deployed and the dependencies that the system has on elements of it.
- **Chapter 9 Implementation Viewpoint** describes the implementation for realizing functionality into real life software systems.

1.4. Intended audience

This document is intended to provide a high-level understanding of the architecture developed in C-MobILE. It is targeted mainly at technical people related to the definition and implementation of the architecture or parts thereof. Whilst in principal of interest for most of the C-ITS stakeholders, as listed in Table 3 below, this document is specifically targeted to architecture stakeholders such as system/software architects, software designers, software developer, technicians, and technical managers. Furthermore, it is intended to give decision makers a good understanding of the architecture introduced by C-MobILE.

1.5. Use

The SysML model concepts and Architecture Framework used at reference level architecture will be further used and revised by D3.2 to achieve mid-level architecture. The generic abstract output from this document will be considered further by comparing standards and provide hints to open solutions and describe how adapter mechanisms should be used to ensure interoperability at D3.2 and D3.4 levels.

1.6. Terms and Definitions

Term	Description
Actor	An actor is a human or machine entity that interacts with the system to perform meaningful work
Application	Software that can be deployed on end user devices that provided functions to the end user
Baseline	A baseline represents a reference/an existing situation without C-ITS service bundles that is compared to the same situation with C-ITS service bundles to assess the impacts that C-ITS service bundles cause in a determined environment (i.e., pilot sites)
Block Definition Diagram (BDD)	To show the system structure and identify the system components (blocks), and describe the flow of data between the components from a black-box perspective.
Customer	A person or an organization that uses services or applications
Data Acquisition	Is the process of sampling or recording data (real world data) for computer processing. It includes acquisition of pure sensor data, as well as acquisition of data from real-time and off-line services, and self-reported data.
Hypothesis	Is a specific statement linking a cause to an effect and based on a mechanism linking both. It is applied to one or more functions and can be tested with statistical means by analysing specific performance indicators in specific scenarios. A hypothesis is expected to predict the direction of the expected change.
Internal Block Diagram (IBD)	Provides the white box or internal view of a system block, and is usually instantiated from the BDD to represent the final assembly of all blocks within the main system block.
Latency	A latency period: the time between stimulus and response. In data acquisition generally the time between real world event (or stimulus) and the recording of that event.
Performance Indicators (PI)	Quantitative or qualitative measurement agreed on beforehand expressed as a percentage, index, rate or other value which is monitored at regular/irregular intervals and can be compared with one or more success criteria. It can be obtained directly from measures or derived, from CAN-bus of the vehicle, from external sensors, simulation procedures, questionnaires or events. [Definition: eCoMove Deliverable 6.2 based on FESTA]

Requirement	A requirement describes a condition or capability to which a system must conform; either derived directly from user needs, or stated in a contract, standard, specification, or other formally imposed document.
Research Question	Is a general question to be answered by compiling and testing related specific hypotheses.
Role	1. The usual or expected function of an actor, or the part somebody or something plays in a particular action or event. An Actor may have number of roles 2. The part that an individual plays in an organization and the contribution they make through the application of their skills, knowledge, experience, and abilities.
Service	Intangible equivalent of economic goods, e.g. a communication service offered by a communication network provider or a traffic information service offered by a traffic control centre.
Service Provider (SP)	An organization supplying services to one or more customers. Customers can include both other organizations, and end users.
Use Case (UC)	A use case is used to describe the system behaviour on a high level. It typically defines external system behaviour from a user perspective respectively it declares interactions between a role (known in UML as an "actor") and a system, to achieve a measurable goal. The actor can be a human or an external system.
User Story (US)	Descriptive way of formulating software requirements. The aim is to describe the system from the user perspective. A User Story can be seen as a script for a demonstrator to be developed. Based on the identified Use Cases and Requirements it should be possible to realize a User Story.
Validation	Validation or "did we build the right C-ITS service bundles /architecture?" is determining if C-ITS service bundles/architecture comply with the requirements and perform functions as intended.
Architecture view	Work product expressing the architecture of a system from the perspective of specific system concerns (ISO/IEC/IEEE 42010:2011)
Architecture viewpoint	Work product establishing the conventions for the construction, interpretation and use of Architecture View to frame specific system concerns (ISO/IEC/IEEE 42010:2011)

Table 1: Terms and Definitions

2. C-MobILE Architecture Methodology

To help reach the project goals, a good architecture, capable of being deployed to whole Europe is needed. The architecture definition process in C-MobILE has been defined to support the following sub-goals:

- To analyse existing C-ITS architectures to provide common concepts and vocabulary.
- To identify a set of patterns that have been detected (or applied implicitly) during the analysis of existing C-ITS architectures and their implementations.
- To create a C-ITS reference architecture that enables pan-European interoperability of C-ITS (concrete/implementation) architectures based on the generalization of existing C-ITS architectures.
- To define an implementation architecture specifying components and their relationships (interfaces) guided and constrained by the C-ITS reference architecture.
- To identify service-relevant parts of the architecture and define services based on the business analysis.

To do this, the architecture process has been split into three parts: the reference architecture for defining high-level architecture, the concrete architecture for refining the high-level architecture into the medium-level architecture, and the implementation architecture for revising further the concrete architecture into more detailed low-level architecture.

In the first design phase, the **reference architecture** has been created by analysing existing architectures, described in section 3. In addition, the current architectures in the various pilot sites have been taken into account. In parallel, use-cases, business-cases, and requirements to the C-MobILE system have been collected in WP2. The C-ITS **architecture framework** is also defined in the scope of this deliverable which will be used to describe the C-MobILE concrete, implementation, and pilot site architectures.

During the second design phase, the reference architecture will be used to create the **medium-level concrete architecture**. Furthermore, services, interfaces, and concepts will be described to provide a guideline for the final stage. In the third design phase, interfaces and concepts will be described in detail to create a **low-level implementation architecture**. All three phases are described in the following section.

2.1. C-MobILE Architecture Concepts

The architecture description in the C-MobILE project is split into three successive phases, which each phase creating a specific architecture description level, refining the descriptions of the previous phase.

	High-Level Reference Architecture	Medium-Level Concrete Architecture	Low-Level Implementation Architecture
Definition	Describes the system's high-level architectural elements, their dependencies and responsibilities.	Enhances reference architecture with functional interfaces, and primary interactions.	Enhances concrete architecture with detailed functions, interfaces, and interactions.
Concerns	Functional capabilities, generic interactions, and functional design philosophy (perspective).	Same as reference architecture but also external interfaces (data, event, control flows), primary internal structure (has impact on the system's quality characteristics).	Same as concrete architecture, but also external interfaces, internal structure (Has impact on the system's quality characteristics).
Model kinds	SysML Block Definition Diagram	SysML structural (Block Definition Diagram, Internal Block Definition diagram) and behavioural diagrams (State Machine, Activity and Sequence Diagrams)	SysML diagrams and UML diagrams for software architecture design

Table 2: Overview of architecture levels

2.1.1. Reference architecture concepts

The C-MobILE reference architecture shall capture the C-MobILE vision and combine the essence of ITS/C-ITS architectures already created in the previous projects and defined for existing pilot sites. Those architectures, listed in

Section 3.1, already cover almost all necessary services and concepts. However, none of those architectures covers all; therefore, a combination of those architectures is necessary. After architecture analysis and reverse architecting phase, the reference architecture concepts that we extracted especially functional viewpoint of the architecture was consistent with the DITCM reference architecture. Given the DITCM reference architecture captured the essence of the existing C-ITS architectures, it was reaffirming from the reference architecture definition point of view. However, some of the detailed descriptions of other aspects will be elaborated in the concrete and implementation architectures following the definition of abstraction levels defined in Table 2. The analysis results are captured in a separate Excel sheet to be provided as a supporting material to the concrete and implementation architects. The C-MobILE reference architecture is defined conforming to the C-ITS architecture framework proposed in this deliverable. This framework is defined according to the ISO/IEC/IEEE 42010 international standard on the architecture description for systems and software engineering [12]. The exact process is described in detail in section 4 of this document.

2.1.2. Concrete architecture concepts

As described above, the concrete architecture will be more detailed than the reference architecture. It will enhance the basic SysML diagrams of the reference architecture with functional interface descriptions, state diagrams, and sequence diagrams to describe the interaction of high-level components.

2.1.3. Implementation architecture concepts

The implementation architecture will describe the concrete architecture in detail, allowing the pilot sites to adapt their implementations to the C-MobILE architecture. The implementation architecture will use SysML and UML structural and behavioural diagrams. Interfaces between components are defined either by referencing the appropriate standard/definition or by describing them in a formal manner with sequence diagrams and class diagrams.

This architecture description level will be described in the follow up Deliverable D3.3.

3. RELATED PROJECTS

3.1. Existing Reference Architectures

Cooperative Intelligent Transportation Systems (C-ITS) are currently under intense development. Several previous projects have already been done regarding C-ITS, which have developed different architectures depending on the needs that the project wanted to cover or the technology that was being used. In C-MobILE, an infrastructure that is able to facilitate large-scale deployment must be developed. This means that the defined infrastructure must support various technologies (e.g., cellular and 802.11p), a wide range of users, while maintaining sustainability and seamless operation within the pilot sites. In order to make this possible, state-of the art solutions will be re-used, creating new solutions only when previous architectures do not meet our needs. In this section of the document, different projects related to C-ITS in Europe are presented. Out of those, the previously defined building blocks, which may be suitable for the C-MobILE project, have been extracted.

3.1.1. DITCM

DITCM is a Dutch program that had the aim to accelerate the deployment at large scale of C-ITS and Connected-Automated Driving (C-AD) and carried out from 2014 to 2015. To achieve this smart roads and smart cars need to work together. The challenges are the mix of old and new systems, evolving technology (moving target) and unpredictable human behaviour. DITCM is a cooperation between Connekt and AutomotiveNL.

In 2014, DITCM initiated a project with the objective to develop a reference architecture that can be used as a basis for future ITS deployment projects in The Netherlands [48]. The ITS architecture consists of a system architecture and a description of the business aspects of an eco-system with stakeholders from public and private parties in a Dutch context. The reference architecture was built based on the existing architectures from past and running C-ITS projects in The Netherlands (such as Shockwave Traffic Jams A58, Praktijkproef Amsterdam, ITS Corridor, MOBiNET, and VRUITS. Figure 1 presents a physical view of the DITCM reference architecture. It captures the decomposition of sub-systems and their dependencies in detail and in C-MobILE architecture this would be captured in the physical view of the concrete architecture as defined in section 2.1.

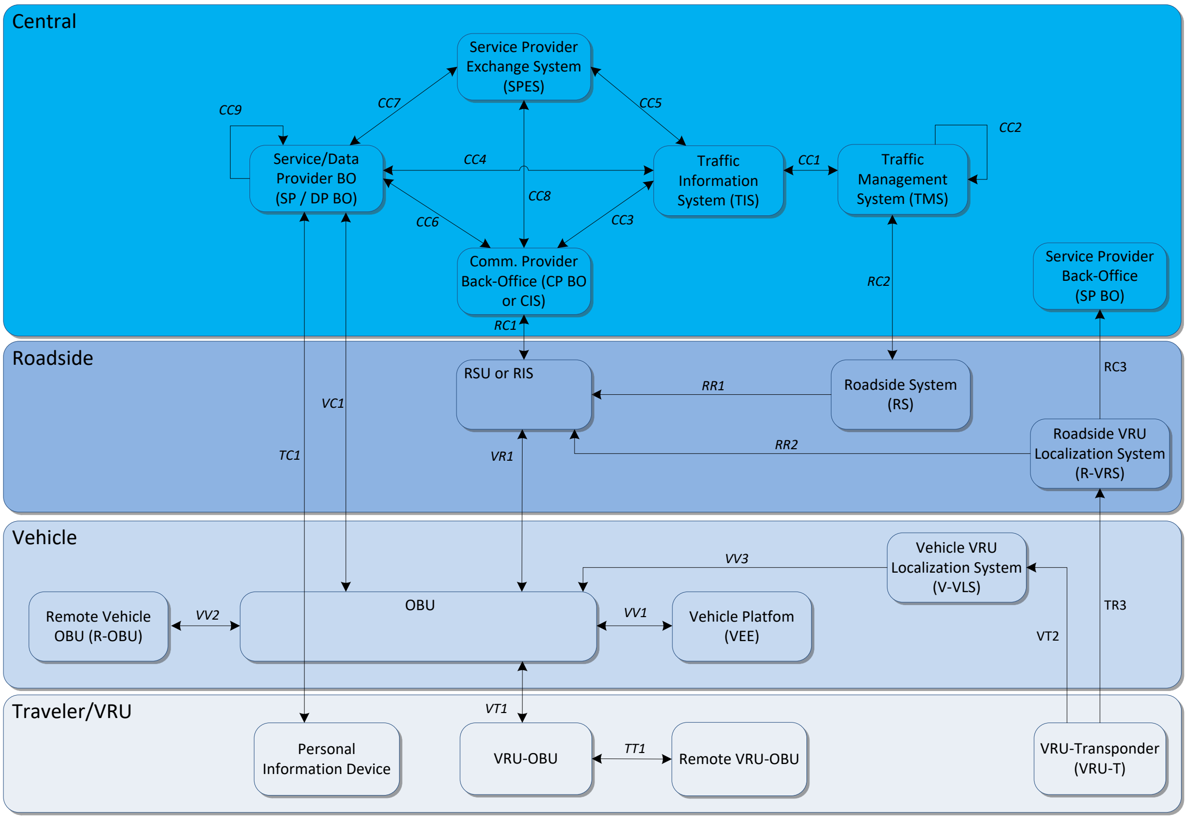


Figure 1: The NL Reference Architecture (physical view with sub-systems, Level 2) [48]

3.1.2. CONVERGE

The German research project CONVERGE [34], performed from 2012 to 2015, created an ITS architecture which was heavily focused on interoperability and economic viability. This architecture, called Car2X Systems Network was intended to be scalable, decentral, secure and not dependent on an operator, thus being resilient to change of the partaking organizations.

The CONVERGE architecture was separated in four major structural layers: the governance layer, the backend layer, the communication network layer and the ITS mobile stations.

The **governance layer** contains entities, which are necessary for contracts, regulations, and general supervision of the network. Most of those entities are not pure technical, but represent 'real-world' institutions. During the project, it was identified, that a kind of 'management board', called C2X Initialization Body, is needed, which will define the rules under which the different parties will work together.

The **backend layer** contains all entities, which are normally placed on servers, e.g. in enterprise networks or 'the cloud'. The most prominent entity being the Service Provider, which is an abstract entity, which is implemented by every party providing a service in the system. Most notably, the layer also contains a system to geo-reference user nodes through multiple communication networks, without the need to make the user nodes known to the Service Providers, the so-called GeoMessagingProxy.

The **communication networks layer** contains several different communication networks, linking backend entities to mobile stations. In CONVERGE, cellular communication and networks of ITS G5 roadside stations have been used, but the architecture is not limited to those.

The **ITS mobile stations layer** contains user related ITS stations, like vehicles or smartphones. The CONVERGE architecture didn't made any assumptions on the internal structure of those stations, but specified some external interfaces, to define how those stations interact with the other entities of the architecture. The architecture is shown Figure 2.

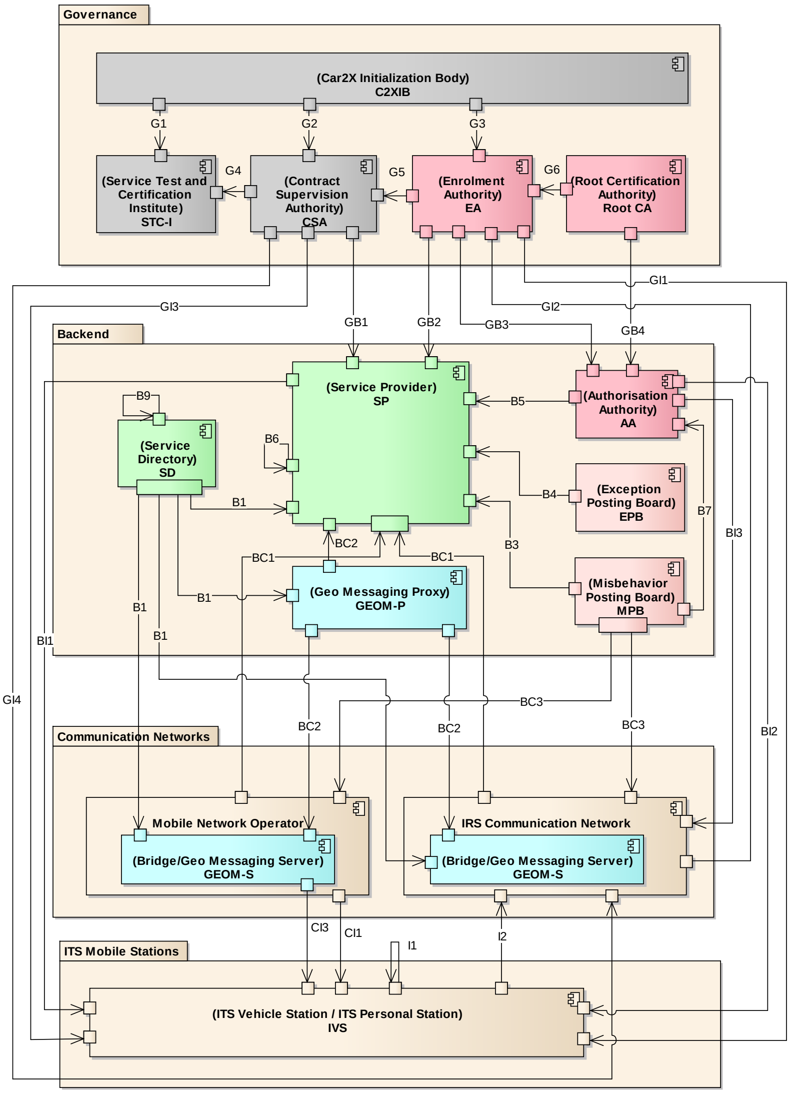


Figure 2: The CONVERGE architecture [44]

Besides those components, CONVERGE described some overarching concepts, needed in the architecture. Those were, for example: geo messaging over different networks, provider independent service directory, pseudonymous service usage and message topic aware communication via the geo networking mentioned above. Most notably, CONVERGE focused on the absence of 'single-points of failure' regarding the implementation of the architecture. For example, the system does not relay on a single Geo Messaging Proxy to be present, but all Service Providers also provide this proxy functionality, thus creating a distributed proxy. By doing this, it is ensured, that a proxy is present, if at least one Service Provider is present.

3.1.3. COMPASS4D

The European project Compass4D focused on three services aiming to increase safety and comfort for drivers by: 1) reducing the number and severity of road accidents, 2) optimising the vehicle speed at intersections, and 3) possibly avoiding queues and traffic jams. The three services were Energy Efficient Intersection Service (EEIS), Road Hazard Warning (RHW), and Red Light Violation Warning (RLVW) [1]. Regarding the Compass4D architecture, the basic idea was to use as many concepts, standards and background from previous projects (COSMO, FREILOT, DRIVEC2X, ECOMOVE, COVEL) as possible, in order to define a consolidated and interoperable architecture for all Compass4D pilot sites. The reference architecture identified three main subsystems, which were used by all pilot sites, while the architecture of the OBU and RSU followed the ETSI ITS station architecture as specified in ETSI ITS [2]. The three subsystems were: 1) OBU, the unit responsible for handling ETSI G5 communication with the RSUs/OBUs or cellular communication (LTE) with the BO, and the driver, 2) RSU, fixed units at the side of the road (e.g. traffic lights), equipped with ETSI G5 communication facilities, 3) BO, consisting in two sub-components, TMC and POMS, in charge of the traffic control and the operation of the pilot sites. The following picture provides an overview of the reference architecture, where the focus was only on facilities and applications layers of the ETSI ITS station architecture [3].

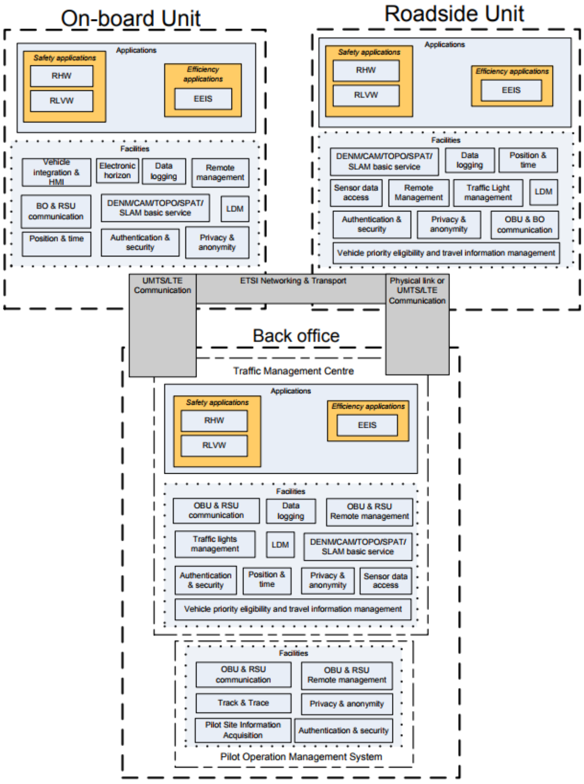


Figure 3: Compass4D Reference Architecture overview

3.1.4. MOBiNET

MOBiNET was a collaborative project, performed from 2012-11-01 to 2017-06-30, that aimed at simplifying the Europe-wide deployment of transport and mobility services by creating an "Internet of Mobility" in which transport users' requests match Service Providers' offers, and promotes openness, harmonisation, interoperability, and quality. It aimed to develop, deploy, and operate the technical and organisational foundations of an open, multi-vendor platform for Europe-wide transport and mobility services.

The key objective of the project is the simplification of the overall process of bringing together mobility service offerings and demand in a common market place. The open platform provides the required "glue" functionality to let Service Providers easily compose their services based on available data or other business-to-business (B2B) services, and to deliver their services to end users. End users will be enabled to easily discover and use these services. During MOBiNET project, ten example services had been identified that can be enhanced by a MOBiNET like platform. Besides these services, there were eight pilot sites EU-wide involved within the project: Aalborg, Helmond, Helsinki, London, Torino, Trikala, Trondheim, and Vigo. Each pilot site hosts one or multiple of the above-mentioned services. [32]

The figures below depict the components and tools as well the conceptual architecture of MOBiNET.

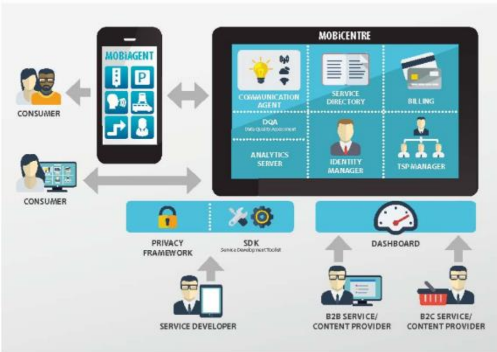


Figure 4: Component and tools [33]

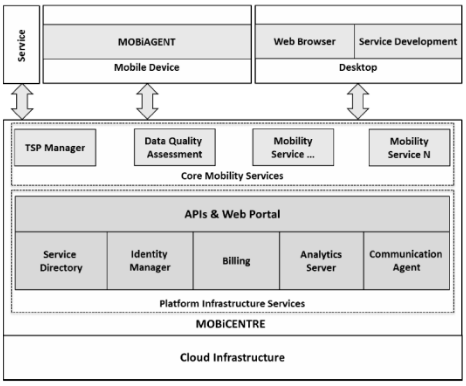


Figure 5: Conceptual architecture of MOBiNET[32]

Key MOBiNET innovations aimed to address the barriers of cooperative system-enabled service deployment, including the lack of harmonised service interfaces, availability of communication means, inaccessibility and incompatibility of transport-related data, fragmentation of end user subscription and payment services, and proprietary technologies in end user devices.

In this view, the architecture and its components create a new ecosystem for drivers, users and providers of transport services. It offers solutions for users including one click access to a one-stop shop for context-aware mobility services, pan-European roaming & coverage, integrated user accounts for transport services throughout Europe and traveller assistance tools for service roaming and virtual ticketing. For road and traffic operators, publishing traffic and travel information to all users, attract new customers and save costs. For the data and services providers, it allows to deliver services to any kind of compliant customer device, directory of all mobility-related data and services and service/data trading without one-to-one negotiation using the automated mechanisms for service orchestration. For developers, it provides an opportunity to develop a broad range of mobility Apps. These activities are further supported by a service development kit. [32][33]

3.1.5. NordicWay

The proposed action, NordicWay, is a pre-deployment pilot of Cooperative Intelligent Transport Systems (C-ITS) services in four countries (Finland, Sweden, Norway and Denmark) which will be followed by wide-scale deployment and potentially to be scaled up to Europe. NordicWay has the potential to improve safety, efficiency, and comfort of mobility and connect road transport with other modes. NordicWay is the first large-scale pilot using cellular communication (3G and LTE/4G) for C-ITS. This access network will be covered in the future by LTE/4G and later by 5G, and no specific investments in the infrastructure will be needed. It offers continuous interoperable services to the users with roaming between different mobile networks and cross-border, offering C-ITS services across all participating countries. NordicWay puts emphasis on building a sustainable business model on the large investment of the public sector on the priority services of the ITS Directive. NordicWay is fully based on European standards and will act as the last mile between C-ITS research and development and wide-scale deployment [19][3].

The NordicWay architecture uses a message queuing approach to transfer messages between the different actors, such as Service Providers, OEMs (Original Equipment Manufacturers, means car makers here), and Traffic Message Centers. A driver, resp. smartphone user, only communicates with a Service Provider/OEM cloud, which in turn communicates with other clouds, which belong to both other Service Providers/OEMs and national Traffic Clouds. Messages between these actors are relayed through the NordicWay Interchange Node, which distributes them to the actors, which have subscribed to the messages.

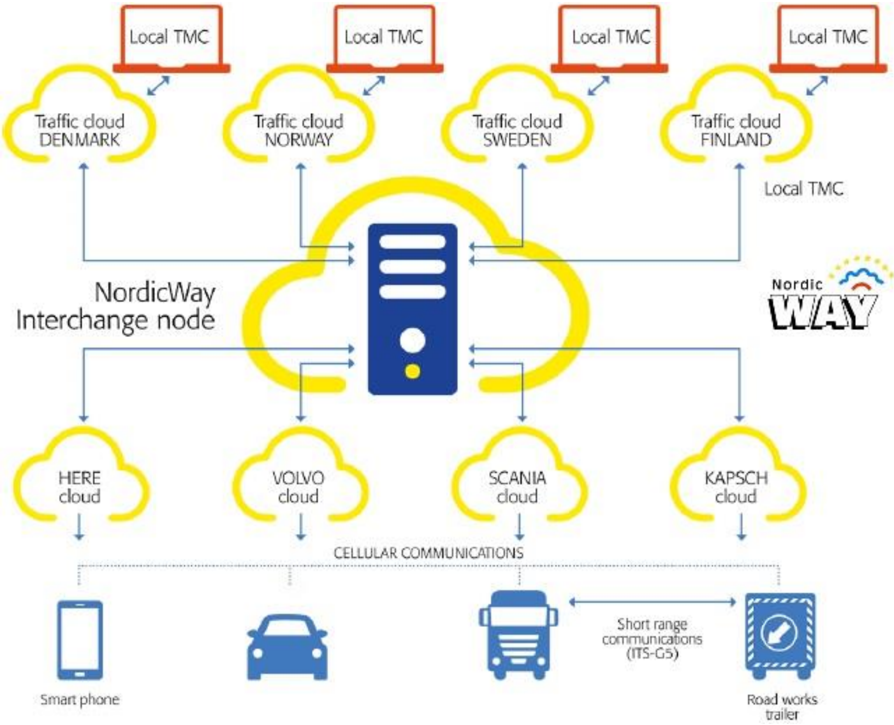


Figure 6: Nordic Way architecture

The NordicWay interchange uses a publish-subscribe AMQP (v1.0) queuing system for distributing messages between connected actors. In this system, all actors take on the role of either a producer or a consumer. A single actor can also be both a producer and a subscriber at the same time.

The data model for exchanged data is based on DATEX II version 2.3 level A with level B extensions. This means that nodes that support exchange of messages, which are conformant with this model, will be interoperable with respect to data exchange in NordicWay. The data definitions for DATEX II will be implemented as XML schemas, and the serialization format for DATEX II messages will be XML.

3.1.6. US ITS

The CVRIA (Connected Vehicle Implementation Architecture) Team, led by the ITS Joint Program Office, is comprised of the National ITS Architecture Team (led by Iteris), the Standards Program Technical Support Services Team (led by Booz Allen Hamilton) and the Policy Team (ITS JPO Policy Program and the Volpe National Transportation Systems Center). CVRIA is being developed as the basis for identifying the key interfaces across the connected vehicle environment, supporting this way further analysis for the identification and prioritization of standards' development activities. The approach taken to develop the CVRIA includes a number of source documents as inputs, such as Concepts of Operation (ConOps) documents from connected vehicle applications, Operational Concepts, the Core System ConOps, existing standards, the existing National ITS Architecture and the Core System architecture, as well as the existing International and Domestic standards. The development of the System Architecture was based on the fundamentals of ISO/IEC/IEEE 42010:2011 standard, including steps to define, data, messages, and the full environment in which the stakeholder concerns are satisfied. "CVRIA aims to become a "framework" for developers, standards organizations, and implementers to all use as a common frame of reference for developing the eventual systems" [4].

The four Views comprising CVRIA are described below [4]:

1. Enterprise: This view describes the relationships between organizations and the roles those organizations play within the connected vehicle environment. Here, the emphasis is given to the set of Enterprise Objects, which interact to exchange information, manage and operate systems beyond the scope of one organization. The Enterprise View includes relationships and interactions between the Enterprise Objects, as well as significant elements for the delivery of services, defined as Resources. The relationships between Enterprise Objects and between Enterprise Objects and Resources are determined by Roles (e.g., owns, operates, develops, etc.). Between Enterprise Objects there can also be Coordination in the form of an agreement or contract, in order to achieve the common purposes necessary to implement and carry-out a connected vehicle application.

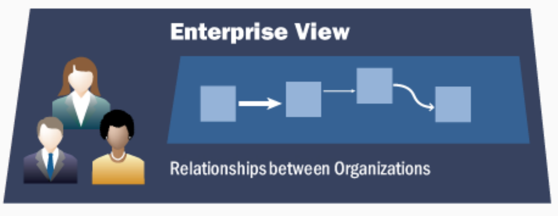


Figure 7: CVRIA Enterprise View

2. Functional: The Functional View addresses the analysis of abstract functional elements and their logical interactions. Here CVRIA is depicted as a set of hierarchically organized Processes (activities and functions), which trace to a set of Requirements derived from the connected vehicle source documents. The data flows between processes and the data stores, where data may reside for longer periods, are all defined in a Data Dictionary. The Process, or Function, is defined by a set of actions performed by this element to achieve an objective or to support actions of another Process. This typically involve data collection, data transformation, data generation, data generation or processing in performing those actions.

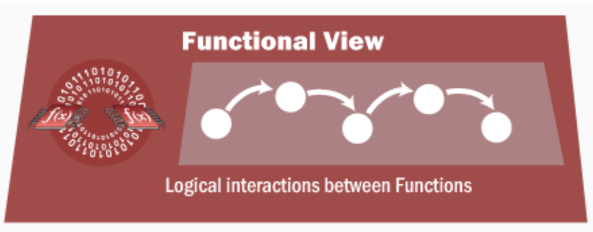


Figure 8: CVRIA Functional View

3. Physical: The Physical view describes the connections between Physical Objects within the connected vehicle environment. CVRIA is depicted as a set of integrated Physical Objects, which interact and exchange information to support the connected vehicle applications. Physical Objects represent the major physical components of the connected vehicle environment, including Application Objects that define more specifically the functionality and interfaces required to support a particular application. Information Flows depict the exchange of information that occurs between Physical Objects and Application Objects, which are identified by Triples. The Triples include the source and destination Physical Objects and the Information Flow that is exchanged. Each Application Object is linked to the Functional View and to the Enterprise view.

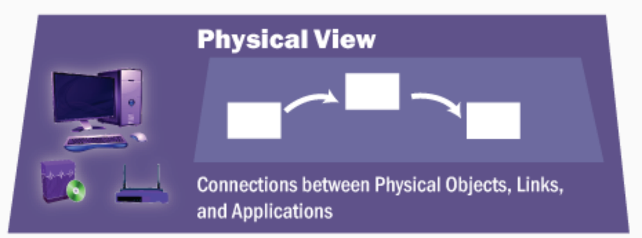


Figure 9: CVRIA Physical View

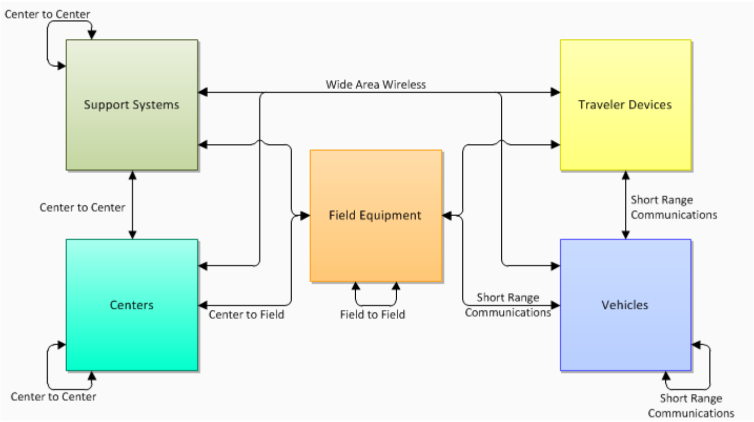


Figure 10: CVRIA Physical View - Interactions between systems

4. The Communications View describes the communications protocols necessary to provide interoperability between the Physical Objects. The CVRIA NITSA Communications Model and Communications Profiles are based on the Open System Interconnection (OSI) Model, the NTCIP Framework, and DSRC/WAVE Implementation Guide.

Figure 11: CVRIA Communications View

The 8 layers of the CVRIA NITSA Communications Model are described below:

- ITS Application Information Layer: The ITS Application information layer standards specify the structure, meaning, and control exchange of information between two end points.
- Application Layer: The application layer standards define the rules and procedures for exchanging encoded data.
- Presentation Layer: The presentation layer standards define the rules for representing the bits and bytes of information content to be transferred.
- Session Layer: The session layer provides the mechanism for opening, closing and managing a dialogue between application processes. Sessions may be asynchronous as in paired requests and responses (information exchanges), asynchronous as in an unsolicited publication of information, and may require acknowledgement or receipt or not.
- Transport Layer: The transport layer standards define the rules and procedures for exchanging application data between endpoints on a network.
- Network Layer: The network layer standards define the routing, message disassembly/re-assembly and network management functions.
- Data Link Layer: The data link layer standards define the rules and procedures for exchanging data between two adjacent devices over some communications media.
- Physical Layer: The physical layer is a general term used to describe the numerous signalling standards within this layer, typically developed for specific communications media and industry needs.

3.2. Related C-ITS Projects

Related projects and their architectural aspects e.g. quality requirements and functional structure models are described in short below.

3.2.1. C-The-Difference

C-The Difference pilot project has been elaborated on the basis of a shared vision developed and adopted by the consortium partners representing demand and supply sides who have been committed for the last 10 years to bring C-ITS (Cooperative Intelligent Transport Systems) to the market through intensive efforts and long lasting investments in the development and deployment of C-ITS services. This group of pioneers are strong believers in the capacity of C-ITS services to bring efficient and cost-effective solutions to address urban mobility problems with respect to traffic efficiency, safety, and impact on the environment.

Success in implementation and long run provision of C-ITS services rely on five golden rules that need to be addressed in a coordinated and integrated way:

- Inter-operability: Thanks to adoption of international standards, C-ITS services are fully inter-operable and continuity of services can be guaranteed independently from geographical location, C-ITS Service Provider and C-ITS system suppliers
- Sustainability: Key actors from public and private sectors involved in the C-ITS service chain are engaged in a long-term cooperation to create added value to all users in their daily mobility, to develop viable business models, to raise awareness on C-ITS benefits, to build European-wide C-ITS market and to contribute to economic growth.
- Scalability: A deployment scenario can be customized according to user needs, urban transport and mobility policies, existing infrastructure, and financial capacity. Thanks to scalable architecture, implementation can start with a first package of C-ITS services that deliver quick benefits with respect to urban mobility priorities and can be further developed in a modular approach by means of additional services and/or extended geographical coverage and/or an increasing number of users with minimum additional costs. Combined use of G5 and 3G/4G communication technologies contribute to speeding up the penetration rate of several C-ITS services.
- Replicability: C-ITS services are not restricted to a small number of front-runner cities. All cities can benefit from experience of early adopters by means of effective knowledge sharing to facilitate decision making in initial C-ITS investment and to accelerate deployment of customized C-ITS solutions.
- Reliability: Cities can rely on sound evidence of C-ITS benefits to take decisions on C-ITS service implementation that can be integrated within existing transport and mobility infrastructure. Cities can invest in confidence in a portfolio of C-ITS services based on mature and cost-effective technologies, and open and standardized architecture enabling high quality service provision and capacity to integrate new features.

Objectives of C-The-Difference are the following:

- Deliver comprehensive and integrated impact assessment by means of enhanced evaluation methodology and up to 18 months operation of C-ITS services package.
- Bridge the gap between most advanced C-ITS implementations in urban environment and large-scale deployment and operations by targeting professionals responsible for urban transport planning and operations, policy makers and decision makers.
- Convince European cities to invest in mature and proven C-ITS solutions by fostering and replication through City Twinning Program.
- C-The-Difference offers the following key innovative solutions:
- A traffic app allowing you to adapt your driving depending on the infrastructures and the different road events that might occur. Deployed and functional on Bordeaux urban area, it can be downloaded in the app stores.
- Green light priority with SSM/SRM functionality for direction dependent priority calls. Unlike older projects like Compass4D, it is possible to request priority for a specific turn direction. This improves the effectiveness for the prioritized vehicle and traffic flow for other vehicles.
- Extended emergency vehicle warning with indication of the direction of the vehicle on the HMI.

3.2.2. SCOOP@F phase 2

SCOOP@F is a C-ITS pilot deployment project, intending to connect approximately 3000 vehicles with 2000 kilometres of roads. It consists of 5 pilot sites, Ile-de-France, "East Corridor" between Paris and Strasbourg, Brittany, Bordeaux and Isère, and is composed of SCOOP@F Part 1, 2014-2015, and SCOOP@F Part 2, 2016-2018. SCOOP@F Part 2 includes the validations of C-ITS services in open roads, cross border tests with other EU Member States and development of a hybrid communication solution (3G-4G/ITS G5) [5]. The architecture of the SCOOP@F project is presented in terms of network architecture its integration in the road operators' networks. Regarding the physical architecture, the servers and equipment needed for operation include PKI servers, to manage the different certificates, road operator servers and OBUs. The communications between RSUs and

OBU are in: GeoNetworking for exchanges of CAM and DENM messages, IPv6 for OBU exchanges with the PKI, and IPv4 for RSUs exchanges with the PKI [6].

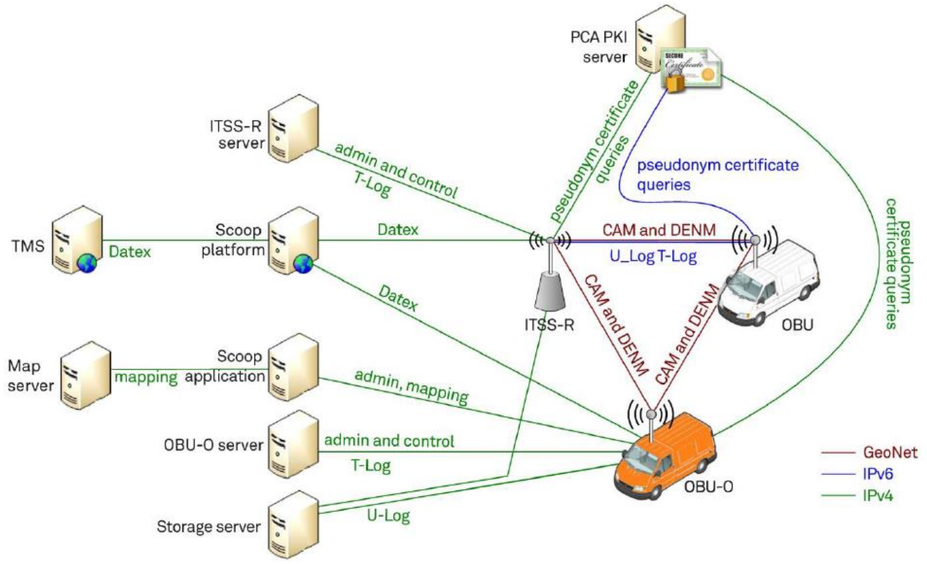


Figure 12: Summary schema of exchanges

The Scoop@F architecture must fit into the existing networks of traffic operators, presented schematically as follows [6]:

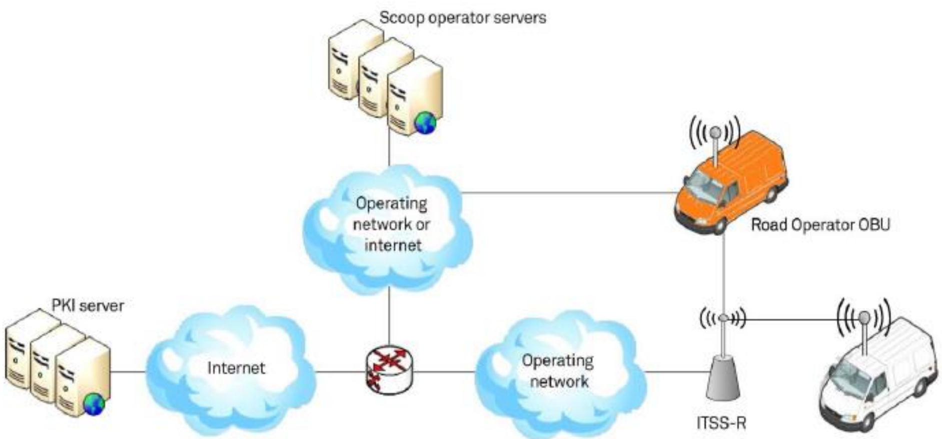


Figure 13: General integration schema

3.2.3. InterCor

InterCor constitutes an action, which aims to streamline C-ITS implementation in 4 member states linking the different national initiatives towards a harmonized strategic rollout and common specification. C-ITS pilot sites to communicate data through cellular and/or ITS G5 networks will be installed in approximately 968 km along the Netherlands, Belgium, UK, and France, for operation and evaluation of C-ITS services. InterCor will focus on the deployment of “Day-1” services as recommended by EC “C-ITS Platform” such as Road works warning, Green Light Optimized Speed Advisory, In-Vehicle-Signage and Probe vehicle data. Furthermore, an additional use case dedicated to transport of goods (MultiModalCargoTransportationOptimization) will also be deployed in several countries in order to integrate logistics and C-ITS services but also to test interoperability between countries. The action is part of the C-Roads platform, which is a platform of Member States working on the deployment of C-ITS services. Cross-border tests will also be conducted with other C-Roads Member States [24].

3.2.4. C-Roads

C-Roads [42] is a platform of Member States working on the deployment of C-ITS services. C-ITS pilot sites will be installed across the EU for testing and later operation of "Day-1" applications as recommended by EC "C-ITS platform". Member States will invest in their infrastructure, while the industry will test components and services. Technical and organisational issues will be tackled by the C-Roads platform to ensure interoperability and harmonisation of C-ITS between pilots.

The project runs from February 2016 until December 2020 and is cofounded by the Connecting Europe Facility (CEF).

The services C-Roads wants to implement are:

- Slow or stationary vehicle(s) and traffic ahead warning
- RWW: Road Works Warning
- Weather Conditions
- Emergency Vehicle Approaching
- OHLN Other Hazardous (Location) Notifications
- IVS: In-Vehicle Signage
- In-Vehicle Speed Limits
- Signal Violation / Intersection Safety
- GLOSA: Green Light Optimal Speed Advisory
- Probe Vehicle Data
- Shockwave Damping

C-Roads published a first deliverable “Harmonised C-ITS Specifications for Europe” for the RWW, IVS, OHLN and GLOSA services.

The Focus is on ITS-G5 communication (ETSI ITS G5) and Hybrid communication with a security mechanism.

3.2.5. CITRUS

CITRUS [35] stands for C-ITS for Trucks. It is a Belgian project, which will run from October 2016 until September 2019, that aims to improve road safety and reduce CO2 emissions of truck traffic. CITRUS is cofounded by the Connecting Europe Facility TRANSPORT.

The project builds a companion mobile app for truck drivers tested with 300 truck drivers. The services are based on existing 3G/4G communication in combination with geographical messaging technologies. ITS-G5 is envisaged.

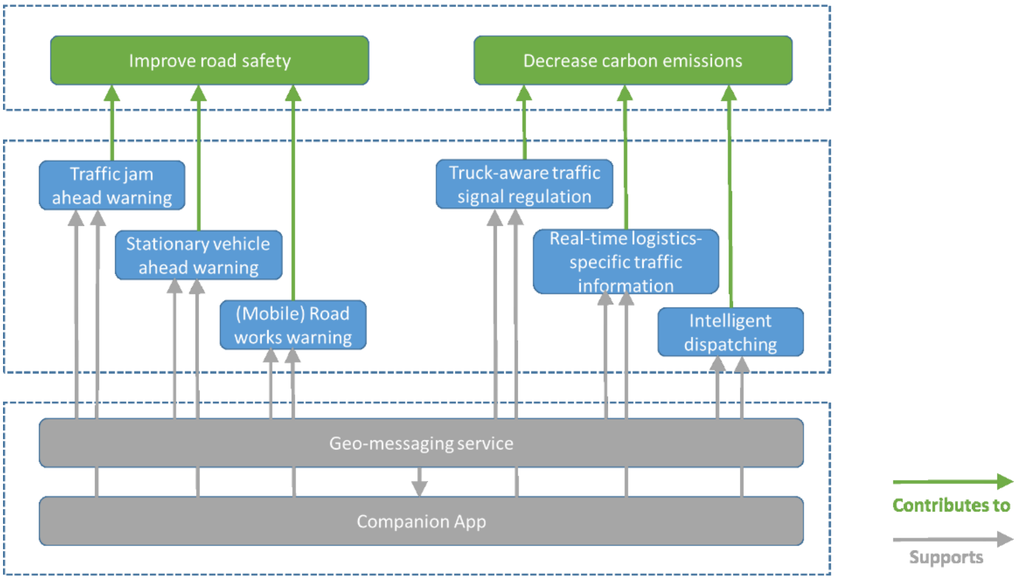


Figure 14: CITRUS architecture

The Day-1 services are:

/ Traffic jam ahead warning

/ Stationary vehicle ahead warning

/ (Mobile) Road works ahead warning

/ Truck-aware traffic signal regulation (GLOSA)

Day-1.5 services are:

/ Real-time logistics-specific traffic information

/ Intelligent dispatching

The project concentrates on the principal highways in Flanders and semi-motorways near Halle where the Colruyt distribution center is located [35] [50].

3.2.6. VRUITS

The EU-funded VRUITS project, which started in 2013, aims at actively integrating the “human” element in the ITS approach by focusing on needs of all relevant stakeholder groups, in order to improve traffic safety and the general mobility of vulnerable road users (VRUs) [7]. The VRUITS architecture supports the following ITS applications [8]:

/ Intelligent Pedestrians Traffic Signal

/ Intersection Safety

/ Powered Two Wheelers (PTW) Oncoming vehicle info system

/ Vulnerable Road User (VRU) Beacon System

/ Roadside Pedestrian Presence

/ Bicycle to car communication

/ Green wave for cyclists

This VRUITS reference architecture, used to develop subsystems and elements for C-ITS applications for VRUs, is through [8]:

5. The physical architecture, which is a high-level description of the physical ITS sub-systems used by VRUs, Vehicles, Roadside and Central layer, together with high-level description of the communication/interaction between these sub-systems.
6. The Functional architecture, which describes the functional elements within the sub-systems in the VRU, Vehicle, Roadside, and Central layer.
7. The Communication architecture, which describes the type of communication and networks used between the functional elements of the physical sub-systems.

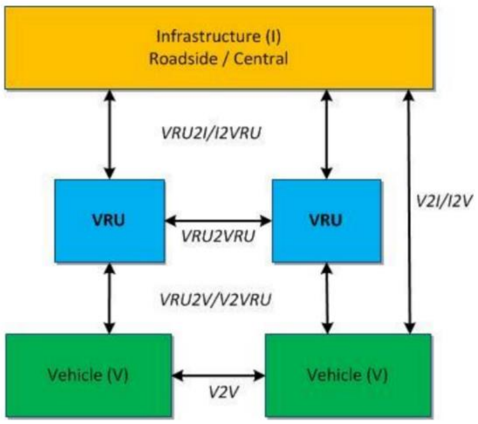


Figure 15: Physical architecture for C-ITS applications with Vulnerable Road Users

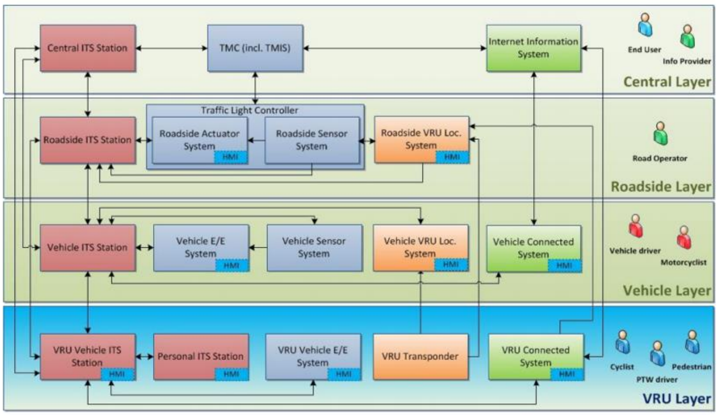


Figure 16: Functional Architecture for C-ITS with Vulnerable Road Users

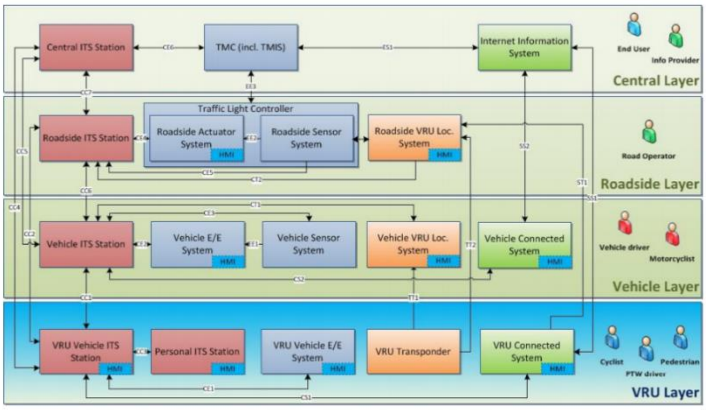


Figure 17: Functional architecture with marked interfaces

3.2.7. iKoPa

The German project iKoPA (Integrated cooperating platform for automated electric vehicles)[36] is a continuation project of CONVERGE. It was started in 2016 and will run until the end of 2018. It focuses on the enhancement of the CONVERGE architecture described above. It will integrate DAB+ as a communication network, assess the usefulness of the architecture for electric mobility and validate the conformity of the architecture with the requirements created by the General Data Protection Regulation (Regulation (EU) 2016/679)[37] coming into effect on 2018-05-25.

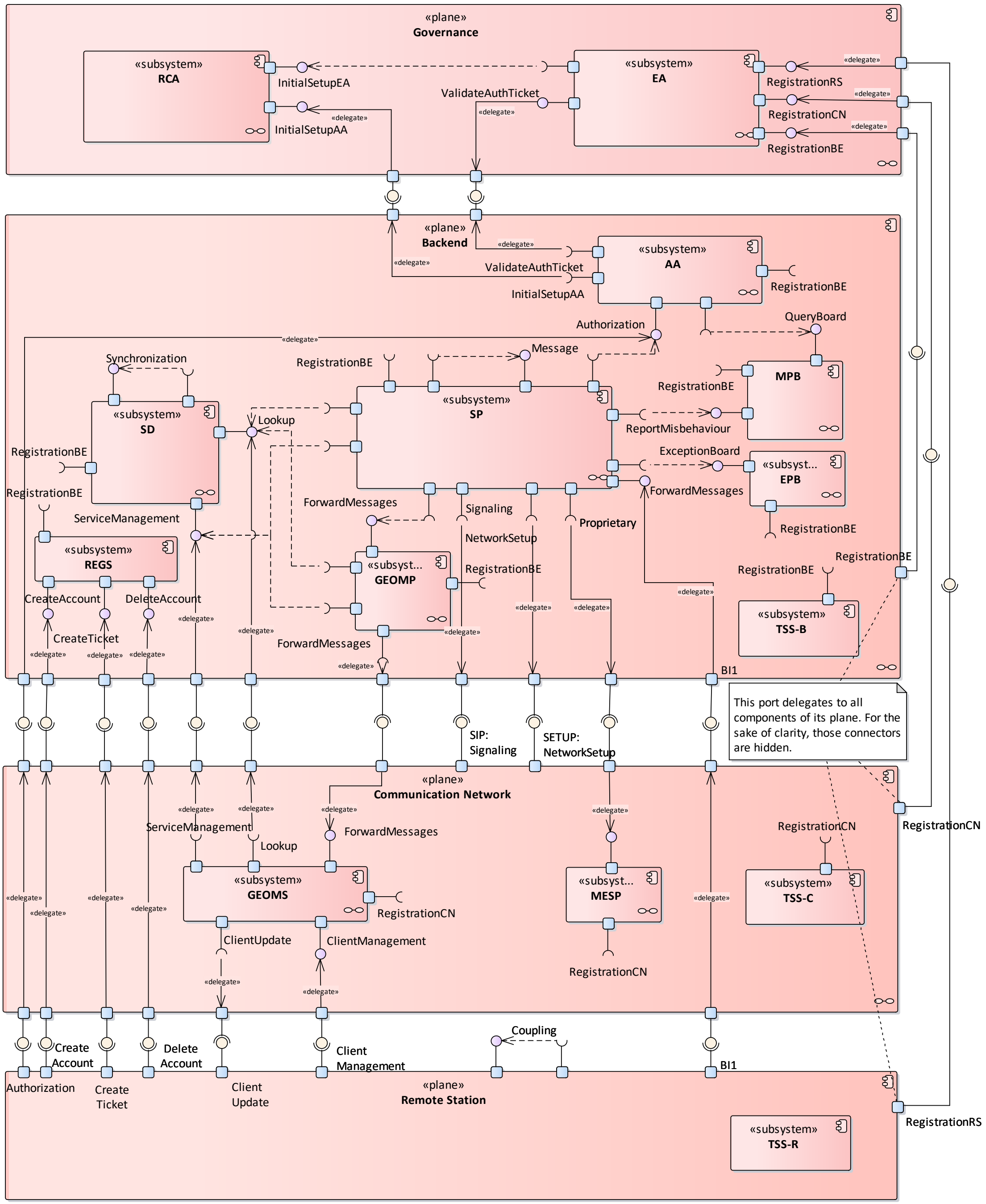


Figure 18: The iKoPA architecture [43]

The basic architecture is very similar to the Car2X Systems Network designed by CONVERGE. However, interfaces have been enhanced and described more independently. In addition, new entities have been introduced, like the Registration Server. This entity decouples the registration process from the service usage, so that a Service Provider is not able to determine the identity of its users.

3.2.8. Talking Traffic

The goal of Talking Traffic [38] is C-ITS deployment. It is a collaboration between the Dutch Ministry of Infrastructure and the Environment, regional and local authorities and (inter)national companies [51].

The use cases are In-vehicle signage, Road hazard warning, Priority at traffic lights, Traffic lights info, Flow optimization and In-vehicle parking info.

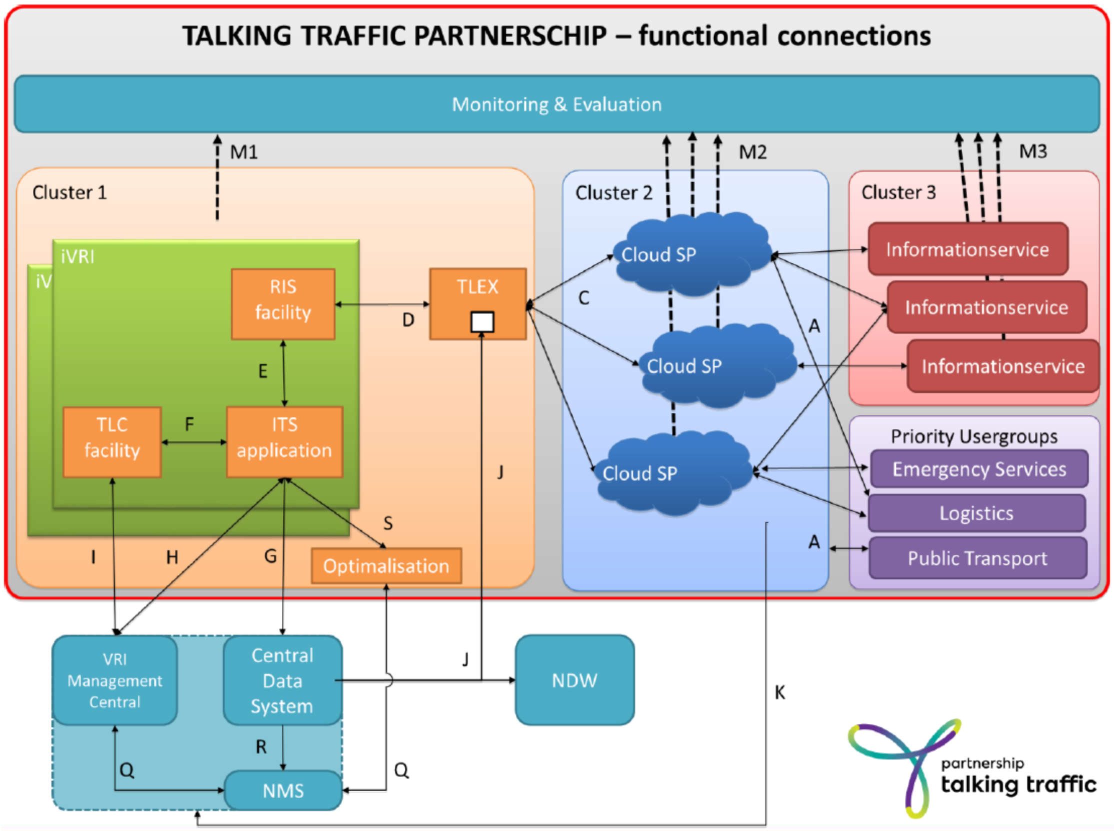


Figure 19: Talking traffic architecture [38]

3.2.9. C-ITS Corridor

German, Dutch and Austrian highway operators, in cooperation with partners from the automotive industry, have launched the gradual deployment of Cooperative Systems. This allows the exchange of traffic-related information among vehicles and between vehicles and the roadside infrastructure. In June 2013, the national transport ministries of Germany, the Netherlands and Austria signed a Memorandum of Understanding (MoU), which marked the start of Cooperative Systems[39]. The ministries agreed:

- to develop a common launch/rollout timetable for the implementation of the first cooperative applications on highways
- to define common conventions that ensure a harmonized interface with vehicles in the three countries
- to implement roadside facilities for the first collaborative applications.

A highway corridor, extending from Rotterdam (Netherlands), via Frankfurt/ M. (Germany) to Vienna (Austria), was chosen as the route for the first deployment - the so-called C-ITS Corridor[39].

The Netherlands, Germany, and Austria have agreed upon the introduction of two cooperative services: (1) roadworks warning and (2) improved traffic management by vehicle data. They are part of the "Day 1" services defined by the Amsterdam Group. Both applications have been selected because of their present relevance, as there are still many accidents related to roadworks, and with regard to the further dissemination of C-ITS technology. In both cases, communication from the vehicle and infrastructure is established via short-range communication (Wifi 802.11p, 5.9GHz) or the cellular network (3G, 4G). Both initial applications increase road safety and provide the basis for an improved traffic flow. Thanks to the collaboration of the automotive industry, service providers, and road operators, cooperative ITS systems will be directly experienced by the road user and are useful for everyone [39].

3.2.10. ECo-AT

ECo-AT (European Corridor – Austrian Testbed for Cooperative Systems) is the Austrian project to create harmonised and standardised cooperative ITS applications jointly with partners in Germany and the Netherlands. The project is led by the Austrian motorway operator ASFINAG and the consortium consists of Kapsch TrafficCom AG, Siemens AG Österreich, SWARCO AG, High Tech Marketing, Volvo Technology AB, ITS Vienna Region, FTW (Forschungszentrum

Telekommunikation Wien) and BASt (Bundesanstalt für Straßenwesen). Within the next years C-ITS shall be developed for being applied in the Cooperative ITS Corridor Rotterdam – Frankfurt/M. – Wien. This happens in close cooperation between the EU-member states Netherlands, Germany and Austria, who have signed a Memorandum of Understanding. Austria has established itself as constructive implementation pioneer of C-ITS applications within the European Union – not least because of the project Testfeld Telematik. The next steps towards pan-European deployment in the European C-ITS Corridor from the Netherlands via Germany to Austria will be prepared and developed within the project ECo-AT for the Austrian section[25].

3.2.11. CODECS

The Coordination and Support Action COoperative ITS DEployment Coordination Support (CODECS) supports the European Commission and the manifold stakeholders involved in C-ITS deployment in finding strategic and technical policy solutions and processes for a consolidated C-ITS roll-out. CODECS serves as hub for transparent information and knowledge transfer on function approaches, experiences and lessons-learned by stakeholders active in the initial deployment. To ensure European-wide seamless (cross-border) interoperability and end-user experiences, CODECS develops a harmonised standards profile supporting a growing amount of C-ITS services. To address key organisational and technology related issues, CODECS will derive a strategic common road map from preferences of the involved stakeholders, giving direction for innovation, testing, standardisation, and deployment beyond Day One. CODECS also supports future C-ITS common deployment by achieving a clear understanding on policies, roles, and responsibilities. CODECS does convey these insights to the C-ITS deployment platform initiated by the European Commission and to the Amsterdam Group [26].

3.2.12. AUTOCITS

The aim of the Study is to contribute to the deployment of C-ITS in Europe by enhancing interoperability for autonomous vehicles as well as to boost the role of C-ITS as catalyst for the implementation of autonomous driving. Pilots will be implemented in 3 major Core Urban nodes (Paris, Madrid, Lisbon) located along the Core network Atlantic Corridor in 3 different Member States. The Action consists of Analysis and design, Pilots deployment and assessment, Dissemination and communication as well as Project Management and Coordination. The three pilots will test and evaluate C-ITS services for autonomous vehicles under the applicable traffic regulation, study its extension to other European countries and contribute to the C-Roads and C-ITS platform as well as to other European standards organizations [27].

3.2.13. CITRUS

The Action takes place in Belgium and studies the technical and economic viability of a companion mobile app for truck drivers. It envisages the development of the app as well as a pilot deployment and testing involving around 300 truck drivers. The app will provide some services relating to "Day 1 services" as identified by the C-ITS platform, like giving safety related warnings or advice as regards speed, routing, and other information. Services deployed will be based on a cellular C-ITS approach in combination with geographical messaging technologies. The project will contribute to improve road safety and reduces CO_2 emissions of truck traffic [28].

3.2.14. PROSPECT

PROSPECT will significantly improve the effectiveness of active VRU safety systems compared to those currently on the market. This will be achieved in two complementary ways, first by expanded scope of VRU scenarios addressed and second by improved overall system performance (earlier and more robust detection of VRUs, proactive situation analysis, and fast actuators combined with new intervention strategies for collision avoidance). PROSPECT targets five key objectives [29]:

- Better understanding of relevant VRU scenarios
- Improved VRU sensing and situational analysis
- Advanced HMI and vehicle control strategies
- Four vehicle demonstrators, a mobile driving simulator and a realistic bicycle dummy demonstrator
- Testing in realistic traffic scenarios and user acceptance study.

The consortium includes the majority of European OEMs (Audi, BMW, DAIMLER, TME and Volvo Cars) currently offering AEB systems for VRU. They are keen to introduce the next generation systems into the market. BOSCH and CONTI will contribute with next generation components and intervention concepts. Video algorithms will be developed by UoA and DAIMLER. Driver interaction aspects (HMI) are considered by UoN and IFSTTAR. Euro NCAP test labs (IDIADA, BAST, TNO) will define and validate test procedures and propose standardization to Euro NCAP and UN-ECE. Accident research will be performed by Chalmers, VTI and BME, based on major in-depth accident databases (GIDAS and IGLAD) and complemented by East Europe data. The work will be done in cooperation with experts in Japan (JARI, NTSEL) and the US (VTTI, UMTRI, NHTSA) [29].

3.2.15. XCYCLE

The XCYCLE [49] project aims to level the treatment of cyclists in road traffic, thus encouraging cycling and increasing its safety margin. The XCYCLE Project has started on June 1st, 2015 and its activities will be completed by November 2018. The project is developing [30]:

- Technologies that improve active and passive detection of cyclists
- Systems informing both drivers and cyclists of a hazard at junctions
- Effective methods of presenting information in vehicles and on-site
- Cooperation systems aimed at reducing collisions with cyclists.

Two relevant use cases will be bicycle interaction with large vehicles and cars at intersections and immediate or extended green traffic light for cyclists approaching traffic signals. An in-vehicle detection system and a system of threat mitigation and risk avoidance by traffic signals will be developed. The components developed and built up will be systematically integrated, implemented, and verified. A new large-scale research infrastructure in the city of Braunschweig (Germany) and a second test mobile platform will be used as test site. A demo bicycle with a cooperative technology will be developed and tested as well. A user-centred approach will be adopted. Behavioural evaluation will part of the whole process: attentional responses using eye tracking data; evaluation of human-machine interface; acceptance and willingness to pay. In the cost-benefit analysis, behavioural changes will be translated into estimated crashes and casualties avoided [30].

3.2.16. SOLRED

The overall objective of the action is to test a new Integrated Fuel and Fleet Management System, the so-called C-ITS Telemat, which enables the automatic real time calculation of the smartest route plan and the automatic estimation, authorisation and payment of the refuelling needed to complete a planned route. Moreover, the system provides the heavy duty vehicles (HDV) drivers and fleet managers with useful notifications concerning maintenance scheduling, eco/safety driving, traffic issues as well as information on the estimated fuel consumption vis-a-vis the real one[31].

The testing of this system will be done through a monitoring network, which will involve approximately 53 Repsol service stations along the Spanish part of the Atlantic and Mediterranean core network Corridors. This C-ITS Telemat is expected to contribute to road safety by avoiding accidents and fuel thefts and fraud, but also will assist in improving traffic flows, and reducing fuel consumption, congestion, and environmental impacts. In particular, this new technology is expected to reduce fuel fraud by 5% and fuel consumption by 3% as compared with the previous ITS Telemat technology used by Repsol service stations[31].

4. C-ITS Architecture Framework

4.1. Architecture Framework

An architecture framework is one of the widely-applied approaches used in software/system architecting of complex systems. The architecture frameworks facilitate communication and cooperation between different stakeholders during architecting and building complex systems such as C-ITS. Many different stakeholders with their interweaving concerns require a systematic approach for addressing complexity and full lifecycle of the system. Example representations of widely applied architecture frameworks include Kruchten's 4+1 View Model, TOGAF, and Zachman framework. 4+1 is developed by Kruchten as a generic architecture framework for describing the architecture of software-intensive systems based on the use of multiple, concurrent views [15]. TOGAF provides a practical and industry standardized approach for designing an enterprise architecture [16]. Zachman framework is used for modeling an enterprise's information infrastructure from six perspectives [17].

According to the ISO/IEC/IEEE 42010:2011 standard, an *architecture framework* establishes a common practice for creating, interpreting, analysing and using architecture descriptions within a particular domain of application or stakeholder community [12]. As a well-defined architecture framework is considered to be an important part of any architecture description [13], we define an architecture framework for the C-ITS domain. To put architecture framework and architecture description concepts in context, we extend the conceptual model of the ISO/IEC/IEEE 42010 architecture framework as illustrated in

Figure 20. Without the common C-ITS architecture framework, different categorizations and ad-hoc notations are used in the existing C-ITS reference architectures. The C-ITS domain covers not only software/system engineering field, but also traffic engineering, civil engineering, information technology etc., which require a unified definition of architecture framework for the C-ITS domain.

The conceptual model of *architecture framework* is highlighted in blue and the relationships of architecture description concepts are added to the original standard diagram. As illustrated, an *architecture description* documents an architecture for the stakeholders using a set of Architecture Views and models. An *Architecture View* conforms to a *viewpoint* which addresses stakeholder concerns and can be shaped by a number of perspectives. The *perspective* defines concerns that guide architectural decision making to help ensure that the resulting architecture quality characteristics considered by the perspective [14].

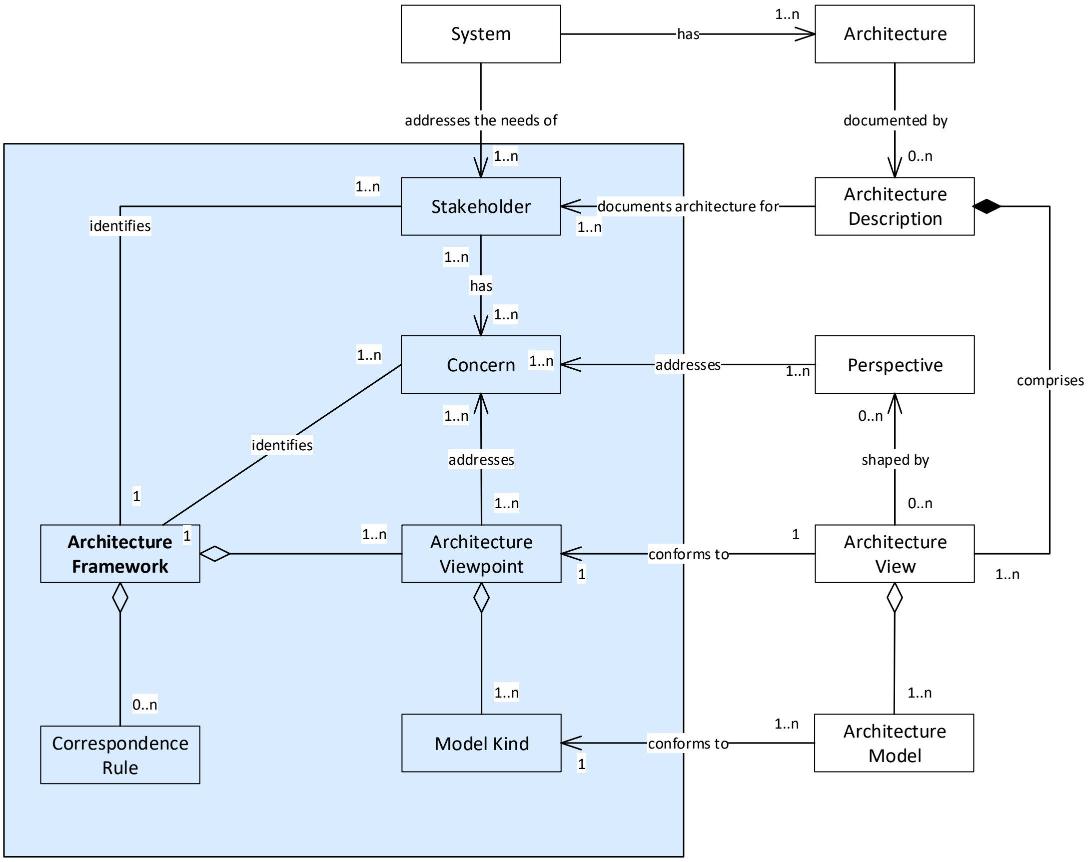


Figure 20: Architecture framework in context

To enable systematic architecture description of C-ITS, the ISO/IEC/IEEE42010:2011 is used to define the C-ITS architecture framework. As described in the definition of an architecture framework, the C-ITS architecture framework specifies stakeholders, their concerns, viewpoints, model kinds, and correspondence rules. In addition, architecture perspectives related to C-MobILE project will be addressed. C-ITS architects can use an architecture framework to represent the C-ITS reference architecture, concrete, implementation and pilot site architectures.

4.1.1. C-ITS stakeholders and Concerns

A *stakeholder* is an individual, team, or organization holding *concerns* for the system such as architect, designer, user, and authority [12]. A *concern* is any interest in the system such as functionality, structure, behaviour, and interoperability [12]. The C-ITS address the needs of following stakeholders.

Stakeholder	Description	Concerns
End-User		
Driver	The real final individual user of the (private or commercial) road network and a service [DITCM] using a motorized vehicle.	/ Functionality / Dependability / Interoperability / Security (Privacy) / Maintainability / Cost

PTW	Individual user using a motorbike, electric bicycle.	/ Functionality / Dependability / Interoperability / Security (Privacy) / Maintainability / Cost
Cyclist	VRU[1] using a bicycle.	/ Functionality / Security (Privacy) / Maintainability / Cost
Pedestrian	VRU, on foot.	/ Functionality / Security (Privacy) / Maintainability / Cost
Visually Disabled Pedestrian (VDP)	Disabled road user [VRUITS].	/ Functionality / Usability (Ease of use) / Security (Privacy) / Maintainability / Cost
Non-Motorized Vehicle User	Users in non-automated vehicles (wheelchairs, buggies, prams).	/ Functionality / Usability (Ease of use) / Security (Privacy) / Maintainability / Cost
Traveller	Individual user that may travel with more than one type of transportation means and uses a C-ITS service.	/ Functionality / Usability (Ease of use) / Dependability / Security (Privacy) / Maintainability / Cost
Fleet Operator	Manages a number of vehicles such as buses, emergency vehicles, trucks or taxi cars [DITCM].	/ Functionality / Usability (Ease of use) / Reliability / Maintainability
Road Works Operator	Road side recovery (tow trucks), Utilities (road sweepers, dustcarts), road works, forensics, etc.	/ Functionality / Dependability

[1] Any kind of vulnerable user.

		/ Interoperability / Security (Authenticity) / Maintainability / Cost
Technological Provider		
Original Equipment Manufacturer (OEM)	Manufacturer and suppliers for cars, trucks, trailers, caravans, commercial vehicles, etc.	/ Functionality / Maintainability / Security (privacy) / Interoperability / Cost
Telecom/Mobile Network Operator	• Network Operators • Infrastructure providers • Connected device and appliance manufacturers • Broadcast	/ Functionality / Maintainability / Security (privacy) / Interoperability / Cost
Maps, Navigation and Data Provider	• Map makers • Parking data providers • Real-time traffic info providers • Commercial information services • Data aggregators/analytics consultants	/ Functionality / Maintainability / Security (privacy) / Interoperability / Cost
C-ITS Service Provider	A private-sector entity which facilitates the C-ITS service by either setting up the required infrastructure in the absence of public infrastructure or the creation and upkeep of software in the absence of public software.	/ Functionality / Maintainability / Security (privacy) / Interoperability / Cost
Parking operator or Parking Service Provider	Can be both a Service Provider arranging reservation and/or payment in a parking space operated by itself or by a 3rd party. Can also be a traditional parking operator that does not facilitate C-ITS (e.g. payment with cash/bank card only)	/ Functionality / Maintainability / Security (privacy) / Interoperability / Cost
Public Transport Operator	Operates a fleet of public transport vehicles, trams, trains or other.	/ Functionality / Maintainability / Security (privacy) / Usability (Ease of use) / Reliability / Interoperability / Cost
Legal Authority		

Road Operator and National/ Local Authority	-Planners -Traffic management -infrastructure owners (tunnels, bridges) -maintainers - Local	/ Functionality / Maintainability / Security (privacy) / Interoperability / Usability (Ease of use) / Reliability / Cost
City or Municipality	Cities and municipalities.	/ Functionality / Maintainability / Security (privacy) / Interoperability / Usability (Ease of use) / Reliability / Cost
European Commission	Providing vision & legal environment.	/ Functionality / Security (privacy) / Interoperability
Policy Advisor, Consultancy, Public-Private Partnership	Provide advice and consultancy for matters regarding policies and vision e.g. ERTICO [53], IRU [54], and FIA [55].	/ Functionality / Security (privacy) / Interoperability

Table 3: C-MobILE stakeholders and their concerns

4.2. Architecture Viewpoints and Views

In this section, we propose a set of six core viewpoints as part of the C-ITS architecture framework: Enterprise, Functional, Information, Implementation, Physical, and Communication. The relationships between views created using these viewpoints are shown in Figure 21. These viewpoints are defined based on the existing literature especially definitions from [14] and ITS reference architectures discussed in the previous sections. We believe that this set of viewpoints enable structured architectural descriptions for the C-ITS systems.

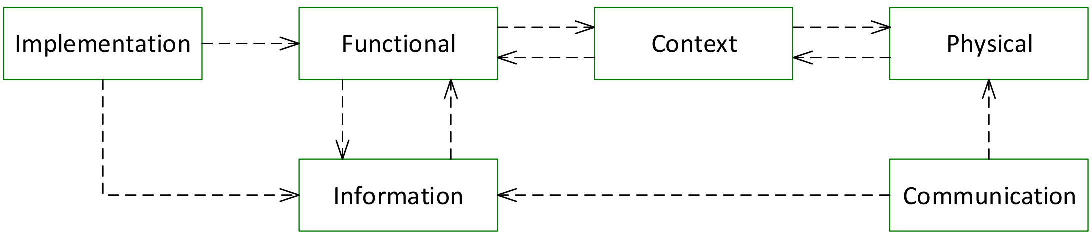


Figure 21: Relationships between C-ITS Architecture Views

Table 4 lists the definitions of these viewpoints and views.

Viewpoint	Definition	Respective View
Context	Describes the relationships, dependencies, and interactions between the system and its environment (people, systems, external entities interacting with the system).	A *context view* helps the system's stakeholders to understand system's responsibilities and how it relates to its organization.

Functional	Describes the system's runtime functional elements, their responsibilities, interfaces, and primary interactions.	A *functional view* helps the system's stakeholders understand the system structures and has an impact on the system's quality properties.
Communication	Describes the communications (e.g. interfaces, communication protocols) between different subsystems deployed on different hardware environment.	A *communication view* supports stakeholders involved in defining/enabling communication between systems.
Information	Describes how the architecture stores, manages, and distributes data and information.	An *information view* provides high-level view of static data structure and information flow to users, developers, testers, and maintainers.
Implementation	Describes the implementation for realizing functionality into real life software systems.	An *implementation view* supports stakeholders involved in building, testing, maintaining, and enhancing the system.
Physical	Describes the physical environment where the system will be deployed and the dependencies that the system has on elements of it.	A *physical view* supports stakeholders involved in deploying, testing, and maintaining the system by capturing the hardware environment that the system needs, the technical environment requirements for each element, and the mapping of software elements to the runtime environment that will execute them.

Table 4: Viewpoint Definition

As the result of the architecture analysis and reverse architecting process, we identified that the DITCM reference architecture covers all the C-MobILE pre-selected services except the "Bundle1: urban efficiency" services. We have extracted the reference architecture from existing reference architectures, which was consistent with the DITCM reference architecture. Therefore, the C-MobILE reference architecture is adopted from the DITCM reference architecture with the additional changes to enable the Bundle 1 services for urban efficiency. The architecture descriptions have three levels, each one refining the previous one: Reference, Concrete, Implementation, and Pilot Site Architectures. To put the architecture framework and architecture description concepts in context, C-MobILE extends the conceptual model of the ISO/IEC/IEEE 42010 international standard for architecture descriptions of systems and software [12] and uses the proposed concept of architecture perspective for shaping the architecture views [14].

In the sections 5 to 10, we present the details for each viewpoint with a guideline to follow when extending the models for each views.

4.3. Architectural Perspectives

We use architectural perspectives to ensure that our architecture exhibit key software quality characteristics as illustrated in

Figure 20. In this section, we identify the key perspectives for large scale demonstration of C-ITS systems. The identified perspectives can be applied to the views. Additional quality characteristics and perspectives may be added in the descriptions of concrete and implementation architectures.

		Perspectives						
		Interoperability	Security	Performance	Usability	Reliability	Availability	Adaptability
Views	Context						x	x
	Functional	Identify interoperable functions & Apply locate tactic	Identify vulnerable functions & Apply security policies	x	x			x

	Information	Determine syntax and semantics of data models to be exchanged	Information security (access control, access classes, object-level security) & Apply Information integrity constraints		x	x		x
	Implementation		Provide guidelines & constraints to developers to ensure security policy	x				x
	Physical	Mapping of software to hardware should take communication interoperability and other quality attributes into account	System's packaging & deployment environment and hardware choices	x	x	x		
	Communication	Resource management needs to be investigated	ETSI security entity & Interface security	x				x

Table 5: Applying Perspectives to C-ITS Views

Architectural perspectives are used to formalize and guide the process of evaluating and reviewing the architectural models to ensure that the architecture satisfies the required quality characteristics and to select architectural tactics when it does not.

We identified the architecture perspectives provided in Table 5 and summarize below the definitions of each perspective which are based on the international ISO/IEC 25010 standard [18]:

- Interoperability is degree to which two or more systems, products or components can exchange information and use the information that has been exchanged.
- Security is degree to which a product or system protects information and data so that persons or other products or systems have the degree of data access appropriate to their types and levels of authorization.
- Performance efficiency is performance relative to the amount of resources used under stated conditions.
- Usability is degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.
- Reliability is degree to which a system, product or component performs specified functions under specified conditions for a specified period.
- Availability is degree to which a system, product or component is operational and accessible when required for use.
- Adaptability is degree to which a product or system can effectively and efficiently be adapted for different or evolving hardware, software or other operational or usage environments.

We also provide architectural tactics for the interoperability and security perspectives so that they could be of use as an inspiration for elaborating the relevant tactics for the concrete, implementation, and pilot site architectures. Defining the perspectives for C-MobILE architecture definition and eventually large-scale demonstration of C-ITS systems would help avoid expensive changes in the later stages of development, therefore need to be further elaborated in D3.2 and D3.4 when quality attributes are further elicited and prioritized.

4.3.1. Interoperability Perspective

One of the main objectives of C-MobILE is large-scale, real-life C-ITS deployments interoperable across Europe. Interoperability is degree to which two or more systems, products or components can exchange information and use the information that has been exchanged [18]. This definition implies both syntactic and semantic interoperability meaning the ability to exchange data but also the ability to correctly interpret the data that is being exchanged [14]. Below we highlight the main tactics to ensure interoperability between C-ITS services:

- In the functional view, it is determined which system responsibilities need to be interoperate with other systems. In addition, the locate tactic can be used to enable service discovery. A *service* can be located by type of service, name, location or other attributes to be found at runtime. MobiNET uses the 'discover service' tactic to enable interoperability between services. Orchestrate and tailor interface tactics [14] can be used to enable interoperability as well.
- In information view, the syntax and semantics of the main data models that need to be exchanged are determined. The syntax and semantics of the data models need to be consistent with the interoperable systems. If they data models are confidential, the proper transformations need to be made.
- Physical view addresses the mapping of software to hardware components/systems. The components that interact with external systems need to be ensured to have the proper communication and should take other quality attributes e.g. performance, availability and security into account.
- Communication view should cover resource management e.g. ensuring the interoperation with another system does not exhaust critical system resources. Resource load for the communication requirements remains acceptable. In addition, binding time to both known and unknown external systems need to be acceptable as well.

4.3.2. Security Perspective

For C-ITS, security is an essential part of the system design, therefore applying security perspective across multiple viewpoints will ensure the security across the C-ITS. Below it is elaborated how security perspective affects each view [14] and how main security principles could be leveraged as architectural tactics for ensuring security for C-ITS:

- In the functional view, it is identified which functional elements need to be protected. On the other hand, security policies may impact the functional structure as well. A security policy defines the set of security-related constraints that the system should be able to enforce.
- In the information view, it is identified which (sensitive) data needs to be protected. Information models can be modified as a result of security design. Information access policy in terms of different types of principles the system has (e.g. administrators, users, public authority) and for different type of information (vehicle information – e.g. location and speed, device information), and different type of access each group requires (e.g. only view the information, change it, share it, delete it). Information integrity constraints must be defined and determined if the communicated data needs to be signed or encrypted.
- In the implementation view, guidelines and constraints need to be provided to the developers to enable security policies. A security policy for the implementation view needs to define how the execution of certain sensitive system operations (e.g. system shutdown) will be controlled.
- In the physical view, it is identified which elements need to be packaged and deployed into one system as a result of security design. Security design has also impact on the system's deployment environment and hardware choices.
- In the communication view, the security entity of the ETSI communication reference architecture is applied. The security entity provides security services to the OSI communication protocol stack, to the security entity and to the management entity [2]. Its overview is shown in Figure 22.

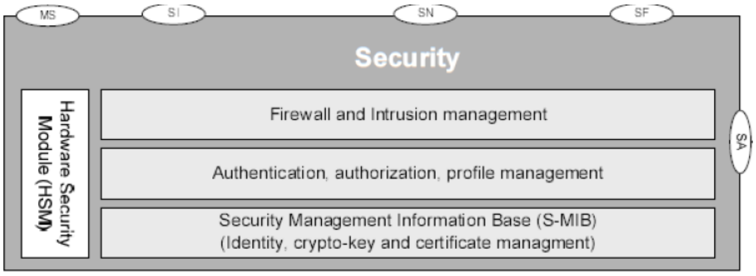


Figure 22: Security entity of the ETSI communication reference architecture [2]

In addition, the design choice needs to be made if a communication channel has to be secured e.g. by using TLS defined.

The summery of the security tactics is provided in Table 5.

4.4. Architectural Representation

We propose to use Systems Modelling Language (SysML) diagram types to for architectural notations of the C-ITS architectures. The SysML is a general purpose modelling language for engineering systems and consists of structure diagram, requirement diagram, and behaviour diagram types as shown in Figure 23 [47].

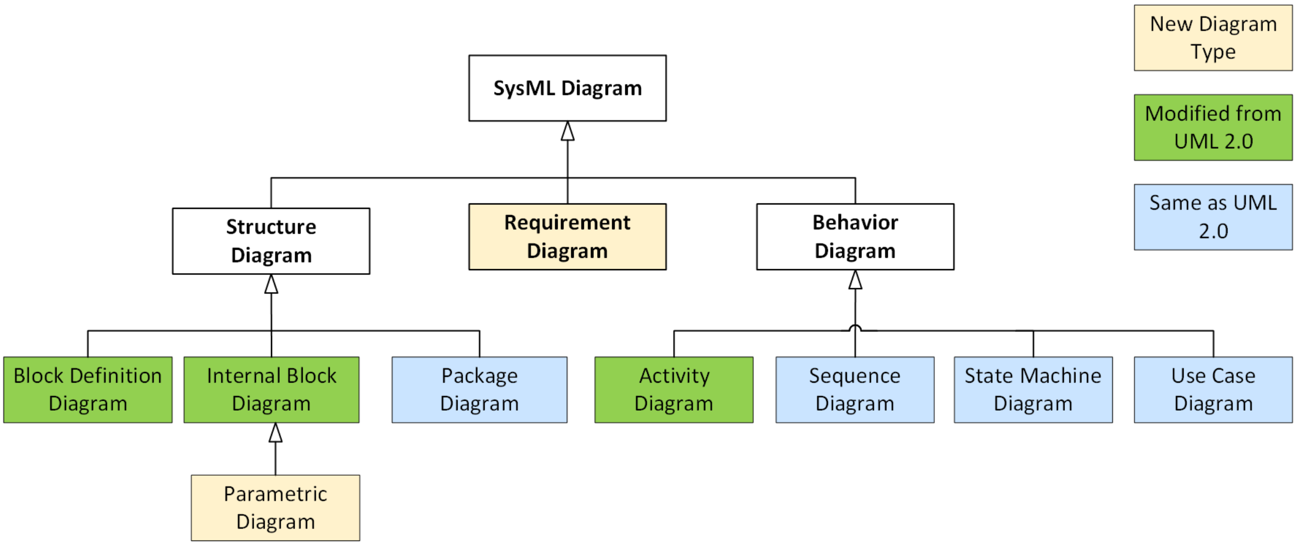


Figure 23. SysML Diagram Types

The SysML diagram types for each architectural viewpoint will be briefly presented in the later sections.

5. Context Viewpoint

5.1. Definition

The context viewpoint describes the relationships, dependencies, and interactions between the system and its environment (e.g. people, systems and external entities) [14]. The context view conforms to the context viewpoint and helps system's stakeholders (e.g. system/software architects, designers, developer and users) understand the system context.

An architect usually needs to include a definition of the system's context as part of their architectural description. The reasons for this include:

- the system context simply not being included in the requirements gathering exercise;
- a system context being loosely defined by requirements analysts, but at a level of detail which means that the architect needs to add significantly to it; and
- the system/software architect needing to reference elements of the system context elsewhere in the architectural description, so making it desirable for this information to be part of the architectural description and under the control of the architect.

5.2. Stakeholders and Concerns

All systems exist in some larger environment, be it a department, an organisation's IT environment, a mobile communications system or even a virtual world. The context view aims to answer questions about this environment and specifically the technical relationships that the system being designed has with the various elements of this wider environment. The concerns that a context view addresses discussed below.

System Scope and Responsibilities

This concern defines system scope briefly and lists high level list of the system's main responsibilities. The system requirements are elicited as part of the requirements analysis, thus this concern do not need to include complete requirements. In addition, some functional exclusions could be highlighted for clarity.

Identity and Responsibilities of External Entities

The key information that the context view must define is the set of external entities that the current system interacts with in some way, the reason for the interaction and the responsibilities that the external entities are assumed to fulfil in the context of this relationship. It is important to make sure that external entities that the system has irregular or occasional interactions with (e.g. systems that are only polled for data at the end of each month) are defined just as carefully as those which the system interacts with continually. Similarly, it is important to consider and carefully define external entities that rely on this system as well as those that this system relies on (it is very easy to worry about what we need while rather neglecting what others are expecting from us!) Also make sure that different types of external entities are considered, including systems supplying or consuming data, systems called as services, systems that call us as services, physical entities such as reports and files and human actors who need to interact with this system.

External Interdependencies and Connections

There are sometimes inter-dependencies between external entities that the system interacts with. An example could be where two systems have a data dependency between them that means that new data should always be sent to one of the systems, and acknowledged, before related data is sent to the other. These dependencies may be subtle and must be identified as part of this process.

Expected External Interactions

Having defined the external entities the next concern is to decide or discover the nature of the connections with them. Connections can vary widely, from high volume messaging or RPC connections, through batch-oriented file or database interfaces, to manual connections involving human interaction or even document scanning. Some connections may need to be secured; some may need to implement very specific protocols. Defining and agreeing the fundamental characteristics of each connection allows the architect to start thinking about their practical implications and helps to identify gaps in knowledge and potential problems.

5.3. Context View

The context view conforms to the context viewpoint. SysML Block Definition Diagram (BDD) is identified as suitable modelling notation for capturing the context viewpoint. SysML Use Case diagram can be used to show the usage of a system. For the reference architecture, the use case diagram is too early to be defined, since the use case list is still being defined.

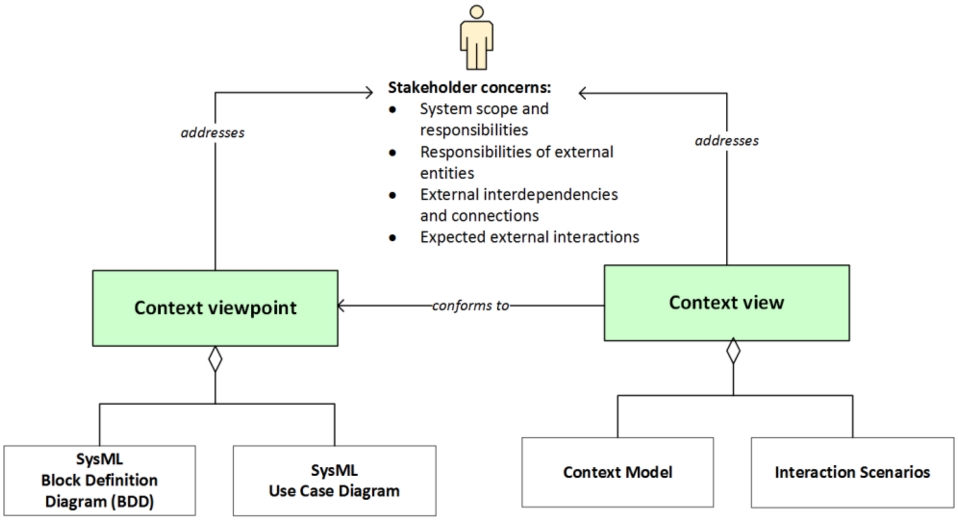


Figure 24: Context viewpoint and view

As illustrated in Figure 24, we define the context model to address the stakeholder concerns on the system scope and responsibilities, interdependencies and interactions.

5.4. Model kinds

SysML Block Definition Diagram and Use Case Diagram can be used for representing the context models. The context diagram is the key model within a context view, placing the system in its environment by relating it to the external actors that it interacts with via explicit relationships that represent the connections to and from it. A context diagram is usually quite simple and contains elements of the following types:

- System – the system being designed, which is treated as a "black box", with its internal structure hidden.
- External Entities – systems, people, groups and other entities that the system interacts with.
- Connections – the interfaces, protocols, and connectors that link the external entities and the system being designed or utilized.

5.4.1. Context Model

The notations that we commonly see used for context diagrams are SysML Block Definition Diagram. C-MobILE consists of five main systems which is depicted as black box and corresponding actors's connections with those systems is shown in Figure 25 below.

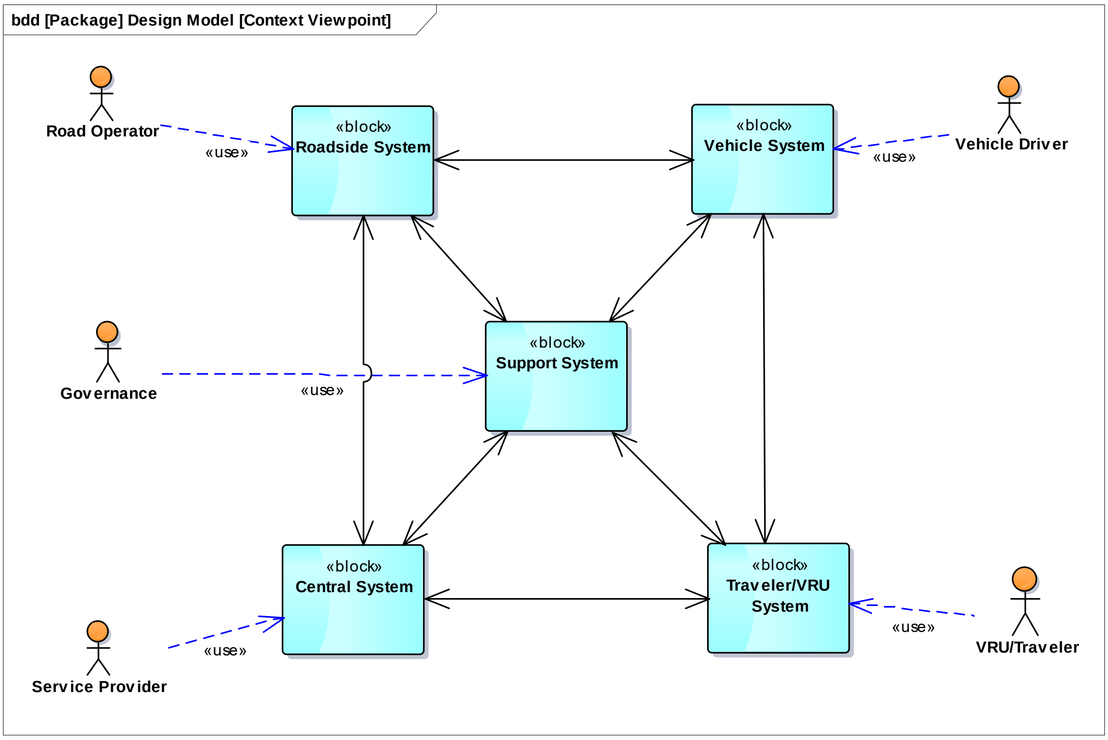


Figure 25: Context model

These five main systems of the C-MobILE architecture are aligned with the five layers of the DITCM reference architecture [48]:

- *Support System*: Comprise of sub-systems performing various tasks e.g.: governance, test and certification management, security and credentials management.
- *Central System*: Comprise of sub-systems to support connected vehicles, field and mobile devices. The sub-systems can be aggregated together or geographical or functionally distributed.
- *Roadside System*: Comprise of sub-systems which covers the ITS infrastructure on or along physical road infrastructure, e.g.: roadside units, signal/lane control etc.
- *Vehicle System*: Comprise of sub-systems which are integrated within vehicle such on-board systems (advanced driver assistance / safety systems, navigation, remote data collection or information).
- *Traveler/VRU System*: Comprise of both personal devices (e.g.: mobile devices, navigations devices) and specific systems connected to vehicles of VRU's (e.g.: tags).

(Human) actors are treated as external entities that interact with the systems:

- *Vehicle Driver*: An actor driving in a vehicle. The vehicle is a motorized vehicle (car, bus, truck) and not a vehicle of a vulnerable road user (bike, moped, motor). An actor in this category is directly concerned with Vehicle System as shown in Figure 25 through various vehicle related interfaces like : OBU , HMI etc.
- *Vulnerable Road User*: A VRU is a human actor like a pedestrian, cyclist or PTW driver; A motorcyclist is also an example of a PTW and is treated as a vulnerable road user in specific road hazard situations with other cars. An actor in this category is directly concerned with Traveler/VRU System as shown in Figure 25 through various interfaces like HMI, tablet, mobile, etc.
- *Road Operator*: An actor responsible for the traffic management of a road network. An actor under this category is directly concerned with Roadside System through various communication channels and is responsible for collecting and evaluating data related to roadside information.
- *Service Provider*: An actor (organization) supplying services to one or more customers. Customers are either other organizations, including government (B2B / B2G / G2B / G2G) or end users (B2C / G2C). An actor under this category is directly concerned with Central System which also is responsible providing various services. Typical examples of a Service Provider are a Navigation Provider as a Service Provider providing navigation services to end users or organizations or a Traffic Information Provider as a Service Provider that provides road traffic related information, like traffic jams, incidents, road works warning etc.

- *Governance*: An actor under this category is directly concerned with Support System whose responsibility is to support various other sub-systems such as legal authorities, test and certification management, security and credentials management etc.

5.4.2. Interaction Scenarios

The expected interactions between the system and the external entities can be described in the context diagram to identify design constraints and un-elicited requirements. When the system usage is unclear or conflicting requirements are identified, it can be useful to make interaction scenarios. SysML Use Case Diagram can be used to give a visual overview of the functionalities provided by a system in terms of actors, their goals (i.e. use cases), and any dependencies among the use cases if the use case list is available. During the definition of the C-MobILE reference architecture, the definitions of use cases are not finalized yet, thus it could be used in the later stages of architecture definitions.

5.5. Correspondence rules

The context viewpoint has correspondences with functional and physical viewpoints as highlighted in Figure 26**¡Error! No se encuentra el origen de la referencia.**.

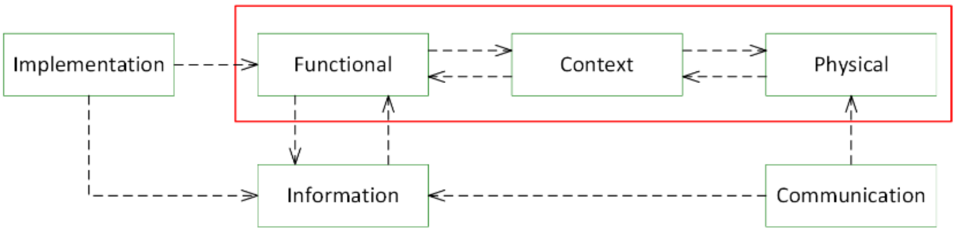


Figure 26: Correspondences for the Context Viewpoint

Functional and physical viewpoints have conformance correspondence to the context viewpoint.

6. Functional Viewpoint

6.1. Definition

We define the concepts for the functional viewpoints, which can be used for modelling functional structures for different type of architectures. Functional viewpoint describes the system's runtime functional elements, their responsibilities, interfaces, and primary interactions. The functional view conforms to the functional viewpoint, helps the system's stakeholders understand the system structures, and has an impact on the system's quality properties.

6.2. Stakeholders and concerns

All stakeholders (particularly system architects, developers and integrators) use the functional viewpoint as it is usually easy to understand and describes the system's functional structure. The functional viewpoint addresses the following concerns of the stakeholders.

Functional capabilities

Functional capabilities describe the functionality that the system is required to provide. In the Architecture Viewpoint for the reference architecture, it defines what the system will explicitly do. The set of functional and quality requirements will be used to define the functionality.

External interfaces

External interfaces address the interactions between systems e.g. based on data/control flow or events. A system can send or receive data either because of an internal state change or a state change in another system. A system can send or receive a request to perform a task or notify that something has been occurred. In the functional view of the reference architecture, high-level generic interfaces are described. The interface syntax and semantics in further detail will be described in the concrete and implementation architectures.

Internal structure

A system can be further decomposed into subsystems or components to meet its requirements. Its decomposition has an impact on the quality attributes e.g. on security, scalability, reliability, and availability. Therefore, the definition of the internal structure should be taken both functional and quality requirements into account. In the functional view of the reference architecture, a system is further decomposed into subsystems, which can be further decomposed into functional components in concrete/implementation architectures.

6.3. Functional view

As presented in Section 4.2, an architecture description includes one Architecture View for each Architecture Viewpoint. Thus, the functional view conforms to the Functional Viewpoint. SysML Block Definition Diagram and Internal Block Definition Diagram are identified as suitable modelling notations for capturing the functional viewpoint.

As illustrated in Figure 27, we define *functional structure model* to address the stakeholder concerns for functional capabilities and external interfaces and *functional internal structure model* to address the internal structure of the element. SysML BDD and IBD can be both used to model the functional structure and the internal structure.

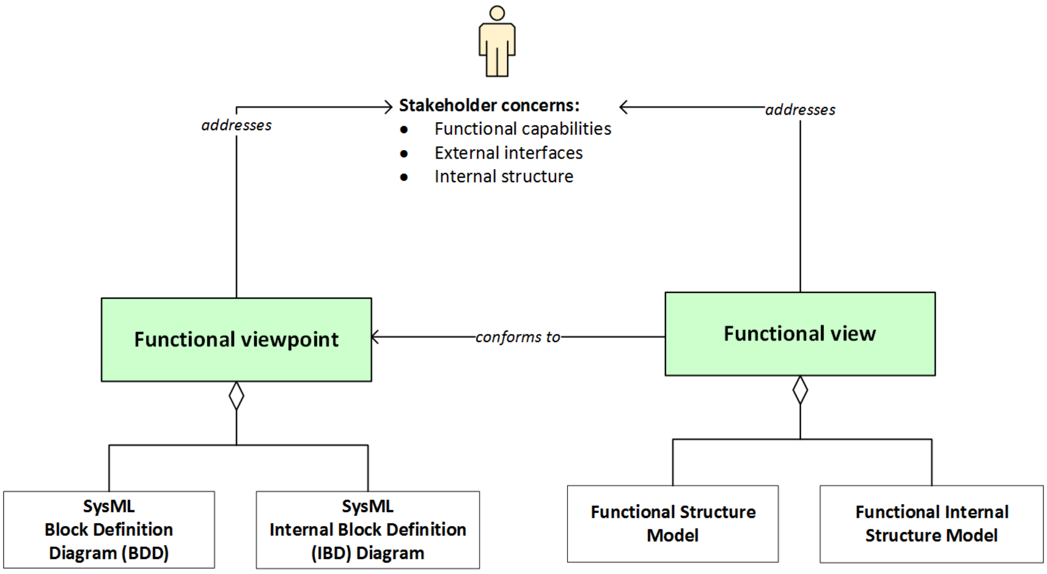


Figure 27. Functional viewpoint and view

The structure of a system is captured in functional structure models using SysML BDD by categorizing into systems and decomposing a system into subsystems. A system defines the functionality and functional data flow interfaces between systems that are required to support a particular ITS application.

6.4. Model Kinds

Below we describe SysML block definition diagram and internal block diagram, which are used for capturing the models for functional structure and internal structure of the system. The Block Definition Diagram and Internal Block Diagram are part of the structure diagram.

6.4.1. Functional Structure Modelling

The SysML Block Definition Diagram provides a black box representation of a system block. In SysML, BDD is the replacement of the UML 2.0 class diagram by replacing classes with blocks, which can be of any type including software, hardware. For representing the functional structure model, blocks, their contents and relationships are shown using the BDD.

The following BDD concepts are used for modelling the functional structure of the block/system:

- Block is a modular unit of system structure that encapsulates its contents [47] which we use to represent a system in the context of C-ITS reference architecture. A system is further decomposed into subsystems or functional components. A block can both provide and require Interfaces for both information and physical flows.
- A relationship between blocks can be represented by a composition (“has a” relationship) with a solid diamond or a reference with an open diamond.
- The flow port is a new definition from SysML. It represents what can go through a block in and/or out (e.g. data, energy), which will be of use in concrete and implementation architectures.

Functional viewpoint describes the system’s runtime functional elements, their responsibilities, interfaces, and primary interactions. The functional view conforms to the functional viewpoint, helps the system’s stakeholders understand the system structures, and has an impact on the system’s quality properties. The system structure is captured in functional models using SysML block definition diagram (BDD) and by categorizing into systems and decomposing a system into subsystems. A system defines the functionality and functional data flow interfaces between systems that are required to support a particular ITS application. Information flows depict the exchange of information between subsystems. **¡Error! No se encuentra el origen de la referencia.** Figure 28 shows the high-level functional architectural model for the C-MobILE applications.

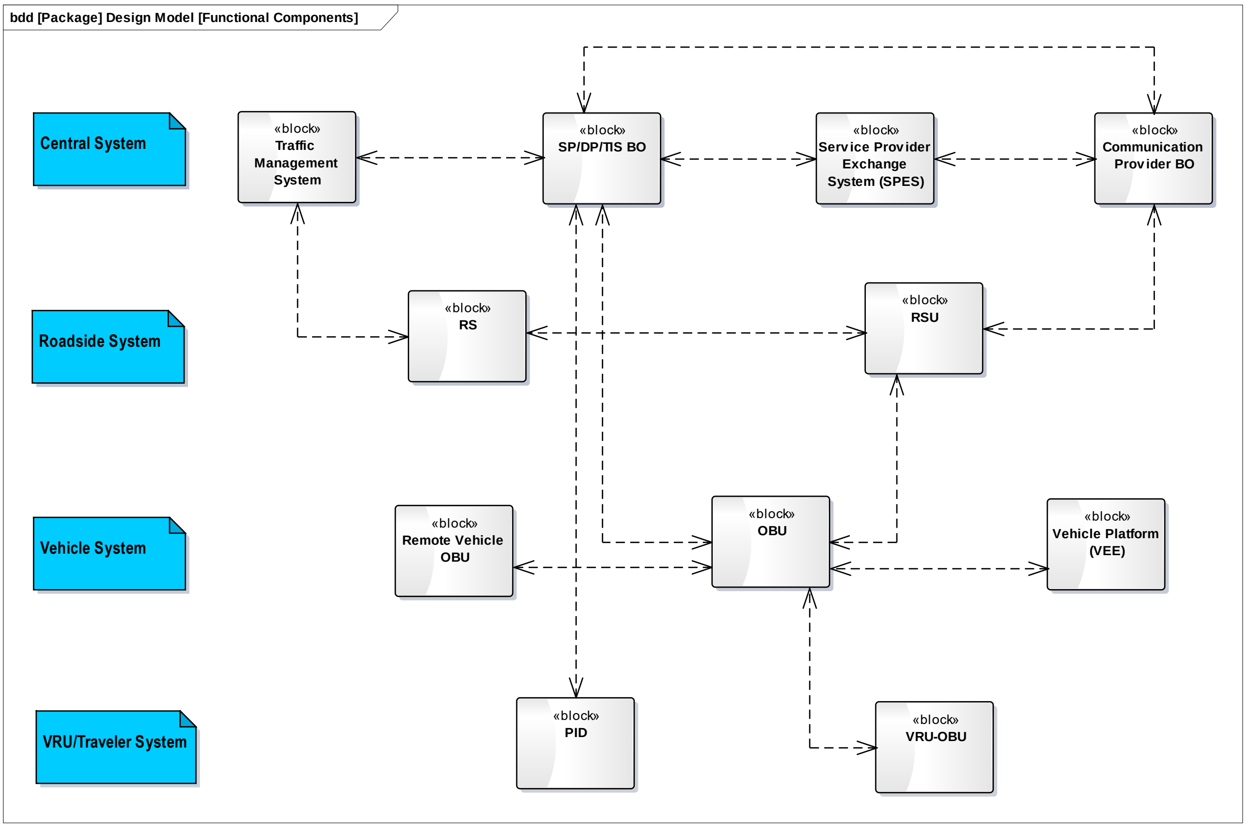


Figure 28: High-level functional model of the C-ITS systems for C-MobILE

The main functions of the systems are sensing, communication, situation monitoring and situation assessment, and acting and trust management. The hierarchic decomposition of these functions is based on geometrical and temporal scale of information and information abstraction. Cardinality (multiple entities with the same functionality) of the system is supported and dependent on physical limitations of sensors and actuators and heterogeneity of goals.

6.4.1.1. Central System

In Figure 29**¡Error! No se encuentra el origen de la referencia.**, the highest-level functional structure of the Central System is illustrated. The Central System consists of the following main subsystems which will be further refined in the concrete architecture by defining its functional components and interfaces between them: TMS, SP/DP/TIS, CP, SPES and IIS.

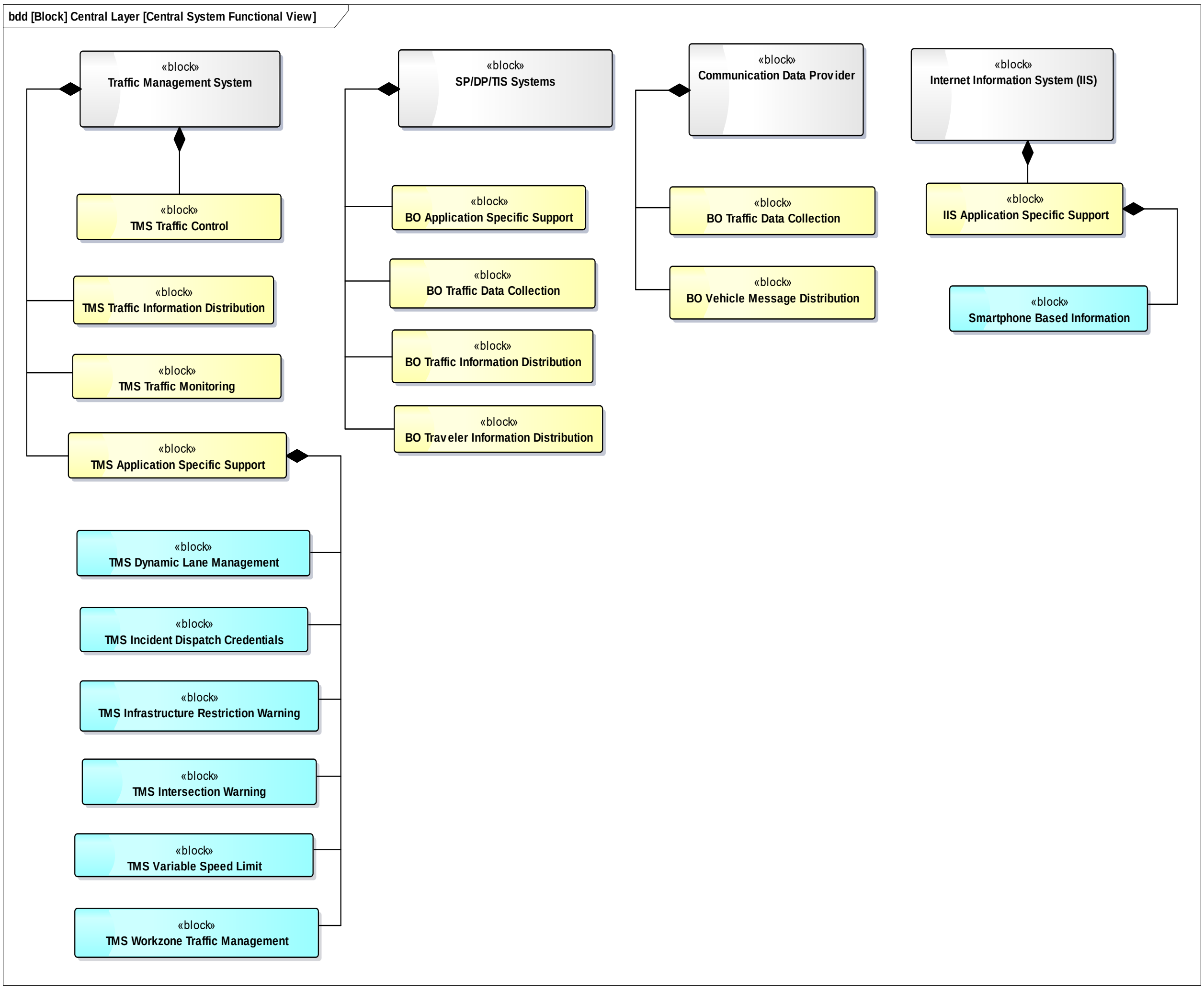


Figure 29: High-level functional model of the Central System

System	Subsystem	C-MobILE Application	Description
TMS	TMS Traffic Monitoring	Road Works Warning, Road Hazard Warning	Remotely monitors traffic sensors and surveillance equipment (cameras), and collects, processes and stores the collected traffic data.
	TMS Traffic Control	Green Light Optimal Speed Advisory (GLOSA)	Controls driver information system field equipment including dynamic message signs, managing dissemination of driver information through these systems
	TMS Traffic Information Distribution	Navigation related services	Disseminates traffic and road conditions, dynamic speed limits, closure and detour information, incident information, driver advisories, and other traffic-related data to other centers, the media, and driver information systems
	TMS Intersection Safety	Signal violation warning Warning system for pedestrian	Controls and monitors Roadside Units (RSUs) that support stop sign, red light, and pedestrian crossing violations. Configures the RSUs for the current intersection geometry and traffic signal control equipment at the intersection

	TMS In-Vehicle Signing Management	In-Vehicle signage	Sign information that may include static regulatory, service, and directional sign information as well as variable information such as traffic and road conditions can be provided to the RSU, which uses short range communications to send the information to in-vehicle equipment
SP/DP/TIS	Back Office (BO) Traffic Data Collection	Probe Vehicle Data	Current traffic information and other real-time transportation information are collected from several sources like TMS, and connected vehicles
	BO Traffic Information Distribution	Navigation related services	Disseminates traffic and road conditions, closure and detour information, incident information, driver advisories, and other traffic-related data to other centers and the media (e.g. radio, Service Providers).
	BO Traveller Information Distribution	Navigation related services	Disseminates traveller information including event information, transit information, parking information and weather information.
	BO Application Specific Support	Motorway/Urban Parking	Disseminates traveller information including urban and motorway parking information.
Communication Provider	BO Traffic Data Collection (from RSU Traffic and situation monitoring)	V2I applications with RSU	Current traffic flow information and other real-time information are collected from equipped cooperative vehicles passing the roadside station of the communication provider.
	BO Vehicle Message Distribution	I2V applications with RSU	Receives information including traffic and road conditions, incident information, maintenance and construction information, event information, transit information, parking information, and weather information.
SPES	Service Directory (SD)		Provides basis capabilities to manage and search service descriptions
	Identity Manager (IM)		Provides capabilities to manage common identities and to handle all security and privacy related concerns
	Billing		Support system that handles all financial transactions and provides a neutral instance which monitors the transactions between different parties.
IIS	IIS Application Specific	V2I, VRU specific services	A VRU or Vehicle Connected System can request for specific information or send information to the IIS

Table 6: Central Sub-systems Functional Descriptions

6.4.1.2. Roadside System

In Figure 30**¡Error! No se encuentra el origen de la referencia.**, the highest-level functional structure of the Roadside System is illustrated. The Roadside System consists of the following main subsystems which will be further refined in the concrete architecture by defining its functional components and interfaces between them: Roadside and Roadside Units.

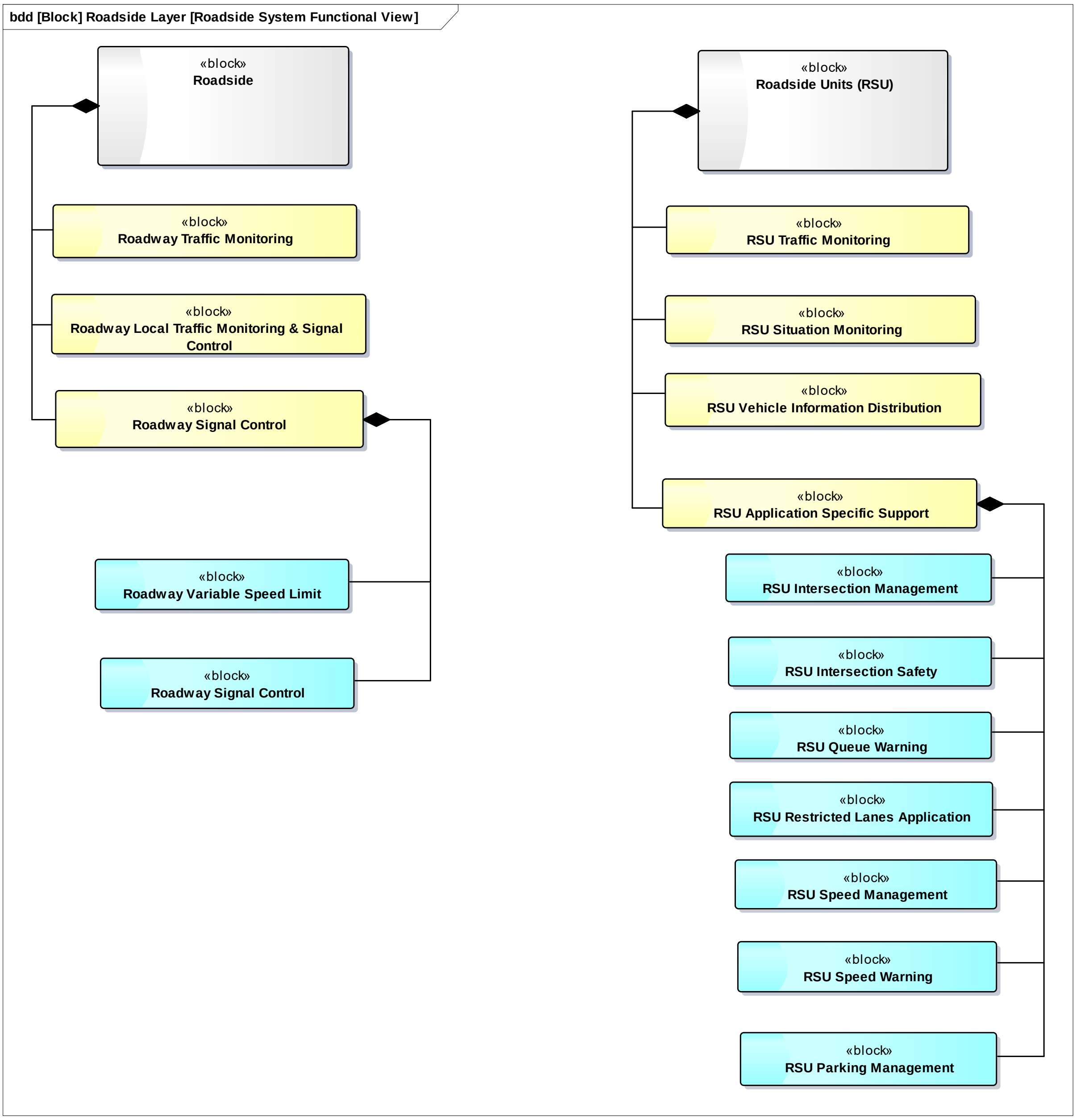


Figure 30. High-level functional model of the Roadside System

System	Subsystem	C-MobILE Application	Description
RSU	RSU Traffic Monitoring	V2I applications via V2V monitoring	Vehicle V2V safety messages that are shared between connected vehicles and distils this data into traffic flow measures that can be used to manage the network in combination with or in lieu of traffic data collected by infrastructure-based sensors. Also supports incident detection by monitoring for changes in speed and vehicle control events that indicate a potential incident
	RSU Situation Monitoring	Probe Vehicle Data	This object collects current status information from local field devices including intersection status, sensor data, and signage data, providing complete, configurable monitoring of the situation for the local transportation system in the vicinity of the RSU.

	RSU Vehicle Message Distribution	I2V applications with RSU	Receives information from the CP BO. Location-specific or situation-relevant information is sent to short range communications transceivers at the roadside
	RSU Intersection Management		It communicates with approaching vehicles and ITS infrastructure (e.g., the traffic signal controller) to enhance traffic signal operations.
	RSU Intersection Safety	Signal Violation Warning, Warning system for pedestrian	It communicates with approaching vehicles and ITS infrastructure to alert and warn drivers of potential stop sign, red light, and pedestrian crossing conflicts or violations.
	RSU Queue Warning	V2I communications	It monitors connected vehicles to identify and monitor queues in real-time and provides information to vehicles about upcoming queues, including downstream queues that are reported by the Traffic Management System
	RSU Speed Management	Slow or Stationary Vehicle Warning, Motorcycle approaching indication	Provides infrastructure information including road grade, roadway geometry, road weather information, and current speed limits to assist vehicles in maintaining safe speeds and headways.
	RSU Speed Warning		This application object works in conjunction with the 'Roadway Speed Monitoring and Warning' application object, which uses traditional ITS field equipment to warn non-equipped vehicles.
	RSU Parking Management	Urban/Motorway Parking	Monitors the basic safety messages generated by connected vehicles to detect vehicles parking and maintain and report spaces that are occupied by connected vehicles.
RS	Roadway Traffic Monitoring	Incident warning , Traffic Jam ahead	Monitors traffic conditions using fixed equipment such as loop detectors and cameras.
	Roadway Signal Control	GLOSA, In-Vehicle Signage	Monitor and control signalized intersections/ramps and dynamic roadway signs
	Roadway local traffic monitoring and control distribution	GLOSA, In-Vehicle Signage	Receive information from the Roadway Traffic Monitoring and send this information via a RSU to vehicles or BO systems.

Table 7: Roadside Sub-systems Functional Description

6.4.1.3. Vehicle System

In Figure 31**¡Error! No se encuentra el origen de la referencia.**, the highest-level functional structure of the Vehicle System is illustrated. The Vehicle System consists of the following main subsystems which will be further refined in the concrete architecture by defining its functional components and interfaces between them: Vehicle Electrical & Electronic System and On Board Unit.

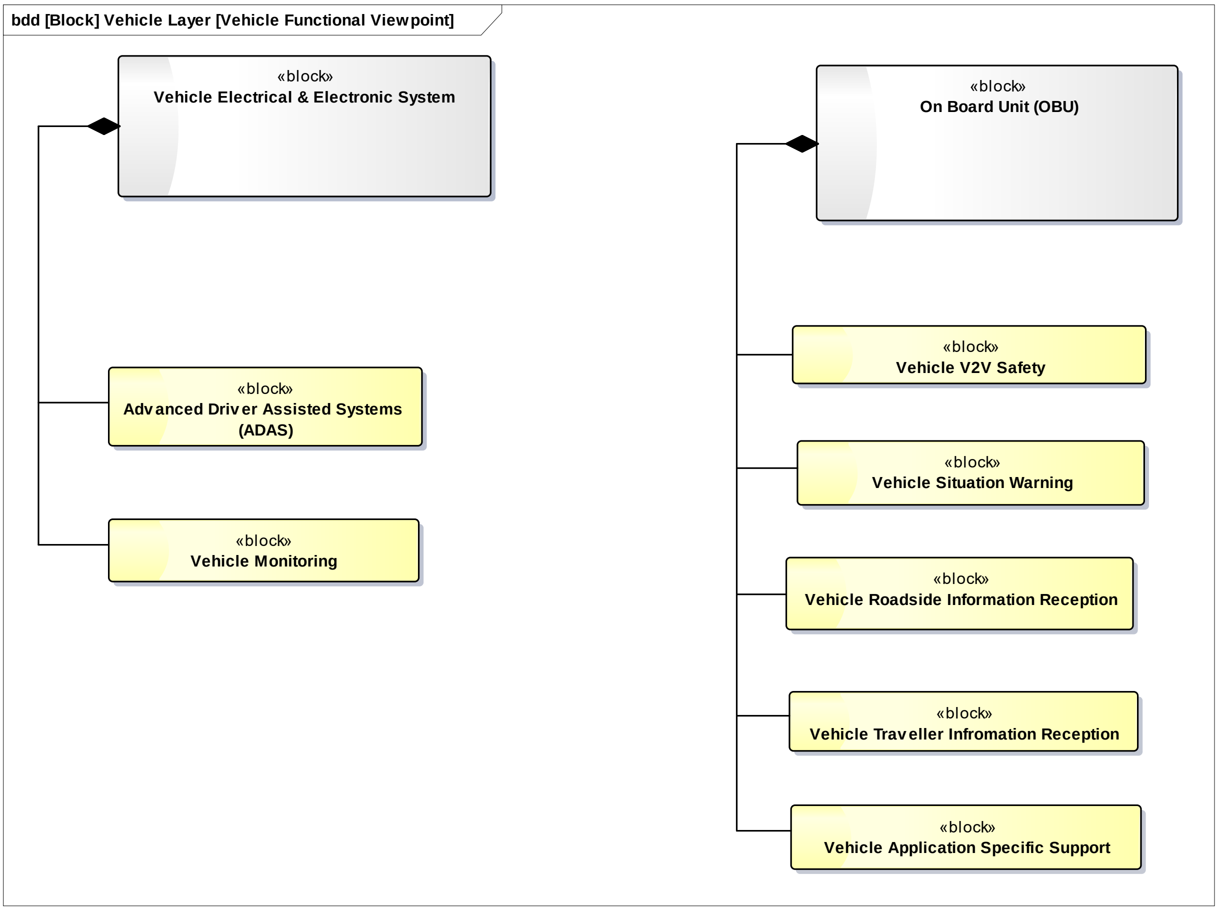


Figure 31. Vehicle functionality decomposition

System	Subsystem	C-MobILE Application	Description
OBU/ RV-OBU	Vehicle V2V Safety	Road Works Warning, Road Hazard Warning, Emergency Vehicle Warning	Exchange current vehicle location and motion information with other vehicles in the vicinity. The information is used to calculate vehicle paths, and warns the driver when the potential for an impending collision is detected.
	Vehicle Situation Monitoring		Collect traffic data and environmental situation data from on-board sensors and systems related to environmental conditions and sends the collected data to the infrastructure (V2I) as the vehicle travels.
	Vehicle Roadside Information Reception	I2V applications with RSU	Information presented may include fixed sign information, traffic control device status (e.g., signal phase and timing data), advisory and detour information, warnings of adverse road and weather conditions, travel times, and other driver information.
	Vehicle Traveller Information Reception	I2V applications with SP BO Motorway/Urban Parking	General traveller information including traffic and road conditions, incident information, maintenance and construction information, event information, transit information, parking information, weather information, and broadcast alerts.
	Vehicle Application Specific Support	Cooperative cruise control	Representation of the functionality required in the vehicle to execute a specific application e.g. cooperative adaptive cruise control, rerouting etc.

Vehicle Electrical and Electronic system (VEE)	Advanced Driver Assisted Systems (ADAS)	Cruise control	Vendor-specific assistance systems to increase safety and comfort of the driver. Examples are lane departure warning, automatic emergency brake, and advanced cruise control.
	Vehicle Monitoring		Access to vehicle-specific sensor and actuator information systems of the vehicle

Table 8: Vehicle Sub-systems Functional Description

6.4.1.4. Traveller/VRU System

In Figure 32**¡Error! No se encuentra el origen de la referencia.**, the highest-level functional structure of the Traveller/VRU System is illustrated. The Traveller/VRU System consists of the following main subsystems which will be further refined in the concrete architecture by defining its functional components and interfaces between them: Personal Information Devices.

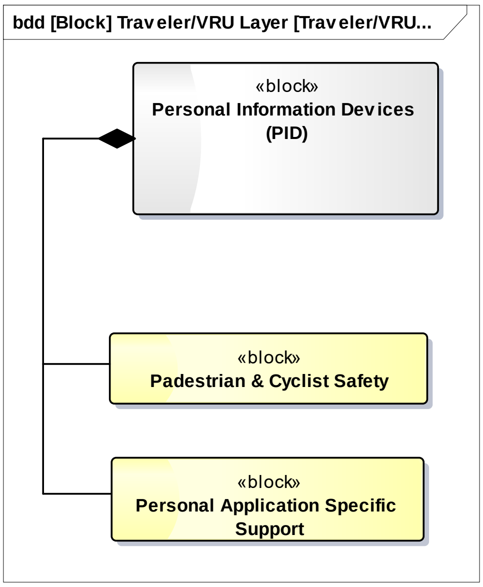


Figure 32. Block Definition Diagram for VRU Functions

System	Subsystem	C-MobILE Application	Description
PID	Personal Pedestrian and Cyclist Safety	All VRU applications	Providing pedestrian and cyclist location information to the infrastructure that can be used to avoid collisions involving pedestrians/cyclists.
	Personal Application Specific Support	Navigation related applications << add specific services>>	Personal Interactive Traveller Information provides traffic information, road conditions, transit information, yellow pages (traveller services) information, special event information, and other traveller information that is specifically tailored based on the traveller's request and/or previously submitted traveller profile information.

Table 9: Traveller/VRU Sub-systems Functional Description

6.4.2. Functional Internal Structure Modelling

The SysML Internal Block Definition Diagram provides an internal or white box representation of a system block. UML 2.0 composite structure diagram, the SysML IBD redefines the composite structure diagram by supporting blocks and flow ports. BDD can be used in combination with IBD i.e. functional structure of the system is represented as trees of modular systems/subsystems, which further refined into the representation of final assembly of all blocks/systems. Blocks that can be further decomposed into IBD, are called composite blocks.

The following IBD concepts are used for modelling the internal structure of a block/system:

- BDD composite blocks are further decomposed into subsystems or functional components (which are usually called in IBD as *parts*). Block or system that is further decomposed into subsystems or functional components are connected via standard ports with exposed interfaces and/or flow ports.

- The item flow is a new definition from SysML as well. It represents the things that flow between blocks/systems and/or subsystems across their connectors, which will be of use in implementation architecture.

6.5. Correspondence rules

The functional viewpoint has correspondences with context, implementation, and information viewpoints as illustrated in Figure 33**¡Error! No se encuentra el origen de la referencia.**.

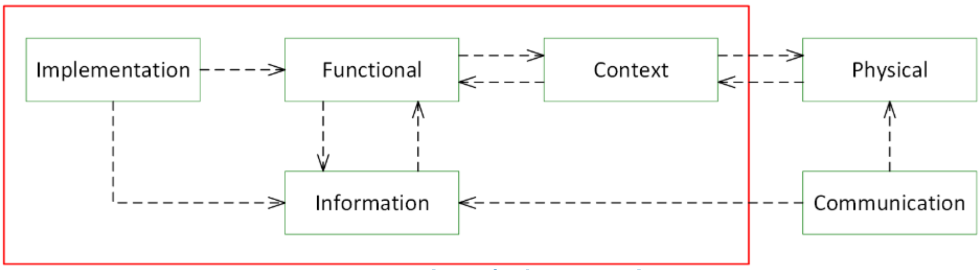


Figure 33: Correspondences for the Functional Viewpoint

Functional viewpoint refines the context, information, and implementation viewpoints needs to conform to the functional viewpoint.

7. Communication Viewpoint

7.1. Definition

Communication viewpoint describes the mode of communication through network interfaces and communication protocols between different systems deployed on different hardware environment. The communication view supports stakeholders involved in defining/enabling communication between systems and shows the interfaces between systems and sub-systems. The communication architecture for C-MobILE conforms to the general communications reference architecture defined in ETSI EN 302 665 which is illustrated in Figure 35.

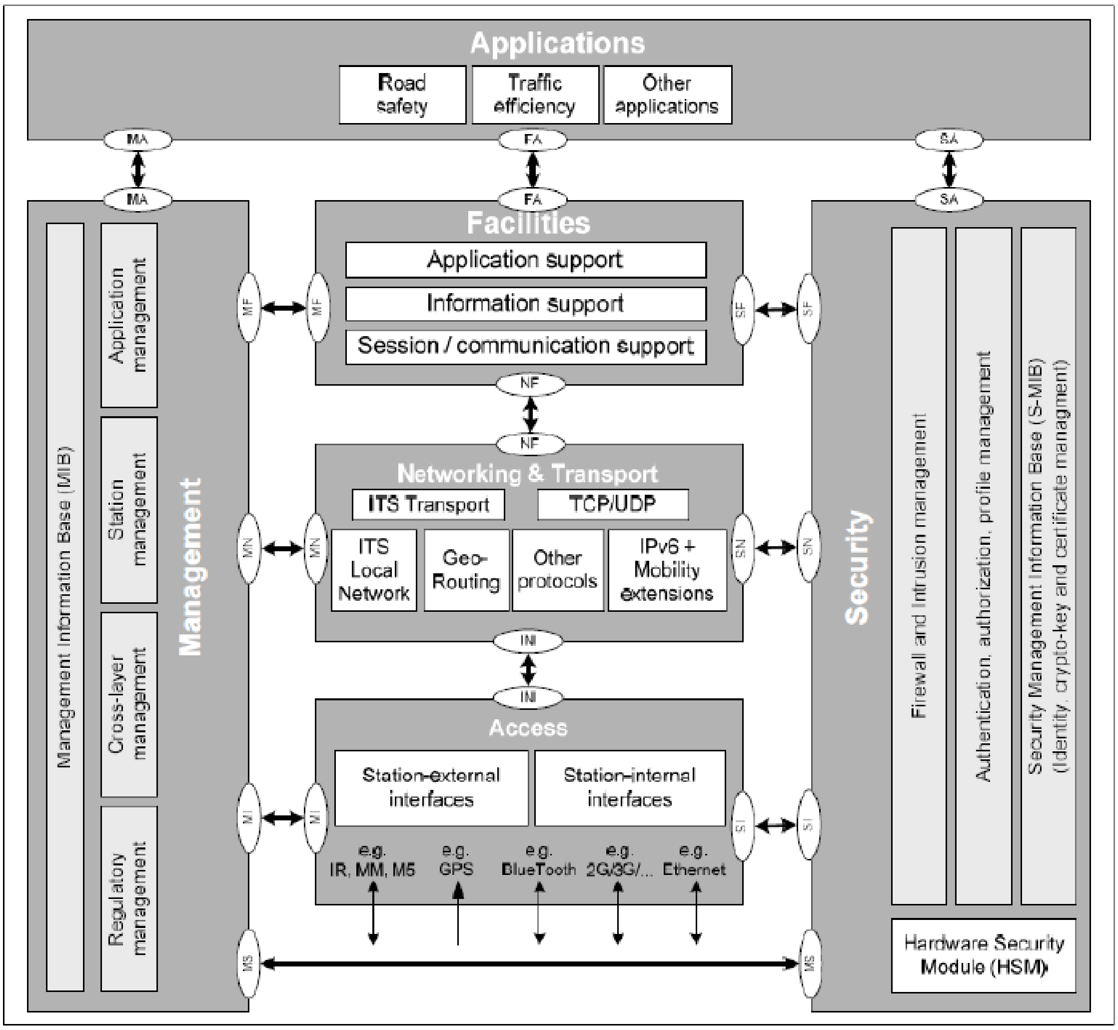


Figure 34. ITS station reference architecture [2]

The reference communication architecture for C-ITS stations consists of four main functional domains namely applications, facilities, network & transport, and access and two support domains (security and management) [2]. This reference communication architecture is valid for all ITS systems, i.e. OBU, RSU and BO systems which are respectively named vehicle ITS, roadside ITS and central ITS in the ETSI definitions.

The ETSI communication reference architecture defines six generic entities [2]:

1. Applications (re)presents the ITS-S applications making use of the ITS-S services to connect to one or more other ITS-S applications. An association of two or more complementary ITS-S applications constitutes an ITS application which provides an ITS service to a user of ITS.
2. Facilities represents ITSC's communication specifications at OSI layers 5, 6 and 7, e.g. cooperative awareness basic service (for CAM, ETSI EN 302 637-2), decentralized environmental notification basic service (for DENM, ETSI EN 302 637-2) and location dynamic map (LDM, ETSI EN 302 895).
3. Networking & transport represents ITSC's communication specifications at OSI layers 3 and 4, e.g. GeoNetworking, IPv6 over GeoNetworking and IPv6 with mobility extensions. To connect to systems via other protocols (e.g. IPv4) a gateway is needed.
4. Access represents ITSC's communication specifications at OSI layers 1 and 2, e.g. on 5.9 GHz spectrum usage, Decentralized Congestion Control (DCC) and coexistence of ITS and EFC (CEN DSRC) services in the 5.8 GHz and 5.9 GHz bands.
5. Management responsible for managing communications in the ITS station. This entity grants access to the Management Information Base (MIB).
6. Security provides security services to the OSI communication protocol stack, to the security entity and to the management entity. "Security" can also be considered as a specific part of the management entity.

7.2. Stakeholders and Concerns

Most of the stakeholders have some level of interest in communication viewpoint however system architects, data modellers , network administrators , network managers , hardware managers, integrators are the main stakeholders. Concerns of these stakeholders is shown below Figure 34.

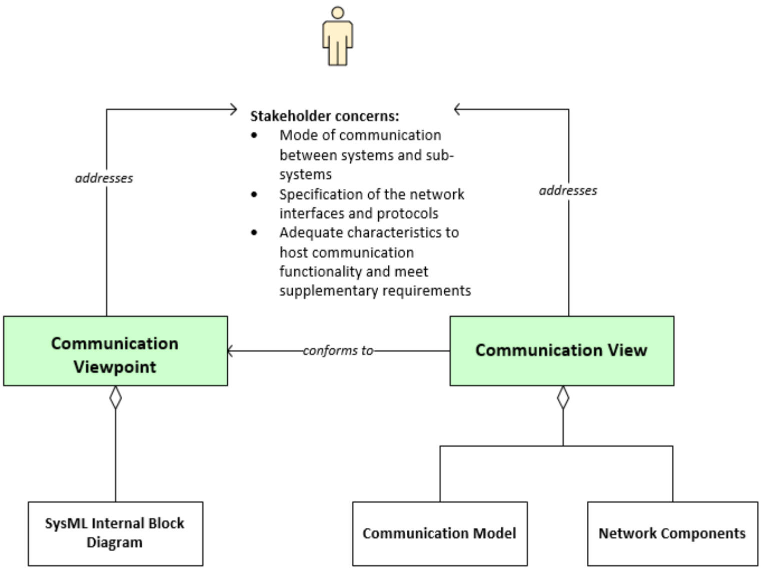


Figure 34: Communication Viewpoint and View

7.3. Communication View

In real-world deployments, the functional systems at one of the four layers (i.e. VRU, vehicle, roadside or central layer), can be deployed in one physical box. However, in case the involved functional systems are deployed in separate physical elements a communication network is needed to interconnect the functional systems. A communication network is needed for the communication between systems at the different layers as well.

The Communication Viewpoint can be mapped to the ISO's Open System Interconnection model as shown in Figure 35 **¡Error! No se encuentra el origen de la referencia.**below.

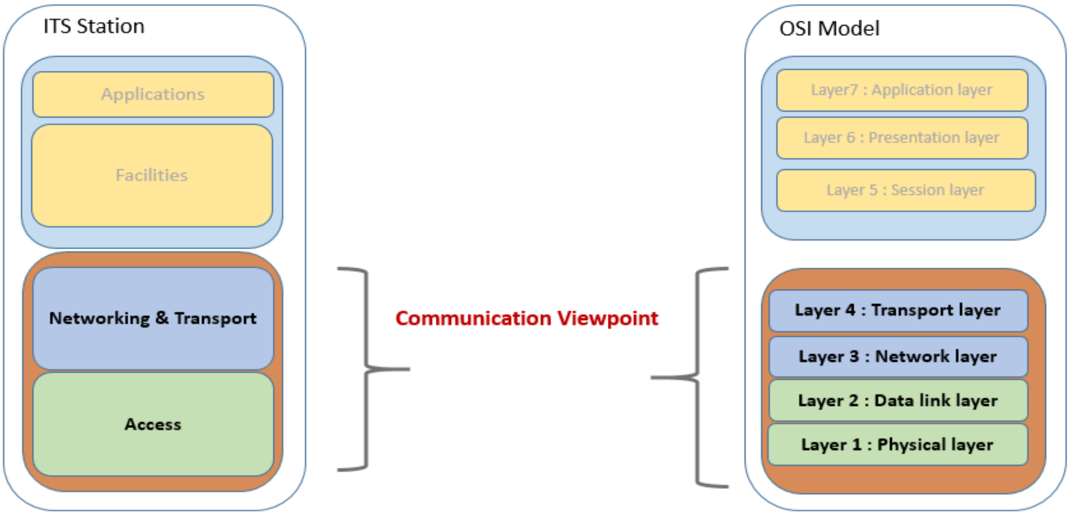


Figure 35: Communication viewpoint mapping of ITS Station and OSI Model

The following networks are identified from communication viewpoint perspective by considering VRUITS architecture [8]:

1. Cooperative ad-hoc networks: best suited for cooperative applications. ITS-G5 with GeoNetworking is used in these networks for VRU2VRU, VRU2V/VRU2I and V2IVRU/I2VRU communication to exchange CAM, DENM and other C-ITS defined messages between cooperative systems.

2. In-vehicle networks for cars and VRU vehicles like PTW three types are defined:

   - Car-specific networks. A serial bus is currently used in the automotive industry to allow microcontrollers and devices to communicate with each other within a car without a host computer. This type of network can be based on CAN, MOST or FlexRay. CAN (Controller Area Network) is one (out of five) message-based protocols within OBD-II and is standardized at the lower layers (physical, data link, transfer layer). Recent network protocols like MOST (Media Oriented Systems Transport) is a high-speed multimedia network technology optimized by the automotive industry which can be used for applications insider or outside the car and FlexRay as successor for CAN. However, FlexRay is not yet widely adopted. EOBD is an EU standard providing diagnostic and reporting capabilities, based on OBD-II (On-Board Diagnostics, release II).

   - VRU-vehicle specific networks. For motorcycles and for other VRU-specific vehicles like mopeds and eBikes such networks with (partly) standardized interfaces are not yet available. Motorcycle vendors also have a proprietary implementation of an OBD-II type of network. Another example of a VRU-vehicle specific network is EnergyBus. It is an open standard for integration of and communication between electric components of light electric vehicles based on DC. The EnergyBus specification is published through the EnergyBus Association, based in Germany.

   - In-vehicle IP networks. This IP network is used to interconnect in-vehicle systems for advanced driver assistance, comfort, and infotainment features. This network is used to connect devices (e.g. smartphone, communication unit) that contain one or more systems like VRU-VIS, VRU CS with HMI. This type of IP-based network can use widespread lower layer technologies like Wi-Fi, Bluetooth as lower layer protocols. Automotive Ethernet is a more recent industry standard for in-car IP networks, developed by the OPEN (one-pair ether-net) alliance special interest group (SIG). Powered VRU-vehicles like eBikes with on-board computers have USB interfaces to connect external systems, (e.g. a smartphone for charging or an external diagnostics system for battery health monitoring) or to an external computer to read in-formation on trips (distance, average speed, etc.). These device-to-device network connections (based on USB or Bluetooth) could be used to connect different systems on powered VRU-vehicles.

3. Public mobile data networks. VRU systems and in-vehicle systems can be connected to central systems via public mobile data networks. These networks are IP-based and use different mobile radio access technologies like GPRS, UMTS, and LTE. ITS systems are mainly the only users on these networks, and mobile operators own these networks and licenses and are responsible for performance and capacity.

4. Other networks:

   - Internet. The central systems in the architecture can be connected via a public IP network.

- Private IP networks: At the roadside layer, the systems are mostly connected via private IP networks, including the connections to central TMC.
- Other short-range wireless networks: systems at the different layers can be connected via other short-range wireless technologies like RF, Bluetooth, BLE or Wi-Fi. IP can be used to exchange information. In some cases, only the wireless MAC identifier and signal strength are used to identify and locate a user. In addition, Bluetooth, BLE, and Wi-Fi can be used to exchange information between components installed in the VRU vehicle and nomadic devices.

7.4. Model kinds and Models

Communication model is considered which shows the mode of communication in terms of network interfaces and protocols. SysML Internal Block Diagram is considered to model the structure of these interfaces and protocols as well as the connection between different networks which will be included during Concrete Architecture D3.2

7.5. Correspondence rules

The context viewpoint has correspondences with information and physical viewpoints as highlighted in Figure 36.

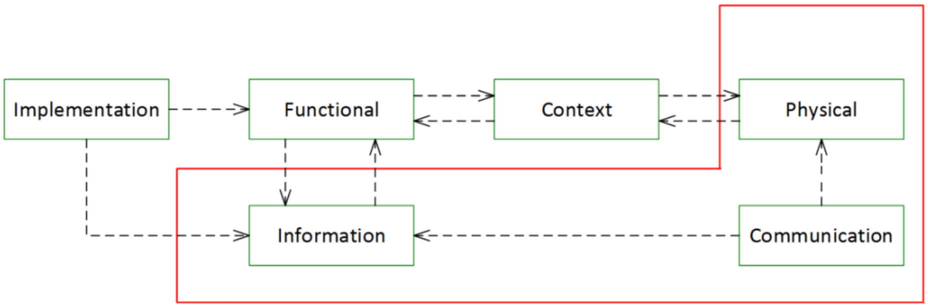


Figure 36: Correspondences for the Communication Viewpoint

Physical and information viewpoints have conformance correspondence to the Communication viewpoint.

8. Information Viewpoint

8.1. Definition

The information viewpoint and view at architectural level help define how the systems will store, manage, manipulate and distribute information [14]. The information view provides high-level view of static data structure and information flow to users, developers, testers, and maintainers. In this section, the main stakeholder concerns, model kinds and models such as data flow (information exchange, protocol message types) are explained.

8.2. Stakeholders and Concerns

Although most of the stakeholders have some level of interest in the information viewpoint, users, architects, data modellers, developers and integrators are the main stakeholders. The information viewpoint addresses the following list of stakeholder concerns.

Information structure and content

It is important to define the structure and content of the information that the C-ITS manages. An architect needs to define it at early stage in alignment with the system's functionality independent of where and how it would be located.

Data flow

The most important data flows need to be considered during the architecture definition. The importance can be decided based on the system's primary responsibilities or its impact on system quality. Data flow can be considered in both functional and information viewpoints.

Data lifecycle

The lifecycle of data is another important concern. It is about the transitions that data items go through in response to external events – transitioning from creation, changes to deletion.

8.3. Information View

The most important data structures and flows need to be considered during the architecture definition. For this purpose, information viewpoint and its respective view are of use. As depicted in Figure 37, we propose the *information structure model* to address the information structure and its content, *data flow model* to address data flow, and data lifecycle model to address data lifecycle concerns.

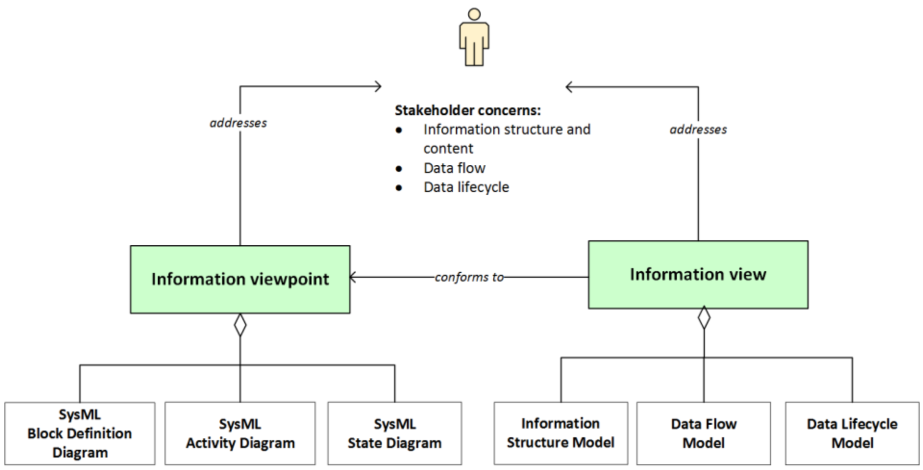


Figure 37: Information viewpoint and view

SysML Block Definition Diagram, activity diagram, and state diagram can be used respectively as a modeling diagram for the information structure, data flow, and data lifecycle models.

Information viewpoint describes how the architecture stores, manages, and distributes data and information. The information view provides high-level view of static data structure and information flow to users, developers, testers, and maintainers. In this section, the data flow (information exchange, protocol message types) is explained.

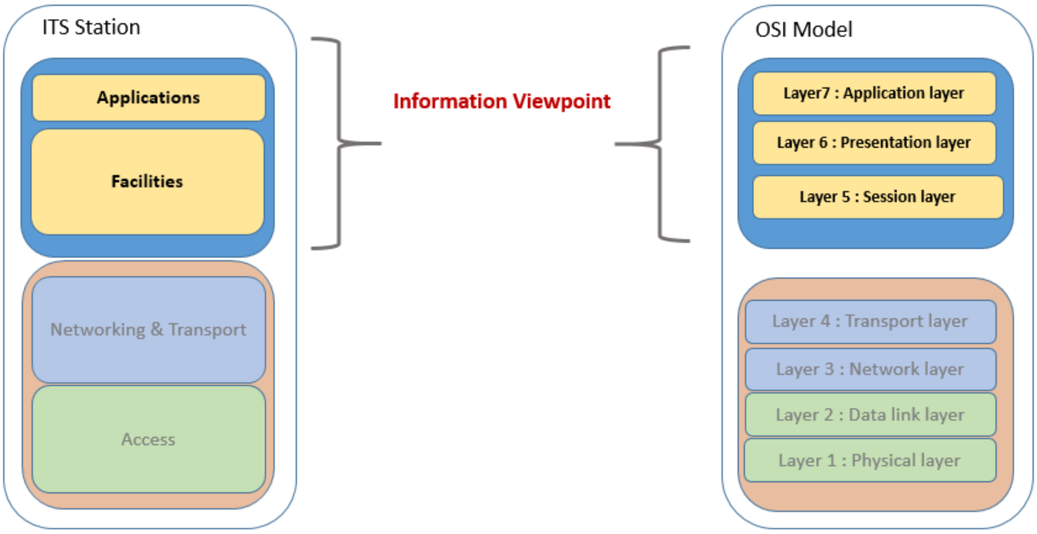


Figure 38: Information viewpoint mapping between ITS station and OSI model.

Following generic exchange message types have been identified from various existing ITS architectures mainly referring to VRUITS architecture [8], which will be utilized communicating between two systems:

- CAM:
  - ETSI has defined the CAM message to inform vehicle status in V2X applications. The CAM message has been specified for ITS-G5 communications, in which vehicles and roadside units make a risk assessment based on the location and speed of vehicles in the safety zone round the vehicle.
  - Vehicles transmit already data, equivalent to the data contained in CAM-messages, on position, speed, and other sensor information to the OEM or Service Provider cloud, which analyses data and provides services to vehicles and customers, e.g. traffic information based on probe vehicle data, and slippery road information.
- DENM: Road Side Units communicate with On-Board Units in vehicles using ITS-G5 standardised communications. DENM (ETSI EN 302 637-3 V1.2.2) is used for this purpose.
- DATEX II: DATEX II (version 2.3) messages can be used for the communication between Service Providers, and between Service Providers and Traffic Data Providers.
- TOPO: Messages are sent from a roadside unit to a vehicle to inform the vehicle about the geometry of an intersection. Message format is inherited from the SAE J2735 DSRC standard.
- SPAT: Message broadcasted by Roadside Units (RSU) to provide current signal status (color) by lane and when the status is expected to change [45]
- LDM: The basic functionality of the LDM is to provide a repository of information for facilities and applications. Facilities such as the CA and DEN basic services can store information into the LDM. Applications can retrieve information from and store information into the LDM [40]

The data flow is explained for the different types of systems and between the systems at VRU, vehicle, roadside, and central system:

- Cooperative ITS systems: Different message types are defined within ETSI TC ITS that can be used for cooperative applications with VRUs. For the specific use cases in this document, the relevant message sets are CAM and DENM. Within CAM the container message sets for PTW, bicycles and pedestrians are not defined in ETSI EN 302 637-2. In addition, other infrastructure-related message sets that are still work-in-progress within ETSI/CEN/ISO are needed. Such messages are for example MAP on Road Topology and Signaling Phase and Timing on signalized intersections (e.g. for INS and IPTS), In-Vehicle Information messages on road signs (IVI, e.g. for GWC) and Cooperative Perception Messages (CPM, e.g. as defined by the Ko-FAS project to broadcast information on detected vulnerable road users to other road users).
- Tag-based systems: tag-based systems can use different technologies for VRU localization relative to their own position, via

wireless communication. The tags used in the system contain IDs to distinguish a VRU or VRU type.

- Smartphone based systems: For smartphone-based applications, information can be exchanged via widely used web-based protocols like SOAP/XML. Special description files are needed per application to describe the information elements for the specific application. Information for VRU-specific applications should preferably be distributed via open data, e.g. on locations for IPTS, GWC trajectories etc.

8.4. Model kinds and Models

As illustrated in Figure 37 the information view consists of Information Structure Model, Data Flow Model, and Data Lifecycle Model, which can be represented using SysML Block Definition Diagram (BDD), Activity Diagram, and State Diagram respectively.

8.4.1. Information Structure Model

Information structure models capture the important data elements and their relationships. Entity-relationship modeling (ER model) is broadly used in data analysis. It composes of entity types and defines their relationships. Besides the ER model, SysML Block Definition Diagram (BDD) can be used for the notation of information structure. The SysML BDD is described in Section 6.4.1. The information structure model should remain high-level and with less detailed information.

8.4.2. Data Flow Model

Data flow models capture the dynamic movement of information or data between system elements and the outside world. It represents the information or data flow between system elements and their directions. Besides the information interface, it needs to capture, flow direction, information scope, volumetric information, the means via which information or data is exchanged (transfer of flat files or real-time exchange of XML messages) [14]. The definition of the data flow model should conform to the interface definitions and function invocations of the functional view.

Data flow can be represented using SysML activity diagrams.

SysML activity diagram captures the overall flow of activities or actions.

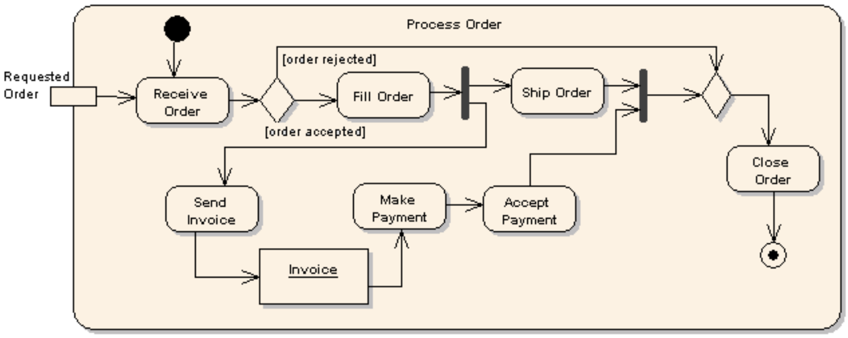


Figure 39: Example activity representation

Figure 39 illustrates an example activity diagram in SysML notation. The main concepts are described below:

- Actions are represented by rounded rectangles e.g. “Receive Order”;
- Decisions represented by diamonds;
- The start (split) or end (join) of concurrent activities are represented by bars;
- The initial node or start of the workflow is represented in a black circle;
- Final node or end is represented by an encircled black circle.
- Arrows running from the start towards the end represent the order in which activities happen.

Besides the SysML activity diagrams, Gane and Sarson or SSADM data flow diagrams can be used to represent information or data flow models.

The data flow within the different systems are described below:

1. **VRU**: at the VRU system the next interfaces are identified

   - VRU-VIS to VRU Vehicle E/E system: For cars and trucks, the CAN-bus (Controller Area Network) interface with EOBD (European On-Board Diagnostics) protocol is used to retrieve status information from the car. For PTW or (electrical) bicycles, no CAN-based interface and protocol has been defined as of today. Interfaces like Blue-tooth and USB are widely available to connect devices, and proprietary EOBD-based protocols can be used to exchange status information retrieved from vehicle sensors e.g. on bicycles or PTW on wheel rotations, steering angle etc. This information is needed to get accurate information on speed, acceleration and direction, together with object state (stop, start, ride, turn).
   - VRU-VIS to VRU-CS: A VRU-VIS can be connected to a VRU-CS (smartphone) to use the HMI of the smartphone. Some of the technologies as described below for vehicles (CE2) can be used to connect a VRU-VIS to an external HMI of a smartphone.

2. **Vehicle**: at the vehicle system the next interfaces are identified:

   - VIS to Vehicle E/E system: The VIS to VEE interface is used in three ways i.e. i) to retrieve information from the VEE on e.g. actual speed, acceleration and exterior lights on/off, ii) to send information on collision risk and request for actuation (e.g. for autonomous emergency braking) and iii) to send information to the in-car HMI. For i) and ii) EOBD with vendor-specific extensions can be used. For iii) the following non-exhaustive list of technologies can be used to connect a VIS to an external HMI:
     - MirrorLink: a technology that bridges the mobile phone and the car. It allows specially written apps running on the phone to be displayed on the car's head unit, where the user can interact with them.
     - iPod Out: Apple supports iPod Out for Apple devices, which allows selected applications to output analogue video to the head unit.
     - HTML5: web technology allows presenting a HMI through the head unit or mobile device, while the application is executed at the VIS.
     - Simple UI Protocol: instead of replicating the complete HMI. The head unit provides simple UI elements, which external applications can use.

3. **Roadside**: at the roadside system the next interfaces are identified:

   - RIS to TLC: The RIS to TLC interface is needed by a RIS to retrieve dynamic information on traffic state or intersection state. Today, no EU standards exist to exchange data from a TLC to external systems like a RIS. In the Netherlands, the IVERA protocol was developed (www.ivera.nl) to ex-change information between TLC and TMC of different vendors. This protocol can be reused as interface between RIS and TLC to exchange information on intersection state. In Germany, a similar protocol was developed by OCIT (www.ocit.org).
   - RIS to Roadside VLS: A roadside VLS system can send information on detected VRUs to a RIS. The interface is system-specific.

4. **Central**: at the central system the next interfaces are identified:

   - CIS to TMC: A TMC with TMIS can be used as central distribution point on all traffic and road state information, i.e. the actual status of flow/speed/travel times and measures, warnings and status of traffic signs. The information could be exchanged by DATEX2 protocol.
   - TMC to IIS: A similar interface as between CIS and TMC can be used between TMC and IIS since similar information of the TMC/TMIS can be sent to the IIS. Also requests to a specific TLC (extended green time) from the IIS can be sent via a central TMC to a specific TLC. Proprietary message formats are available today to send these types of message requests (e.g. IVERA in the Netherlands or OCIT in Germany).
   - TMC to TLC: Dynamic information from RSS or RAS systems at TLCs or other roadside systems can be exchanged to other systems like CIS or IIS a central TMC.
   - TMC to DIS: Proprietary protocols for DIS (Data Interchange Server) can be used to exchange traffic information and road state for interoperability between cross borders. The service allows applications to publish events about logistics entities to which other part subscribe to. The format for message transport is AMQP, although the subscription and publication protocols are at a higher (REST) level. The message payload is any message format that sender/receiver agree.

8.4.3. Data Lifecycle Model

Data or information lifecycle models are used to analyze the way information or data values change over time [14]. SysML state diagram can be used to model the data lifecycle model by capturing the state transition of system element in response to external stimuli.

State diagrams represent the life-cycle behavior of a SysML block in terms of its states and transitions. The transitions that data items undergo in response to external events (from creating data through updating it to deleting the data) can be represented in state diagrams after the functional requirements are elicited.

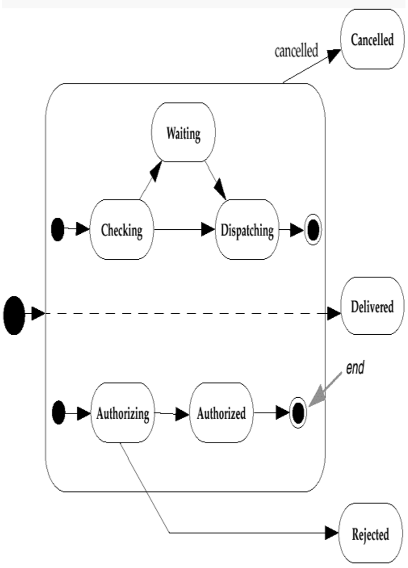


Figure 40: Example state diagram [41]

SysML state diagram has the following main concepts, which are shown in an example diagram Figure 40

- State is captured in a rounded rectangle or circle. It is a significant condition in the life of a SysML block.
- The initial node or start of the state diagram is represented in a black circle;
- Final node or end is represented by an encircled black circle.
- Concurrency is expressed by an orthogonal state.

8.5. Correspondence rules

The information viewpoint has correspondences with functional viewpoint as highlighted in Figure 41 Information viewpoint conforms to the Functional viewpoint and vice versa.

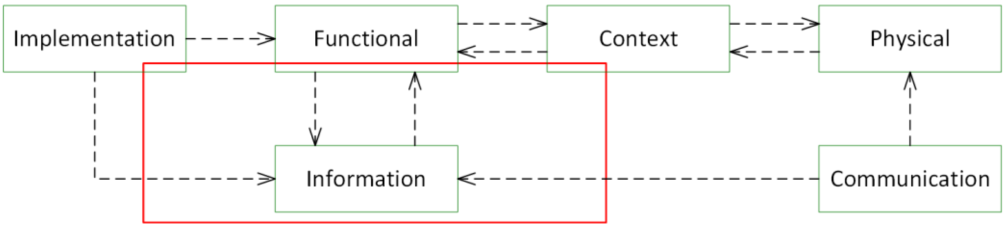


Figure 41: Correspondences for the information viewpoint

The implementation and communication viewpoints conform to the information viewpoint. The definition of the information models is iterative thus requires close interactions with the functional viewpoint.

9. Physical Viewpoint

9.1. Definition

The physical view depicts the system from a system engineer's point of view. It is concerned with the topology of sub-systems at each respective domain of interest, as well as the physical connections between these sub-systems to support the C-ITS applications implemented in the different C-MobILE pilot sites. Sub-systems include functional components that define more specifically the functionality and interfaces that are required to support a particular connected vehicle application.

9.2. Stakeholders and Concerns

Although most of the stakeholders have some level of interest in the Physical Viewpoint, users, architects, system maintainers, OEM (Original Equipment Manufacturer) are the main stakeholders. The physical viewpoint addresses the stakeholders concerns which is mentioned in below Figure 42.

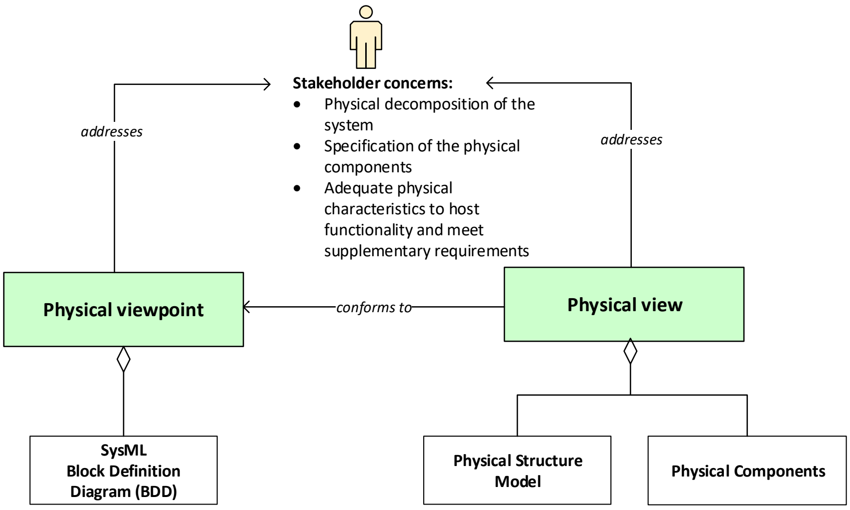


Figure 42: Physical viewpoint and view

SysML Block Definition Diagram can be used as a modelling diagram to depict physical components involved in C-MoBILE.

9.3. Model kinds and Models

As illustrated in Figure 42, the Physical View consists of Physical Structure Model which can be represented using SysML Block Definition Diagram (BDD).

9.3.1. Physical Structure Model

The Physical structure is based on best common practice in previous ITS projects such as [48] i.e. a split in five main physical layers for Vehicle, Roadside, Central (or Back Office), Traveller/VRU System and Support System. These layers are termed as system such as Central System for as similar set of sub-systems are grouped together performing similar set of functionalities for C-ITS.

The physical structure is divided in five 'Systems as shown in Figure 43

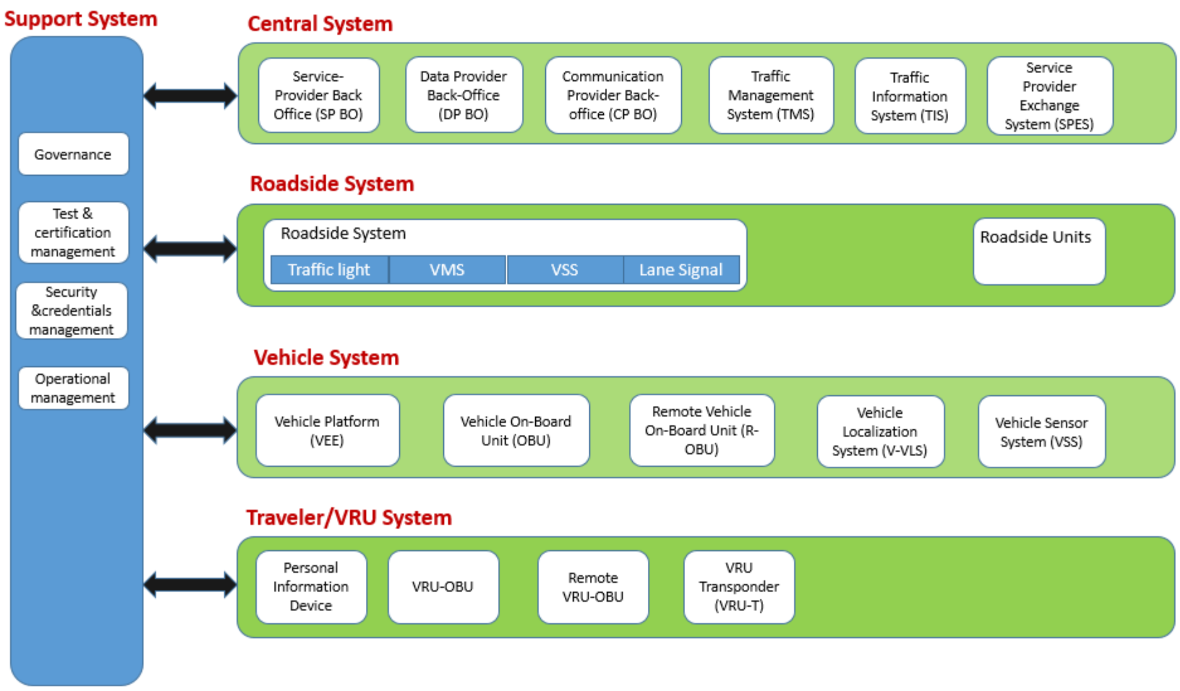


Figure 43: High-level physical structure

Besides the High-level physical model, SysML Block Definition Diagram (BDD) can be used for the notation of physical structure where each main system and its sub-systems are depicted in terms of Blocks.

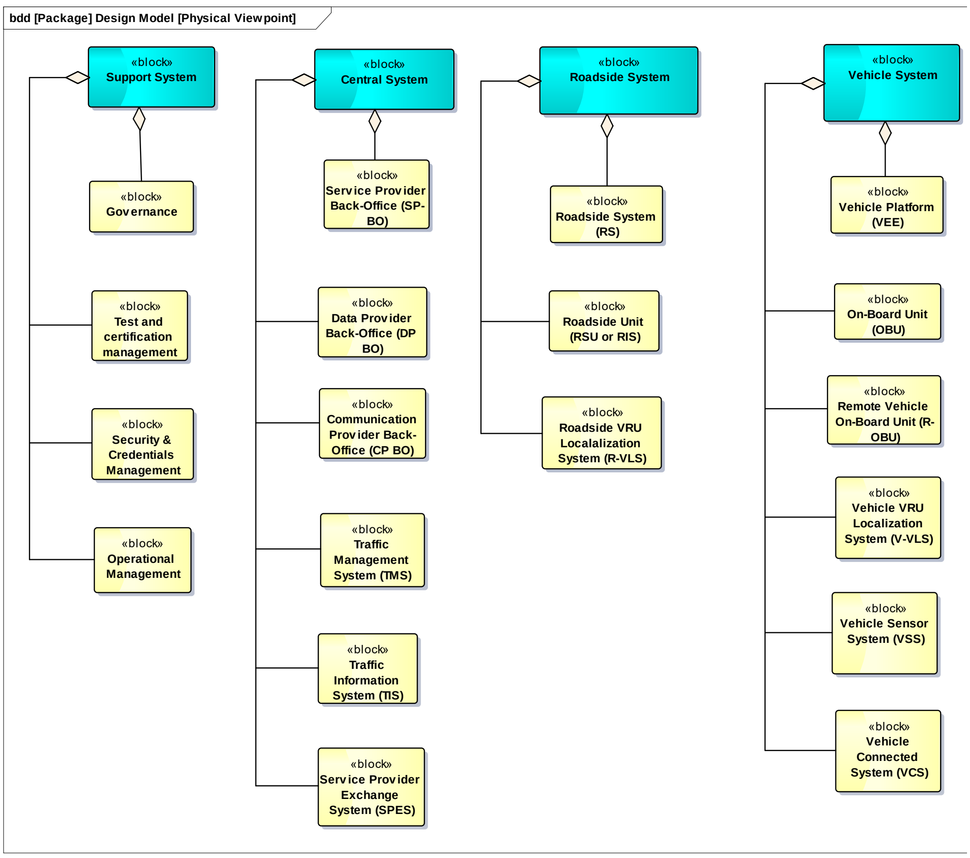


Figure 44: Block Definition Diagram for Physical Viewpoint

The description of each Sub-systems are described below

9.3.1.1. Central System

Sub-systems to support connected vehicles, field, and mobile devices and to perform management and administration functions. The sub-systems in this system are typically virtual systems that can be aggregated together or geographical or functions distributed

Following are the systems mainly involved at Central system:

- Traffic Management System (TMS): A TMS is the functional back-office system of the responsible road operator to enforce legal actions on urban or high-way road sections or intersections based on real-time traffic data from loops, cameras, speed sensors, etc. or actions by traffic controllers. The real-time data that flows from the Traffic Info System is integrated and processed by the TMS (e.g. for incident detection), and may result in traffic measures (e.g. traffic routing, dynamic speed limits) with the goal of improving safety and traffic flow. Cloud servers can be deployed across borders to enable interoperability such as Ericsson Interchange server of Nordic Way Architecture and AEON server of Co-GISTICS
- Traffic Information System (TIS)4: A Traffic Information System is the functional back-office system of a road operator to collect and process real time traffic data from traffic data systems (e.g. roadside sensor systems (loops, cameras) or connected vehicles) and to distribute real-time and/or aggregated information on traffic state (speed, flow and travel times) or road state to TMS or external systems like a SP BO. In practice several distributed TIS from different road operators can be interconnected to a central TIS (e.g. from NDW), which provides aggregated information for the Netherlands.
- Service Provider Back-Office (SP BO): A generic back-office system of a Service Provider used for the specific services of the SP to connected drivers or end-users to inform end users or other SP BO systems from providers. A SP BO can be used to support personal information services for, e.g. navigation or traffic information applications on OBU/PID. A SP BO can also be used to gather floating car data from OBU/PID;
- Data Provider Back-Office (DP BO): A Data Provider BO system is a data system that collects and fuses floating car data and real-time traffic data from roadside sensor systems to increase insight in actual and expected traffic state (e.g. on traffic jams). The DS also distributes enriched (aggregated) information on traffic state (speed, flow and travel times) to Service Providers.
- Communication Provider Back-Office (CP BO) or Central ITS System (CIS): A generic back-office system of a communication provider used for access at several communication systems from other BO systems (like SP BO, TMS,TIS etc.) to send and receive ITS information to/from vehicles or other road users.
- Service Provider Exchange System (SPES): an e-Market (‘broker’) system for discovery and exchange of ITS (end-user) services and ITS communication services; the SPES can support functions like service discovery, service authentication, authorization, accounting (AAA) and billing.
- Data Interchange Server (DIS): Cloud based server specifically to solve interoperability concerns by interconnecting data from cross-border providing data, information, materiel, and services to, and accept the same from, other systems…and to use the data, information, material and services so exchanged to enable them to operate effectively together.

Other back-office systems can also be located at this layer depending on the type of application. One example is a Fleet and Freight Management System which provides the capability for commercial drivers and fleet-freight managers to receive real-time routing information and access databases containing vehicle and/or freight equipment locations as well as carrier, vehicle, freight equipment and driver information. Fleet and Freight Management Center also provides the capability for fleet managers to monitor the safety and security of their commercial vehicle drivers and fleet.

9.3.1.2. Roadside System

Covers the ITS infrastructure on or along the physical road infrastructure, e.g. surveillance or control devices (signal/lane control, ramp meters, or systems to supply information to connected vehicles.

In the roadside (or field) area, the following sub-systems are defined:

- Roadside System (RS): Different types of existing roadside systems are identified:
  - Roadside Substation (RSS): a system deployed along highways and includes sensors (loops), control logic, and actuators. The system can run as a stand-alone closed loop system i.e. run standalone local traffic control functions (e.g. traffic jam tail detection and warning via Variable Message Signs) or can be controlled by the TMS.
  - Traffic Light Controller (TLC): a TLC is a specific type of roadside system. It includes the input from loop detectors or other sensors, a control logic, and the actuation of the traffic lights. A TLC can be run as a stand-alone closed-loop traffic control system. A TLC can also be controlled by a central TMs, e.g. in green wave applications between different TLC’s. A TLC is deployed on urban road or can be deployed at highway access roads for access control.

- Roadside Unit (RSU) or Roadside ITS System (RIS): A RSU/RIS is a cooperative roadside communication system responsible for the two-way communication functionality at a part of a road network (typically an intersection or a road section of 500m – 1km). This physical object is responsible for implementing communication functionality in the roadside system and optionally also application functions. A RSU/RIS is included in the ITS reference architecture standardized by ETSI ITS. A RSU/RIS can be part of the roadside communication network with distributed radio units, and centralized functions in the Communication Provider Back-Office.

9.3.1.3. Vehicle System

In the vehicle area the following sub-systems are defined:

- Vehicle Platform or Vehicle E/E system (VEE): The Vehicle Electrical and Electronic system (E/E) system includes all in-car sensors (speed, lights, etc.) and actuators (brake, etc.). The Vehicle Electrical and Electronic system provides sensor information (e.g. speed) from a vehicle to an external C-ITS system and optionally enables the control/actuation (e.g. speed control) of that vehicle by an external system. The Vehicle E/E must include safety measures to ensure the safe operation of the vehicle,independent of the interaction between the Vehicle E/E and external subsystems.A further differentiation can be made per vehicle type, e.g. emergency vehicle, commercial vehicle or (public) transport/transit vehicle.
- Vehicle On-board Unit (OBU) or Vehicle ITS Station (VIS): An on-board unit is a sub-system attached to a car and needed for driver assisted applications to inform / advise a driver via a HMI. The OBU provides the vehicle-based processing, storage, and communications functions necessary to support connected vehicle operations. The radio(s) supporting V2V and V2I communications are a key component of the Vehicle OBU. Four different types of implementations are represented by the Vehicle OBU:
  - Vehicle Awareness Device – This is an aftermarket electronic device, installed in a vehicle without connection to vehicle systems and is only capable of sending the basic safety message over short-range communications. Vehicle awareness devices do not generate warnings.
  - Aftermarket Device – This is an aftermarket electronic device, installed in a vehicle, and capable of sending and receiving messages over a wireless communications link. The self-contained device includes GPS, runs connected vehicle applications, and includes an integrated driver interface that issues audible or visual warnings, alerts, and guidance to the driver of the vehicle;
  - Retrofit Device – This is an electronic device installed in vehicles by an authorized Service Provider, at a service facility after the vehicle has completed the manufacturing process (retrofit). This type of device provides two-way communications and is connected to a vehicle data bus to integrate the device with other on-board systems. Depending on implementation, the device may include an integrated driver interface and GPS or integrate with modules on the vehicle bus that provides these services.
  - Integrated System – This is a system of one or more electronic devices integrated into vehicles during vehicle production. The Integrated System is connected to proprietary data busses to share information with other on-board systems. The Integrated System may include many control modules.

  In retrofit and integrated implementations, the Vehicle OBU interfaces to other on-board systems through a vehicle bus (e.g., CAN), represented as the Vehicle Platform, this interface provides access to on-board sensors, monitoring and control systems, and information systems that support connected vehicle applications. The vehicle bus may also be the source for GPS location and time, and the access point for the vehicle's driver-vehicle interface. Self-contained devices include an integrated GPS and driver interface that supports direct visual, audible, or haptic interaction with the driver. The Vehicle OBU includes the functions and interfaces that support connected vehicle applications for passenger cars and trucks. Many of these applications (e.g., V2V Safety applications) apply to all vehicle types including personal automobiles, commercial vehicles, emergency vehicles, transit vehicles, and maintenance vehicles. The Vehicle OBU is used to model the common interfaces and functions that apply to all of these vehicle types, i.e. also commercial, public transport or emergency vehicles.
- Remote Vehicle OBU (R-OBU): Remote Vehicle OBUs represents other vehicles that are communicating with the host vehicle. The host vehicle onboard unit, represented by the Vehicle OBU physical object, sends information to, and receives information from the Remote Vehicle OBUs to model all vehicle V2V communications.

9.3.1.4. Traveler/VRU System

At the traveller / VRU system the following sub-systems are defined:

- Personal Information Device (PID): A personal information device is typically a smart phone or personal navigation device used by an end-user. The PID provides the capability for travellers to receive formatted traveller information wherever they are. Capabilities include traveller information, trip planning, and route guidance. It provides travellers with the capability to receive route planning from the infrastructure at home, at work, or on-route using personal devices that may be linked with connected vehicle on-board equipment. A PID might include the communication functionality of a Personal ITS station, as specified in ETSI ITS specifications;

- VRU Vehicle OBU (VRU-OBU): an on-board unit is a sub-system attached to a VRU vehicle (e.g. moped, electric bike) and needed for VRU assisted applications to inform / advise a driver via a HMI.
- Remote VRU OBU (R-VRU-OBU): Remote VRU Vehicle OBUs represent other VRU vehicles that are communicating with the host VRU vehicle. The host VRU vehicle on-board unit, represented by the VRU-OBU physical object, sends information to, and receives information from the Remote Vehicle OBUs to model all VRU related V2V communications.
- VRU Transponder (VRU-T): A V RU transponder is part of a tag-based communication system. A transponder can be active (=with own battery, sending data at constant time intervals), semi-passive (with own battery, sending message at request of an interrogator) or passive tag/chip (without own battery, responding to interrogator request). The tags communicate with an external interrogator, called VRU Localization System, which can be integrated in a vehicle (car, bus, truck) or in a roadside system:
  - Vehicle VRU Localization System (V-VLS): A VRU Localization System is part of a tag-based communication system.
  - Roadside VRU Localization System (R-VLS): A VRU Localization System is part of a tag-based communication system. The VRU transponder carried by a VRU, is an active (=with own battery) or passive tag/chip that can respond on an interrogation signal (trigger) from the VRU Localization System. A VRU Localization System can be integrated in, e.g., a Traffic Light to detect the presence of a specific user, e.g. a person with a disability.

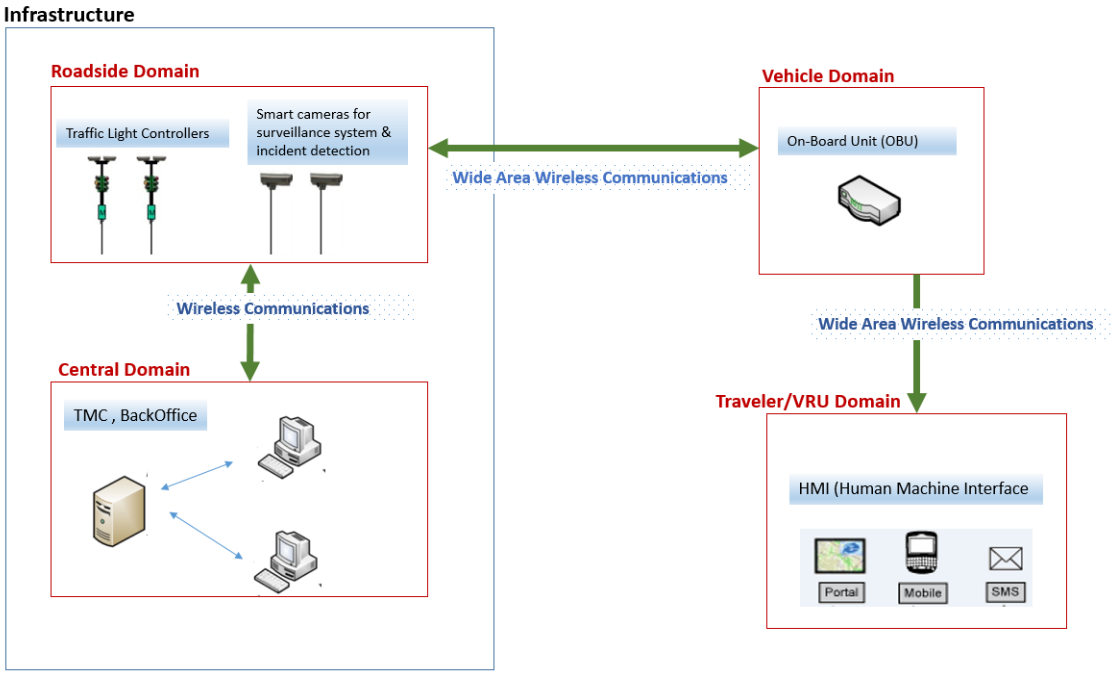


Figure 45: Connections between physical components of C-MobILE

9.4. Correspondence rules

The Physical Viewpoint has direct correspondence with Context Viewpoint.Communication Viewpoint conforms to Physical Viewpoint in terms of communication interfaces and protocols between physical components

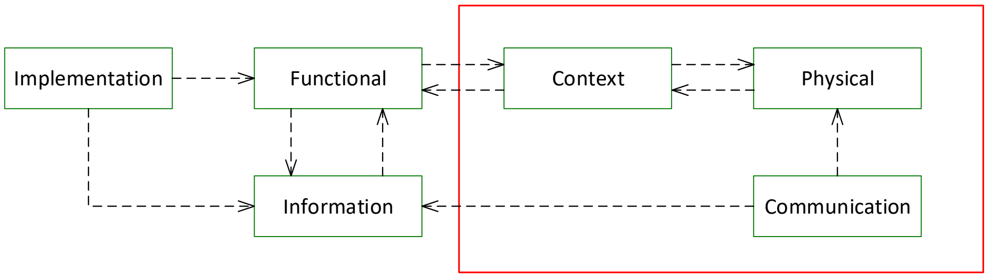


Figure 46: Correspondences for the Physical Viewpoint

10. Implementation Viewpoint

The implementation viewpoint captures the design and implementation details of the C-ITS functional architecture by realizing the functionality into software and hardware components. The C-MobILE reference architecture focuses more on an abstract level independent of design and implementation details.

Main stakeholders for the implementation viewpoint are end user, system/software architect, designer, and tester. Their key concerns are implementation of functional components into software and hardware components, optimal resource utilization, allocation, performance estimation, security etc. The implementation views may consist of application software view, function allocation view, and execution platform view.

For describing the application software view, SysML BDD and IBD can be used. Allocation table of software components to hardware components can be used for the function allocation view. For execution platform view, SysML BDD and activity diagrams can be used.

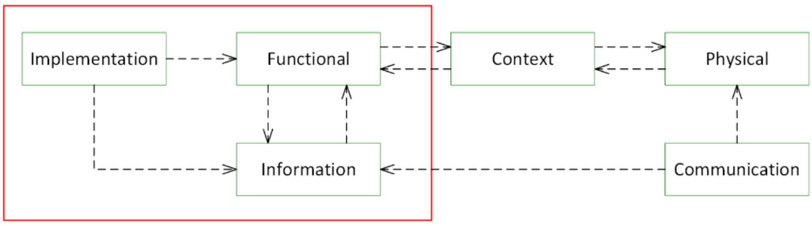


Figure 47. Correspondences for the Implementation viewpoint

As highlighted in Figure 47, the implementation viewpoint has *realization* correspondence to the functional and information viewpoints. It means *n-to-m mappings* by realization relationships between entities in the implementation viewpoint and entities in the functional and information viewpoints.

The implementation viewpoint is out of the scope of this report, however, having the initial definition of the implementation viewpoint and its respective views will ensure the consistency between the architectures and the software/hardware development of the C-ITS services for the pilot sites. Therefore, the implementation viewpoint and its respective views will be elaborated further in the WP3 T3.2 and T3.3 deliverables.

11. Summary

The main objective of WP3 is to provide an architecture for the C-MobILE C-ITS environment [GA]. The purpose of this deliverable is to create a C-ITS reference architecture that enables pan-European interoperability of C-ITS architectures based on the generalization of existing C-ITS architectures.

The diagram below depicts in brief the relation of each deliverable involved within WP3.

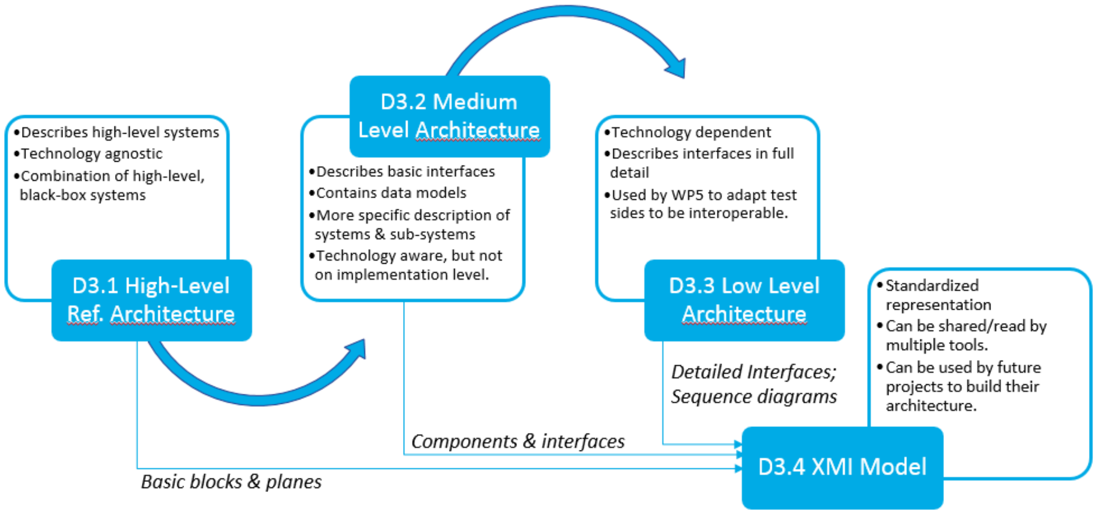


Figure 48: Dependencies between WP3 deliverables

We have analysed existing C-ITS architectures, especially CONVERGE, MOBiNET, and Dutch C-ITS Reference Architecture (in Chapter 3), to define common concepts and vocabulary for the C-MobILE reference, concrete, and implementation architectures. The different architectures from the pilot sites are used as an input for defining a single homogeneous reference architecture in line with current standards which will be further refined in other tasks of WP3, namely T3.2 and T3.3 as illustrated **¡Error! No se encuentra el origen de la referencia.**.

In Chapter 2, Architecture Methodology is defined for different abstraction levels e.g. Reference Architecture, Concrete Architecture, and Implementation Architecture. The C-ITS architecture framework was developed in Chapter 4 by defining Architecture Viewpoints and their respective views addressing respective stakeholder concerns. This will be used for the definition of concrete, implementation, and pilot site architecture, which will facilitate the communication between different stakeholders. SysML is selected as a formal modelling language for describing the architectures.

The high-level reference architecture models are developed in Enterprise Architect using SysML profile. SysML block definition diagram is mainly used to create architectural models for different views and will be provided as input to the concrete architecture and will be decomposed further in details in the form of SysML block definition and internal block diagrams. In the concrete and implementation architectures, interfaces based on existing (open) standards where possible will be specified. However, due to the heterogeneous nature of such interfaces this will not be possible for all interfaces of the architecture. For example, there exists several competing standards for roadside infrastructure to communicate with traffic management centres. C-MobILE does have neither the resource nor the intention to change/redefine all those standards. Instead, we reused existing standards for the definition of communication and information views. In the next tasks of the architecture definition, the architecture tactics to ensure quality attributes will be further detailed.

References

[1] Compass4D. Compass4D; Piloting Cooperative Services for Deployment; Final Results. [Accessed 09/10/2017]. http://www.compass4d.eu/download/compass4d_final_results_final.pdf.

[2] ETSI EN 302 665 - V1.1.1: Intelligent Transport Systems (ITS); Communications Architecture

[3] G. Alcaraz, S. Tsegay, M. Larsson, G. Vernet, E. Koenders, J. Vreeswijk, F. Ophelders, P. Mathias, F. Couly, J. Fernandez, A. Perpey, M. Feringa, and M. Annoni. 30/01/2015. Compass4D; D2.2 Overall reference architecture. [Accessed 20/10/2017]; Version 2.0. http://www.compass4d.eu/download/compass4d_d2_2_overallreferencearchitecture_v20_ec.pdf.

[4] CVRIA; Connected Vehicle Reference Implementation Architecture. [Accessed 09/10/2017]. http://local.iteris.com/cvria/html/about/about.html.

[5] European Commission. SCOOP@F Part 2; 2014-EU-TA-0669-S. [Updated October 2015, Accessed 20/10/2017]. https://ec.europa.eu/inea/sites/inea/files/fiche_2014-eu-ta-0669-s_final.pdf.

[6] E. Broquaire, MC. Esposito, and JM. Tissier. 20/4/2017. Network architecture of the scoop@F project for road operators. [Updated 23/3/2017, Accessed 20/10/2017]. Version 2.00.

[7] European Commission; CORDIS; Projects and Results; IMPROVING THE SAFETY AND MOBILITY OF VULNERABLE ROAD USERS THROUGH ITS APPLICATIONS. VRUITS. [Updated 19/06/2014, Accessed 09/10/2017]. http://cordis.europa.eu/project/rcn/186986_en.html.

[8] K. Moerman, J. Scholliers, M. Penttinen, J. Luoma, M. Kallio, M. Kulju, H. Sintonen, M. Jutila, M. Valta, M. de Goede, M. van Sambeek, E. Wilschut, L. Kroon, J. González L., T. Paadín, F. Morte, J. Carranza, A. B. G. Meléndez, Ó. M. Pérez, S. P. Crespo, C. Cacciabue, and M. Cassani. 23/10/2014. VRUITS; Deliverable D4.2 Architecture for integration of VRUs and draft recommended practices for usability & user acceptance. [Accessed 20/10/2017]. Version 1.2. http://www.vruits.eu/themes/bamboo/deliverables/VRUITS_D42_141029_wm.pdf.

[9] Ministry of Infrastructure and the Environment of the Netherlands, Federal Ministry of Transport and Digital Infrastructure, Austrian Ministry for Transport, Innovation and Technology. Cooperative ITS Corridor Joint deployment. [Updated 2015, Accessed 09/10/2017]. http://c-its-korridor.de/data/download/Flyer%20C-ITS-en.pdf.

[10] Cooperative ITS Corridor Joint Deployment. [Accessed 22/10/2017]. https://itscorridor.mett.nl/English/Project+details/Cooperative+services/default.aspx.

[11] European Corridor – Austrian Testbed for Cooperative Systems. 29/07/2016. SWP 3.7 – Living Lab Definition; Living Lab; WP 3 – Functional Specification & Development. [Accessed 23/10/2017, Updated 29/07/2016]. Version 03.60.

[12] ISO/IEC/IEEE 42010:2011. Systems and software engineering – Architecture description. http://www.iso-architecture.org/

[13] Emery, D., and Rich H. Every architecture description needs a framework: Expressing architecture frameworks using ISO/IEC 42010. Software Architecture, 2009 & European Conference on Software Architecture. WICSA/ECSA 2009. Joint Working IEEE/IFIP Conference on. IEEE, 2009.

[14] Rozanski, N. and Woods, E. Software systems architecture: working with stakeholders using viewpoints and perspectives. Addison-Wesley, 2011.

[15] Kruchten, P.B. "The 4+ 1 view model of architecture." IEEE software 12.6 (1995): 42-50

[16] Haren, Van. TOGAF Version 9.1. Van Haren Publishing, 2011.

[17] Zachman, J. "The zachman framework for enterprise architecture." Zachman International 79 (2002).

[18] International Organization for Standardization. ISO-IEC 25010: 2011 Systems and Software Engineering-Systems and Software Quality Requirements and Evaluation (SQuaRE)-System and Software Quality Models. ISO, 2011.

[19] European Commission. NordicWay; 2014-EU-TA-0060-S. [Updated 09/2015, Accessed 09/10/2017]. https://ec.europa.eu/inea/sites/inea/files/fiche_2014-eu-ta-0060-s_final.pdf.

[20] CVRIA; Connected Vehicle Reference Implementation Architecture. [Accessed 09/10/2017]. http://local.iteris.com/cvria/html/about/about.html.

[21] C-ROADS; HARMONISATION OF C-ITS RELATED DEPLOYMENTS THROUGHOUT EUROPE. [Accessed 09/10/2017]. https://www.c-roads.eu/platform/about/about.html.

[22] European Commission; CORDIS; Projects and Results; IMPROVING THE SAFETY AND MOBILITY OF VULNERABLE ROAD USERS THROUGH ITS APPLICATIONS. VRUITS. [Updated 19/06/2014, Accessed 09/10/2017]. http://cordis.europa.eu/project/rcn/186986_en.html.

[23] C-ROADS. [Accessed 09/10/2017]. https://www.c-roads.eu/platform.html.

[24] European Commission. InterCor; 2015-EU-TM-0159-S; North Sea–Mediterranean Corridor. [Updated 11/2016, Accessed 09/10/2017]. https://ec.europa.eu/inea/sites/inea/files/fiche_2015-eu-tm-0159-s_final.pdf.

[25] ECo-AT; The Austrian contribution to the Cooperative ITS Corridor. [Accessed 09/10/2017]. http://www.eco-at.info/home-en.html.

[26] European Commission. 2017. CODECS; COoperative ITS DEployment Coordination Support. [Updated 31/07/2017, Accessed 09/10/2017]. http://cordis.europa.eu/project/rcn/194851_en.html.

[27] European Commission. AUTOCITS; 2015-EU-TM-0243-S; Atlantic Corridor. [Updated 09/2016, Accessed 09/10/2017]. https://ec.europa.eu/inea/sites/inea/files/fiche_2015-eu-tm-0243-s_final.pdf.

[28] European Commission. C-ITS for Trucks (CITRUS); 2015-BE-TM-0391-S; Rhine-Alpine, North Sea–Mediterranean Corridor. [Updated 11/2016, Accessed 09/10/2017]. https://ec.europa.eu/inea/sites/inea/files/fiche_2015-be-tm-0391-s_final.pdf.

[29] European Commission. PROSPECT; PROactive Safety for PEdestrians and CyclisTs. [Updated 17/08/2017, Accessed 09/10/2017]. http://cordis.europa.eu/project/rcn/193275_en.html.

[30] European Commission; Horizon 2020; XCYCLE. XCYCLE; Advanced measures to reduce cyclists' fatalities and increase comfort in the interaction with motorised vehicles. [Accessed 09/10/2017]. https://ec.europa.eu/inea/en/horizon-2020/projects/h2020-transport/safety/xcycle.

[31] European Commission. SOLRED C-ITS Monitoring Network (SolC-ITS); 2015-ES-TM-0079-S; Mediterranean & Atlantic Corridors. [Updated 11/2016, Accessed 09/10/2017]. https://ec.europa.eu/inea/sites/inea/files/fiche_2015-es-tm-0079-s_final.pdf.

[32] Architecture and use cases of the mobility service platform MOBiNET, Tobias Schlauch, Janki Dodiya, Andreas Rickling, Lars Mikkelsen, Dirk Beckmann, 12th ITS European Congress, Strasbourg, France, 19-22 June 2017

[33] MOBiNET: the Europe-wide e-market place of mobility services for businesses and end users. [Accessed 28/11/2017]. http://mobinet.eu/

[34] COmmunication Network VEhicle Road Global Extension. [Accessed 28/11/2017]. https://converge-online.de

[35] CITRUS: Coöperatieve ITS for Trucks. [Accessed 28/11/2017]. https://www.citrus-project.eu/

[36] integrierte Kooperationsplattform für automatisierte Elektrofahrzeuge. [Accessed 28/112017]. https://ikopa.de/

[37] EUR-Lex Acess to European Union law. [Accessed 28/11/2017] http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32016R0679

[38] Partnership Talking Traffic. [Accessed 28/11/2017]. https://www.partnershiptalkingtraffic.com

[39] ASFINAG, Federal Highway Research Institute, – Federal Ministry for Transport – Innovation and Technology, Rijkswaterstaat Ministry of Infrastructure and Environment, Hessen Mobil Road and

Traffic Management, Federal Ministry of Transport and Digital Infrastructure, Rijkswaterstaat Ministry of Infrastructure and Environment. DEPLOYMENT OF COOPERATIVE SYSTEMS ON THE C-ITS CORRIDOR IN EUROPE. [Updated 2015, Accessed 21/11/2017]. http://c-its-korridor.de/data/download/Revue%20RT-Decembre_2015_p.50.pdf.

[40] DEN/ITS-0010005. ETSI EN 302 895 - V1.0.0: Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Local Dynamic Map (LDM).

[41] Concurrent State Diagrams. [Accessed 28/11/2017]. http://etutorials.org/Programming/UML/Chapter+8.+State+Diagrams/Concurrent+State+Diagrams/

[42] C-ROADS – The Platform of Harmonised C-ITS Deployment in Europe. [Accessed 28/11/2017]. https://www.c-roads.eu

[43] Htw saar, bmt, SIT, SWARCO, ULD, FOKUS, DCAITI, NXP, ULD. iKoPA Deliverable D1v1 Requirements Analysis and System Architecture. 28/02/2017. Version number 1.0. [Accessed 28/11/2017]. https://ikopa.de/wp-content/plugins/download-attachments/includes/download.php?id=99

[44] BASt, BMW, Bosch, Ericsson, Frauenhofer AISEC, Frauenhofer FOKUS, Hessen Mobil, HTW, Opel, Vodafone, Tactilo, Volkswagen. Converge Derliverable D4.3 "Architecture of the Car2X Systems Network". 31/01/2015. Version number 1.2. [Accessed 28/11/2017]. http://www.converge-online.de/doc/download/Del%2043%20Masterdocument.zip

[45] Perry, F. HNTB. Overview of DSRC Messages and Performance Requirements [Updated 03/2017, Accessed 23/11/2017]. http://www.transportation.institute.ufl.edu/wp-content/uploads/2017/04/HNTB-SAE-Standards.pdf

[46] Towards an Architecture for Cooperative-Intelligent Transport System (C-ITS) Applications in the Netherlands (Borgert van der Kluit, et al., April 2015)

[47] System Modelling Language http://sysmlforum.com/sysml-resources/publications/

[48] van Sambeek, M. Turetken, O., Ophelders, F., Eshuis, R., Bijlsma, T. Traganos, K., Grefen. P (2015) Towards an Architecture for Cooperative ITS Applications in the Netherlands, Version 1.0. DITCM. April 17, 2015. Available at: https://goo.gl/pHSD9D

[49] XCYCLE. [Accessed 30/11/2017]. http://www.xcycle-h2020.eu

[50] Kenis, E. Autonomousdriving & Co-operative ITS in Flanders, Department of Mobility & Transport Policy Unit, ITS Belgium Congress, 2016

[51] Vandenberghe, W. Partnership Talking Traffic, Dutch Ministry of Infrastructure & Environment, ITS Belgium Congress, 2017

[52] Talking traffic project [Accessed 30/11/2017]. https://www.partnershiptalkingtraffic.com/

[53] ERTICO. [Accessed 30/11/2017]. http://ertico.com/vision-and-mission/

[54] International Road Transport Union. [Accessed 30/11/2017]. https://www.iru.org/who-we-are/about-iru/history

[55] Federation Internationale de l'automobile. [Accessed 30/11/2017]. https://www.fia.com/organisation

[56] Dajsuren, Y. "On the design of an architecture framework and quality evaluation for automotive software systems." PhD thesis. Eindhoven University of Technology, 2015.