The handbook for Vehicle-to-X cooperative systems simulation

CAR 2 CAR Communication Consortium

Partners of the C2C-CC

Document information

Number:	C2C 607 001	**Version:**	1.1.0	**Date:**	2011-05-19
Title:	The handbook for Vehicle-to-X cooperative systems simulation			**Document Type:**	Specification
Release Status:	released				
Status:	Draft – Functional Completed				

Author:

Company /Institute	Author	Chapter
C2C-CC WG SIM		

Approval:

Function	Name, Company	Date	Signature
Chair of WG SIM	Klaus Jaschke, Deutsches Zentrum für Luft- und Raumfahrt e.V.	10.05.2011	Klaus Jaschke
Co-Chair of WG SIM	Alexander Geraldy, Robert Bosch GmbH	10.05.2011	Alexander Geraldy

Outstanding Issues

Issue	Author	Chapter

Changes since last version

Title:	The handbook for Vehicle-to-X cooperative systems simulation
Explanatory notes:	

Issue	Rev.	Date	Changes	Edited by	Approved
1	1.0	10.05.2011	First release of handbook	WG SIM	WG SIM
1	1.1	19.05.2011	Minor changes and restructure of TeXFiles	K. Jaschke	WG SIM

Content

List of figures

List of tables

1 Introduction

This chapter will give a twofold motivation. On the one hand it will motivate why and when simulations should be used and on the other hand why the handbook was developed.

1.1 Motivation for Simulation

Simulation can be seen as a tool for the verification and validation of cooperative systems. It will flank the development process from the concept phase to market introduction.

During the concept phase, simulation can support in proving and even improving the concept. Simulations are useful to evaluate the hypothesis made during the concept considering different constraints, borders and parameter sets. E.g. various use cases could be simulated to assess the impact for cooperative systems without building any prototype or even real hardware. This will decrease the costs and complexity in the early phase of the development process. Basic requirements could be derived and adjusted with simulations during a very early stage.

Going one step further to the prototype development, simulations are used as a tool for the qualification of prototypes. It is possible to perform impact assessments or performance analyses with the prototype of the cooperative system. The simulation represents the environment and stimulates the system to be able to perform the tests. These simulations and tests could also be performed in later stages of the development process, e.g. in serial production.

Another goal for simulations is the derivation of test cases and test scenarios. These test scenarios could either be used in a simulation environment or in the field to perform tests. Here simulations could help to pre-qualify scenarios for tests and even obtain parameters (e.g. timing phases, penetration rates, etc.) for optimal test setups.

FOTs and simulations go hand in hand and can effectively improve each other.

1.2 Motivation for the Handbook

Within the CAR 2 CAR Communication Consortium there are various topics that are dealing with simulation. For most partners similar questions arise while planning or preparing simulations, like how to verify requirements in a C2X scenario, how to test and evaluate applications, user acceptance, energy efficiency, safety, etc. or how to assess the impact of various penetration rates. But even if the research or study questions are well formulated other questions have to be answered like:

- Which simulations to use?
- Which models (driver, traffic flow, communication, etc.) to include in the simulation?
- Which stimuli, which penetration rate, etc. to use?
- How to evaluate the simulation output
- How to ensure that various results are comparable?

To support a common simulation and test approach the WG Simulation designed this handbook. It tries to cover all kinds of simulations – vehicle dynamics, traffic flow, communication, etc. – and is not restricted to a specific simulation domain. The handbook could be used as a guideline to perform simulations and to ascertain the steps for the preparation, the selection of the test environment, etc. Due to the large variety of studies and tests to be performed the first version of the handbook will not be comprehensive, but should be seen as a living document that will be amended and verified in a later stage.

The overall process for the test approach will be motivated and shown in the next chapter. The following chapters will dig deeper into the details, making recommendations for those involved in simulation as to what should be done when and how during the process. Three use cases – Green Light Optimal Speed Advisory (GLOSA), Intersection Collision Warning (ICW) and Hazardous Location Notification (HLW) – are used as examples for the illustration of the shown process steps and will also be used to approve the handbook approach after the release of Version 1.0.

Additionally, the annex will give an overview of already performed simulation studies within the Car-2-Car Communication Consortium as well as the simulations that are used within the consortium.

1.3 Survey of document

This document contains the ”The handbook for Vehicle-to-X cooperative systems simulation” for the CAR 2 CAR Communication Consortium.

2 The Simulation Approach of the Handbook

In this section, the simulation approach of the handbook is presented. It is more or less a process chain that should be followed for performing simulations and tests. It helps to ensure the validity, the correctness of the test implementation and could be used to optimize the parameterization of applications' variables by using simulation techniques.

The basic approach of how these goals could be achieved is presented in this chapter. A more detailed description of the single steps is given in the following sections.

2.1 Overview

The basic process of the approach is presented in figure 2.1. As can be seen, major activities are depicted as boxes and arrows show the interaction between these activities. The approach covers the whole process starting with descriptions of use cases which should be tested and ends with conclusions derived from test results.

It also allows iteration either with more details, other parameter sets, other implementations (e.g. starting with a model and ending with real HW in the loop), etc. to either gain deeper insight or optimize the implementations.

In the following, a short description of the activities is given:

Use Cases

This activity covers the creation of use case descriptions. The goal is to achieve a common understanding about the use case to be tested and also to present an approach for the standard description of the use cases. The use case can be seen as a starting point for a simulation, therefore a good description (function, what is to be achieved, actors, components involved, environment, penetration rate, etc.) should contain an illustration of the scenario, the constraints and limitations.

Therefore the handbook gives a template for a use case description; the details can be found in section 3.2.

Requirements

In this activity, the detailed requirements of the use case will be derived. So this step implies a refinement of the use case descriptions created in the previous step.

These requirements will be used on the one hand for the realization or implementation of the use case and on the other hand to establish further requirements for the test regarding the test environment and the performance indicators.

Realization

The realization itself is not part of the simulation handbook, as the implementation of the function/system could be seen as a prerequisite for the simulation. It only has to be mentioned that for the simulation, the realization could either be done by another simulation, software representing the real implementation or the real implementation on the final target hardware (hardware-in-the-loop testing).

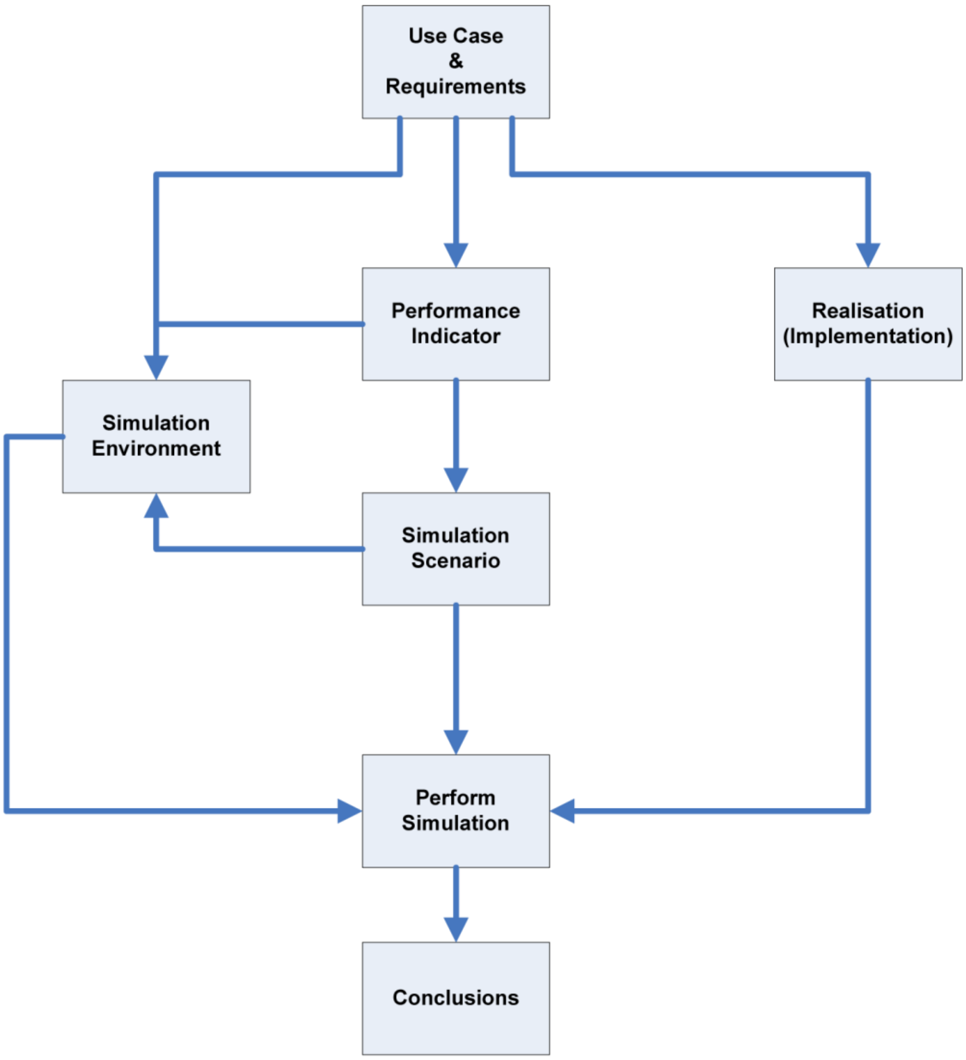

Figure 2.1: Overview of the Simulation Approach

Performance Indicators

The performance indicator will be used to show what to evaluate and how to judge the results by using well defined indicators. The derivation and detailed description of performance indicators will be given in chapter 4.

Simulation Scenarios

A simulation scenario describes the situation in which an application will be investigated. This includes the description of

- states
- properties
- course of action

at a moment in time for each entity (test sequence) as well as the performance indicators for the evaluation of the application.

Within the handbook, the approach is to provide a scenario template as a XML schema with

- basic data type definitions, like info (with name, short description, author, specification link, etc.), position

an hierarchical scenario structure

Details can be found in section 6.1.

Simulation Environment

The simulation environment consists of the test hull in which the *system under test* (SUT) is embedded including all necessary components like simulators or trace recorders for the SUT's output. The simulation environment is used to run the tests. For the choice of the simulation environment for a realistic simulation of V2X communication various aspects have to be considered:

- Vehicular traffic including ego- and other vehicles
- V2X communication including delays, bandwidth constraints, etc.
- how to integrate the application - may include a simulation of the application or the application itself (HW/SW in the loop)
- environment, including road network, weather conditions, etc.

and also the right environment (type of simulation: integrated simulator, fixed or flexible coupling of simulators) has to be selected.

Perform Simulation

Within the process chain (figure 2.1) now all major steps or pre-requisites to run simulations are fulfilled. Therefore the next step is the test run itself. It comprises also the data collection and the pre- and post-processing of these data.

Conclusion

Finally after the performance of the test(s) and the evaluation of data conclusions could be drawn. These conclusions could be used to assess the results or even to perform iterations for the optimization of the results by adopting parameters of the implemented system.

The following section will go further into details for the use case description and will also present a common template.

3 Use Case Description

3.1 Common Use Case Template

Within this section a common use case template is presented, that can be used to package the necessary information covering all major topics needed to describe the use case. Details and the description of the contents of the template will be given in the next section.

Basic Information	
Use Case - ID	
Name of Use Case	
Author/Last Change	
Short Description	
Goal/Target	e.g. safety, efficiency, business

List of Actors and Components	
Actor and Component:	**Description of role:**
e.g. Driver	
e.g. OBU	

Additional Information	
Requirement:	**Description:**
e.g. about System Components	
e.g. about environment	
e.g. about Penetration Rate	

List of Assumptions/Boundaries (e.g. legal aspects)	
Assumption:	**Description:**

List of Stakeholder	
Stakeholder:	**Description**
e.g. Drivers	
e.g. Road Operators	
Objectives	
Scenarios and General Function	
List of Limitations	

3.2 Use Case Overview

This section comprises the description of all of the items mentioned in the template (see section 3.1). It will also advise on how to fill the template.

Basic Information

This part of the use case template contains the basic information and very short descriptions.

Use Case - ID

The unique identifier of the use case is to provide the possibility to reference it unambiguously.

Name of Use Case

The short name of the use case which should indicate the goal of the use case.

Author/Last Change

The names of the authors and the date of the last changes.

Short Description

Short description of the use case in one or two sentences. The description should point out the most important benefit for the user.

Goal/Target

A nested bullet point list of the goals/targets.

List of Actors and Components

This field should give an overview about the involved actors and components which are relevant for the use case.

Actor

Actors are in interaction with the system but are never part of them. They just play a role in the whole system and don't have to be real persons. It's also possible that actors are sensors or other devices.

Component

Components are parts of the system. As this is a use case description, just relevant high level components should be mentioned.

Description of Role

A short description in a few sentences of the role of the actor or component within the whole system.

Additional Information

This part gives the author of the use case description the possibility to state additional information. The information might be the author's experience or results from previous realizations.

List of Assumptions/Boundaries

This part provides the possibility to state assumptions (a hypothesis that is taken for granted [1]) and boundaries (the line or plane indicating the limit or extent of something [1]). Those assumptions and boundaries could also cover legal aspects.

List of Stakeholders

Stakeholders are persons or authorities who are interested or concerned in the results of the system.

Objectives

A description of the goals of the use case that should be attained. The description should consist of more than one or two sentences. Moreover, it should contain important benefits for the user and a brief explanation about how these goals should be reached.

Scenarios and General Function

This part provides the possibility to list typical situations and the corresponding reaction of the system.

Name

The name of the scenario.

Trigger

A short but precise description of a situation which should cause a reaction of the system.

Description

A short but precise description of the reaction of the system upon the trigger condition.

List of Limitations

This part provides the possibility to state scenarios which are not covered by the use case. The aim is to prevent any misunderstanding.

Chapter 6 will give examples of use case descriptions based on the template proposed in section 3.1.

3.3 Overview of Requirements Engineering

Requirements are the basis for the implementation of the application and are also part of the criteria for the validation process. They ensure that all selected use cases could be covered with the derived implementation. Additional requirements ensuring that the system is testable should be covered in these requirements, too. Within the handbook, guidelines are given only on how to derive and document requirements but no recommendation for any tool is provided.

The guidelines highlight the important properties which should be considered during the definition of requirements. These properties are not independent and as a whole validate each other.

Guidelines for requirements management [2–4]:

- **Complete:** A complete requirement or specification must clearly define the required capabilities and functions describing the system.
- **Correct:** For a requirements specification to be correct it must define the real world operational environment of the desired capability, its interface to that environment and its interaction with that environment.
- **Consistent:** A consistent specification is one where there is no conflict between individual requirement statements that define the behavior of essential capabilities and specified behavioral properties and constraints do not have an adverse impact on that behavior.
- **Modifiable:** In order for requirements specifications to be modifiable, related concerns must be grouped together and unrelated concerns must be separated. This characteristic is exhibited by a logical structuring of the requirements document.
- **Testable:** In order for a requirement specification to be testable it must be stated in such a manner that pass/fail or quantitative assessment criteria can be derived from the specification itself and/or referenced information
- **Traceable:** Each requirement must be uniquely identified to achieve traceability. Uniqueness is facilitated by the use of a consistent and logical scheme for assigning identification to each specification statement within the requirements document.
- **Unambiguous:** A statement of a requirement is unambiguous if it can only be interpreted one way.

- **Verifiable:** In order to be verifiable, requirement specifications at one level of abstraction must be consistent with those at another level of abstraction.

- **Ranked:** Ranking specification statements according to importance is established in the requirements document's organization and structure.

Most, if not all, of these attributes are subjective and a conclusive assessment of the quality of a requirements specification requires review and analysis by technical and operational experts in the domain addressed by the requirements.

4 Performance Indicators

4.1 General Performance Indicators (PIs)

4.1.1 Definition

PIs are quantitative or qualitative measurements, agreed on beforehand, expressed as a percent- age, index, rate or other value, which are monitored at regular or irregular intervals and can be compared with one or more criteria. [Glossary of WG SIM]

Some of the PIs could be directly derived from the requirements. But some PIs must be abstracted further from the requirements, particularly if the requirements are formulated unclearly.

When defining and using performance indicators we have to account for the following characteristics of the PIs:

- **Covered parts of the system:**
  - The application: The indicators measure the expected effects of the application, i.e. safety, efficiency, effects on driver.
  - Parts of the system: In some cases they must be satisfied to fulfill some PIs for the verification of the application. E.g., the message delay must be below an upper limit to enable the application. In other cases these PIs are technical constraints which must be fulfilled for the application to be approved, e.g. a maximum network load or minimum data precision.
- **Type of indication:**
  - Verification of the application: The application has requirements to fulfill. From these requirements hypotheses of the expected (positive) results of the application can be derived. The indicators are used to verify these hypotheses.
  - Approval of the application and the system: The application and the system show no negative effects beyond any given limits (e.g. bandwidth, negative effects on efficiency) and fulfill regulatory requirements.
- **Type of Limit:**
  - Absolute limit: The limits are either given as fixed values or as scenario-specific values.
  - Relative limit: The limits are given as a ratio between the performance with/without the application.
- **Statistical Results:**
  - Limit must be kept in average (*A*)
  - Limit must never be exceeded (*S*)
  - Limits must be kept in *x*% of tests (*P*)
  - Limits must be kept for *x*% of the vehicles (*V*)

4.1.2 Indicator domains

The performance indicators can be assigned to different domains based on which effect is to be achieved and proven. One indicator might belong to more than one domain. In the following list you can see an incomplete list of the most important indicator domains:

1. Effects on driving and traffic safety

    - Number of (near) crashs

      Although some measures related to safe driving are directly derived from the simulation such as maximum deceleration or vehicle-vehicle spacing, making a judgment of when a situation is “hazardous” or a “near-miss” is difficult for two reasons. The first is the lack of trust in the absolute value of measured variables in the simulation. Defining a “near-miss”, for example, usually requires both a measure and a threshold and hence requires trust in the absolute value. For this reason also, occurrence of actual “accidents” in the simulation (for example, when two vehicle positions occlude each other) are either prevented by the driver model, or not usually accepted as significant events. Usually, two or more simulations are compared for a directional difference in outcome. This leads us to the second difficulty, which is defining an indicator which we can compare across simulations. Many indicators (including simply averaging the simplest measures like headway or deceleration) have been proposed. They range from simple indicators like average Time-To-Collision[5] (TTC, time until collision if one vehicle is closing in on another) to more complicated ones like deceleration rate needed to avoid a crash (DRAC)[6]. One useful methodology, detailed by Gettman et al., is to detect conflicts that fall below one or multiple thresholds of an indicator[5]. The number of such incidents will be then be counted and/or averaged. The freely available Surrogate Safety Assessment Model software tool (SSAM, Siemens ITS America[7] can evaluate stored trajectories from multiple simulators and yield statistics on the number of detected conflicts according to chosen thresholds and filters (including spatial or link-orientated filters).

    - Speeds, decelerations, brake pressure, steering angle

      These indicators may indicate a danger if high values are measured near to the simulated critical situations. In contrast to the number of (near) crashes, these indicators can often be measured directly in the simulation, but cannot simply be evaluated using static limits.

2. Effects on driving and traffic efficiency

    - Costs, pollutants, time, distance travelled

      The travel time and distance as well as the gasoline consumption can often be measured or computed during the simulations. An extended list of metrics may be found in [8]. Further costs might be computed using an own cost function.

3. Driver related indicators

    - Driver acceptance

The driver acceptance might be measured using driving simulators and surveys.

- Measurable effects on driver, i.e., heart rate, eye movement, electro dermal response
  In addition to the driver acceptance, physiological indicators might be measured during driving simulations.

4. System related indicators

   - System load (cpu, memory, I/O operations)
   - Data Security (level of security)
   - Communication (packet rate, packet delays, etc.)

5. Conformity with statuatory framework

4.1.3 Measurements

In the following (Tab. 4.1) a list of commonly used measurements with their units and typical ranges (from experience within several projects) can be found. If these measurements are used as performance indicators, the precision of the measurement has to be taken into account. The precision varies between sensors and simulators so typical precision values are not listed. If the measurement depends on the test execution or there is no experience with that measurement, the units and ranges are omitted.

Table 4.1: Performance Indikators

Measurement (acronym, unit)	Typical range
Vehicle	
Velocity (v, $\frac{m}{s}$)	[−20, +100]
Yaw Rate(Ψ_z, $\frac{}{s}$)	[−90,+90]
Longitudinal Acceleration (a_{Lon}, $\frac{m}{s^2}$)	[−10,+10]
Lateral Acceleration (a_{Lat}, $\frac{m}{s^2}$)	[−2, +2]
Steering Angle (δ, $^\circ$)	[−45,+45]
Travel time (T_{travel}, s)	[0, ∞[
Fuel consumption (b,l)	[0, ∞[
Amount of start/stop events (h,−)	[0, ∞[
Duration of stops (t_h,s)	[0, ∞[
Collision speed (v_{coll},$\frac{m}{s}$)	[0, ∞[
Traffic	
Number of Accidents (c,−)	[0, ∞[
Traffic density (k, $\frac{1}{km}$)	[0, 150]
Traffic flow (q, $\frac{1}{h}$)	[0, 4000]
Min, avg, max and sum of any measurement	see vehicle data
Queue length at traffic light (L_Q,−)	[0, 500]
Driver	
Time to react (t_r,s)	[0, ∞[
Heart rate (r_h, $\frac{1}{s}$)	[40, 240]
Mental load (D_{ml},%)	
Driver Acceptance (D_{acc},%)	
Adherence of ADAS (D_{adh},%)	
V2X System	
Reception Ratio (r_r,%)	[0, 100]
Network load (N_l,1/s, $bytes/s$ or % bandwidth)	[0, ∞[or [0, 100]
Message update delay (d_c,s)	[0, ∞[
DNM latency (d_d,s)	[0, ∞[
Data security level (N_{sl},$A \ldots F$)	?
System load (S_l,%)	[0, 100]
False negative, false positive, true positive detections ($D_{fn/fp/tp}$,−)	[0, ∞[
Conformity with statuatories	
Speed limit (v_{max},$\frac{m}{s}$)	[0, ∞[
Right of way (RoW,true/false)	

We will see some examples in chapter 7. There x+ denotes a value of a measurement x in scenarios with an active ADAS application and under the required penetration rate. By x− the value in scenarios without the application is denoted.

5 Simulation Frameworks

5.1 Simulation Architectures

A realistic simulation of V2X communication scenarios needs to consider various aspects of simulation, i.e. the vehicular traffic, the V2X communication network, the application, and the environment. Vehicular traffic includes the physical movements of vehicles on an arbitrary road network. The simulation of the V2X communication network handles the wireless message transmission among vehicles, and between a vehicle and the fixed infrastructure. Application simulation means the simulation of applications that are to be integrated into real world vehicles. For this purpose, inner vehicle interfaces have to be emulated to allow the application to interact with other modules of the vehicle, e.g. GPS sensors. The last aspect is the environment simulation which includes the road network itself as well as temporary events, such as weather conditions and roadworks.

In general, existing V2X simulation architectures can be divided into three different areas [9–11]:

- An integrated simulator covering all V2X domains,
- a fixed coupling of different simulators, and
- a flexible simulator coupling realized by
  - a simulation runtime infrastructure,
  - a service-oriented architecture.

Integrated simulator covering traffic, communication, and applications domains

An integrated simulator covers the three domains important for V2X simulations, i.e. vehicular traffic, communication, and application simulations. Only one clock for the time management exists and no further synchronization is necessary. Here, the integration of different simulation domains is a major challenge for the simulator developers. In the history of computer simulations, vehicular traffic and wireless communication have been divergent domains without connections. In general, an expert in developing traffic simulators does not necessarily have detailed knowledge about the simulation of wireless communication and vice versa. Thus, the development of such a tool with a high accuracy in traffic, communication, and application simulation has proven to be difficult. As a result, existing integrated simulators are rather suited for high-level simulations.

Fixed coupling of different simulators for the different domains

Fixed simulator couplings are couplings of independent simulators where each simulator is specified for one of the domains, i.e. vehicular traffic, communication, or application simulation. Each simulator has its own clock for its time management. Thus, additional synchronization mechanisms have to be implemented in the coupling mechanism to ensure that all events of the coupled simulators are processed in the correct order. The coupling component is adapted to the used simulators and integrated simulation tools cannot be exchanged. A disadvantage of the fixed coupling is that a re-implementation is not only necessary if a simulator is to be exchanged, but also the integration of new versions of one of the coupled simulators often requires making adjustments. So, a fixed simulator coupling only works well as long as all simulation scenarios have similar requirements that are fulfilled by the existing coupling.

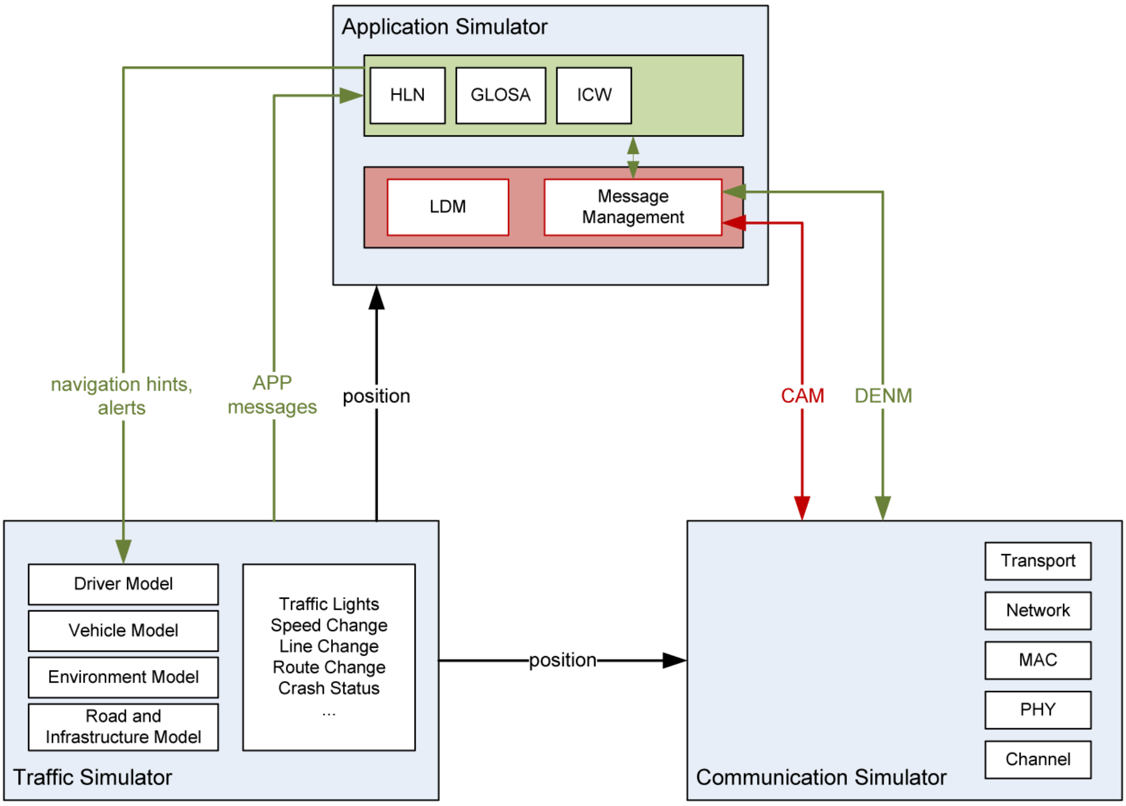

Figure 5.1: Interaction of the different simulators

Simulation runtime infrastructure (RTI) with common interfaces for flexible simulator coupling

If requirements vary depending on the simulated scenarios, it is not satisfying to use simulator couplings that are adapted to specific simulators and cannot be exchanged. To master this challenge, a simulation runtime infrastructure (RTI) with common interfaces allows the integration of arbitrary discrete event-based simulators. The coupling via common interfaces provides the flexibility to exchange simulators according to the specific requirements of a simulation scenario. A central management is provided by the simulation runtime infrastructure and offers services to handle synchronization, communication among the coupled simulators as well as lifecycle management of each component. A solution can be inspired by the IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA). Thus, attaching a simulator only requires the implementation of the interfaces of the simulation runtime infrastructure and to realize the commands specified within. The internals of the underlying implementation are hidden.

Service-oriented Architectures with common interfaces for flexible simulator coupling

Often the performance of V2X related test and evaluation studies requires an exchangeability of used models and applications. It has to be easy to integrate different models and/or applications in different contexts. This means that the software model can be easily integrated and the developer only needs to take care of the in- and outputs needed by the model or applications. An architecture concept providing an appropriate way of re-orchestrating software components is a service oriented architecture. There a service represents a delimited and defined performance, which is produced by an application module and consumed by other application modules. The service interface and the functional specification is strictly defined between using and providing application modules. These services are able to collude which means that services from different contexts could be integrated within a new overall context (orchestration). The loose coupling offers a high level of autonomy to service developer/provider.

5.2 Communication Simulator

The communication simulator should model the properties of the communication system connecting the participating stations (ITS Vehicle Stations and ITS Road Side Stations).

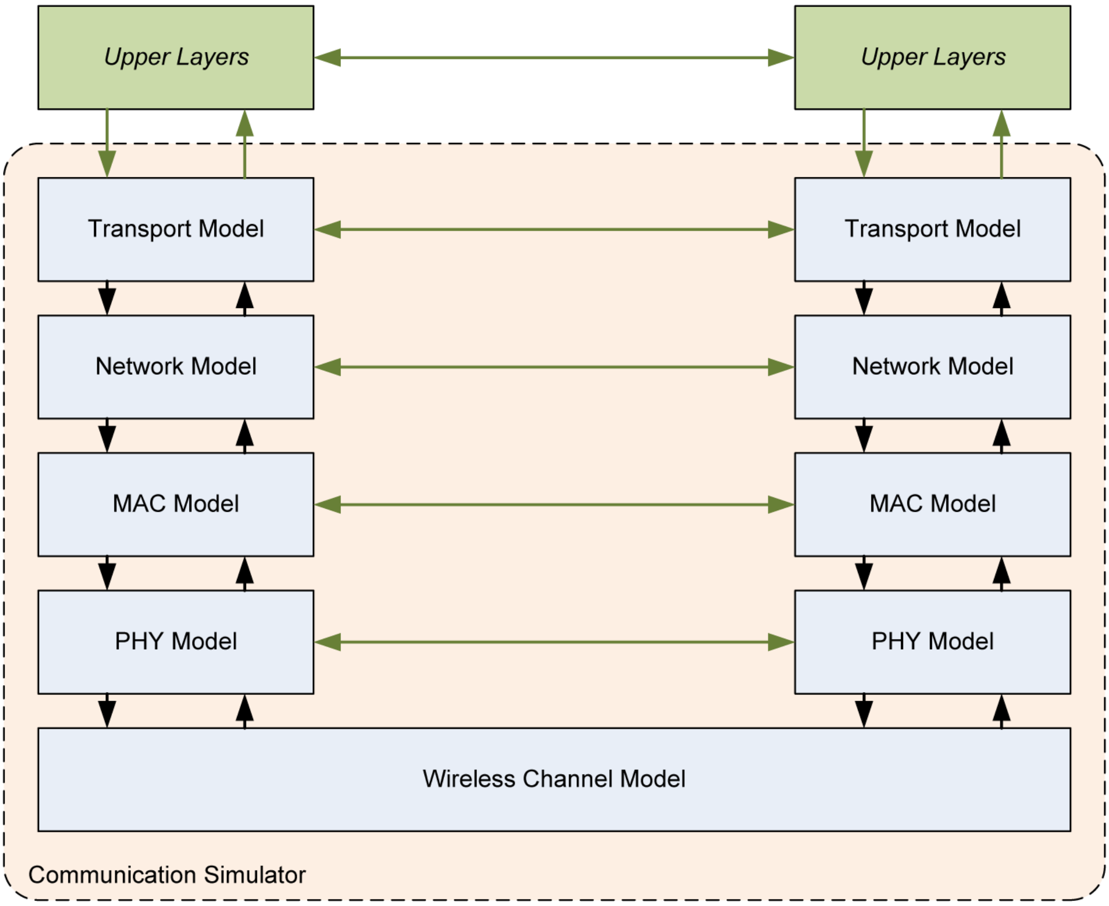

Figure 5.2: Interaction of the different models of the Communication Simulator

5.2.1 Main requirements

- Read description of the scenario relevant for the communication simulator
  - Parameter for wireless channel model, network model and facilities
  - Road and infrastructure topology
  - Stationary and mobile nodes
- Interactions with other simulators
  - Receives GPS coordinates from traffic simulator
  - Receives message to transfer (e.g. CAM or DENM sent by a vehicle) from application simulator
  - Sends message (e.g. transferred CAM or DENM received by a vehicle) to application simulator

5.2.2 Wireless Channel Model

The Wireless Channel Model describes the properties of the physical signal transmission from one station to another. It is often characterized as a time- and location-dependent attenuation of the signal power due to path loss, multi-path propagation, interference and obstacles. Today, the properties of the wireless channel are usually modeled by statistical measures. However, with increasing computational power more realistic and thus more complex models (e.g. raytracing) may also be a valid option.

Level details

- No path loss
- No path loss, but consideration of propagation speed
- Time invariant no obstacle model
  - 2D: Depending only on distance (Friis model)
  - 3D: Depending on the height of the antennas, LOS + reflection on the ground (two-ray ground model)
- Time-variant no obstacle model
  - Large scale fading (e.g. Log-Distance Shadowing)
  - Small scale fading (e.g. Rician, Raleigh, Nakagami, ...)
- Time invariant obstacle model
  - Modeling of obstacles like buildings or trees in LOS
- Time variant obstacle model
  - Modeling of obstacles like buildings or trees in LOS
  - Fading
- Raytracing

5.2.3 PHY Model

The PHY Model describes the properties of the physical network interfaces. On the one hand it may provide functionality like sensing the medium for idle times (used by many MAC layer proto- cols) and different modulation schemes for the transmitter. On the other hand the PHY Model is used to determine under which conditions data symbols or data packets (on the physical layer) are received correctly.

Levels of detail

- Receive all packets
- Receive all packets above RX power threshold (Circle model / Single Threshold Model)
- Receive all packets above RX power threshold and no other transmission above interference threshold (Two Thresholds model)
  - RX Threshold

– Interference Threshold

- Receive all packets above SNR threshold
  – Integration of RX power over preamble / integration over packet

- Symbol level RX model (Receive symbols above SNR)
  – Modulation scheme
  – Forward Error Correction

5.2.4 MAC Model

The MAC Model describes the protocols used by stations within direct communication range to exchange information and each model usually implements one specific protocol of the ISO/OSI medium access layer. For C2C communications the main focus of the MAC Model are strategies to access the shared communication medium and to resolve conflicts due to its concurrent use by more than one station within communication range.

Levels of detail

- No model
- CS (Aloha)
- CSMA/CA, TDMA,…
- 802.11p EU profile or other full featured MAC protocol

5.2.5 Network Layer Model

The Network Layer Model describes the protocols used to exchange information between stations which are not in direct communication range and therefore need a routing protocol to find a communication path to the destination station or stations. For C2C communications position-based routing approaches (e.g. GeoNet project) play an important role.

Levels of detail

- No routing (Single hop communication)
- Static routing (predetermined routes e.g. for connections between RSUs)
- Dynamic routing protocols (e.g. position-based routing)

5.2.6 Transport Layer Model

The Transport Layer Model describes the protocols used by the applications for end-to-end communication. The chosen protocol has a strong influence on the reliability of the communication service but may introduce a significant overhead both in complexity as well as transmitted data.

Levels of detail

- No transport layer protocol
- BTP (Basic Transport Protocol - ETSI TS 102 636-5-1) / UDP
- TCP / RTP / other full featured transport layer protocol

5.3 Traffic Simulator

In traffic research, three main classes of traffic flow models are distinguished according to the level of detail of the simulation. In macroscopic models traffic flow is the basic entity. Microscopic models simulate the movement of every single vehicle on the street, mostly assuming that the behavior of the vehicle depends on both the vehicle's physical abilities to move and the driver's controlling behavior [12]. Mesoscopic simulations are located at the boundary between microscopic and macroscopic simulations. Herein, vehicle movement is mostly simulated using queue approaches and single vehicles are moved between such queues.

In the field of V2X communication every vehicle and infrastructure station should be simulated as an entity, hence most traffic simulators are microscopic simulators. Within the microscopic simulation different levels of detail of the used models match different focus of the simulation. The main models in a traffic simulation are the Driver Model, Vehicle Model, Road and Infrastructure Model and the Environmental Model. Very detailed modeling is sometimes referred as sub-microscopic simulation in the literature.

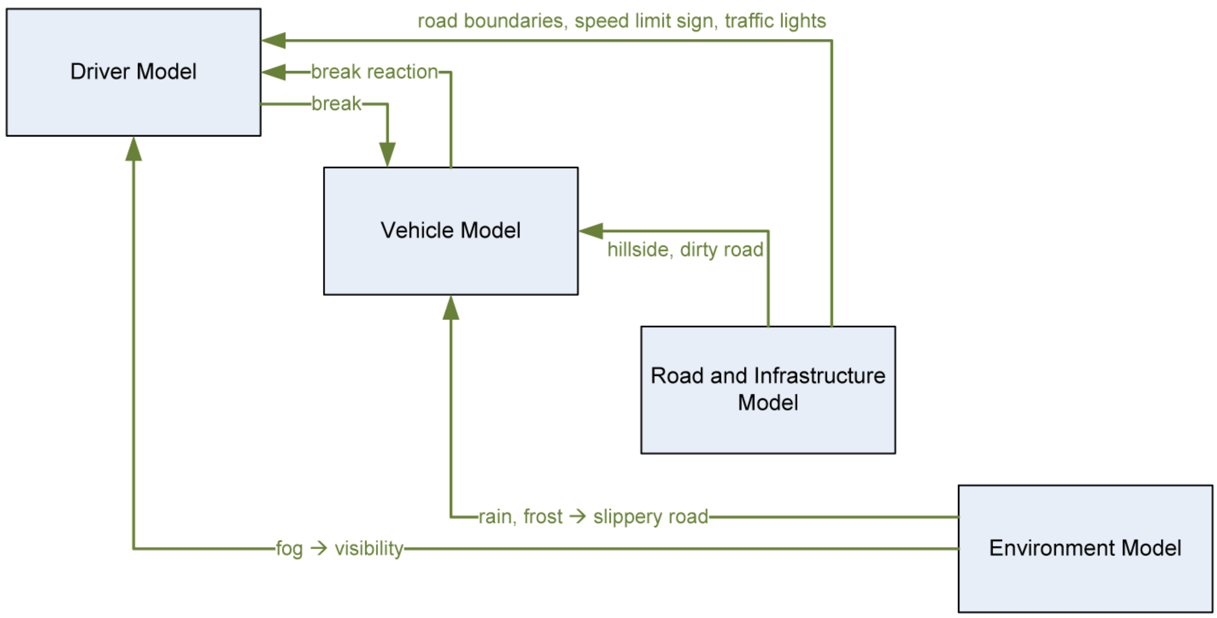

Figure 5.3: Interaction of the different models of the Traffic Simulator

5.3.1 Main requirements

A traffic simulator within the C2CCC domain should meet following main requirements.

- Read description of the scenario relevant for the traffic simulator
- Ability to export topology information
- Interactions with other simulators
- Send location information (e.g. GPS coordinates) to communication simulator and application simulator
  - Send information about traffic infrastructure stations (e.g. traffic light data) to application simulator
  - Receive information about application-triggered speed changes from application simulator
  - Receive information about application-triggered lane changes from application simulator
  - Receive information about application-triggered route changes from application simulator
- Support of different driver models in the same simulation
- Support of different vehicle models in the same simulation

5.3.2 Driver Model

The Driver Model affects the movement of the vehicle depending on age, experience, reaction time, preferences, etc. of the driver. It consists of the movement and the crash model.

Real world drivers may not always react to the provided traffic and route information. Virtual drivers should also be able to follow the received advice or simply ignore it. This behavioral modeling can deliver important results about the effectives of the advice in scenarios where not all drivers are reacting to the provided information.

Movement Model

- No vehicle movement
- Static (predefined) vehicle movement
  - Constant speed, constant direction
  - Constant speed
  - Random waypoint movement
- Dynamic model
  - Interaction with other vehicles
  - Car-following model, e.g. Krauss Driver Model, Wiedemann Driver Model, Gipps, Intelligent Driver Model (IDM)

  - Interaction with other models, e.g. environment: limited viewing distance due to fog or rain results in slower driving speeds
  - Reaction to events (reception of traffic information, ...)

- Movements generated by using a driving simulator

Crash Model

- No collision
- Statistical model (probability of a crash is modeled with certain parameters)
- Interaction based model
- Crashes generated by using a driving simulator

5.3.3 Vehicle Model

The Vehicle Model describes physical (translational and rotational) capabilities of the vehicle. The usage of vehicle modeling makes it possible to prove the benefits of V2X applications for heterogeneous vehicle types.

Levels of detail

- Discrete point model
- Physical dimensions: vehicle type, size
- Physical modeling of the physical states of different degrees of freedom
- Interaction with other models (e.g. environment for slippery roads)

5.3.4 Road and Infrastructure Model

The Road and Infrastructure Model describes the road topology and infrastructure components. It enables the construction of realistic road topology scenarios including tunnels, bridges, multilane highways and complex intersections. Infrastructure components such as traffic lights, variable speed limit signs, etc. are used to control the traffic in real world scenarios. These road components can also be modeled in the traffic simulation and act as the infrastructure traffic control elements.

Levels of detail

- No road model
- Road network (including lanes and geographical information)

- Infrastructure components which provide and collect information about current road traffic situation (traffic lights, speed signs, speed sensors, etc.)

- Interaction with other models

5.3.5 Environmental Model

The Environmental Model describes environmental conditions like obstacles, street and weather conditions (fog, heavy rain, slippery road, etc) that can influence the visibility and the reaction time of the driver and also the modeling of the wireless channel. This model can be used to test the benefits of the V2X application in different environmental conditions without changing the road topology.

Levels of detail

- Ideal environment (no influence on other models)

- Constant environment (fog, dirty roads, etc.) in whole simulation area

- Constant environment in defined simulation areas

- Dynamic environment

5.4 Application Simulator

The application simulator provides the environment where the applications are executed in e.g. on a vehicle, a ITS roadside station, or traffic light. The individual applications can influence the global simulated vehicle traffic.

5.4.1 Main Requirements

- Deploy implemented application logic

- Interactions with other simulators
  - Receives GPS coordinates from traffic simulator
  - Sends message to transfer (e.g. CAM or DENM sent by a vehicle) to communication simulator
  - Receives message (e.g. transferred CAM or DENM received by a vehicle) from communication simulator
  - Receives traffic light data from traffic simulator
  - Sends information about application-triggered speed changes to traffic simulator
  - Sends information about application-triggered lane changes to traffic simulator
  - Sends information about application-triggered route changes to traffic simulator

5.4.2 Functionality Model

The Functionality Model describes how the logic of an application is implemented. Here, different levels of abstraction are possible presenting the application logic. On the lowest level, only a very simple behavior is defined. A fully implemented application deployable in a real vehicle is used on the highest abstraction level. The level of abstraction depends on the simulation focus, e.g. a simple behavior of an application might be enough if communication protocols are to be tested at an early stage. In contrast, a detailed implementation of an application is necessary if the effectiveness of the application is to be optimized.

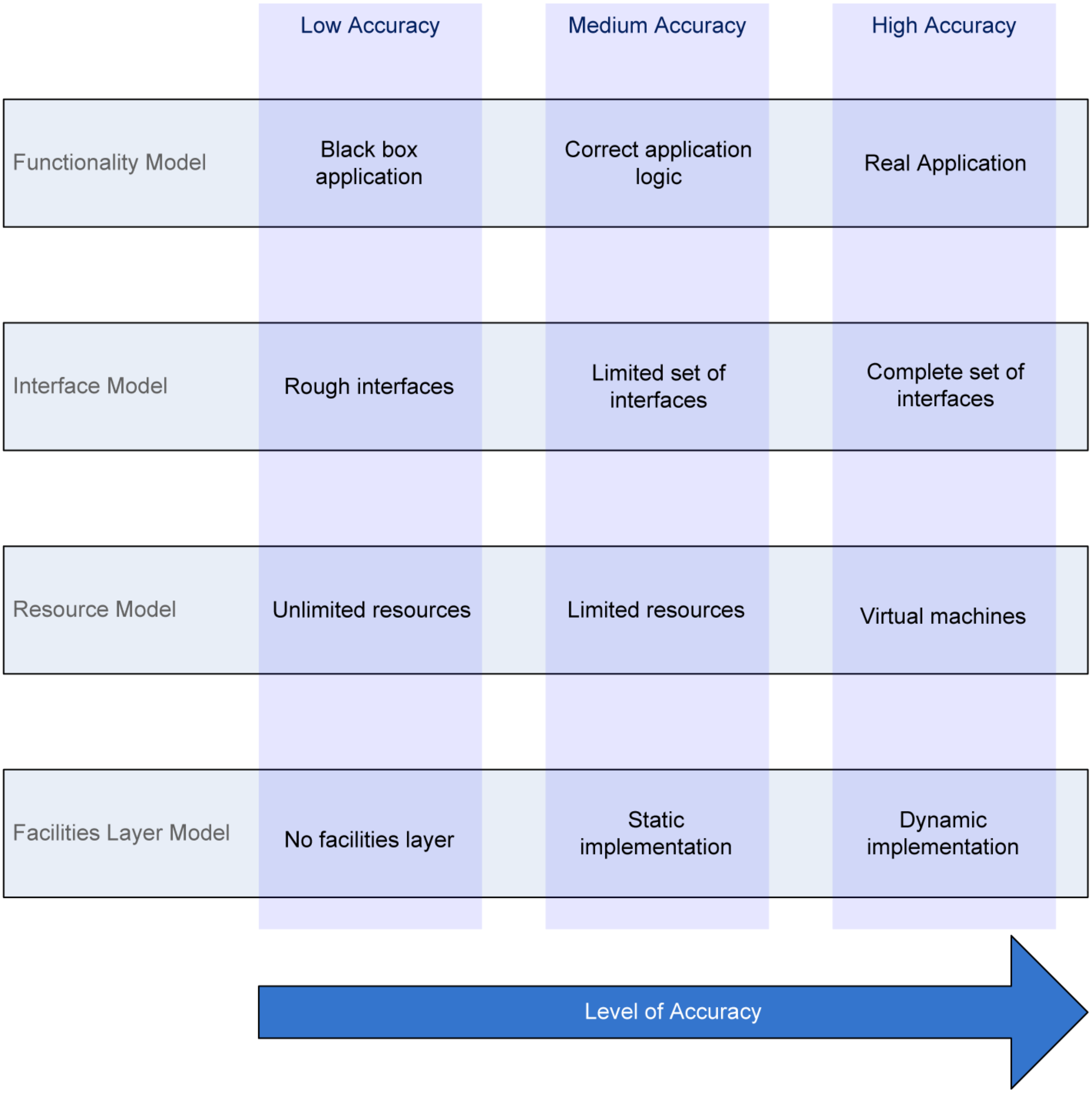

Figure 5.4: Detail levels of the different models of the Application Simulator

Levels of detail

- Black box application: only the rough output of the application in reaction to an input is simulated
- Correct application logic: application logic is nearly identical to the real application but the implementation is not
- Real Application: real implementation of the application without modification

5.4.3 Interface Model

This model describes the interfaces for the interaction between the application and its environment. In real vehicles, applications are embedded in a container (middleware) that hides the vehicle hardware and allows the data exchange with other components, e.g. vehicle sensors. How detailed the implementation of the interfaces has to be depends on the simulation focus and the application itself. In general, the higher the level of accuracy in the Functionality Model, the more detailed the Interface Model has to be.

Levels of Detail

- Rough interfaces: all necessary data between application and other components can be exchanged but the interfaces are not equal to the real interfaces
- Limited set of interfaces: only a subset of the real interfaces accessible by the application is provided
- Complete set of interfaces: all interfaces accessible by the application deployed in the destination environment are provided

5.4.4 Resource Model

Here, the provided hardware resources are defined. In general, applications running in real vehicles have limited resources in contrast to applications deployed on a powerful simulation server. To get more realistic simulation results regarding the application performance, a limitation of hard- ware resources makes sense.

Levels of Detail

- Unlimited resources: applications can use all resources provided by the simulation environment
- Limited resources: resources available for each application are limited
- Virtual machines: distinct virtual machines for each application provide a close approximation of a real deployment

Real machines: distinct real machines for each application provide a test environment com- parable to a real deployment

5.4.5 Facilities Layer Model

The Facilities Layer provides generic support facilities to applications. On the one hand, the facility layer can be divided into basic facilities, which are mandatory for every station, and domain facilities, which are required by specific applications. On the other hand, the facility layer is composed of three functional categories according to the ETSI specifications (see [13, 14]): information support, communication support and application support. While applications may exhibit some common requirements, their assigned use cases may also add some specific requirements.

Application Support Facilities

The application support is the kernel of common functions supporting the applications. It consists of station lifecycle management, automatic services discovery, download and initialization of new services, generic HMI capabilities, and many others.

Information Support Facilities

The information support covers the presentation layer of the OSI reference model and holds the role of data management. The main entity that supports this is the Local Dynamic Map (LDM) that is able to take data from different sources and from received ITS messages to build a data model of the local environment.

Communication Support Facilities

The communication support, which includes the session layer of the OSI Reference model. It will cooperate with the transport and network layer to achieve the various communication modes required by the applications.

Levels of Detail

Since all facilities differ both in complexity and in their interfaces, the level of detail has to be considered for each facility individually. The levels of detail proposed here focus on the implementation detail of a single (abstract) facility. However, some levels may not be suitable for all facilities.

- Facility not implemented: If required by the application, the application simulator or the application itself has to provide the functionality of the facility.

- Static implementation: The interfaces of the implementation may only consist of a subset of the specified interfaces. The behavior of the facility model is defined by static data (e.g. configuration file, database) and does not consider interaction with other components.

- Dynamic implementation: The implementation consists of all necessary interfaces and the behavior fully complies to the specification - including interaction with other components.

6 Description of Simulation Scenarios

6.1 Scenario Template

Scenarios are often described in a natural language and must then be translated into simulation- specific scenario descriptions before a simulation can be run. Simulator-specific scenarios are typically not reusable with a different simulator and can't easily be transformed into the needed scenarios.

For this reason, we want to describe abstract tool-independent scenarios which can then either be read by simulators or transformed into tool-specific scenarios. We have chosen XML (Extensible Markup Language) as a basis for the scenario description since it offers several advantages:

- XML is a well-known, structured language which allows the definition of domain-specific languages (XML schema definition, XSD) and supports hierarchical scenarios.
- Many tools are available to develop the XML schema and to write XML files.
- XML files can be easily parsed, checked and processed automatically. So the scenario descriptions can be read by simulators or transformed into tool-specific scenarios easily.

The scenario template is structured into basic data types which are reused several times within the scenario template and the scenario structure. The scenario structure contains an environment description, describing especially the communication models, weather and topography, and the traffic description with the description of the traffic entities and traffic demands. The scenario template is currently under development and can be found on the SVN server of the C2CCC WG SIM in trunk/WorkPackages/WP06 TestScenarios/schema. We will now describe the current scenario template elements and structure.

6.1.1 Basic Data Types

The basic data types (BasicTypes.xsd) provide basic scenario data types for the scenario definitions and general auxiliary types.

ItemInfoType

The ItemInfoType is an auxiliary data type providing additional documentation about the data element of the scenario description. For this reason, ItemInfotype is part of almost all complex data types of the scenario template hierarchy.

ItemInfoType enables the developers to reuse self-contained ready-to-use scenario elements containing any functional data and the respective documentation. Therefore the ItemInfoType contains the following information (each of type string):

Table 6.1: XML type ItemInfoType

ItemInfoType		
Name	[1..1]	Name of the element.
ShortDescription	[1..1]	Gives a short description of the element.
Author	[1..*]	The Author(s) of this element with contact data.
SpecificationLink	[1..*]	Links to the specification of the element.
DocumentationLink	[1..*]	Links to further documentation of the element.
LegalInformation	[0..*]	Restrictions to the usage of this element.

PositionRefType

The PositionRefType is a general address or coordinate description. It allows for different forms of position specification, e.g. WGS84, postal address, AGORA-C. Others may be added in the future.

It contains the following information (types are defined later on):

Table 6.2: XML type PositionRefType

PositionRefType		
ItemInfo	[0..1]	ItemInfoType. Gives additional information about the Position as defined by ItemInfoType.
CHOICE:		
* WGS84	[1..1]	WGS84Type. The position is given as WGS84 coordinate.
* AGORAC	[1..1]	AGORA C Type. The position is given as AGORA-C reference.
* PostalAddress	[1..1]	PostalAddressType. The position is given as postal address.

WGS84Type

The WGS84Type describes a WGS84 coordinate by latitude, longitude and elevation:

Table 6.3: XML type WGS84Type

WGS84Type		
Latitude	[1..1]	LatitudeType. The Laditude of an position in the WGS84 coordinate system.
Longitude	[1..1]	LongitudeType. The Longitude of an position in the WGS84 coordinate system.
Elevation	[0..1]	ElevationType. The Elevation of an position in the WGS84 coordinate system.

PostalAddressType

The PostalAddressType describes a location as a human readable address containing state, city, postal code, street name and number. Any element of the address may be omitted, but an empty address is meaningless.

The PostalAddressType consists of the following elements (each of type string):

Table 6.4: XML type PostalAddressType

PostalAddressType		
StreetName	[0..1]	Name of the street.
StreetNumber	[0..1]	Number within the given street.
PostalCode	[0..1]	The postal code of the region, city or district.
City	[0..1]	The City must not be ambiguous. If the string does not define a unique City, State and PostalCode must be given.
State	[0..1]	State and/or country.

WayPointType

The WayPointType describes one single position, if needed for a certain time.

Table 6.5: XML type WayPointType

WayPointType		
Time	[0..1]	TimeType. The time for which this information is valid.
Speed	[0..1]	SpeedType. The speed at the given time (if needed).
Position	[1..1]	PositionRefType. Describes the position (at the given time).
Signals	[0..*]	ParameterType. Desribes optionally the signals set at the specified time.

ParameterType

The parameter of the model to use are given as key/value pairs. They have to be mapped to the used model's parameters.

Table 6.6: XML type ModelParameterType

ModelParameterType		
Name	[1..1]	StringType. Parameter name.
Value	[1..1]	StringType. Parameter value.

Simple Types

Table 6.7: XML type Simple

AGORA C Type	hexBinary	coded AGORA-C location reference
DimensionType	double, [0,∞[	Dimension of any object in m.
Directiontype	double, [0,360[	Movement angle or orientation on earth in degrees. 0 degrees is north, 90 is east.
DistanceType	double	Distance in m.
Durationtype	double	Duration in s.
ElevationType	double, [-1000,10000]	Elevation in m.
ElevationDegType	double, [0,360[	Elevation in degrees
IntensityType	double	Intensity of sunlight
LatitudeType	double, [-90,90]	Latitude in degrees.
LongitudeType	double, [-180,180]	Longitude in degrees.
Precipitation-IntensityType	double	intensity of rain/snow
SpeedType	double, [-30,150]	velocity in *m/s*.
StringType	string	General string type
TemperatureType	double	temperatre in degrees Celcius
TimeType	long	Unix time in *s* since epoch.

6.1.2 Scenario Structure

The scenario is the overall description of a situation or an action of a set of entities within an environment at a certain moment or for a given duration. The simulation platforms and field tests which may be used to realize these scenarios possess different structures and different partitioning of this task. Therefore the structure of the scenario template is independent of any simulation tools. So the scenario comprises one instance of the EnvironmentType and one instance of the TrafficType. The Environment consists of all objects and models which are not directly related to mobile traffic entities (e.g. cars, pedestrians). The TrafficType contains models for these mobile entities. Additionaly one instance of the ItemInfoType is required to describe the scenario.

Table 6.8: XML type ScenarioType

ScenarioType		
ItemInfo	[1..1]	ItemInfoType. Description of this scenario model.
Map	[0..1]	OpenDRIVEType. The map for the simulation.
Environment	[1..1]	EnvironmentType. Describes non-traffic elements of the scenario.
Traffic	[1..1]	TrafficType. The traffic description of the scenario.

EnvironmentType

The EnvironmentType contains all information which is not directly related to the traffic participants, especially the description of the topography and the physical models.

Table 6.9: XML type EnvironmentType

EnvironmentType		
ItemInfo	[1..1]	ItemInfoType. Description of the environment.
TopographyElement	[0..*]	TopographyElementType. Description of the topography (including buildings)
Network	[0..*]	NetworkType. Describes communcations models.
Weather	[0..1]	WeatherType. Describes the weather situations.

TopographyElementType (tbd)

NetworkType

The NetworkType contains all basic communication network definitions which are not specific to any traffic participant. This is mainly the propagation model and the communication protocols used by a set of communicating entities.

Table 6.10: XML type NetworkType

NetworkType		
ItemInfo	[1..1]	ItemInfoType. The description of the network model.
PropagationsModel	[0..1]	PropagationModelType. Gives the configuration of the propagations models.
Protocol	[0..*]	ProtocolType. Gives the configuration of the protocol stack.

WeatherType

The WeatherType contains all information about a global static weather condition.

Table 6.11: XML type WeatherType

WeatherType		
ItemInfo	[1..1]	ItemInfoType. Description of the weather model.
Temperature	[0..1]	TemperatureType. The current temperature.
Precipitation	[0..1]	PrecipitationType. The precipitation specification.
Wind	[0..1]	WindType. The specification of the wind.
Sun	[0..1]	SunType. The specification of the sun intensity.

PrecipitationType

The PrecipitationType contains information about rain and snow.

Table 6.12: XML type PrecipitationType

PrecipitationType		
ItemInfo	[1..1]	ItemInfoType. The precipitation description.
CHOICE:		
* Rain	[1..1]	PrecipitationIntensityType. Intensity of rain.
* Snow	[1..1]	PrecipitationIntensityType. Intensity of snowfall.

WindType

The WindType contains information about the wind.

Table 6.13: XML type WindType

WindType		
ItemInfo	[1..1]	ItemInfoType. The wind description.
Speed	[1..1]	SpeedType. The speed of the wind.
Direction	[1..1]	DirectionType. The direction of the wind.

SunType

The SunType specifies the current parameters of the sun.

Table 6.14: XML type SunType

SunType		
ItemInfo	[1..1]	ItemInfoType. Description of the sun model.
Intensity	[1..1]	IntensityType. The sun intensity.
Direction	[1..1]	DirectionType. The direction of the sun (from test area)
Elevation	[1..1]	ElevationDegType. The elevation of the sun.

TrafficType

The TrafficType defines the traffic participants and the models related to them. Since the map is traffic related, it is also included in the TrafficType.

Table 6.15: XML type TrafficType

TrafficType		
ItemInfo	[1..1]	ItemInfoType. Description of traffic
TrafficEntities	[0..*]	TrafficEntityType. Specification of traffic participants.

TrafficEntityType

The TrafficEntityType contains all information related to the traffic participant. As an example, the configuration of the V2X equipment is entity specific and part of that type. The basic technical in- formation (length, width, acceleration, fuel consumption, maximum speed, etc.) should be defined in the entity’s TrafficModel parameters.

Table 6.16: XML type TrafficEntityType

TrafficEntityType		
ID	[1..1]	StringType. A unique ID.
ItemInfo	[1..1]	ItemInfoType. Description of the traffic entity.
TrafficModel	[1..1]	TrafficModelType. The description of the entity’s behaviour (model).
WayPoints	[2..*]	WayPointType. List of Waypoints to travel.

TrafficModelType

As different models have to be supported, the values used for describing the properties of a traffic entity must be formulated in a way which allows to convert them into the used simulator’s parameter. Due to this, it is assumed, that the model type and its parameters should rather describe the behavior, instead of listing parameters of a certain, abstract model. Nonetheless, the usage of explicite physical values is adviced, allowing an easier mapping of the parameters. These parameter may include physical properties of the vehicle and/or the driver. It is proposed to use a prefix to distinguish between them, i.e. ”vehicle.length”, or ”driver.reaction time”.

Table 6.17: XML type TrafficModelType

TrafficModelType		
ItemInfo	[1..1]	ItemInfoType. Description of the traffic model.
Name	[0..1]	StringType. Name of the model to use.
Parameters	[0..*]	ParameterType. The parameters to use.

7 Proof of Concept

7.1 Introduction

This chapter gives an illustration of '"how to apply"' the handbook. It is using the three use cases green light speed advisory, intersection collision warning and hazardous location warning as examples. In a later stage this chapter will be amended and it will give a proof of the concept of the handbook. Therefore the handbook will be applied to the three use cases and the tests and simulations will be performed to verify the shown approach.

At this stage this chapter mainly gives an illustration of the use cases and their performance indicators.

7.2 Use Case: Green Light Optimal Speed Advisory

This section will describe the use case Green Light Optimal Speed Advisory by the use of the common use case template (see section 3.1). The filled template might not be comprehensive, but will give a first overview:

Table 7.1: GLOSA Use Case

Basic Information	
Use Case - ID	UC - 0001
Name of Use Case	Green Light Optimal Speed Advisory (GLOSA)
Author/Last Change	Sebastian Röglinger (HAW Ingolstadt)/14.02.2011
Short Description	The system provides a speed advice which enables the driver to pass the stop line during the green phase of a traffic light controlled intersection or pedestrian crossing without having to stop the vehicle.
Goal/Target	• improve traffic efficiency – the traffic flow should be smoother – the traffic should flow more energy efficiently – the traffic flow should remain fair for all traffic participants • improve driving efficiency (benefits for the drivers of equipped vehicles) – less or attenuated deceleration and acceleration maneuvers – more energy efficient driving – reduced traveling time

List of Actors and Components	
Actor and Component:	**Description of role:**
Driver	The driver could take the advice.
IVS/AU	The IVS/AU receives the signal phase timing data provided by the IRS and is aware of the intersection topology. Based on the information, the IVS determines the advice.
Traffic Light/ Intersection IRS	The IRS provides information about the signal phase timing.

Additional Information	
Requirement:	**Description:**
Environment	• The current traffic situation could influence the speed advisory. • Availability of high-precision localization service. The distance to the stop line should be computable with the required accuracy.
Required penetration Rate	at least: 1 car + 1 intersection

List of Assumptions/Boundaries (e.g. legal aspects)	
Assumption:	**Description**
Yellow and Red/ Yellow signal	The application's advice assists the driver in passing the traffic light while green. So, for other signal states, no advices are given.
Boundaries for speed advisories	The upper speed advisory must be equal or below the speed limitation defined by public authorities. Additionally, there might be a lower speed limit which should also be respected by GLOSA implementations.

List of Stakeholder	
Stakeholder:	**Description:**
Drivers	interested in better flowing road traffic and saving fuel
Citizens	interested in less noise and environmental pollution
Public authorities	interested in less CO_2 emission
OEMs	interested in low costs and high user acceptance

Objectives
The objective of this use case is to assist the driver to pass a traffic light controlled intersection's or crosswalk's stop line by green without stopping the vehicle. This is done by providing an optimal speed advisory if the vehicle approaches an traffic light. The goal is to prevent high deceleration and high acceleration maneuvers. Moreover, the traffic should flow more smoothly in areas with equipped intersections. Additionally, reduced fuel consumption should be achieved at least for the equipped vehicles.

Scenarios and General Function		
Name:	**Trigger:**	**Desprition:**
1. Passing the stop line by green	**a.** Passing the stop line by green without stopping the vehicle is possible if the vehicle speed is adjusted.	The driver receives a speed advisory which has to be between the upper and the lower boundaries of the allowed speed advisories. If the optimal speed is not between these boundaries, no advisory is given.
	b. Passing the stop line by green without a speed adjustment is possible.	The driver receives the advisory to hold the current vehicle speed.
2. Intersection is not equipped with a V2X RSU	N/A	No advisory is given.

List of Limitations	
Limitations:	**Description:**
Scenario: Arriving at the stop line by red	The driver receives no advice from the GLOSA system if the signal phase forecasting determines that passing the stop line by green is not possible.

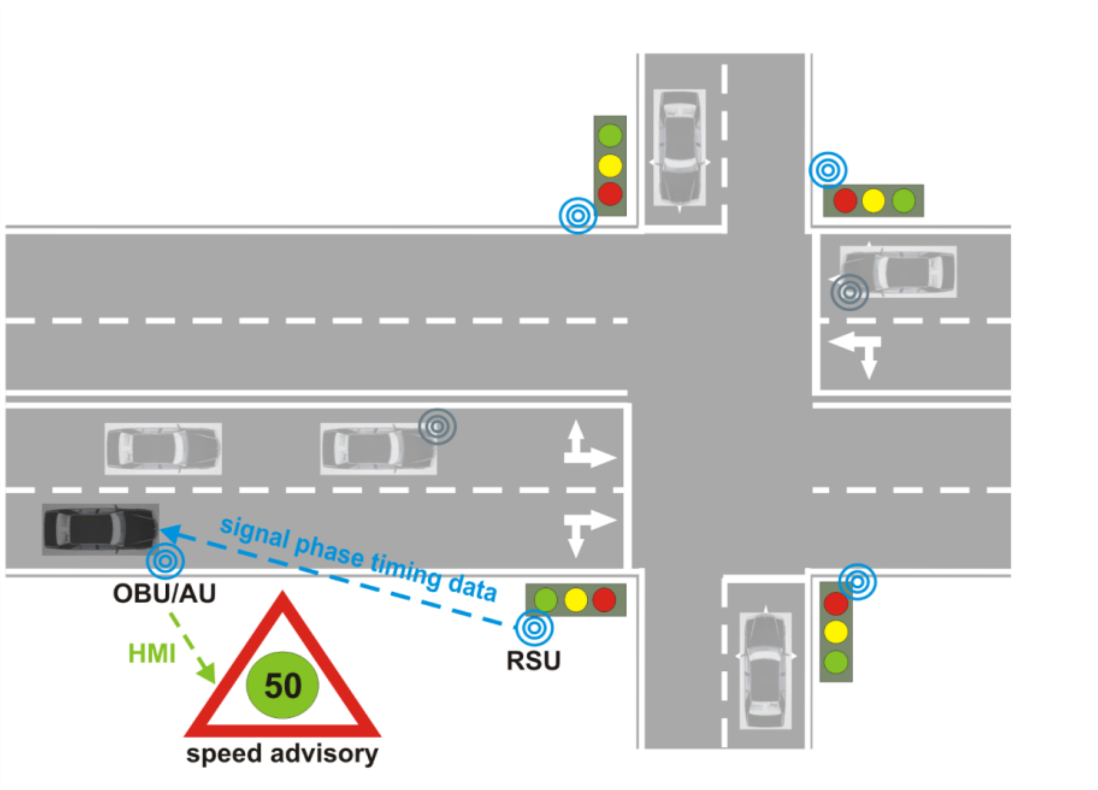

Figure 7.1: Scenario 1 with speed advisory of 50 km/h

7.2.1 Performance Indicators

The system provides a speed advice which enables the driver to pass the stop line of a traffic light controlled intersection without requiring the vehicle to stop.

The verification of the application consists mainly of

- Safety
- Efficiency
- Driver Acceptance

Table 7.2: GLOSA performance indicators

Safety		
Hypothesis: The GLOSA does not lead to hazardous situations (overtaking).	Performance Indicator: Number of hazardous situations	$A: c^{+} \leq c^{-}$
	Performance Indicator: Amount of (near) accidents	$P(x\%): c^{+} \leq c^{-}$
Efficiency		
Hypothesis: Average travel time is reduced	Performance Indicator: Average travel time for all vehicles	$P(x\%): T^{+}_{travel} < T^{-}_{travel}$
Hypothesis: Optimization regarding gasoline consumption	Performance Indicator: Mean gasoline consumption	$A: b^{+} < b^{-}$
	Performance Indicator: Gasoline consumption of a single vehicle (real world, traffic simulation)	$V(x\%): b^{+} < b^{-}$
Hypothesis: The travel times for vehicles equipped with this function are reduced.	Performance Indicator: Travel time for a vehicle (traffic simulation)	$P(x\%): T^{+}_{travel} < T^{-}_{travel}$

Hypothesis: The speed profile of a vehicle equipped with this function is harmonized.	Performance Indicator: Average absolute value of acceleration of a vehicle	$\frac{1}{T}\sum \lvert a^{+}(t)\rvert \leq x \cdot \frac{1}{T}\sum \lvert a^{+}(t)\rvert$ $0 \leq x \leq 1$
Hypothesis: Stopping times for all equipped drivers and a proportion of un-equipped (following) drivers are reduced.	Performance Indicator: Waiting/Stopping Time	$A : \sum t_h^{+} < \sum t_h^{-}$
	Performance Indicator: Number of start/stop events	$A : h^{+} < h^{-}$
	Performance Indicator: Queue length at traffic light	$A : L_Q^{+} < L_Q^{-}$
Hypothesis: Queue length at traffic lights is reduced.	Performance Indicator: Number of cars at red traffic lights	

Driver Related Indicators		
Driver Acceptance		
Hypothesis: The function is used by the driver and eases the driving task.	Performance Indicator: Rate of drivers following the speed advisory	$A : A^+ \geq x\%$
	Performance Indicator: Driver satisfaction	$A : A^+ \geq x\%$
Hypothesis: Increase of driving convenience.	Performance Indicator: Number of start/stop events	$A : h^+ < h^-$
	Performance Indicator: Average absolute value of acceleration of a vehicle	$\frac{1}{T} \cdot \lvert a^+(t) \rvert \leq x \cdot \frac{1}{T} \cdot \lvert a^+(t) \rvert$ $0 \leq x \leq 1$
Effects on Driver		
Hypothesis: The function reduces negative effects on driver.	Performance Indicator: Average heart pulse rate	$P(x\%) : {r_h}^+ < {r_h}^-$
	Performance Indicator: Maximum heart pulse rate	$P(x\%) : max({r_h}^+) < x \cdot max({r_h}^-)$ $0 \leq x \leq 1$
System Related Indicators		
Security		
Hypothesis: Open for your input	Performance Indicator: . . .	YYY

Communications		
Hypothesis: The GLOSA message will be received in time.	Performance Indicator: Message update delay	$d_{msg} < d_{threshold}$
Hypothesis: The GLOSA message is received by relevant cars.	Performance Indicator: Reception ratio in relevant area	$r_r > r_{threshold}$
Conformity with statuatories		
Hypothesis: The GLOSA gives no speed advisory that is above the speed limit.	Performance Indicator: Current speed advisory	S: advised speed ≤ speed limit

7.3 Use Case: Intersection Collision Warning

This section will depict the use case ”Intersection Collision Warning”:

Table 7.3: ICW Use Case

Basic Information	
Use Case - ID	UC - 0002
Name of Use Case	Intersection Collision Warning (ICW)
Author/Last Change	Tobias Lorenz (DLR)/15.02.2011
Short Description	Intersection collision warning systems use V2X communication and sensors to monitor traffic approaching intersections and warn drivers of approaching cross traffic as well as a possible risk of collision.
Goal/Target	• improve traffic safety at intersections – decrease the number of collisions at intersections – mitigate the collision impact
List of Actors and Components	
Actor and Component:	**Description of role:**
Driver	The driver can react according to the system information.
IVS/AU	The IVS/AU receives the position and vehicle dynamics data of the surrounding vehicles in the intersection area as well as the intersection topology, provided by the intersection IRS, and determines the information/warning concerning a possible risk of

	collision based on this information.
Intersection IRS	The IRS provides information about the intersection topology.

Additional Information

Requirement:	**Description:**
Environment	Availability of localization service (accuracy not specified yet)
Penetration Rate	at least: 2 vehicles + 1 intersection, IRS (optional)
Communication Range	A road side unit may act as relay station for the signal in case line of sight is blocked between vehicles as e.g. common in urban areas.

List of Assumptions/Boundaries (e.g. legal aspects)

Assumption:	**Description**
Traffic Participants	Only motorized vehicles are considered. Pedestrians and bicycle riders are excluded.

List of Stakeholder

Stakeholder:	**Description:**
Drivers	interested in safer driving
Road Operators	interested in fewer traffic jams
Public authorities	interested in fewer accidents

Objectives

The objective of this use case is the avoidance of collisions or at least a mitigation of collision impact at intersections. Thereby the driver is informed/warned about approaching cross traffic as well as about possible risks of collision with the cross traffic. To achieve this goal, possible driver information/warning could include e.g. an advice for speed reduction when approaching the intersection, in-vehicle signage, etc.

Scenarios and General Function

Name:	**Trigger:**	**Despription:**
1. Intersection is equipped with a V2X RSU	**a.** The ego-vehicle enters the intersection area and no cross traffic is present in the intersection area.	No advisory is given.
	b. The ego-vehicle enters the intersection area and cross traffic non-equipped with V2X communication technology is present in the intersection area.	No advisory is given.

	c. The ego-vehicle enters the intersection area and cross traffic non-equipped as well as equipped with V2X communication technology is present in the intersection area.	The driver receives the information/warning of approaching cross traffic and a possible risk of collision if necessary. The information/ warning refers to equipped vehicles only.
	d. The ego-vehicle enters the intersection area and cross traffic equipped with V2X communication technology is present in the intersection area.	The driver receives the information/warning of approaching cross traffic and a possible risk of collision if necessary.
2. Intersection is non-equipped with a V2X RSU	N/A	No advisory is given.

List of Limitations	
Limitations:	**Description:**
Scenario: Cross traffic nonequipped as well as equipped with V2X communication technology is present in the intersection area	information/warning only referring to equipped vehicles.

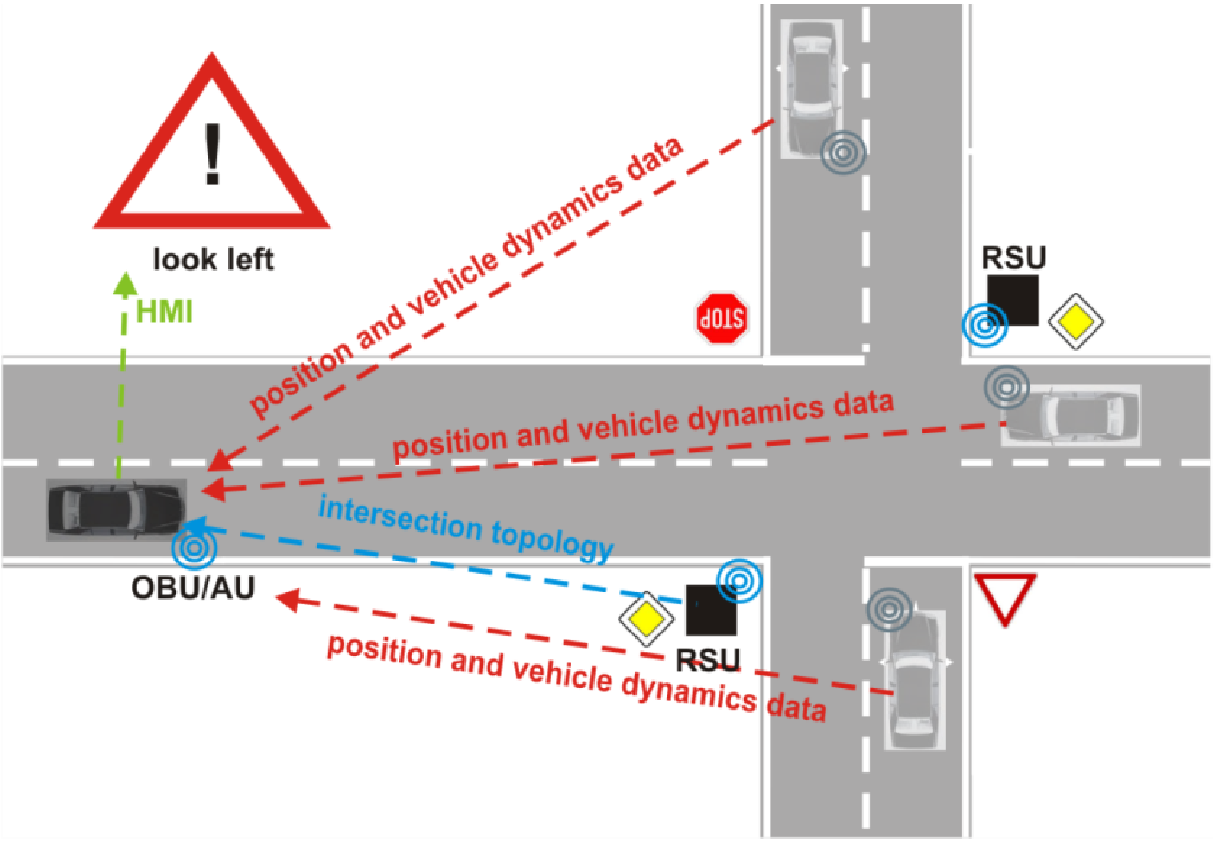

Figure 7.2: Scenario 1 with a simple intersection within a city

7.3.1 Performance Indicators

Intersection collision warning systems use V2X communication and sensors to monitor traffic approaching nonsignalized intersections and warn drivers of approaching cross traffic as well as a possible risk of collision.

The verification of the application consists mainly of

- Safety
- Efficiency
- Driver Acceptance

Table 7.4: ICW performance indicators

Safety		
Hypothesis: Number of accidents is reduced by ICW	Performance Indicator: Number of accidents	$P(x\%) : c^{+} \leq x \cdot c^{-}$ $0 \leq x \leq 1$
Efficiency		
Hypothesis: Open for your input	Performance Indicator: . . .	YYY
Driver related Indicators		
Driver Acceptance		
Hypothesis: The function is used by the driver and eases the driving task.	Performance Indicator: Rate of drivers following the given warning (driving simulator)	$A : A^{+} \geq x\%$
	Performance Indicator: Driver satisfaction (driving simulator)	$A : A^{+} \geq x\%$

Effects on Driver		
Hypothesis: The function reduces negative effects on drivers	Performance Indicator: Average heart pulse rate	$P(x\%) : r_h{}^+ < r_h{}^-$
	Performance Indicator: Maximum heart pulse rate	$P(x\%)$: $max(r_h{}^+) < x \cdot max(r_h{}^-)$ $0 \leq x \leq 1$
System Related Indicators		
Security		
Hypothesis: Open for your input	Performance Indicator: . . .	YYY
Communications		
Hypothesis: Open for your input	Performance Indicator: . . .	YYY
Security		
Hypothesis: Sender must be authenticated	Performance Indicator: . . .	-
Conformity with statuatories		
Hypothesis: The function does not give advices contradicting to traffic lights and regulations.	Performance Indicator: Accordance with current traffic lights and priorities.	YYY

7.4 Use Case: Harzardous Location Notification

Another example for a filled use case template is presented here for the use case ”Harzardous Location Notification”. It could be described as follows:

Table 7.5: HLN Use Case

Basic Information	
Use Case - ID	UC - 0003
Name of Use Case	Hazardous Location Notification (HLN)
Author/Last Change	Sebastian Röglinger (HAW Ingolstadt)/14.02.2011
Short Description	“This use case informs vehicles of any hazardous location either temporary or permanent (i.e. long term).” [13] Hazardous locations are among others bad weather (e.g. fog and rain), slippery road segments and road damages.
Goal/Target	“Reduce the risk of accident which could be caused by an Hazardous location.” [13] reduce the risk posed by hazardous locations • reduction of accidents • decrease in dangerous situations • increase in passengers’ subjective feelings of safety

List of Actors and Components	
Actor and Component:	**Description of role:**
Driver	The driver can react according to the system information.
IVS/AU	The IVSs/AUs collect localized information (e.g. concerning road condition or weather) and distribute those information to IRSs and other vehicles.
IRS	The IRSs collect information from different services (e.g. road operators, local weather services, passing vehicles) and distributes it to the passing vehicles.
Road Operator	The road operator can provide information about the road condition.
Weather Service	A local weather service can provide information about the weather.

Additional Information	
Requirement:	**Description:**
Penetration Rate	The quality of the information (e.g. timing, localization, completeness, correctness) is highly correlated with penetration rate. In cases of fewer equipped surrounding vehicles, such as shortly after system introduction or late at night, only the information from the road operators and weather services could be used.

List of Assumptions/Boundaries (e.g. legal aspects)	
Assumption:	**Description**
Correctness	The distributed information should not be invalid or outdated.
Completeness	The distributed information should cover all or as many as possible hazardous locations.
Localization	Ambiguities such as parallel road segments or elevated roads should be avoided.

List of Stakeholder	
Stakeholder:	**Description:**
Drivers	interested in information about driving circumstances to avoid surprising situations
Road Operators	interested in well flowing traffic without incidents
Public authorities	interested in less accidents

Objectives

The objective of this use case is to inform the driver about relevant hazardous locations ahead or on the route. This should support the driver by avoiding surprising and thus critical situations. To avoid a distraction of the driver, only relevant information should be presented.

Scenarios and General Function		
Name:	**Trigger:**	**Description:**
1. High Penetration Rate	**a.** Heavy rain ahead on the route detected by vehicles.	The vehicles which are already in the location of heavy rain detect the weather situation (e.g. by their rain sensors or wiper configurations) and distribute this information to neighboring vehicles and IRSs. The receiving vehicles and IRSs aggregate the information and distribute it. The IVSs/AUs additionally present relevant notifications to the driver.
	b. Heavy rain detected by weather services.	The weather services or neighboring vehicles determine local weather situations and distribute the information via V2X communication using IRSs or provide it on the internet which is available via mobile radio.
2. Low Penetration Rate	**a.** Heavy rain ahead on the route of a vehicle.	The location of heavy rain might not be detected because no equipped vehiclesare present. However, weather services or road operators could distribute the weather situation by using IRSs or the internet. However, the quality of the notification can suffer from inaccurate or less accurate information.

List of Limitations	
Limitations:	**Description:**
Notification Quality	The quality of the notifications might not be guaranteed, especially if the penetration rate is low or road operators and weather services

provide notifications of low quality.

7.4.1 Performance Indicators

This use case informs vehicles of any hazardous location either temporary or permanent (i.e. long term). Hazardous locations are among others bad weather (e.g. fog and rain), slippery road segments and road damages.

The verification of the application consists mainly of

- Safety
- Efficiency
- Driver Acceptance

Table 7.6: HLN performance indicators

Safety		
Hypothesis: More drivers arrive at hazardous location with appropriate speeds	Performance Indicator: Average speed at hazardous location	$A : v^{+} < v^{-}$
	Performance Indicator: Average speed at hazardous location	$V : v^{+} \leq MAXSPEED$
Hypothesis: Number of critical ma-noeuvers is reduced.	Performance Indicator: Absolute values of lateral ac-celerations	$P(x\%) : a^{+}_{Lat} < xa^{-}_{Lat}$ $0 \leq x \leq$
	Performance Indicator: Absolute values of longitudi-nal accelerations	$P(x\%) : a^{+}_{Lon} < xa^{-}_{Lon}$
	Performance Indicator: Maximum steering angle at this location	$P(x\%) : \delta^{+}_{max} < \delta^{-}_{max}$

Hypothesis: Number of accidents is reduced by HLN	Performance Indicator: Number of accidents	$P(x\%) : c^{+} \leq 0.5c^{-}$
Efficiency		
Hypothesis: HLN has no strong negative effects on traffic flow	Performance Indicator: Average travel time	$A : T^{+}_{travel} \leq T^{-}_{travel}$
	Performance Indicator: Traffic flow at hazardous location	$A : q^{+} \leq q^{-}$
Driver related indicators		
Driver acceptance		
Hypothesis: The function is used by the driver and eases the driving task.	Performance Indicator: Rate of drivers following the given warning	$A : A^{+} \geq 75\%$
	Performance Indicator: Driver satisfaction (real world test, driving simulator)	$A : A^{+} \geq 75\%$
Effects on driver		
Hypothesis: The function reduces negative effects on drivers	Performance Indicator: Average heart pulse rate	$P(x\%) : {r_h}^{+} < {r_h}^{-}$
	Performance Indicator: Maximum heart pulse rate	$P(x\%) : max({r_h}^{+}) < x \cdot max({r_h}^{-}$ $0 \leq x \leq 1$

System Related Indicators		
Security		
Hypothesis: Open for your input	Performance Indicator: . . .	YYY
Communications		
Hypothesis: The HLN message will be received in time.	Performance Indicator: Message update delay delay	$d_{msg} < d_{threshold}$
Conformity with statuatories		
Hypothesis: Open for your input	Performance Indicator: . . .	YYY

Glossary

A

term	abbr	definition	source
Access Layer		The access layer is an ITS data plane layer comprising the ISO/OSI physical and data link layers.	ETSI TS 102 655
Accuracy		Accuracy is the degree of conformity of a measure or calculated value to a standard or a true value.	DATE 20/05/2010: http://www.merriam-webster.com/dictionary/accuracy
Ad Hoc Network		An ad hoc network is a type of a wireless network based on self-organization without the need for a coordinating infrastructure. Nodes in an ad hoc network can have many wireless links. Typically, nodes in an ad hoc network are mobile.	ETSI TS 102 636
Advanced Driver Assistance System	ADAS	An ADAS interacts with the driver with the main purpose of supporting the driving task on the tracking and regulating levels.	ETSI TR 102 762
Application	APP	An application is a computer program that defines and implements a use case or a set of use cases. **NOTE**: A vehicular application is composed of software modules which receive data from vehicle sensors, communication networks and data bases, process them and deliver expected, consistent results to human users (via an HMI), generate communication or directly control the vehicle electronics.	ETSI TR 102 698

Application Unit	AU	The AU is a physical unit within an ITS station that executes applications and uses the communication services of a communication and control unit (CCU).	ETSI TS 102 636
Architecture		Architecture is the structure of components in a program/system, their inter-relationships as well as the principles and guidelines governing their design and evolution over time.	DATE 20/05/2010: http://www.defence.gov.au/Capability/adso/docs/PT_6-GLOSSARY.pdf

B

term	abbr	definition	source
Bandwidth		The bandwidth is the range of frequencies that allow a data channel to be transported. It is defined as the difference between the lowest and highest frequencies (measured in Hertz [Hz]) transmitted.	DATE 20/05/2010: http://www.tempus.co.uk/glossary.asp
Basic Set of Applications	BSA	The BSA is a group of mature applications set up by ETSI (European Telecommunications Standards Institute) supported by a mature, relevant vehicular communication system.	ETSI TR 102 638
Beacon		A beacon is a network layer control data packet which is sent in 1-hop broadcast mode and which includes control data on neighboring nodes e.g. identifier and position of that node.	Car2Car CC
Bit Error Rate	BER	In digital transmission, the bit error rate or bit error ratio (BER) is the number of received bits that have been altered due to noise, interference and distortion, divided by the total number of transferred bits during a studied time interval. BER is a unit less performance measure, often expressed as a percentage number.	DATE 25/04/2010: http://en.wikipedia.org/wiki/Bit_error_rate
Black-Box testing		A testing technique where the design of tests is based on just the requirements or specification of the system under test, not on knowledge about the implementation of the system.	Utting, M. and Legeard, B. Practical Model-Based Testing: A Tools Approach Morgan Kaufmann, 2007

Broadcast	A means of transmitting a message to all devices connected to a network. Normally, a special address, the broadcast address, is reserved to enable all the devices to determine that the message is a broadcast message.	Car2Car CC

C

term	abbr	definition	source
Car-Following Model		In microscopic road traffic simulations, a vehicle's longitudinal behavior is normally modelled using a car-following model, i.e. a vehicle's speed is described by a function of the distance to this vehicle's leader and this leader's velocity.	Statistical Physics of Vehicular Traffic and Some Related Systems; Debashish Chowdhury, Ludger Santen, Andreas Schadschneider; http://arxiv.org/pdf/cond-mat/0007053
Car-to-Infrastructure		see V2I	
Car-to-Car		see V2V	
Car-to-X		see V2X	
Certification		The determination that a process, vehicle, hardware or dataset meets a standard or pre-defined terms and conditions.	DATE 20/05/2010: http://www.defence.gov.au/Capability/adso/docs/PT_6-GLOSSARY.pdf
Control Station		A control station is a facility which provides the individual responsible for controlling the simulation and which provides the capability to implement simulation control.	DATE 20/05/2010: http://www.defence.gov.au/Capability/adso/docs/PT_6-GLOSSARY.pdf
Communication and Control Unit	CCU	The CCU is a physical communication unit within an ITS station that implements communication protocols and provides communication services.	ETSI TS 102 636
Communication Profile	CP	A CP is a consistent single combination of layer 1 to layer 4 protocols and technics associated to a particular media / frequency channel.	ETSI TS 102 637 - 4
Cooperative Awareness Message	CAM	A CAM is a facilities layer heartbeat message periodically transmitted by the ITS station. For vehicle ITS station, the CAM includes position, speed, direction, and basic sensor data.	ETSI TS 102 637 - 2

D

term	abbr	definition	source
Data		Representation of facts, concepts, or instructions in a formalized manner suitable for communication, interpretation or processing by humans or automatic means.	DATE 20/05/2010: http://www.defence.gov.au/Capability/adso/docs/PT_6-GLOSSARY.pdf
Data Acquisition		The process by which events in the real world are translated into machine-readable signals.	DATE 20/05/2010: http://www.novalynx.com/glossary.html
Decentralized Environmental Notification Message	DENM	The DENM is a facilities layer event driven message, providing the event type, the event position as well as the event evolution of a detected event.	ETSI TS 102 637 - 3
Derived Measure		A derived measure is a single measure calculated from a direct measure (e. g. by applying mathematical or statistical operations) or a combination of one or more direct or derived measures.	EU-project FOTNet
Direct Measure	DENM	Direct measures are measures logged directly from a sensor, without further manipulations except linear transformations.	EU-project FOTNet
Driving Efficiency		Driving Efficiency describes the effect of an application on a single vehicle (from driver’s perspective).	own

E

term	abbr	definition	source
Entity		Entities refer to those identifiable individual components within a simulation. An entity might be a platform (vehicle, pedestrian, etc.), a human being, or any other component that interacts with the simulation.	DATE 20/05/2010: http://www.defence.gov.au/Capability/adso/docs/PT_6-GLOSSARY.pdf
External Measure		An external measure is provided by data sources outside of the log equipment used in the study (e.g. measures extracted from map data, weather data bases, etc.)	EU-project FOTNet
Emulation		Technique of one machine obtaining the same results as another	WordNet 3.0, Cognitive Science Laboratory, Princeton Univ., Alexical database for the English language, last access on Sep 2010 at: http://wordnet.princeton.edu(2006)

F

term	abbr	definition	source
Facilities Layer		The facilities layer is an ITS data plane layer comprising the ISO/OSI session, presentation and application layers.	ETSI TS 102 655
Facilities		Facilities are functionalities, services or data provided by the facilities layer. **NOTE:** These functionalities, services and data are gathered in the ITS facilities layer, which contains some generic application elements (middleware), presentation and session layers of the OSI (Open System Interconnection) reference model.	ETSI TS 102 637
Failure		A failure is the symptom of a fault. For example, it is the wrong output, the crash, or the infinite loop that may happen when we execute a fault in the SUT.	Utting, M. and Legeard, B. Practical Model-Based Testing: A Tools Approach Morgan Kaufmann, 2007
Fault		A Fault is a defect in the SUT. Also known as a bug or an error. When a fault is executed, it may cause a failure.	ETSI TS 102 655
Field Operational Test	FOT	An FOT is a study to evaluate an application within an outdoor, real environment (with normal operating conditions) which is typically using a larger number of equipment (a fleet of vehicles, an equipped stretch of road) e.g. a city, or even a region.	EU-project ITS Test Beds

Functional Requirements	FR	In software engineering, a functional requirement defines a function of a software system or its component. **NOTE:** as example, a functional requirement may be defined for Application FR, communication FR, and simulation module FRs.	DATE 30/04/2010: http://en.wikipedia.org/ wiki/Functional_requirement\

G

term	abbr	definition	source
GeoNetworking		GeoNetworking is an addressing scheme for message propagation based on the geographical positions of the vehicles.	DATE: 24/06/2010: Car-to-Car Manifesto http://car-to-car.org/fileadmin/downloads/C2C-CC_manifesto_v1.1.pdf

I

term	abbr	definition	source
Identity management		The identity management is a component in the ITS station to manage different identifiers used by the station (Station identifier, node identifier, application identifier, service identifier etc.).	own
Intelligent Transportation System	ITS	An ITS represents a certain group of ADAS using information and communication technologies for gathering and providing data to improve road safety and traffic efficiency.	N/A
Interoperability		Interoperability is the ability of software and hardware on multiple machines from multiple vendors to communicate and to work together.	DATE 20/05/2010: http://dli. grainger.uiuc.edu/glossary.htm
ITS Roadside Station		An ITS roadside station is an ITS station deployed at the roadside, such as at or in a traffic light, active traffic sign or roadside wireless access point.	ETSI TS 102 636
ITS Station		An ITS station is a core element of ITS with different instantiations, such as ITS vehicle station or ITS roadside station.	ETSI TS 102 636
ITS Vehicle Station		An ITS vehicle station is an ITS station deployed in a road vehicle, such as car, truck, motor cycle or bus.	ETSI TS 102 636

L

term	abbr	definition	source
Latency		The time between stimulus and the recordable or noticeable response.	DATE 20/05/2010: http://www.merriam-webster.com/dictionary/latent+period

M

term	abbr	definition	source
Macroscopic (Traffic) Simulation		Macroscopic models aggregate the description of traffic flow and the measures of effectiveness, which are speed, flow, and density.	DATE 20/05/2010: http://swutc.tamu.edu/publications/technicalreports/167602-1.pdf
Measure		A measure is the magnitude of a quantity such as length or mass relative to a unit of measurement, such as a meter or a kilogram.	EU-project FOTNet
Microscopic (Traffic) Simulation		Microscopic models continuously or discretely predict the state of individual vehicles and primarily focus on individual vehicle speeds and locations.	DATE 20/05/2010: http://swutc.tamu.edu/publications/technicalreports/167602-1.pdf
Model		A physical mathematical or otherwise logical representation of a system, entity, phenomenon, or process.	DATE 20/05/2010: http://www.defence.gov.au/Capability/adso/docs/PT_6-GLOSSARY.pdf

N

term	abbr	definition	source
Networking Layer	NET	The networking layer is an ITS data plane layer comprising ISO/OSI layer three.	ETSI TS 102 655

O

term	abbr	definition	source
On-Board Unit	OBU	In V2X context: An OBU is an ITS Vehicle Station.	
One-Hop Broadcast		To send a data packet to all direct neighbors of a node. No further forwarding of that data packet is applied.	Car2Car CC
On-Vehicle Sensor Data		Data collected via on-vehicle sensors.	EU-project FESTA
Operational requirements	OR	Operational requirements are qualitative and quantitative parameters that specify the desired capabilities of a system and serve as a basis for determining the operational effectiveness and suitability of a system prior to deployment.	DATE: 30/04/2010: http://encyclopedia2.thefreedictionary.com/operational+requirements
Oracle (Test Oracle)		Oracle is a mechanism for analyzing SUT output and deciding whether test has passed or failed.	Utting, M. and Legeard, B. Practical Model-Based Testing: A Tools Approach Morgan Kaufmann, 2007

P

term	abbr	definition	source
Packet Error Rate	PER	The packet error rate (PER) (or symbol or block error rate) is the number of incorrectly transferred data packets (etc.) divided by the number of transferred packets. A packet is assumed to be incorrect if at least one bit is incorrect.	DATE: 30/04/2010: http://en.wikipedia.org/wiki/Bit_error_rate
Performance Indicator	PI	PIs are quantitative or qualitative measurements, agreed on beforehand, expressed as a percentage, index, rate or other value, which are monitored at regular or irregular intervals and can be compared with one or more criteria.	EU-project FESTA
Physical Time	T_p	Physical Time denotes the time represented by the physical model of the simulation. Since each simulation model is based on a real-world model, this model inherits the representation of time that originates in its physical model. **Example:** A vehicular simulation can be used to simulate two hours of rush hour traffic in an urban area.	Richard M. Fujimoto. Parallel and Distribution Simulation Systems. John Wiley and Sons, Inc., New York, NY, USA, 1999. ISBN

	These 120 minutes denote the physical time that the simulation is based on. In fact, the simulation does not simulate each millisecond of these 120 minutes, but proceeds in steps of the size of one second, thus creating an abstraction of the physical time, i.e., simulation time. For instance if the simulation takes eight minutes to complete while it is executed on a specific computer system, that time is called the wall-clock time.	0471183830.
Primary Driving Task	The primary driving task includes tasks which are related to safe driving e.g. steering, braking, etc.	ETSI TR 102 762

R

term	abbr	definition	source
Real Time	RT	A simulation where Simulation Time = Wall Clock Time is called real-time simulation system, i.e. the wall clock time progress during the simulation equals the simulation time progress.	Richard M. Fujimoto. Parallel and Distribution Simulation Systems. John Wiley and Sons, Inc., New York, NY, USA, 1999. ISBN 0471183830.
Real World		The set of real or hypothetical causes and effects that Simulation technology attempts to replicate.	DATE 20/05/2010: http://www.defence.gov.au/Capability/adso/docs/PT_6-GLOSSARY.pdf
Repeatability		Repeatability is defined as the closeness of agreement between the results of successive measurements.	ISO
Reproducibility		Reproducibility is the closeness of agreement of the results of successive measurements wherein the individual measurements are carried out with changed conditions, such as: method of measurement, observer, instrument, location, conditions of use, and time.	ISO
Requirements	Req	In engineering, a requirement is a singular documented need of what a particular product or service should be or perform. It is most commonly used in a formal sense in systems engineering or software engineering. It is a statement that identifies a necessary attribute, capability, characteristic, or quality of a system in order for it to have value and utility to a user.	DATE: 27/09/2010: http://en.wikipedia.org/wiki/Requirement
Road Side Unit	RSU	in V2X context: An RSU is an ITS Roadside Station	ETSI TR 102 762

S

term	abbr	definition	source
Scalability		Scalability is a desirable property of a system, a network, or a process, which indicates its ability to either handle growing amounts of work in a graceful manner, or to be readily enlarged.	DATE 20/05/2010: http://en.wikipedia.org/wiki/Scalability
Secondary Driving Task		The secondary driving task includes tasks which are not driving related, but refer to information and entertainment e.g. navigation system, radio, etc.	ETSI TR 102 762
Self-Reported Measure		Subjective data reported via questionnaires, interviews, focus groups, etc.	EU-project FOTNet
Sensor		A device that responds to a physical stimulus (as heat, light, sound, pressure, magnetism or a particular motion) and transmits a resulting impulse which can be interpreted as a measure by an instrument/observer.	DATE 20/05/2010: http://www.merriam-webster.com/dictionary/sensor
Service		A service is a functionality offered by an entity to another entity, e.g. by an ISO/OSI layer to the next upper ISO/OSI layer.	ETSI TS 102 655
Service Access Point	SAP	The SAP is a point of an interface between adjacent layers in which an instance of a layer of the ITS station protocol stack provides a service. The SAP is used by instances of the adjacent layers to access the service.	ETSI TS 102 636

Service announcement	SA	SA is a V2X message/packet that announce a supportive service and/or the associated communication profile to be used to access the service by an ITS station.	own
Simulation		Simulation is the reproduction of behavior of a real or imaginary system including the internal dynamic processes using a model to get knowledge which is transferable to real world.	VDE DIN 3663
Simulation Granularity		For road traffic simulations, a clear distinction between granularity classes is made: microscopic, mesoscopic, macroscopic, and sub-microscopic. This classification clearly describes how traffic flow is modeled. It is more explicit than "Simulator Level of Detail", but does not describe certain features on the other hand.	Statistical Physics of Vehicular Traffic and Some Related Systems; Debashish Chowdhury, Ludger Santen, Andreas Schadschneider; arXiv:condmat/ 0007053v1
Simulation Time	T_S	In contrast to the Physical Time which, in most cases, is based on a continuous model, simulation time forms a discretized abstraction of Physical Time that is used by the simulation system. Instead of Simulation Time, sometimes also the term Logical Time can be found in literature. **Example:** A vehicular simulation can be used to simulate two hours of rush hour traffic in an urban area. These 120 minutes denote the physical time that the simulation is based on. In fact, the simulation does not simulate each millisecond of these 120 minutes, but proceeds in	Richard M. Fujimoto. Parallel and Distribution Simulation Systems. John Wiley and Sons, Inc., New York, NY, USA, 1999. ISBN 0471183830.

		steps of the size of one second, thus creating an abstraction of the physical time, i.e., simulation time. For instance if the simulation takes eight minutes to complete while it is executed on a specific computer system, that time is called the wall-clock time.	
Simulator		A simulator is a device which employs simulation to replace a real world system or apparatus, eg for training purposes. A simulator generally has three elements – a modelled process which represents the real world system, a control system, and a man-machine interface.	DATE 20/05/2010: http://www.defence.gov.au/Capability/adso/docs/PT_6-GLOSSARY.pdf
Simulator Level of Detail	SLoD	The SLoD describes the property of a simulator to model real world effects with a certain accuracy.	own
Stakeholder		Stakeholder are stakeholder persons, groups, or organizations that has direct or indirect stake in an organization or project because it can affect or be affected by the organization’s or project’s actions, objectives, and policies.	DATE 27/09/2010: http://www.businessdictionary.com/definition/stakeholder.html

T

term	abbr	definition	source
Test Bed		A test bed represents an in-house, laboratory-like, controlled environment which is typically using one or a very limited number of mock-ups of equipment e.g. laboratory.	EU-project ITS Test Beds
Test Case		A sequence of SUT interactions. Each SUT interaction includes the SUT inputs and usually includes the expected SUT outputs or some kind of oracle for deciding whether the actual SUT outputs are correct.	Utting, M. and Legeard, B. Practical Model-Based Testing: A Tools Approach Morgan Kaufmann, 2007
Test Scenario		A test scenario describes the situation where an application will be investigated in. This includes the description of states, properties, course of action at a moment in time for each entity occurring as well as the PIs for the evaluation of the application.	SISO / Military Scenrio Definition Language (MSDL)
Test Sequence		A test sequence is a synonym for test case.	Utting, M. and Legeard, B. Practical Model-Based Testing: A Tools Approach Morgan Kaufmann, 2007
Test Site		A test site represents an outdoor, controlled environment which is typically using one or a very limited number of real equipment – real, driving vehicles, real roadside units e.g. a test track.	EU-project ITS Test Beds
Test Suite		A test suite is a collection of test cases.	Utting, M. and Legeard, B. Practical Model-Based Testing: A Tools Approach Morgan Kaufmann, 2007
Time Loss	T_{loss}	The time loss is the difference between the best time possible and the real time needed.	National-Project simTD

Traffic State		The traffic state (communication) is a parameter to define the QoS requirement (level) for a V2X message.	own
Traffic Density		The traffic density is the number of vehicles on a road per unit road length; $[\frac{vehicles}{km}]$	N/A
Traffic Efficiency		Traffic Efficiency describes the effect of an application on the whole traffic flow (from traffic flow perspective).	own
Traffic Flow		The traffic flow is the amount of vehicles passing a place on a road; $[\frac{vehicles}{hour}]$	Statistical Physics of Vehicular Traffic and Some Related Systems, Debashish Chowdhury, Ludger Santen, Andreas Schadschneider, arXiv:condmat/ 0007053v1
Traffic Occupancy		The traffic occupancy is the percentage of road covered by vehicles, [%]	Statistical Physics of Vehicular Traffic and Some Related Systems, Debashish Chowdhury, Ludger Santen, Andreas Schadschneider, arXiv:condmat/ 0007053v1
Transport Layer	TRA	The TRA is an ITS data plane layer comprising ISO/OSI layer four.	ETSI TS 102 655
Travel Time	T_{travel}	The travel time is the time needed from the starting point to destination.	Nation-Project simTD

U

term	abbr	definition	source
Usability		Usability is the set of attributes that bear on the effort needed for use, and on the individual assessment of such use, by a stated or implied set of users.	ISO 9126
Use Case	UC	A use case is a methodology used in system analysis to identify, clarify and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal.	EU-project ITS Test Beds
User Acceptance Testing	UAT	Formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system.	DATE 06/07/2010: http://www.coleyconsulting.co.uk/uatdefinition.htm

V

term	abbr	definition	source
Validation		Concerned with building the right model. It is utilized to determine that a model is an accurate representation of the real system. Validation is usually achieved through the calibration of the model, an iterative process of comparing the model to actual system behavior and using the discrepancies between the two, and the insights gained, to improve the model. This process is repeated until model accuracy is judged to be acceptable.	CSE808 Modeling and Discrete Simulation (Fall 2003), Michigan State University
Variance of travel time		The variance of travel time is a repeated measurement of travel time to a fixed point of time to get the distribution.	National-Project simTD
Vehicle-to-Infrastructure	V2I	V2I communication is the data exchange between vehicles and ITS roadside stations.	own
Vehicle-to-Vehicle	V2V	V2V communication is the data exchange between vehicles only.	own
Vehicle-to-X	V2X	V2X communication is the data exchange between vehicles and/or vehicles and ITS roadside stations.	own
Verification		Concerned with building the model right. It is utilized in the comparison of the conceptual model to the simulation representation that implements that conception. It asks the questions: Is the model implemented correctly in the simulation? Are the input parameters and logical structure of the model correctly represented?	CSE808 Modeling and Discrete Simulation (Fall 2003), Michigan State University

W

term	abbr	definition	source
Wall Clock Time	T_W	Wall Clock Time denotes the actual time that passes while the simulation is executed. **Example:** A vehicular simulation can be used to simulate two hours of rush hour traffic in an urban area. These 120 minutes denote the physical time that the simulation is based on. In fact, the simulation does not simulate each millisecond of these 120 minutes, but proceeds in steps of the size of one second, thus creating an abstraction of the physical time, i.e., simulation time. For instance if the simulation takes eight minutes to complete while it is executed on a specific computer system, that time is called the wall-clock time.	Richard M. Fujimoto. Parallel and Distribution Simulation Systems. John Wiley and Sons, Inc., New York, NY, USA, 1999. ISBN 0471183830.
White-box testing		White-box testing is a box testing technique wherein the design of tests uses knowledge about the implementation of the system (e.g., tests designed to ensure that all statements of the implementation code are tested).	Utting, M. and Legeard, B. Practical Model-Based Testing: A Tools Approach Morgan Kaufmann, 2007

References

[1] Cognitive Science Laboratory, Princeton University, "WordNet 3.0 - a lexical database for the english language," http://wordnet.princeton.edu, September 2010.

[2] A. van Lamsweerde, "Requirements engineering - from system goals to uml models to software specifications," Wiley, 2009.

[3] K. Pohl, "Requirements Engineering - Grundlagen, Prinzipien, Techniken," dpunkt.verlag, Heidelberg, 2008.

[4] D. Leffingwell and D. Widrig, "Managing software requirements: a unified approach," Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 2000.

[5] D. Gettman and L. Head, "Surrogate safety measures from traffic simulation models, final report FHWA-RD-03-050," Federal Highway Administration, 2003.

[6] F. Cunto, "Evaluation of safety countermeasures at intersections using microscopic simulation," Revista Tecnologia Fortaleza, vol. 28, pp. 110–120, 2007. [Online]. Available: http://www.unifor.br/notitia/file/1309.pdf

[7] Siemens Energy and Automation, "Surrogate safety assessment model (SSAM)," http://www.itssiemens.com/research/ssam/, 2008.

[8] R. Blokpoel, D. Krajzewicz, and R. Nippold, "Unambiguous Metrics for Evaluation of Traffic Tetworks," in Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on, 2010, pp. 1277 –1282.

[9] T. Benz, R. Kernchen, M. Killat, A. Richter, and B. Schünemann, "A comprehensive simulation tool set for cooperative systems," Advanced Microsystems for Automotive Applications 2010: Smart Systems for Green Cars and Safe Mobility, pp. 411–422, May 2010.

[10] B. Schünemann, J. W. Wedel, and I. Radusch, "V2X-based traffic congestion recognition and avoidance," Tamkang Journal of Science and Engineering, vol. 13, no. 1, pp. 63–70, March 2010. [Online]. Available: http://www2.tku.edu.tw/_tkjse/13-1/catology.htm

[11] iTETRIS, "An Integrated Wireless and Traffic Platform for Real-Timeroad Traffic Management Solutions," http://www.ict-itetris.eu/10-10-10-community/, 2011.

[12] D. Chowdhury, L. Santen, and A. Schadschneider, "Statistical physics of vehicular traffic and some related systems," Physics Reports, vol. 329, no. 4-6, pp. 199–329, 2000.

[13] ETSI, Intelligent Transport Systems (ITS), "TR 102 638, vehicular communications, basic set of applications, definitions, v1.1.1," ETSI, 2009.

[14] ——, "TS 102 894, user & application requirements; facility layer structure, functional requirements and specifications, v0.0.6," February 2011.

[15] F. Köster, J. Gačnik, and M. Hannibal, "Serviceorientierung als Zugang zur Strukturierung von In-Vehicle Softwaresystemen" in 4. Braunschweiger Symposium: IMA 2008 Informationssysteme für mobile Anwendungen, Braunschweig, 2009.

[16] J. Gačnik, O. Häger, and M. Hannibal, “A service-oriented system architecture for the human centered design of intelligent transportation systems,” in European Conference on Human Centered Design for Intelligent Transport Systems, Lyon, 2008.

[17] F. Köster, M. Hannibal, T. Frankiewicz, and K. Lemmer, “Anwendungsplattform Intelligente Mobilität (AIM) – Dienstespektrum und Architektur” in AAET 2011. Automatisierungssysteme, Assistenzsysteme und eingebettete Systeme für Transportmittel, 2011.

[18] V. Kumar, L. Lin, D. Krajzewicz, F. Hrizi, O. Martinez, J. Gozalvez, and R. Bauza, “iTETRIS: Adaptation of its Technologies for Large Scale Integrated Simulation,” in 3rd IEEE International Symposium on Wireless Vehicular Communications: IEEE WiVEC 2010, Taiwan, 2010.

[19] D. Krajzewicz, R. Blokpoel, F. Cartolano, P. Cataldi, A. Gonzalez, O. Lazaro, J. Leguay, L. Lin, J. Maneros, and M. Rondinone, “iTETRIS - A System for the Evaluation of Cooperative Traffic Management Solutions,” in 14th International Forum on Advanced Microsystems for Automotive Applications (AMAA 2010), Berlin, 2010.

[20] D. Rieck, B. Schünemann, I. Radusch, and C. Meinel, “Efficient traffic simulator coupling in a distributed V2X simulation environment,” in SIMUTools ’10: Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2010, pp. 1–9.

[21] J. W. Wedel, B. Schünemann, and I. Radusch, “V2X-based traffic congestion recognition and avoidance,” Parallel Architectures, Algorithms, and Networks, International Symposium on, vol. 0, pp. 637–641, 2009.

[22] N. Naumann, B. Schünemann, I. Radusch, and C. Meinel, “Improving V2X simulation performance with optimistic synchronization,” in Services Computing Conference, 2009. APSCC 2009. IEEE Asia-Pacific, Dec. 2009, pp. 52–57.

[23] N. Naumann, B. Schünemann, and I. Radusch, “VSimRTI - simulation runtime infrastructure for V2X communication scenarios,” in Proceedings of the 16thWorld Congress and Exhibition on Intelligent Transport Systems and Services (ITS Stockholm 2009). ITS Stockholm 2009, September 2009.

[24] T. Queck, B. Schünemann, I. Radusch, and C. Meinel, “Realistic simulation of V2X communication scenarios,” in APSCC ’08: Proceedings of the 2008 IEEE Asia-Pacific Services Computing Conference. Washington, DC, USA: IEEE Computer Society, 2008, pp. 1623–1627.

[25] T. Queck, B. Schünemann, and I. Radusch, “Runtime infrastructure for simulating Vehicle-2-X communication scenarios,” in VANET ’08: Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking. New York, NY, USA: ACM, 2008, pp. 78–79.

[26] R. Protzmann, B. Schünemann, and I. Radusch, “The influences of communication models on the simulated effectiveness of V2X applications,” in Vehicular Networking Conference (VNC), 2010 IEEE, Dec. 2010, pp. 102 –109.

[27] A. Wegener, M. Piórkowski et al., “Traci: An interface for coupling road traffic and network simulators,” Proceedings of the 11th Communications and Networking Simulation Symposium (CNS’08), Ottawa, Canada, 2008.

[28] D. Krajzewicz, M. Bonert et al., "The open source traffic simulation package SUMO," RoboCup 2006, 2006-06-14 - 2006-06-20, Bremen, Germany, 2006.

[29] D. Krajzewicz, D. T. Boyom et al., "Untersuchungen der Performanz einer auf C2CKommunikation basierenden, autonomen Routenwahl bei Stauszenarien," Heureka '08, 2008-03-05 - 2008-03-06, Stuttgart, Germany, 2008.

[30] J. Härri, D. T. Boyom et al., "Car-2-X simulations: Tools, methodologies and performance results," Proceedings of the 5th InternationalWorkshop on Intelligent Transportation (WIT'08), Hamburg, Germany, 2008.

[31] S. Wang and C. Chou, "NCTUns tool for wireless vehicular communication network researches," Simulation Modelling Practice and Theory, Vol. 17, No. 7, pp. 1211-1226, 2009.

[32] M. Balmer, M. Rieser et al., "MATSim-T: Architecture and simulation times," Multi-Agent Systems for Traffic and Transportation Engineering, chapter 3, pages 57-78. IGI Global, 2009.

[33] A. Wegener, H. Hellbrück, C. Wewetzer, and A. Lübke, "VANET simulation environment with feedback loop and its application to traffic light assistance: New Orleans, Louisiana, USA, 30 November - 04 December 2008," Piscataway, NJ, 2008.

[34] S. Assenmacher, M. Killat, F. Schmidt-Eisenlohr, and P. Vortisch, "A simulative approach for the identification of possibilities and impacts of V2X-communication," in 15th World Congress on Intelligent Transport Systems, 2008, delivered but not published due to copyright.

■ End of Document ■

A Simulators & Models and Simulation Studies

A.1 Simulators & Models

A.1.1 Integrated Simulators

Matlab

Name	Matlab
Partner	DLR
Legal Status	Customer
Hardware Environment	PC
Operating System	Linux, Mac OS X, Windows
Programming/Scripting Language	Matlab
Availability (estimated costs)	commercial
Additional Software needed	none
Simulation Entities	Data packets, channels, communication link
Type of Simulation	micro
Link to V2X	C++ models can be derived for coupling with other simulators (e.g., PER models to be coupled with traffic)
Documents and Papers	n/a
Typical and maximal size of APP	Typical: a single link (Tx and Rx) Interference can be added as well
Typical and maximal time frame of APP	Typically used to derive statistics over many packets
Typical Result	Bit error rate (BER) (or packet error rate (PER)) statistics as function of channel, received SNR and modulation and code rate
Typical Runtime	Typical: to derive sufficient BER or PER statistics, typically many simulations required.
Specific characteristics (V2X-related)	• IEEE 802.11p channel model • IEEE 802.11p modulation and code rate • IEEE 802.11p front-end non-idealities can be modeled • Cumulative noise
Other important information	The standard tool for detailed physical modelling

A.1.2 Flexible Coupling

CANoe

Name	CANoe.Car2x
Partner	Vector Informatik GmbH
Legal Status	
Hardware Environment	PC
Operating System	Windows
Programming/Scripting Language	• CAPL (C-like programming language) • APIs for .Net languages, COM, OCX • APIs for external tools like Matlab • XML • Custom DLLs
Availability (estimated costs)	commercial
Additional Software needed	none
Simulation Entities	Communication, Systems, Networks, Devices.
Type of Simulation	micro

Link to V2X	• Gateway capabilities for other in-vehicle networks • Simulation of radio frames • Support of communication protocols • Test Feature Set for automated test execution • Supported of 802.11p Hardware to send and receive radio frames • Analyses of radio frames (protocols like Network Header, Geonetworking Basic Transport Protocol, access to protocol fields) • WLAN Packet Builder to send single frames • Simulation of multiple nodes (e.g. ITS stations), gateway to other networks like CAN can be implemented in a Gateway node • Integrated Test Feature Set for automatic testing • Interpretation of application messages like CAM, DENM • GPS map to display multiple ITS stations and their behavior • Possibility to setup a conformance test
Documents and Papers	http://www.vector.com/vi_car2x_en.html
Typical and maximal size of APP	Typical: 1-30 nodes Max: No restrictions
Typical and maximal time frame of APP	A couple of minutes up to multiple hours/days
Typical Result	• Make statements about network behavior (from communication point of view) • Generated test reports with Test Feature Set for quality assurance • Log files of network traffic
Typical Runtime	A couple of minutes up to multiple hours/days (Controlled by the user)
Specific characteristics (V2X-related)	Available for all kind of automotive networks
Other important information	

DOMINION

Name	DOMINION
Partner	Deutsches Zentrum für Luft- und Raumfahrt e.V. - Institut für Verkehrssystemtechnik (DLR/TS)
Legal Status	Developer
Hardware Environment	PC, Mobile Devices
Operating System	Linux, Windows
Programming/Scripting Language	C/C++
Availability (estimated costs)	
Additional Software needed	none
Simulation Entities	Applications, Communication, Driving Simulators.
Type of Simulation	micro
Link to V2X	• Middleware concept for coupling different simulators and applications from different domains • Communication Simulation • RTE for driver acceptance studies supports HiL, SiL tests • Traffic Simulation with different simulators • Relevant interfaces are available in an API
Documents and Papers	References: [15, 16]
Typical and maximal size of APP	Typical: 1-50 nodes Max: limited by hardware
Typical and maximal time frame of APP	200 Hz - 1 Hz
Typical Result	• database with the selected relevant values over time
Typical Runtime	Not restricted
Specific characteristics (V2X-related)	• used in the Application-Platform Intelligent Mobility (AIM) of the DLR [17] • RTE for all research facilities at TS • provides interfaces to CAN

Other important information	• DOMINION is SoA based (Serviceoriented-Architecture) • provides interfaces via an API • supports the use of Java • allows the use of an application in all research facilities of TS without any software changes • enables the coupling of driving simulators (simulator-FOT) *MoSAIC*

iTETRIS

Name	iTETRIS Large Scale ITS Simulator
Partner	EURECOM & DLR
Legal Status	Developer
Hardware Environment	PC
Operating System	Linux, Windows Coupled simulators can run remotely on other machines and under other operating systems
Programming/Scripting Language	C++, Python, Java
Availability (estimated costs)	GPL License
Additional Software needed	Need to install NS-3 and SUMO
Simulation Entities	Applications, Communication, Driving Simulators.
Type of Simulation	micro
Link to V2X	• Interfacing concept, the iTETRIS Control System (iCS), for coupling different simulators and applications from different domains • Communication Simulation with NS-3 • Traffic Simulation with SUMO • Application Simulation following Multilanguage APIs • Implementation of the Facilities layer in the iCS and NS-3 • Relevant interfaces are available following push/pull APIs
Documents and Paper	• iTETRIS Community (http://www.ict-itetris.eu/10-10-10-community/) • iTETRIS Manual • Reference: [11, 18, 19]
Typical and maximal size of APP	No restrictions by iTETRIS, depends on coupled simulators. Typically: mid-size cities
Typical and maximal time frame of APP	No restrictions by iTETRIS, depends on coupled simulators

Typical Result	• iTETRIS can provide results from all components specified or requested by users. • iTETRIS can also provide gas emission statistics as well as fuel consumption, as part of SUMO functionalities.
Typical Runtime	Depends on the runtime of the coupled simulators
Specific characteristics (V2X-related)	• The current release of iTETRIS couples NS-3 and SUMO. • The iTETRIS can also be run based on a subset of the available components, choice left to the user. • Can simulate several applications running on one OBU/RSU • Can simulate OBUs or RSUs (non-IP stack), IP-stations, and TLC. • Can simulate Facilities-related functions: Technology selector (802.11p, WLAN, UMTS, DVBH)
Other important information	The application simulator of iTETRIS is not restricted to any particular programing language. It contains a set of APIs the application should be compliant with to interact with the iCS. APIs in C++, Java and Python are provided by default. The iCS represents more than a synchronization interface between components, as it also contains an implementation of the facilities layer as specified in the ETSI as well as a Local Dynamic Map (LDM) for data management. The NS-3 version coupled with iTETRIS has been extended from NS-3.7 to support the ETSI TC ITS protocol stacks. A patch for NS-3 is provided by iTETRIS. An installation manual guides the reader to the installation of the iCS, SUMO and NS-3.

VSimRTI

Name	VSimRTI V2X Simulation Runtime Infrastructure
Partner	Fraunhofer FOKUS
Legal Status	Developer
Hardware Environment	PC
Operating System	Linux, Windows Coupled simulators can run remotely on other machines and under other operating systems
Programming/Scripting Language	Java
Availability (estimated costs)	Free Research License
Additional Software needed	No additional software needed. For immediate use, a set of simulators is already coupled with VSimRTI: the traffic simulators VISSIM and SUMO, the communication simulators JiST/SWANS and OMNeT++, the application simulator VSimRTI-App, and several visualization and analysis tools. Moreover, it is possible to couple arbitrary simulation systems with a remote control interface.
Simulation Entities	All entities/features supported by the coupled traffic simulators, communication simulators, application simulators, visualization and analysis tools, and by all other coupled components.
Type of Simulation	discrete event-based

Link to V2X	The V2X Simulation Runtime Infrastructure VSimRTI is a lightweight framework for the preparation and execution of V2X simulations. VSimRTI couples different simulators thus allowing the simulation of the various aspects of future intelligent transportation systems. The easy integration and exchange of simulators enables the coupling of the most relevant simulators for a realistic presentation of vehicle traffic, emissions, wireless communication, and the execution of V2X applications. All management tasks, such as synchronization, interaction and lifecycle management are handled completely by VSimRTI. Several optimization techniques, such as optimistic synchronization, enable high performance simulations by VSimRTI. Various configuration options and enhanced user documentation assure maximum usability. The integrated application simulator VSimRTIApp provides a sandbox with interfaces available in real vehicles. Thus, real V2X applications can be adapted for the simulation environment with less effort. Special V2X features, e.g. traffic lights, roadside stations, both CAM and DENM types, application-triggered slow-down, application triggered lane changing, and application triggered route changing, are supported by VSimRTI.
Documents and Papers	• VSimRTI User Manual • Reference: [10, 20–26]
Typical and maximal size of APP	No restrictions by VSimRTI, depends on coupled simulators.
Typical and maximal time frame of APP	No restrictions by VSimRTI, depends on coupled simulators
Typical Result	All results, delivered by the coupled simulation components, are available. The desired file and data format of the simulation results can be specified by the user. Furthermore, results can be stored in a database directly. Visualization and analysis tools for simulation monitoring and evaluation of the results available.
Typical Runtime	Depends on the runtime of the coupled simulators

Specific characteristics (V2X-related)	In contrast to existing fixed simulator couplings, the VSimRTI simulation infrastructure allows the easy integration and exchange of simulators. Thus, the high flexibility of VSimRTI enables the coupling of the most appropriate simulators for a realistic presentation of vehicle traffic, emissions, wireless communication, and the execution of V2X applications. Depending on the specific requirements of a simulation scenario, the most relevant simulators can be used.
Other important information	VSimRTI uses an ambassador concept inspired by some fundamental concepts of the High Level Architecture (HLA). Thus, it is possible to couple arbitrary simulation systems with a remote control interface. Attaching an additional simulator only requires that the ambassador interface is implemented and then the specified commands are executed. For immediate use, a set of simulators is already coupled with VSimRTI. For example, the traffic simulators VISSIM and SUMO, the communication simulators JiST/SWANS and OMNeT++, the application simulator VSimRTI-App, and several visualization and analysis tools are prepared for VSimRTI.

VSimRTI-App

Name	VSimRTI-App - Application simulator for the simulation of V2X applications
Partner	Fraunhofer FOKUS
Legal Status	Developer
Hardware Environment	PC, Server
Operating System	Linux, Windows
Programming/Scripting Language	Java
Availability (estimated costs)	Free Research Licence
Additional Software needed	none
Simulation Entities	Communication simulator, traffic simulator coupled to VSimRTI
Type of Simulation	applications
Link to V2X	
Documents and Papers	[10, 20–26]
Typical and maximal size of APP	Maximal: unlimited Typical: mid-sized cities
Typical and maximal time frame of APP	Maximal: unlimited Typical: 1 second
Typical Result	Depends on simulated application
Typical Runtime	Depends on combined simulators
Specific characteristics (V2X-related)	• Several applications of one OBU/RSU can be simulated at the same time • Detailed configuration of application distribution for different vehicles/RSUs possible • Simulation of RSUs, OBUs and Traffic Light applications possible
Other important information	

A.1.3 Traffic Simulation

eWorld

Name	eWorld traffic simulator extension for the integration of additional features and the analysis of simulation results
Partner	Fraunhofer FOKUS
Legal Status	Customer (Open Source) Developer of plug-ins
Hardware Environment	PC
Operating System	Linux, Windows
Programming/Scripting Language	Java
Availability (estimated costs)	open source
Additional Software needed	none
Simulation Entities	Environmental / Traffic Events
Type of Simulation	micro (individual network nodes)
Link to V2X	• Enrichment of C2X simulation via environmental event notification • Feature to edit and export map data • Visualization and evaluation of Simulation • Interface to VSimRTI available • Extendible interfaces via custom adapter plug-ins
Documents and Papers	• eWorld User Manual • eWorld Developer Manual • Large documentations and source code free available: see http://eworld.sourceforge.net
Typical and maximal size of APP	Typical: 10-100 sqkm Maximal: Unlimited, memory critical
Typical and maximal time frame of APP	Unlimited
Typical Result	• Event notification messages that can be consumed by other simulation components (VSimRTI Integration available) • Charts and Tables for evaluation of simulation result

Typical Runtime	• Depending on the size of the map and the Number of events. • Usually faster than real-time
Specific characteristics (V2X-related)	Enrichment of map data and integration of environmental events in C2X simulations can facilitate realistic simulation scenarios. For instance hazardous weather conditions or blocked lanes caused by accidents.
Other important information	Open Source Software that can be extended for custom simulation purposes using a well documented plug-in architecture.

SUMO – Simulation of Urban Mobility

Name	SUMO – Simulation of Urban MObility
Partner	DLR
Legal Status	Developer
Hardware Environment	PC, Server
Operating System	Linux, Windows
Programming/Scripting Language	C++, additional tools in Python
Availability (estimated costs)	open source
Additional Software needed	none
Simulation Entities	vehicles
Type of Simulation	micro (individual vehicles and routes)
Link to V2X	• TraCI (http://sumo.sourceforge.net/wiki/index.php/TraCI), [27] • TraNS (http://trans.epfl.ch/) • Internal communication model (http://elib.dlr.de/50466/)
Documents and Papers	• Project description (http://elib.dlr.de/46740/), [28] • Project web site (http://sumo.sourceforge.net/) • C2X usage: [29]
Typical and maximal size of APP	Not restricted (only by system memory) Typical: mid-sized cities
Typical and maximal time frame of APP	Typically 1 day, maximum $2^{1}6_{s}$
Typical Result	• Network statistics (mean speed, mean travel time, number of halts,) per edge • Vehicle statistics (trip duration, number of halts,) • Simulated detectors
Typical Runtime	• Depends on scenario size • Approx. 200000 vehicle movements in real time

Specific characteristics (V2X-related)	• Unrestricted in size/vehicle number • Portable • Open source • Import modules for different network formats
Other important information	

VanetMobiSim

Name	VanetMobiSim
Partner	EURECOM & INSA Lyon
Legal Status	Developer
Hardware Environment	PC
Operating System	Linux, Windows, Mac OS
Programming/Scripting Language	Java and Eclipse
Availability (estimated costs)	open source
Additional Software needed	none
Simulation Entities	vehicles, road network
Type of Simulation	micro, macro, social
Link to V2X	• Agent-based modeling - detailed driving modeling of each vehicle • Built-in various driver models
Documents and Paper	• VanetMobisim Website (http://sourceforge.net/projects/vanetmobisim/)
Typical and maximal size of APP	Metropolitan area highway and road networks, urban road networks. Restriction only by system memory.
Typical and maximal time frame of APP	Typical: 1hr 1 day
Typical Result	• mobility traces for network simulators (NS-3, Qualnet) • Traffic flow, traffic density, average speed, travel time measurements also obtained based on external scripts
Typical Runtime	Depends highly on number of vehicles and size of network
Specific characteristics (V2X-related)	• GUI-based Real Map import • Multi-traffic class support • Basic Embedded V2X communication support
Other important information	• GUI-based configuration and run • Intelligent Driver Model (IDM) and MOBIL lane changing modeling

VISSIM

Name	VISSIM Verkehr In Städten - Simulation
Partner	TUM-VT
Legal Status	Customer (PTV AG)
Hardware Environment	PC, Server
Operating System	Windows
Programming/Scripting Language	C++, VB, Scripting in Python, Java via COM bridge
Availability (estimated costs)	commercial
Additional Software needed	none
Simulation Entities	Vehicles, Road Network
Type of Simulation	micro
Link to V2X	• Dedicated C2X Interface • Detailed driving behavior analysis regarding C2X and ADAS • C2X API for C++ / Python scripting C2X Apps • Comprehensive COM Interface
Documents and Papers	• PTV VISION VISSIM 5.20 User Manual • VISSIM Website(http://www.ptvag.com/software/transportation-planning-traffic-engineering/software-system-solutions/vissim/)
Typical and maximal size of APP	Metropolitan area highway and road networks, urban road networks. Restriction only by system memory.
Typical and maximal time frame of APP	Typical: 1hr 1 day Maximal: ca. 11 days
Typical Result	• Traffic flow, traffic density, average speed • Travel time measurements • Vehicle parameters • Queue lengths, congestion durations, number of stops • Signal times and changes
Typical Runtime	Depends highly on number of vehicles, size of network, use of external modules (as C2X or signal control)

Specific characteristics (V2X-related)	• Real-world applications can be run without modification • Open source • Seamless integration of real nodes into simulated networks
Other important information	• Enhanced Wiedemann driving behavior model • Exact modeling of lane geometry and network element positions • Interface to macroscopic VISUM simulation • 2D & 3D GUI

A.1.4 Communication Simulation

The Network Simulator - NS-2

Name	The Network Simulator - ns-2.34
Partner	Fraunhofer ESK
Legal Status	
Hardware Environment	PC, Server
Operating System	Linux, Mac OS X, Windows (via Cygwin)
Programming/Scripting Language	C++, OTcl
Availability (estimated costs)	open source, GPL
Additional Software needed	none
Simulation Entities	Network nodes
Type of Simulation	micro (discrete event simulation)
Link to V2X	• Many technologies and protocols implemented • Widely used in research for many years • Interfaces to SUMO exist • TraCI client patch to interact with SUMO (http://sumo.sourceforge.net/wiki/index.php/TraCI) • TraNS (http://trans.epfl.ch/)
Documents and Papers	• Project web site (http://www.isi.edu/nsnam/ns/) • Official wiki (http://nsnam.isi.edu/nsnam) • C2X usage: [30]
Typical and maximal size of APP	Not restricted (only by system memory), some users report difficulties with more then 3000nodes Typical: 5-500 nodes
Typical and maximal time frame of APP	Not restricted, depends on simulated scenario
Typical Result	Trace file with: • Packet related events (sent, received, dropped) • Packet related information (type, size, ttl, ...) • Node positions
Typical Runtime	• Depends on scenario size and packet rate • Varies from few minutes to several days

Specific characteristics (V2X-related)	• Portable • Open source • Many different networking technologies and protocols available
Other important information	• Development focus is on ns-3 now (only few releases in recent years, mainly bug fixes) • Visualization: Nam (http://www.isi.edu/nsnam/nam)

The Network Simulator - NS-3

Name	The Network Simulator - ns-3.7
Partner	Fraunhofer ESK
Legal Status	
Hardware Environment	PC, Server
Operating System	Linux, Mac OS X, Windows (via Cygwin)
Programming/Scripting Language	C++, Python (optional)
Availability (estimated costs)	open source, GPL
Additional Software needed	none
Simulation Entities	Network nodes
Type of Simulation	micro (discrete event simulation)
Link to V2X	• Many technologies and protocols implemented • 802.11p WiFi channel integrated since version 3.7 • VANET integration in review process (http://codereview.appspot.com/179068) • Integration with SUMO planned as part of the iTetris project (http://www.ict-itetris.eu)
Documents and Papers	• Project web site (http://www.nsnam.org) • Official wiki (http://www.nsnam.org/wiki)
Typical and maximal size of APP	Not restricted (only by system memory), some users report difficulties with more than 3000 nodes Typical: 5-500 nodes
Typical and maximal time frame of APP	Not restricted, depends on simulated scenario
Typical Result	Traces of packet level events in different formats (e.g. ASCII, pcap)
Typical Runtime	• Depends on scenario size and packet rate • Varies from few minutes to several days
Specific characteristics (V2X-related)	• Portable • Open source • Many different networking technologies and protocols available

Other important information	• Still under development • Reliable visualization tools for simulations of wireless communication still missing

NCTUns

Name	NCTUns 6.0 Network Simulator and Emulator
Partner	Fraunhofer ESK
Legal Status	
Hardware Environment	PC
Operating System	Linux (NCTUns 6.0 supports Fedora 11 directly)
Programming/Scripting Language	C++
Availibility (estimated costs)	open source
Additional Software needed	none
Simulation Entities	Network nodes
Type of Simulation	Combines discrete event simulation with kernel re-entering simulation methodology (a reallife UNIX kernel s protocol stack is directly used)
Link to V2X	• Integrated traffic simulator • Pre-defines network nodes for cars and road side units
Documents and Papers	• Project Website (http://nsl.csie.nctu.edu.tw/nctuns.html) • [31]
Typical and maximal size of APP	Not restricted (supports cluster computing)
Typical and maximal time frame of APP	Not restricted (supports cluster computing)
Typical Result	Output generated by the applications in use
Typical Runtime	Depends on simulated scenario, finishes quickly due to kernel re-entering methodology
Specific characteristics (V2X-related)	• Real-world applications can be run without modification • Open source • Seamless integration of real nodes into simulated networks

Other important information	• Graphical network editor • Well documented • Small community compared to other simulation tools • Continuously maintained and improved

OMNeT++

Name	OMNeT++
Partner	Fraunhofer ESK
Legal Status	
Hardware Environment	PC
Operating System	Windows, Linux, Mac OS X
Programming/Scripting Language	C++, NED (Network Description Language)
Availability (estimated costs)	• Free for academic use (OMNet++) • Commercial version: OMNEST (Simulcraft)
Additional Software needed	• INET Framework (for IP, TCP, UDP...) • Mobility Framework (wireless networking) • MiXiM (wireless networking)
Simulation Entities	Modules (e.g. network nodes, protocols)
Type of Simulation	micro (discrete event simulation)
Link to V2X	• Many technologies and protocols implemented • Used in research for many years • Interfaces to SUMO exist • TraCI client modules (SUMO integration) available: http://www7.informatik.unierlangen.de/~sommer/omnet/traci/
Documents and Papers	• Project website (http://www.omnetpp.org) • INET website (http://inet.omnetpp.org/)
Typical and maximal size of APP	Not restricted (with MPI support)
Typical and maximal time frame of APP	Not restricted (with MPI support)
Typical Result	Vector and scalar files (text format)
Typical Runtime	Depends on scenario size and packet rate
Specific characteristics (V2X-related)	• Portable • Modular design encourages extension by the community • Many P2P protocols implemented
Other important information	• Graphical network editor

MATSim

Name	MATSim - Multi-Agent Transport Simulation
Partner	TUM-VT
Legal Status	Developer
Hardware Environment	PC; Maximum scenario below uses 8 dual-core processors with 2.2 GHz clock rate each and about 22 GByte of RAM
Operating System	Any (Java)
Programming/Scripting Language	Java
Availability (estimated costs)	open source
Additional Software needed	Java Development Environment (e.g. Eclipse, Netbeans)
Simulation Entities	agents
Type of Simulation	Agent-based, activity modelling, mesoscopic traffic flow (individuals vehicles and routes, flow via (extended) queue model)
Link to V2X	• Individual agent plans (allow rerouting) • Plans can be altered/improved between iterations (long term-effect simulation) • Used in COOPERS Project (Realtime rerouting simulation) • Custom solutions possible via new Java Class instantiation
Documents and Papers	• Project web site (http://www.matsim.org/) • Reference: [32] • Further publications available from: http://www.matsim.org/publications
Typical and maximal size of APP	Metropolitan area highway and road networks, urban road networks. Restriction only by system memory.
Typical and maximal time frame of APP	Not restricted (only by system memory/CPU power) Typical: entire cities to medium-sized countries: Several million agents, several thousand links Maximum: 24,000 nodes, 60,000 links, 1.7 million facilities, five different activities, 7 million agents, 22 million trips of which 7.1 million are motorized individual transport trips.

Typical Result	• Network statistics (mean speed, mean travel time, number of halts,) per edge • Vehicle statistics (trip duration, number of halts)
Typical Runtime	• Depends on scenario size; hardware/ scenario here needs 5 days for 100 iterations • *Whole days within minutes* Project Website
Specific characteristics (V2X-related)	• Unrestricted in size/vehicle number • Portable (Java), Open source • Highly modular software structure easily extendable and many custom modules contributed by users
Other important information	Data taken from paper in Documents and Papers section above.

JiST/SWANS

Name	JiST/SWANS Scalable Wireless Ad Hoc Network Simulator
Partner	Fraunhofer FOKUS
Legal Status	Customer
Hardware Environment	PC, Server
Operating System	Linux, Windows
Programming/Scripting Language	Java
Availability (estimated costs)	Free Research Licence
Additional Software needed	none
Simulation Entities	Data packets, network nodes and links
Type of Simulation	micro (individual network nodes)
Link to V2X	• Can be used with VSimRTI to simulate the wireless network communication of vehicles, RSUs and traffic lights • Already used in combination with VSim- RTI • Extensions and improvements for vehicular ad-hoc networks developed by University of Ulm (http://vanet.info/jist-swans)
Documents and Papers	• JiST: An efficient approach to simulation using virtual machines. Software Practice & Experience, 35(6): 539-576, May 2005 • Density-Independent, Scalable Search in Ad hoc Networks. Proceedings of IEEE International Symposium on Personal Indoor and Mobile Radio Communications. September 2005. • User and Developer Documentation http://jist.ece.cornell.edu/docs.html • Extension by University of Ulm http://vanet.info/jist-swans
Typical and maximal size of APP	Maximal: not specified Typical: mid-sized cities
Typical and maximal time frame of APP	Maximal: not specified Minimal: 1 nano second
Typical Result	

Typical Runtime	Depends on number of network nodes, message count and system memory
Specific characteristics (V2X-related)	• Unrestricted in size/vehicle number • Portable
Other important information	Several extensions/improvements added by University of Ulm

Paper ID	**1**
Title	V2X-Based Traffic Congestion Recognition and Avoidance
Authors	Jan W. Wedel, Björn Schünemann, Ilja Radusch
Published conference/journal	10th International Symposium on Pervasive Systems, Algorithms and Networks, VCNA2009
Brief Description	Vehicular traffic congestion is a global phenomenon that has increased in importance in the last decades and has caused economically and ecologically negative effects. Thus, finding a way to improve traffic efficiency is a high frequented problem to be solved by scientists and politicians worldwide. One new promising approach is the usage of decentralized wireless vehicle to vehicle communication based on the V2X technology. The idea is that vehicles share information about the current local traffic situation and use this information to optimize their routes. In this paper, we introduce a new algorithm that can be used by navigation systems to calculate routes circumnavigating congested roads. For this purpose, each vehicle transmits its average speed of a road segment to vehicles in the neighbourhood. As a result, vehicles receiving this information can recalculate their routes based on the knowledge about the current possible speeds in the road segments of their neighbourhood. To evaluate the improvements that can be achieved by our algorithm, simulations have been done. Our results show that navigation systems using the V2X technology for a more intelligent route calculation can improve the traffic efficiency of future transport systems.
Target Application	Decreasing Travel Time by using V2X communication
Used Simulators	VSimRTI (coupling), JiST/SWANS (communication), SUMO (traffic)
Reference scenario	Cologne, inner city area
Key related work/citations	–
Brief reviewers's impression	The paper offers valuable information about a V2X application that acts as intelligent navigation system. Vehicles send the time they needed to pass through a road segment to other vehicles in their vicinity. This information is used as weighting factor for recalculating the vehicle's route. As a result, vehicles drive the fastest route depending on the current traffic situation.

Bibtex entry for fast citation	[21] – @article{wedel:i-span2009, author = {Jan W. Wedel and Bj\"{o}rn Sch\"{u}nemann and Ilja Radusch}, \\ title = {V2X-Based Traffic Congestion Recognition and Avoidance}, \\ journal ={Parallel Architectures, Algorithms, and Networks, International Symposium on}, \\ volume = {0}, \\ isbn = {978-0-7695-3908-9}, \\ year = {2009}, \\ pages = {637-641}, \\ doi = {http://doi.ieeecomputersociety.org/10.1109/I-SPAN.2009.71}, \\ publisher = {IEEE Computer Society}, \\ address = {Los Alamitos, CA, USA}, \\ }

Paper ID	**2**
Title	VANET Simulation Environment with Feedback Loop and its Application to Traffic Light Assistance
Authors	A. Wegener; H. Hellbrück; C. Wewetzer; A. Lübke
Published conference/journal	New Orleans, Louisiana, USA, 30 November - 04 December 2008; [including nine workshops]
Brief Description	Traffic applications, in which vehicles are equipped with a radio interface and communicate directly with each other and the road traffic infrastructure, are a promising field for adhoc network technology. Vehicular applications reach from entertainment to traffic information systems, including safety aspects where warning messages can inform drivers about dangerous situations in advance. As performance tests of the real system are very expensive and not comprehensive, today's evaluations are based on analysis and simulation via traffic simulators. In order to investigate the impact of traffic information systems there are two options: First, traffic simulators can be extended by application code and a simplified model for wireless communication. Second, existing network simulators can be coupled with existing traffic simulators. We favor the coupling of existing and well known simulators as we believe that the wireless communication characteristics influence the data transfer significantly and an oversimplified transmission model can lead to flawed results. In this paper we describe the feedback loop between traffic and network simulators named Traffic Control Interface (TraCI) and outline its versatility. We explain its use to determine possible energy consumption reduction when traffic lights send their phase schedules to vehicles.
Target Application	Evaluating the possible impact of traffic light assistance depending on the penetration rate using a simple scenario.
Used Models	The authors describe the test setup and the boundary conditions. Furthermore a simple fuel consumption model and vehicle behavior model is described.
Used Simulators	SUMO
Reference scenario	The scenario is provided by a simulated ring track with one traffic light. The total length of the ring is taken from the average distance between traffic lights in Germany.
Key related work/citations	–

Brief reviewers's impression	The paper addresses a very important topic at the beginning of the development phase of a traffic light assistance system. The possible impact of such a system is calculated using a simple approach. There SUMO is used in combination with the assistance system and fuel consumption model. Finally the impact to be expected is presented outlining the possible decreasements in fuel consumption.
Bibtex entry for fast citation	[33] – @misc{Wegener.2008, author = {Wegener, A. and Hellbr{\"u}ck, H. and Wewetzer, C. and L{\"u}bke, A.},\\ year = {2008},\\ title = {VANET Simulation Environment with Feedback Loop and its Application to Traffic Light Assistance: New Orleans, Louisiana, USA, 30 November - 04 December 2008 ; [including nine workshops]},\\ keywords = {cooperative traffic lights, reduction of fuel consumption, VANET, SUMO, simulator coupling, study}, address = {Piscataway, NJ},\\ publisher = {IEEE}, isbn = {978-1-4244-3061-1},\\ institution = {{Institute of Electrical and Electronics Engineers}},\\ originalyear = {30.12.2008} \\ }

Paper ID	**3**
Title	Unambiguous metrics for evaluation of traffic networks
Authors	Robbin J. Blokpoel, Daniel Krajzewicz and Ronald Nippold
Published conference/journal	International IEEE Conference on Intelligent Transportation Systems (ITSC), September 2010, Madeira Island (Portugal).
Brief Description	This paper presents an extensive set of unambiguous metrics that can be used for evaluation of new ITS applications. Currently in the literature most authors define their own metrics and small differences in definitions can lead to confusion when comparing the results. To derive the set of metrics presented in this paper, several steps have been taken. First, a list has been made with all metrics known by the research partners. Afterwards, a set of base measures has been defined. Using that set, clear formulas for all metrics have been derived and are reported in this paper. Finally, an application example about a cooperative traffic light controller is given.
Target Application	Traffic Efficiency - Improve traffic flows
Used Models	Performance Indicators
Used Simulators	iTETRIS & SUMO
Reference scenario	Urban Intersection and Traffic Light Control
Key related work/citations	–
Brief reviewers's impression	The paper offers valuable list of metrics for ITS studies. From a list of base metrics, the authors define and extract a large list of metrics in major classes of ITS applications (travel time, emission, LoS, intersection saturation). Finally, they applied their defined metrics in a basic intersection example and illustrate the importance of non-ambiguous metrics definitions. The defined metrics could be a guideline for performance indicators defined in the Simulation Handbook.

Bibtex entry for fast citation	[8] – @INPROCEEDINGS{blokpoel10, author={Blokpoel, R.J. and Krajzewicz, D. and Nippold, R.}, \\ booktitle={Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on}, \\ title={Unambiguous metrics for evaluation of traffic networks}, \\ year={2010},\\ month=sept.,\\ volume={}, \\ number={}, \\ pages={1277 -1282},\\ doi={10.1109/ITSC.2010.5625135},\\ }

Paper ID	4
Title	A Simulative Approach for the Identification of Possibilities and Impacts of V2XCommunication
Authors	Silja Assenmacher, Moritz Killat, Felix Schmidt- Eisenlohr and Peter Vortisch
Published conference/journal	15th World Congress on Intelligent Transport Systems
Brief Description	V2X allows a faster and more individualized information transmission to the driver and facilitates enhanced traffic and environmental data collection by radio-equipped vehicles. For a successful implementation of V2X-Applications the analysis of their potentials and impacts in advance is indispensable. In this paper we address computer simulation as an appropriate means to show the impact of V2X communications on vehicular traffic. We couple the vehicular traffic simulator VISSIM with a newly developed communication module VCOM and thus allow a simulative impact analysis for scenarios including thousands of vehicles. By means of a concluding simulation, the impact of different penetration rates of radio-equipped vehicles on the speed of the traffic flow is demonstrated in a highway scenario.
Target Application	Technical Demonstration, Decreasing Delay by using V2X communication
Used Models	Which models (according to the model chapter) have been used
Used Simulators	VCOM (communication), VISSIM (traffic)
Reference scenario	30km of motorway near Frankfurt, Germany
Key related work/citations	–
Brief reviewers's impression	–
Bibtex entry for fast citation	[34] – @conference{Assenmacher2008Simulative, author = {Assenmacher, S. and Killat, M. and Schmidt-Eisenlohr, F. and Vortisch, P.}, title = {A Simulative Approach for the Identification of Possibilities and Impacts of V2X-Communication}, booktitle = {15th World Congress on Intelligent Transport Systems}, year = {2008}, note = {Delivered but not published due to copyright}\\ }