SAFESPOT SPECIFICATION METHOD: AN EXAMPLE WITH INFRASTRUCTURE BASED APPLICATIONS

Fabien Bonnefoi
R&D Engineer
COFIROUTE
6 à 10 rue Troyon, 92316 Sèvres, FRANCE
+33 1 41 14 71 52, fabien.bonnefoi@cofiroute.fr

Fahim Belarbi
Head of Traffic Management Systems and R&D Department
COFIROUTE
6 à 10 rue Troyon, 92316 Sèvres, FRANCE
+33 1 41 14 71 05, fahim.belarbi@cofiroute.fr

ABSTRACT

SAFESPOT is an Integrated Project co-funded by the European Commission, under the strategic initiative "eSafety Cooperative Systems for Road Transport". The Goal of SAFESPOT is to understand how intelligent vehicles and intelligent roads can cooperate to produce a breakthrough in road safety. This paper presents the SAFESPOT specification method and environment through an example with infrastructure based applications. The method covers the needs of the European global approach around ITS development and harmonization, and introduces different steps to ensure consistency of the different diagrams and definition produced.

INTRODUCTION

The 6th European Framework Program promotes a huge portfolio of projects in order to achieve the challenging goal to halve the number of road accidents by 2010; one of these initiatives is the Integrated Project SAFESPOT (1). By combining data from vehicle-side and road-side sensors the SAFESPOT project will allow to extend the time in which an accident is forecasted, from the range of "milliseconds" up to "seconds". The transmission of warnings and advices to approaching vehicles, by means of vehicle-to-vehicle and vehicle-to-infrastructure communications, will extend in space and time the driver's awareness of the surrounding environment (1). SAFESPOT integrated project is divided into eight sub-projects, including fifty-one partners from thirteen European countries. The specification method presented in this paper was elaborated by the sub-project SCORE (SAFESPOT Core Architecture) which is composed of relevant partners from the other different sub-projects.
The first part of this paper presents constraints and specification method used SAFEPOT. The second part describes an example of the use of this specification method applied to one SAFEPSOT infrastructure based application of the sub-project COSSIB (Cooperative safety systems infrastructure based).

SPECIFICATION METHOD

Specification main requirements

The specification method used in SAFESPOT was elaborated by the sub-project SCORE (SAFESPOT Core Architecture) considering multiple complex aspects. The first requirement comes from the need of an harmonized European approach on ITS research projects, to ease the emergence of European standards in the different layers of a complete intelligent road transport system. As partners of the SAFESPOT project come from different fields and countries : research institutes, road operators, car manufacturer or automotive suppliers. The second requirement should ensure an efficient communication and cooperation between partners of the project to build consistent specifications which can be used on different European contexts.

A European Context

The SAFESPOT project is a part of an European approach around road safety and efficiency implying other European projects as shown in figure 1. All these projects, except the APROSYS project, imply cooperation between vehicles and the infrastructure by means of wireless communications.

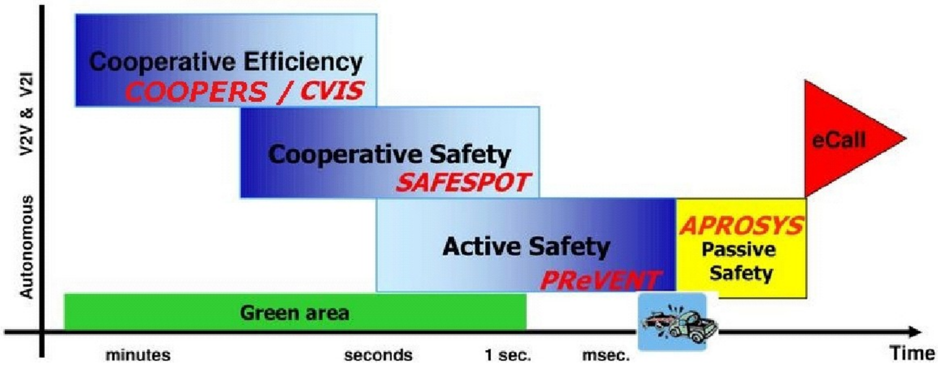

Figure 1 - Time-to-crash of the SAFESPOT approach

The CVIS and COOPERS projects mainly focus on increasing efficiency in the road network. They provide reliable information about road status, traffic jams, road works or accidents, local services like the presence of parking lots, and may suggest alternate roads to the drivers. They use long and medium range communication to significantly improve traffic control via effective and reliable transmission of information adapted to the current location of the vehicle.

SAFESPOT focuses on detecting potential dangerous situations and provides warnings to drivers in real-time. It is complementary to COOPERS and CVIS programs since it represents the last chance to keep the driver inside a safe area (in green colour on the figure 1), where he still has time to react to unpredicted events.

The PREVENT project aims to increase the active safety by controlling vehicle and activating for instance immediate braking of vehicles or avoiding dangerous manoeuvres of vehicles. Then, APROSYS and eCall systems are designed to mitigate the consequences of an accident.

Behind the conceptual complementarity of these projects, a concrete technological solution for the cooperation is based on the CALM architecture that intends to provide an unified interface for the different communication systems.

To ease the harmonization of these different projects and to push forward the emergence of European standards based on those projects, the European Commission has given two main recommendations on the specification method: the use of parts of the Frame methodology (2) and the use of a common diagram language, UML (3).

The different steps and building blocks of the specification method

Prerequisites: Before the specification phase was started, a detailed analysis of users needs based on road data accident was produced (4). Also, analyses of different execution environments, which correspond to different road types, lead us to define different system requirements depending on the execution environment (5). To conclude this preliminary step of the system analysis and definition, a detailed description of the foreseen applications including use cases and a list of requirement was produced (6). In parallel, expected impact evaluation was done to validate the described use cases.

The guide: The SAFESPOT project includes a high number of partners coming from different countries and companies, allocated into the different sub-projects of SAFESPOT. Therefore, the first step of the specification method was to define an ad-hoc specification guide, with concretes examples taken from the project, to help partners to understand and follow the specification recommendations. This guide describes the specification method with different steps of specification. The UML diagrams and descriptions required in the specification documents were also explained. The Figure 2 shows and example of recommendations described in the SAFESPOT specification guide for the specification of different system modules.

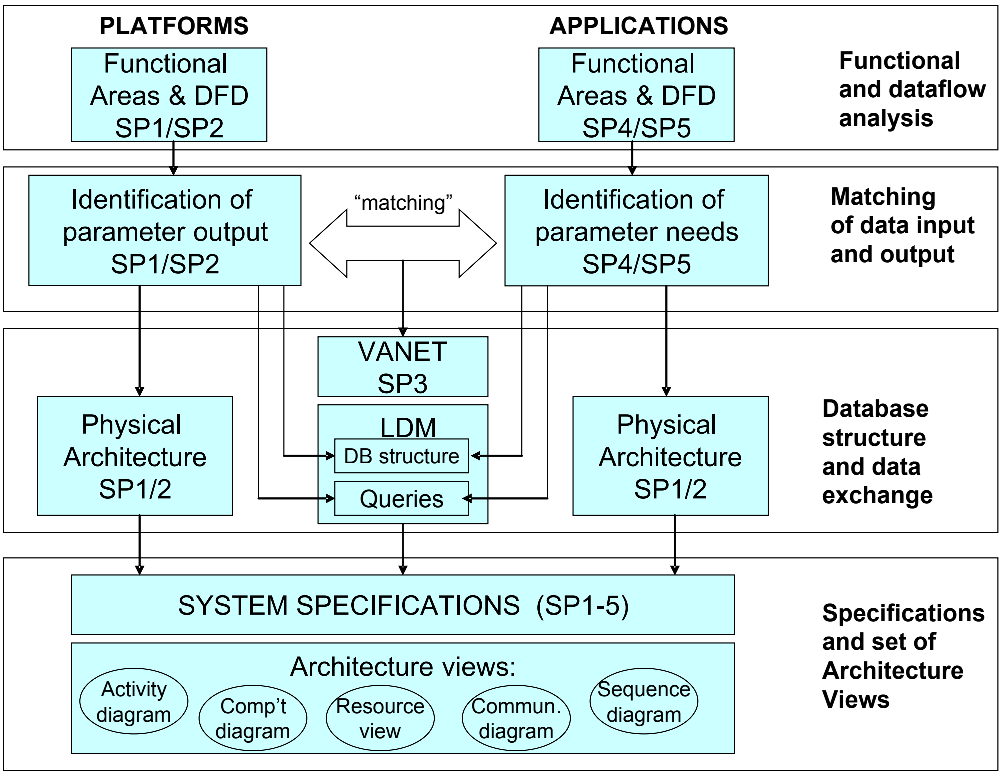
PLATFORMS
APPLICATIONS
Functional Areas & DFD SP1/SP2
Functional Areas & DFD SP4/SP5
Functional and dataflow analysis
Identification of parameter output SP1/SP2
"matching"
Identification of parameter needs SP4/SP5
Matching of data input and output
VANET SP3
LDM
DB structure
Queries
Physical Architecture SP1/2
Physical Architecture SP1/2
Database structure and data exchange
SYSTEM SPECIFICATIONS (SP1-5)
Architecture views:
Activity diagram
Comp't diagram
Resource view
Commun. diagram
Sequence diagram
Specifications and set of Architecture Views

Figure 2 - Workflow for deriving the System Specifications from the Functional Blocks

The high level architecture: Once the specification method was described in an example based document, description of the high level functional architecture started. The aim of this high level architecture is the identification of main modules, interfaces between modules, and the main data and control flows on SAFESPOT system. This task was lead by representative partners from different sub-projects, and has implied a very strong cooperation between sub-projects.

The data flow: Another important step of the specification was the definition of data types of the system. The objective of this task is to define a common document and verify that all necessary data types were defined. Contrary to the high level architecture, this description of data types has significantly evolved during the overall specification phase. The act of specifying modules of the system lead us to define new needed data types to enable efficient cooperation between modules.

The sub-system modules: Once all these tasks were achieved, the detailed specification of sub-system modules was ready to start. Each module was described following the dedicated guide and therefore introduced a specification including different required UML diagrams and informal description as described in the following parts.

UML diagrams

A set of minimal UML diagram was required for the specification of each module of the system. These diagrams were:

- A high level component diagram to explain the functional structure of the system and identify main interfaces as shown in figure (2)
- Interfaces diagrams that detail the access method and data exchanged between modules from the different sub project. These diagrams ensure the consistency of the specifications with the high level specification.
- Sequences diagrams were required to illustrate examples of execution of a modules and its interaction with other modules
- Activity diagram were required to specify the basic steps of important algorithms.

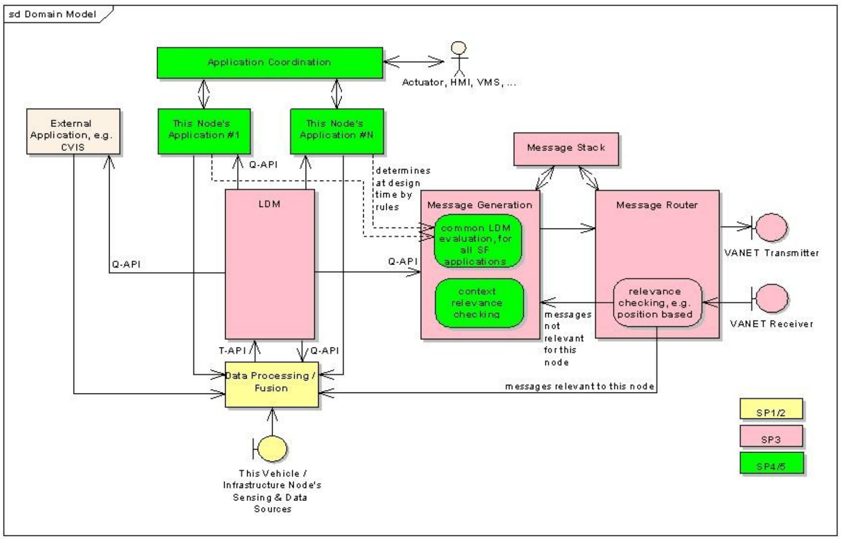
sd Domain Model
Application Coordination
Actuator, HMI, VMS, ...
External Application, e.g. CVIS
This Node's Application #1
This Node's Application #N
Message Stack
Q-API
determines at design time by rules
LDM
Message Generation
Message Router
common LDM evaluation, for all SF applications
VANET Transmitter
Q-API
Q-API
context relevance checking
relevance checking, e.g. position based
VANET Receiver
messages not relevant for this node
T-API
Q-API
Data Processing / Fusion
messages relevant to this node
SP1/2
SP3
SP4/5
This Vehicle / Infrastructure Node's Sensing & Data Sources

Figure 3 - SAFESPOT high level component diagram

- Finally, the use of class diagrams was proposed if needed. For example, the class diagram of the central database was a key diagram of its specification.

Harmonization layers

The interfaces: The different components of the system are connected through interfaces. Some of them were identified to be very strategic in regards to the global system. For example the access to the "Local Dynamic Map" (LDM) or the access to the "Vehicular Ad-hoc NETwork" (VANET) was shared among the majority of the sub modules of the system; therefore, their definitions are used by almost all specification sub-project of SAFESPOT. For this reason, special documents and an early harmonization planning were defined to validate a common definition of these two interfaces.

The data flow: Another important harmonization layer of SAFESPOT was built on the definition of the different data types and messages that are exchanged in the different data flows and interfaces. As explained above, this data types and messages definition has evolved along the specification phase. Two main constraints have driven this definition. The first one was the need to save bandwidth in the use of the wireless network between SAFESPOT entities that lead us to define the minimum necessary number of byte for each data objects. The second main constraint result of the precision of sources of data, which are mainly sensors and Traffic Information Centers, and the possibility to handle data fusion for different sources of the same information. Finally, the set of required data for the control flow was also defined.

Requirement matching

To ensure consistency of the specification in regards to the need and requirements on the system, a systematic verification of the requirement list was required. This steps leads to a strong cooperation between sub-projects as many requirements were impacting multiple sub-projects or components. After verification, some of the expressed requirements were not satisfied. Therefore it was necessary to revise some assumptions on which the specification was built. We present an example in the next section of this paper a mismatched requirement that impacted the definition of one SAFESPOT application

AN EXAMPLE WITH INFRASTRUCTURE BASED APPLICATION DEFINITION

After the definition of high level architecture of the system, the different sub-projects of SAFESPOT started the definition of the different modules under their responsibility. We briefly present here the application of the SAFESPOT specification methodology applied to an applicative module: the "Hazard and Incident Warning" application.

Integration of the application in the whole system

One of the first steps in specifying a module of SAFESPOT system is to synthesise the provided and required interfaces of a module. Using a common language like UML allows avoiding ambiguity and confirming correctness of the resulting diagram. The Figure 4 shows the first interface diagram of H&IW application. During the specification of the different module, the LDM module has strongly evolved providing more and more functionalities in its interface. This was impacted on the different interface diagrams of the project like the one presented Figure 4.

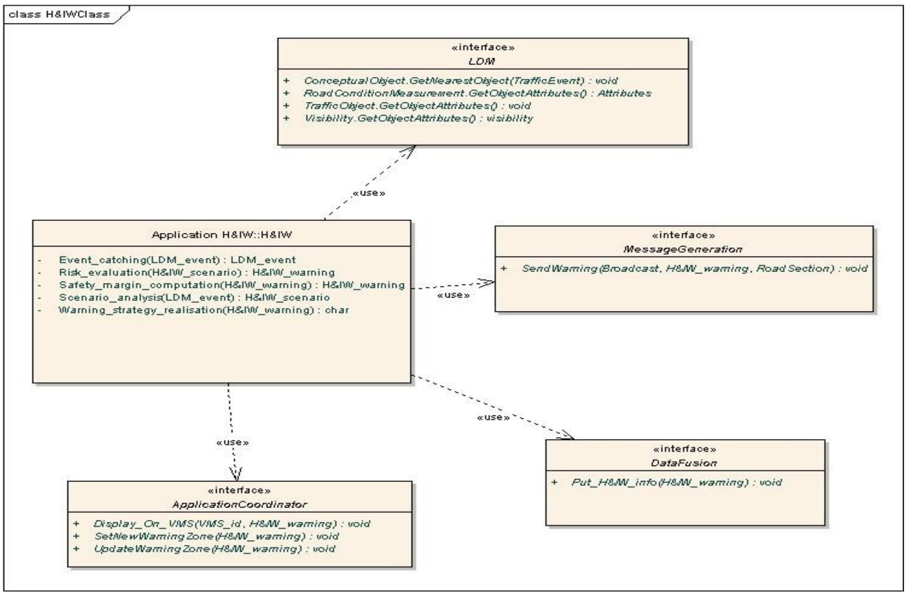
class H&IWClass
«interface»
LDM
+ ConceptualObject.GetNearestObject(TrafficEvent) : void
+ RoadConditionMeasurement.GetObjectAttributes() : Attributes
+ TrafficObject.GetObjectAttributes() : void
+ Visibility.GetObjectAttributes() : visibility
«use»
Application H&IW::H&IW
- Event_catching(LDM_event) : LDM_event
- Risk_evaluation(H&IW_scenario) : H&IW_warning
- Safety_margin_computation(H&IW_warning) : H&IW_warning
- Scenario_analysis(LDM_event) : H&IW_scenario
- Warning_strategy_realisation(H&IW_warning) : char
«interface»
MessageGeneration
+ SendWarning(Broadcast, H&IW_warning, RoadSection) : void
«use»
«use»
«use»
«interface»
ApplicationCoordinator
+ Display_On_VMS(VMS_id, H&IW_warning) : void
+ SetNewWarningZone(H&IW_warning) : void
+ UpdateWarningZone(H&IW_warning) : void
«interface»
DataFusion
+ Put_H&IW_info(H&IW_warning) : void

Figure 4 - H&IW first required interfaces

Another important aspect is to validate the different interactions between modules. For example, "Hazard and Incident Warning" application is interacting with "Speed Alert Application" through the "Application Coordinator" module. The sequence diagram of Figure 5 shows an example of interaction between those modules. It was made in common with partners responsible of the three modules.

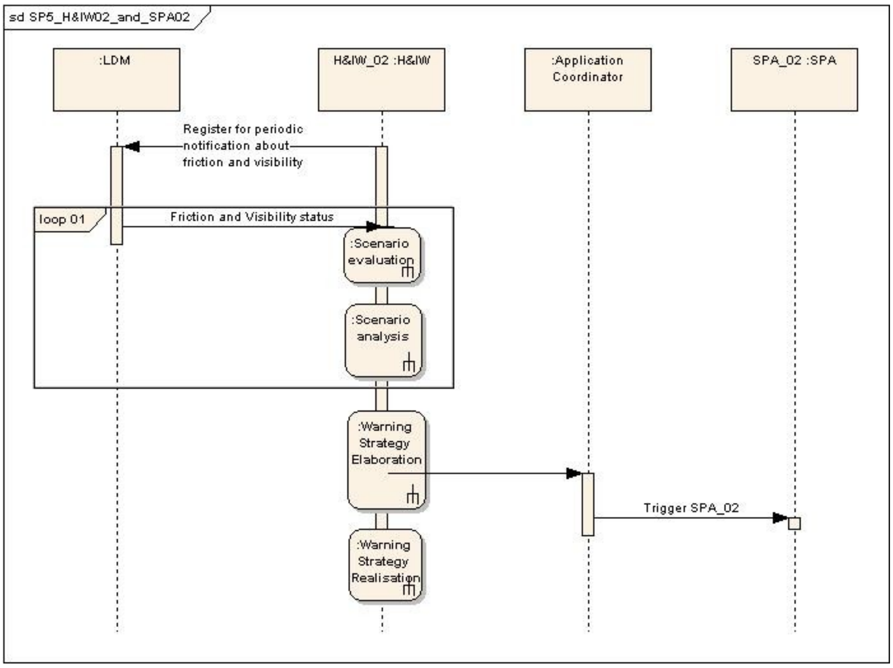
sd SP5_H&IW02_and_SPA02
:LDM
H&IW_02 :H&IW
:Application Coordinator
SPA_02 :SPA
Register for periodic notification about friction and visibility
loop 01
Friction and Visibility status
:Scenario evaluation
:Scenario analysis
:Warning Strategy Elaboration
Trigger SPA_02
:Warning Strategy Realisation

Figure 5 - Activity diagram showing an interaction between two applications

The Safety margin concept

The SAFESPOT project aims to design an alert system deployed both on vehicles and infrastructure. Different safety applications are defined to cover major dangerous road events. These applications analyse data, coming from the sensors and from exchanges through the

VANET communication network, and decide to trigger or not an adapted warning. The conditions which trigger a warning and the associated warning strategy define what is called the “Safety Margin”. Each application defines its own safety margin and all together, they define the SAFESPOT Safety Margin. An important aspect of the specification was to describe safety margins and check consistency of their definitions in regards to the whole system specification. Figure 6 shows an aspect of safety margin definition for “Hazard and Incident Warning” application in case of presence of an obstacle on road. This schema was associated in the specification document with description of the function calculating the size of different warning zone. Due to a mismatch of some requirements related to this safety margin and the network specification, it has been necessary to enlarge the different warning zone of the application. This also highlight that the FRAME methodology and the use of UML was not sufficient to clearly express some domain specific aspect of the system. Therefore the use of mathematic descriptions and informal descriptions like the Figure 6 was necessary to complete the specification.

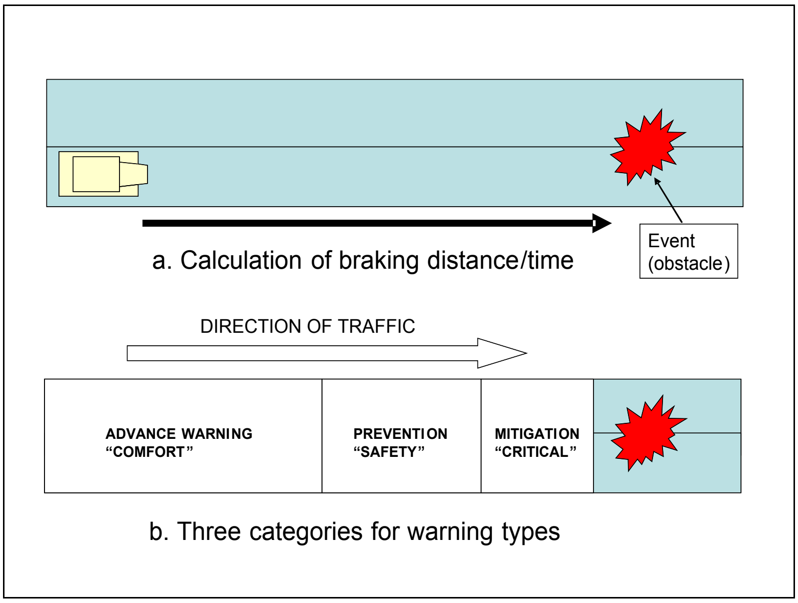

Figure 6 - Example of the "Safety Margin" of "Hazard and incident Warning" application

The application behaviour

Figure 3 shows the high level activity diagram of this application. It shows main interactions of the application with other components of the system, the high level data flow description and major steps of the generic algorithm. Other activity diagrams were defined to describe the content of the major steps of the application like “scenario analysis” etc.

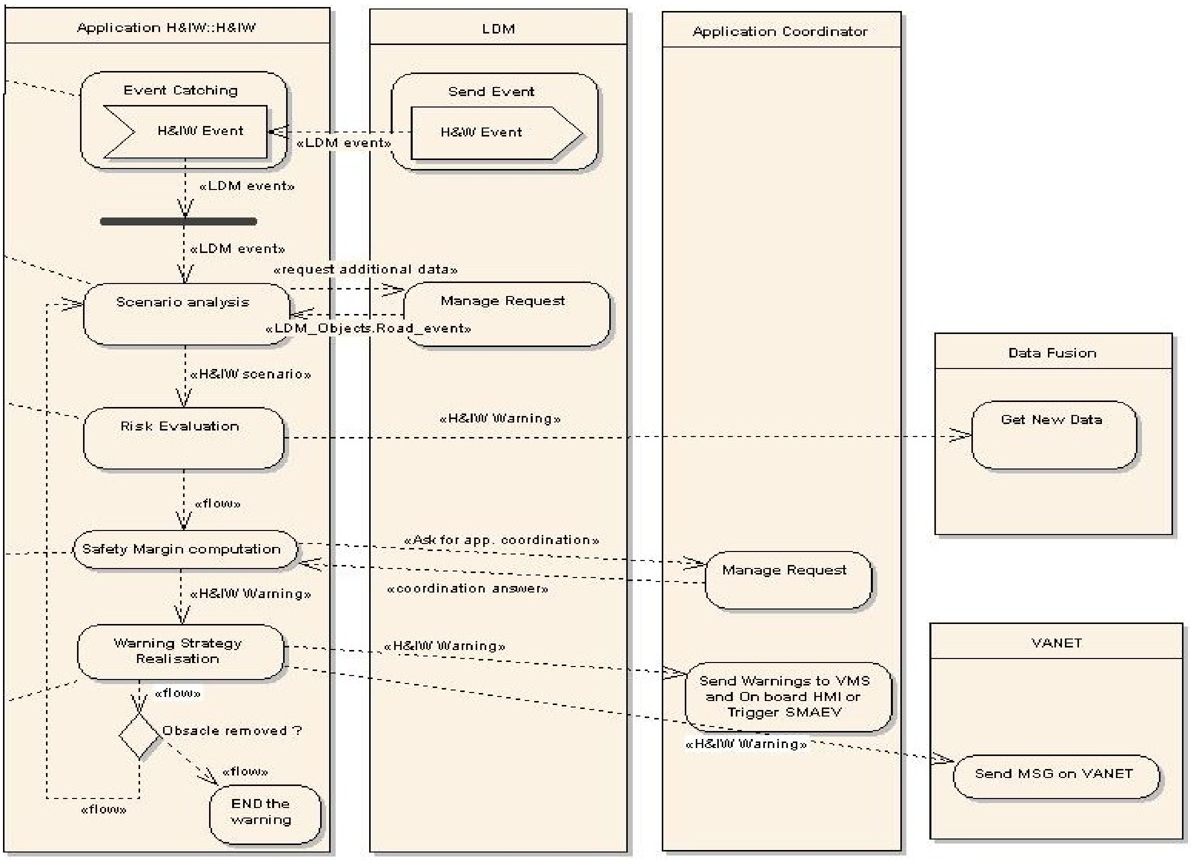

Figure 7 - Example of H&IW high level Activity diagram

The aim of this kind of diagram is to help defining the behaviour of a given application but also facilitate transition to the implementation phase. It helps developers to communicate with responsibles of the specifications.

Requirement matching

The table 1 shows an extract of the requirement list related to the "Hazard and Incident Warning" Application. The requirement "SP5_RQ_55_27_v1.0" on the bottom of the list shows for instance interaction between SP5 sub-project and SP3 sub-project which was responsible of the specification of the "VANET" components.

ID	Name	Requirement Definition	Application	Observation
SP5_RQ 04_36_v1 .0	Prediction of Trajectories	The system shall be able to predict the vehicles' trajectories.	H&IW_01 EXTENDED	Considered: This is foreseen for the dangerous overtaking sub-application (intention to overtake).
SP5_RQ 05_36_v1 .0	Identify Safety Critical Situations	The system shall be able to identify safety critical situations surrounding a critical point e.g. an urban intersection.	H&IW ALL	Considered: Identification of the safety-critical conditions is the role of the Scenario Analysis module in all H&IW sub-applications
SP5_RQ 52_36_v1 .0	Static Vehicle Data	The system shall receive static vehicle data like width, length, type of vehicle, mass.	H&IW ALL	Considered: This data (deriving from the VANET beaconing) will be obtained by querying the LDM.

SP5_RQ 55_27_v1 .0	Data from Vehicles	The system shall receive in the vicinity of a critical point in a motorway the position, speed and possibly acceleration with a frequency of 5/sec or shorter.	H&IW_01 H&IW_02b	Considered: This data is part of the information sent via VANET beaconing) and is therefore available at 1 second intervals.

Table 1 - Extract from the Hazard and incident warning application requirement list during its revision

This shows the identification of a requirement mismatch that impacted the specification of "Hazard and Incident Warning" application.

CONCLUSION

The specification phase in a large project like SAFESPOT is very important as well as very difficult due to the size of the foreseen system and the number of partners involved. Also, the need of an European harmonisation between road safety oriented projects requires the definition of clear and efficient specification methodology.

The use of the SAFESPOT specification methodology and its multiple re-iterations has lead to the identification and resolution of different problems. For example, mismatch of requirements, wrong interpretation of interfaces or lack of needed functions has been solved using this methodology. Also, the use of formalism like UML enables an efficient link with the next development phase of the project.

Anyway, the decision to use a specific language to ease the understanding of all partners is very important but introduces a learning phase for some of the partners not familiar with the formalism. This drawback has been strongly reduced by the creation of a clear and ad-hoc specification guide with concrete examples.

REFERENCES

(1) R. Brignolo, *"Co-operative Road Safety - The SAFESPOT Integrated Project"* in APSN - APROSYS Conference, May 2006.

(2) The Frame Forum, "*FRAME: European ITS Framework Architecture"*, 2004 [Online] http://www.frame-online.net/

(3) OMG, *"Unified Modeling Language: Superstructure – Version 2.0 formal/05-07-04"*, OMG, March 2006. [Online]. Available: http://www.uml.org/

(4) F. Bonnefoi, F. Bellotti and T. Schendzielorz.*From User Needs to Applications: The Safespot Approach Based on Road Accident Data Analysis.* Proceedings of the 6th European Congress and Exhibition on Intelligent Transport Systems and Services, June 2007.

(5) F. Bonnefoi, F. Bellotti, T. Schendzielorz, F. Visintainer,*Infrastructure-Based Co-operative Architectures: How Safespot Deals with Different Road Network Areas.*, ITS World Conference, 9-13 October 2007, Bejing.

(6) F. Bonnefoi, F. Bellotti, T. Schendzielorz, F. Visintainer,*SAFESPOT Applications for Infrastructure Based Co-operative Road Safety*, ITS World Conference, 9-13 October 2007, Bejing.