Embedding of real OSGi based V2X applications in a simulation environment

Stefan Reichel
University of Potsdam
Hasso-Plattner-Institute for IT Systems Engineering

Proposal for the master thesis

I. Introduction

In the last years many approaches for car to car communication and its simulation were developed. Complex simulator couplings were designed, which involved traffic, network, environment and application simulators. They all have one aim, create the most realistic simulation data. Therefore the same algorithms and protocols are used, but still differences between the real world and simulation exist. One of them are the used V2X applications, which are core components of every use case. These little programs decide how received messages are interpreted and how a car should react on them. Nevertheless every application simulator needs its own adaption of such a program, which makes the simulation results less comparable and less realistic. In the last year, first field tests were planned within the German SimTD project. Within this project also a special OSGi based architecture for V2X applications was designed.

The objective of the master thesis is to integrate this approach into the VSimRTI simulation coupling and in effect to make this coupling compatible to the developed and in cars deployed SimTD V2X applications. It would be possible to compare simulation results with the identical real life setup.

II. Architecture of the application simulator of VSimRTI

An advanced simulator couplings is VSimRTI[1], which also integrates an own application simulator. The so called App-Sim provides several layers and interfaces to control the vehicle behavior, send and receive messages and obtain status information. In that way the application simulator can exacly control the access rights of every application. Each of those applications is executed in same process but in different threads. This is done to improve the performance and the memory foot print in comparison to a one process per unit approach. In that way App-Sim is able to simulate even large amounts of vehicles and road side units at the same time.

III. Architecture defined by SimTD

The SimTD project created a first application and application environment architecture based on OSGi. The Java based OSGi realizes a component model which is based on a service provider and consumer principle. In the SimTD architecture the car provides several services e.g. to send and receive messages, obtain vehicle information or to control the car behavior. Each of those services has to be registered within the OSGi framework in advance. Afterwards service consumers can access their methods and e.g. retrieve the current driving speed. In SimTD this is done via the VAPI-Client service, which can provide information on demand but also based on subscription. In that way it is e.g. possible for the consumer to be informed automatically when the vehicle speed drops below a defined level. This high flexibility can also be found in the design of the message receiving service, which allows the creation of filters.

IV. Challenges and upcoming tasks

The integration of OSGi[2] in an application simulator can not be done by just integrating an OSGi framework for each on board unit. This would create an own virtual machine process for every simulated vehicle. In effect the performance and especially the memory footprint would suffer dramatically. Therefore a solution must be developed which only needs one JVM to provide the OSGi framework and its basic services. Nevertheless OSGi was not developed for such a use case and in result some challenges arise. When only one OSGi framework is used, VAPI-Client services must be registered for every simulated vehicle. Nonetheless OSGi only allows one service with the same name and version to exist. In effect the framework can only be integrated by changing the OSGi framework implementation itself. At the same time the applications, in SimTD called functions, should not be altered at any point. These demands require the development of a complete new OSGi concept, which also has to satisfy the simulation performance criteria .

In addition to that several services have to be implemented from scratch to provide the exact footprint of the SimTD services. The information provided by the VAPI-Client service will raise another last challenge, simply because VSimRTI lacks most of them.

Within this work concepts for the integration of an OSGi based architecture are shown and evaluated in the context of a state of the art simulation coupling, VSimRTI. In addition to that performance analysis are executed to investigate the impact on the total simulation speed.

A. The following tasks have to be fulfilled

1) A concept to support multiple services of the same kind(same name and version) within OSGi has to be created and evaluated. The special simulation performance criteria have to be considered.
2) This concept has to be implemented in an open source OSGi implementation e.g. Knopflerfish. In effect multiple instances of the same service can be provided and consumed by so called ”bundles” e.g. SimTD applications.
3) Analysis of the SimTD specifications and guidelines to create an overview about the available services and their provided interfaces.
4) Creation of services for example the VAPI-Client, which have same foot print as SimTD services. Only if the new services provide the same foot print as their SimTD counterparts, the OSGi applications can be embedded without any changes. After this step the SimTD ”functions” should accept their new environment by consuming the new services without any errors.
5) Design of own application simulator service to feed VAPI-Client and communication services. This will create a backend for the VAPI-Client and communication services to obtain information, send and receive messages and control the vehicle behavior.
6) Creation of coupling between App-Sim and created service. In this step the OSGi framework will be integrated into the existing application simulator. Afterwards the OSGi integration is in general completed and first performance evaluations can be done.
7) Increase of available vehicle information in VSimRTI. Only by providing needed additional information the SimTD ”functions” can work properly.

V. Thesis

A concept to embed OSGi based applications in a simulation environment can be created and implemented. Those applications can be simulated with a feasable performance. In effect real SimTD ”functions” can be simulated and the aggregated data compared to field test results.

References

[1] T. Queck, *Runtime Infrastructure for Simulating Vehicle-2-X Communication Scenarios* Hasso Plattner Institute for IT Systems Engineering, 2008

[2] OSGi-Allicance, *OSGi Alliance Specification 4.2* http://www.osgi.org/Specifications/HomePage?section=2