

Co-simulation of vehicles and crowds for rescue trials

Yun-Pang Flötteröd^{1*}, Michael Behrisch¹, Martijn Hendriks², Jean-Benoît Bonne³, Erik Vullings⁴, Rinze Bruining⁴

¹ German Aerospace Center (DLR), Institute of Transportation Systems, Germany

² XVR Simulation, The Netherlands

³ Thales SIX GTS France, ThereSIS, France

⁴ The Netherlands Organisation for applied scientific research (TNO), The Netherlands

*yun-pang.floetteroed@dlr.de

Abstract

In this paper, the focus is put on the integration of XVR, SE-Star and SUMO simulators via the Driver+ test-bed, where XVR provides different learning environments for all levels of incident command, SE-Star handles crowd simulation and SUMO focuses on vehicular simulation and routing. With the test-bed and the provided services these simulation tools can synchronically exchange information with each other, creating a common simulation space that offers more possibilities for CM-training, trials and tests. A simulation scenario around the train station in Rotterdam, the Netherlands, is established for demonstration of the connected systems.

1 Introduction

Crisis management (CM) is getting more and more attention worldwide. Different tools and solutions have been developed, but the interoperability between these applications is still project-specific and needs to be enhanced in order to handle crisis situations with various aspects (e.g. rescue procedure, vehicle routing, crowd evacuation, flood prediction) in a complete manner. To facilitate efficient and effective CM-training, trials and tests the EU-funded project Driver+ has developed an open source cloud-based test-bed, based on the distributed messaging platform Apache Kafka (Driver+, Driver+ project, 2019). Different solutions and simulators were successfully connected to this test-bed in order to meet the needs of proposed training and experiments.

In addition to develop a pan-European test-bed for crisis management Driver+ also aims at capability development, emerging a portfolio of solutions and building a shared understanding in crisis management across Europe. To carry out these objectives, four trials have been planned and executed in Poland, France, Austria and Netherlands respectively. Different solutions and simulators are tested according to the corresponding trial goals. Moreover, the annual Innovation for Crisis Management (I4CM) event takes place for knowledge/experience dissemination and sharing and addressing different crisis management issues.

This paper aims to integrate three traffic/visualization-related simulators for establishing a common simulation environment, so that different issues of crisis incidents, e.g. crowd evacuation issue and vehicle routing issue, can be taken into account for trials and exercises. The simulated travel durations of proposed route alternatives, the simulated accessibility information of

selected/affected areas, the simulated evacuation times and so on can also be used to support the decision-making process of crisis management. In Section 2, the Driver+ test-bed, the concept to couple the simulators are firstly introduced. The adopted simulators and their main functions related to the test-bed will then be explained. After that, the applied scenario for testing/demonstrating the coupling work is described in Section 3. The respective action plan is clarified and illustrated afterwards. In the end, the conclusion is made and the planned work in the next phase is presented.

2 Test-bed and simulators

To simulate a pre-defined crisis situation with different aspects, the related simulators communicate and synchronize with each other via the Driver+ test-bed based on the Kafka messaging system. In the following, Kafka's framework and its current development status and functions will be introduced. After that, the concept to couple the three simulators, XVR, SE-Star and SUMO, is explained. The main characters and applied functions of the simulators and the ways to connect them to Kafka are explained accordingly.

2.1 Driver+ Test-bed

The Driver+ test-bed is the essential technical infrastructure for all trials and exercises in the project. The objectives of the Driver+ test-bed are not only to exchange information, but also to test solutions with given fictitious incidents, created by simulators. The former one can be carried out with the Common Information Space (CIS) of the test-bed, while the latter one can be achieved with the Common Simulation Space (CSS), as illustrated in the test-bed framework in Figure 1. Multiple simulators which can simulate an aspect of the crisis can connect to the CSS part of the test-bed. They can operate independently or interoperate with other simulators, where the necessary couplings need to be established. There is a gateway to connect the CIS and CSS. With this gateway, the created fictitious incidents can be provided to the CIS and solutions can request changes to the simulation (Vullings, van Campen, Hameete, & Hendriks, 2019). The test-bed also provides additional tools, such as a time service for keeping track of the simulation time, and a scenario manager to prepare and run a scenario with events (see Figure 1). These tools, including the test-bed, are available as Docker images and can be easily used and combined in a Docker environment.

With regards to the flexibility of open source software and the available adapters to connect to the test-bed, the development of the CSS and CIS are based on the open source streaming platform, Apache Kafka. It is set up using Docker, such that it can run both in cloud and locally. The Avro schema format is used to define the respective data schemas. Currently, connectors in Java, C#, TypeScript and Python are available to connect to the Driver+ test-bed from those programming languages. Additionally, a REST adapter is available. Using these adapters, the Driver+ test-bed facilitates users to easily add additional modelling and simulation functionality.

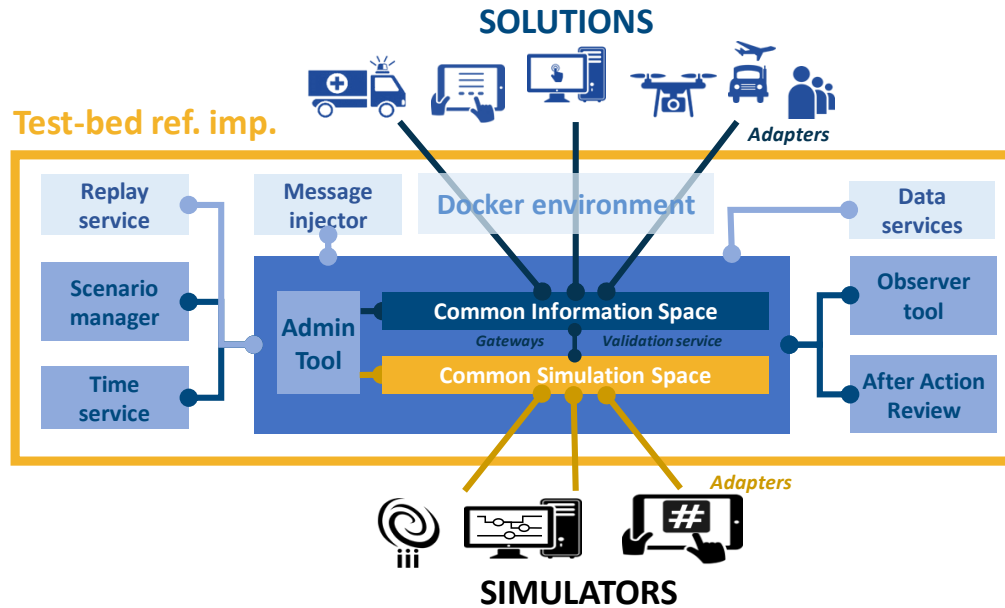


Figure 1 Overview of the framework of the test-bed (Vullings, van Campen, Hameete, & Hendriks, 2019).

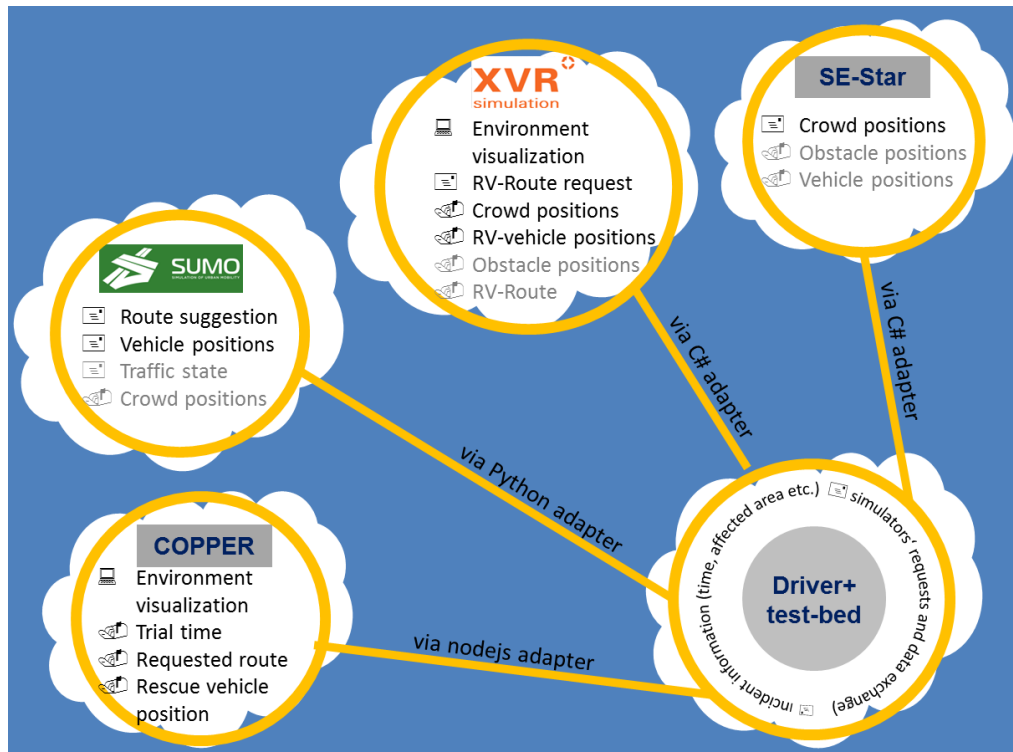
2.2 Coupling concept

The main goal of the current coupling work is to carry out a common simulation environment with consideration of vehicles, pedestrians, and real-time vehicle dispatching, routing and positioning. This simulation environment can then be interactively used by trainees for crisis trials and exercises.

The whole coupling work is based on the Driver+ test-bed and illustrated in Figure 2. All simulators are synchronized with the time service provided by the test-bed. There is a trial manager tool in the test-bed for directing the connected tools and simulators on relevant phases inside the training or trial session. XVR (OS and RM) is used as visualization and training platform. XVR OS continuously updates crowd movements provided by SE-Star and interacts with trainees according to the pre-defined training procedure, while XVR RM sends requests for vehicle dispatching, routing or/and positioning. Vehicular positions (and routes) will be indicated in the XVR RM portal.

On the traffic simulation side, SUMO executes firstly the requests from the trial manager to start the simulation and then block roads upon request. During the simulation, SUMO sends vehicles into the simulation with the start and end positions, given by XVR. SUMO will also search the fastest route for certain vehicles and send back the respective route information, vehicle positions and traffic state according to the requests from XVR and the trial manager. On the crowd simulation side, SE-Star receives real-time obstacle positions from XVR and vehicle positions from SUMO for simulating crowd movements, and delivers crowd positions to XVR OS.

The main coupling work is finished. Some of the above-mentioned functions, indicated in grey in Figure 2, will be implemented in the next phase.



*: RV: rescue vehicle; **: functions in grey will be implemented in the next phase.

Figure 2 The concept to couple XVR, SUMO and SE-Star.

2.3 SE-Star

Since 2008, inside Data Science & Artificial Intelligence Labs of THALES, a Crowd Simulation engine, called SE-Star (Navarro, Flacher, & Meyer, 2015), has been designed. It is a life simulator managing: internal variables, motivations & emotions, stimuli, personality and behaviors. SE-Star uses a Biomimetic control architecture based on navigation (Hippocampus), action selection (Basal ganglia) and planning (Frontal cortex). Use-cases are: design of critical infrastructures and business processes, supervision (video & crowd) operators' training, data generation for deep neural networks training (video & dataset generation), decision support, what-if scenario exploration and collaborative systems testbed. SE-Star can provide information from high level (density, alarms and situation reports) to low level (individual's information, devices' status). It can simulate between 5000 and 10000 peoples on one computer in real time and can work in distributed mode pushing limits (tested with 80000 peoples on 10 Intel™ NUC computers). SE-Star has been used in the EU projects (Opti-Alert, SECURE-Ed, iCore) and in THALES projects covering airports (Pisa), train stations (Gare du Nord) or crowded places (Mecca). Easily extendable, it has been connected to real systems (crowd monitoring, airport supervision) using standard interfaces and protocols (SOAP, REST, RTSP).

In the Driver+ project, the C# test bed adapter has been selected and extended to support SE-Star network messages. Those network messages allow controlling the simulation, sharing Entities and Objects information and controlling them.

- Control the simulation: From the test bed time service, it's possible to pause, play and change speed of the SE-Star simulation.

- **Share Entities and Objects:** SE-Star creates entities and objects relevant for the crowd simulation. Objects can be entry points for building (shops, restaurants, main entrances ...) or security equipment like speakers for evacuation. SE-Star gets objects and entities that can impact the crowd simulation. It can be vehicles from XVR (ambulances) or SUMO (road traffic and sign) and entities from XVR like a firefighter.

2.4 XVR

With the XVR Simulation platform (XVR, 2019) users can teach, train, exercise, assess and evaluate; for single and multi-agency purposes, on the bronze, silver and gold level within the Crisis Management field.

XVR On Scene (XVR OS) is designed as a 3D supporting tool, giving instructors maximum flexibility to create virtual scenarios according to a specific training objective. XVR OS can be used with a great variety of training set ups: from team training and exercising, to classical classroom teaching as well as individual training and assessment. It helps trainees understand concepts by experience, let's them acquire practical knowledge in a relatively short timeframe, practice decision making skills and experience situations that rarely occur in real life.

XVR Resource Management (XVR RM) is a training & education tool that allows users to train and assess incident management and response in any geo-specific environment. The map-based environment features realistic characteristics and constraints of a specific region. The essence of exercising with XVR RM is to see how course participants, in a given scenario, set-up the command structure together and deal with the efficient deployment of scarce resources. Staff, equipment and vehicles can only be deployed at a single location, forcing participants to deal with the dilemma's that arise thereof.

Both XVR OS and XVR RM are based on the Unity3D game engine and use the C# adapter to connect to the Driver+ test-bed for the following interactions with the other simulators:

- XVR RM enables the user to transport a given unit (ambulance, police, fire) to a given location. The simulator will send out a unit transport request to SUMO. XVR RM will update the vehicle position of this unit based on the update messages send from SUMO.
- XVR OS will receive update messages from SE-Star regarding the positions of simulated pedestrians in SE-Star. The simulator will visualize these pedestrians in the 3D environment, based on the information provided by SE-Star.

Some application studies can be found in (Lamb, Booseman, & Davies, 2015), (Barta, 2017) and (Meresse, Turpin, Boosman, & van Campen, 2013).

2.5 SUMO-Connector

SUMO is a microscopic traffic simulation mainly developed by DLR (Lopez, et al., 2018). In addition to motorized and electric vehicles other means of transport (such as railway, tram, bicycle and moped) and pedestrians can be simulated as well. To be able to interact with other simulators during a simulation the SUMO component TraCI (Traffic Control Interface) is applied. TraCI can give access to a running road traffic simulation and allows retrieving values of simulated objects, to manipulate their behaviors on-line and to change the usage status of the infrastructure, e.g. road/lane use and traffic light control (SUMO, 2019). Currently, most of TraCI functions are implemented in Python. Therefore, the python test-bed adapter is adopted to connect SUMO and the test bed Kafka.

According to the defined coupling concept in 2.2 the required messages are divided into three categories.

- Configuration

In this category, the start and end times of a simulation and the simulation configure file name are defined. Moreover, the data aggregation intervals for the traffic in the whole network, each single vehicle and the traffic in the affected area are defined respectively, where the latter two intervals are optional items. The corresponding data will be sent back to the test-bed at the pre-defined intervals.

- Affected areas

Each affected area is defined with ID, its polygon information as well as the begin and end times. Furthermore, the status of broken traffic lights in the area, i.e. true or false, and the restricted vehicle types also need to be defined. All roads in the affected areas will be closed for the defined restricted vehicle types according to the given begin and end times.

- Request

It is mainly to handle three requests: (1) to insert rescue vehicles in the simulation with the given start and end geo-coordinates; (2) to find the corresponding fastest routes and send them back to the test-bed, and (3) to send the respective vehicle positions at each time stamp, which is configurable, back to the test-bed.

The developed SUMO-Connector for the test-bed and some scenario examples are freely available (Driver+, sumo-connector, 2019).

2.6 COPPER Common Operational Picture-Tool

In order to get an overview of the simulation, one can use the COPPER COP-tool (Driver+, copper, 2019). It connects to the testbed and listens to the messages that are being published by the other tools. Subsequently, the information in these messages is displayed in multiple formats inside the tool. Mainly, geographical information is displayed on a map. All other information, such as the simulation time, is displayed as text. The simulation time is always visible as a digital clock, whereas other text updates are stored in a log list.

COPPER is a client – server dashboarding application built with TypeScript. The server side of the application connects to the testbed in order to receive messages published by the other tools. It then processes the messages and distributes it to all active clients (which could be multiple instances).

At the moment, COPPER processes and displays the following information:

- Trial time – The simulation time as published on the Driver+ testbed.
- Route request – If a route request is made, COPPER will read this request and display it on the geographical map. Also, a log entry is made that textually represents the message.
- Rescue vehicle position – If a rescue vehicle is driving towards an incident location, its actual and previous positions will be displayed on the map. The location updates are also stored in the log.

Except the commercial software XVR and SE-STAR all the related codes, schemas and tools are committed in the github platform and can be found in (Driver+, Driver-EU, 2019). The ways to set up the test-bed and to connect to SUMO are also described there to facilitate users to set up their own co-simulation.

3 Scenario

3.1 Description

To test the proposed coupling work a synthetic scenario around the main railway station in Rotterdam, the Netherlands has been established. In this scenario, a vehicle incident has occurred at the main railway station. No trains are allowed to run. The surrounding roads need to be closed and the people inside and around the railway station need to be evacuated. An overview about the scenario is illustrated in Figure 3.

Furthermore, the crisis manager needs to dispatch the rescue team, police and medical staffs from different units to different locations at different times according to the rescue progress.

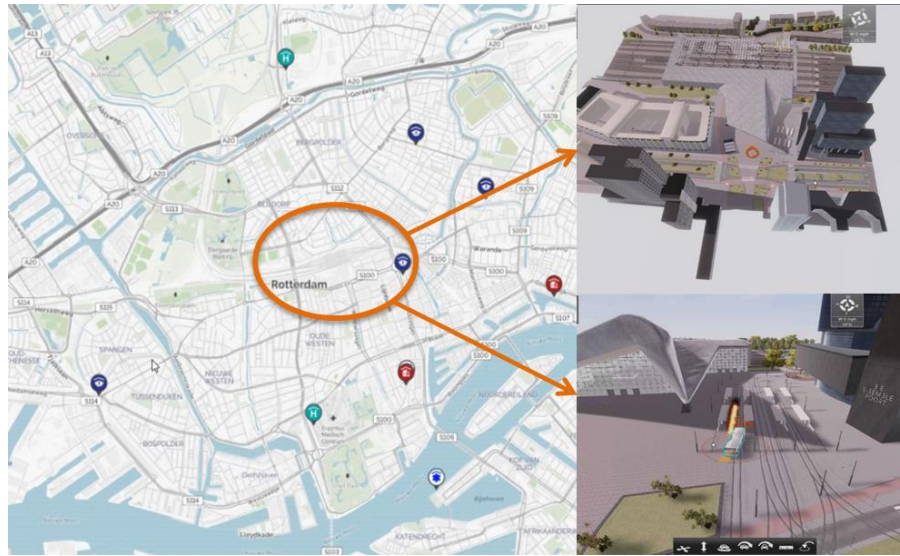


Figure 3 Overview of the synthetic scenario at the main railway station in Rotterdam.

3.2 Simulation setup

OpenStreetMap (OSM) is used as the basic map for all simulators. To ensure the position accuracy all simulators use geo-coordinates information for communication instead of the IDs of OSM. Moreover, the 3D map at the main railway station is also applied for crowd simulation and trial visualization. Regarding traffic demand, synthetic traffic, including passenger cars, buses, trucks, trams, bicycles and pedestrians, is generated. Figure 4 shows an overview about the simulated crowd movements with SE-Star, while Figure 5 shows the network for traffic simulation and the road closure area, indicated in yellow. During the road closure period only authority vehicles can run in the closed area. There is no change in traffic light control. Together with the crowd movements, vehicle positions, which are sent upon request, will be sent to the test-bed. XVR will then receive these data and visualize them in the trial simulation environment (XVR OS) and the traffic information portal (XVR RM) accordingly. Figure 6 illustrates the trial simulation environment with XVR OS.

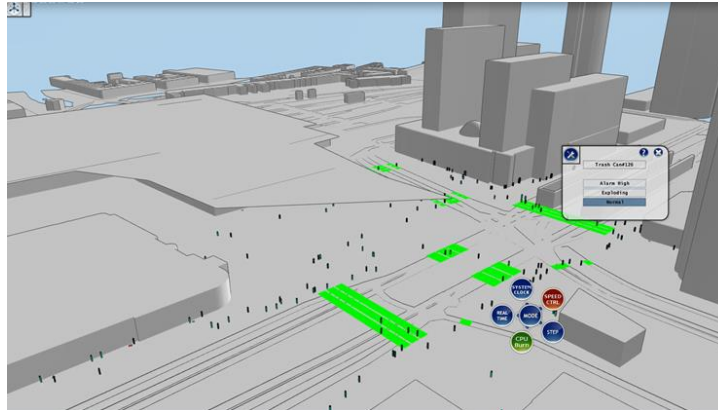


Figure 4 Simulated crowd movements in the Scenario at the main railway station in Rotterdam.



* All the roads in the yellow area are closed due to the vehicle incident.

Figure 5 Overview of the microscopic traffic simulation network.

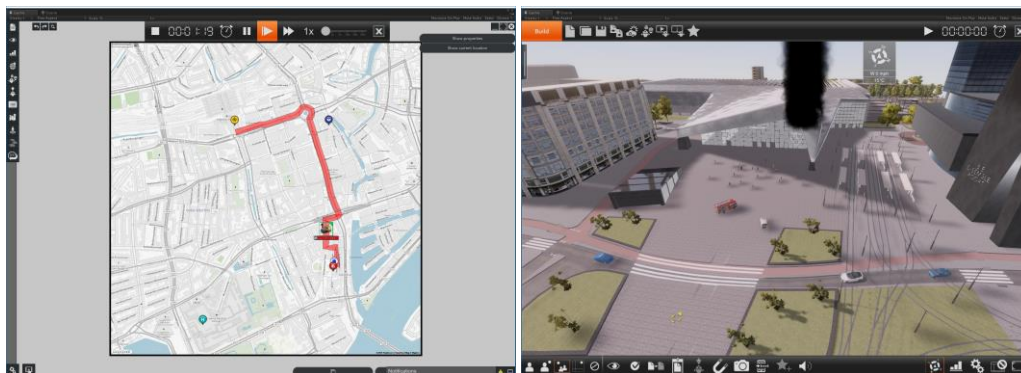


Figure 6 Illustration of the trial simulation environment with use of XVR RM (left) and XVR OS (right).

3.3 Action plan

According to the scenario description in 3.1 the corresponding action plan is developed for testing the coupling work. This plan covers from test-bed connecting, simulators starting, request sending, data sending to simulation ending. Figure 7 shows the sequence of all actions and the messages needed to be exchanged in the scenario. According to this action plan, the coupling test will be manually executed and demonstrated at the SUMO User Conference.

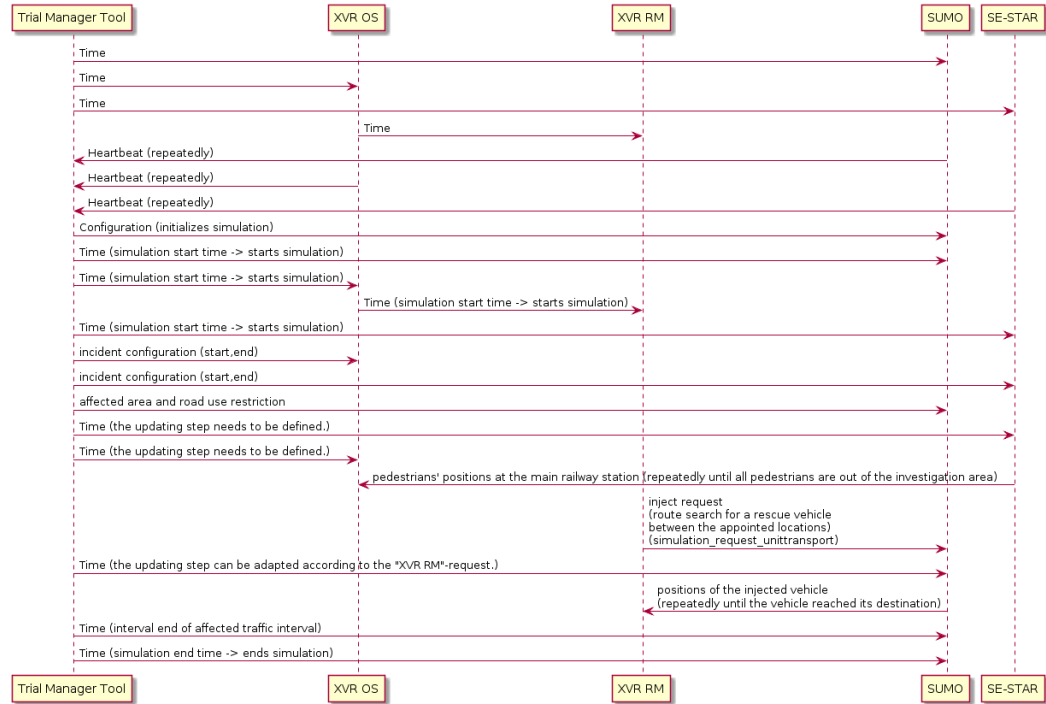


Figure 7 Sequence diagram of the proposed action plan

4 Conclusion and future work

The current main coupling work focuses on the basic communication and data exchange between XVR, SUMO and SE-Star and the visualization environment for trials and exercises. The first results show that the positions of simulated vehicles/routes/pedestrians are with high accuracy. The coupled simulation can be conducted in real-time in the proposed scenario with synthetic traffic demand. If a whole metropolitan needs to be simulated, the timing issue needs to be considered. In the next phase, the required messages for the coupled simulation will be fully integrated into Driver+'s trial manager tool, so that users can use these three simulators for setting up a common simulation directly via the trial manager tool. Moreover, the planned functions, indicated in grey in Figure 2 will also be implemented to make a common simulation environment more complete. More functions either for gathering specific simulation information or for interfering with the simulated infrastructure or road users can also be developed according to the users' needs.

5 Acknowledgements

The authors gratefully acknowledge the funding of the Driver+ project received from the European Union's 7th Framework Programme for research, technological development and demonstration under Grant Agreement no. 607798.

References

- Barta, J. (2017). Comparison of Simulators Used for Education and Practical Training of the Critical Infrastructure Staff. *E-learning (vol 9): Effective Development of Teachers' Skills in the Area of ICT and E-learning*, pp. 279-293.
- Driver+. (2019). *copper*. Retrieved from <https://github.com/DRIVER-EU/copper>
- Driver+. (2019). *Driver+ project*. Retrieved from <https://www.driver-project.eu/driver-project/>
- Driver+. (2019). *Driver-EU*. Retrieved from <https://github.com/DRIVER-EU>
- Driver+. (2019). *sumo-connector*. Retrieved from <https://github.com/DRIVER-EU/sumo-connector>
- Lamb, K., Booseman, M., & Davies, J. (2015). Introspect Model: Competency Assessment in the Virtual World. *Proceedings of the ISCRAM 2015*. Kristiansand.
- Lopez, P. A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.-P., Hilbrich, R., . . . Wießner, E. (2018). Microscopic Traffic Simulation using SUMO. *21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2575–2582.
- Meresse, P., Turpin, E., Boosman, M., & van Campen, S. (2013). The International Wildfire Simulation Training Project: 3D serious game-based training and exercising of wildfire response professionals. *UK Wildfire Conference 2013*. Vale of Glamorgan.
- Navarro, L., Flacher, F., & Meyer, C. (2015). SE-Star: A Large-Scale Human Behavior Simulation for Planning, Decision-Making and Training. *Proceedings of the AAMAS 2015*. Istanbul.
- SUMO. (2019). *TraCI*. Retrieved from <https://sumo.dlr.de/wiki/TraCI>

Vullings, E., van Campen, S., Hameete, P., & Hendriks, M. (2019). Cloud-based M&S for Trails and Exercises. *30th International Forum for the Military and Civil Simulation, Training and Education Community (ITEC)*. Stockholm.

XVR. (2019). *XVR simulation*. Retrieved from <https://www.xvrsim.com/en/>