

Technological challenges for Autonomous driving systems validation in heterogeneous environments



OPAL-RT
TECHNOLOGIES



SUMMARY

Context and issues

From Simulation to Real hardware testing

- Simulations : why ? how? To what extent ?
- Hardware and software based co-simulation.

Opal-RT added value.

- Multiple hardware platforms support,
- RT-LAB / Orchestra : A common and unique software that handles all your testing phases and requests,
- Hosting the simulation environment and Opal-RT software and hardware solution on the same physical machine.

What is to come



CONTEXT AND ISSUES (1/2)

Vehicles complexity

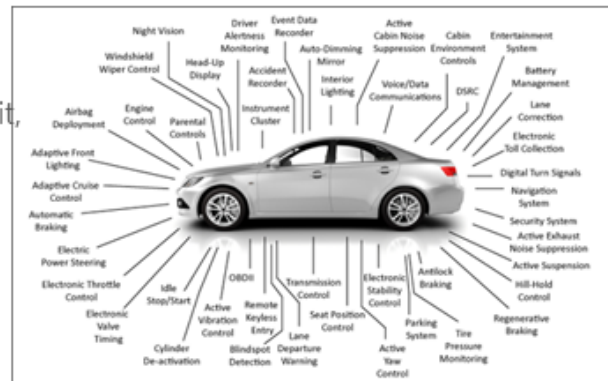
- Multitude of sensors/actuators, Heterogeneous communication interfaces, Variety of processing unit,
- Reliability concerns -> redundancy, increases complexity,
- High value components and new concepts / trends.

Reducing cost VS accelerating validation

- Automotive market is volume driven,
- ECUs cost including testing and validation: 35% (2020) - 50% (2030) of the total vehicle cost, billion of miles of testing,
- Need to optimize the validation processes and leverage new testing methods.

Heterogeneous architectures

- Multiple busses: CAN, FlexRay, Automotive Ethernet, V2X, etc.
- Multiple subsystems: not all made to communicate or interact, running at different rates and different hardware architectures,
- Data sources, sinks and flows: two or more systems can share the same sensor on different interfaces,
- Different timing constraints: a video stream for entertainment VS video stream for an AEB system, Systems response time



CONTEXT AND ISSUES (2/2)

Communication and cybersecurity

- Wireless communication technologies: Enhance vehicles interaction with their environment, Constitute an additional source of information, Can create an entry point for a cybersecurity threat,
- Shared mediums / backbone: Ensure sensors/ subsystems interactions, Can propagate the risk to safety critical systems.

Artificial intelligence and data fusion

- Constitute the senses and the brain of the intelligent and autonomous vehicle,
- A must for richer information construction and environment awareness,
- Not an easy task but can be of a high value,
- No clear approach to how to test and validate such systems.

Regulatory constraints

- Real world testing,
- Responsibility and liability.



SOLUTIONS: FROM SIMULATION TO REAL HARDWARE TESTING (1/2)

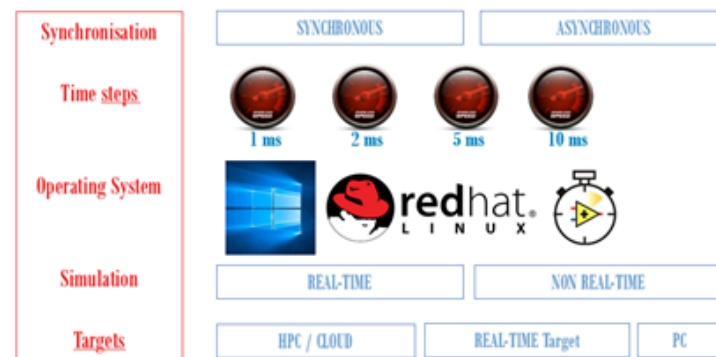
5

Simulation : Why ?

- Easy to fit, lower cost, faster validation,
- Validation at earlier stages, easier debugging, corrections and design changes,
- Interoperability, Hardware abstraction / virtualization,
- Easy upgrade of the simulation environment compared to the real-world testing,

Simulation: How?

- A complete simulation based testing solution does not exist
- Interoperability between simulation environments to achieve a common goal,
- Tests repeatability and optimization,
- Time constraints
 - off-line (as a first step)
 - real-time (for safety critical systems)
 - faster than real-time (accelerated),

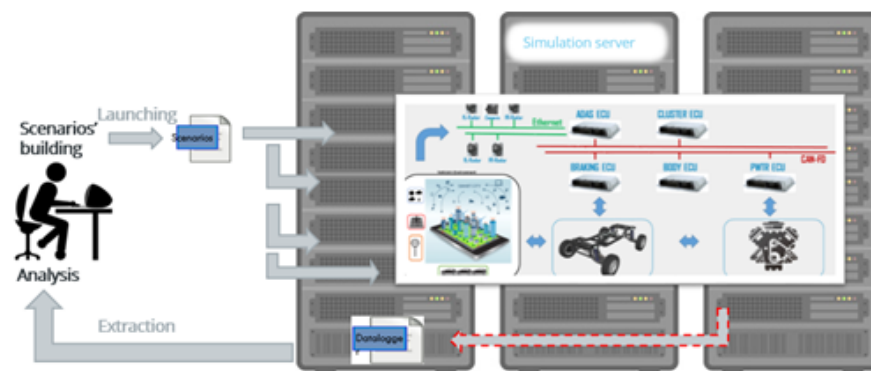


SOLUTIONS: FROM SIMULATION TO REAL HARDWARE TESTING (2/2)

6

Simulation: To what extent? Massive!!!

- Billions of miles driven to validate the entire vehicle,
- Simulations and scenarios optimization: can be AI based,
- Divide and rule: from the subsystems to the entire architecture,
- Reduce the duration of the physical testing,
- Needs of AI simulation based procedures for AI systems testing,



Hardware and software based co-simulation

- Simulate the environment while testing real ECUs,
- Lower risks and testing systems versatility,
- Explore multiple processing architecture with lower costs.



SOLUTIONS: OPAL-RT ADDED VALUE (1/4)

RT-LAB :

- A Real-Time architecture for distributed simulations,
- Ensures models execution over a variety of processing cores (X86, ARM, FPGA, etc.),
- Supports a multitude of Interfaces on Real-Time Targets for enhanced versatility.

ORCHESTRA :

- An Application-level data communication layer that leverages RT-LAB Framework,
- A plug and play tool for data mapping and synchronization between Co-Simulation environments,
- Data-centric for ease of integration and interoperability,
- Easily configurable through an XML-based DDF and GUI,
- Does not interfere with the simulation performances (high performances, scalability, etc.),

RT-LAB Orchestra is a high-performance Co-Simulation framework optimized for laboratory-scaled real-time HIL simulation and High-Performance Supercomputer using private and public servers



SOLUTIONS: OPAL-RT ADDED VALUE (2/4)

8

Targets listing

Model/Project Explorer

I/O, Panels

View and config.

The screenshot displays the RT-LAB v11.3.3.62 software interface. On the left, the 'Project Explorer' shows a tree structure with 'Targets' (localhost, r16.5), 'Models' (OrchestraFramework, model1, model2, model3, model4), 'I/O Interfaces', 'Panels', 'Recorders', and 'Configuration (Default)'. The 'Execution Properties' panel is open, showing 'Real-time simulation mode' set to 'Simulation' and 'Real-time communication link type' set to 'Simulation with low priority'. The 'Performance Properties' panel shows 'Action to perform on overruns' set to 'Continue'. The 'Subsystem settings' dialog is open, showing 'Subsystem' settings for 'SM_Model1'. The 'Subsystem Core Selection' dialog is also open, showing 'Total Number of Allocated Cores' set to 2. The 'Monitoring' tab is active, displaying a table of performance metrics for 'Model: model1 Ts=0.001[s] T=476.334[s] Number of overruns:0'. The table includes columns for 'Probes', 'Info', 'Usage [%]', 'Min', 'Max', 'Mean', and 'Data' (columns 1-7). The 'Data' column shows values for 'dt= 21.93 [us]', 'dt= 49.03 [us]', 'dt= 25.62 [us]', 'dt= 28.73 [us]', 'dt= 24.18 [us]', 'dt= 24.20 [us]', 'dt= 25.16 [us]', 'dt= 25.05 [us]', 'dt= 25.91 [us]', and 'dt= 25.37 [us]'. The 'Usage [%]' column shows values for 'dt= 2.56%', 'dt= 0.02%', 'dt= 2.29%', 'dt= 0.04%', 'dt= 2.56%', 'dt= 2.63%', and 'dt= 0.0%'.

Model execution controls: simulation mode, Target, stop and pause time, etc.

Processing cores assignment view

Models performances monitoring: based on last 100 Time Steps, Configurable performances probes

Probes	Info	Usage [%]	Min	Max	Mean	Data			
Model: model1 Ts=0.001[s] T=476.334[s] Number of overruns:0									
dt= 21.93 [us]	dt= 49.03 [us]	dt= 25.62 [us]	dt= 28.73 [us]	dt= 24.18 [us]	dt= 24.20 [us]	dt= 25.16 [us]	dt= 25.05 [us]	dt= 25.91 [us]	dt= 25.37 [us]
dt= 0.02 [us]	dt= 2.51 [us]	dt= 0.21 [us]	dt= 0.19 [us]	dt= 0.17 [us]	dt= 0.17 [us]	dt= 0.20 [us]	dt= 0.16 [us]	dt= 0.16 [us]	dt= 0.16 [us]
dt= 19.34 [us]	dt= 46.00 [us]	dt= 22.85 [us]	dt= 25.96 [us]	dt= 21.25 [us]	dt= 21.43 [us]	dt= 22.44 [us]	dt= 22.35 [us]	dt= 23.26 [us]	dt= 22.81 [us]
dt= 0.04 [us]	dt= 0.73 [us]	dt= 0.43 [us]	dt= 0.44 [us]	dt= 0.52 [us]	dt= 0.67 [us]	dt= 0.49 [us]	dt= 0.44 [us]	dt= 0.33 [us]	dt= 0.34 [us]
dt= 0.28 [us]	dt= 0.73 [us]	dt= 0.43 [us]	dt= 28.73 [us]	dt= 24.18 [us]	dt= 24.20 [us]	dt= 25.16 [us]	dt= 25.05 [us]	dt= 25.91 [us]	dt= 25.37 [us]
dt= 21.93 [us]	dt= 49.03 [us]	dt= 25.62 [us]	dt= 29.64 [us]	dt= 25.40 [us]	dt= 24.91 [us]	dt= 25.98 [us]	dt= 25.65 [us]	dt= 26.52 [us]	dt= 25.93 [us]
dt= 22.43 [us]	dt= 49.68 [us]	dt= 26.33 [us]	dt= 0.00 [us]	dt= 0.00 [us]	dt= 0.00 [us]	dt= 0.00 [us]	dt= 0.00 [us]	dt= 0.00 [us]	dt= 0.00 [us]



SOLUTIONS: OPAL-RT ADDED VALUE (3/4)

9

Recorders listing
for Data logging

3 Output files formats
Supported: CSV, MAT,
OPREC

Data Logger recorders
configuration panel: file format, recording duration,
triggers configuration, signal to recorder assignment,
recording rate compared to the model execution rate, etc.

Real-Time recorded signals values view,
Trigger selection,
Signals selection for logging, etc.

The screenshot displays the RT-LAB v11.3.3.62 software interface. The top menu bar includes File, Edit, Navigate, Search, Simulation, Tools, Window, and Help. The Project Explorer on the left shows a project structure with folders for Models, I/O Interfaces, and Panels. Under the Recorder panel, a list of recorders is shown, including SIGNAL_GROUP_1 and several model-specific recorders. A green box highlights the output file names, showing formats like CSV, MAT, and OPREC. The main window displays the 'RECORDER_OrchestraFramework_SM_Framework Configuration' panel. It has tabs for General, Signal Groups, and Configuration. The General tab is active, showing a table of parameters for SIGNAL_GROUP_1, such as Export format (CSV), Output file auto naming (checked), Decimation rate (1), Log file length (2), Log file size unit (Minutes), and Show trigger configuration (unchecked). Below this, the 'Output file name > data' section shows a table of recorded signals with columns for Name, Path, Alias, and Value. The table lists multiple instances of signal1 and signal2. A green box highlights this table. The bottom status bar shows 'No operations to display at this time.'

Parameter	Value
Export format	CSV
Output file auto naming	<input checked="" type="checkbox"/>
Decimation rate	1
Log file length	2
Log file size unit	Minutes
Show trigger configuration	<input type="checkbox"/>

Save	Trigger	Name	Path	Alias	Value
<input checked="" type="checkbox"/>	<input type="radio"/>	signal2	cleanProject/OrchestraFramework/SM_Framework/Simulink M...	???	???
<input checked="" type="checkbox"/>	<input type="radio"/>	signal1	cleanProject/OrchestraFramework/SM_Framework/Simulink M...	???	???
<input checked="" type="checkbox"/>	<input type="radio"/>	signal1	cleanProject/OrchestraFramework/SM_Framework/Simulink M...	???	???
<input checked="" type="checkbox"/>	<input type="radio"/>	signal1	cleanProject/OrchestraFramework/SM_Framework/Simulink M...	???	???
<input checked="" type="checkbox"/>	<input type="radio"/>	signal1	cleanProject/OrchestraFramework/SM_Framework/Simulink M...	???	???
<input checked="" type="checkbox"/>	<input type="radio"/>	signal2	cleanProject/OrchestraFramework/SM_Framework/Simulink M...	???	???
<input checked="" type="checkbox"/>	<input type="radio"/>	signal2	cleanProject/OrchestraFramework/SM_Framework/Simulink M...	???	???
<input checked="" type="checkbox"/>	<input type="radio"/>	signal1	cleanProject/OrchestraFramework/SM_Framework/Simulink M...	???	???

SOLUTIONS: OPAL-RT ADDED VALUE (4/4)

Hosting RT-LAB and the simulation environment on the same physical machine: all is simulated

- Reduced latency and optimized data flows,
- Accommodate massive deployment on cloud data center,
- Optimizing simulation environment execution,

Hosting RT-LAB and the simulation environment on the same physical machine: combination of simulation and real hardware / interfaces

- In addition to the previous advantages,
- Access to highly efficient I/O such as FPGA-based ones,
- Optimizing simulation environment execution if not ensuring its execution in real time,
- Possibility to handle hardware based co-simulation: GPU for sensors simulation, FPGA for I/O handling, and CPU for controllers testing and validation,
- Supports a multitude of Interfaces on Real-Time Targets for enhanced versatility.



WHAT IS TO COME

Standard interface for co-simulation platforms: ACOSAR, SYSML,
GPU support for AI and Data Fusion processes,
Multi-instance multi-project support optimization for massive
simulation deployment,
Optimized testing scenarios selection based on AI,
Cybersecurity testing and protection.

