Extending OMNeT++ Towards a Platform for the Design of Future In-Vehicle Network Architectures

Till Steinbach Stefan Buschmann Philipp Meyer Franz Korf

> Hamburg University of Applied Sciences philipp.meyer@haw-hamburg.de

OMNeT++ Community Summit 15. September 2016, Brno University of Technology, Czech Republic





Hochschule für Angewandte Hamburg University of Applied Sciences



SPONSORED BY THE

Agenda



Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

Simulation Models

TOOICHAIH

/orkflow

Conclusion & Outlook

1 Introduction

2 Simulation Models

3 Toolchain

4 Workflow

Introduction



■ In-vehicle networks face a paradigm change

■ Communication architectures today:

- CAN
- FlexRay
- LIN
- MOST
- Switched real-time Ethernet is promising candidate for future communication architectures¹
- Stepwise transition from heterogeneous bus architecture towards a single flat Ethernet topology

Extending OMNeT++ Towards a Platform for the Design of Future In-Vehicle Network Architectures

Philipp Meyer

Introduction

Simulation Models

Vorkflov

¹Kirsten Matheus and Thomas Königseder:Automotive Ethernet. Jan. 2015.

Introduction



Current tools focus on bit-correct fieldbus simulation

■ New tools are required for design and evaluation

 These environments have to support analysis of congestion and jitter

■ The OMNeT++ platform provides a perfect base

■ We want to provide an easy to use environment

■ In this work we contribute:

■ Simulation models

■ Tools to design

■ Tools to evaluate

A uniform workflow

Extending OMNeT++ Towards a Platform for the Design of Future In-Vehicle Network Architectures

Philipp Meyer

Introduction

Simulation Models

Vorkflow

Agenda



Extending OMNeT++ Towards a Platform for the Design of Future In-Vehicle Network Architectures

Philipp Meyer

Simulation Models

Conclusion & Outlook

Introduction

2 Simulation Models

3 Toolchain

Workflow

Simulation Models Overview



Vehicle network model

SignalsAndGateways
Signal sources, Gateways

CorealNet
Real-time Ethernet
INET Framework
Internet Technologies / Protocols

OMNet++
IDE and Simulation kernel

provided

contributed

Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

Simulation Models

TOOICHAI

N/orlefton

- CoRE4INET
 (Communication over Real-time Ethernet for INET)
- FiCo4OMNeT (Fieldbus Communication for OMNeT++)
- SignalsAndGateways



Simulation Models



Currently supported standards:

■ TTEthernet protocol suite (AS6802)

AVB traffic shapers (IEEE 802.1Qav)

■ Ethernet with priorities (IEEE 802.1Q)

■ Currently supported features:

■ Models to map IP traffic to real-time traffic classes

■ Incoming traffic selection and constraint checks

■ Models for oscillators, timers and schedulers

Application models for traffic patterns

■ Flexible combining of media access strategies

Checked against analytical models and empirical tests

Extending OMNeT++ Towards a Platform for the Design of Future In-Vehicle Network Architectures

Philipp Meyer

ntroduction

Simulation Models

TOOICHAII

Vorkflow



Simulation Models



Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

Simulation Models

rooiciiai

Norkflow

Conclusion & Outlook

Currently supported standards:

- CAN
- FlexRay
- Currently supported features:
 - Models for oscillators and timers
 - Application models for traffic patterns
- Checked against results of CANoe simulation environment

Simulation Models

SignalsAndGateways



Extending OMNeT++
Towards a Platform for

the Design of Future In-Vehicle Network Architectures

Philipp Meyer

ntroduction

Simulation Models

rooiciiai

Norkflov

Conclusion & Outlook

■ Fills the gap between CoRE4INET and FiCo4OMNeT

- Gateway translate between (real-time) Ethernet and fieldbusses
- For flexibility it contains three submodules:
 - Routing
 - Buffering
 - Transformation
- Gateway can host applications

Agenda



Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

Simulation Madala

Toolchain

orkflow

Conclusion & Outlook

1 Introduction

2 Simulation Models

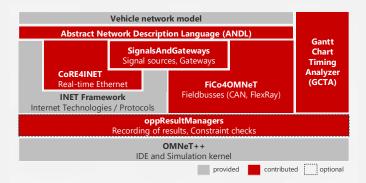
3 Toolchain

4 Workflow



Toolchain Overview





Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

itroduction

mulation Models

Toolchain

Workflow

- Eclipse Updatesite (https://sim.core-rg.de/updates)
 - -> get plugins
- CoRE model installer (OMNeT++ plugin)
 - -> get simulation models



Toolchain

Abstract Network Description Language



 Configuring large heterogeneous networks is complex and lengthy

Domain Specific Language (DSL) reduces effort

Eclipse plugin using Xtext technology

Supported features:

Syntax highlighting

Code completion

Scheduling algorithms (for TDMA technologies²)

Simple inheritance

Inline ini configuration

Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

troduction

Simulation Models

Toolchain

Vorkflow

² Jan Kamieth et al.: "Design of TDMA-based In-Car Networks: Applying Multiprocessor Scheduling Strategies on Time-triggered Switched Ethernet Communication". 2014.

Toolchain Gantt Chart Timing Analyzer



Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

Simulation Models

Toolchain

Workflow

Conclusion & Outlook

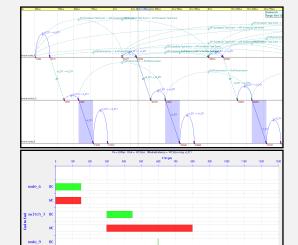
■ Specialized analysis tool as OMNeT++ plugin

- Traces jitter and delay in cyclic communication
- Uses a timing log (.tlog) file written during simulation
- GCTA compresses all occurrences of a cyclic message into one single chart

Toolchain

Gantt Chart Timing Analyzer





■ Best Case (BC) ■ Worst Case (WC)

Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

Introduction

Simulation Models

Toolchain

Workflow

Conclusion & Outlook



Hochschule für Angewandte Wissenschaften Hamburg

Hamburg University of Applied Sciences

Toolchain oppResultManagers



 Set of modules for OMNeT++ simulations: called ResultManagers

■ OMNeT++ vector and scalar recording are build-in

Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

instances of ResultManagers

Philipp Meyer

■ Contributed in oppResultManagers:

ntroduction

PCAPng

Simulation Models

SQLite & postgreSQL

Toolchain

■ Constraint Checks

۸/ ۵ سا دا ۱ ۵۰۰۰

■ Multiple

Conclusion & Outlook

 Functionality is not restriced to our simulation models

Agenda



Introduction

Towards a Platform for the Design of Future In-Vehicle Network Architectures

Philipp Meyer

Extending OMNeT++

Workflow

Conclusion & Outlook

Simulation Models

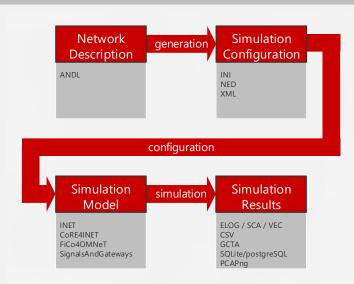
3 Toolchain

4 Workflow



Workflow Overview





Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

Simulation Models

Workflow

Network Description



Abstract Network Description Language File:

```
//Types can be defined and reused
tvpes std {
  ethernetLink ETH { // Definition for Ethernet link
    bandwidth 100Mb/s; //Link has bandwidth of 100MBit/s
    //it is also possible to define types in a separate file
network smallNetwork{ //network name is smallNetwork
  inline ini{
              //Inline ini for special parameters
    record-eventlog = false
                      //Parameters are inserted into .ini
  devices {
                      //Define all devices in the network
    canLink bus1: // First CAN bus
    canLink bus2: // Second CAN bus
    node node1: // First CAN node
    node node2: // Second CAN node
    gateway gwl; //Gateway for first CAN bus
gateway gw2; //Gateway for second CAN bus
    switch switch1; //Real-time Ethernet Switch
```

Extending OMNeT++ Towards a Platform for the Design of Future In-Vehicle Network Architectures

Philipp Meyer

troduction

Simulation Models

TOOICHAI

Workflow

Conclusion & Outlook

Continued on next slide...



Network Description



```
connections { // Physical connections (Segments = groups)
  segment backbone { // Ethernet Backbone part
   gw1 \iff \{new \ std . ETH\} \iff switch1; //Ethernet Link
   gw2 \iff \{new \text{ std.ETH}\} \iff switch1; //Ethernet Link
  segment canbus { // CAN bus part (busses share config)
   node1 <--> bus1: //CAN node connected to first bus
   gw1 <--> bus1; // Gateway connected to first bus
   node2 <--> bus2: //CAN node connected to second bus
   gw2 <--> bus2: //Gateway connected to second bus
communication { // Communication in the network
  message msg1{ // Message definition
   sender nodel; // First CAN node is sender
    receivers node2; //Second CAN node is receiver
   payload 6B; //Message payload is 6 Bytes
    period 5ms; //5ms cyclic transmission
   mapping { // mapping to traffic class, id, gw strategy
     canbus: can{id 37;}; // Message ID 37 on CAN
     backbone: tt{ctID 102;}; //TT traffic on backbone
     gw1: pool gw1_1{holdUp 10ms;}; //Aggregation time
```

Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

imulation Models

Loolchai

Workflow

Simulation Configuration / Model



OMNeT++IDE File Edit Navigate Search Project Run Window Help Simulation - CoRE4INET/examples andI/AS6802/small network.andI - OMNeT++ IDE - /Users/tillsteinbach/work B W * O + Q + A + B F + B + C ← + □ + A Project Explorer 23 ▼ 3 > CoRE4INET [CoRE4INET integration] network smallNetworkTTE { devices { ▶ ∰ Binaries node node1: ▶ Mincludes node node2: ▶ #src node node3; ▶ (= .settings switch switch1: ▶ @ doc ▶ (2) > examples ▼ A examples and I connections { ▼ @ AS6802 segment default { small_network.ar node1 <--> { new std.ETH_100MBIT } <--> switch1; New ▶ @ avb_AS6802 node2 <--> { new std.ETH_100MBIT } <--> switch1: README node3 <--> { new std.ETH 100MBIT } <--> switch1: Open Std.andl Open With ► @ > images ▶ @ > other Copy #C ation f ▶ @ out Paste age vl 100 { ▶ @ simulations ¥ Delete sender node1; receivers node2, node3; Move payload 468; Rename... Properties 🛭 🔡 Outline period 5ms: small_network.an...ples_ import... default : tt { ctID 100: }: Export... Property Resource **∀Info** Clean Local age vl_101 derived fall sender node2: Refresh receivers node3: editable tru payload 46B: last m... 29 Make Targets period 1ms: location /U Profiling Tools 1odule Hierarchy (♣ NED Parameters (♣ NED Inheritance 💂 Console 🖷 P Profile As path /C Debug As size Run Ae Model Updates (Sleeping) Compare With Replace With - Generate Network W Run C/C++ Code Analysis Properties 961 small network.andl - CoRE4

Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

imulation Models

Loolchail

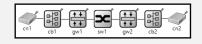
Workflow



Simulation Configuration / Model



- Uses all three simulation models and INET
- Generated config (.ini/.ned/.xml) > 250 lines
- Resulting network:



- Additional configuration of ResultManagers in ini file:
 - postgreSQL:

outputscalarmanager - class="cPostgreSQLOutputScalarManager" outputvectormanager - class="cPostgreSQLOutputVectorManager" postgresqloutputmanager - connection="dbname=testdb_user= testuser_password=testuser_port=15432"

■ PCAPng:

eventlogmanager - class = "PCAPNGEventlogManager"

Hochschule für Angewandte Wissenschaften Hamburg

Hamburg University of Applied Sciences

Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

imulation Models

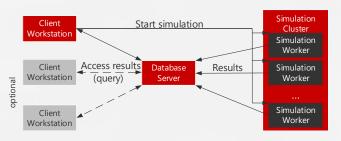
Workflow

Simulation Results

CORE_

- OMNeT++ (scalar, vector and eventlog)
- WireShark (PCAPng)
- GCTA (timinglog)
- Database (mySQL, postgreSQL)

Use-case postgreSQL database:



Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

Simulation Models

Toolchai

Workflow

Agenda



1 Introduction

2 Simulation Models

3 Toolchain

4 Workflow

5 Conclusion & Outlook

Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

Simulation Models

Toolchain

Vorkflow

Conclusion



■ In-car communication technologies are changing

■ Simulation on system-level supports the process

■ We contribute a simulation environment with:

■ Simulation models

■ Development tools

Analysis tools

Specialized tools can support the workflow

 OMNeT++ is a solid foundation for the development of such plugin tools Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

Simulation Models

Workflov

Outlook



Extending OMNeT++ Towards a Platform for the Design of Future In-Vehicle Network Architectures

Philipp Meyer

Adding new technologies to our simulation suite:

- Ethernet with frame preemption (discussed in IEEE 802.1Qbu)
- CAN with flexible data rate (CAN FD)
- Refinement of result analysis tools

atroduction

imulation Models

Vorkflov

Questions



Thank you for your attention!



- Website of CoRE research group: https://core-rg.de/
- Website of simulation models: https://sim.core-rg.de/

SPONSORED BY THE



Extending OMNeT++
Towards a Platform for
the Design of Future
In-Vehicle Network
Architectures

Philipp Meyer

ntroduction

Simulation Models

Workflow

References 1

[1]

[2]



Extending OMNeT++ Towards a Platform for the Design of Future In-Vehicle Network Architectures

Philipp Meyer

Conclusion & Outlook 19th IEEE International Conference on Emerging Technologies and Factory Automati

Barcelona: IEEE Press, 2014, pp. 1–9. DOI:

Kirsten Matheus and Thomas Königseder.

Applying Multiprocessor Scheduling Strate-

Cambridge University Press, Jan. 2015.

10.1109/ETFA.2014.7005119. ieeexplore: 7005119.

Automotive Ethernet. Cambridge, United Kingdom:

Jan Kamieth et al. "Design of TDMA-based In-Car Networks:

gies on Time-triggered Switched Ethernet Communication". In:

