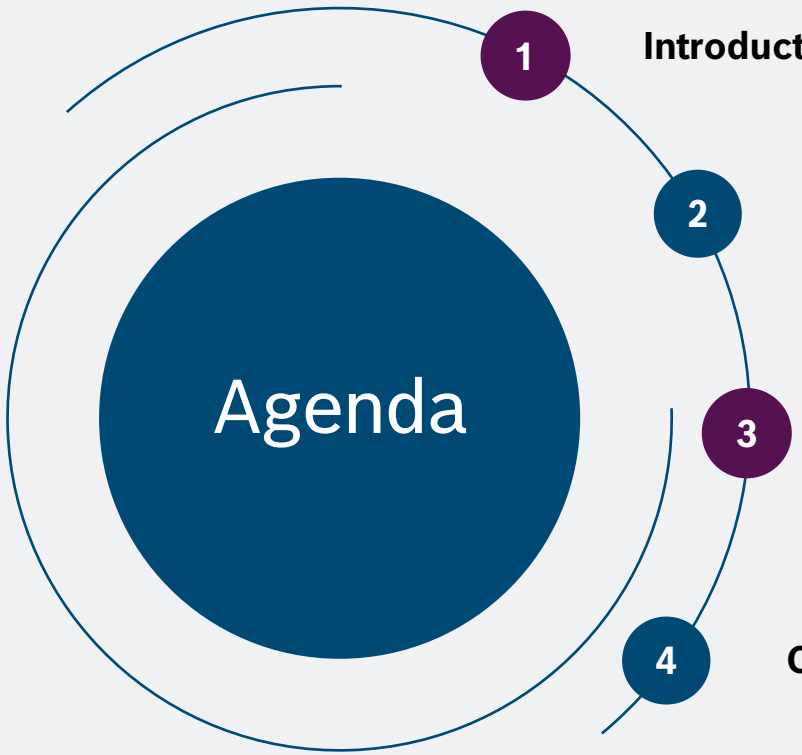
Automotive Ethernet Congress 2024

Semi-Automated Method for Design, Deployment, Verification, and Validation of In-Vehicle Service-Oriented-Architectures

Ricardo Gonzalez de Oliveira (Robert Bosch GmbH/Uni.Lu)
Jens Bierschenk (Robert Bosch GmbH)

uni.lu UNIVERSITÉ DU LUXEMBOURG
AUTOMOTIVE ETHERNET CONGRESS
BOSCH

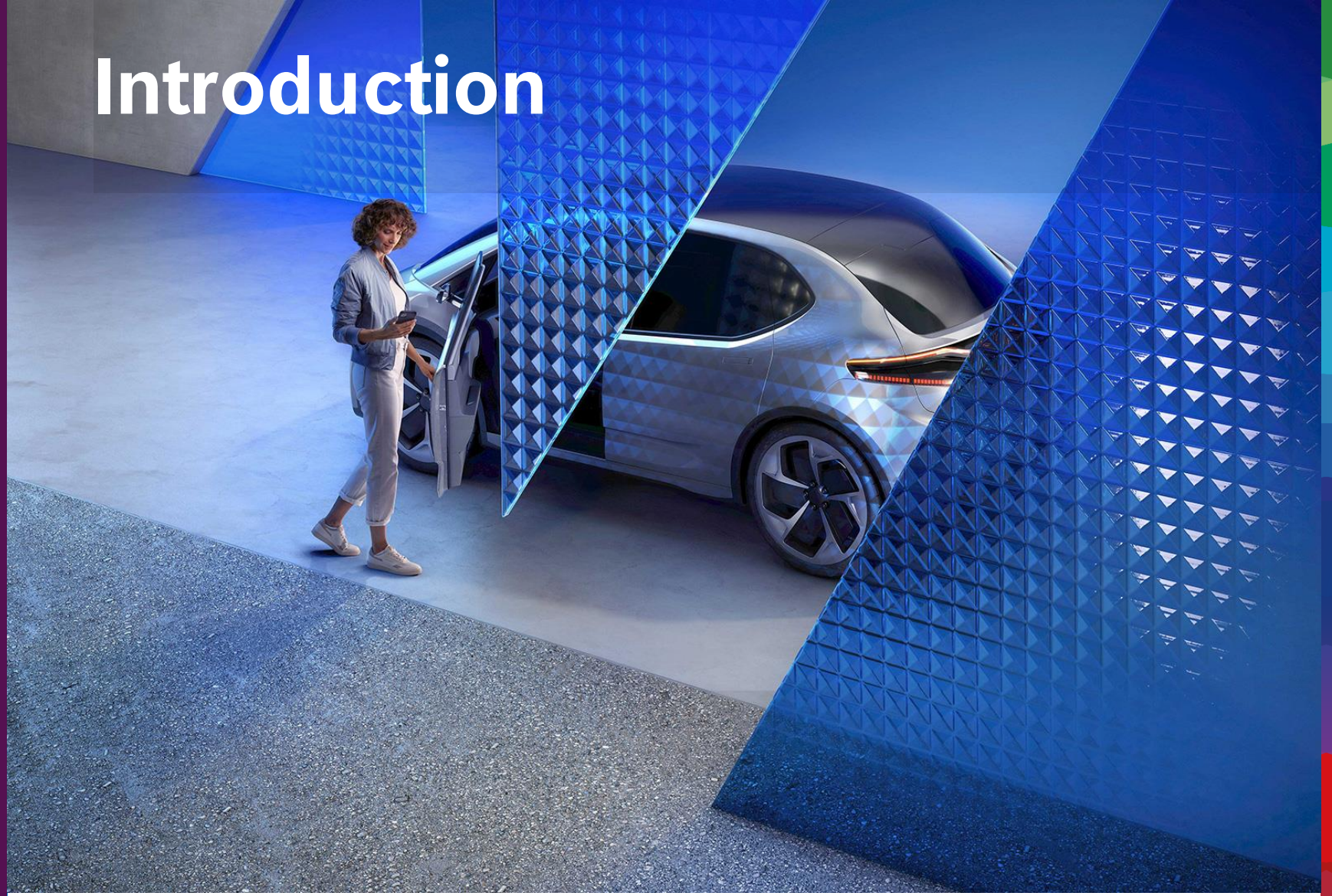
Agenda



1. **Introduction**
2. **Methodology for in-vehicle networks**
   From design to validation
3. **Applying the methodology**
4. **Conclusion** and **next steps**

01

Introduction



Introduction
Future perspectives for automotive Eth.-based E/E architectures

Automotive SPACE megatrends


Software and Services


Personalized


Automated


Connected


Electrified

Push to use solutions from IT to enable megatrends

- Service Oriented Architectures
- Multi-Gig Ethernet
- High-Perf. Computer
- Time Sensitive Networks
- Software Defined Networks
- Continuous Integration/Deployment

Future vision target for ethernet-based in-vehicle communication networks

IVN procedural design and deploy
Systematic generation and configuration of IVN to fit application requirements.

Abstraction
HW from SW and Signal from Services through harmonized gateways and interfaces.

IVN update / upgrade
(re)configurations and CI/CD practices through over-the-air updates enabled by software-defined-networks.

Scaling
Easy of reuse of E/E elements and configurations within different platforms.

E/E: Electric/Electronic; IVN: In-Vehicle Networks
CI/CD: Continuous Integration / Continuous Deployment

02

Methodology for in-vehicle networks

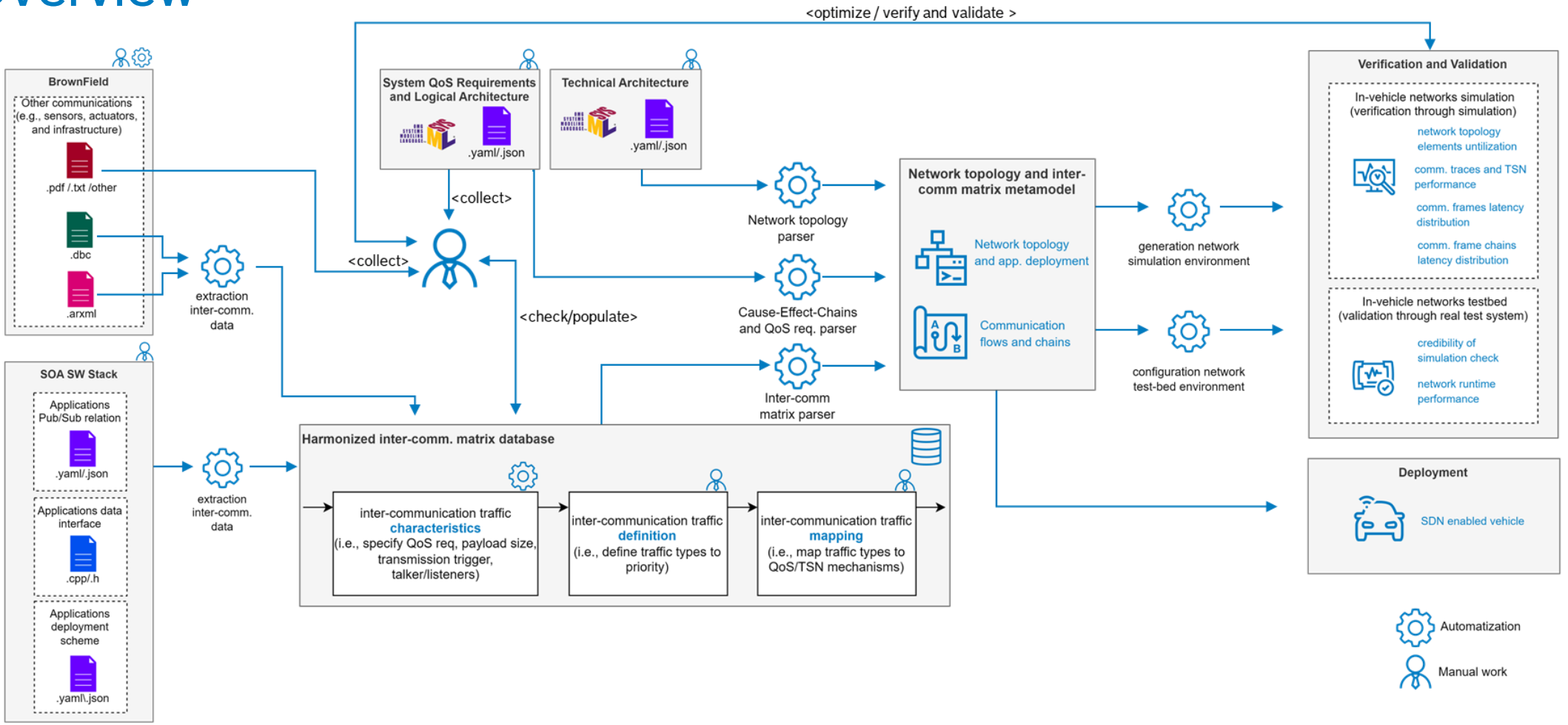
From design to validation

AUTOMOTIVE ETHERNET CONGRESS

BOSCH

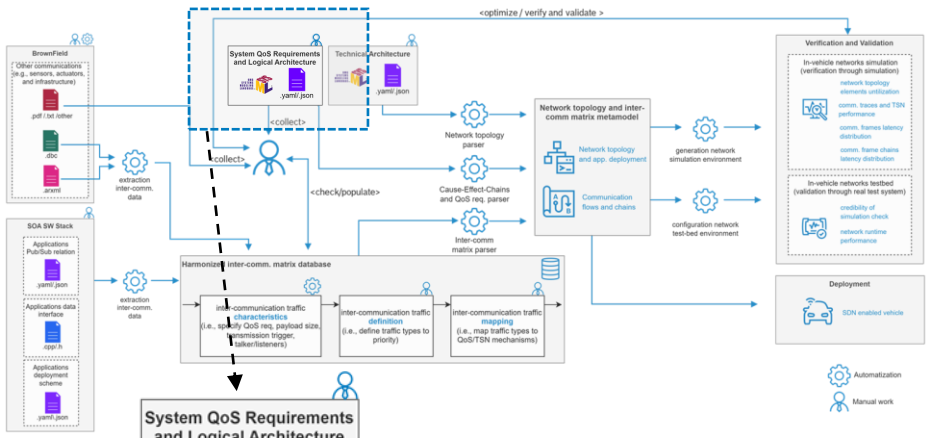
Methodology for in-vehicle networks. From design to validation

Overview



Methodology for in-vehicle networks. From design to validation

System requirements and logical architecture



Intersection driving scenario (showing ego vehicle)

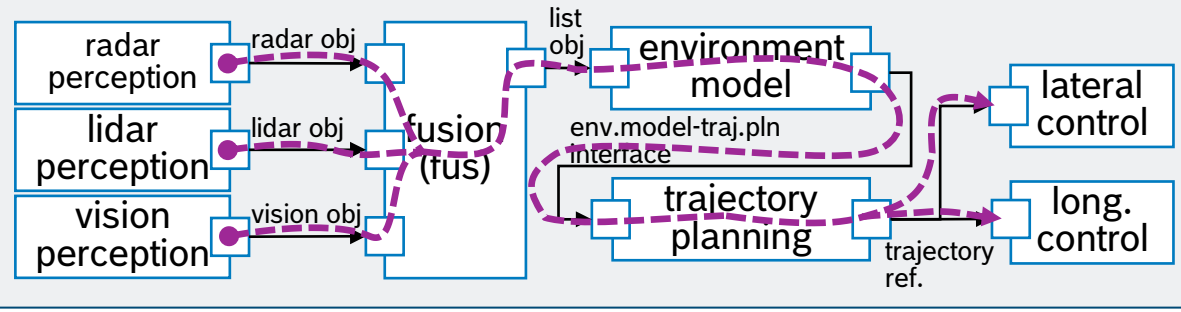
System requirements (example)

1) The *ego vehicle* shall be able cross intersections safely.

1.1) The ego vehicle shall use three different sense modalities.

1.2) The ego vehicle shall be able react to the intersection dynamic objects in less them x (ms).

Logical Architecture (example)



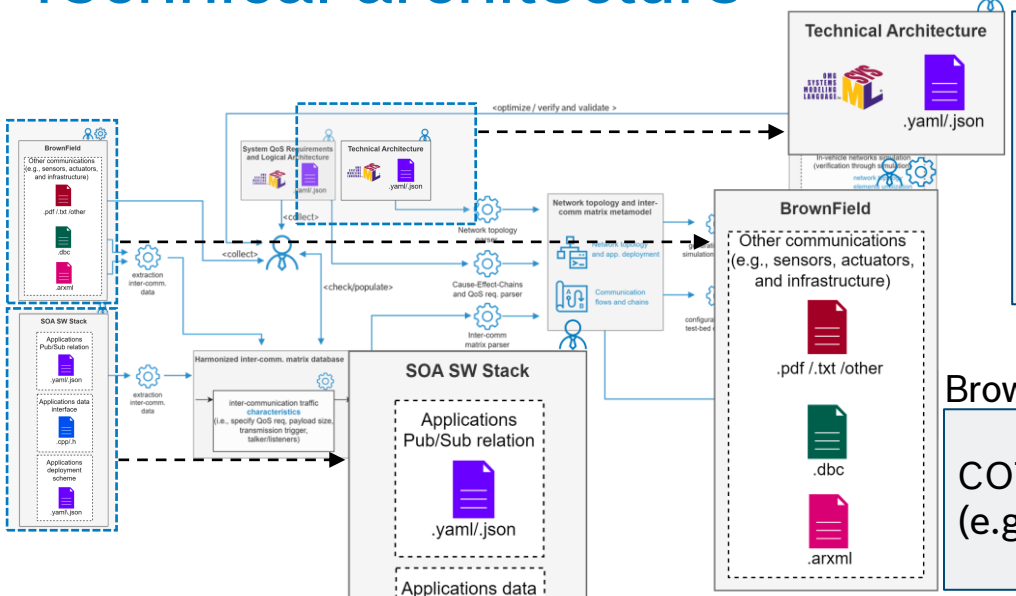
Functional and non-functional req. will lead to the logical and later technical arch. design including the QoS targets and chains

Cause-effect-chain

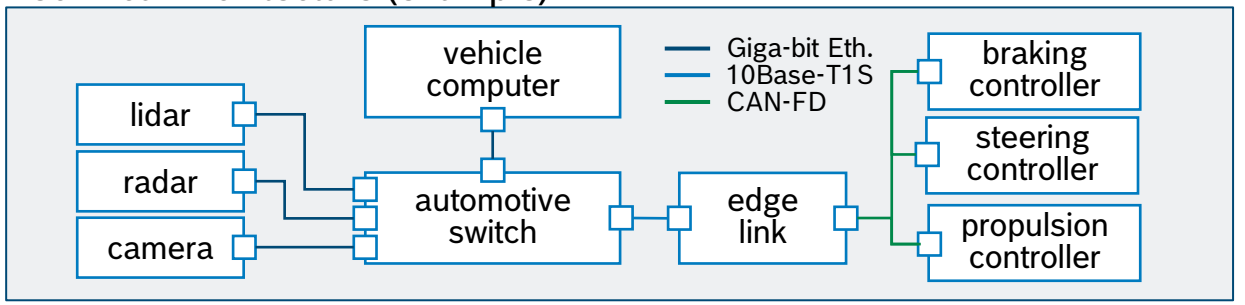
QoS: Quality of Service

Methodology for in-vehicle networks. From design to validation

Technical architecture



Technical Architecture (example)



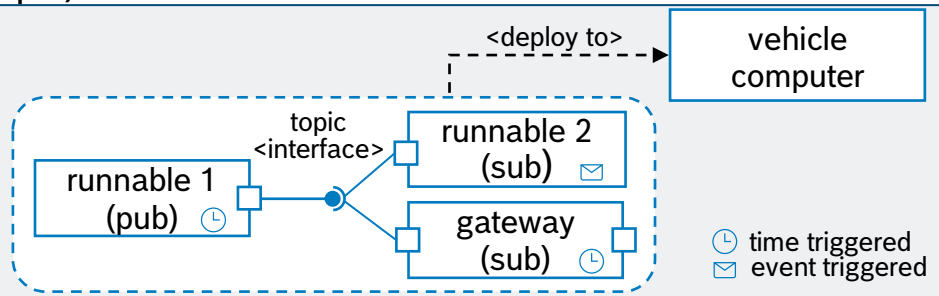
Brown Field (example)

COTS with predefined comm. traffic (e.g., interface, payload, cycletime)

Service Oriented Communication (example)

SW implementation leading to:

- publisher/subscriber relations
- scheduling parameters
- data interfaces parameters
- deployment scheme



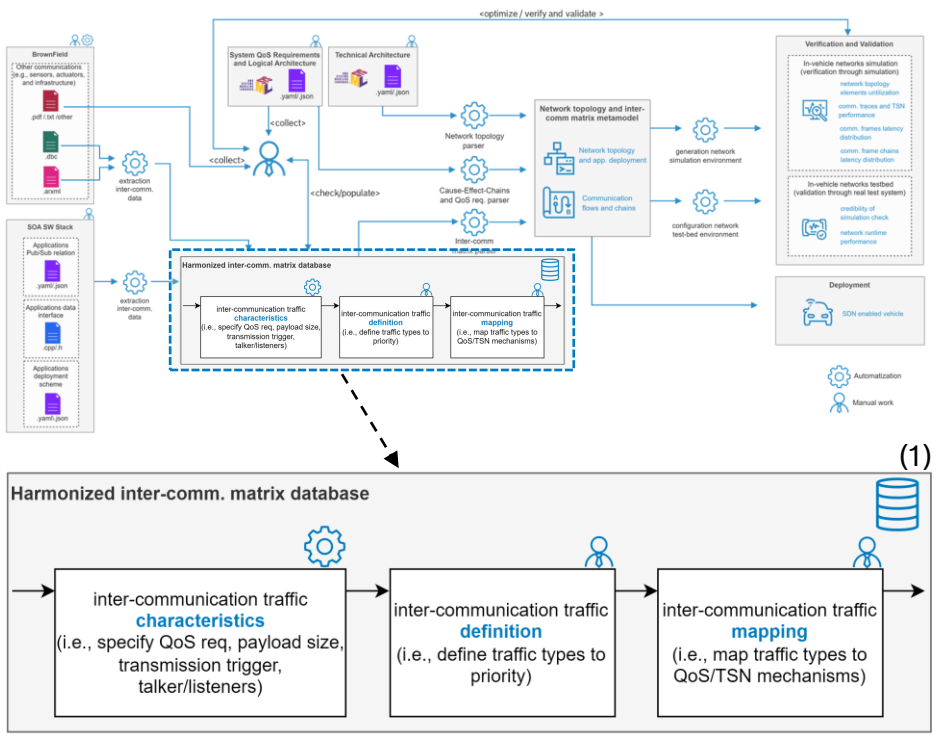
COTS: Comercial off-the-shelf
SOA: Service Oriented Architecture

AUTOMOTIVE ETHERNET CONGRESS

BOSCH

Methodology for in-vehicle networks. From design to validation

Communication matrix synthesis



(1)

Characteristics

- Talker and Listeners
- Data Transmission Trigger
- Data Delivery Guarantee
- Tolerance to Loss
- Data Size
- Data Criticality

Definition

According to characteristics, allocate traffic to types:
e.g., Critical Applications, Network Control, Diagnosis, Video Stream, Audio/Voice Stream, Best Effort.
Each type, can have an allocated priority:
e.g., Critical Applications priority > Video Stream priority> Best Effort priority

Mapping

Aggregate traffic types to queues and define TSN mechanisms accordingly.
e.g.,

Queue/Priority	Traffic type	TSN protocol
0	Best Effort	*none*
1	Video stream	1Qav(2)
2	Critical Application	*none*

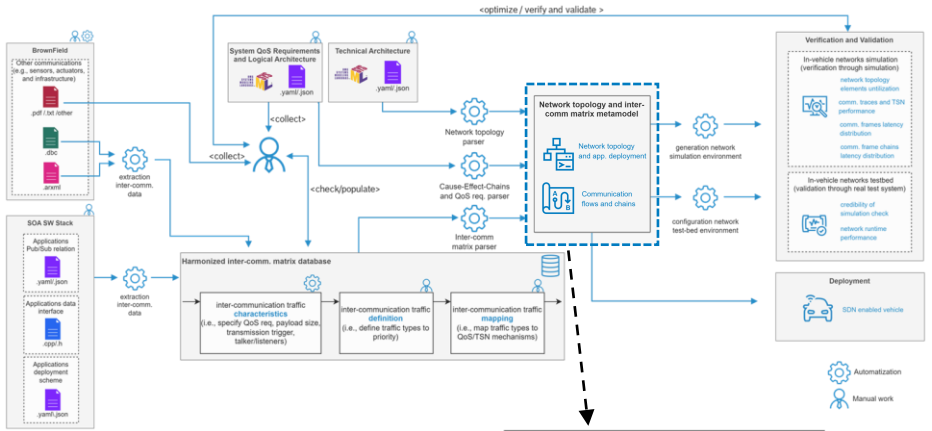
The activity to define *traffic types* according to QoS requirements and to correlate them to TSN mechanisms according to the number of queues makes possible a more transparent in-vehicle network design.

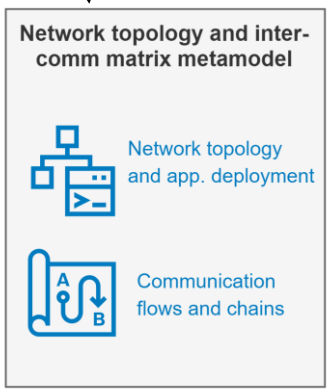
(1): Motivation IEEE 802.1DG (TSN Profile for Automotive In-Vehicle Ethernet Communications)
(2): IEEE 802.1Qav credit-based traffic shaper
TSN: Time Sensitive Networks; QoS: Quality of Service

AUTOMOTIVE ETHERNET CONGRESS BOSCH

Methodology for in-vehicle networks. From design to validation

In-vehicle networks common metamodel





Topology and deployment metamodels

Network topology:

- Description of the arrangement of the IVN elements to exchange data
  - Physical topology: Describes the structure of the network cabling
  - Logical topology: Controls the data flow between the end devices

Application deployment:

- Description of the end device hosts in the IVN where the functions will run

Communication flows and chains metamodels

Communication flows:

- Compilation of all exchange of data present at the IVN
  - Inter- and intra-communication flows and the correlated characteristics
  - Inter- and intra-communication flows and the correlated QoS req.

Communication chains:

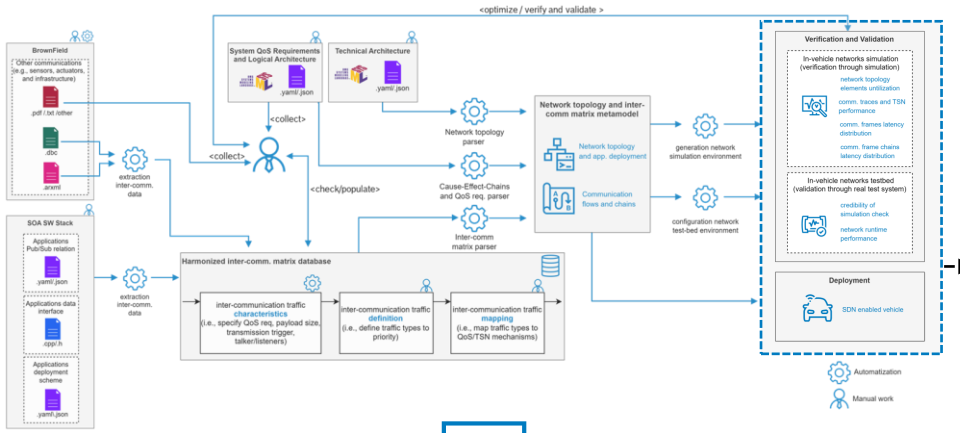
- Communication sequences according to functional chains and correlated QoS req.
- Gateways and tunneling functionalities

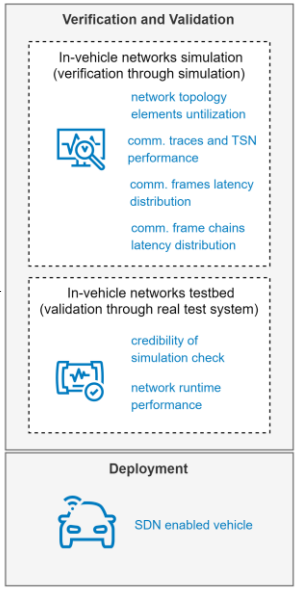
The metamodel for the IVN topology and communication flows enables a systematic interface to the V&V tooling.

QoS: Quality of Service;
IVN: In-Vehicle Network;
V&V: Verification and Validation

Methodology for in-vehicle networks. From design to validation

Verification, validation, and deployment



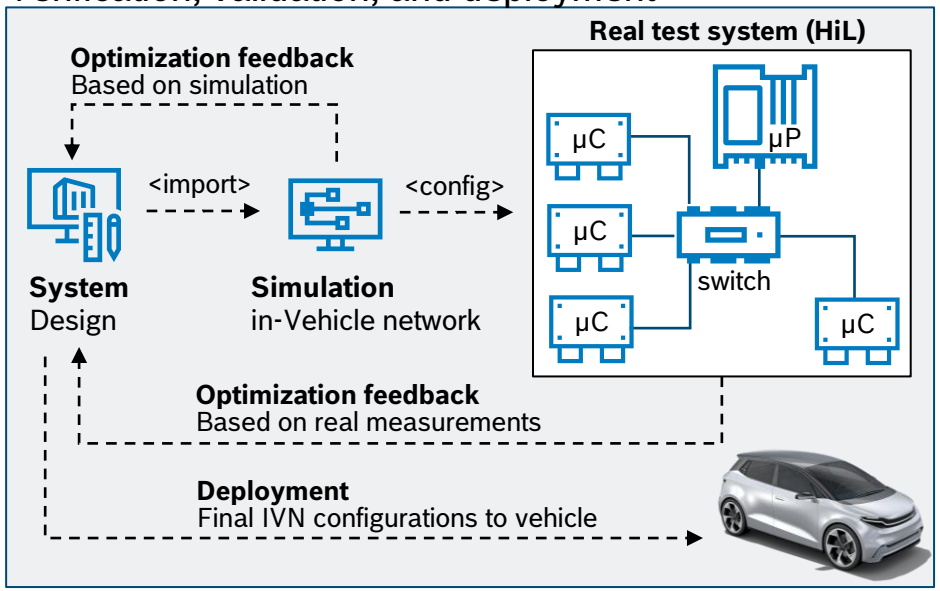


Tool / SDN Controller

<protocol>

Generic Switch or End-Nodes Module(s)
Model abstraction API (e.g., YANG, ARXML)
Connector Interface
Vendor HW API
Vendor Hardware Driver
Automotive Switch or µC/µP

Verification, validation, and deployment



The deployment steps requires a standardized procedure, for that we need:

- **An automotive model definition** for the IVN elements based on YANG/ARXML
  - Temporarily, AUTOSAR and YANG models shall be used in parallel;
  - **Finally**, YANG shall be the **data modeling language** for automotive

To achieve our methodology, an automotive SDN profile is necessary.
Let us work on further standardization for YANG models in IEEE and IETF.

API: Application Interface
HiL: Hardware-in-the-Loop
SDN: Software-Defined-Networks

03

Applying the methodology

Applying the methodology
Proof of concept and results

Proof of Concept Vehicle:

- Methodology applied to an in-house project for an ADAS/AD system.
- The vehicle consists of diverse sensors and actuators connected through a network to high-performance computers running complex algorithms based on SOA.

Results:

- Systematic IVN architecture designs and optimization without overprovisioning achieved.
- QoS requirements and cause-effect-chains identified and used for V&V and implications on technical architecture.

The proposed approach reduced manual work, using automation to accelerate the IVN's design time and optimization while mitigating human errors.

SOA: Service-Oriented-Architecture; IVN: In-vehicle Networks; QoS: Quality of Service
ADAS: Advanced Driving Assistance System; AD: Automated Driving
(1): https://www.bosch.com/stories/topics/automated-driving/

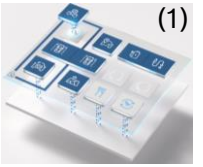
04

Conclusion and next steps

Conclusion and **next steps**
State-of-the-art, future work, and final comments

- **State-of-the-art:**

(2) MBSE


(1) SOA and interfaces

Comm. design rules

- **Future work:**

IVN comm. design rules

Determinisms over OSI layers

SDN for automotive

Most of the pieces of the proposed methodology are available and proved by the in-house project. Let us work together on the missing pieces to systematic design and deploy eth-based IVN.

MBSE: Model-Based-Systems-Engineering; SOA: Service-Oriented-Architecture;
IVN: in-Vehicle-Networks; OSI: Open Systems Interconnection; SDN: Software Defined Vehicle
(1): https://www.bosch-mobility.com/en/solutions/software-and-services/open-in-vehicle-api/
(2): http://www.omgsysml.org/

#LikeA**Bosch**

Thank you for **your attention**