

# Empowering tomorrow's automotive software

## Verification and Traceability in Automotive Software

In cooperation with



# Verification and Traceability in Automotive Software

ETAS

ETAS GmbH



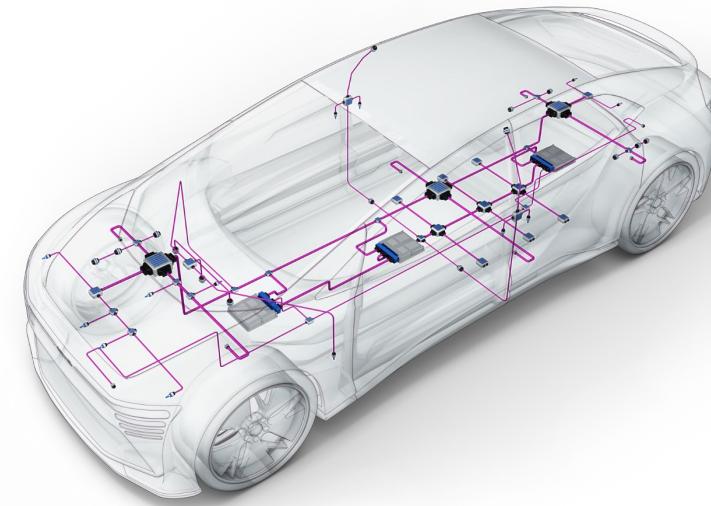
**Middleware Solutions** for AUTOSAR and automotive applications such as Advanced Driver Assistance Systems



**Measurement and Calibration Solutions** for fast and efficient automotive system development



**Security and Diagnostics Solutions** for protecting automotive systems against cyber threats and provide diagnostics services for maintenance



## Proven Expertise:

- Our middleware runs on more than **four billion** electrical control units (ECUs)
- Global presence
- 350+ satisfied clients
- 30 years of industry experience

# Verification and Traceability in Automotive Software

## Today's Automotive Challenges



### **Market Disruption and Accelerated Timelines:**

Automotive software must be updated quickly to keep pace with market demands while ensuring compliance with security, safety and traceability requirements



**System Complexity:** Modern vehicles integrate dozens of ECUs and millions of code lines, requiring systematic verification beyond expert intuition



**Fragmented Development Ecosystem:** Tools lack seamless integration, creating barriers and requiring resource-intensive manual processes

# Verification and Traceability in Automotive Software

## Solution Approach



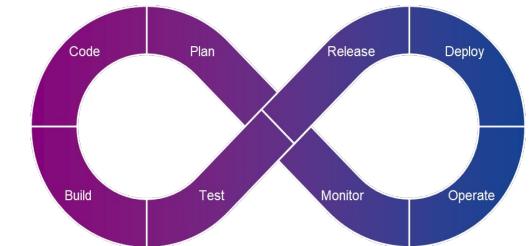
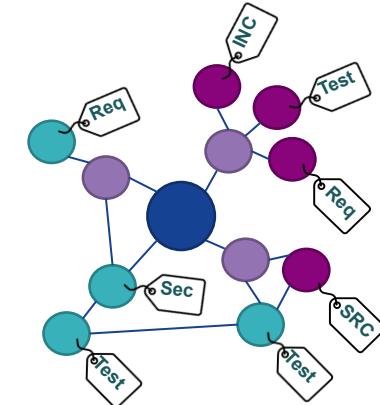
**Integrated Data Platform:** Centralizing data from different departments and systems makes it easier to access and collaborate effectively



**Data-Driven Processes:** Automate routine tasks and implement feedback loops, supporting continuous development and operation (DevOps)

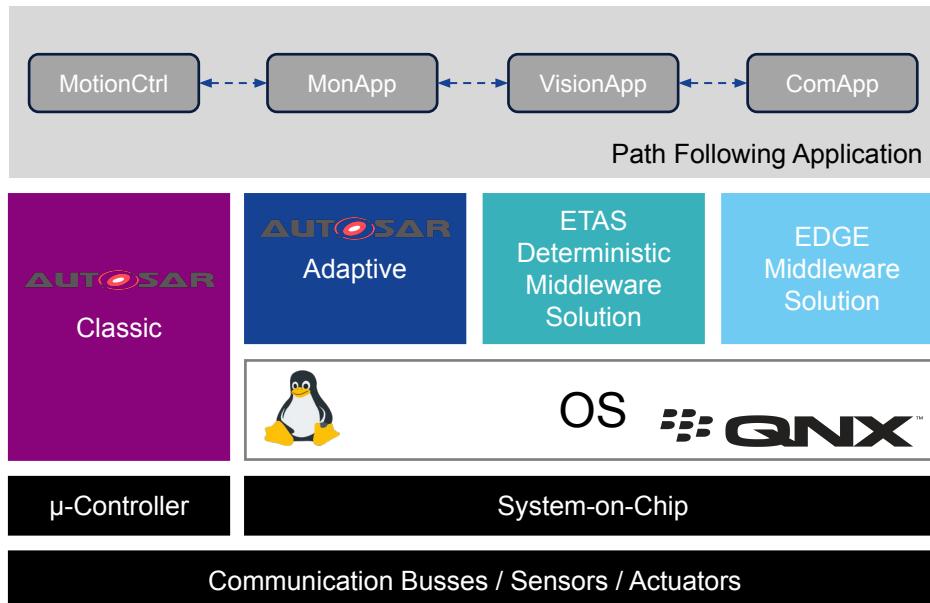


**Enable smart technologies** such as artificial intelligence (AI) through data engineering

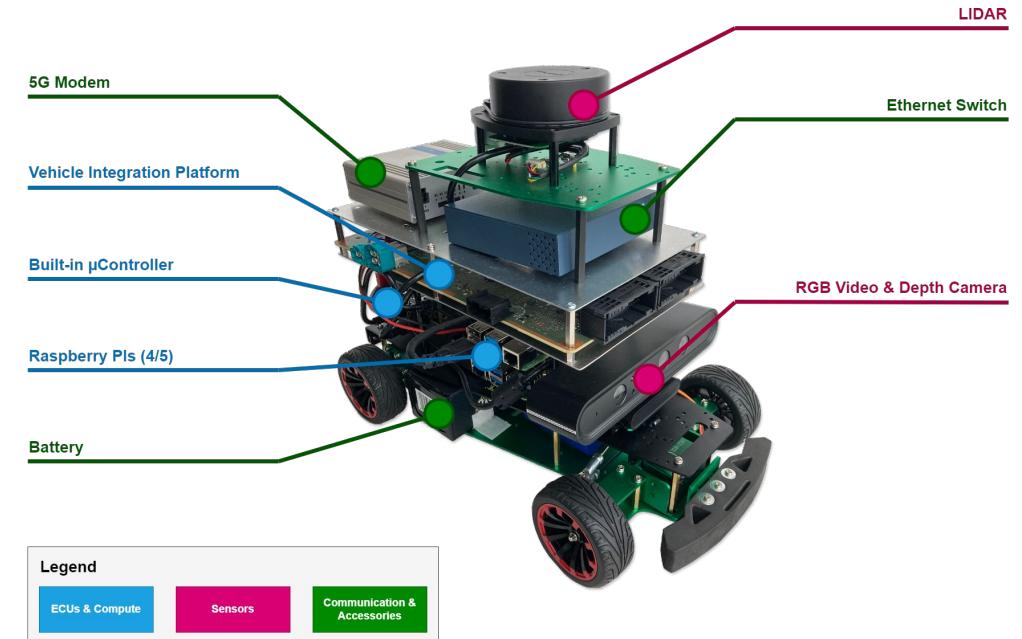


# Verification and Traceability in Automotive Software

## Automotive Demonstrator



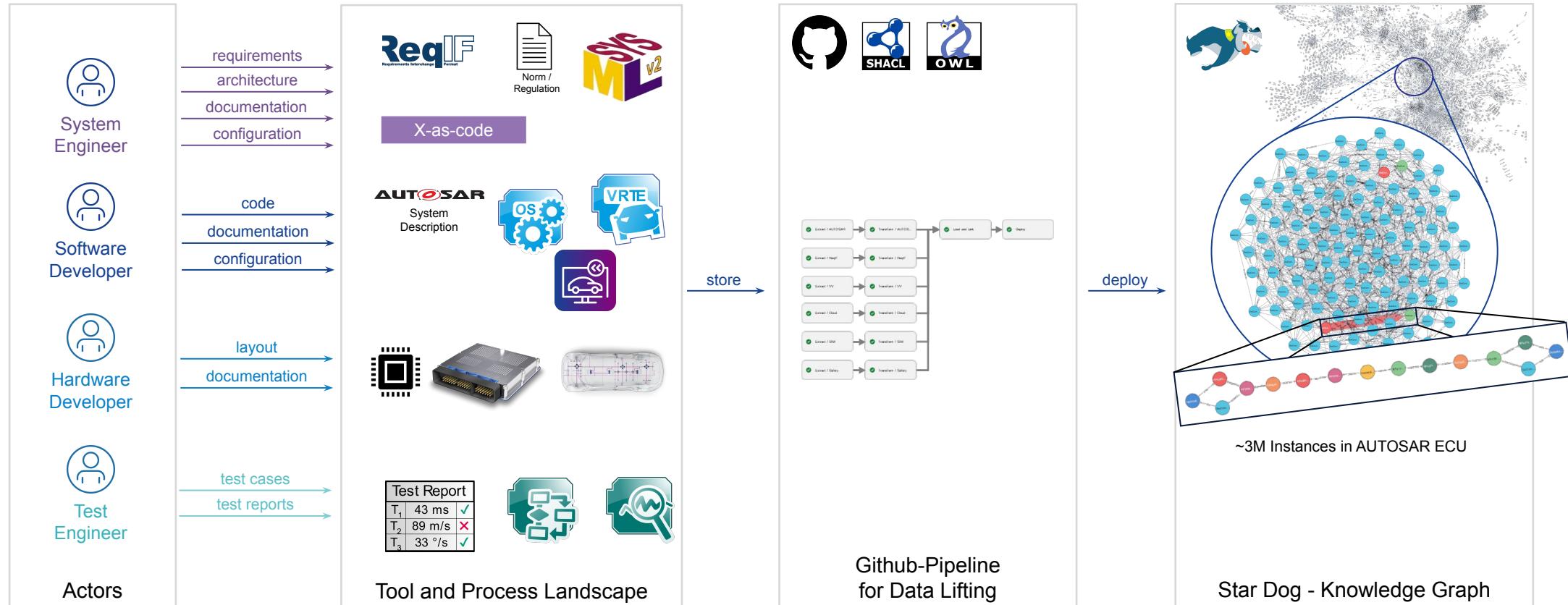
Software Stack



Demo-Car

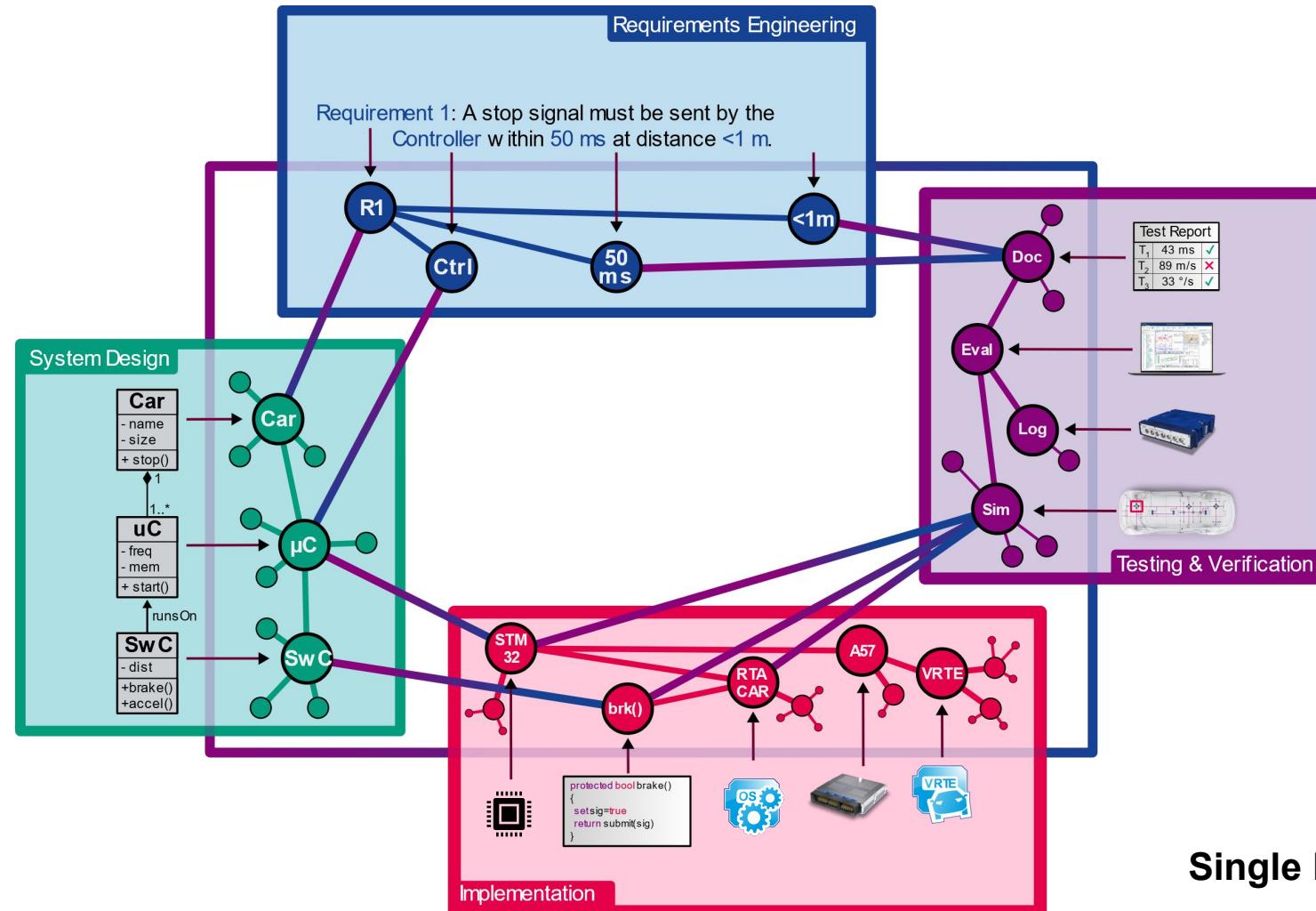
# Verification and Traceability in Automotive Software

## From Data to Knowledge



# Verification and Traceability in Automotive Software

## Data Mesh



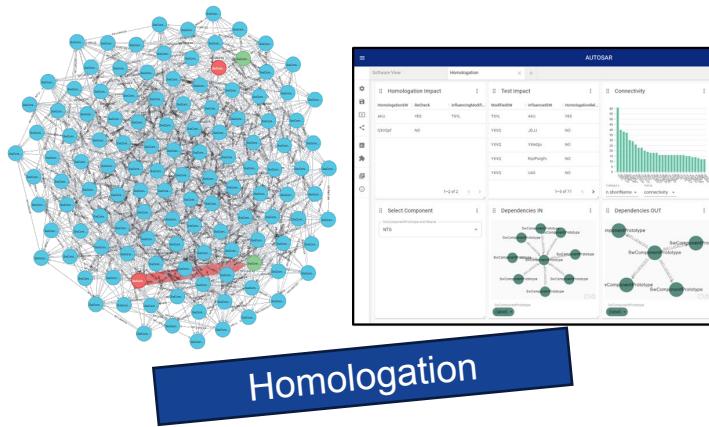
# Verification and Traceability in Automotive Software

## Use Cases in System Design

### Analytics and Reporting

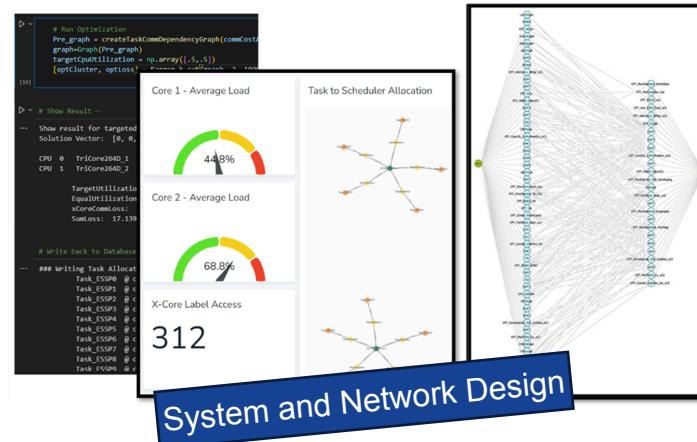
Graph analytics examines relationships within a network of connected nodes to reveal hidden patterns and insights.

In this use case, it facilitates an impact analysis for **homologation relevance checks** and generates a test campaign.



### Optimization

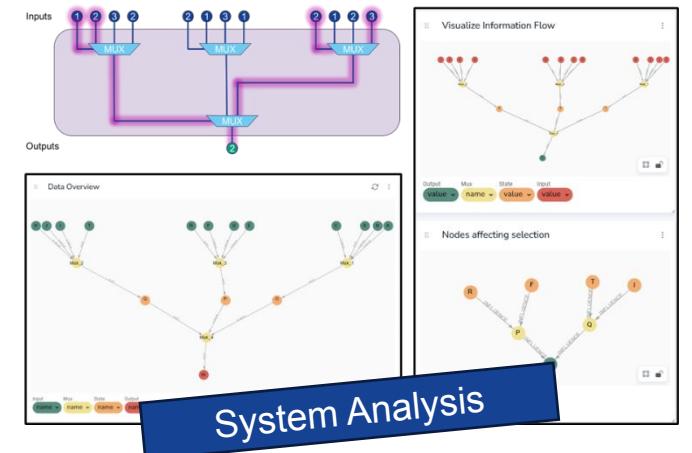
By analyzing existing data, connection patterns can be optimized. In the first use case (left), the **assignment of software functions to hardware** is automated using a machine learning-based optimization algorithm applied to the ECU's graph model. The second use case (right) demonstrates an **optimization of the communication matrix design**.



### Inference

Inference involves deriving new information from existing data using logical rules, enabling the automatic generation of statements that are not explicitly stored in the database.

In this use case, the **impact analysis is enhanced** by propagating values within the graph and pruning uninvolved nodes.

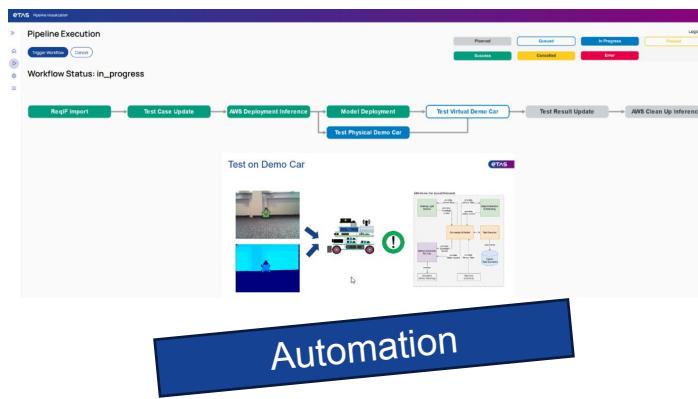


# Verification and Traceability in Automotive Software

## Use Cases in System Design

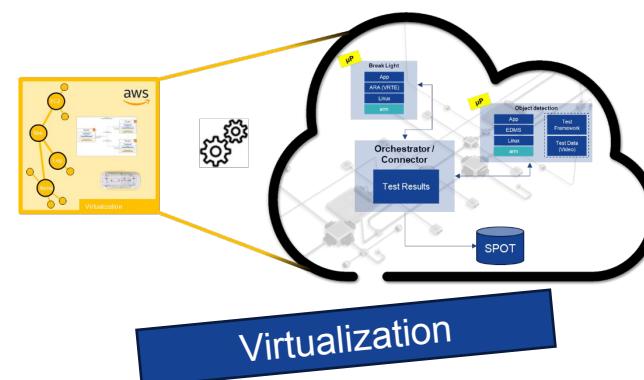
### Holistic Pipeline Automation

Data artifacts accessible via an API provided by a Single Point of Truth (SPOT) enable **process automation**. This is particularly valuable for automatically adapting test protocols to meet changing requirements, ensuring continuous compliance without additional manual effort.



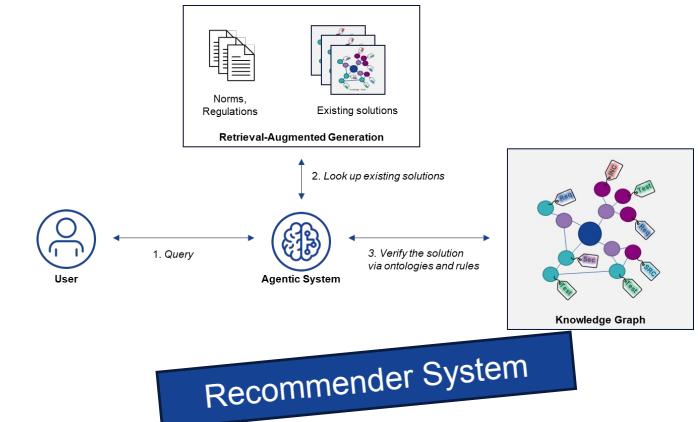
### System Virtualization and Testing

With available system data, connections and dependency patterns enable the automatic creation of **virtual testing** environments through **cloud deployment** matching the underlying target environments. This supports scalable testing with representative results and quick feedback loops.



### AI Perspective

AI-based systems, especially **Agentic AI**, are prone to errors in less commonly represented knowledge domains such as automotive engineering. **Knowledge engineering** is an **essential prerequisite** for a factual grounding of AI-based systems such as those relying on Large Language Models.



Thank you

Marco Schlichting - ETAS GmbH