

new eagle



RAPTOR®  
INNOVATION  
SUMMIT 2025

Michael Callaway, Senior Software Engineer  
New Eagle

# Simulation-Driven Engineering: Faster Iterations, Smarter Development

# Agenda



- Challenges with HIL
- Raptor Cloud
- Raptor-Dev Integration
- ETAS Toolchain
- Virtual HIL Use-Cases

new eagle



RAPTOR®  
INNOVATION  
SUMMIT 2025

# Challenges with Traditional Hardware-In-The Loop (HiL)

## Reduce risk from delays in BSW and HW development

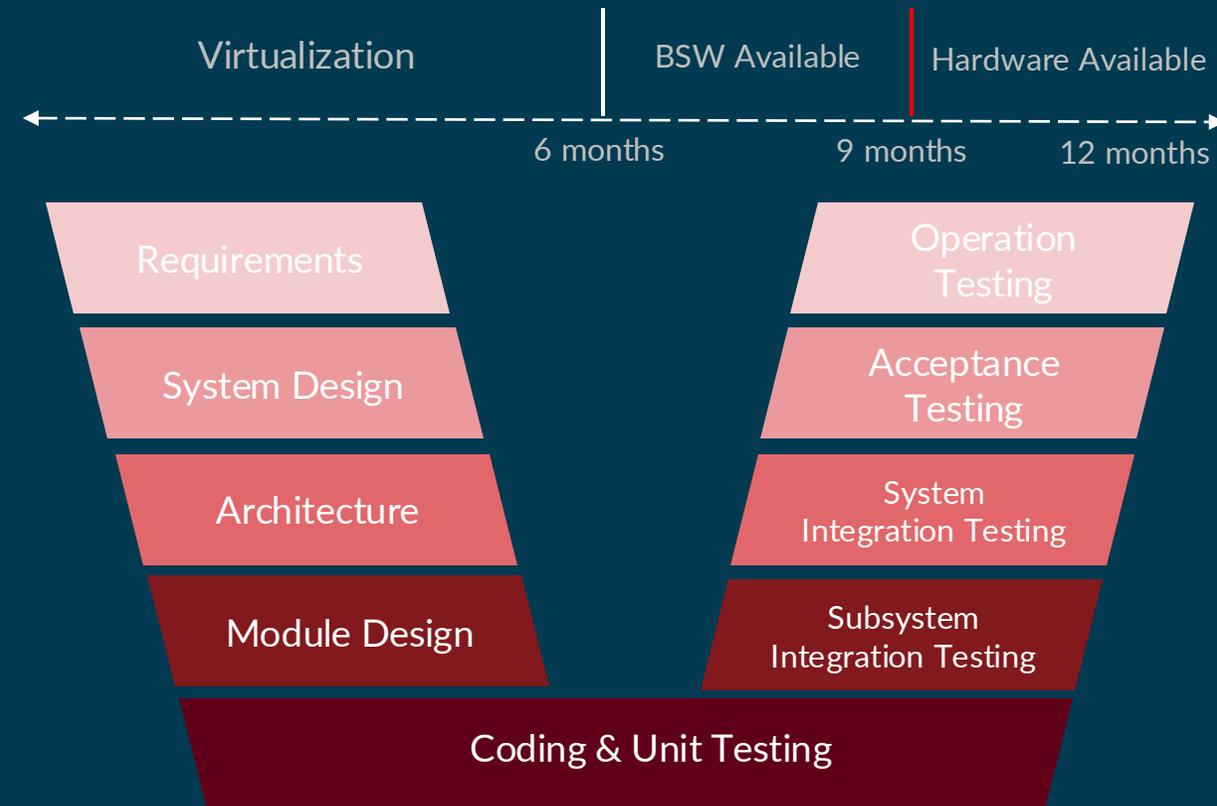
Waiting on Hardware (HW) and Board-Support Software (BSW)

- Slow feedback cycles

Time/Cost budgeting for validation

- Runtime on expensive HiL environments
- Repeated validation efforts

Limit Geographically Locked Resource





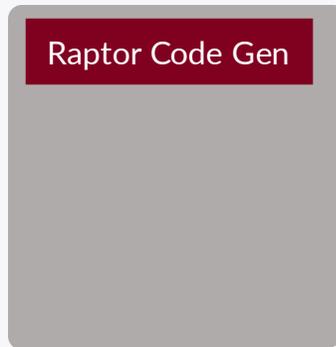
# How Do We Virtualize ECUs

## Our Immediate Focus

### Level 1 vECU

Integrates ASW only with signal level communication

- Enables quick application development & iteration
- Supports virtual HIL workflow

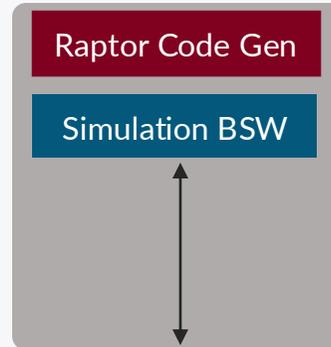


Signal Level  
Communication

### Level 2 vECU

Integrates ASW with simulated BSW

- Enables realistic network integration of multiple vECUs
- Supports virtual integration testing very early in project

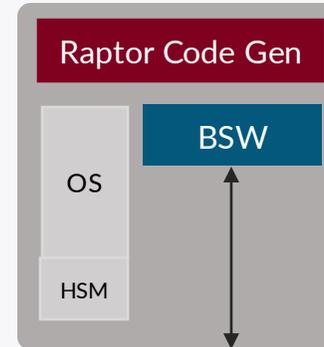


Signal + Bus Level  
Communication

### Level 3 vECU

Integrates ASW with production BSW

- Enables full simulation of everything north of the MCAL
- Supports full integration testing of ASW + BSW before hardware

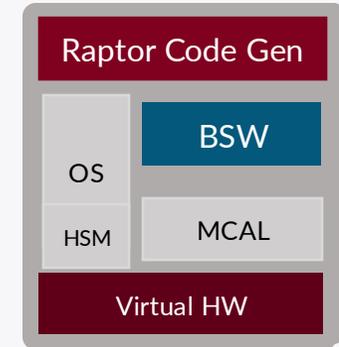


Bus Level  
Communication

### Level 4 vECU

Integrates ASW with production BSW and true hardware simulator

- Simulates execution of full production software stack on a virtual representation of the hardware



Bus Level  
Communication

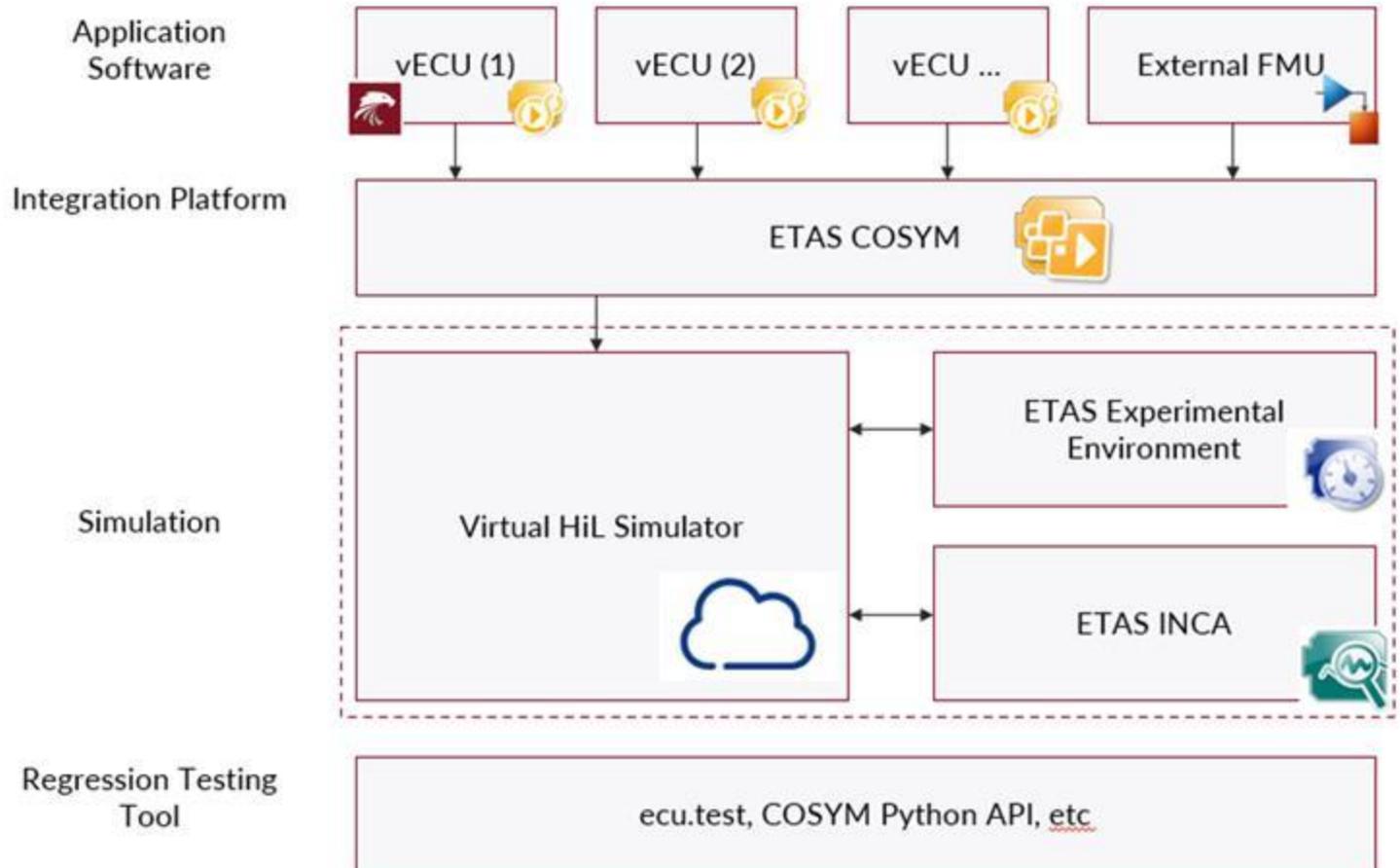


# Solving using Raptor ECU Virtualization Toolbox

Versatile ETAS software platform, easily accessible through “single-button build” workflow

Integrate existing applications into virtual workflow

- Simulation of complex vehicle networks
- Open platform for integration of controllers and plant models
- Flexible automated workflows



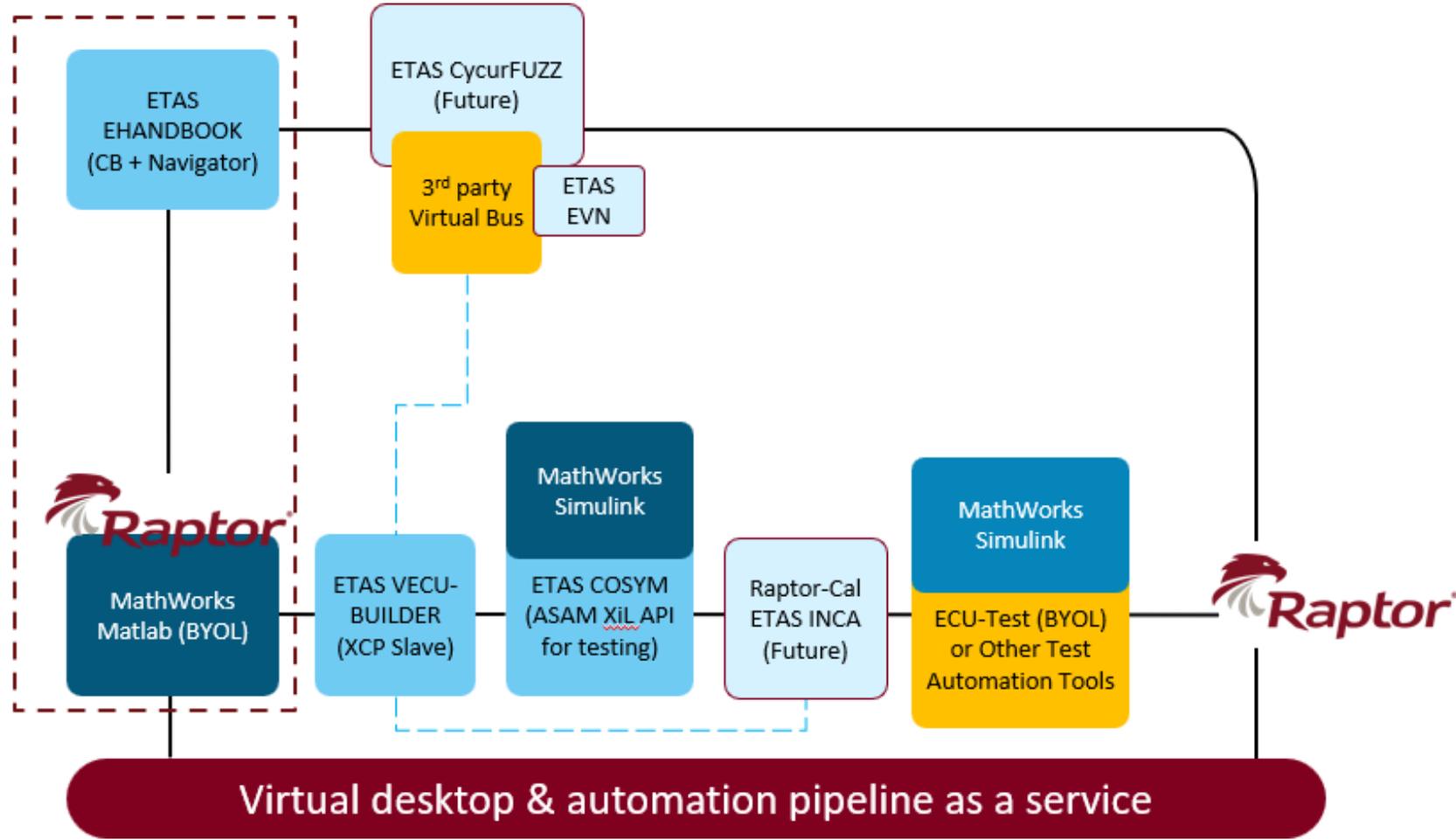
# Agenda



- Challenges with HIL
- **Raptor Cloud**
- Raptor-Dev Integration
- ETAS Toolchain
- Virtual HIL Use-Cases



# Raptor Cloud



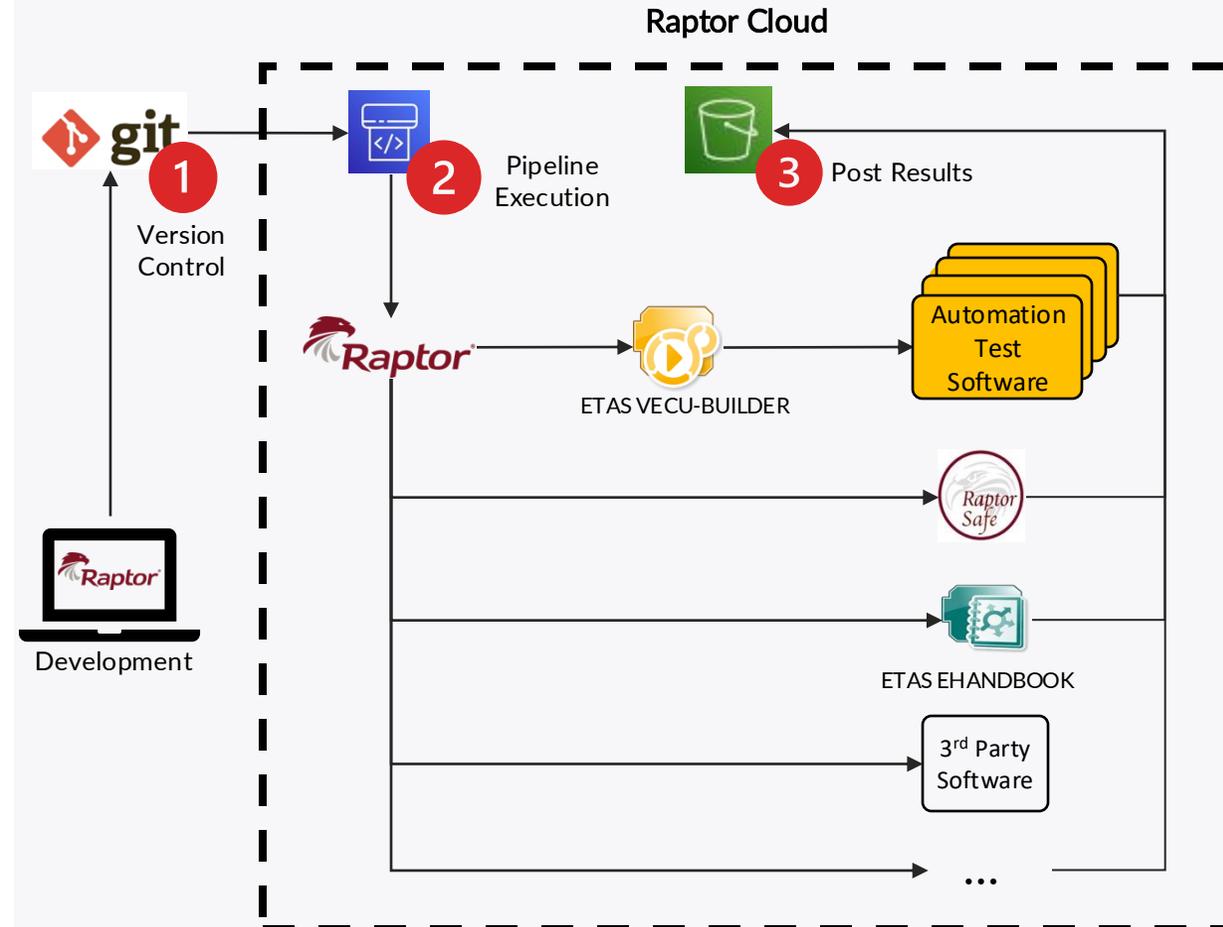
# Turnkey CI/CD Pipeline

## Raptor Cloud can build, test, and analyze your software, allowing you to focus on results

Process can be triggered from activity in your existing GIT repository

- Raptor-Dev will build and virtualization your ECU(s) in preparation for testing
- Pipeline is flexible to support third-party test tools that may already be in place
- Raptor-Safe will analyze your source code, stack consumption, and execution time
- You have the confidence in knowing your changes passed rigorous testing and analysis w/o expending effort

The pipeline structure is flexible to incorporate with a wide range of third-party products you may already be using



# Scalable Test Execution in the Cloud

## Scalable cloud resources to fit any computation and cost requirements

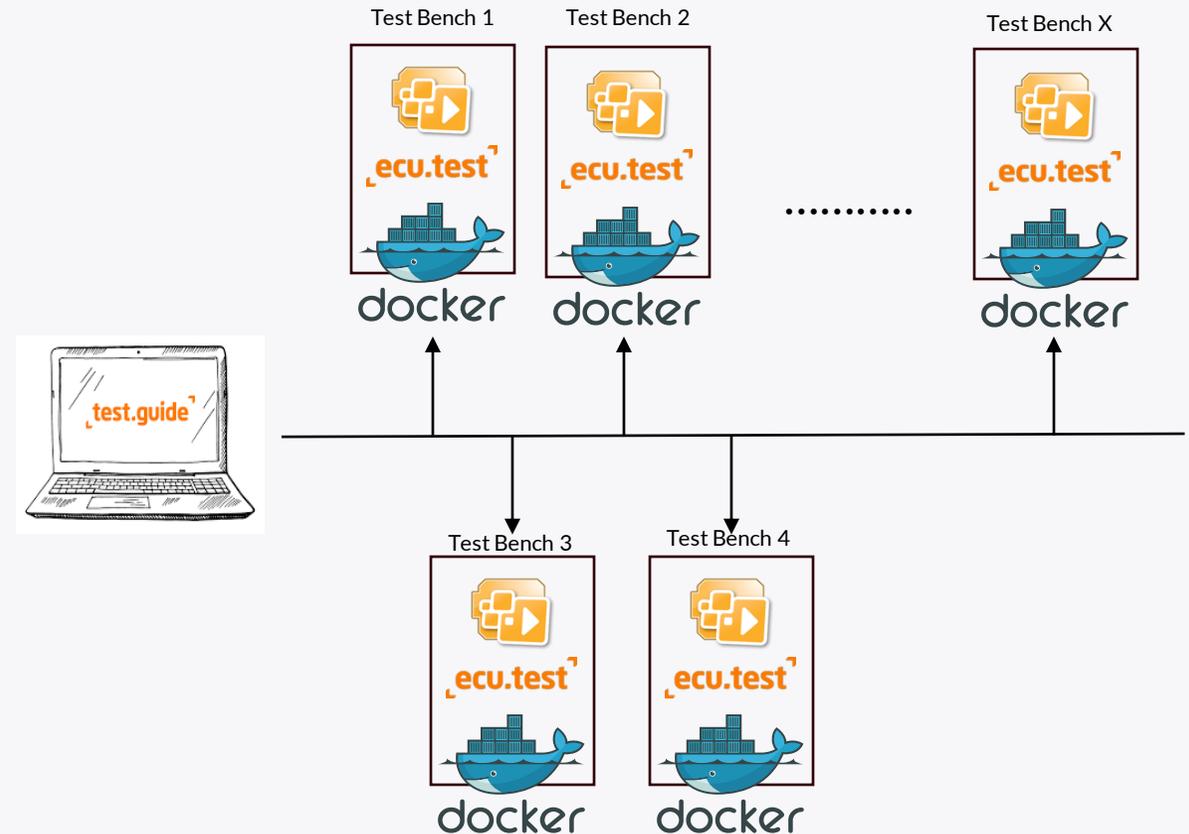
### Flexibility of AWS Computation Resources

- Scale up/down computation resources
- Allocate numerous instances of resources

### Controlled repeatable execution environments with Docker containers

- Identical test and development environments
- Confidence in reproducibility of issues in cloud and desktop

### Controlled facilitate test coordination through tracetronic test.guide and ecu.test



# Agenda



- Challenges with HIL
- Raptor Cloud
- **Raptor-Dev Integration**
- ETAS Toolchain
- Virtual HIL Use-Cases

new eagle



RAPTOR®  
INNOVATION  
SUMMIT 2025

# Raptor Virtual Target

## Create New Eagle virtual ECUs by adding a single block to your model

Initial support will be for Level 2 vECUs featuring:

- CAN, ETH, and LIN Virtualization
- I/O Virtualization
- XCP Calibration
- ATS/Soft Real Time Simulation

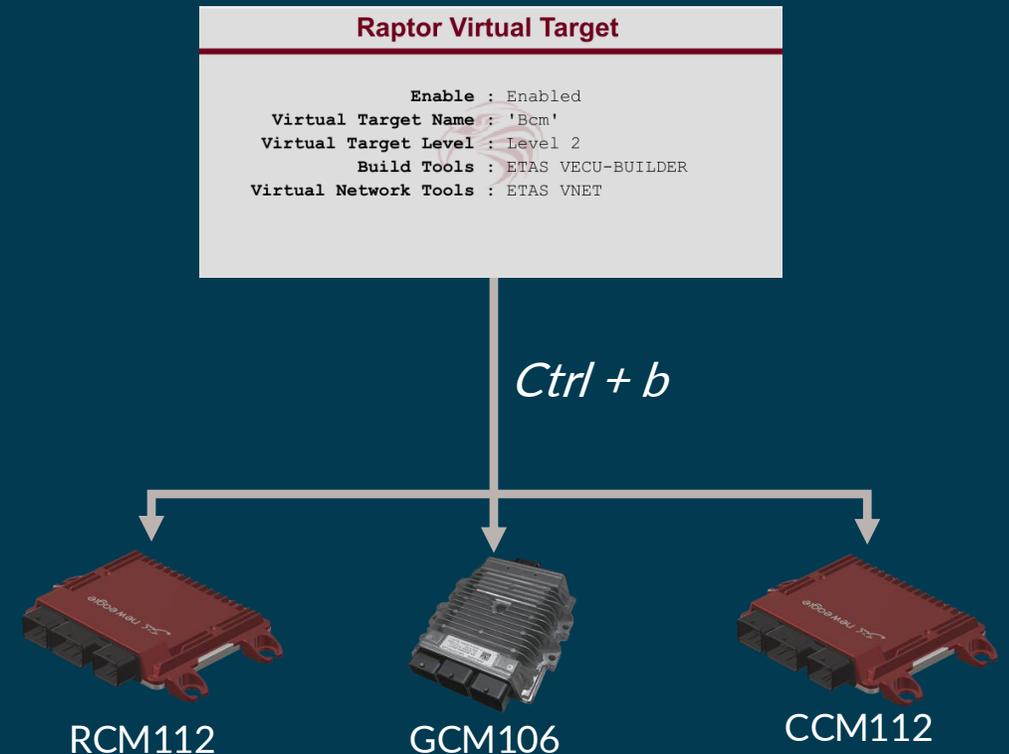
Initially supported for RCM112 and GCM106 targets

- Support for additional targets will be continuously added
- Future updates will include support for Level 3 vECUs

Available from Raptor 2025b!



## *A Single Block In Your Raptor Model*



# Raptor-Dev Configuration

The screenshot displays the Simulink Raptor-Dev configuration environment for a component named `gcm106_virtual_target`. The interface includes a top menu bar with tabs for SIMULATION, DEBUG, MODELING, FORMAT, APPS, and C CODE. Below the menu is a toolbar with icons for Custom Target, Quick Start, C/C++ Code Advisor, Settings, Code Interface, Build, View Code, Verify Code, and Share. The main workspace shows a block diagram with several configuration blocks:

- Raptor Advisor**: A panel on the left showing a list of checks and their status, such as "Raptor checks: 182/182" and "DataLog checks: 18/18".
- ECU Definition**: A central block containing ECU details like "ECU: STM32F711-106-248T", "Default USB refresh: Enabled", "UDS Address: 0x3000", "UDS Tool Address: 0x0300", "ECU Name: gcm106\_virtual\_target", "Cell Location: ECU300N", and "Compiler: Tricore Developer...".
- CAN Definition**: Three blocks defining CAN bus parameters, including "CAN Bus", "Hardware Resources", "Baud Rate", "Tx Queue", and "Rx Queue".
- XCP Protocol Definition**: A block defining XCP protocol settings, including "CAN Bus", "XCP Name", "Broadcast", "Receive", "Transmit", "ECU ID Storage", "ECU ID Storage", "TX ID Storage", "TX ID Storage", "Max Lists per ID", "Security", and "Access".
- Timed Trigger**: A block labeled "Foreground" with a "Name" field set to "Timer@Foreground".
- Power Up\_Down**: A block at the bottom left of the diagram.

The bottom status bar shows "Code Mappings - Component Interface", "Ready", "52%", and "FixedStepDiscrete".

# Agenda



- Challenges with HIL
- Raptor Cloud
- Raptor-Dev Integration
- **ETAS Toolchain**
- Virtual HIL Use-Cases

new eagle



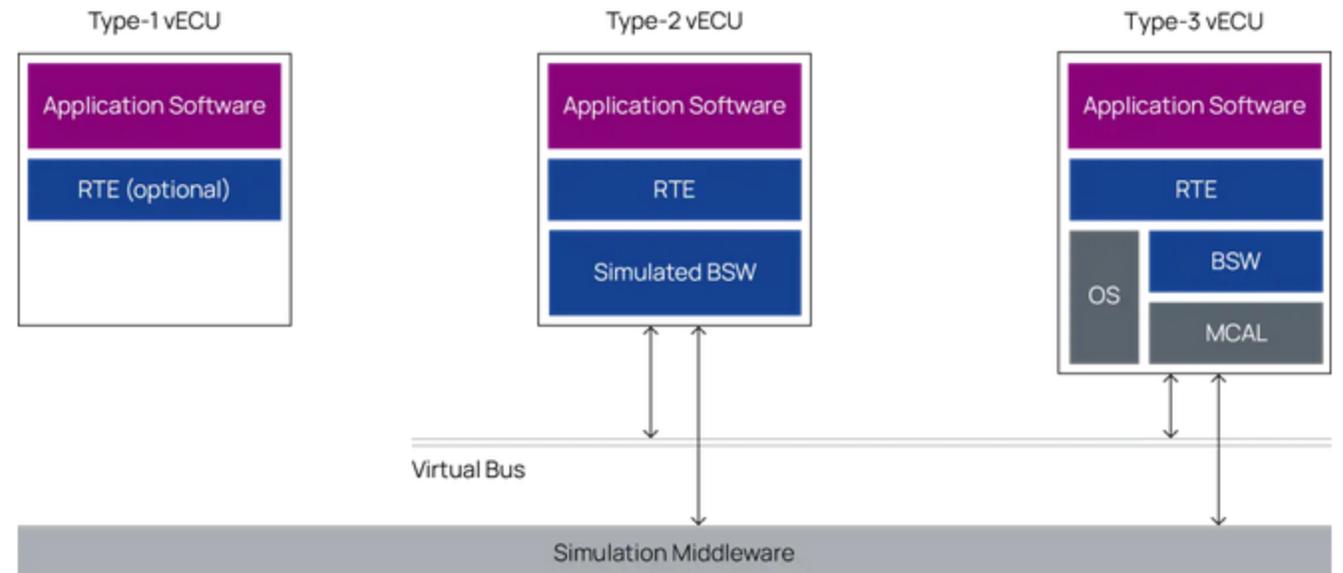
RAPTOR®  
INNOVATION  
SUMMIT 2025

# ETAS VECU-BUILDER



## Generates FMU from Raptor Code-Generation

- Fully integrated into Raptor-Dev, vECU ready to be built with no additional development time
- Type-2 vECU initially supported by Raptor
- Communication at signal and bus network levels
- Highly portable FMU format to integrate into existing toolchains

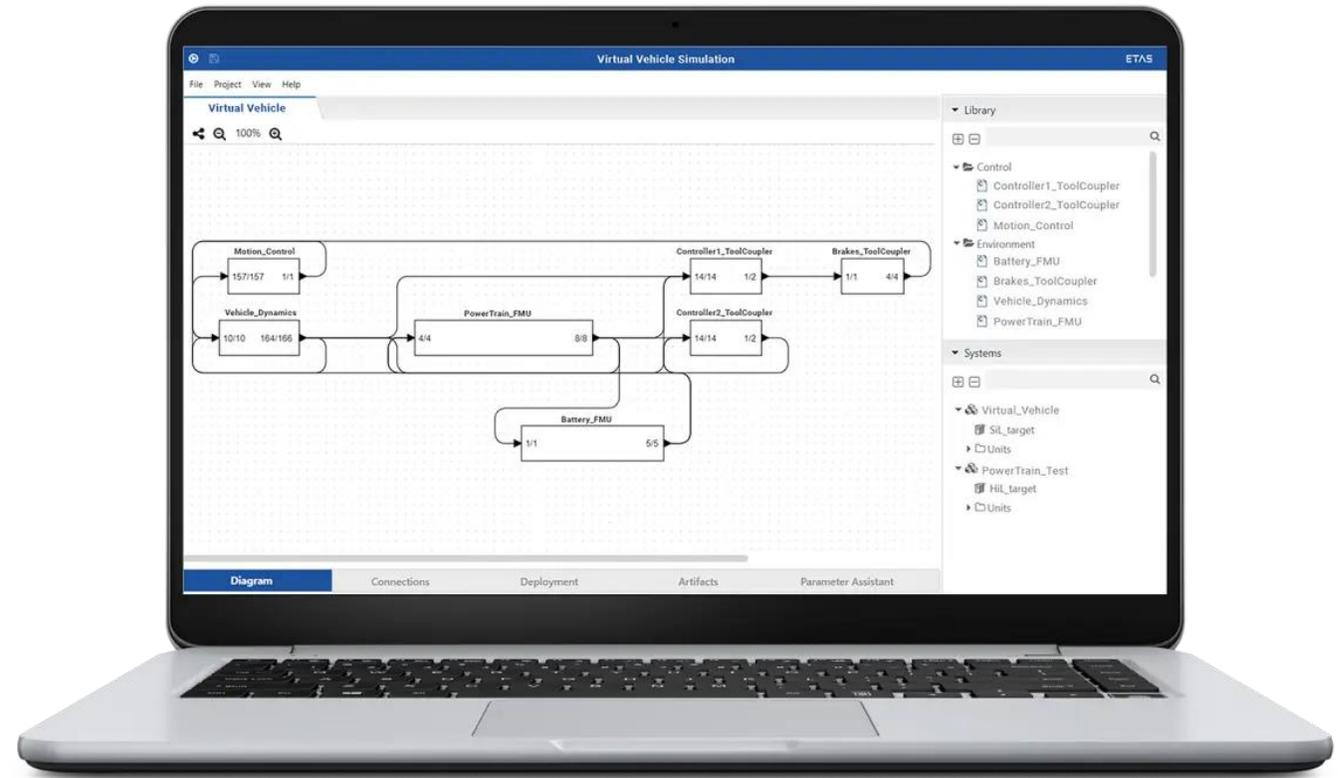


# ETAS COSYM



## Co-Simulation and Integration Platform

- Functional testing early in development cycle
- Integration with MATLAB/Simulink, C code, or other FMUs
- ETAS Virtual Network Integration
- Simulate multiple vECUs
- Real Soft-Time or Adaptive Time Simulation (ATS)
- Calibration (XCP) available vECU





# ETAS Virtual Network (vNET)

## Virtualized Automotive Networks

### CAN

Simulate high-speed CAN and CAN-FD networks

- Support for CAN and CAN-FD
- Extended and standard CAN frames
- Bus arbitration based on frame IDs
- Configurable baud rates

### LIN

Simulate Master and Slave nodes in complex LIN networks

- Supports typical baud rates
- Supports Master/Slave mode
- Standard LIN frame sizes

### Ethernet

Simulate high bandwidth Automotive Ethernet networks

- Standard Ethernet frames with standard headers
- VLAN header extension with VLAN ID, priority
- Supports multiple Ethernet networks, with multiple network nodes, with unique MAC addresses

### Additional Features

Expand testing capabilities beyond data logs

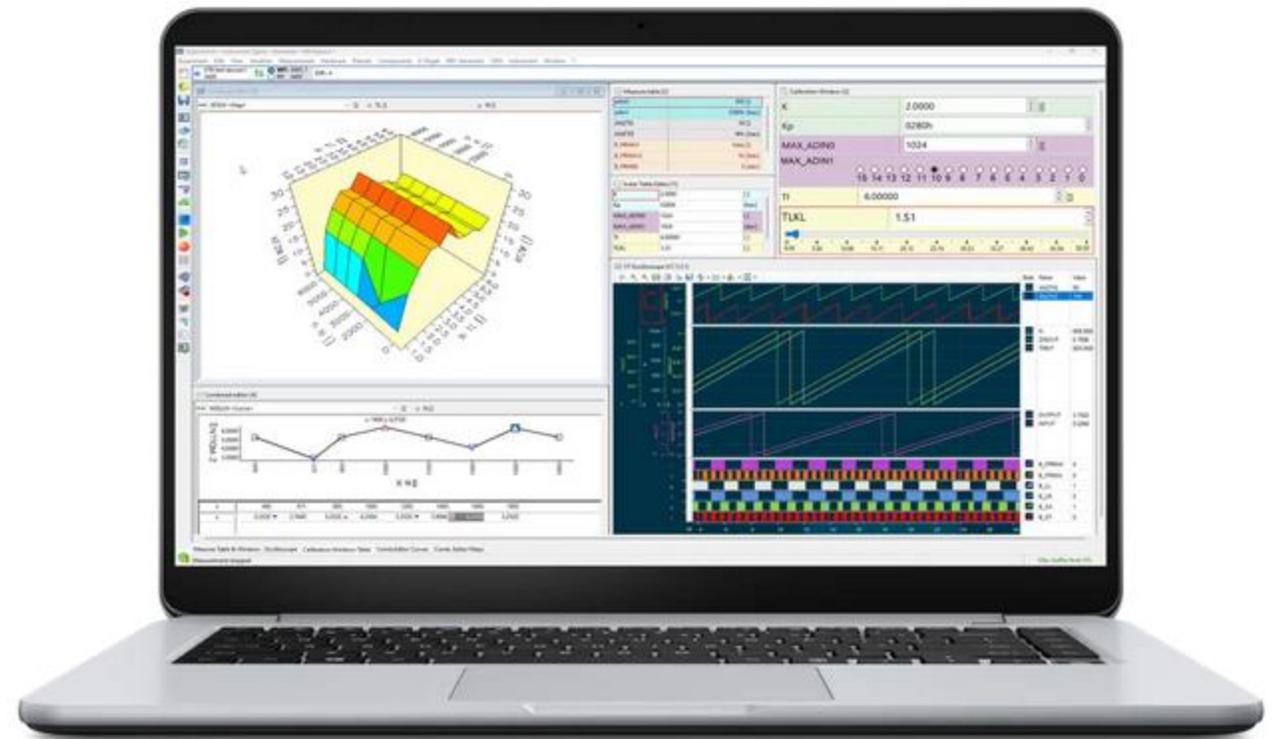
- Restbus Simulation via common formats such as DBC files
- Define custom C Code for Restbus module behavior
- Connectivity to ETAS INCA
- Self-reception capable (loop-back)

# ETAS INCA



## Tools for measurement, ECU calibration, and diagnostics

- Fully capable of support virtual ECU through calibration and CAN bus monitoring
- Replay recorded data (MF4, MATLAB, and other standard formats)

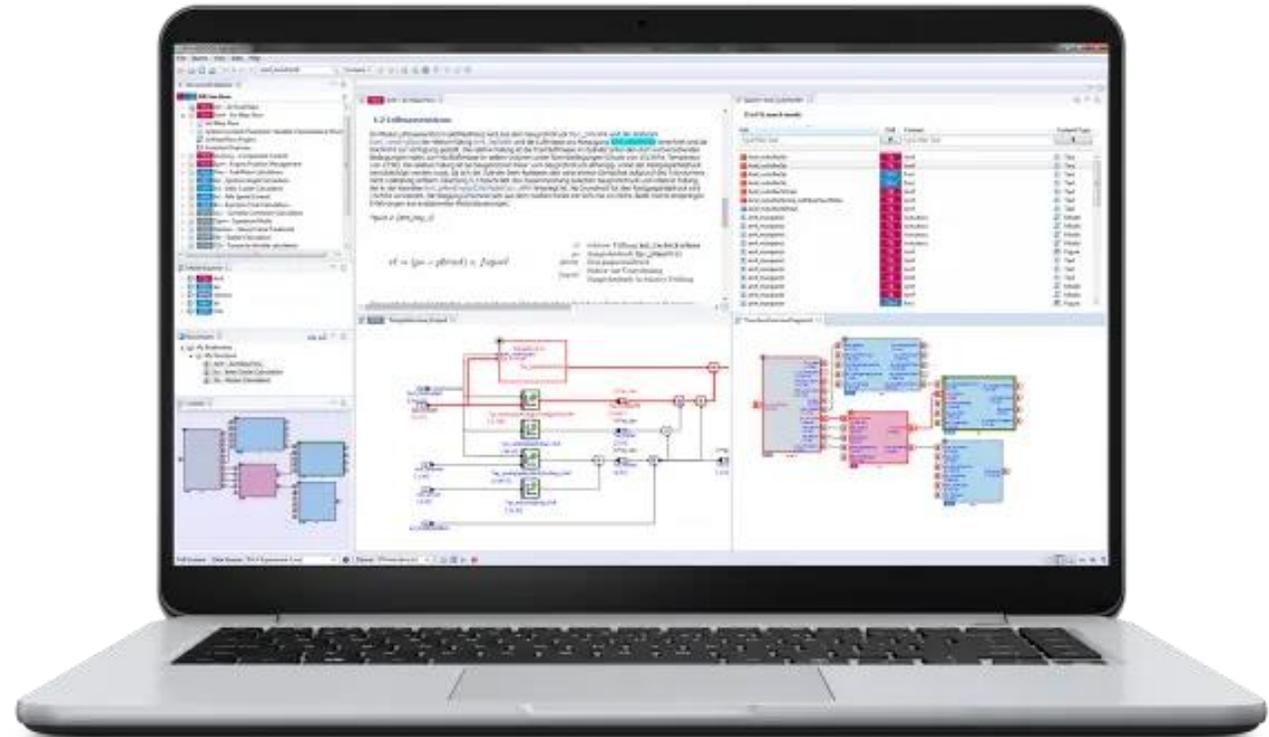


# ETAS E HANDBOOK



## Interactive lightweight documentation tools

- Generate interactive documentation from Raptor-Dev models
- Navigate Raptor-Dev model outside of MATLAB/Simulink environment
- Connect to INCA for interactive calibration



# Agenda



- Challenges with HIL
- Raptor Cloud
- Raptor-Dev Integration
- ETAS Toolchain
- **Virtual HIL Use-Cases**

new eagle

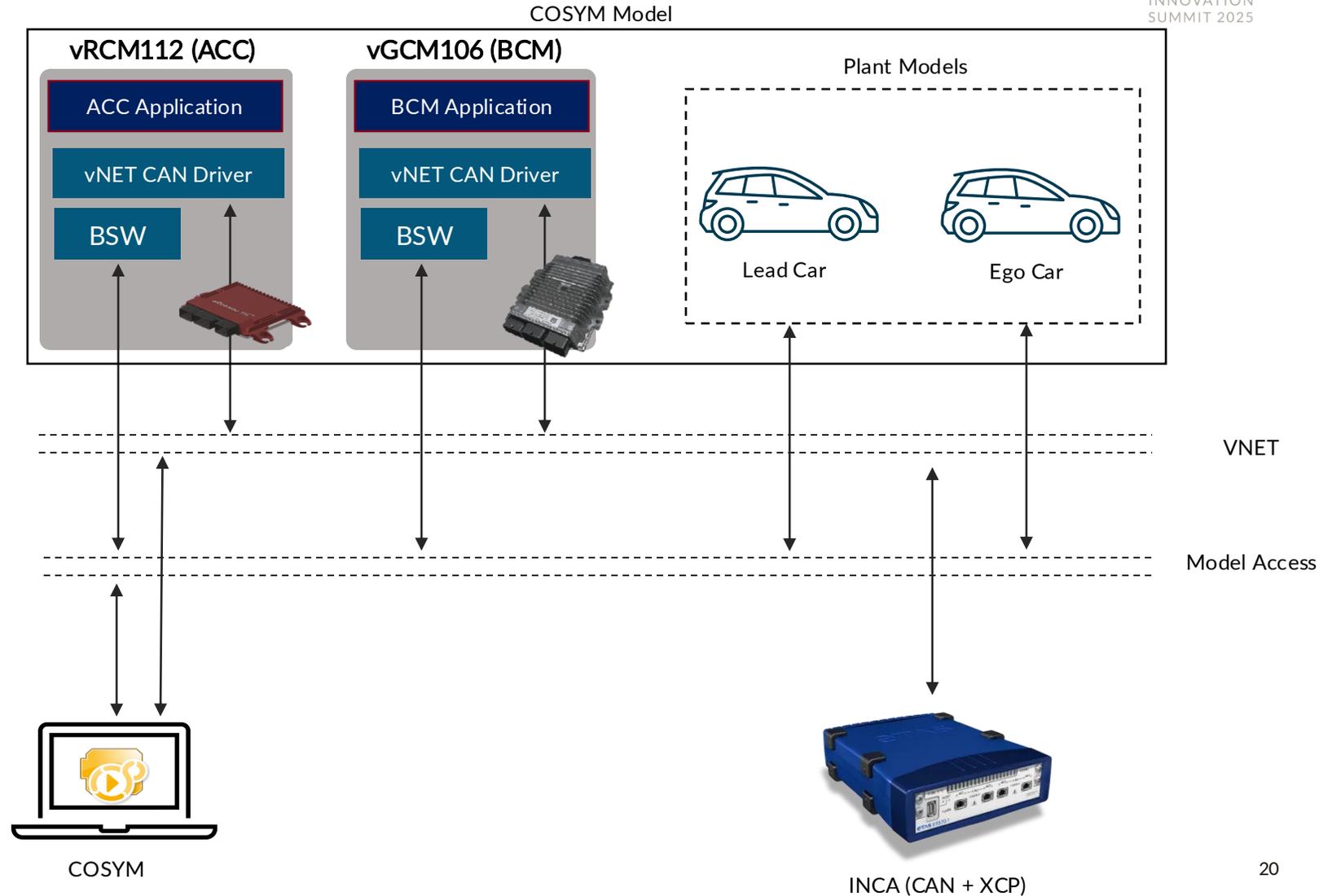


RAPTOR®  
INNOVATION  
SUMMIT 2025

# Virtual HiL Use-Case



- Adaptive Cruise Control (ACC)
- Body Control Module (BCM)

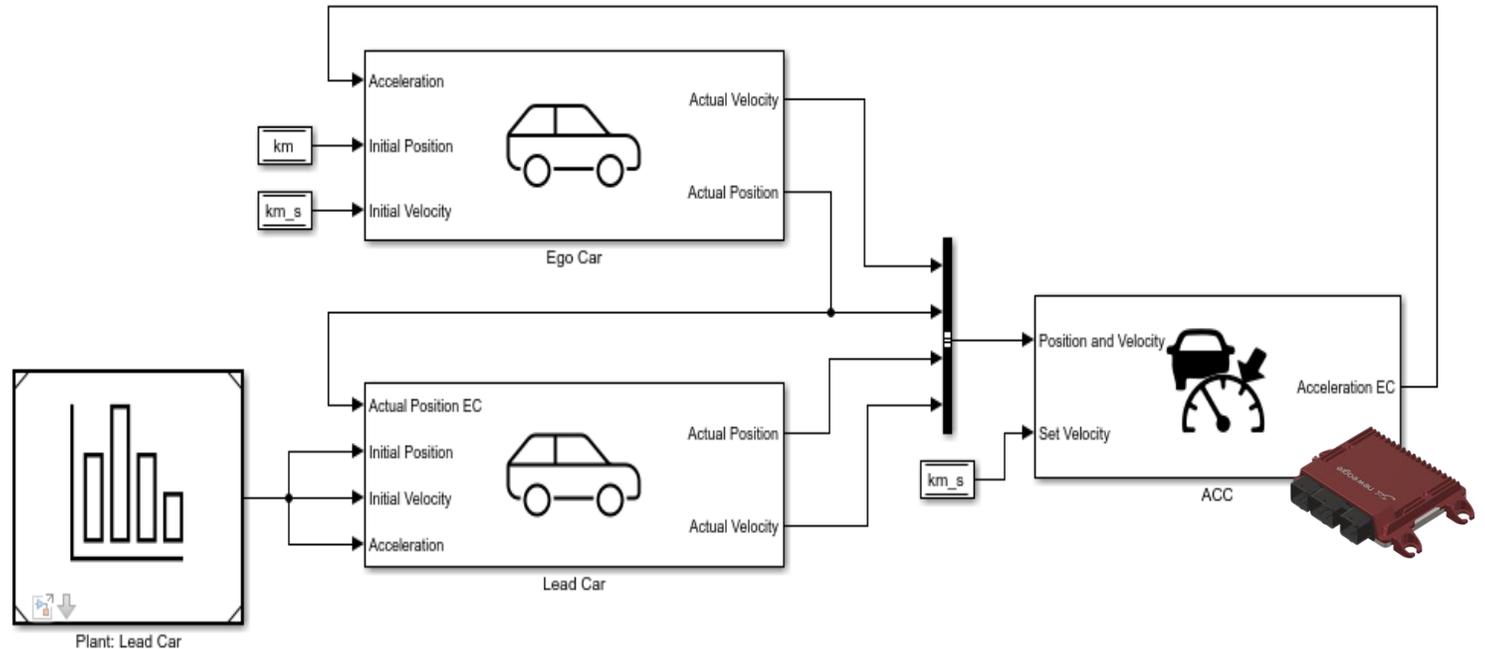




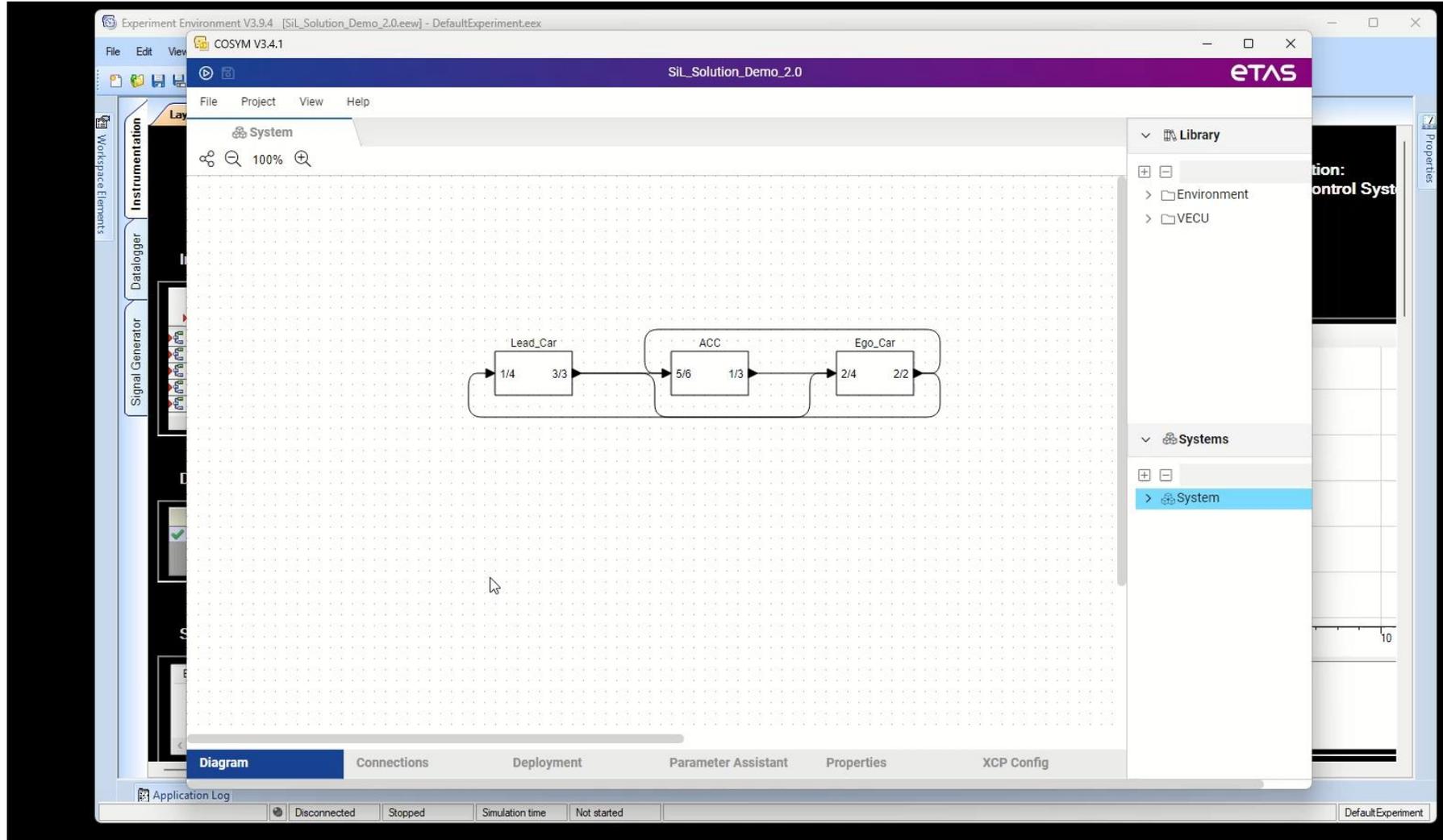
# Adaptive Cruise Control (ACC) Use Case

## Virtual RCM112

- Controller logic, with vehicle simulation for Follow and Lead Cars
- Control based on relative distance between Follow and Lead Car
- Calibratable gains for acceleration of Follow Car



# Video Demo ACC

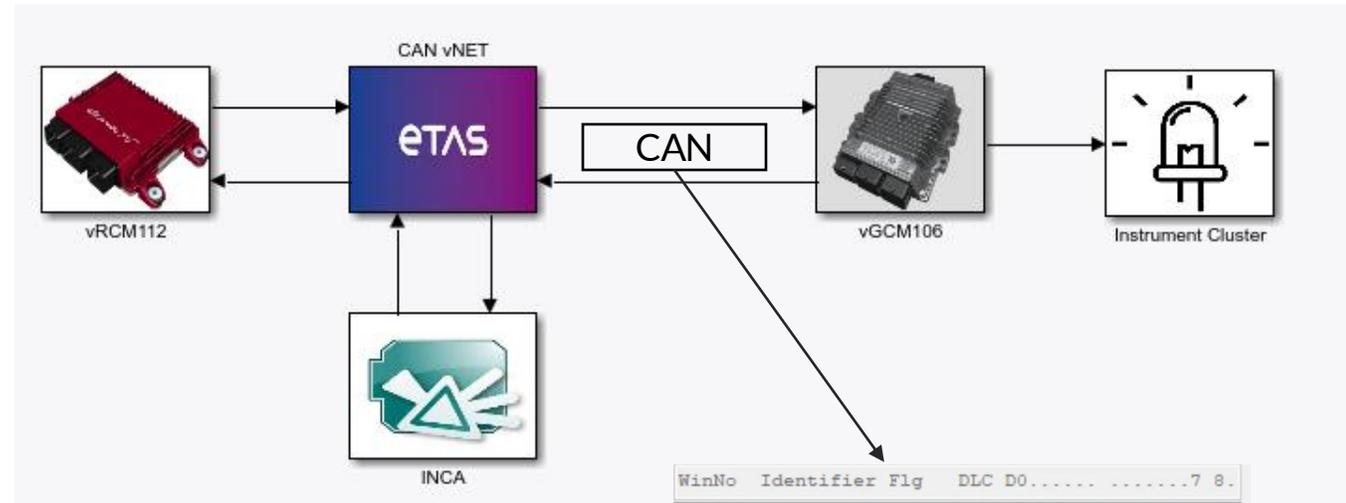




# Body Control Module (BCM) Use Case

## Virtual GCM106 (M2-702)

- vRCM112 commands BCM (GCM106) via CAN vNET
- Simulated I/O of BCM
- Monitor calibrations of BCM via XCP/CAN (INCA)

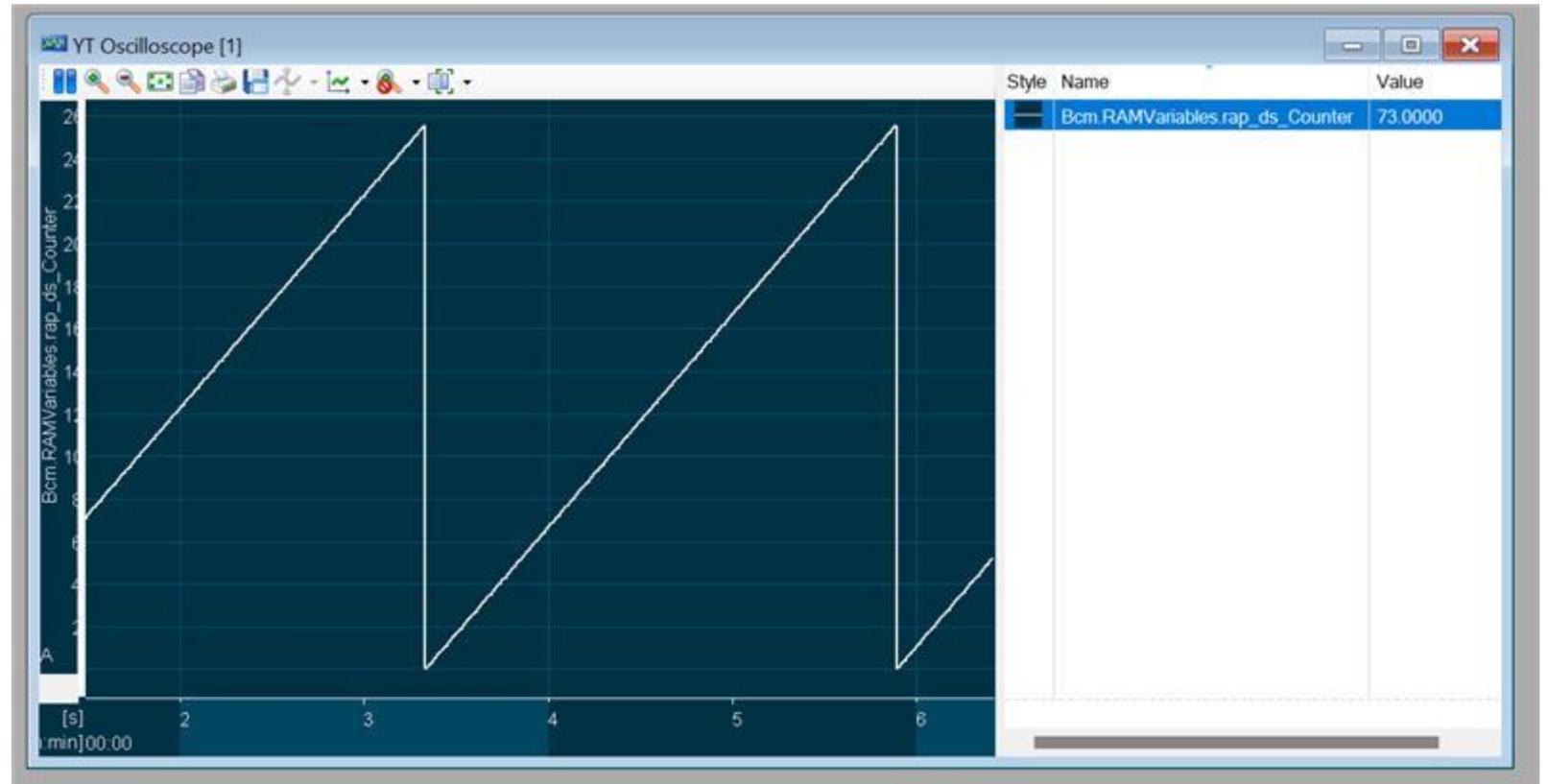


WinNo	Identifier	Flg	DLC	D0.....	.....7	8.
CAN 1	0CF00400	X	8	607D8450	1400F084	^
CAN 1	0CF00400	X	8	607D8450	1500F084	
CAN 1	0CF00400	X	8	607D8450	1600F084	
CAN 1	0CF00400	X	8	607D8450	1700F084	
CAN 1	0CF00400	X	8	607D8450	1800F084	
CAN 1	0CF00400	X	8	607D8450	1900F084	
CAN 1	0CF00400	X	8	607D8450	1A00F084	
CAN 1	0CF00400	X	8	607D8450	1B00F084	
CAN 1	0CF00400	X	8	607D8450	1C00F084	
CAN 1	0CF00400	X	8	607D8450	1D00F084	
CAN 1	0CF00400	X	8	607D8450	1E00F084	
CAN 1	0CF00400	X	8	607D8450	1F00F084	
CAN 1	0CF00400	X	8	607D8450	2000F084	
CAN 1	0CF00400	X	8	607D8450	2100F084	
CAN 1	0CF00400	X	8	607D8450	2200F084	



# Calibrate vECU via INCA (XCP)

- Calibrate vECU during development
- Read/Write calibrations during test execution
- Store calibrations in vECU (NV)
- Monitor Virtual CAN Bus during test execution



# Video Demo BCM



The screenshot displays the Experiment Environment V3.9.4 interface. On the left, a 'Measure Window [1]' shows a table of variables and their values. The 'Reset' variable is highlighted with a red border. The main workspace, titled 'ACC BCM Control', contains several data display groups:

- Group8: HSO1 DC (0), HSO1 Period (100000)
- Group6: Can Rx Msg Age (38), Can\_Rx\_Msg\_Led (0), Counter (64)
- Group9: CAN Read BCM Counter (24.000)
- Group5: DG1\_LED (False)

The bottom of the interface features an 'Application Log' with the following entries:

Time	Component	Message
11:31:52	SIL SignalSource	Download inport settings complete. (2 ms)
11:31:57	SIL SignalSource	Simulation mode is set.
11:31:57	SIL SignalSource	Simulation started.

The status bar at the bottom indicates 'Connected', 'Running', 'Simulation time: 0:03:01 [hh:mm:ss]', '1.01 X faster than real-time', and 'DefaultExperiment'.

Questions?



Presented by:

Michael Callaway,  
Senior Software Engineer,  
New Eagle



# Learn More in the NEST

Scan the QR code to access related resources, technical content,  
and additional insights.



new eagle



RAPTOR®  
INNOVATION  
SUMMIT 2025