



# Eclipse SDV

Modernizing Automotive Development Toolchains with  
Metadata Driven Orchestration Blueprint

Gianluca Vitale, Hannes Fuchs, AVL  
Haishi Bai, Filipe Prezado, Microsoft

March 19-20<sup>th</sup>, 2024

# Toolchain Challenges

- Different programming languages and tools
- Different operating systems and runtime environments
- Different application package formats
- Different network topologies
- Different scopes and restrictions
- Different hardware profiles
- Different management toolchains
- Different monitoring toolchains
- Different policies and security constraints

# We Need



**Composability:** We need a highly composable system so that artifacts and components from multiple sources can be selected and assembled in various combinations to create development, testing and simulation environments.



**Extensibility:** We need to ensure the system is easily extensible to continue to support the pace of innovation of software defined vehicle through the addition of new toolchain capabilities and functionality.

# Eclipse SDV.Dev

## Metadata Driven Orchestration Blueprint



### **GOAL:**

The ECLIPSE SDV.DEV Metadata Driven Orchestration Blueprint seeks to demonstrate and build the necessary open-source non-differentiating engineering building blocks to enable DevOps processes for end-to-end development, testing and validation, and improve efficiency, productivity, quality and security.

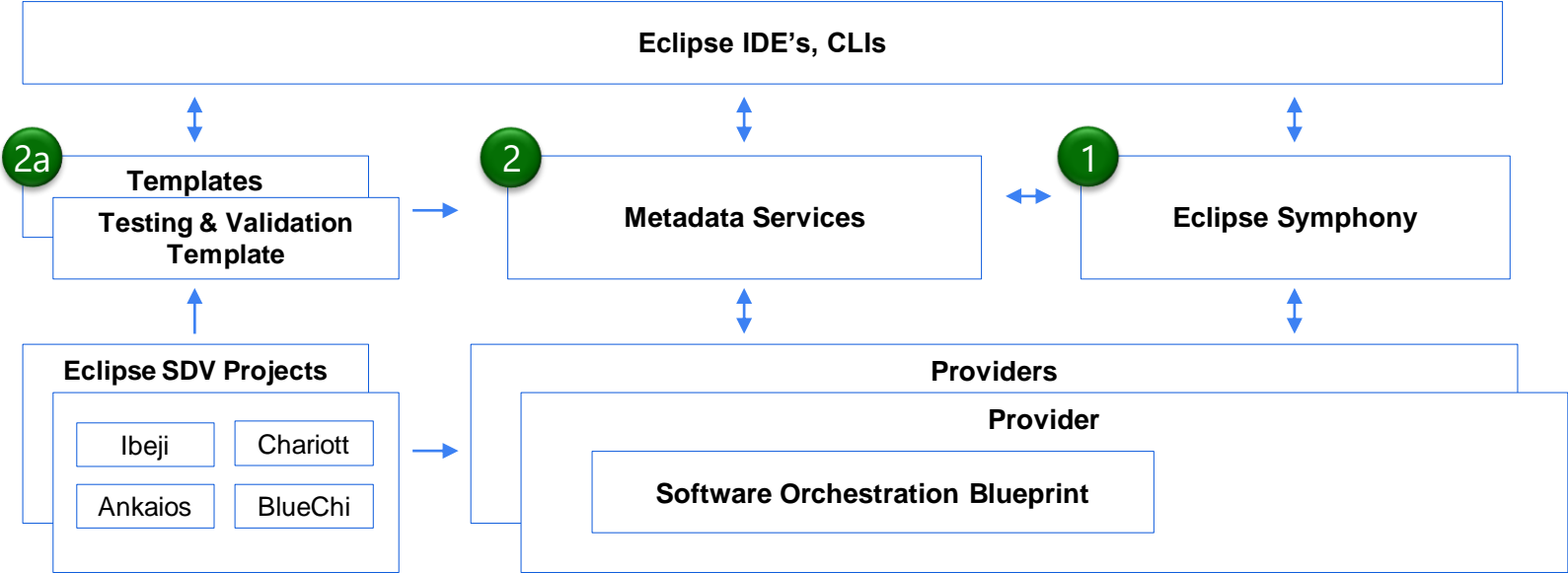
The initial submission will demonstrate the orchestration of a testing and validation process.



### **NON-GOAL (AGAINST):**

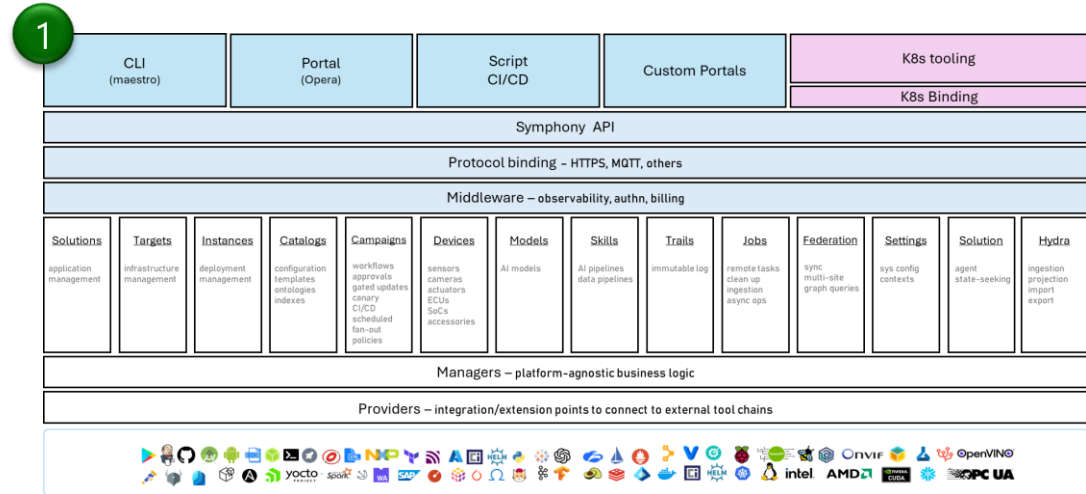
**Replace existing, well established and developed methods, toolchains and tools**

# Eclipse SDV.Dev Metadata Driven Orchestration Blueprint



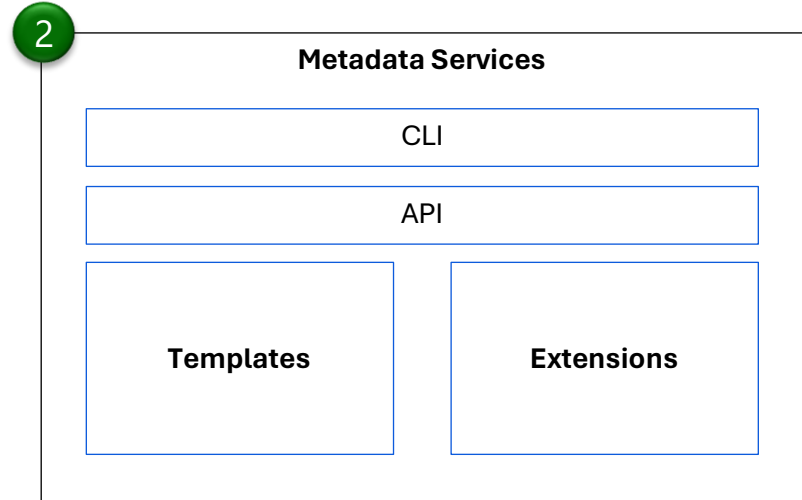
# Eclipse SDV.Dev Orchestration Services - Eclipse Symphony

- **Unified Object Model** - defines the units of orchestration and the target state for these units.
- **Orchestration API** - provides the orchestration capabilities that are state seeking, information graph and workflow.
- **Providers** - encapsulate and isolate platform-specific operations for each of the systems and devices that participate in the system orchestration and enable extensibility.



# Eclipse SDV.Dev Metadata Services

- Creates, manages and run templates for Software Defined Vehicle development and testing.
- Provides coherent artifact workflow that describes the full stack of solutions, test cases and targets
- **Templates** - describe, discover and assemble scattered artifacts into consistent views.
- **Extensions** - pluggable software components that lets you add custom commands to project artifacts into orchestrator unified object model.



# Eclipse SDV.Dev Metadata Services | Testing & Validation Template

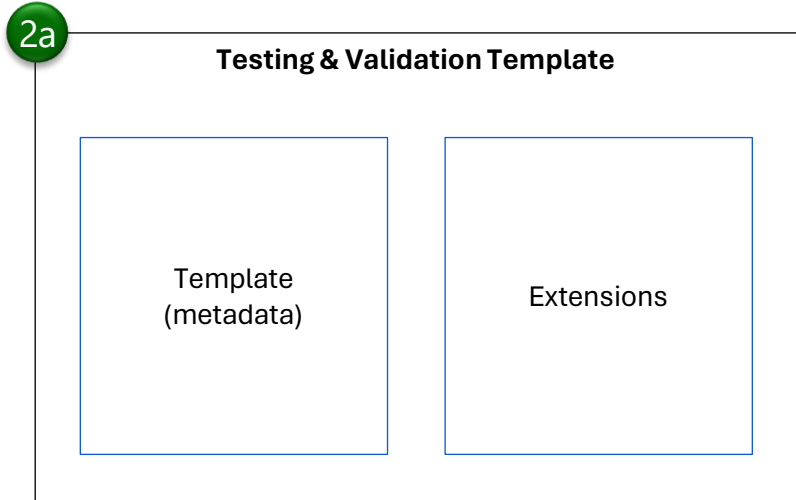
- **Case Study** - we will adopt the **Eclipse Symphony** and the **Metadata Services** in the **AVL DevOps Pilot** and do a **case study** on the impact and benefits.
- **Template (metadata)** – we will provide template examples for the configuration of **AVL Model.Connect™** in combination with **AVL CAMEO 5™ Functional Testing**



**CAMEO 5™**  
Functional Testing

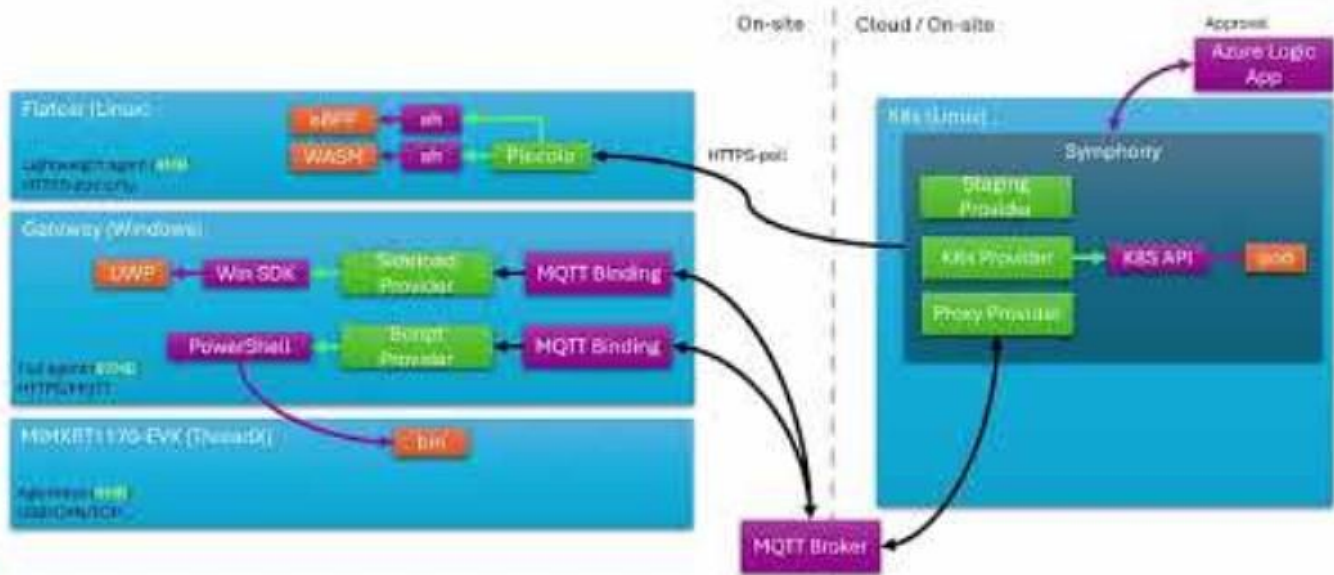


**Model.CONNECT™**  
Co-Simulation Platform





# Eclipse SDV.Dev Metadata Driven Orchestration Blueprint Prototype



# Next Steps

- Community discussions

- Submit Blueprint Proposal



# Q&A!





# THANK YOU!

# Links

[Eclipse Software Orchestration Blueprint](#)

[Eclipse Symphony](#)

[Multi-OS App Deployment with Symphony](#)