# Hybrid System Simulation Standards

Martin Barnasconi (NXP)
Mark Burton (GreenSocs)

28 October 2021

accellera SYSTEMS INITIATIVE™

SYSTEMC™ EVOLUTION DAY
OCT 28, 2021 | VIRTUAL WORKSHOP

# Copyright Permission

- A non-exclusive, irrevocable, royalty-free copyright permission is granted by **NXP** and **GreenSocs** to use this material in developing all future revisions and editions of the resulting draft and approved Accellera Systems Initiative **SystemC** standard, and in derivative works based on the standard.

# Outline

- Hybrid System Simulation Initiative
  - Problem statement
  - Current status, standards identified so far
  - Next steps
- Potential role of SystemC
  - SystemC in a Hybrid Simulation environment
- Summary
- (Backup material)

# Problem: We’re living in Silos

# Problem statement

- Each industry is developing and standardizing its own system simulation environment and coupling mechanisms to simulate and model complex systems
- No standard available to address the common multi-domain aspects to connect the multiple systems together

**Avionics**

VISTAS / VHTNG, …

**Space**

SMP2, …

**Semi**

SystemC / TLM, CCI, IP-XACT, …

**Automotive**

OpenADx, OpenMCx, …

**Mechatronics**

FMI/FMU, …

# Hybrid System Simulation Initiative

- Bring industries together to discuss differences and commonalities between different standards
  - Users to better understand standards coverage
  - Standard experts to understand requirements, potential overlaps, and points of interaction
- Foster and initiate actions to improve and co-ordinate standardization
  - Foster and initiate a collaborative action to make standards evolve according to common industry needs
  - Enable cross-over between various standardization organizations

# Example: Hybrid System Simulation

Application

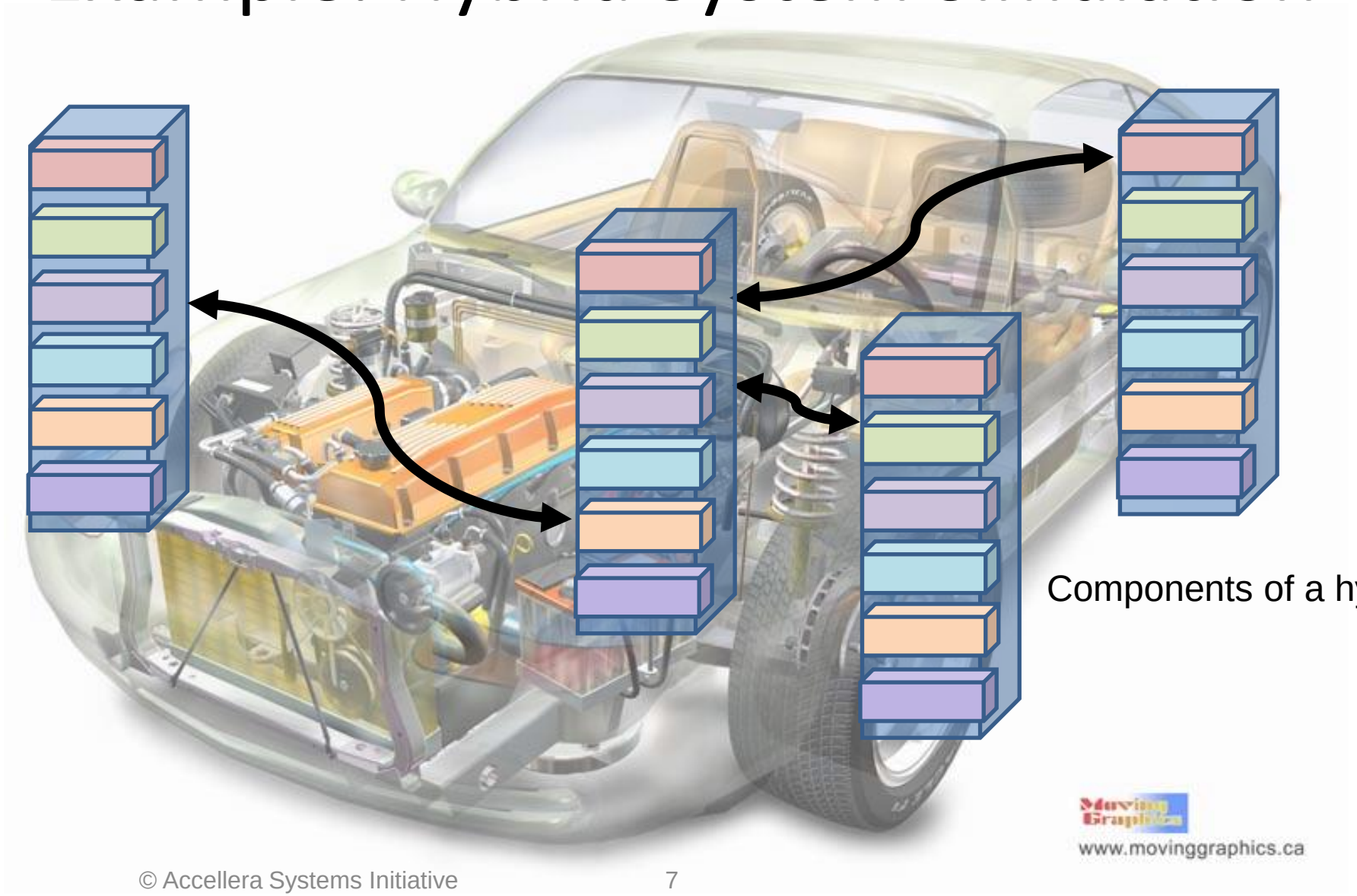
Middleware

O/S

Firmware

Hardware

Environment

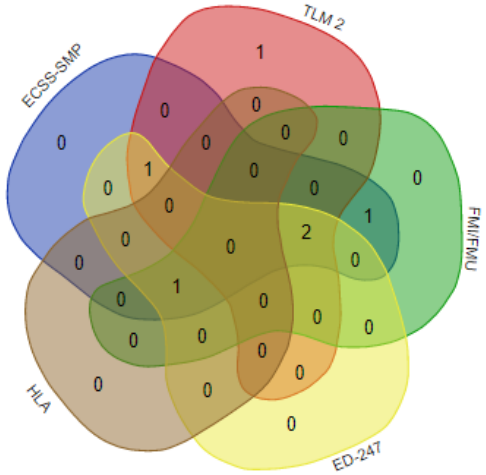


Components of a hybrid model

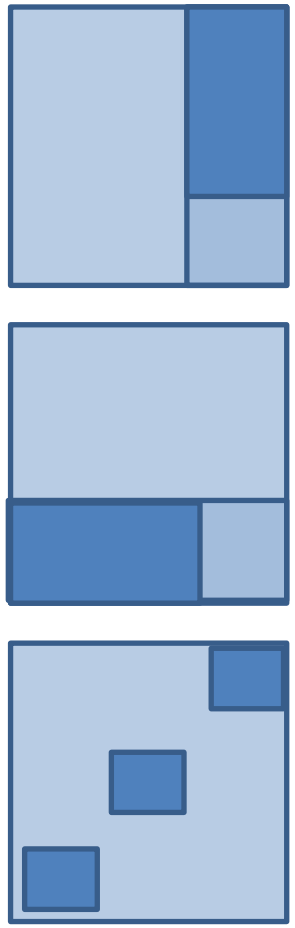
# Current status

- Workshop given at DVCon Europe 2020
  - Introducing diversity of standards, setting the problem space
  - Parties expressed interest to co-operate and align on interoperability and standardization needs
- Formation of a “Core team” to continue alignment
  - Industries involved: NXP, GreenSocs, Bosch, Airbus, Spacebel, IRT Saint-Exupery, Samares Engineering, Raytheon
  - Define goals and objectives, charter
  - Start inventory of existing standards and initiatives

| TBD | Column1 | Column2 | Purpose | Industrial domain |
|---|---|---|---|---|
| *(reserved column for company or standard list)* | *Link to standard(s)* | *Executable model / underlying* | *Simulation Modeling* | *Automotive Space Avionics* |
| IEEE 1666 (SystemC TLM-2) | https://standards.ieee.org/standard/1666-2011.html | yes | Simulation & Modeling; > model packaging gap | Avionics |
| ED-247 (VISTAS) | https://www.sae.org/publications/technical-papers/content/2018-01-1949/ | | Simulation > model packaging gap | Avionics |
| ED-247 (VISTAS) | https://www.sae.org/publications/technical-papers/content/2018-01-1949/ | | | |
| SMP | https://ecss.nl/standard/ecss-e-st-40-07c-simulation-modelling-platform-2-march-2020/ | | SMP defines: - The API a simulator shall present to model, but implementation is user defined. - The API a model shall have to | Space |
| OpenADx | https://openadx.eclipse.org/ | | Development Framework incl. Simulation | Automotive |
| HLA | https://www.sisostds.org/ProductsPublications/Standards.aspx | | | |
| SpaceFOM | https://www.sisostds.org/ProductsPublications/Standards.aspx | | | |
| DCP | https://dcp-standard.org/dcp/ | | equivalent to ED247 different domain | |
| FMU/FMI | https://fmi-standard.org/ | | Simulation | Avionics |
| Matlab/simulink | Non standard.... | | | |
| SysML & AADL | | | Modeling | Avionics |
| Capella | Modelling languages? UML etc? : (maybe look at what connections tc no | | | |
| | | | | This column needs |
| Industrial automation | | | | |
| Use case and application level standards | | | | |
| Opendrive/open scenario | | | | |
| What about other doma | Can we identify "one" per domain/calss of abstraction/domain.... E.g. matlab/simulink is one such example. | | | |



# Standards identified (so far)



- Domain-specific eco-systems of many different types of components (e.g. OpenADx)
- Ecosystems of common components types (e.g. SystemC/TLM, SMP2)
- Model ‘writing’
  - Model communication/Synchronisation (e.g. FMI/FMU)
  - Model/System construction (e.g. IP-XACT)
  - Model Debug, tool interaction (e.g. SystemC CCI)
  - . . .

# *Intermezzo*: *Questions for YOU. . .*

- What simulation models do you aggregate?
  - Covering what subsystems?
  - Covering what type of simulations (hydraulic, electronic,…)?
  - Covering  what time-scale?
  - For what purpose?
  - With what constraints e.g. wrt real-time?
- What are the standards currently used?
- What issues are there with current standards (such as HLA, for instance) ?
- What is missing in existing standards
- Do we need new standards or to align existing standards and develop interoperability layers?

# Hybrid System Simulation Initiative

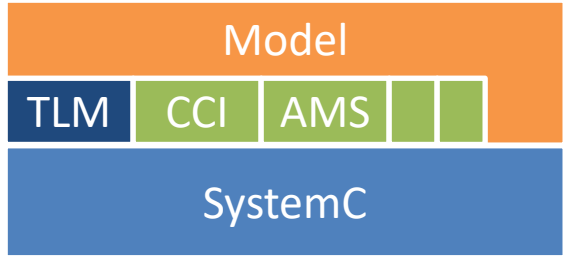
Establish inter-industry collaboration to improve the
**interoperability of product and environment simulation frameworks**
based on
co-ordination between **existing and new open standards**

# Next steps

- Conclude on goals and objectives for Hybrid System Simulation Initiative
- Finding an organization willing to host this initiative
  - Establish “official” Working Group and (legal) framework to encourage collaboration, exchange of ideas and technologies, etc.
- Agree on the (technical) approach to converge the different standards
  - Develop interoperability layers / adapters between existing standards, or
  - Define yet-another standard acting as ‘central framework’ enabling integration, or
  - …

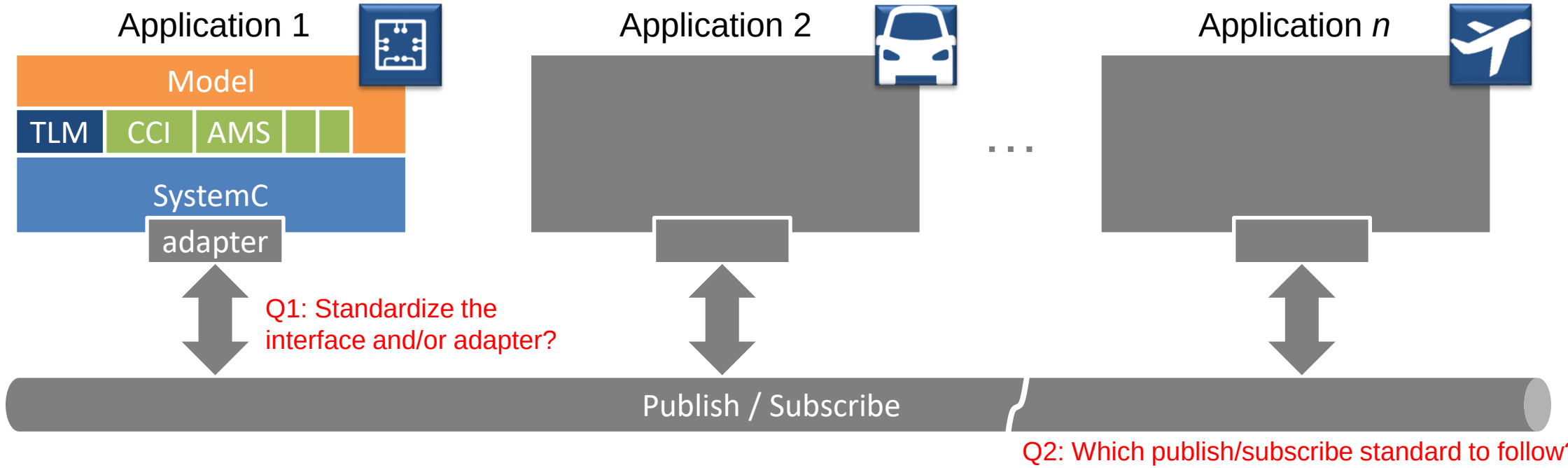
# The (potential) role of SystemC

- SystemC standard defines means to *model* Semi IP
- SystemC standard defines *execution semantics* (aka simulator) to create virtual prototypes
- SystemC standard defines *abstract communication* approach between IP based on TLM
- SystemC defines various *extensions* for e.g. model configuration (CCI), analog/mixed-signal (AMS), system verification (SCV, UVM-SystemC), etc.

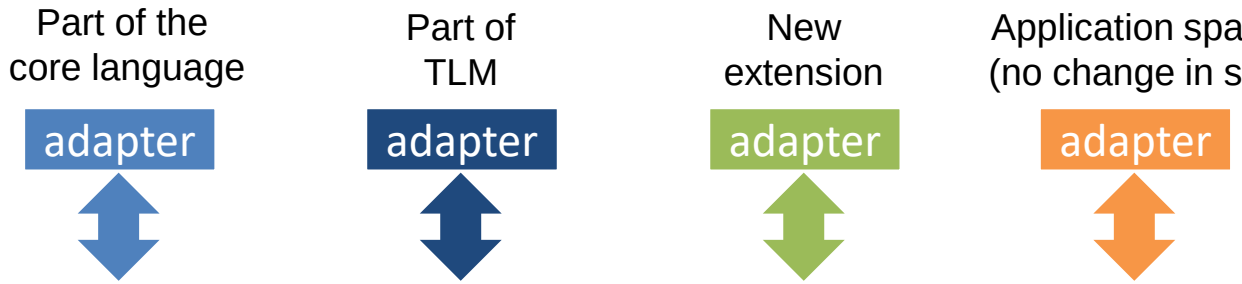



Should SystemC be one of the standards applied in such Hybrid System Simulation Environment?

# SystemC in a Hybrid Simulation env.




- Standardization options for interface and/or adapter:



# Summary

- Hybrid System Simulation Initiative – running for >1 year now
  - Core team conducted inventory of standards and approaches
  - Finding ways to become an ‘official’ working group hosted by governing body
  - Conclude on goals and objectives
- We are facing an organizational and technical challenge
  - Different industries have developed their own standards and simulation ecosystem over the last decades
  - How to best consolidate and align the various standards in a consistent hybrid system simulation environment
  - Define mechanisms how SystemC could be an integral part of such environment
- Please let us know if you like to contribute to this initiative!

# BACKUP MATERIAL

# Example: Hybrid System Simulation

Application

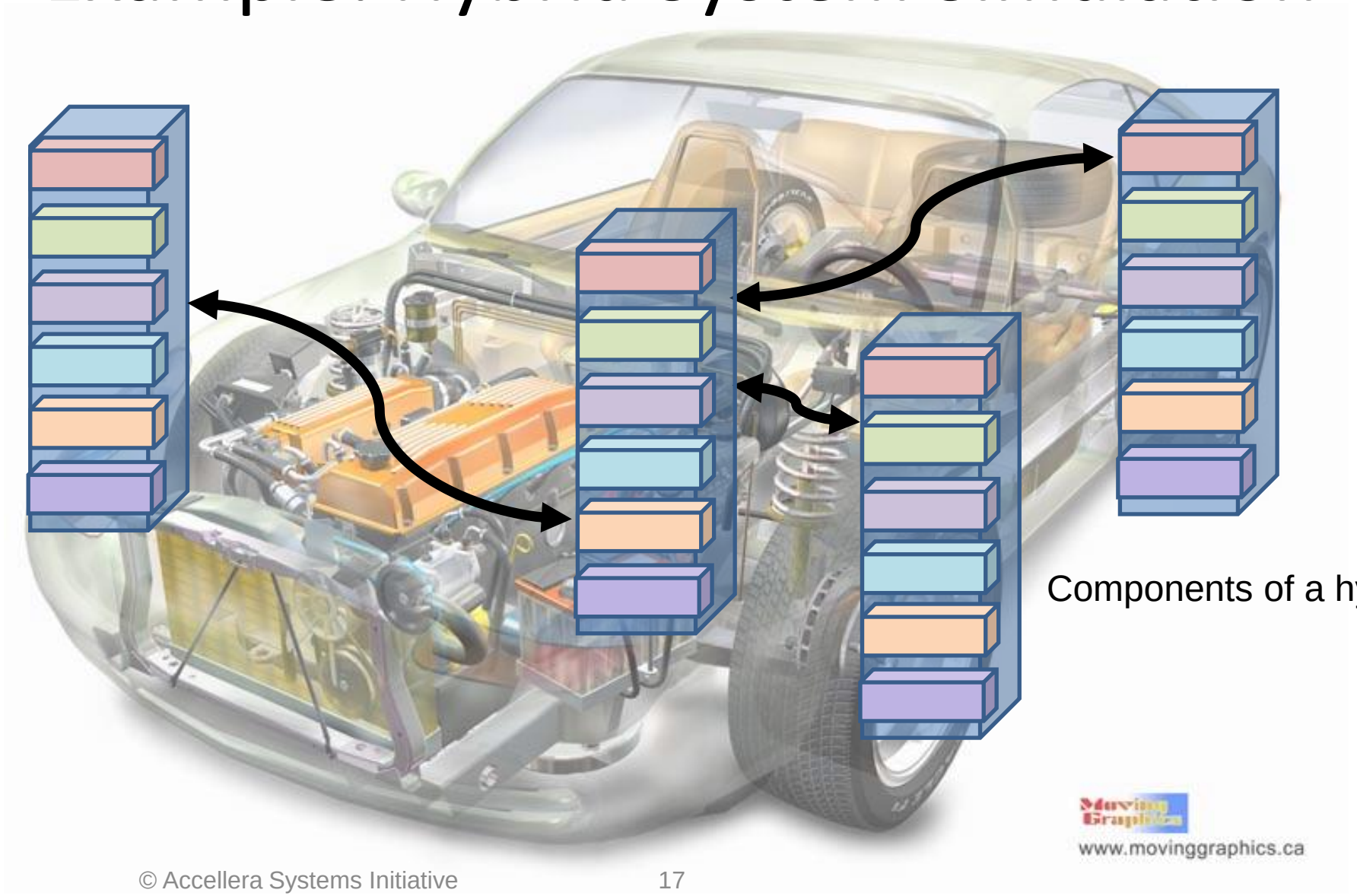
Middleware

O/S

Firmware

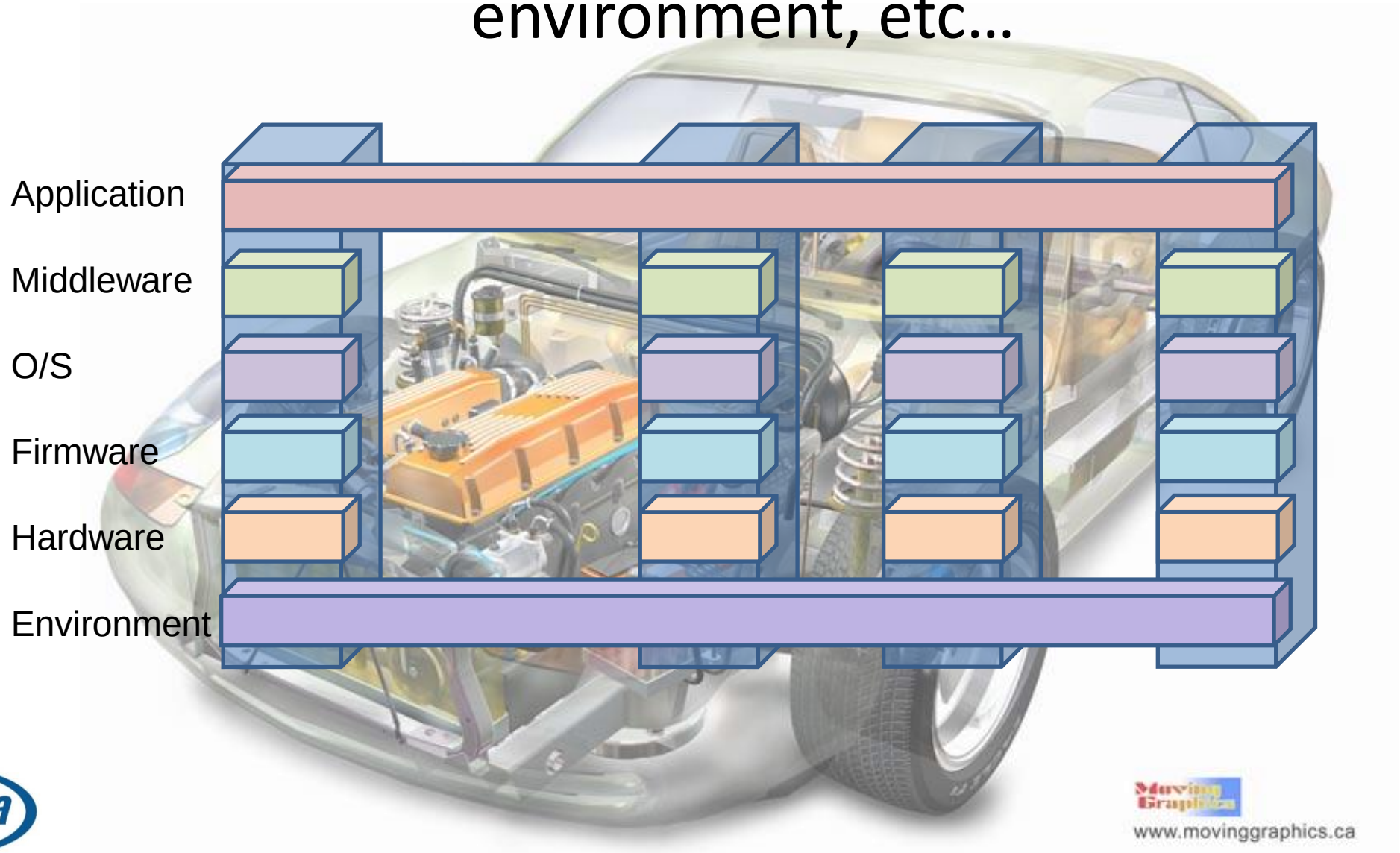
Hardware

Environment



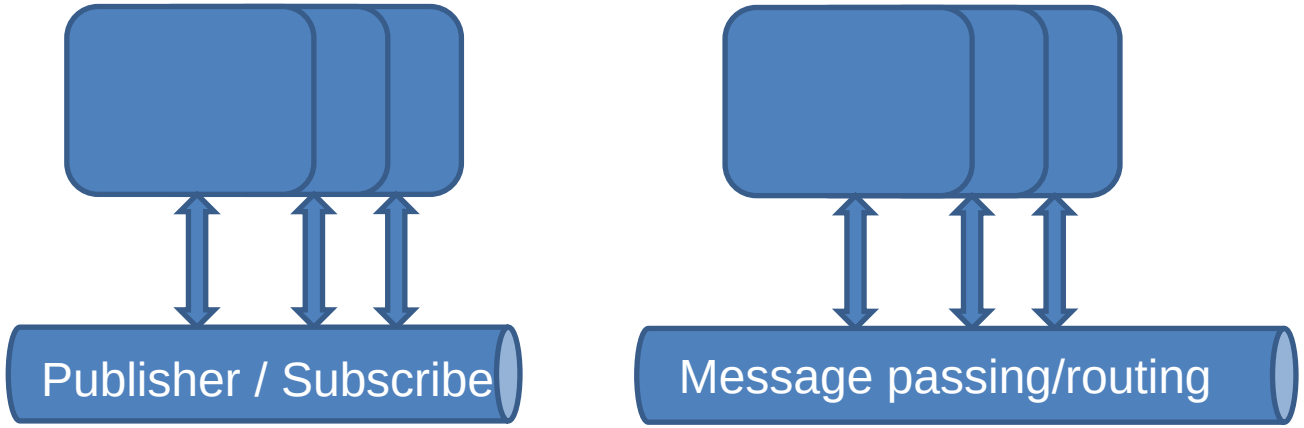
Components of a hybrid model

# Components may share the same application, environment, etc…

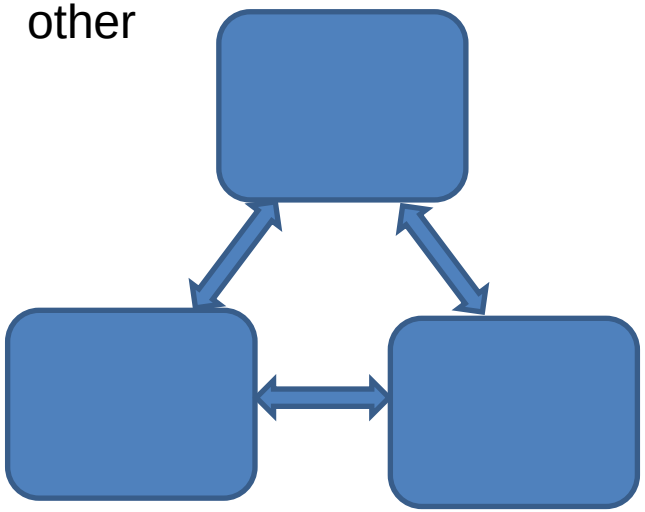


# Simulation standards use various connection mechanisms

“models” connect to a central ‘simulation’ bus/fabric

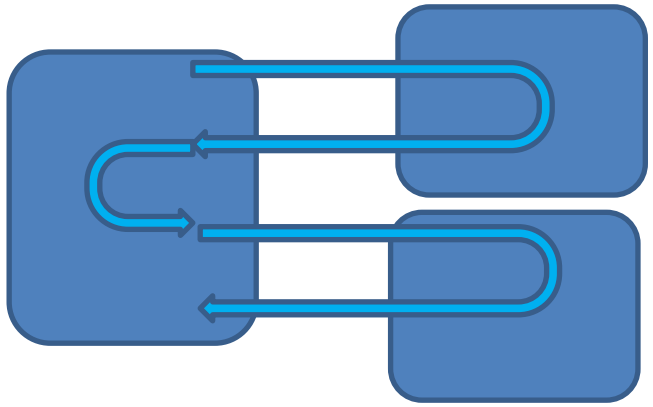



“models” connect to each other

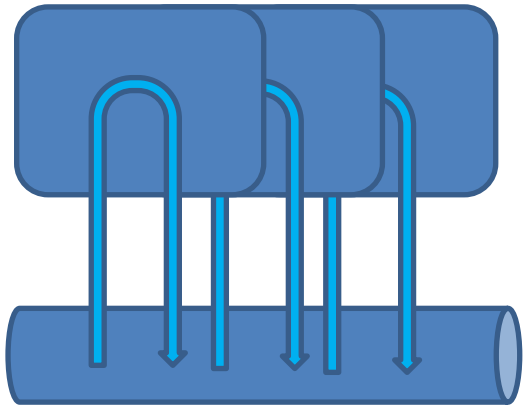


# Simulation standards use different notions of time

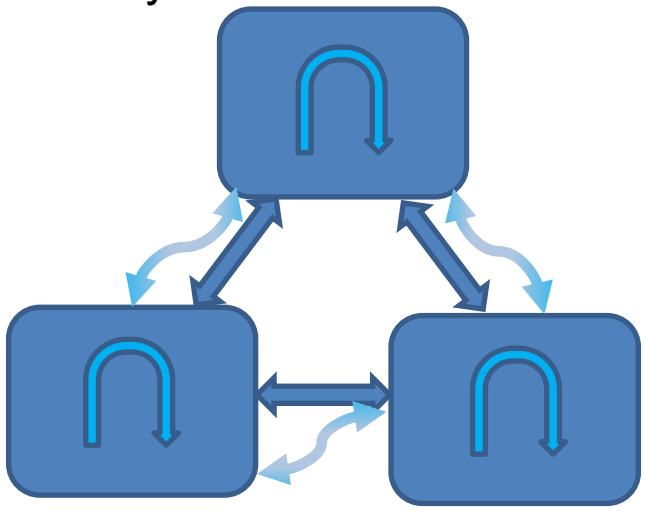
"time" and ordering flows though models



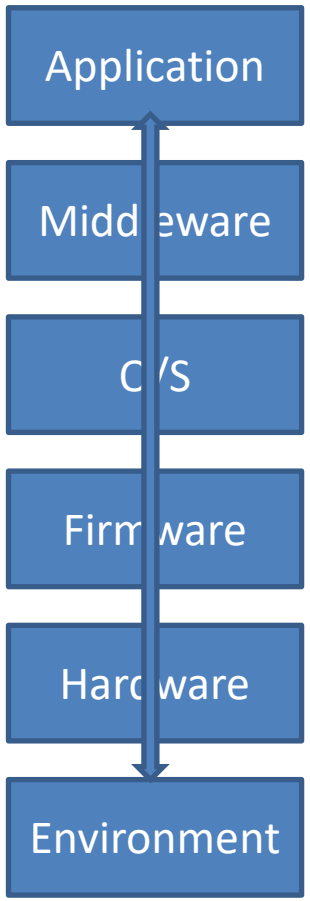
"time" and ordering flows though simulation system



"time" is distributed, and "synchronised"

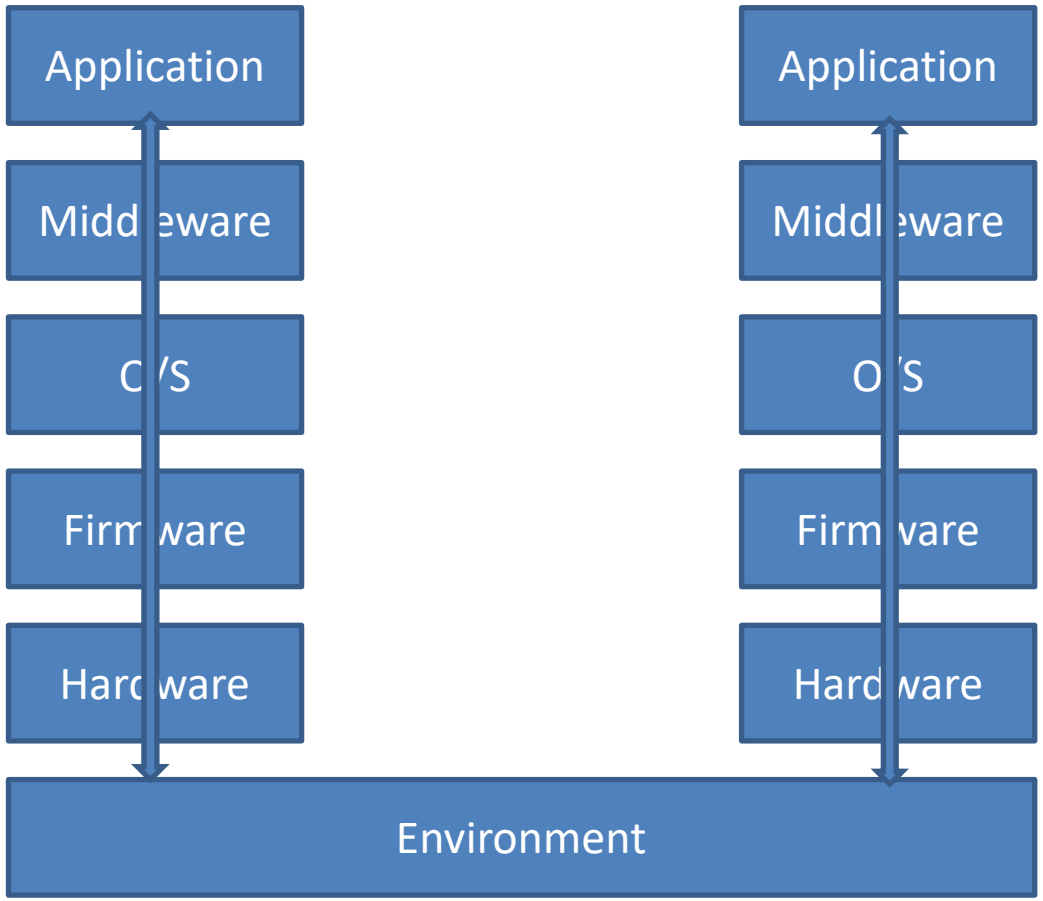


# Hybrid Simulation Approach



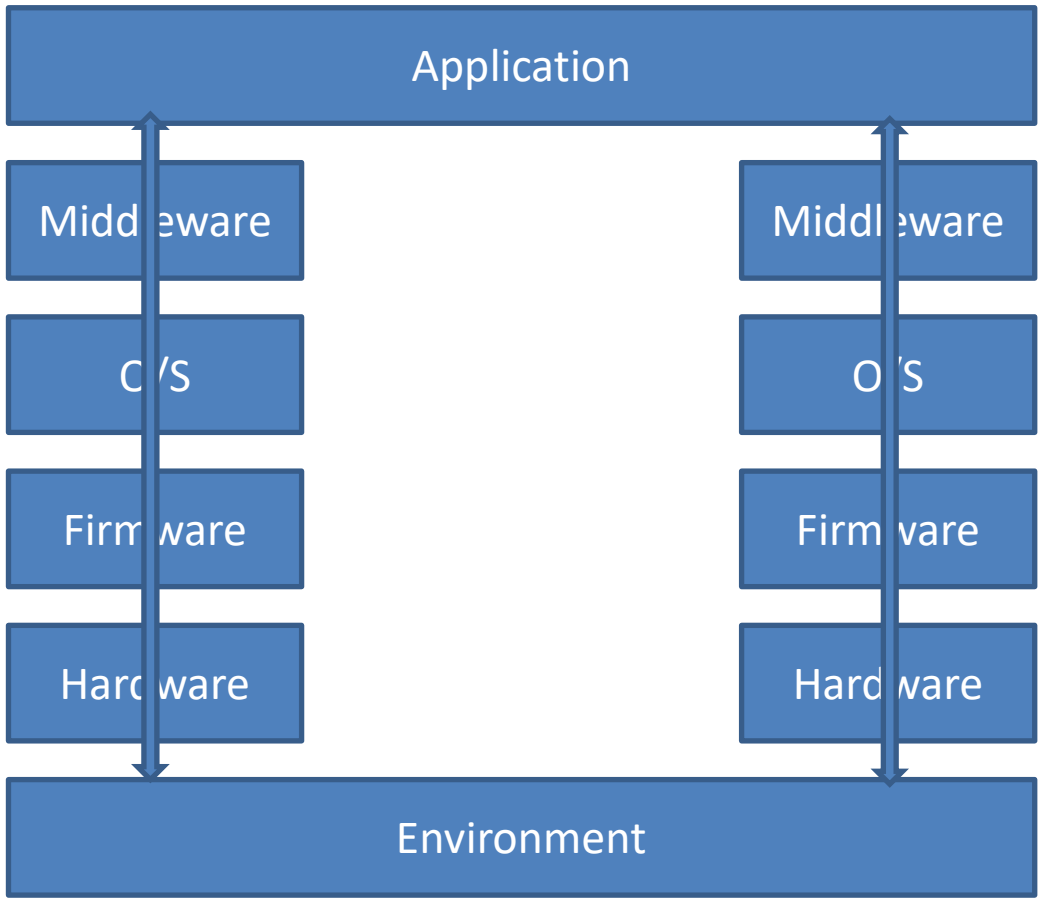

API's between components are specific

# Hybrid Simulation Approach




In reality, independent systems communicate through the environment

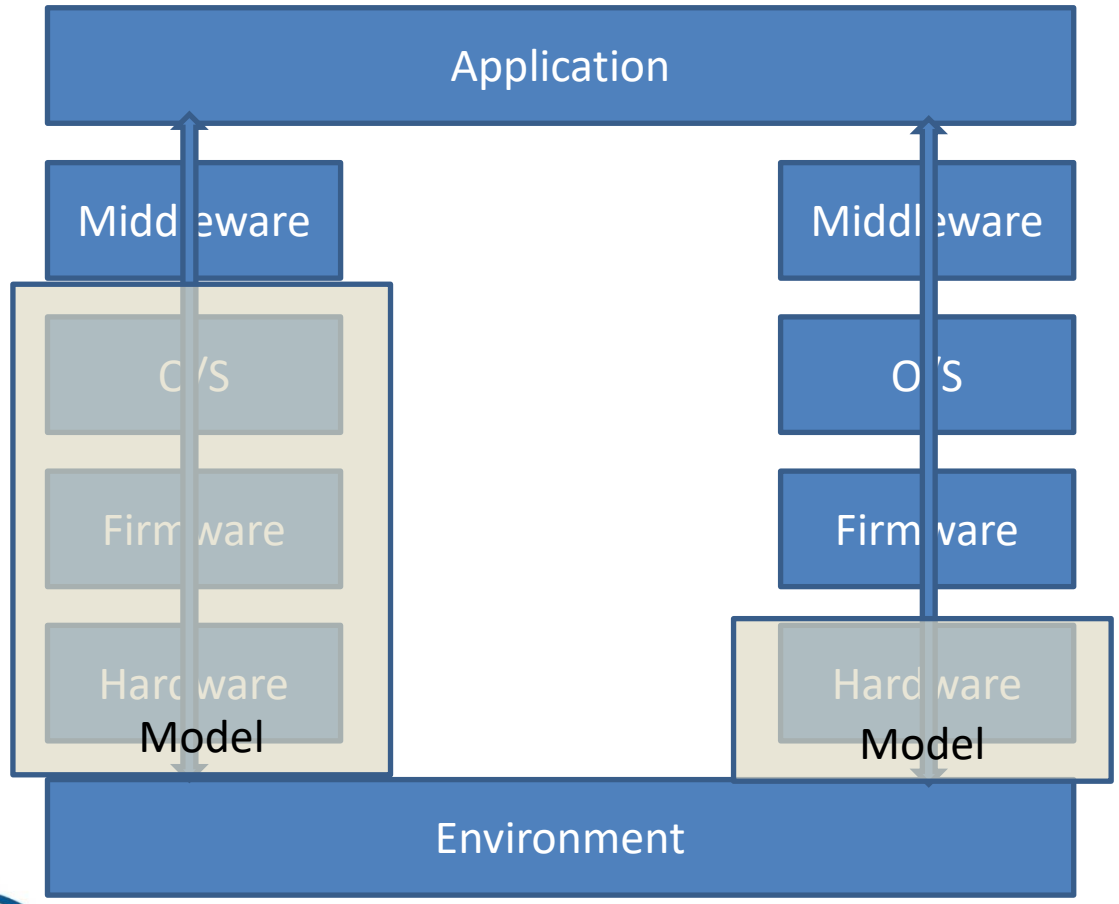
# Hybrid Simulation Approach




In reality, independent systems communicate through the environment

The ‘overall’ application may be the combination of several systems

# Hybrid Simulation Approach

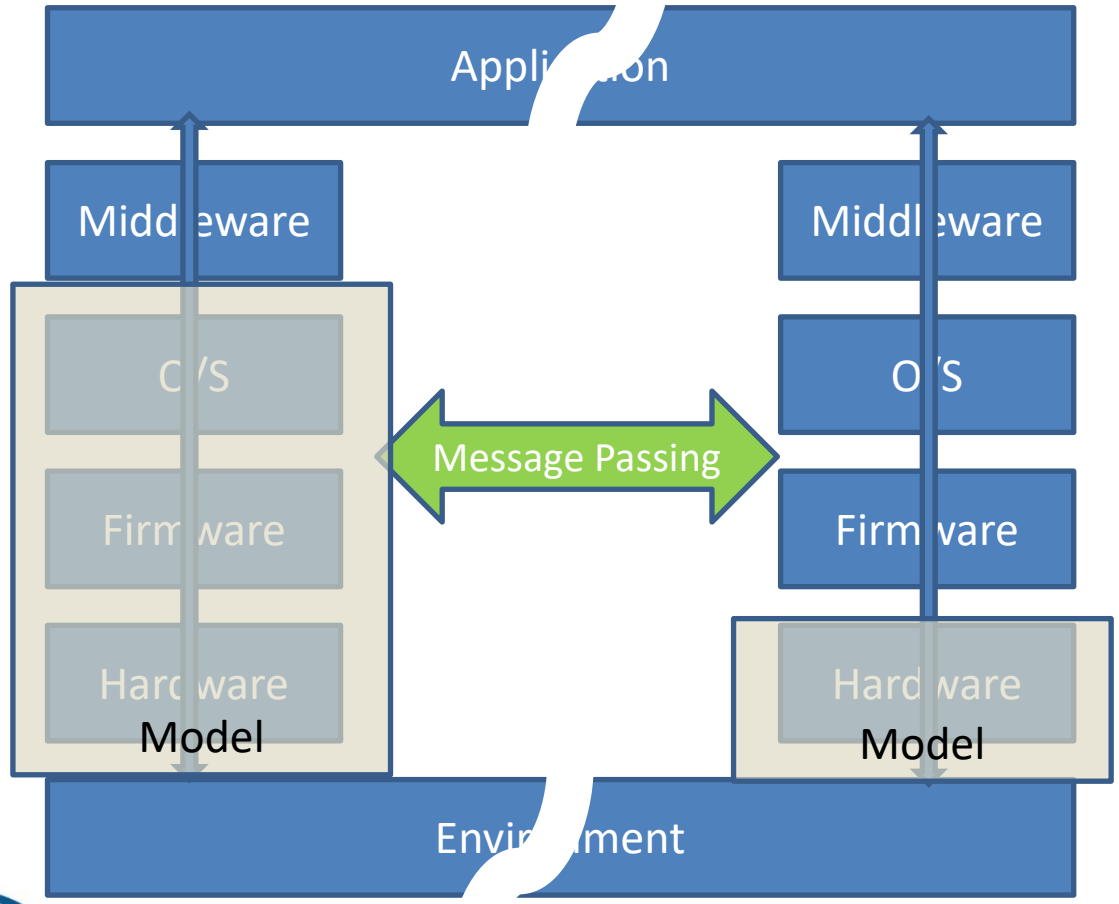



In reality, independent systems communicate through the environment

The ‘overall’ application may be the combination of several systems

The goal is to allow different combinations of ‘real’ and ‘virtual/modelled’ components to work together.

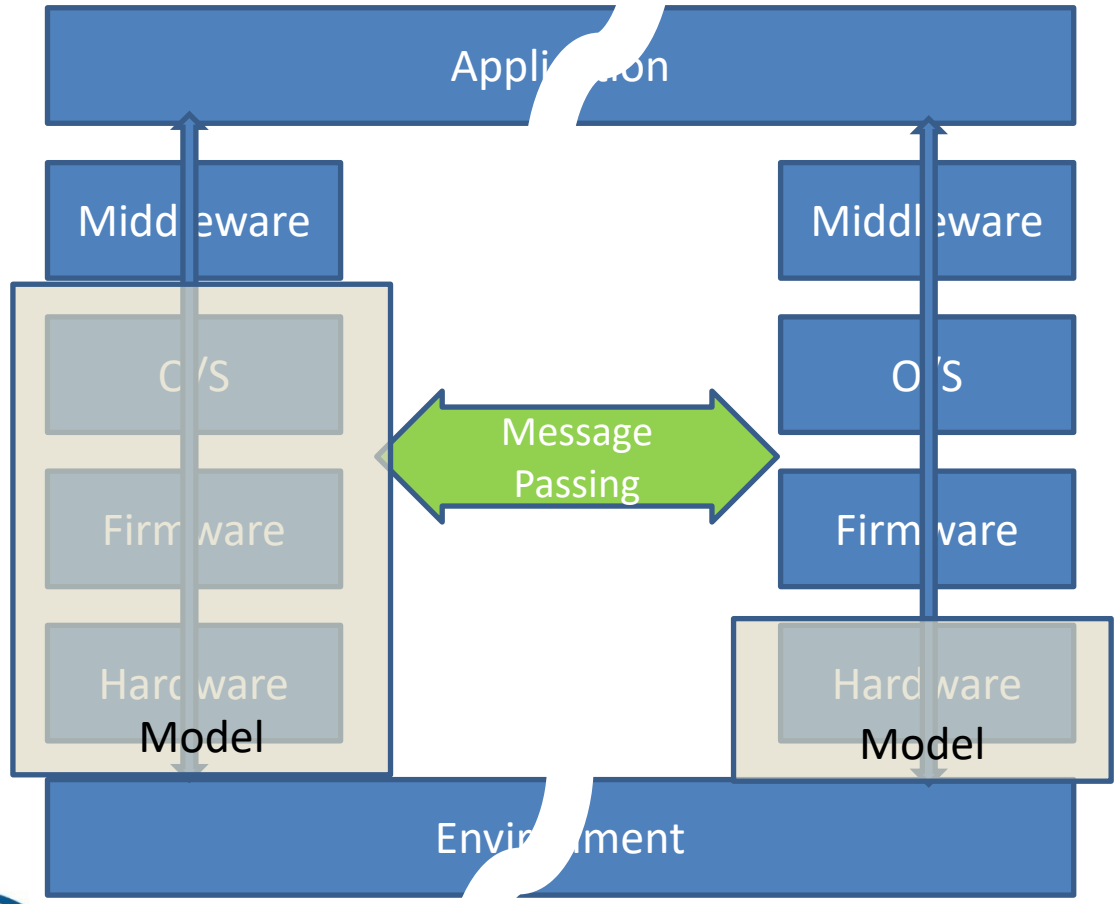
# Hybrid Simulation Approach




Individual systems typically manage only their part of the environment and application

Combining them requires a separate ‘communication’ mechanism

# Hybrid Simulation Approach



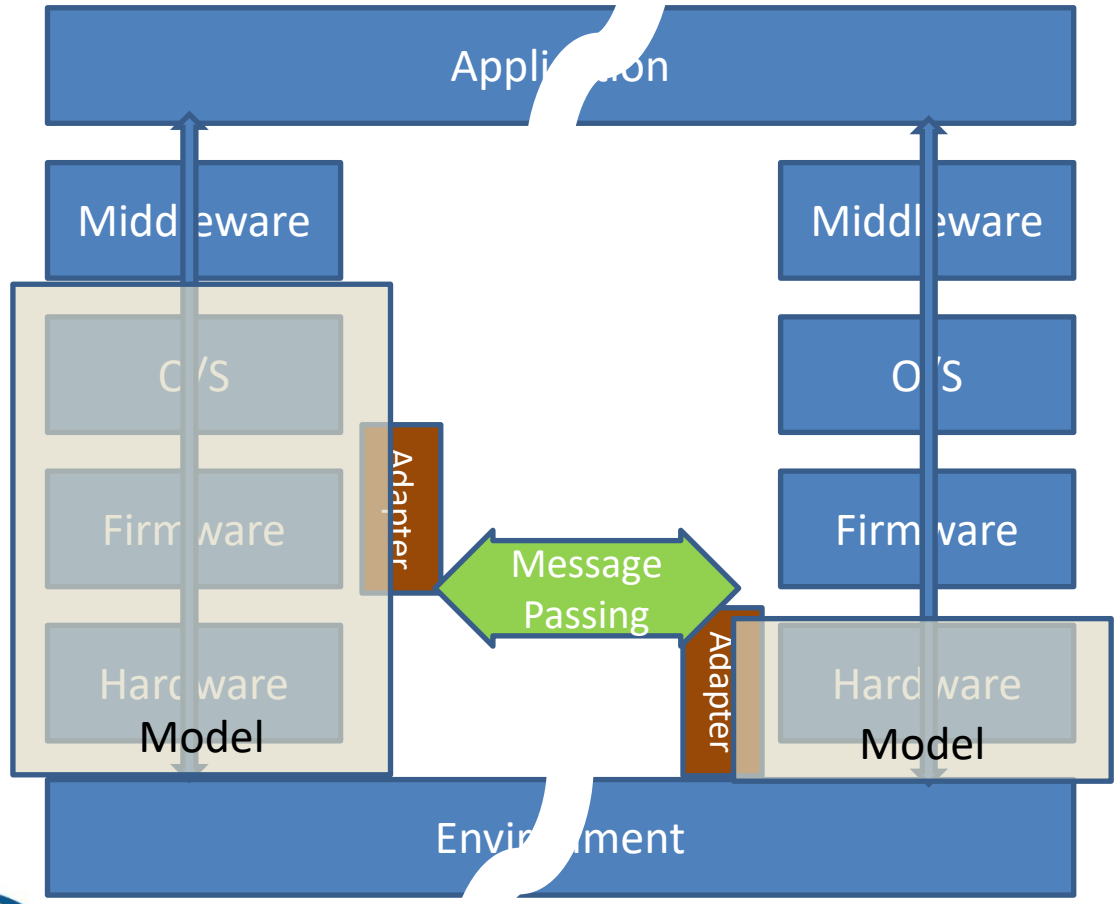

This backbone is what we are standardising

How it works

How to connect to it

How to implement connections to it.

# Hybrid Simulation Approach



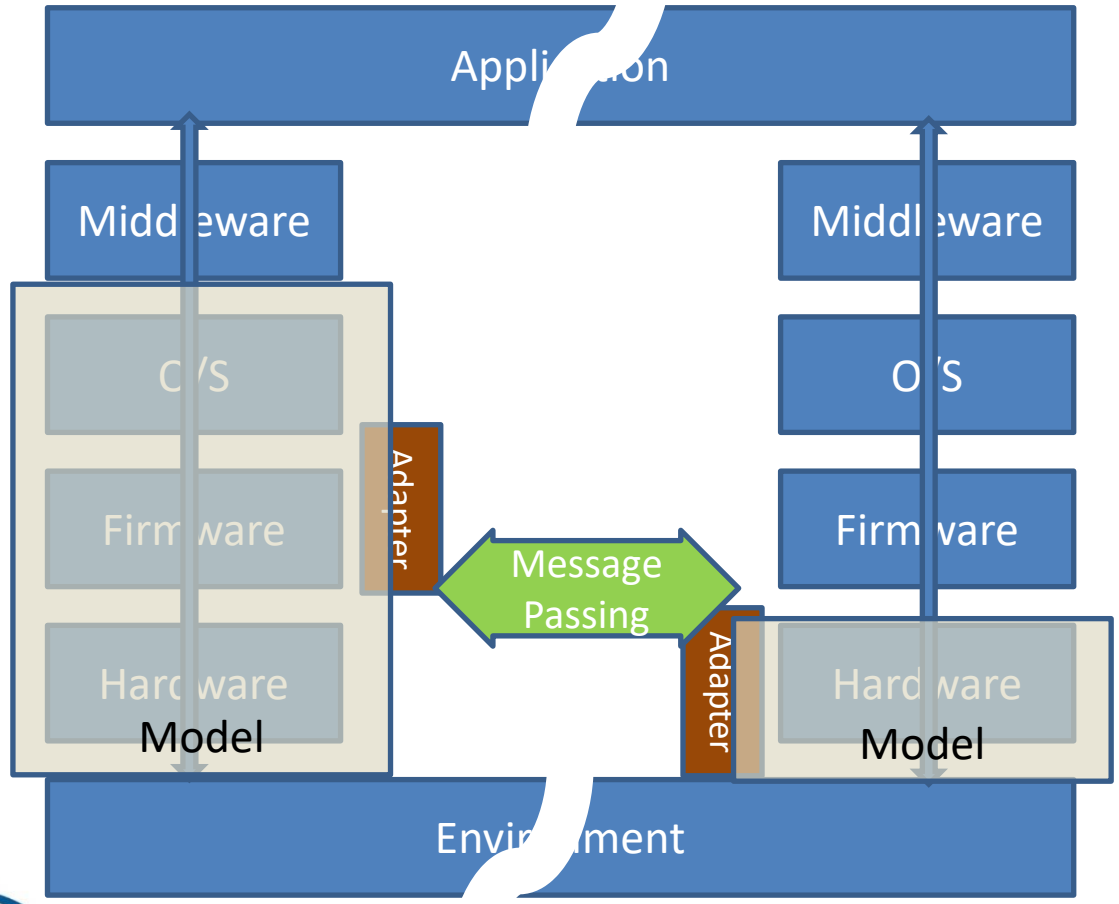

This backbone is what we are standardising

How it works

How to connect to it

How to implement connections to it.

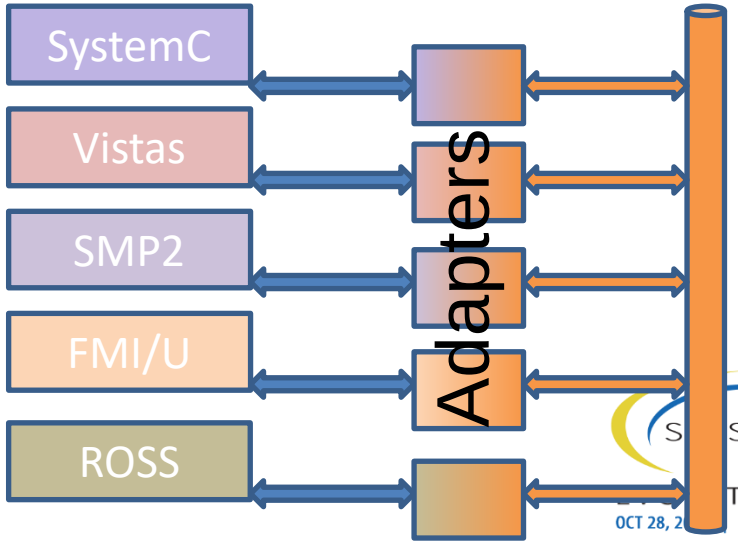
# Hybrid Simulation Approach




There are existing standards that do all of this within one domain.

Others which provide connections at specific levels

Our aim is not to replace them but to standardise how they can be adapted to work with each other



# *Example from ROS world*

Source:
https://www.researchgate.net/publication/309128426_Exploring_the_performance_of_ROS2