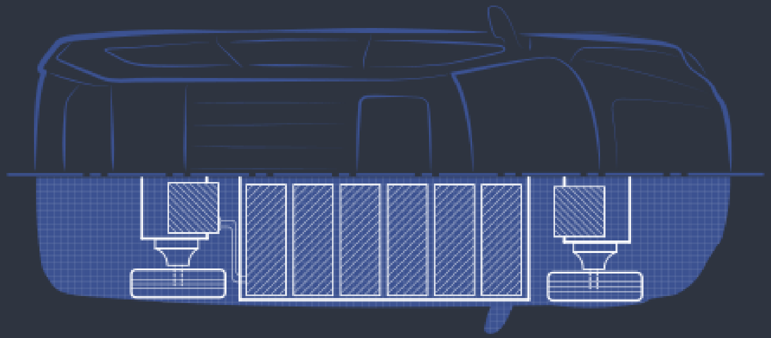
VOLVO

VSim @ Volvo Cars

Architecture and evolution from Simulink to Modelica



Anders Bergqvist, Johannes Emilsson, Fredrik Henningsson, Manikanta Venkatesh

VSim Heritage: 30 Years of System Simulation

VSim first released in 1995 by VTEC

Originally developed in Matlab/Simulink for energy efficiency and performance calculations

1999: Split into separate versions for Volvo Cars and Volvo Group (Ford acquisition)

2017: MOSAIC architecture introduced - preparing for acausal paradigm

Usage expanded: system design, certification, AD/ADAS, HIL, SIL, and more

Reaching the Limits of Causal Modeling

Matlab/Simulink VSim served its purpose but had reached its limits

Need for drastic change to expand platform usability

Key characteristics

- Sufficiently accurate
- Versatile
- Modular
- Fast
- Cost-effective
- Scalable

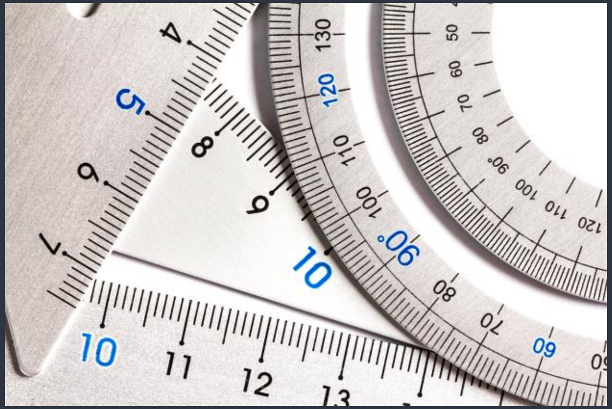
Sufficiently accurate

Trade-offs between execution speed and model detail

Accuracy depends on purpose—the questions you seek to answer

Sufficiently accurate while preserving high execution speed

Seamlessly change model configuration for more or less detail



Versatile

Causal: Information flow predetermined during model development

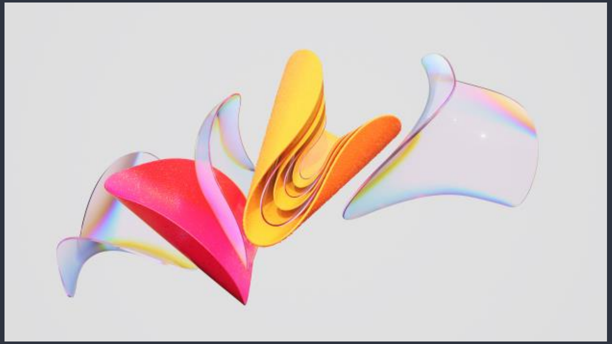
Acausal: Bidirectional information flow through physical connections

Seamlessly switch between use cases

Redefine quantity of interest without model restructuring

Example

Electric drive—unclear whether torque or speed is quantity of interest



Modular

Reconfigure system models by swapping individual modules

Comprehensive design space exploration

Resilience against technological change

Develop and integrate new modules without architectural changes

Platform remains relevant across vehicle generations

Speed and Cost-Effectiveness

Fast

Evaluate thousands of design variants in fraction of physical testing time

Run many simulations quickly for uncertainty quantification, design space exploration, and optimization

Enables better-informed decisions earlier in development

Reduces risk of costly design changes later

Cost-effective

License model that doesn't incur higher costs as usage increases

Scalable

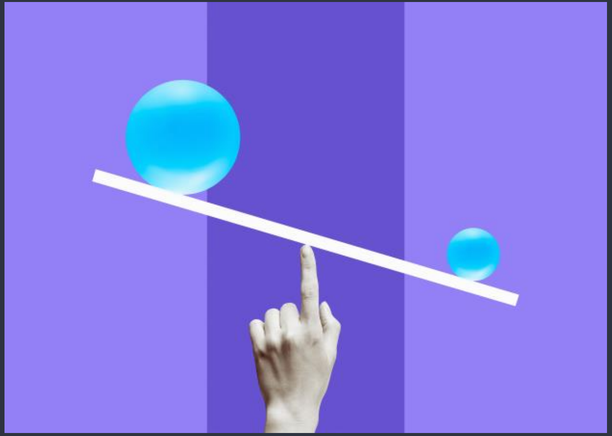
Effective from complete vehicle architectures to individual components

Accommodate seamless integration of additional models, domains, and analysis methods

No fundamental restructuring required as platform evolves

Investments in model development remain valuable

Efficiently adapt existing models to emerging challenges



VSim Architecture

Guidelines - Automated quality assurance and modeling best practices

Physical mapping - Intuitive representation and correlation

Modular design - Component independence and reusability

Testing - Comprehensive validation strategy

Modeling Guidelines

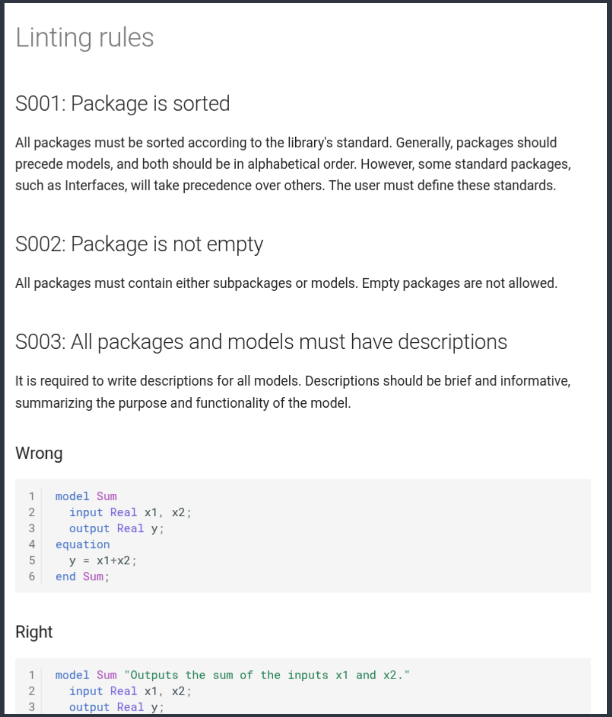
Codify modeling best practices for consistent quality and maintainability

Examples of rule checks

- Packages cannot be empty
- All models must include meaningful descriptions
- Naming conventions

Benefits

- Reduce errors
- Improve readability
- Facilitate collaboration
- Ensure reusability



Linting rules

S001: Package is sorted

All packages must be sorted according to the library's standard. Generally, packages should precede models, and both should be in alphabetical order. However, some standard packages, such as Interfaces, will take precedence over others. The user must define these standards.

S002: Package is not empty

All packages must contain either subpackages or models. Empty packages are not allowed.

S003: All packages and models must have descriptions

It is required to write descriptions for all models. Descriptions should be brief and informative, summarizing the purpose and functionality of the model.

Wrong

```
model Sum
  input Real x1, x2;
  output Real y;
equation
  y = x1+x2;
end Sum;
```

Right

```
model Sum "Outputs the sum of the inputs x1 and x2."
  input Real x1, x2;
  output Real y;
```

VOLVO

Physical Mapping

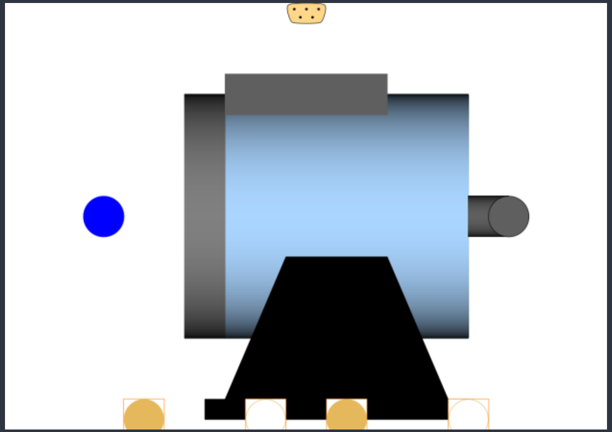
Graphical layout mirrors physical reality

Interfaces represent actual physical connection points

Correspondence between model interfaces and measurement points

Measurement points on physical components have exact analogs in models

Simplifies correlation and increases confidence in predictions



Modular Design

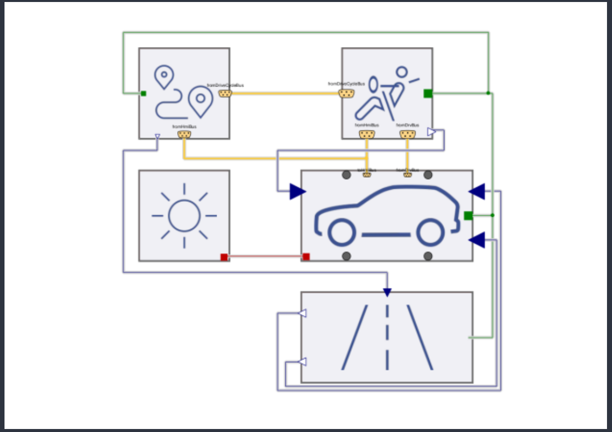
Self-contained modules with well-defined interfaces

Acausal connectors ensure compatibility

Complex systems built from simpler subsystems

Rapid assembly of vehicle variants from validated components

Components reused across vehicle programs with minimal modification



VOLVO

Testing

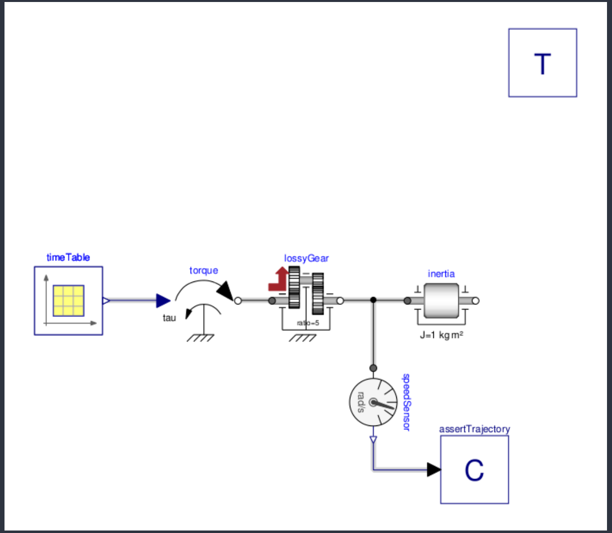
Component-level testing

Unit tests verify correct model behavior and meaningful results

Integration testing

Verify modules work correctly when combined

- Interface consistency
- Physical connections behave as expected
- Energy conserved across boundaries



Challenges: Tools and Standards

Formatter

No central formatter/style guide checker — developed our own

Tool differences

Testing in both Dymola and OpenModelica

- Strength: Ensures conformity to Modelica specification
- Challenge: Models don't always exhibit same behavior across tools

Testing framework

No formal testing framework in Modelica

- Developed separate library for unit tests in Modelica
- Unit test execution configured in Python using pytest

Challenge: Learning Curve

Changing tools always requires training and adaptation

Matlab & Simulink dominant environment for system simulation

Many engineers have strong background in Matlab scripting

Need for comprehensive training and support during transition

What's Next for VSim

Platform still under development

Enhanced thermal system model

Increase number of use cases

Model sharing

VOLVO

Questions?

johannes.emilsson@volvocars.com