Model-Based Engineering Platform to Manage Complexity and Scale

Ramamurthy Mani
Engineering Director, Simulink Semantics

Our theme today:
Evolution

Stateflow

RTW

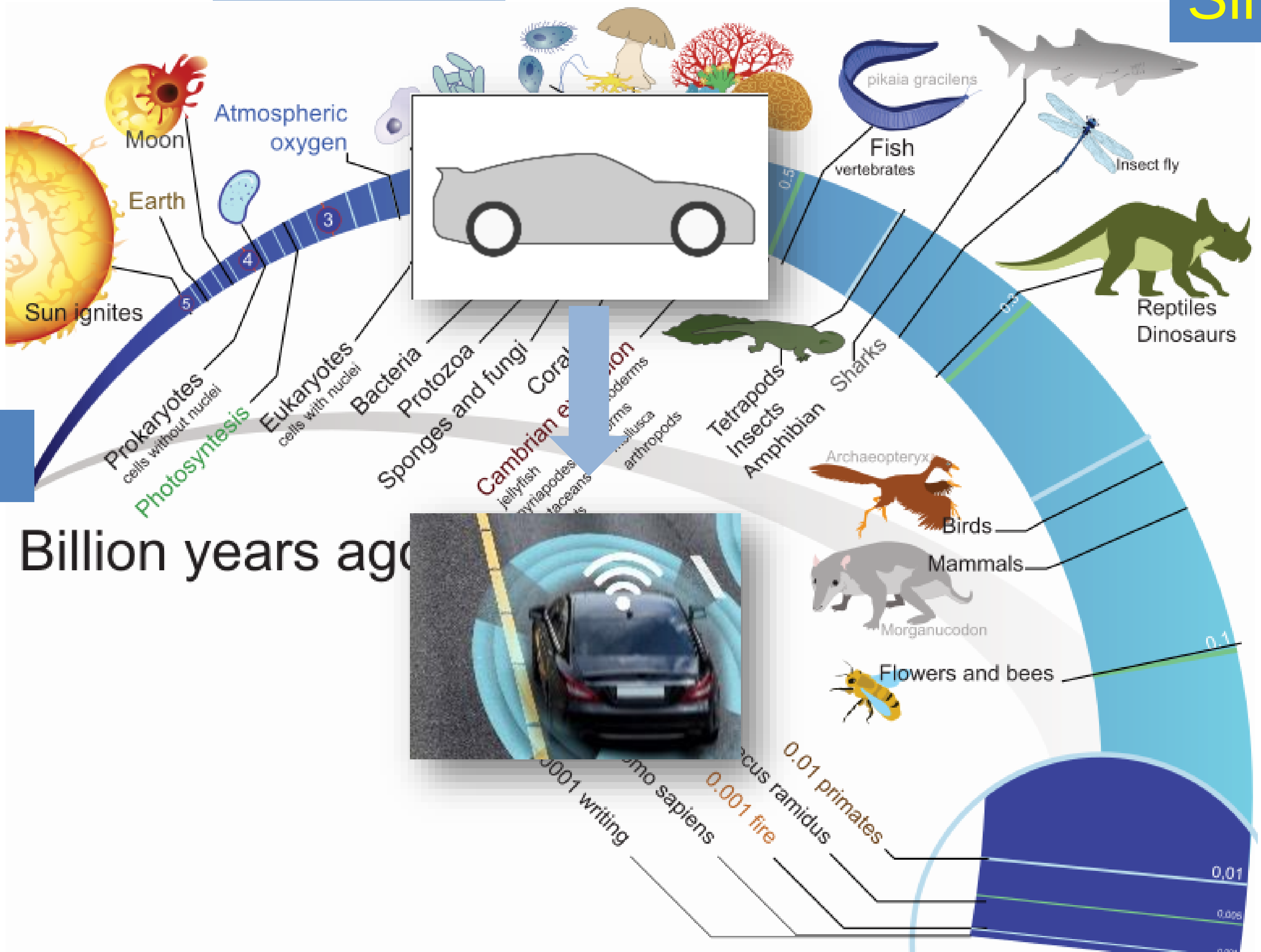
Simscape

Coder

V&V

Simulab

1990



Simulink

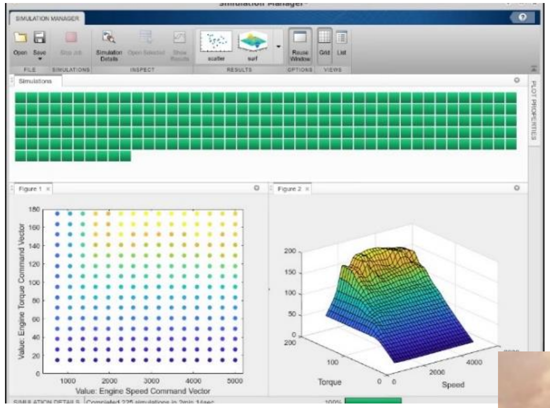
R2020a

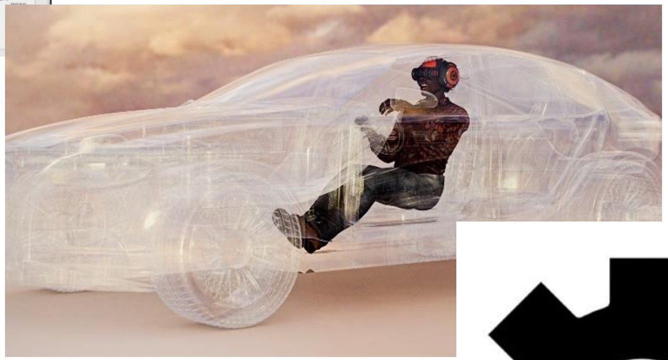
The Three Evolutionary Forces at Play

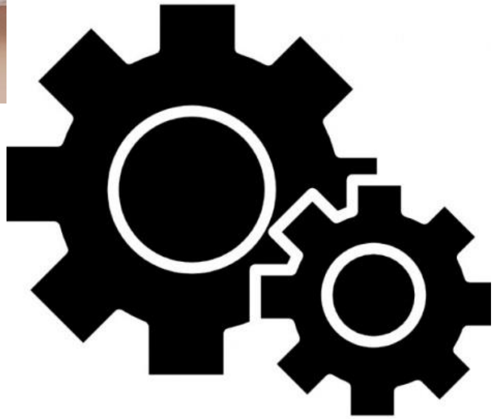
1. Simulation Scale

2. Design Complexity

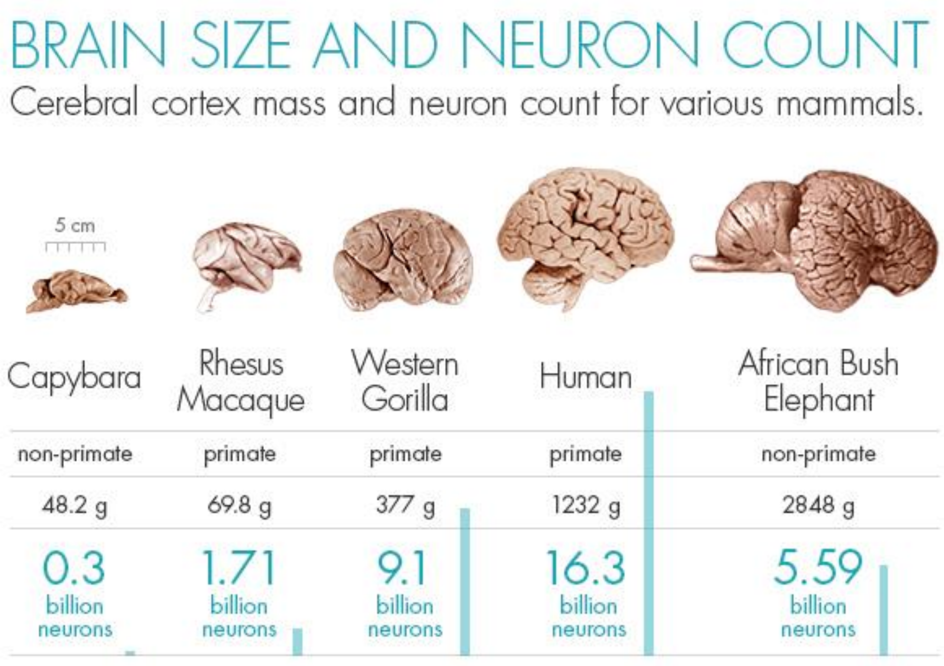
3. Collaborative Engineering







Evolving for **Simulation Scale**



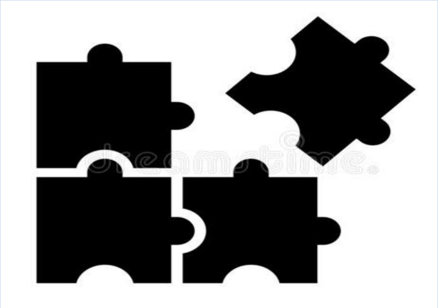
https://www.quantamagazine.org/how-humans-evolved-supersize-brains-20151110/

Trend: Demand for scaled up simulation capabilities

Full Vehicle Simulation

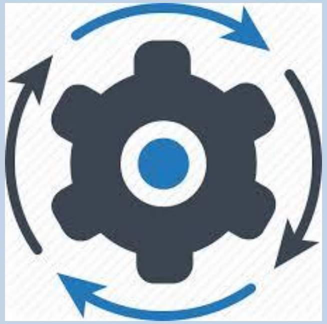
Strategy: Continuously evolve Simulink to be a best in class Simulation Integration Platform

The primary challenges for simulation scale



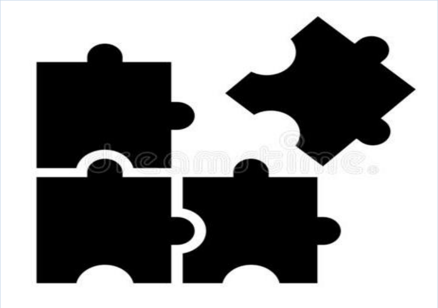
Integration

Performance



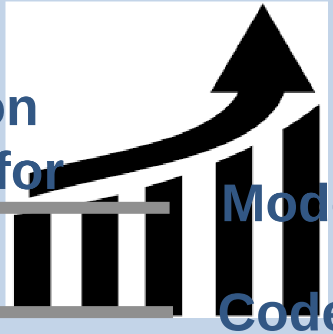
Operationalization

Integration of algorithms with multiple simulation interfaces is key



Integration

Integration
Platform for

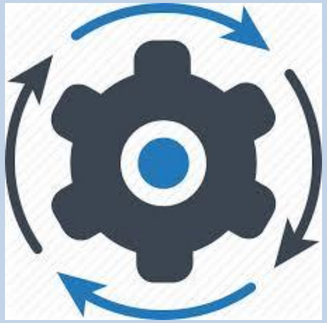


Models

Code

Performance

Other tools



Operationalization

For Models, core modularity principles underpin integration

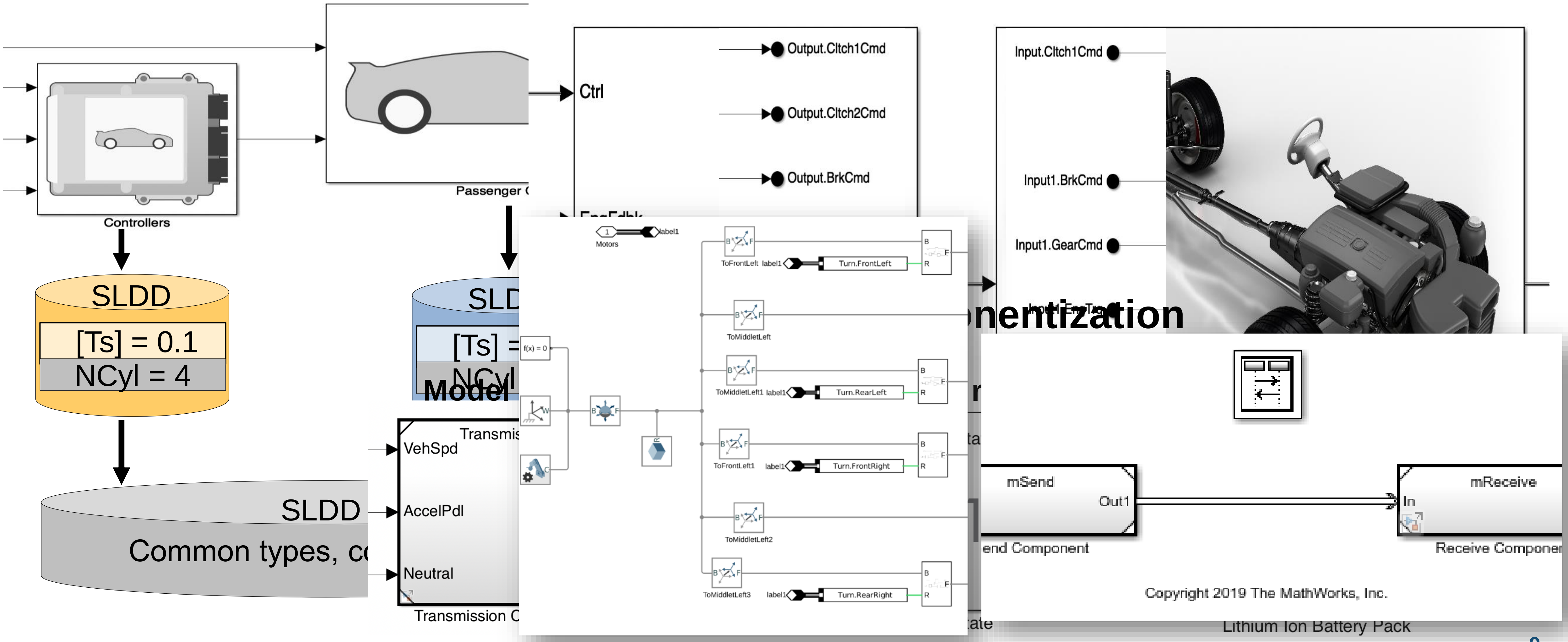
Models

Code

Tools

Data Encapsulation

Interface Management

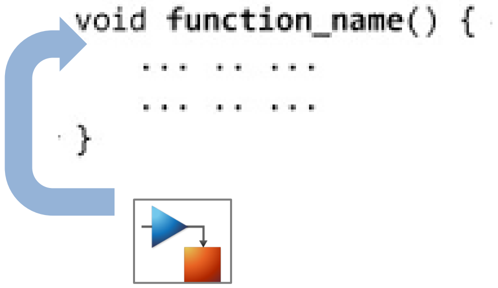


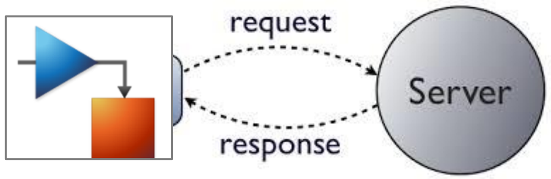
You can easily bring C/C++ code into Simulink

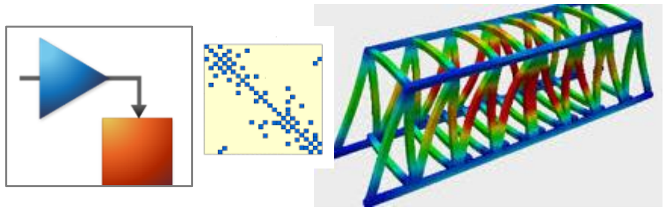
Models

Code

Tools





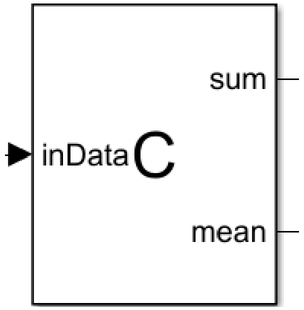


Basic

Advanced

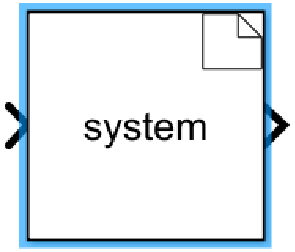
C Caller

R2018b

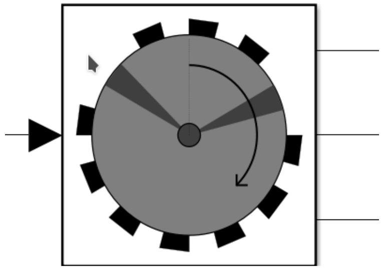


C Function

R2020a



S-Function Builder



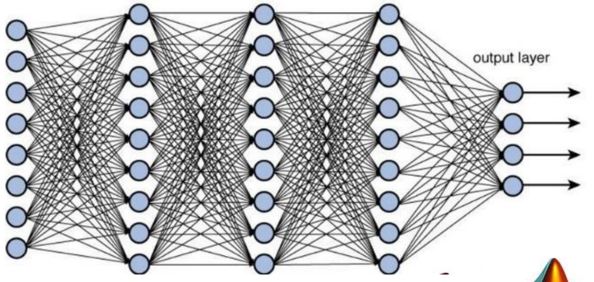
S-Function

You can use MATLAB algorithms like the Deep Learning Toolbox in Simulink

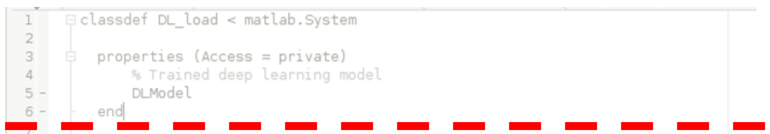
Models

Code

Tools

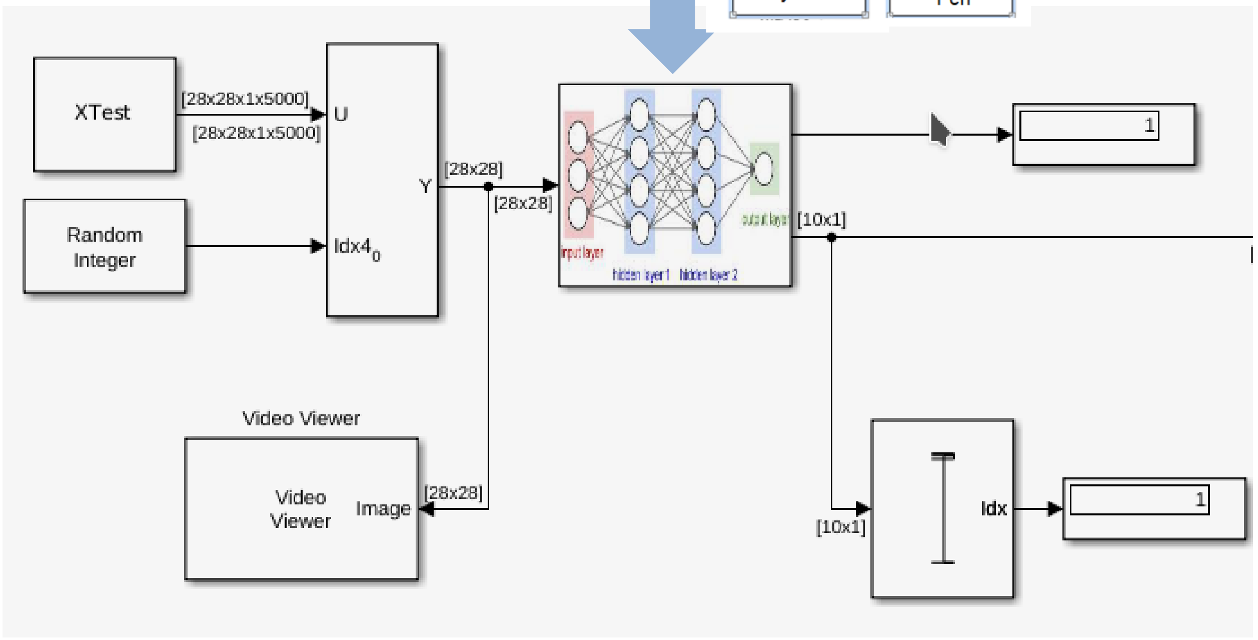


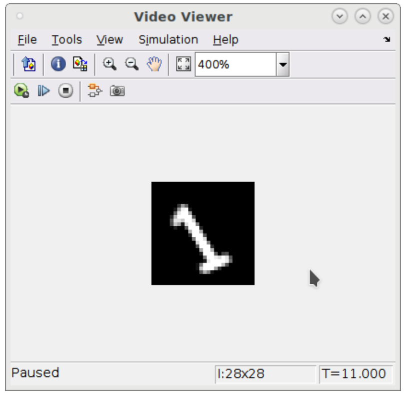
Deep Learning Toolbox



obj.DLModel = coder.loadDeepLearningNetwork(‘mydnn.mat’ , ‘network’)

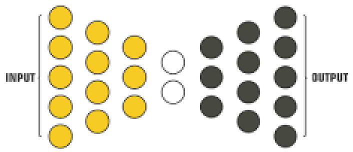
R2020a



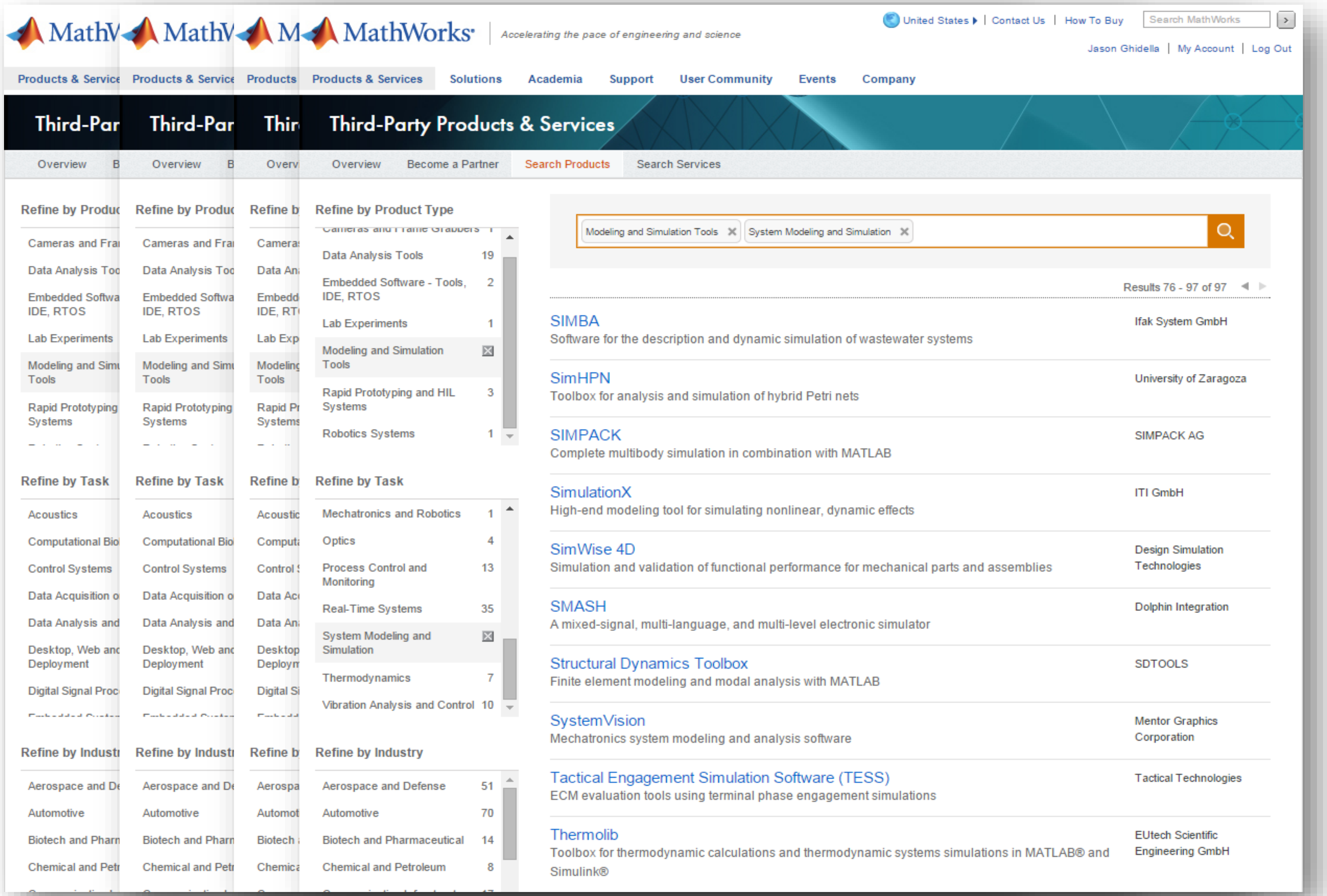


Tensorflow-
Keras Importer

Keras TensorFlow



Simulink has simulation interfaces to 190 connection partner products and services primarily through the S-Function interface



Models

Code

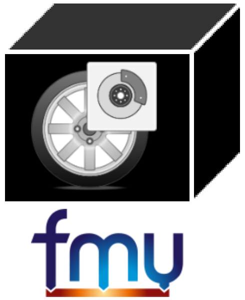
Tools

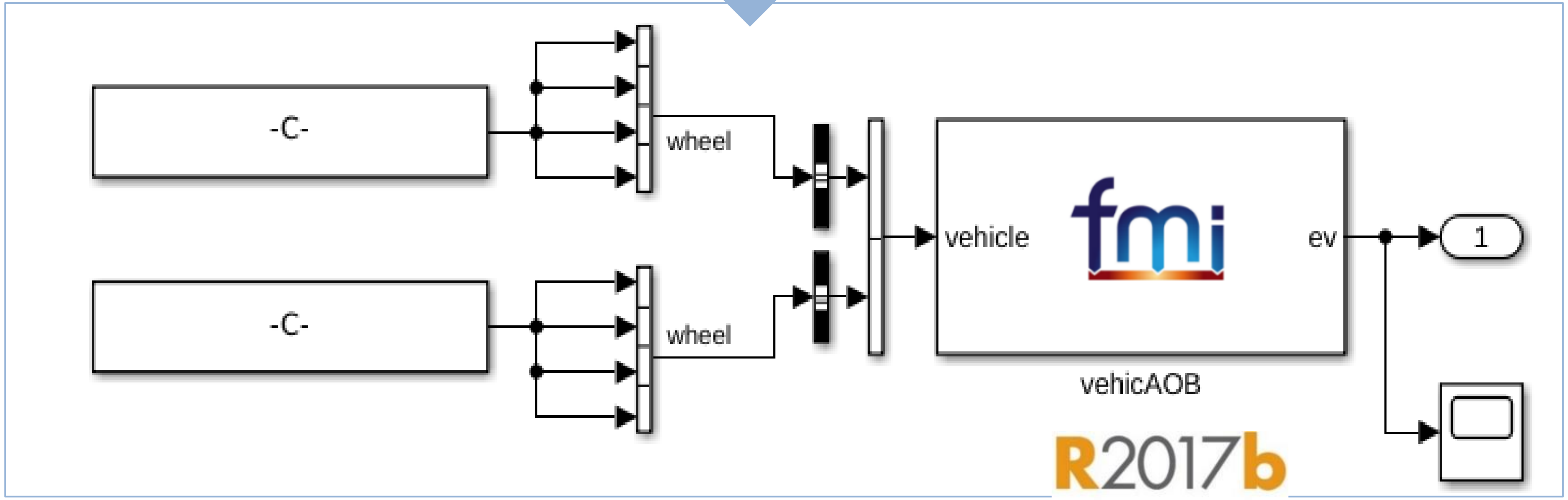
MathWorks Partner

Using FMUs inside Simulink is easy and expressive

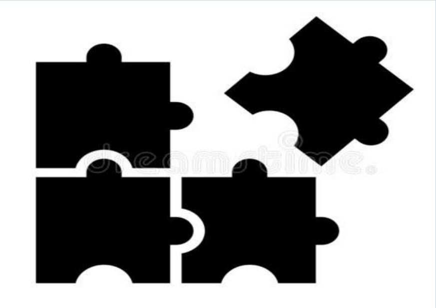
- Models
- Code
- **Tools**

3rd party tool

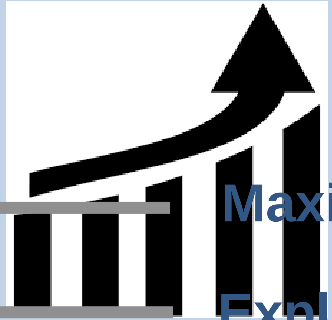




System-level simulations are computationally expensive



Integrating models and components

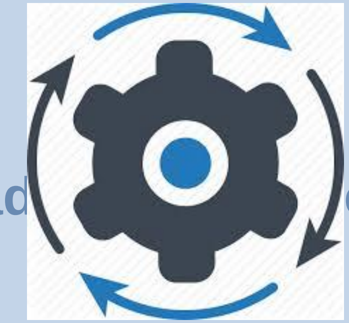


Simulation Performance

Maximize single thread ... **e**

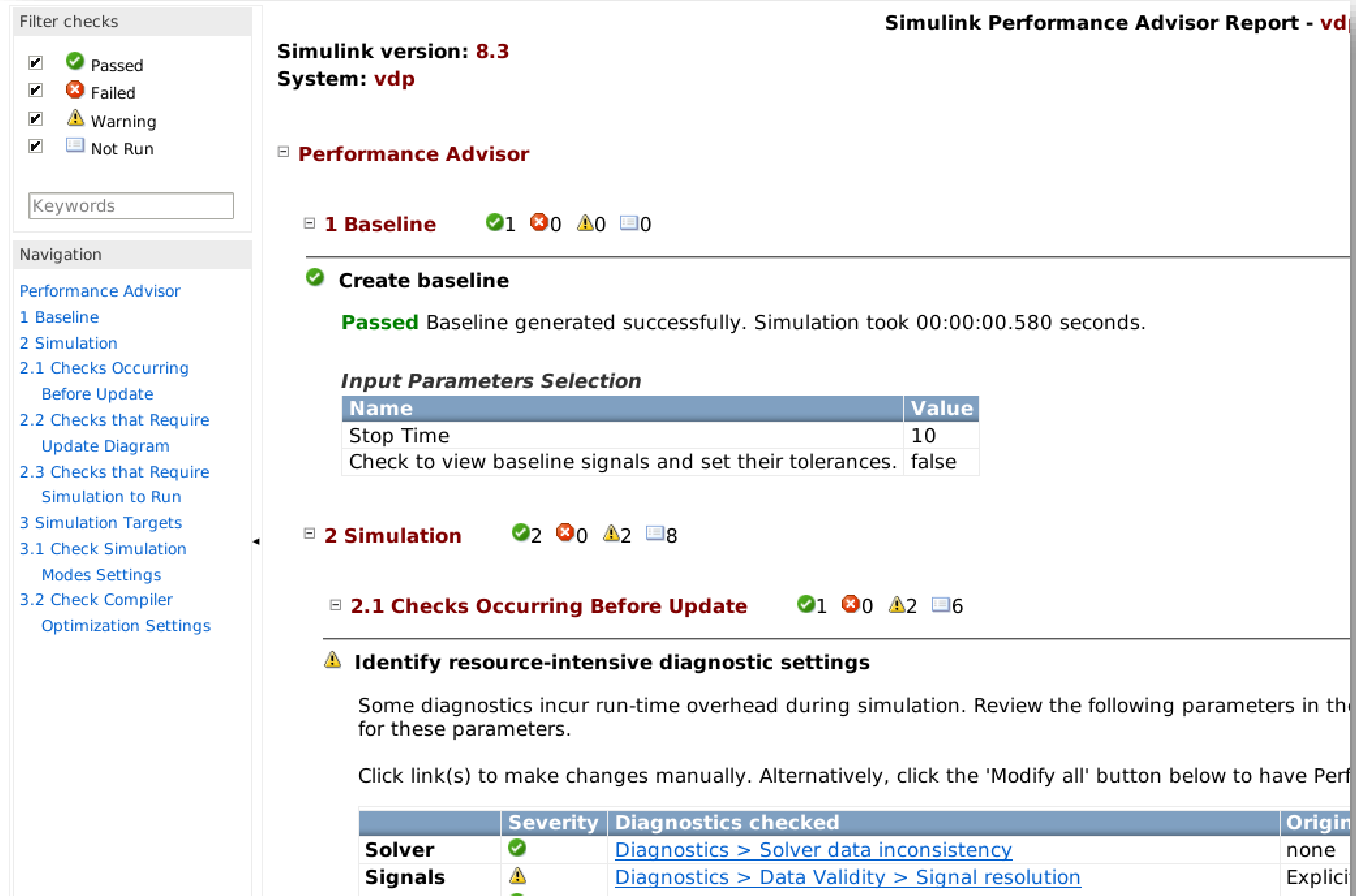
Exploit parallelism

Enable massive simulation runs



Operationalization

Maximizing performance by discovering speed-up opportunities: Performance Advisor



- Consolidated advice on performance
- Gives advice that works!
- Helps discover performance focused capabilities

Invest in multiple parallelization techniques for boosting performance

Model block, S-function, FMU import

R2018a

R2018a

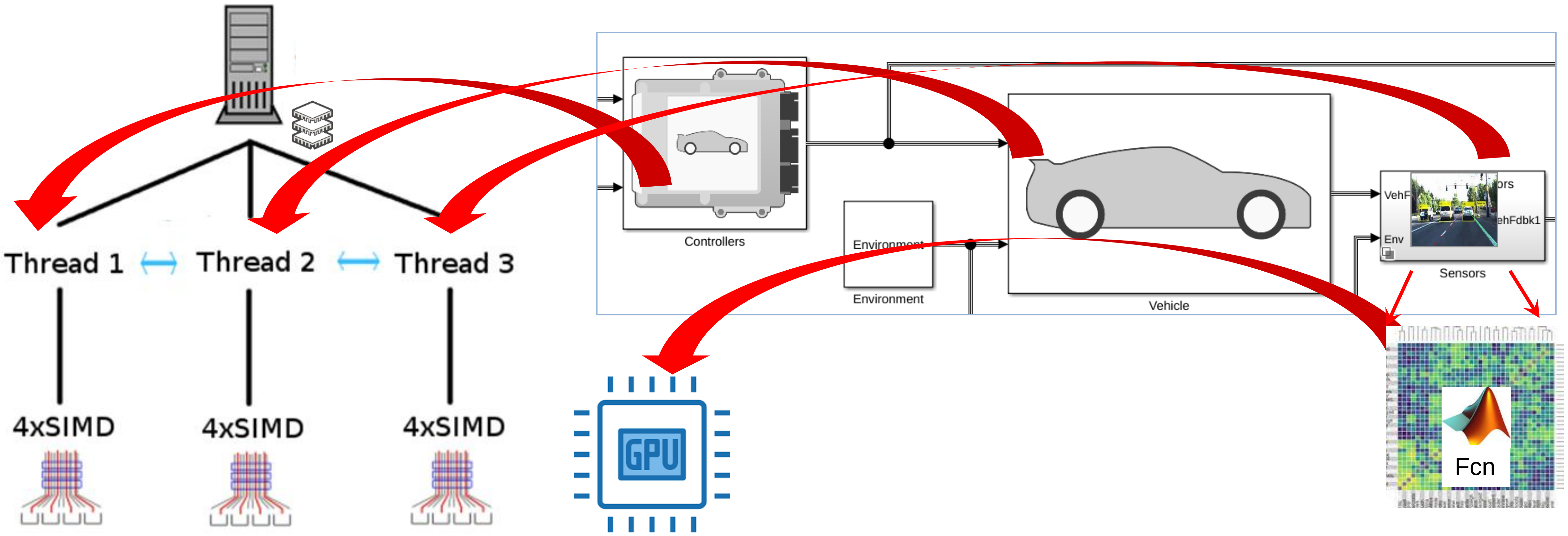
Dataflow SIMD

ForEach Subsystem Parallelization

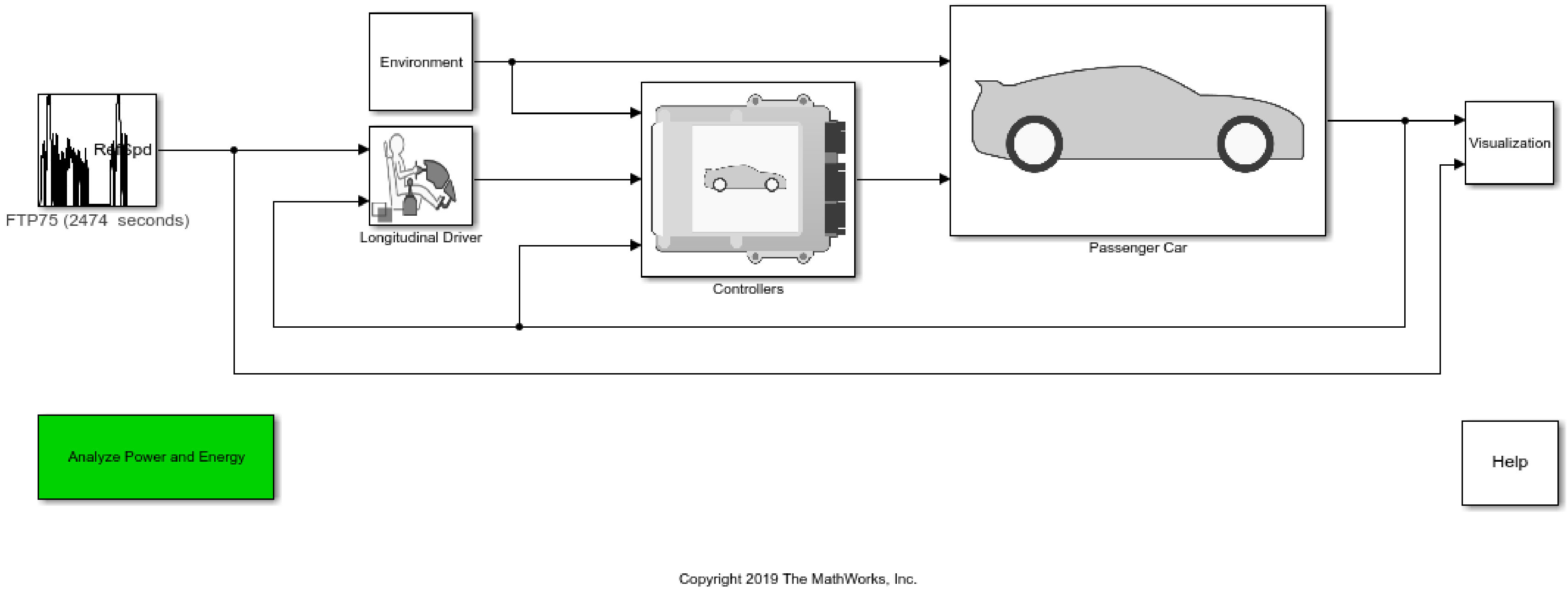
MATLAB Function GPU acceleration

Compute Clusters

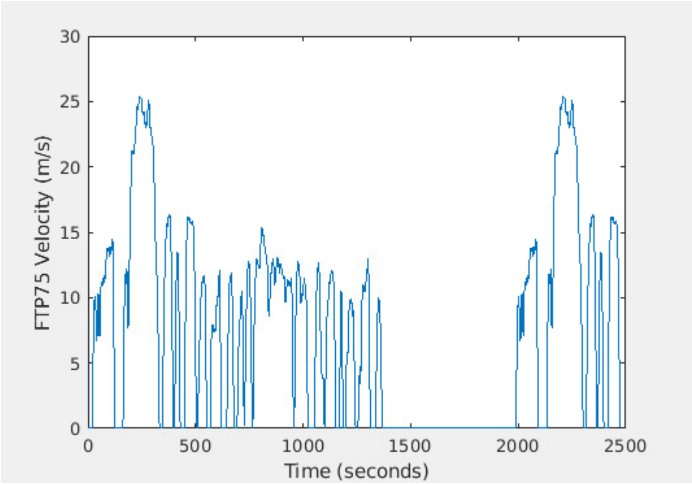
FUTURE RELEASE



Design envelope studies require a large number of simulations



Full vehicle model



Driving cycle

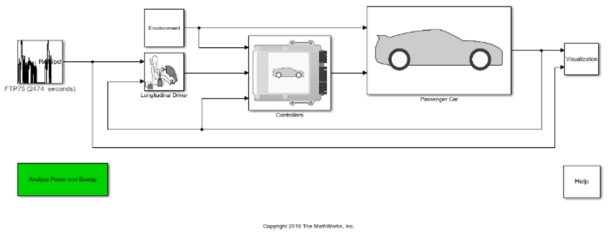
100 drive cycles × 10 vehicle loadings × 10 weather conditions

10,000 simulations

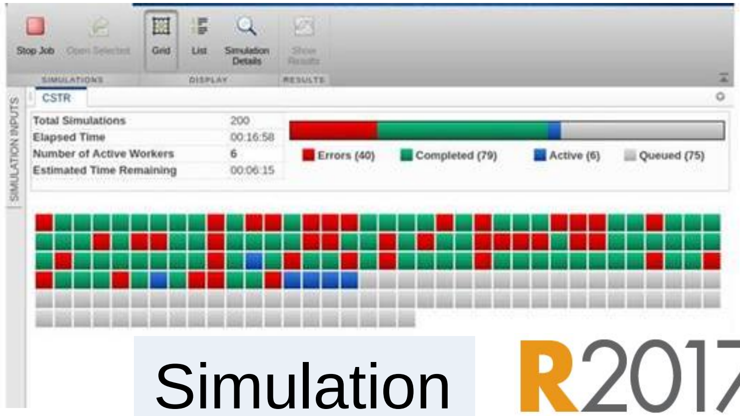
Optimize gear ratios

Simulink enables massive simulation workflows

Setup



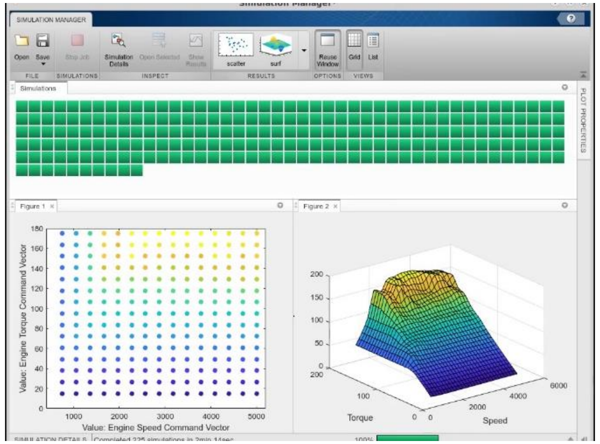
Simulate



Simulation Manager

R2017b

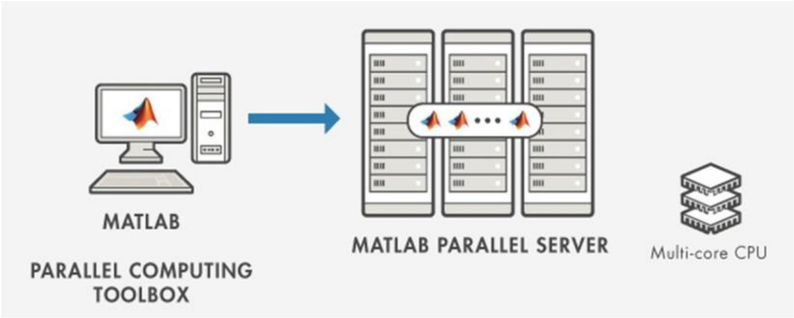
Analyze



Simulation Manager

R2019b

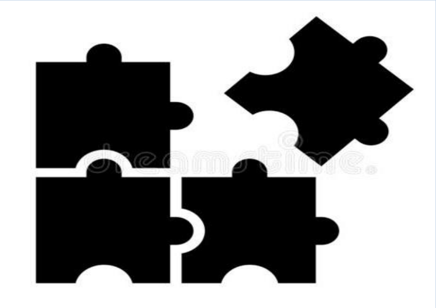
FUTURE RELEASE



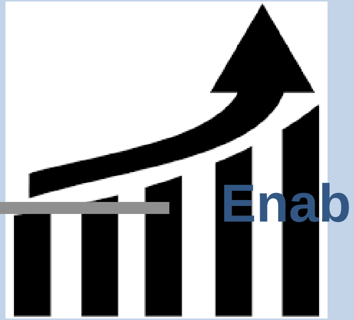
`parsim` R2017b

`batchsim` R2018b

Extend simulations to Operational phases of the system

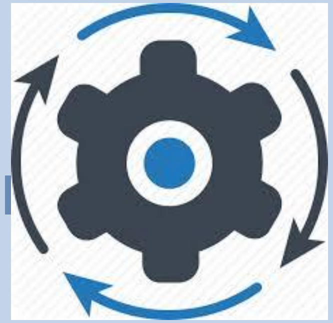


Integrating models and components



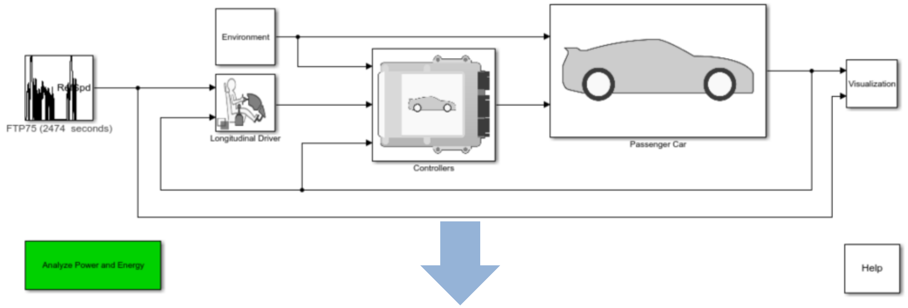
Simulation Performance

Enable simulation depl



Operationalization

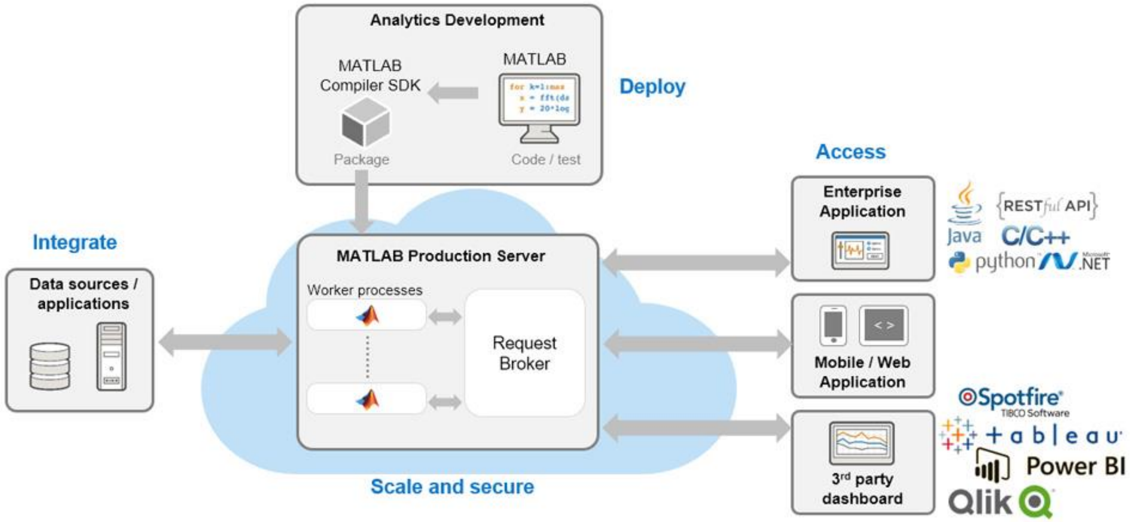
Simulink Compiler enables deployment of simulations



Simulink Compiler

R2020a

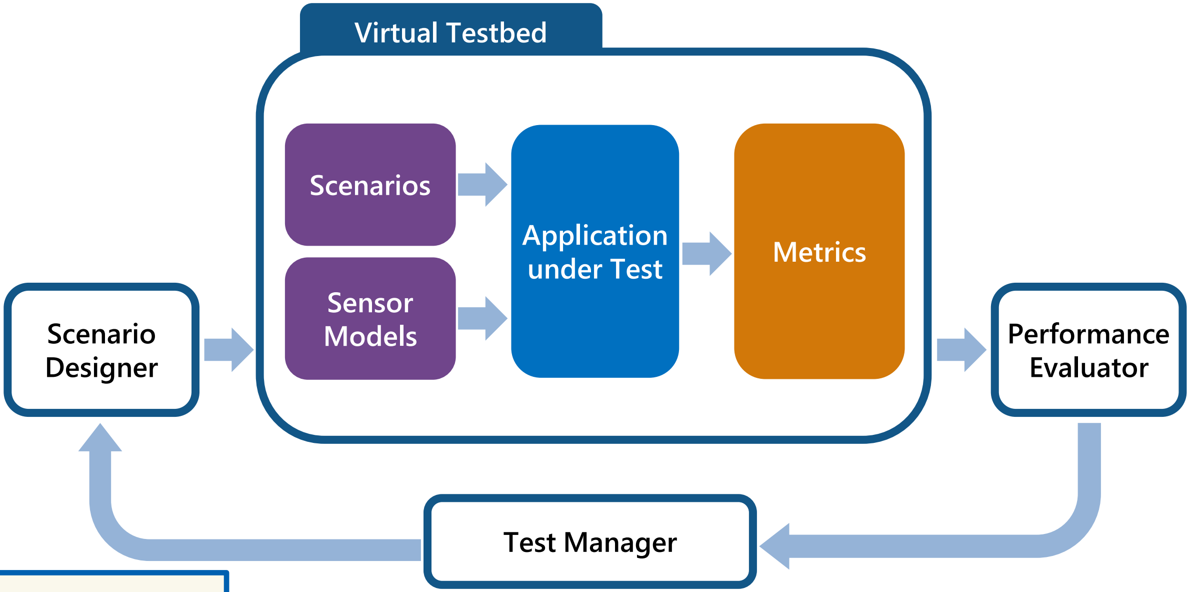
Integrate as Enterprise Application

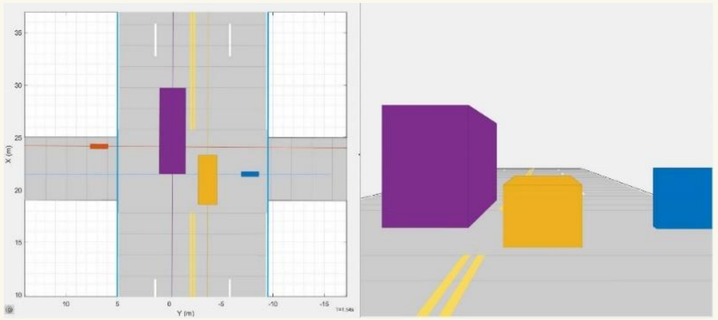


Trend: Demand for simulating complex scenarios with multiple actors is increasing

Scenario Simulations for Autonomy

Strategy: Create a platform for system-of-systems (scenario) simulations





Cuboid Driving Simulatio

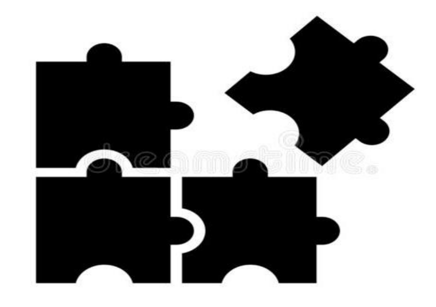
Unreal Engine Driving Simulation

RoadRunner

Simulink platform is evolving to meet the demands of scaled up simulations

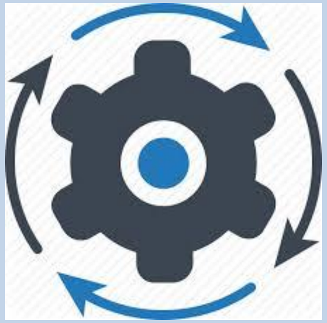
Full Vehicle Simulation

Scenario Simulations for Autonomy



Integrating models and components

Simulation Performance



Operationalization

Scenario Simulation

Evolving for Design Complexity

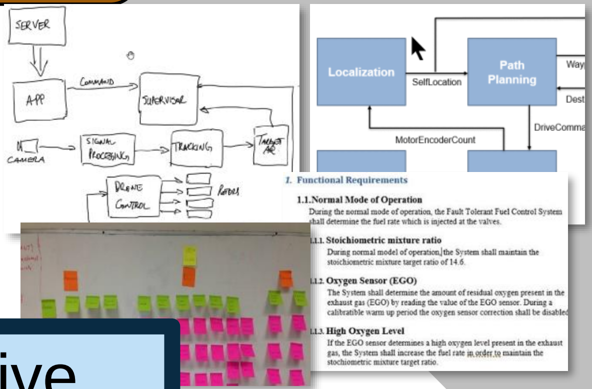
https://en.wikipedia.org/wiki/Tiktaalik

Trend: Some rumblings in the force

MAB Breakout session 2012 on *System Architecture*

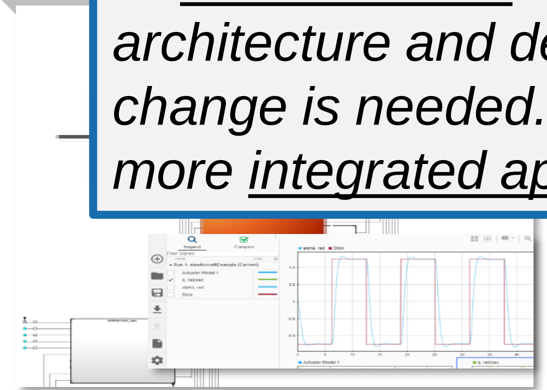
Why the discontent?

Stakeholder Needs



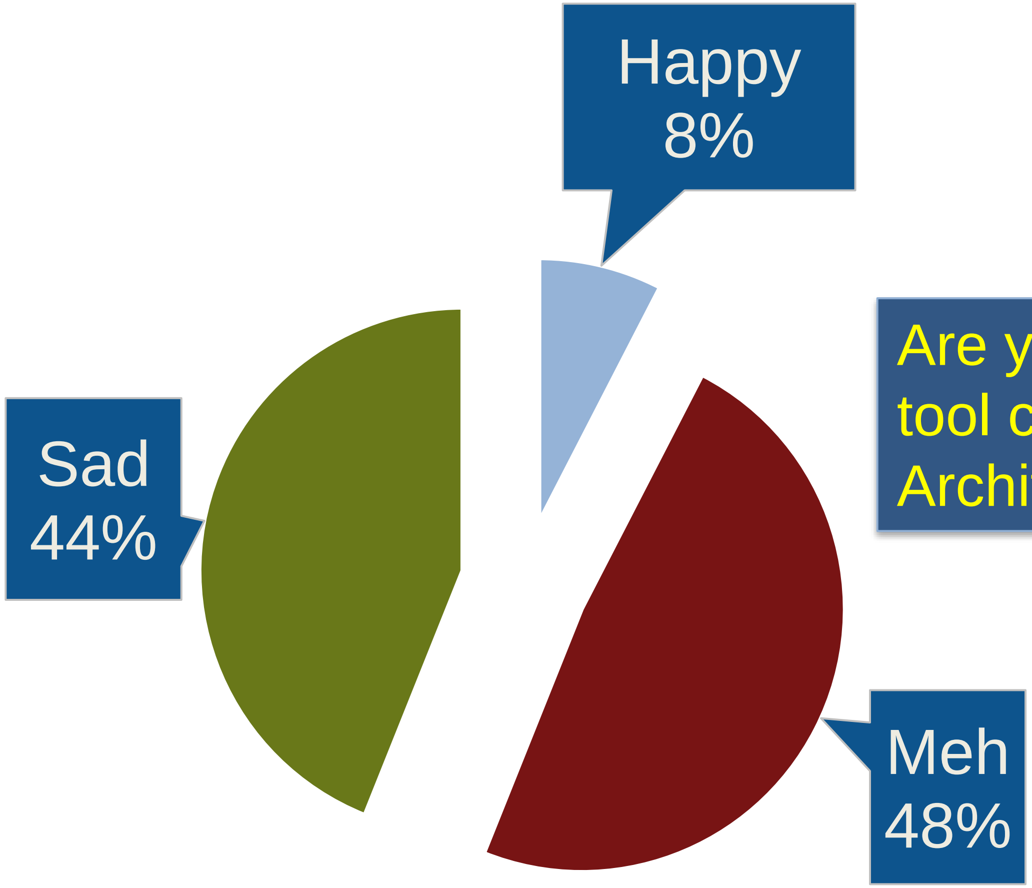
Descriptive Architectures

Implementations



Customer quote:
"We have tried to build the architecture model in SysML and connect it to the design in Simulink … … <u>does not work</u> without rework both in the architecture and design worlds whenever a change is needed. It is <u>broken</u> and we need a more <u>integrated approach</u>"

Survey @ *Modeling System Architecture* Breakout
Newton MAB - 2018

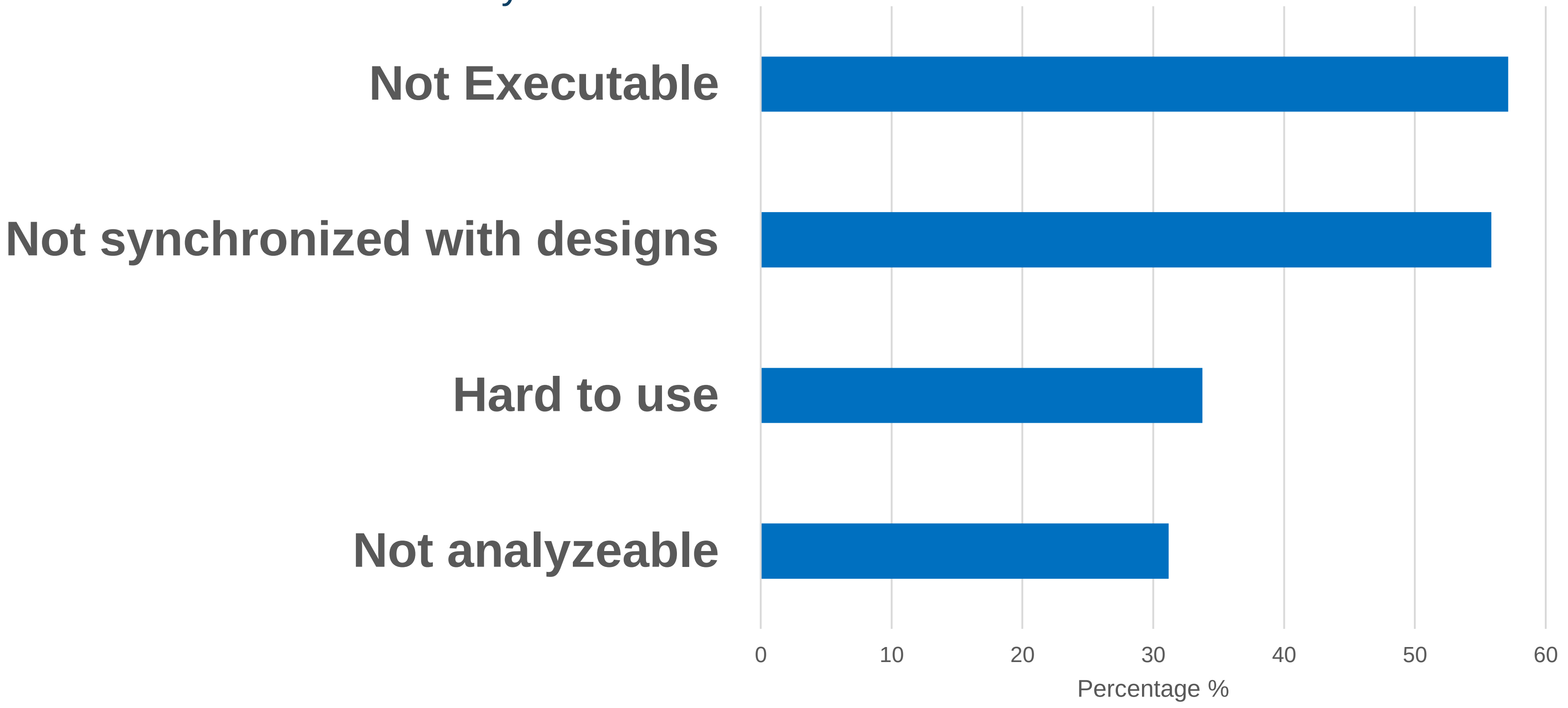


Are you happy with your current tool choices for Modeling System Architecture?

More specifically, what are the pains?

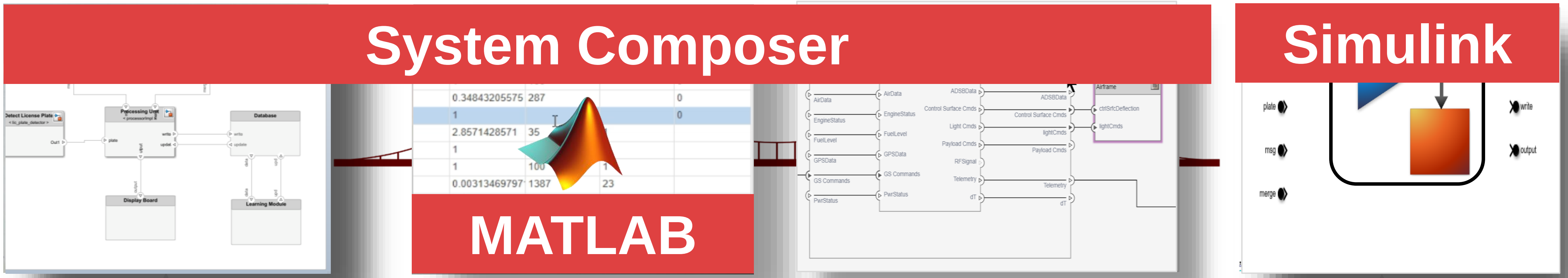
“We do not like our current System Architecture solution because they are:”

Newton MAB Survey 2019



Strategy: Build an MBSE Ecosystem that fits with MBD

Be Intuitive | Facilitate Analysis | Tackle Complexity | Enable Implementation



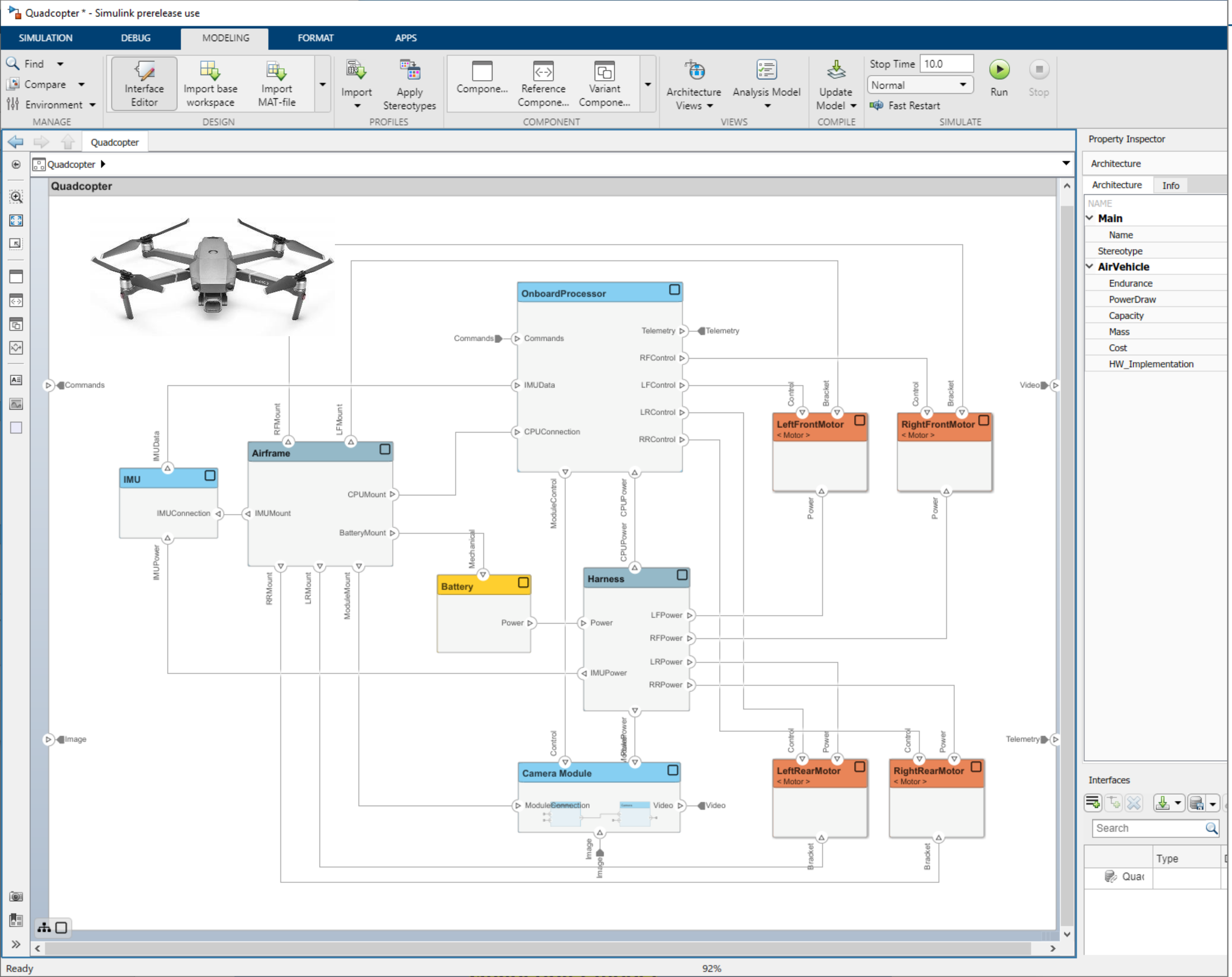
System Composer

MATLAB

Simulink

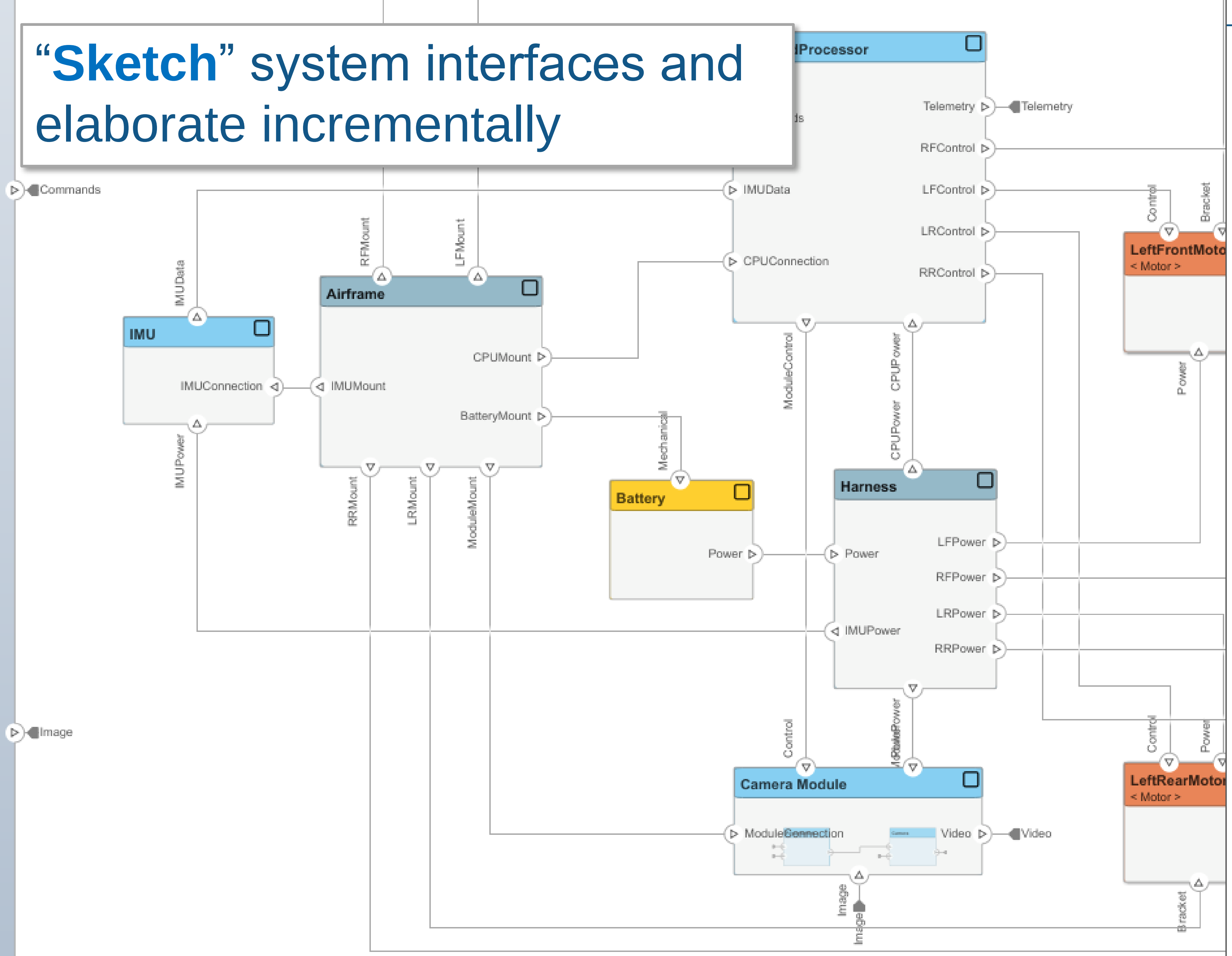
Requirements Coverage Reporting and Impact Analysis

Simulink Requirements


Quadcopter * - Simulink prerelease use
SIMULATION
DEBUG
MODELING
FORMAT
APPS
Find
Compare
Environment
MANAGE
Interface Editor
Import base workspace
Import MAT-file
DESIGN
Import
Apply Stereotypes
PROFILES
Compone...
Reference Compone...
Variant Compone...
COMPONENT
Architecture Views
Analysis Model
VIEWS
Update Model
COMPILE
Stop Time
10.0
Normal
Fast Restart
Run
Stop
SIMULATE
Quadcopter
OnboardProcessor
Commands
Telemetry
RFControl
IMUData
LFControl
LRControl
CPUConnection
RRControl
IMU
IMUConnection
Airframe
IMUMount
CPUMount
BatteryMount
Battery
Power
Harness
LFPower
RFPower
LRPower
RRPower
IMUPower
LeftFrontMotor
< Motor >
RightFrontMotor
< Motor >
LeftRearMotor
< Motor >
RightRearMotor
< Motor >
Camera Module
ModuleConnection
Video
Image
Property Inspector
Architecture
Info
NAME
Main
Name
Stereotype
AirVehicle
Endurance
PowerDraw
Capacity
Mass
Cost
HW_Implementation
Interfaces
Search
Type
Ready
92%

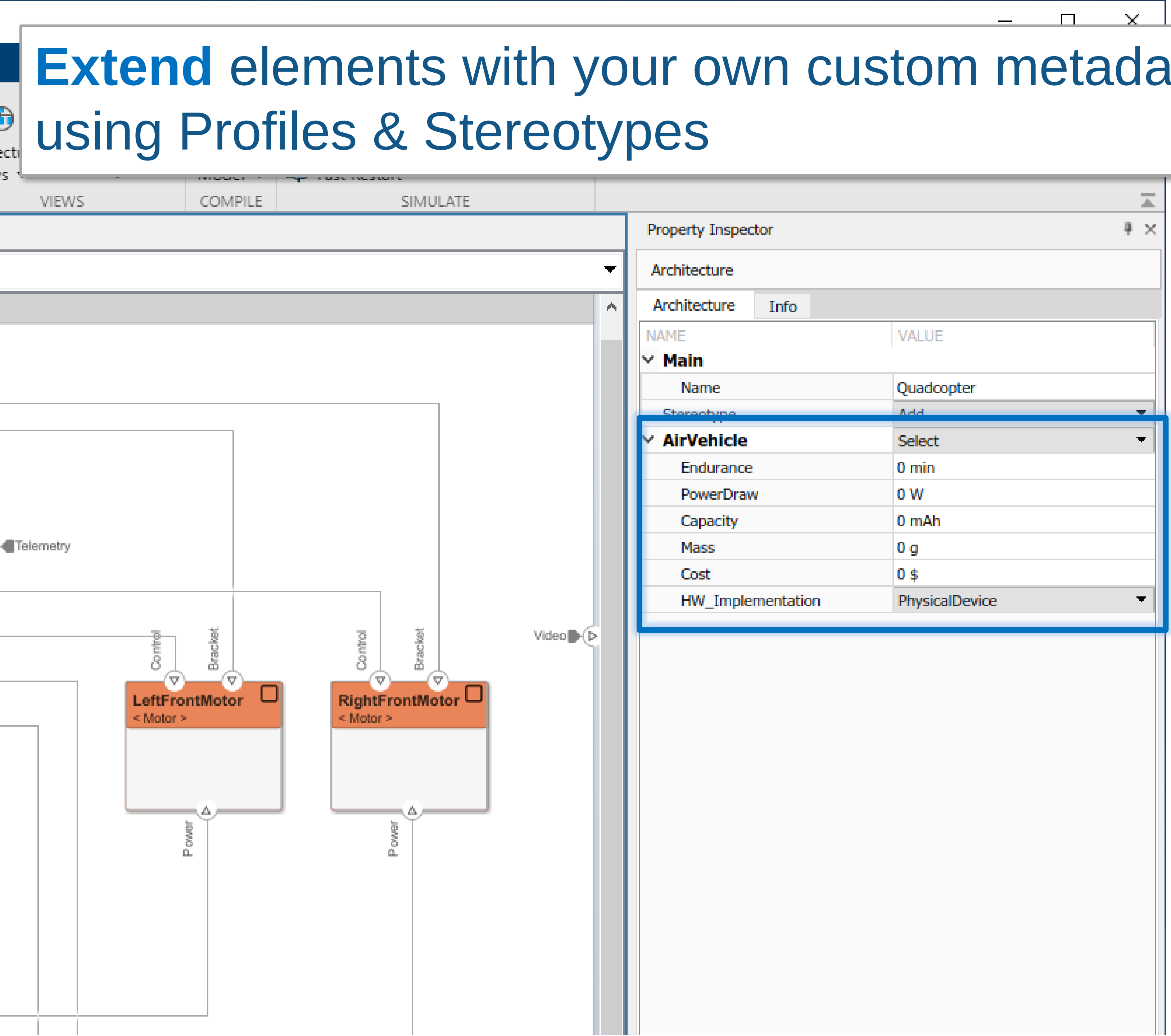
R2019a

“**Sketch**” system interfaces and elaborate incrementally



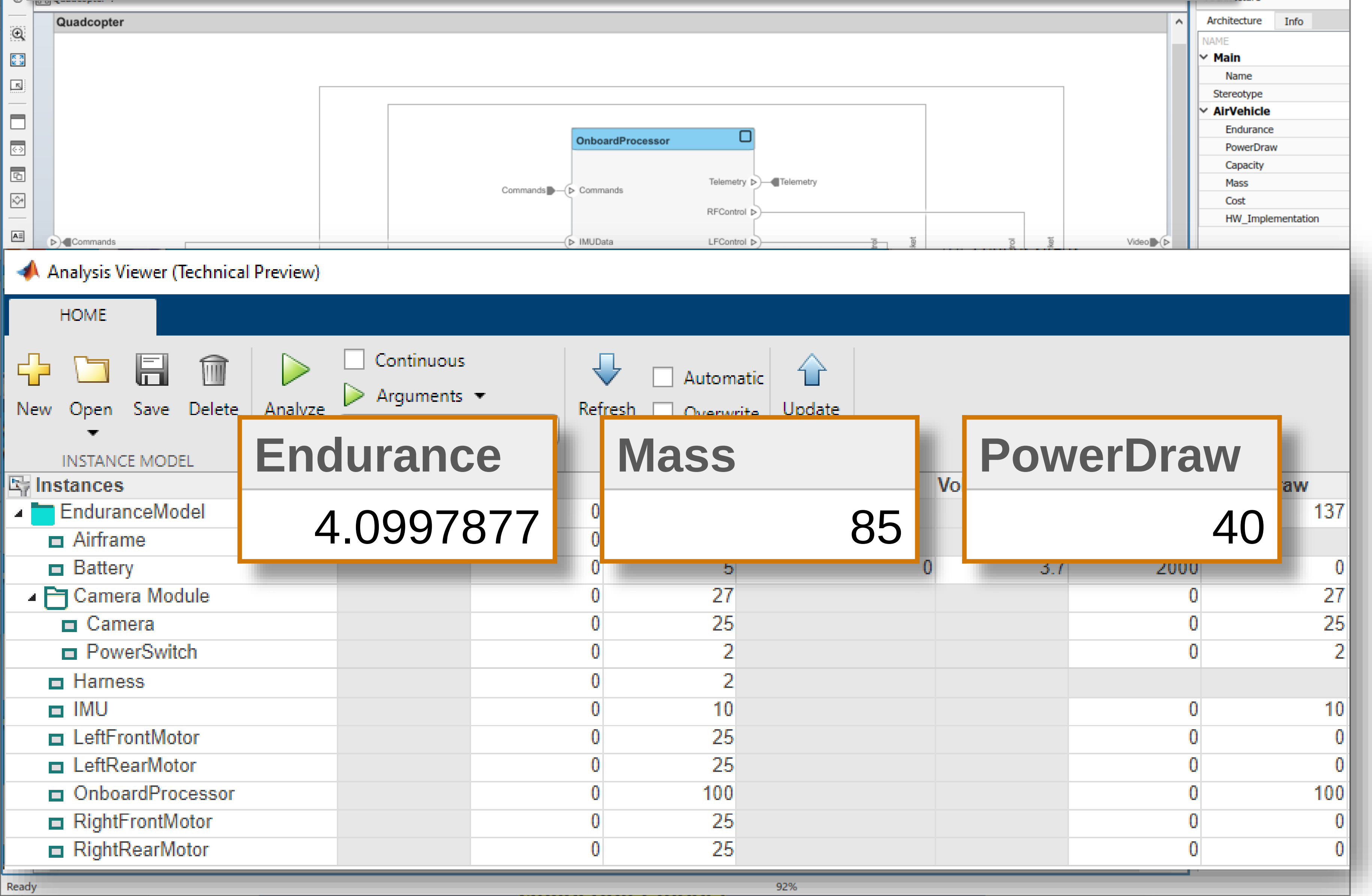
R2019a

Extend elements with your own custom metadata using Profiles & Stereotypes



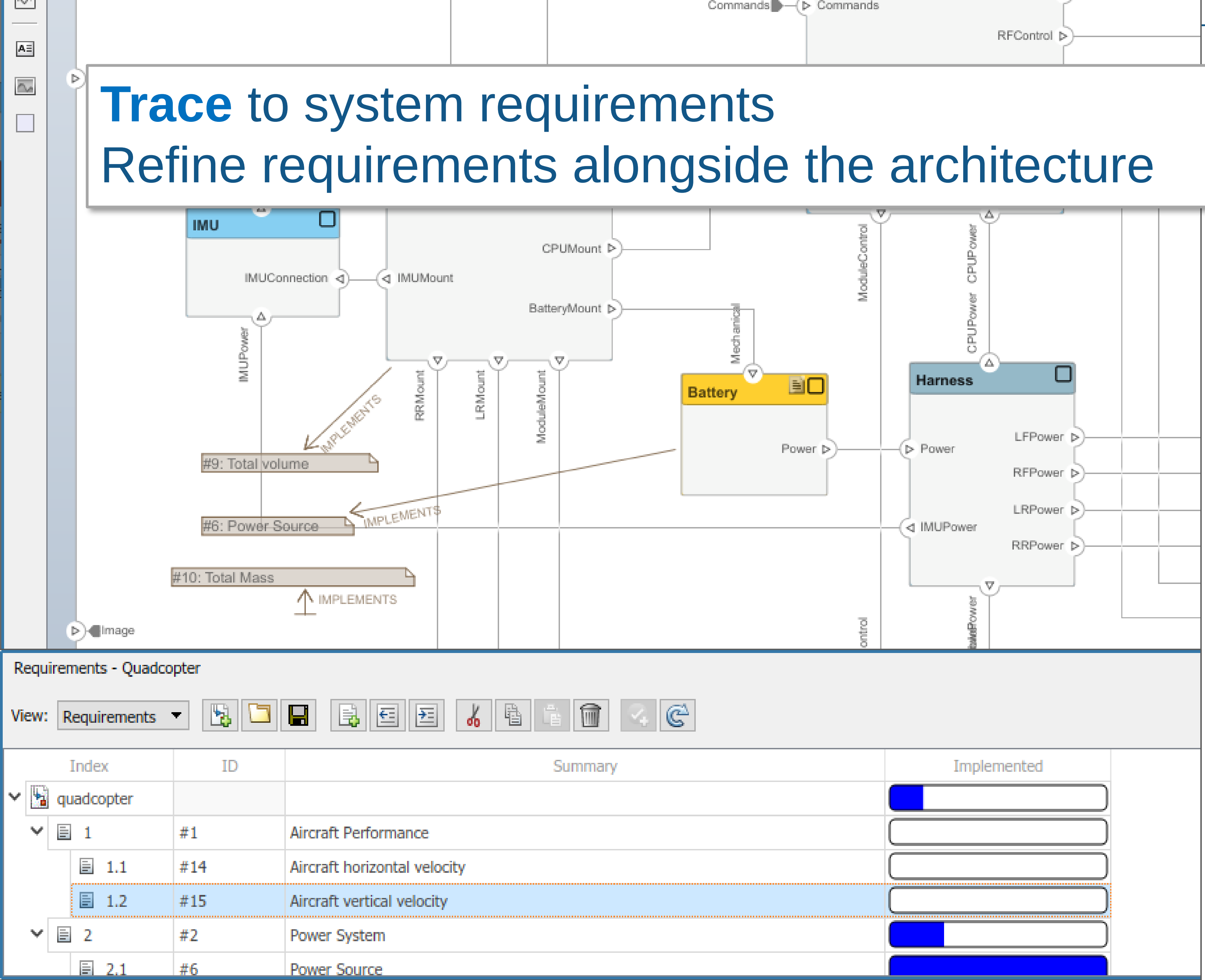
R2019a

Analyze system characteristics and quantitatively evaluate choices using MATLAB



R2019a

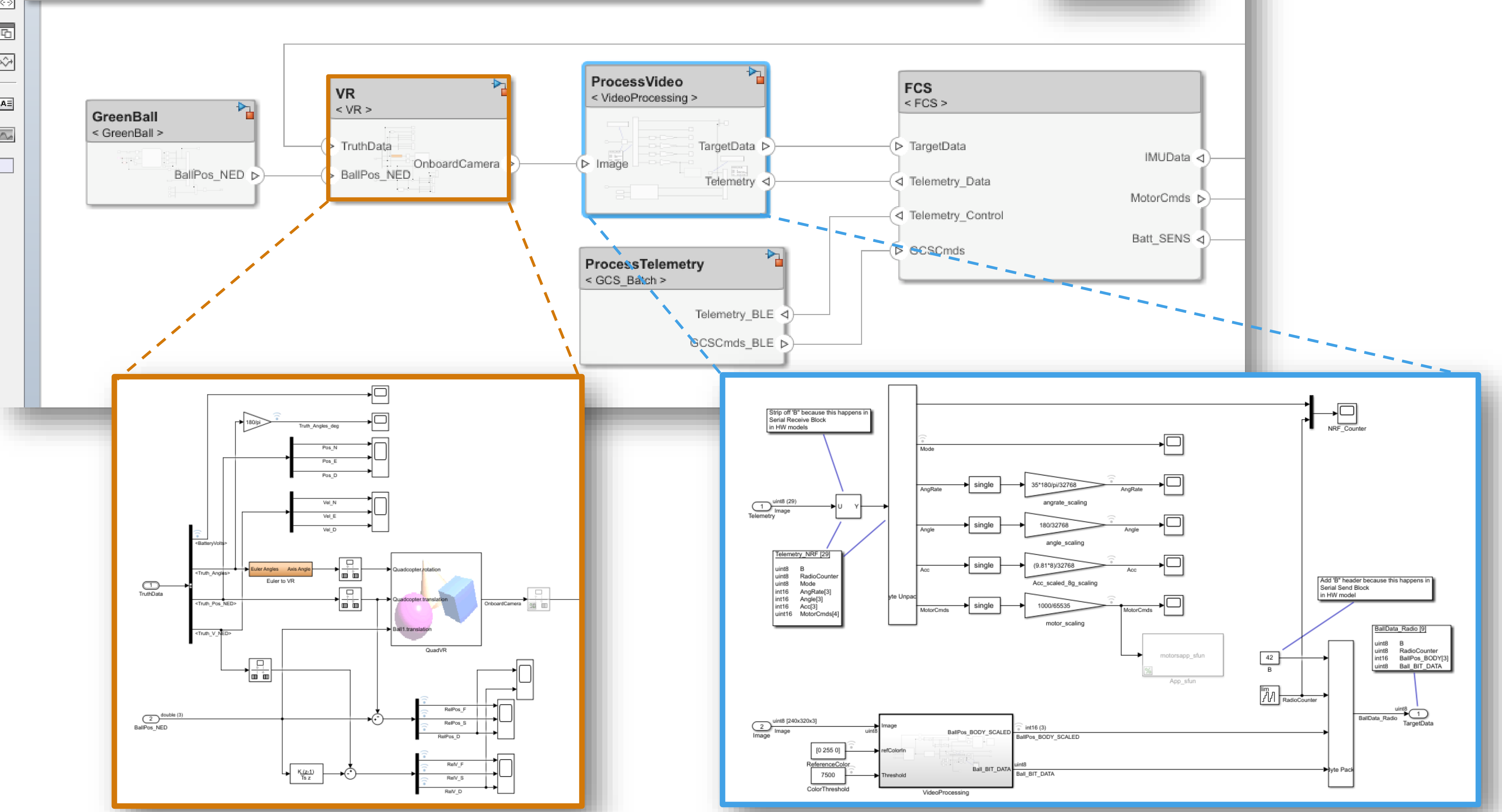
Trace to system requirements
Refine requirements alongside the architecture



R2019a

With Simulink Requirements

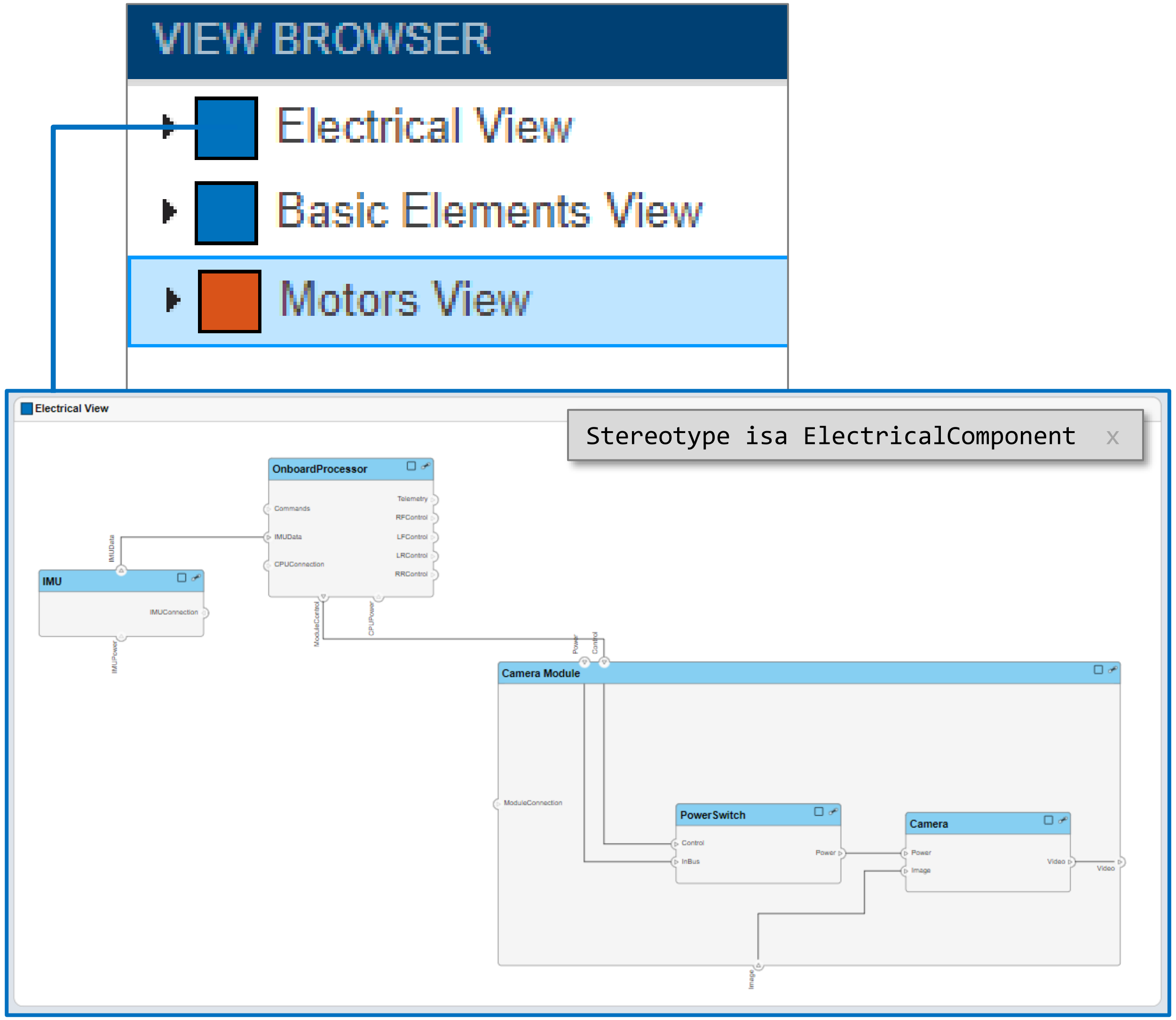
Link design models to components and ensure consistent interfaces



R2019a

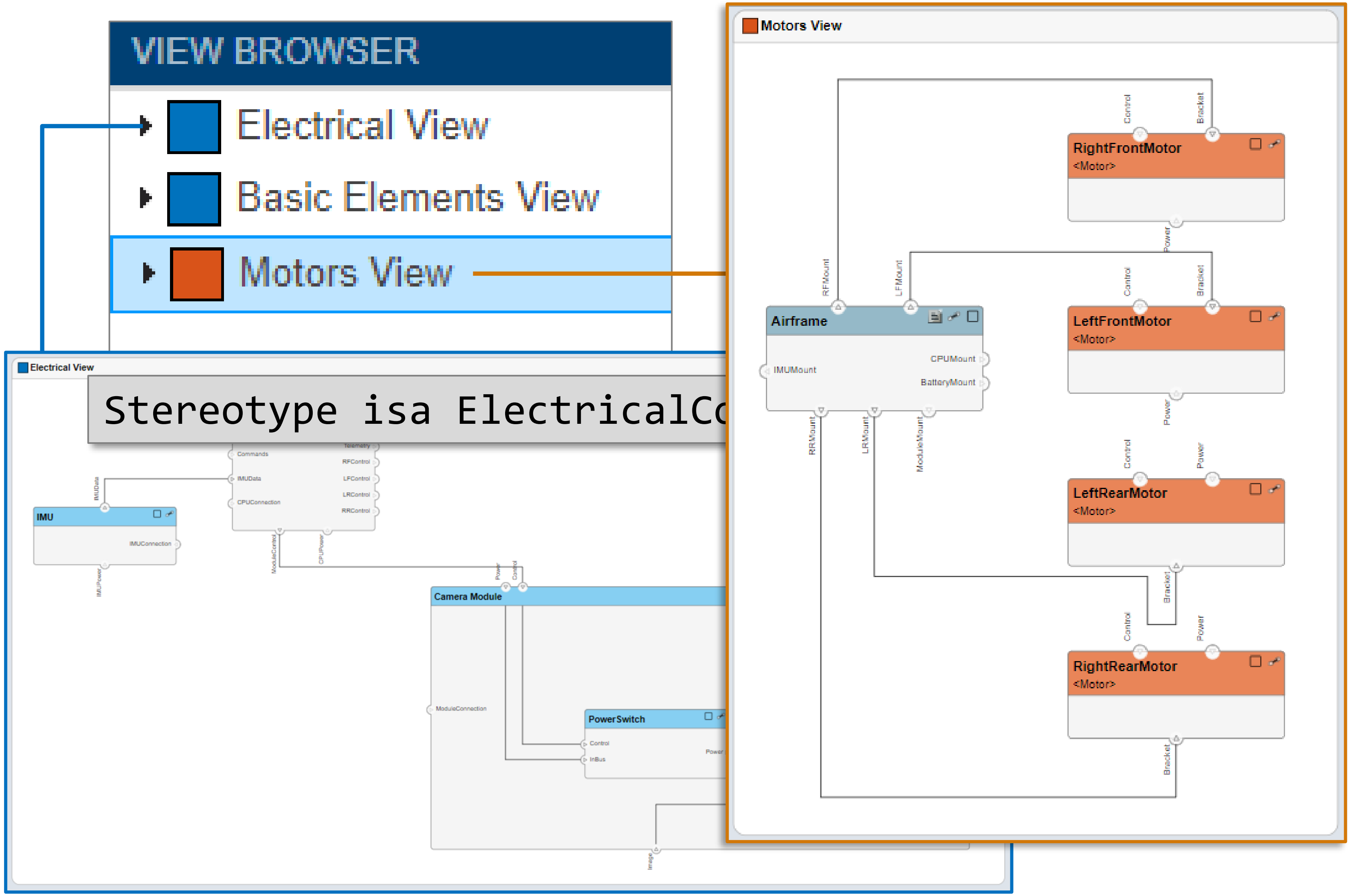
Simplify the complex with Filters and autogenerated Views

R2019b



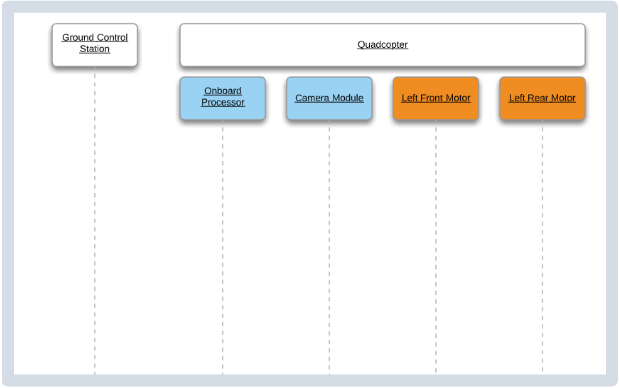
Simplify the complex with Filters and autogenerated Views

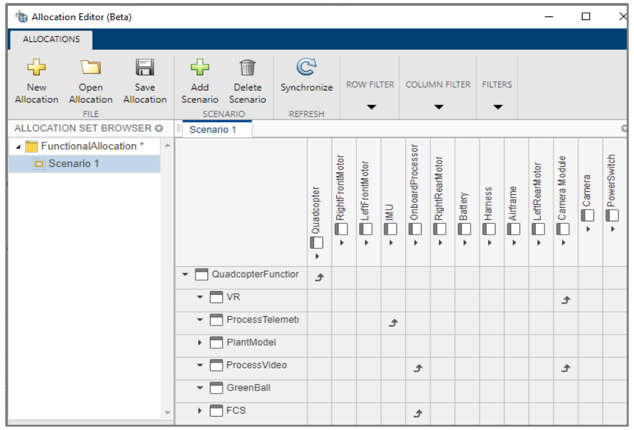
R2019b

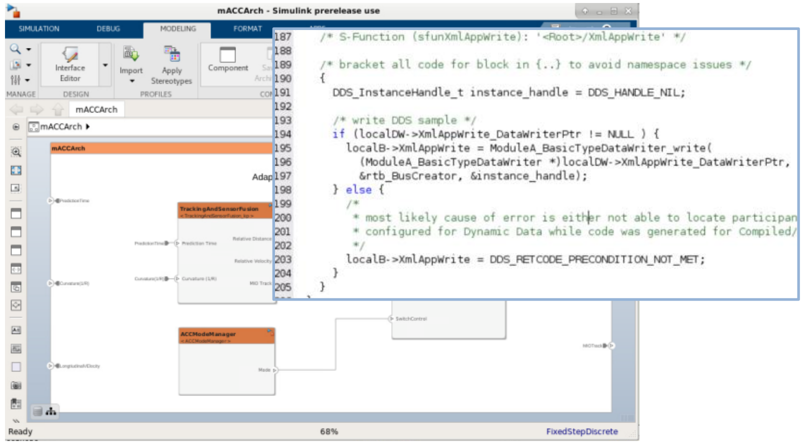


And we are only getting started. Coming soon:

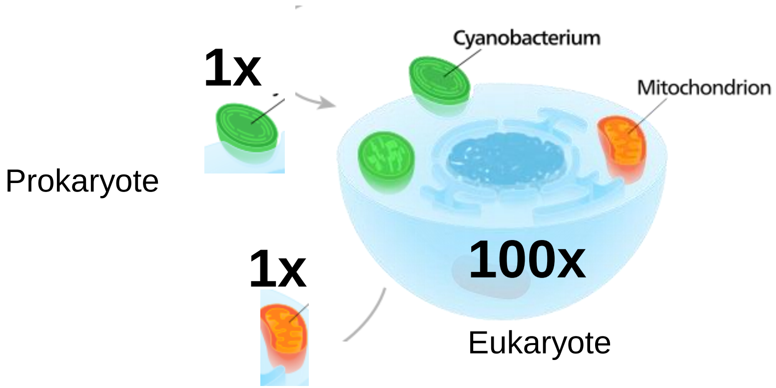
- Behavior modeling using Sequence Diagrams
- Architecture Allocations though Analysis (e.g. Logical to Physical)
- Software Architecture Modeling
  - Link to AUTOSAR (R2019b)
  - Other middlewares such as DDS
- And much more!





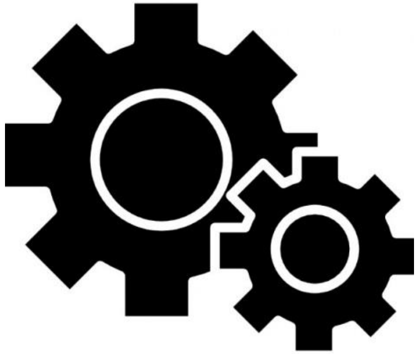


Evolving for **Collaborative Engineering**

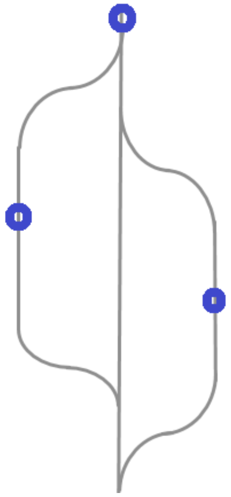


https://en.wikipedia.org/wiki/Symbiogenesis

Trend: An increased demand for Agile team-based workflows



Shared team environment



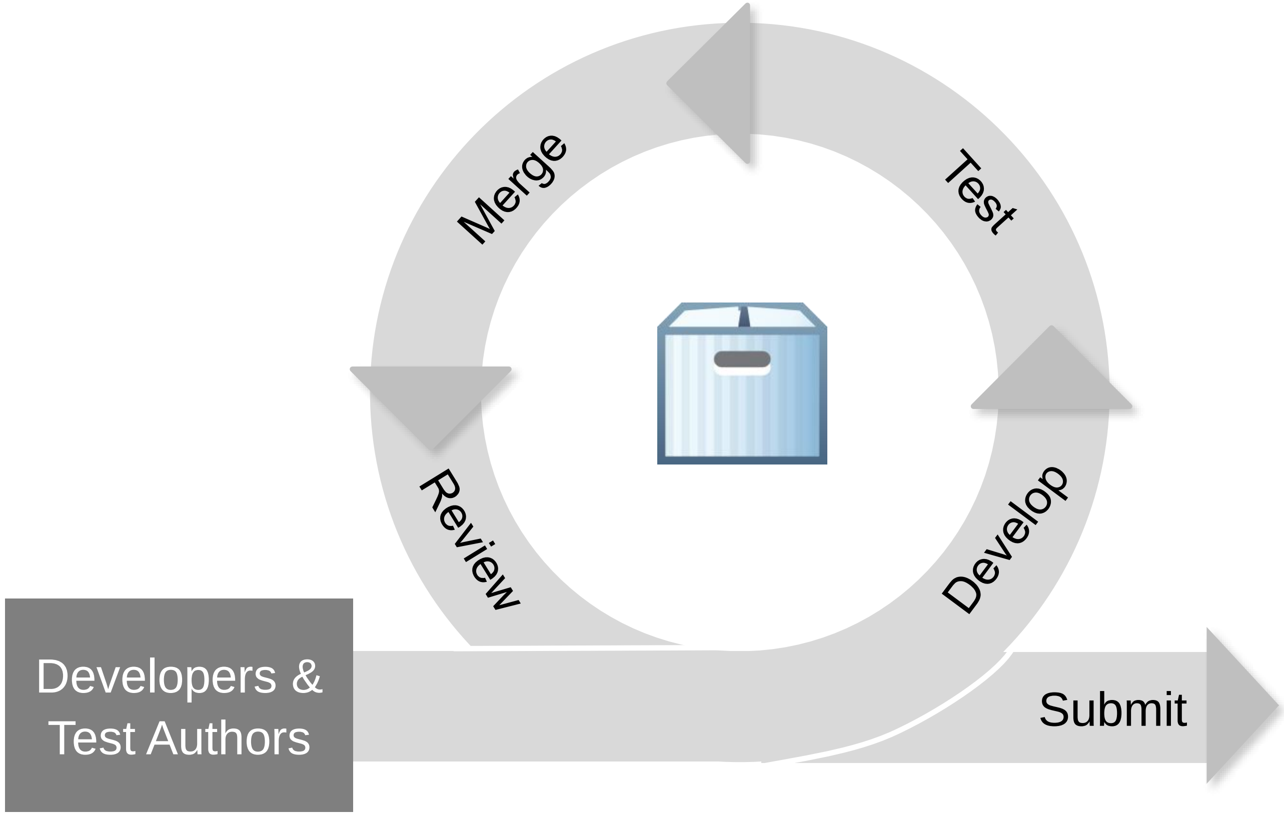
Collaboration

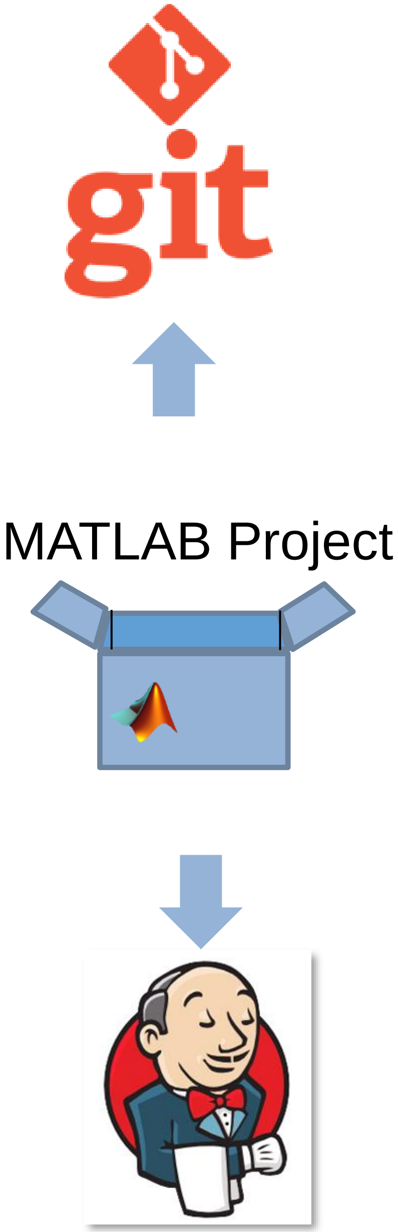
Simulate

Jenkins

Continuous Integration & Test

Strategy: Continued investments to facilitate Continuous Integration as a critical lynch-pin in Agile workflows



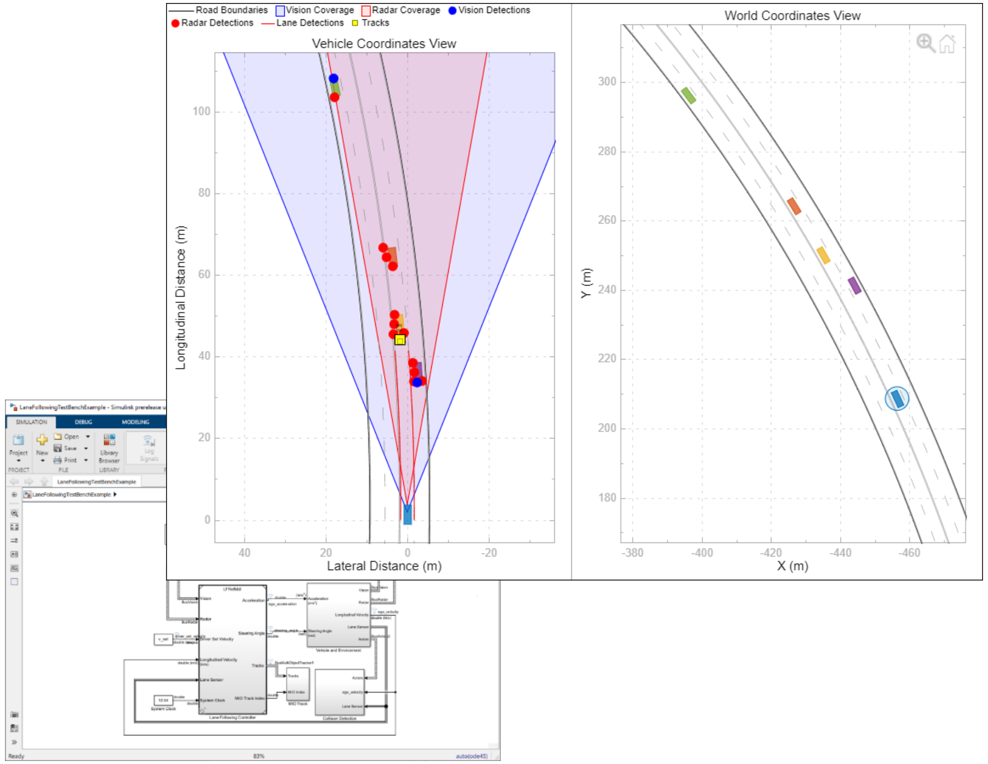


Can I do CI today in Simulink?

Yes, lets consider an example from

R2019b

Lane Following Assist Example

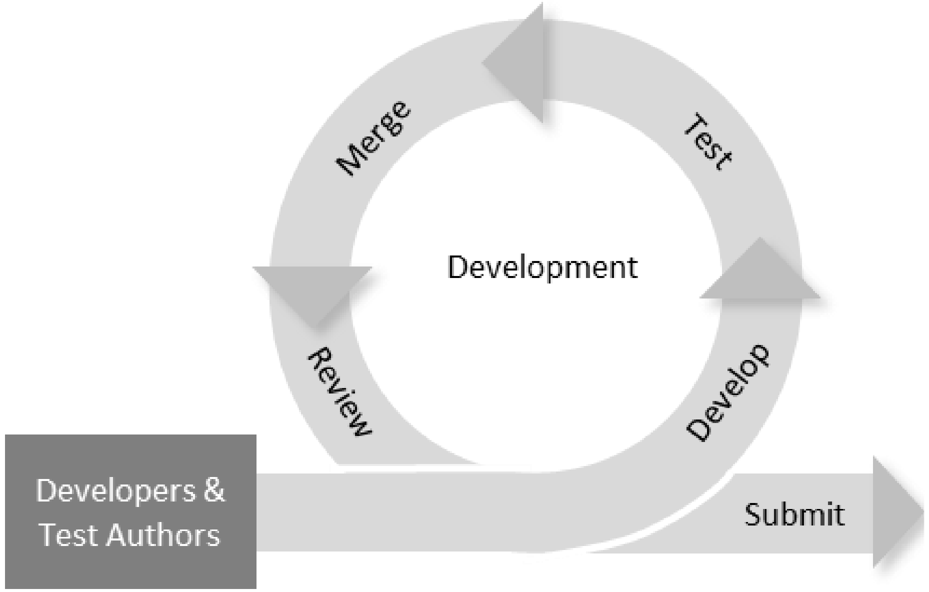


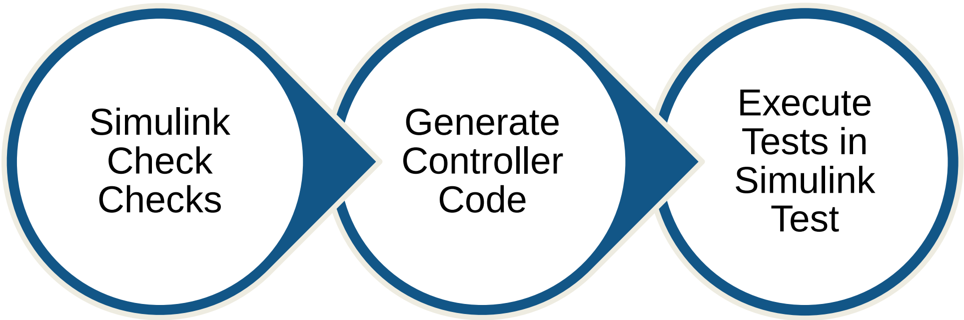
Simulink Check
Checks

SIL Code
Generation

SIL Testing
Simulink Test

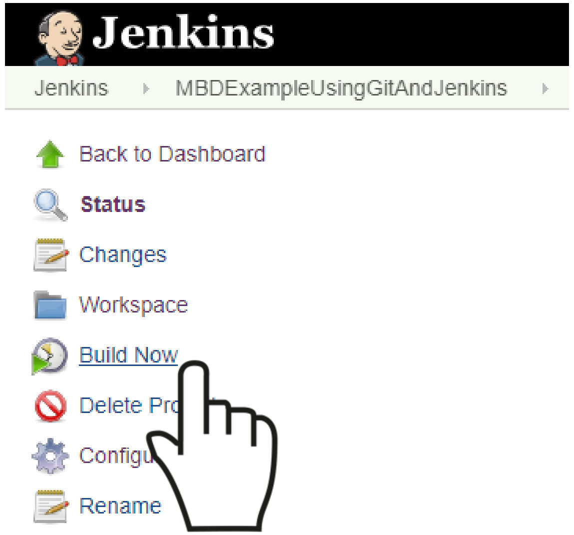
How Does It All Fit Together?





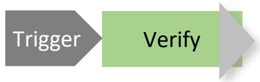
1. Trigger

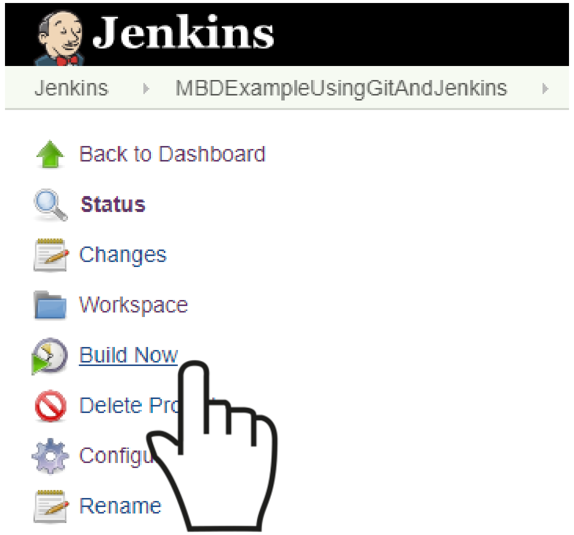
Trigger



1. Trigger

Continuous
Integration

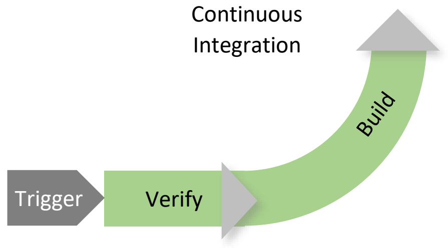


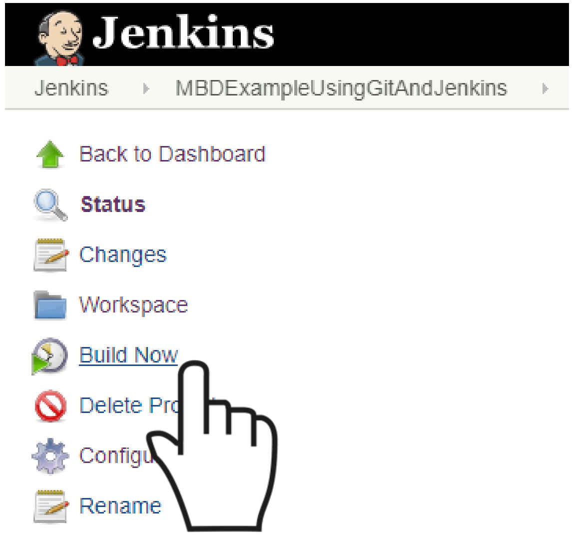



```
Running LaneFollowingModelAdvisorChecks
.
Done LaneFollowingModelAdvisorCheck
```

Simulink Check

1. Trigger





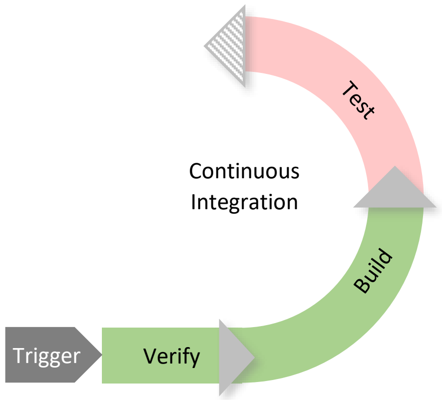

```
Running LaneFollowingModelAdvisorChecks
.
Done LaneFollowingModelAdvisorCheck
```

Simulink Check

```
## Caching model source code
......................................
......................................
### Writing header file rtGetNaN.h
### Writing source file rtGetNaN.cpp
### Writing header file rt_defines.h
### Writing header file rt_nonfinite.h
### Writing source file rt_nonfinite.cpp
```

Code Generation

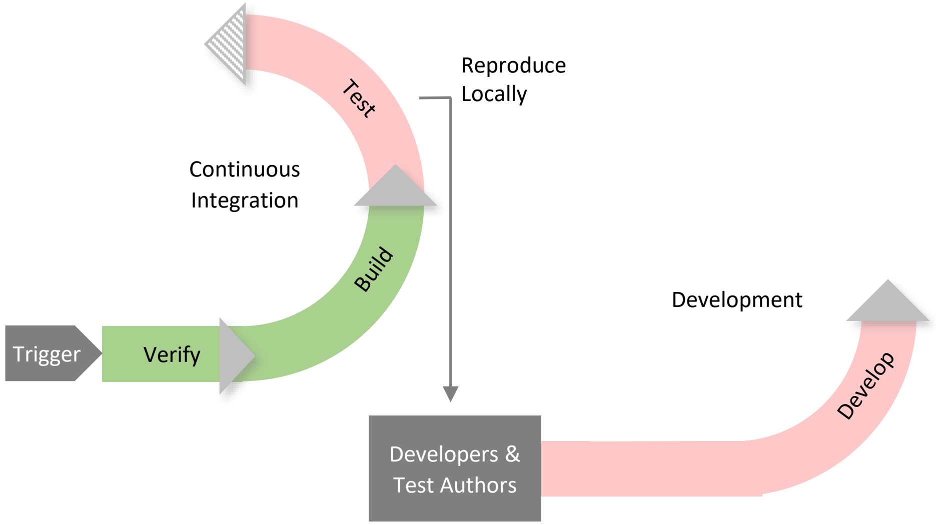
2. Detect



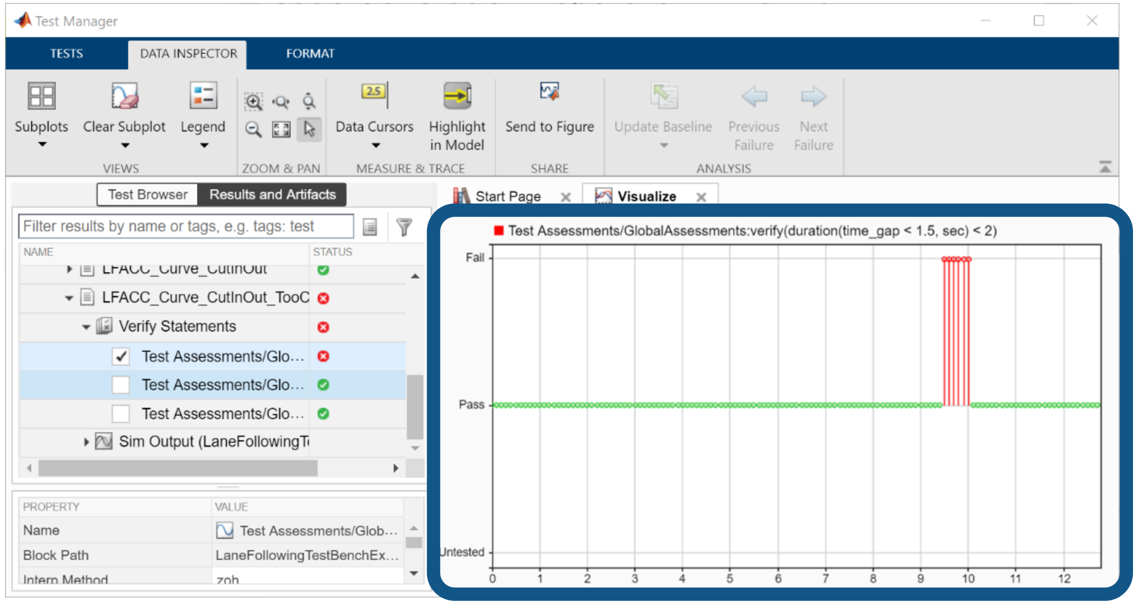

```
Failure Summary:

     Name                                                                    Failed  Incomplete  Reason(s)
    ==================================================================================================================
     LaneFollowingTestScenarios > Scenarios/LFACC_Curve_CutInOut_TooClose      X                 Failed by verification.
ERROR: MATLAB error Exit Status: 0x00000001
Build step 'Run MATLAB Tests' changed build result to FAILURE
Finished: FAILURE
```

Simulink Test

3. Reproduce

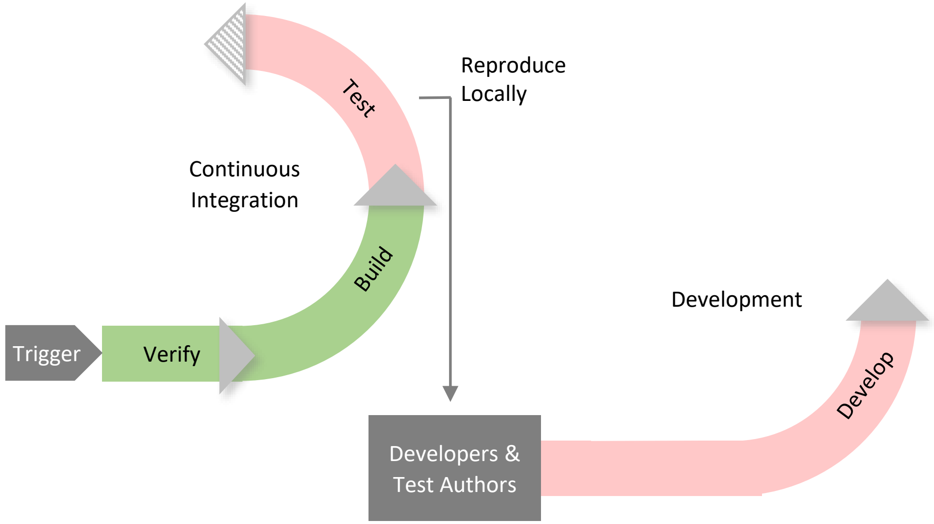


Reproduce Locally
Test
Continuous Integration
Build
Trigger
Verify
Development
Develop
Developers & Test Authors

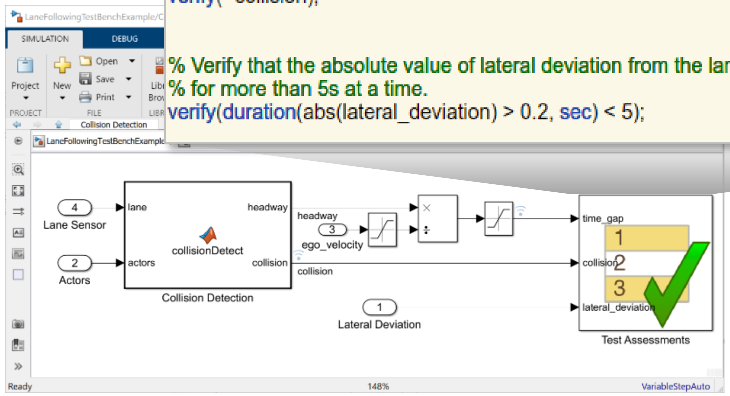


Test Manager
TESTS
DATA INSPECTOR
FORMAT
Subplots
Clear Subplot
Legend
VIEWS
ZOOM & PAN
Data Cursors
Highlight in Model
MEASURE & TRACE
Send to Figure
SHARE
Update Baseline
Previous Failure
Next Failure
ANALYSIS
Test Browser
Results and Artifacts
Filter results by name or tags, e.g. tags: test
NAME
STATUS
LFACC_Curve_CutInOut
LFACC_Curve_CutInOut_TooC
Verify Statements
Test Assessments/Glo…
Test Assessments/Glo…
Test Assessments/Glo…
Sim Output (LaneFollowingT
PROPERTY
VALUE
Name
Test Assessments/Glob…
Block Path
LaneFollowingTestBenchEx…
Interp Method
zoh
Start Page
Visualize
Test Assessments/GlobalAssessments:verify(duration(time_gap < 1.5, sec) < 2)
Fail
Pass
Untested
0
1
2
3
4
5
6
7
8
9
10
11
12

4. Fix Locally

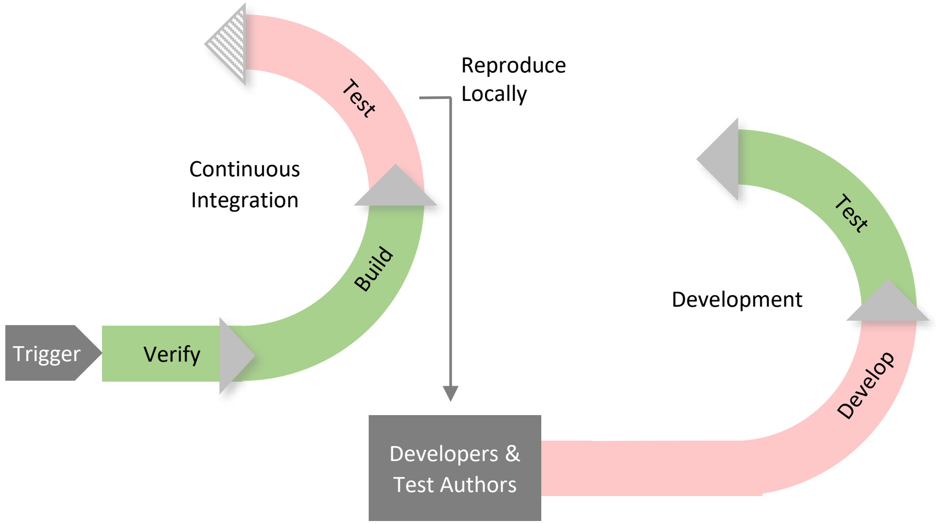


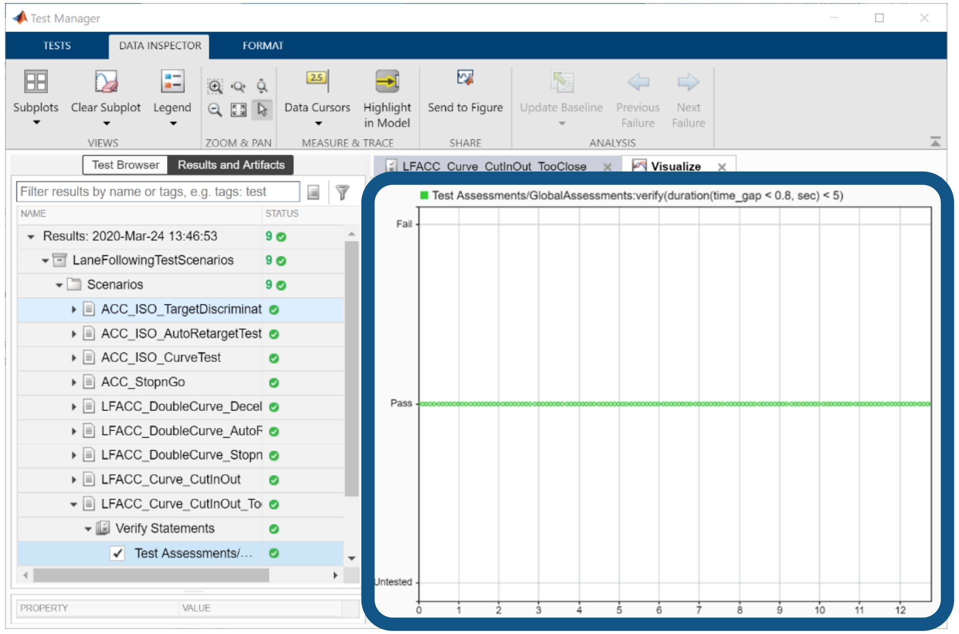

```
GlobalAssessments

% Ensure that the time gap between the ego vehicle and lead vehicle does not dip below
% 1.5s for more than 2s at a time.
verify(duration(time_gap < 1.5, sec) < 2);

% Verify that no collision was detected
verify(~collision);

% Verify that the absolute value of lateral deviation from the lane centerline does not exceed 0.2m
% for more than 5s at a time.
verify(duration(abs(lateral_deviation) > 0.2, sec) < 5);
```



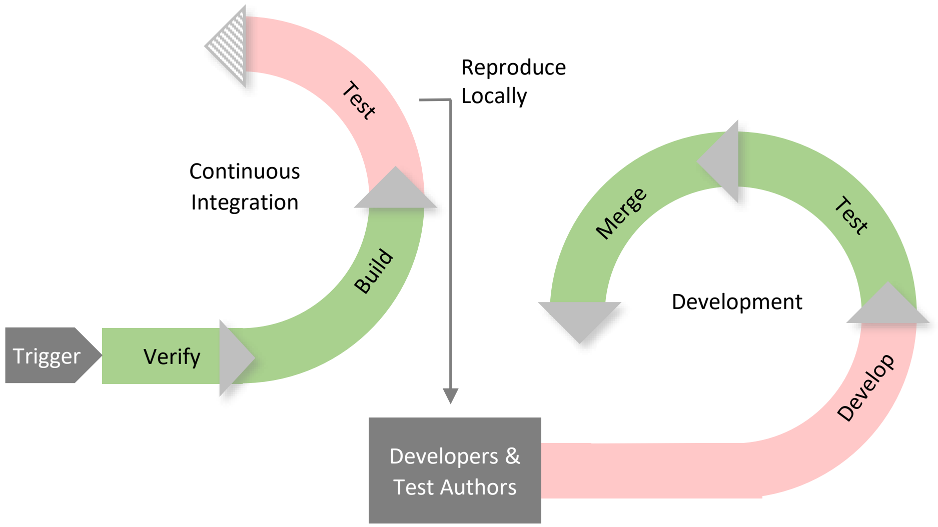

```
GlobalAssessments

% Ensure that the time gap between the ego vehicle and lead vehicle does not dip below
% 0.8s for more than 5s at a time.
verify(duration(time_gap < 0.8, sec) < 5);

% Verify that no collision was detected
verify(~collision);

% Verify that the absolute value of lateral deviation from the lane centerline does not exceed 0.2m
% for more than 5s at a time.
verify(duration(abs(lateral_deviation) > 0.2, sec) < 5);
```

5. Test Locally

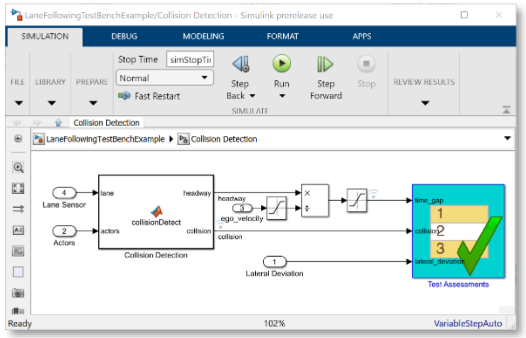


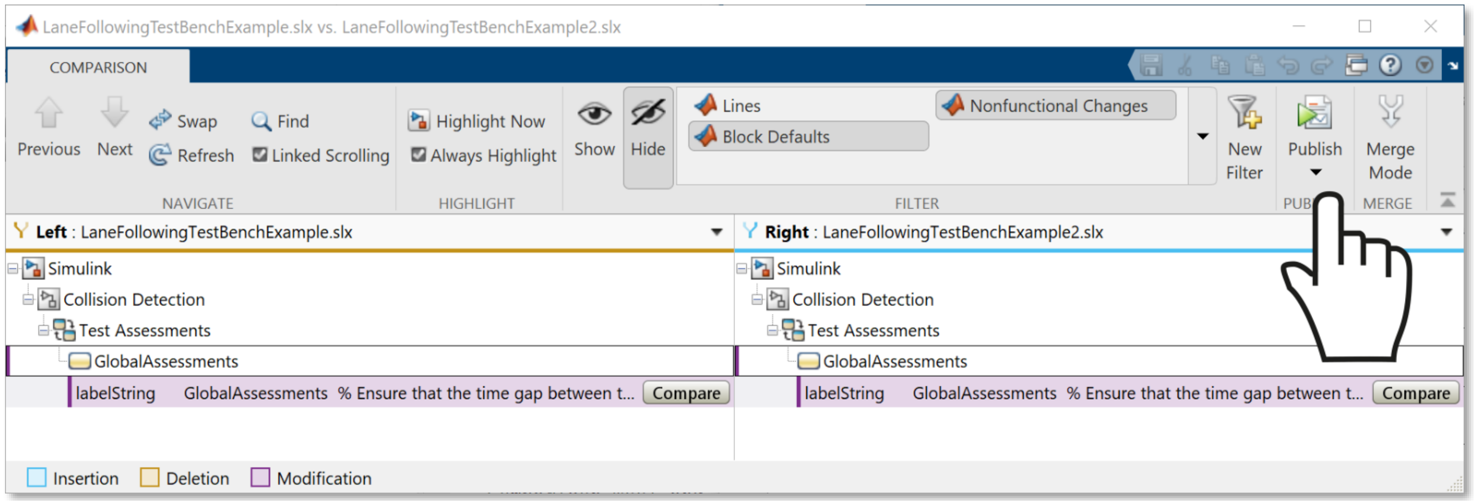


6. Merge



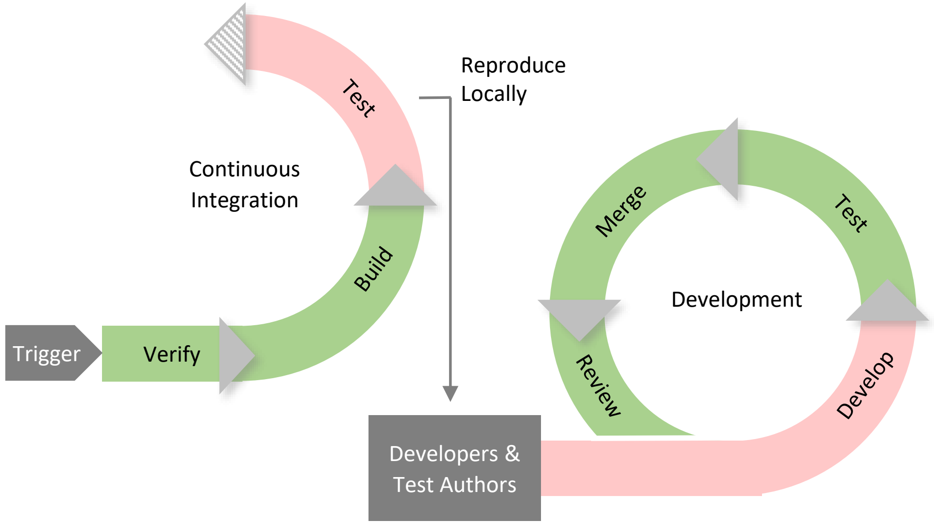
Reproduce
Locally
Test
Continuous
Integration
Build
Trigger
Verify
Merge
Test
Development
Develop
Developers &
Test Authors



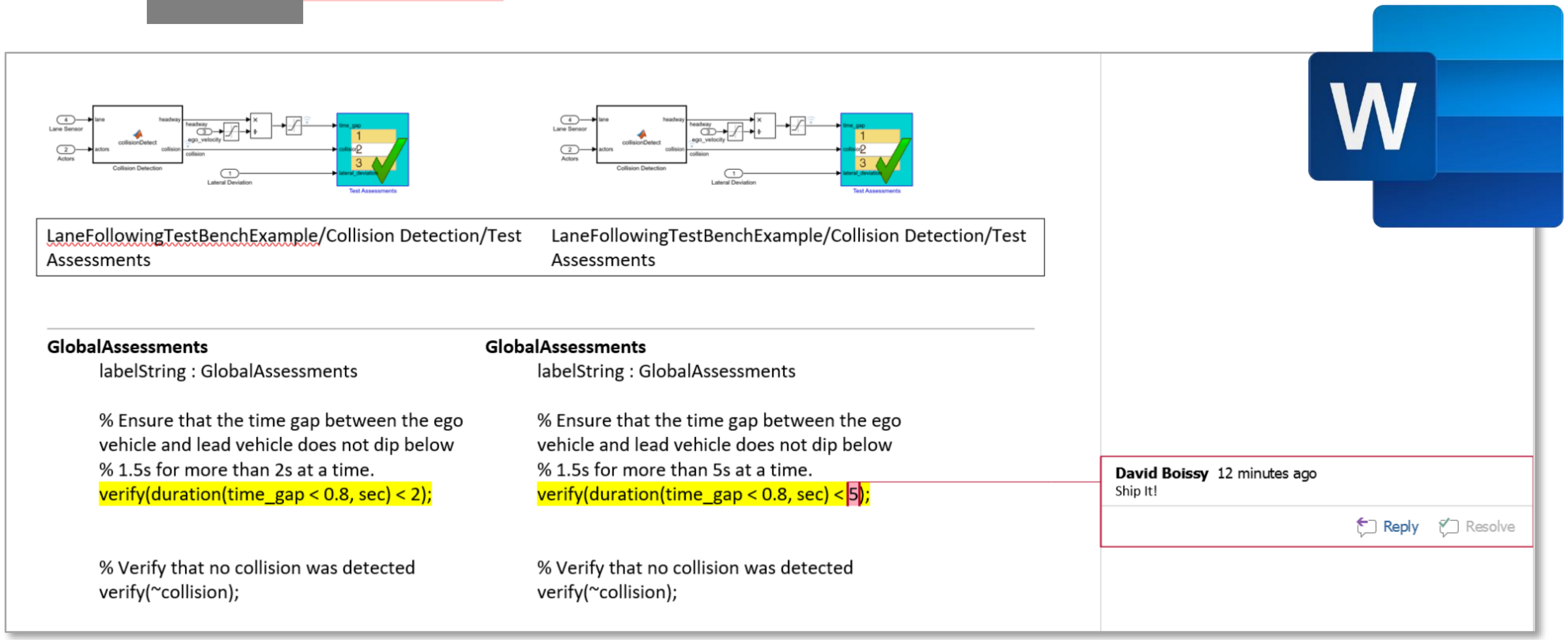


LaneFollowingTestBenchExample.slx vs. LaneFollowingTestBenchExample2.slx
COMPARISON
Previous
Next
Swap
Refresh
Find
Linked Scrolling
Highlight Now
Always Highlight
Show
Hide
Lines
Block Defaults
Nonfunctional Changes
New Filter
Publish
Merge Mode
NAVIGATE
HIGHLIGHT
FILTER
MERGE
Left : LaneFollowingTestBenchExample.slx
Right : LaneFollowingTestBenchExample2.slx
Simulink
Collision Detection
Test Assessments
GlobalAssessments
labelString GlobalAssessments % Ensure that the time gap between t... Compare
Insertion
Deletion
Modification

6. Review

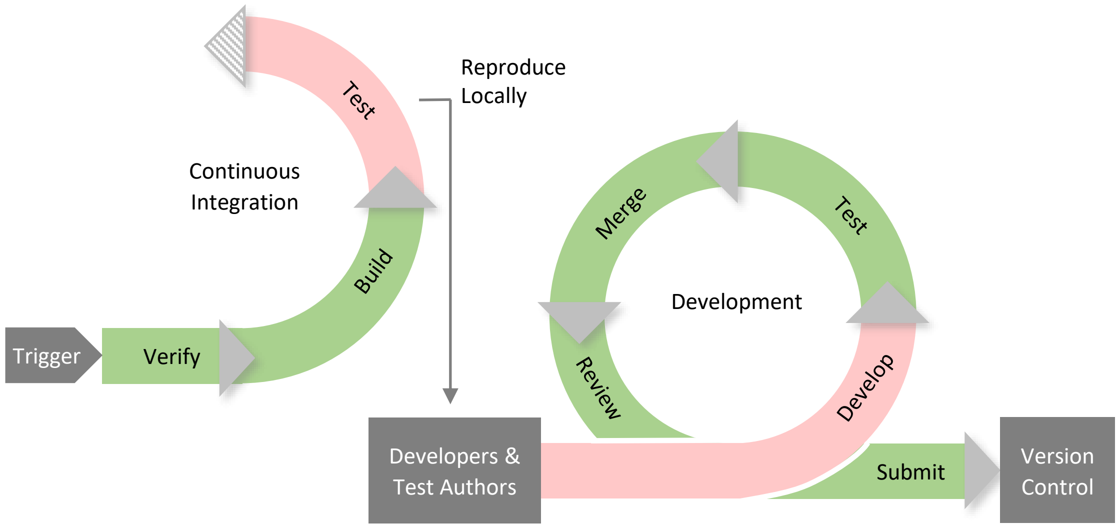


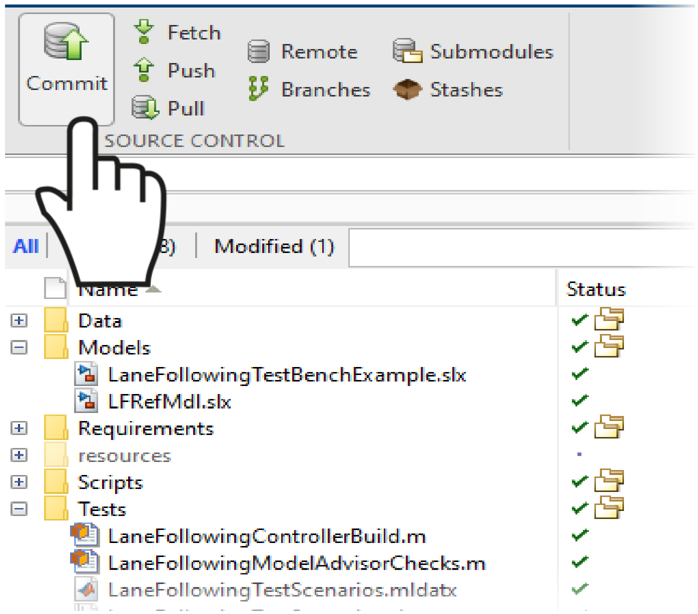
Test
Reproduce Locally
Continuous Integration
Build
Trigger
Verify
Merge
Test
Development
Review
Develop
Developers & Test Authors

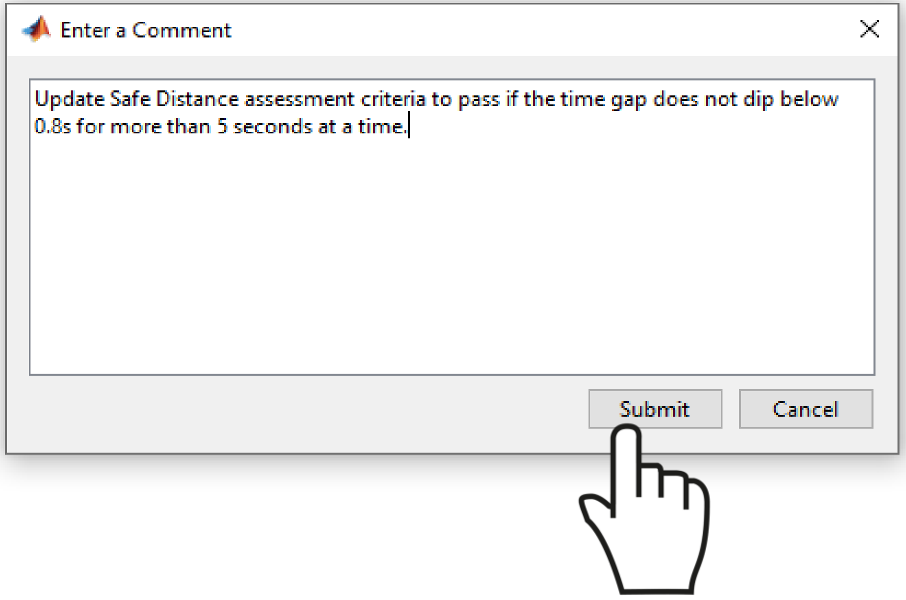


LaneFollowingTestBenchExample/Collision Detection/Test Assessments
LaneFollowingTestBenchExample/Collision Detection/Test Assessments
GlobalAssessments
labelString : GlobalAssessments
% Ensure that the time gap between the ego vehicle and lead vehicle does not dip below
% 1.5s for more than 2s at a time.
verify(duration(time_gap < 0.8, sec) < 2);
% Verify that no collision was detected
verify(~collision);
GlobalAssessments
labelString : GlobalAssessments
% Ensure that the time gap between the ego vehicle and lead vehicle does not dip below
% 1.5s for more than 5s at a time.
verify(duration(time_gap < 0.8, sec) < 5);
% Verify that no collision was detected
verify(~collision);
David Boissy 12 minutes ago
Ship It!
Reply
Resolve

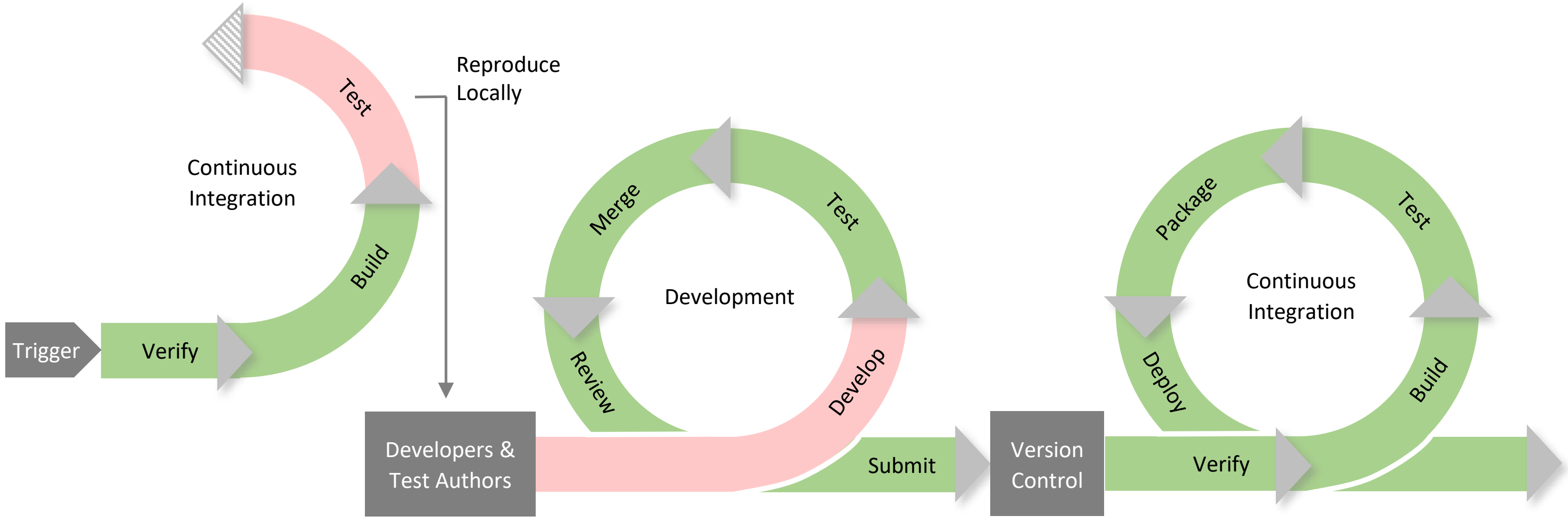
7. Commit


Reproduce
Locally
Test
Continuous
Integration
Build
Trigger
Verify
Merge
Test
Development
Review
Develop
Developers &
Test Authors
Submit
Version
Control


Commit
Fetch
Push
Pull
Remote
Branches
Submodules
Stashes
SOURCE CONTROL
All
Modified (1)
Name
Status
Data
Models
LaneFollowingTestBenchExample.slx
LFRefMdl.slx
Requirements
resources
Scripts
Tests
LaneFollowingControllerBuild.m
LaneFollowingModelAdvisorChecks.m
LaneFollowingTestScenarios.mldatx


Enter a Comment
Update Safe Distance assessment criteria to pass if the time gap does not dip below 0.8s for more than 5 seconds at a time.
Submit
Cancel

8. Verify, Build, Test



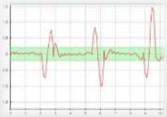

```
Finished: SUCCESS
```

Continuous Integration Success is within your reach

1 Tooling

Jenkins Plugin

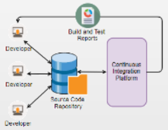
MATLAB Unit



Simulink Test

2 Documentation

Technical Article



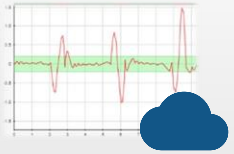
Documentation Hub

Solutions Page

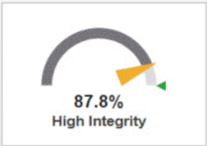
3 Future

Pipeline

Server Workflows



Test Results Online



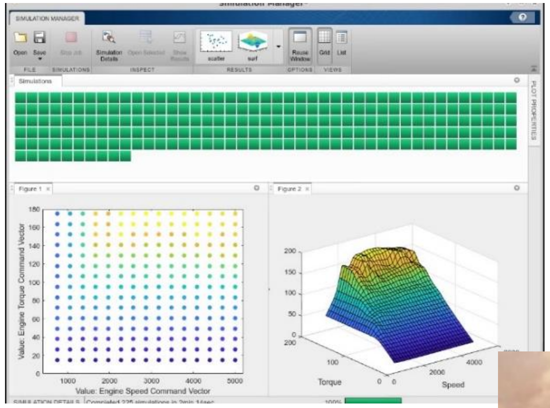
Dashboards

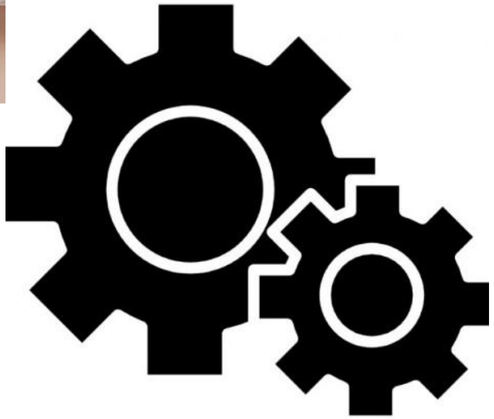
Lets go back to the broad forces that shape our platform evolution

1. Simulation Scale

YOU!

3. Collaborative Engineering





Q&A

Please contact us with questions

mani@mathworks.com

MathWorks®