



MathWorks 

@MathWorks

Share the EXPO experience
#MATLABEXPO



What is your favourite MATLAB command?



MATLAB EXPO

 UNITED KINGDOM

10th October 2024 | Silverstone

Accelerating Product Development Through Simulation

Dr Lachlan Jardine, MathWorks



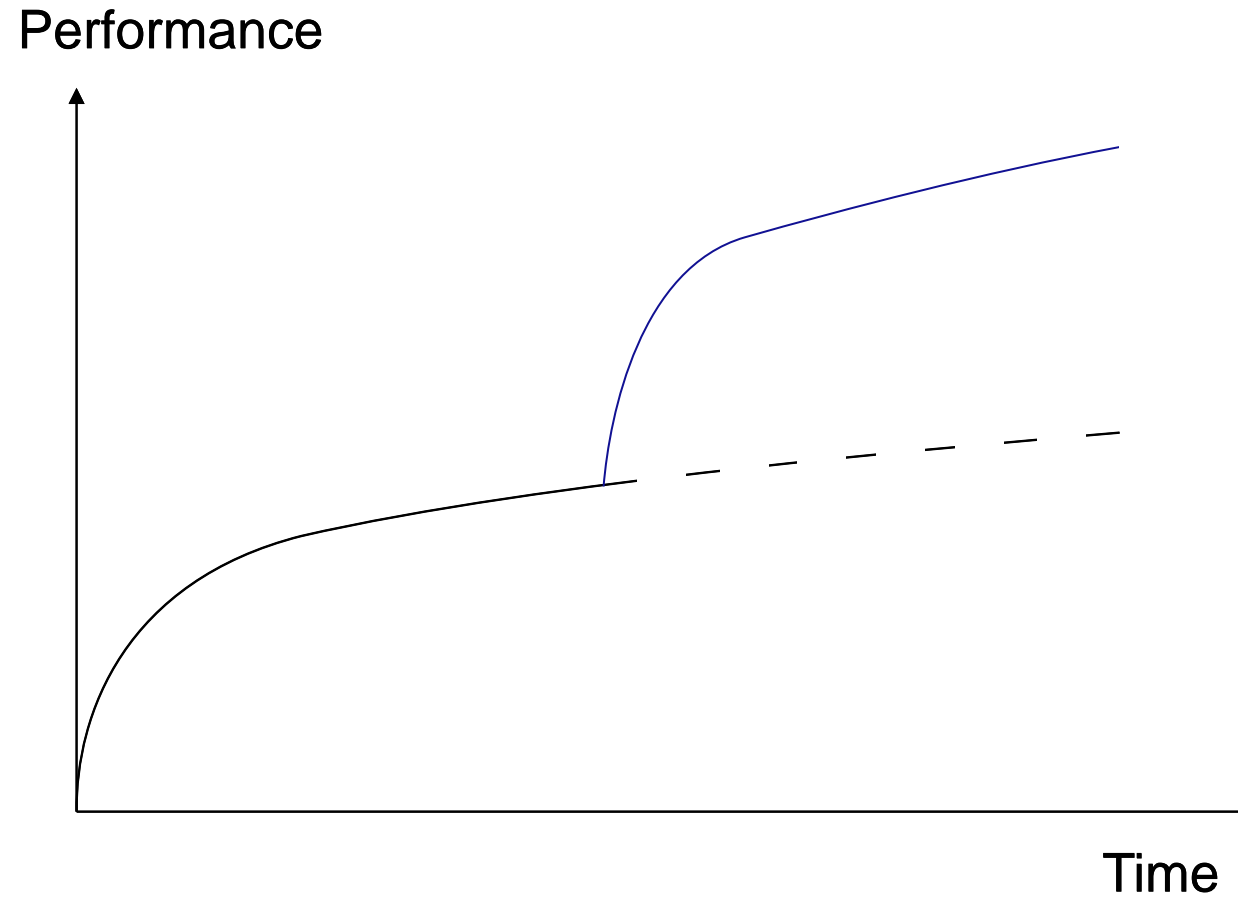
ljardine@mathworks.com



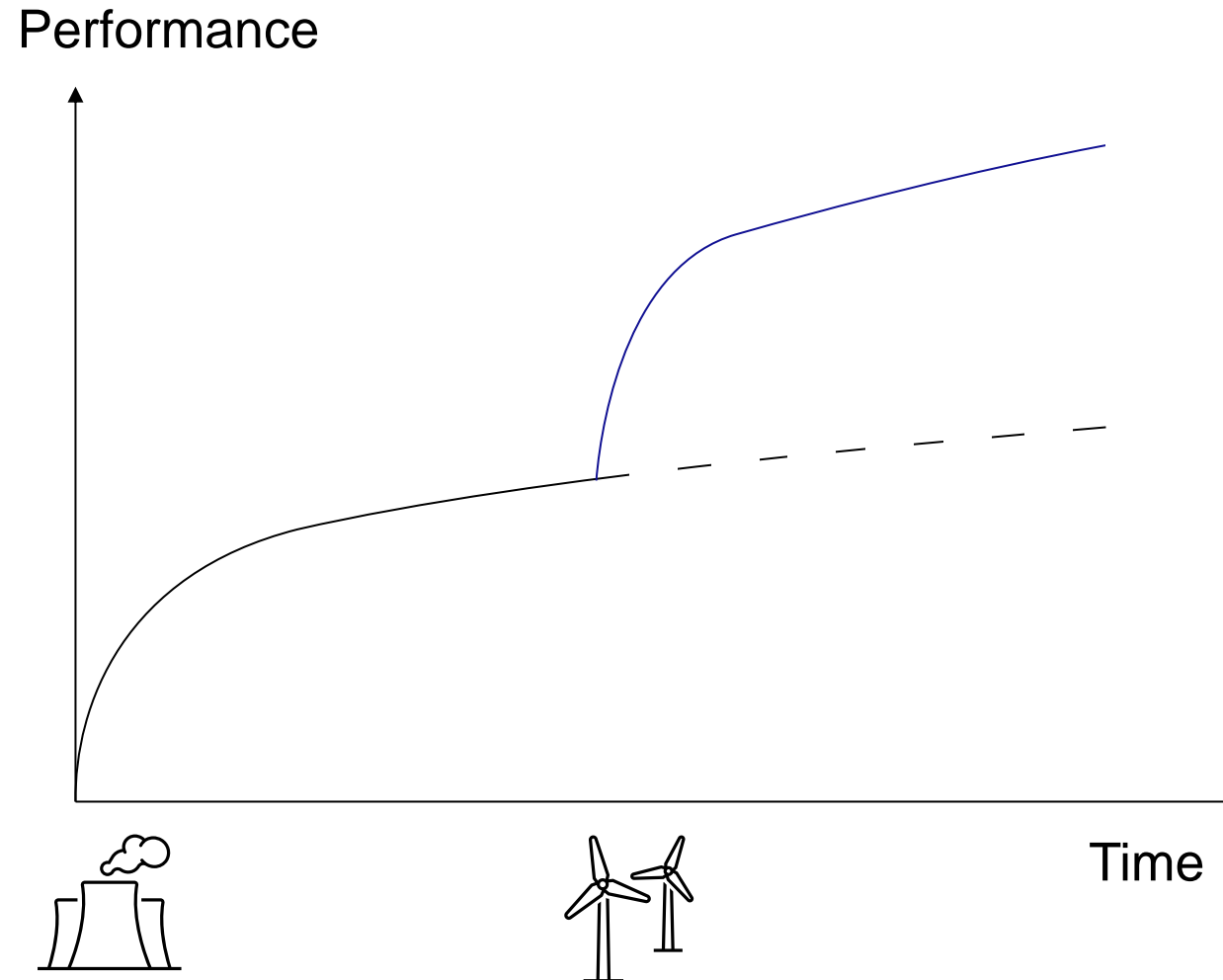
lachlanjardine



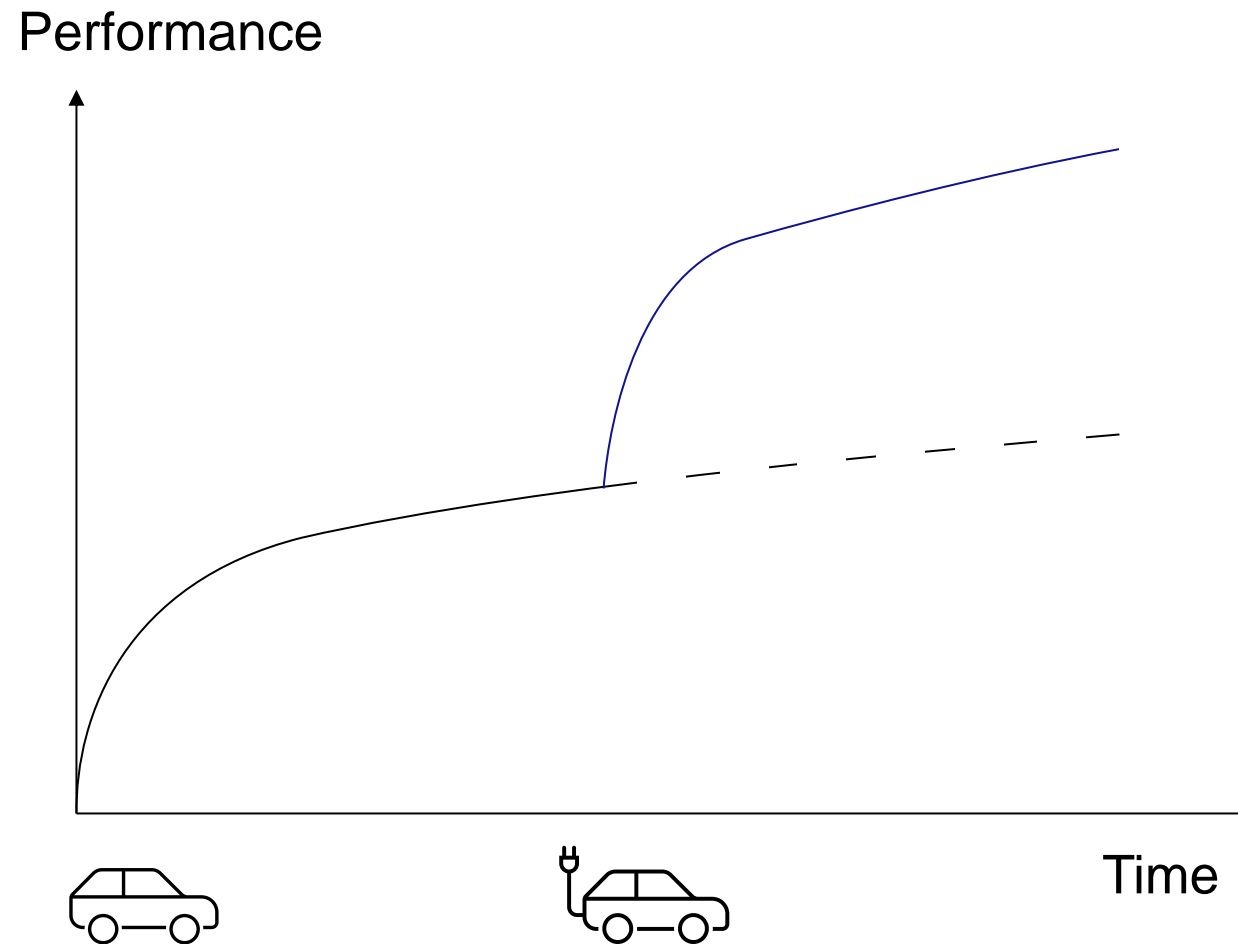
Addressing climate change requires disruptive innovation



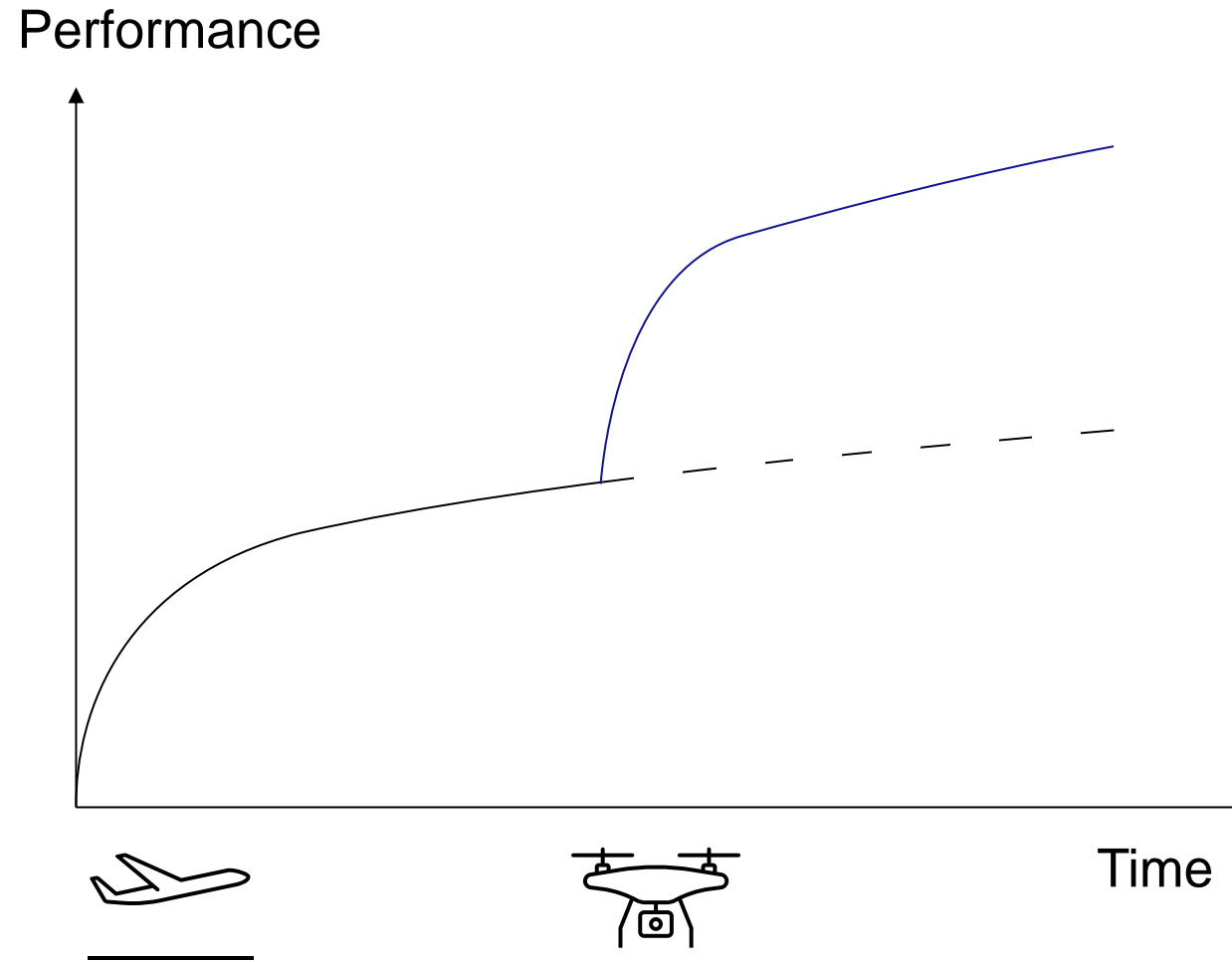
Addressing climate change requires disruptive innovation



Addressing climate change requires disruptive innovation

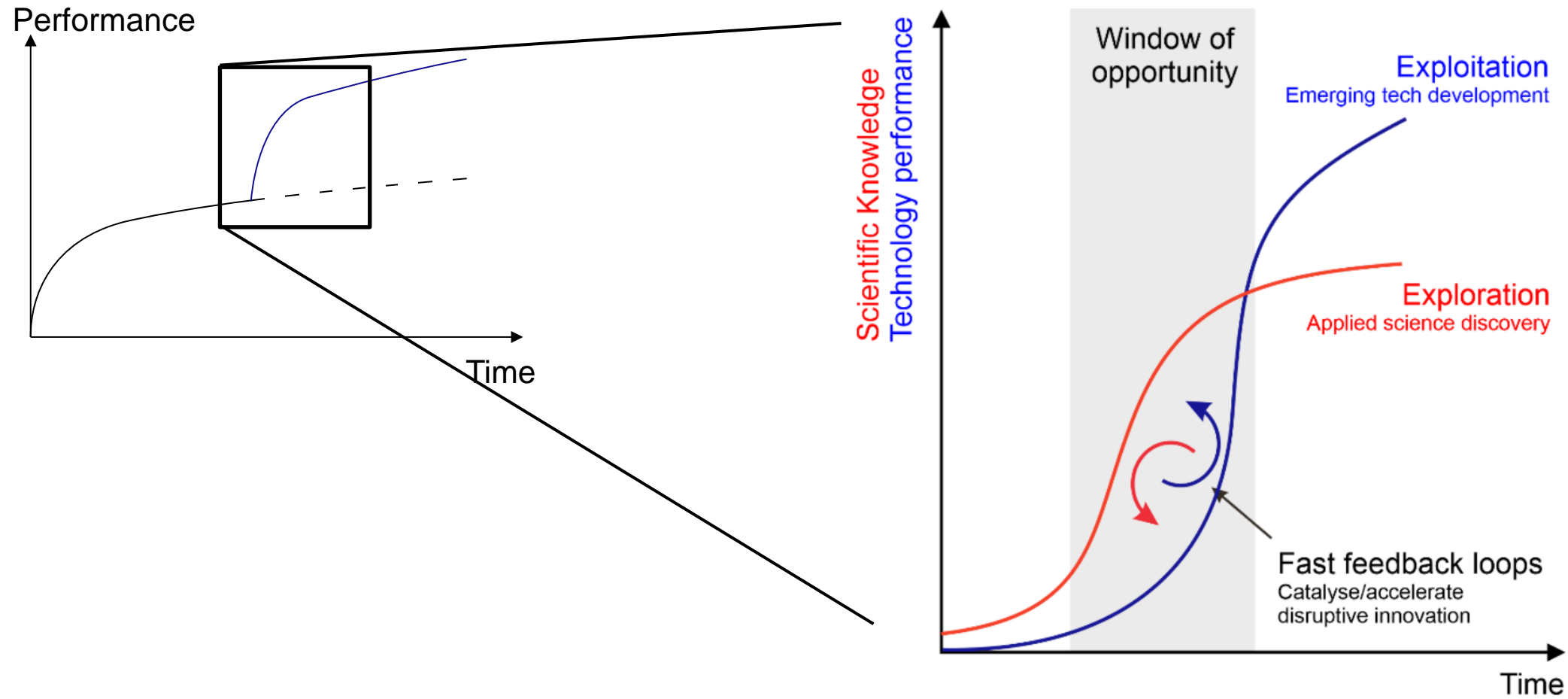


Addressing climate change requires disruptive innovation



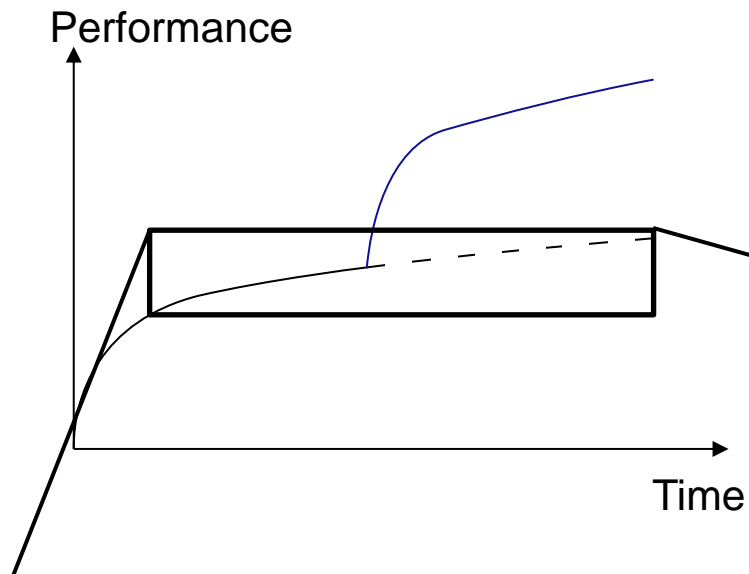
1. Iterating fast

There are **Windows of Opportunity** for disruptive innovation

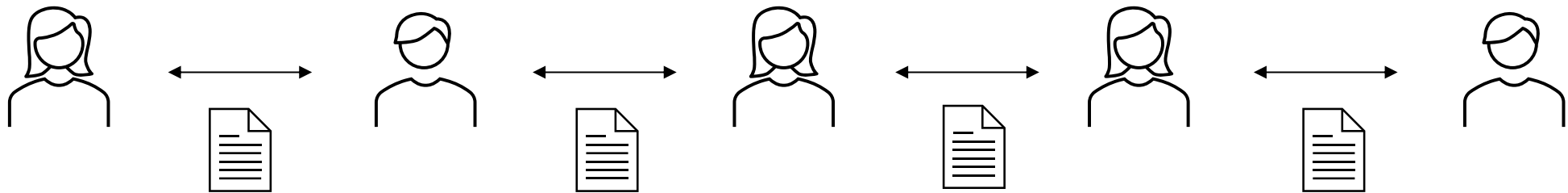


2. Building the plane while flying

The product and ecosystem are being **developed at the same time**

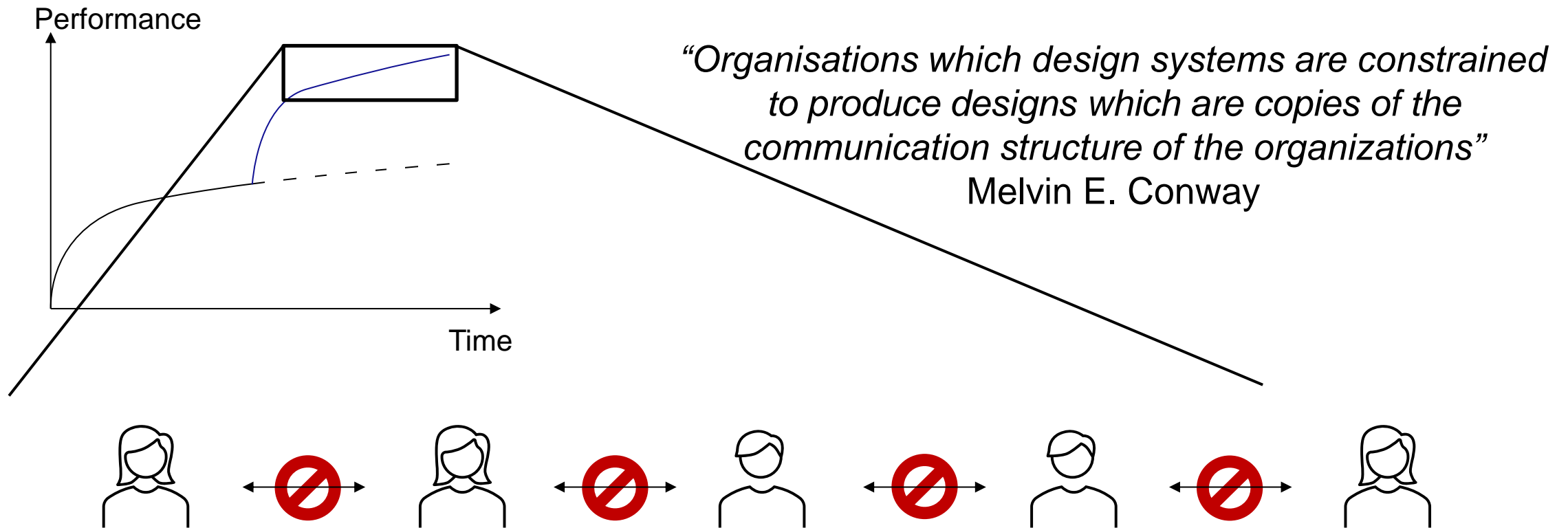


“Organisations which design systems are constrained to produce designs which are copies of the communication structure of the organizations”
Melvin E. Conway



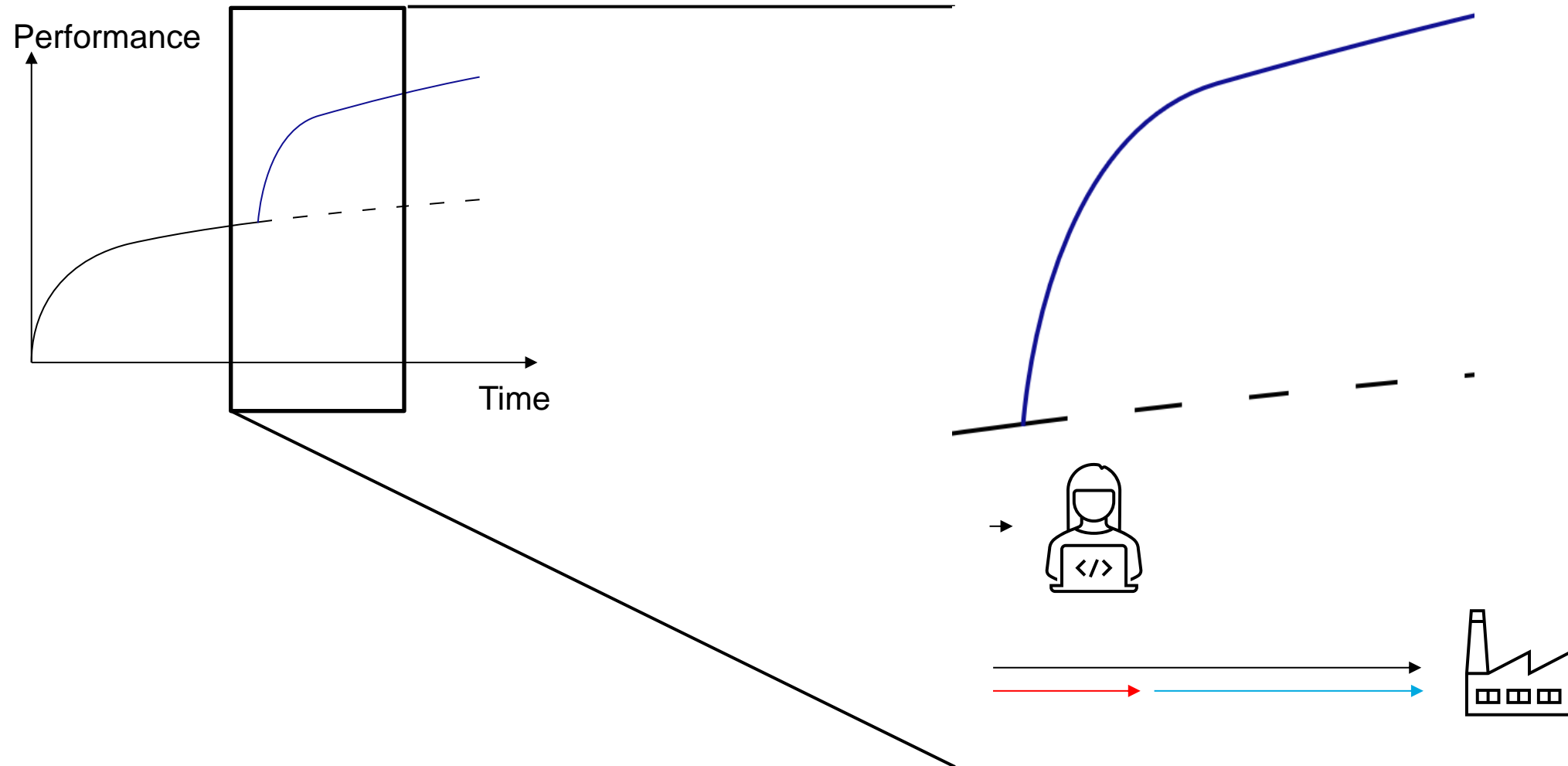
2. Building the plane while flying

The product and ecosystem are being **developed at the same time**



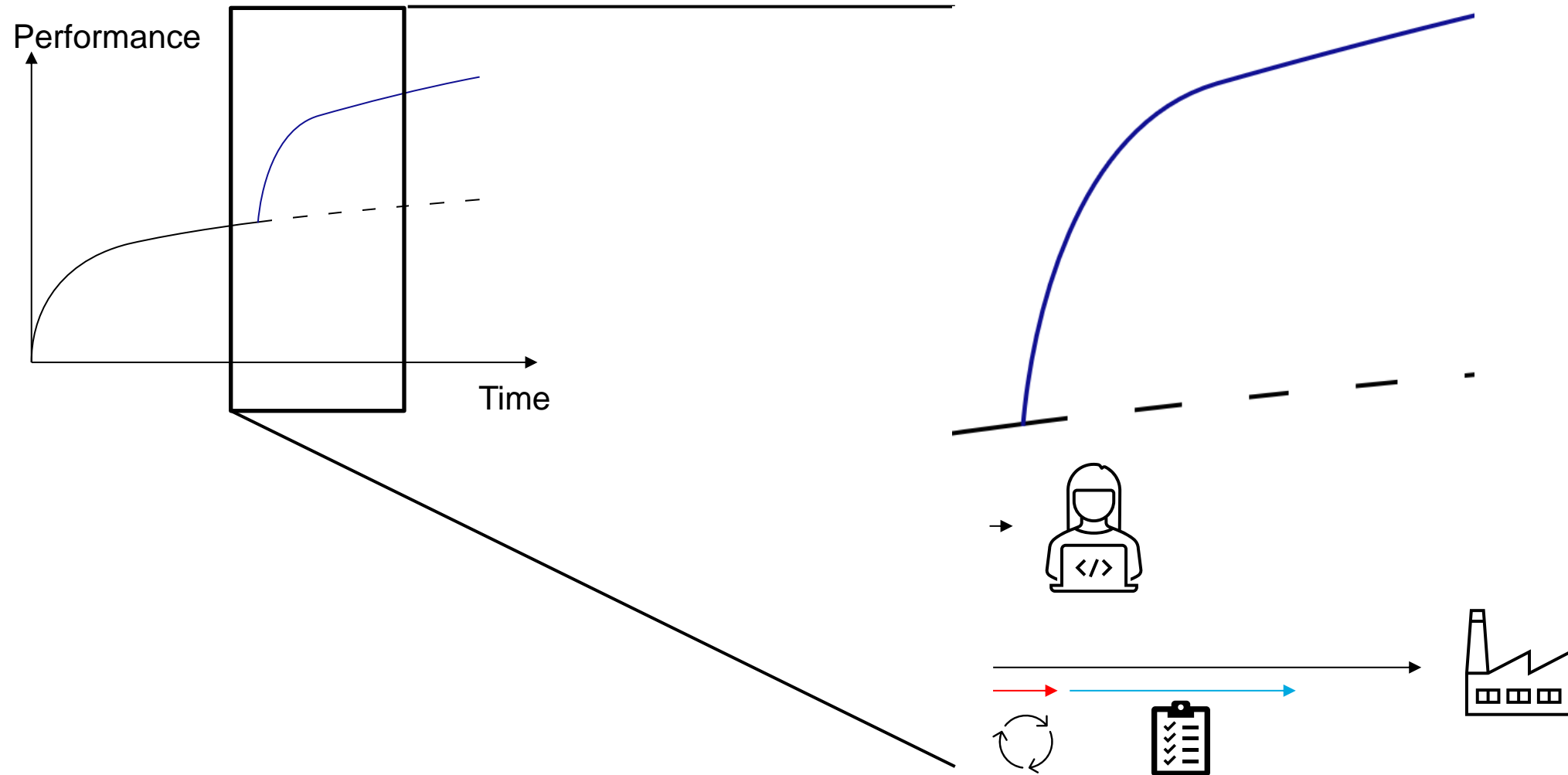
3. Designing for life

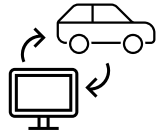
Cyber-physical systems have a **large variation in lifecycle timescales**



3. Designing for life

Cyber-physical systems have a **large variation in lifecycle timescales**





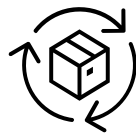
Iterating fast

Simulation of physical systems



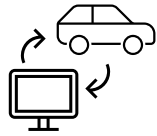
Building the plane while flying

Collaboration and sharing your work



Designing for life

Software practices for cyber-physical systems



Iterating fast

Simulation of physical systems



Building the plane while flying

Collaboration and sharing your work



Designing for life

Software practices for cyber-physical systems

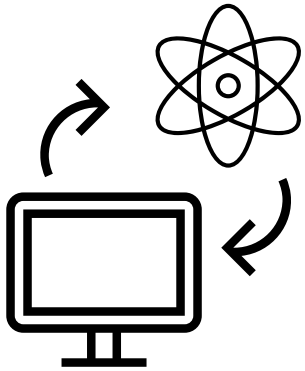


Reduce programme risk by identifying issues as early as possible

Through an integrated, transient, real-time model

Fail fast and learn from it

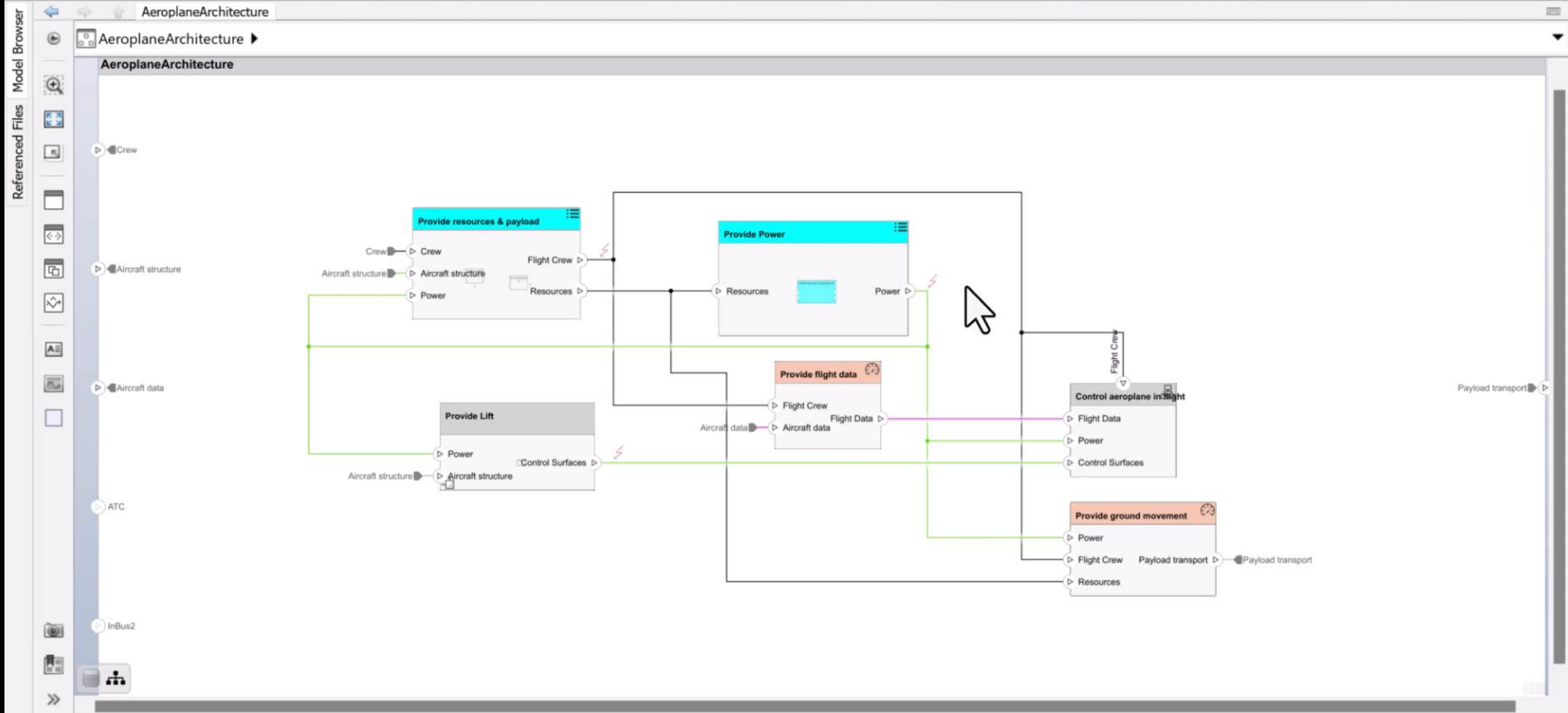
How can we **fail fast and learn?**



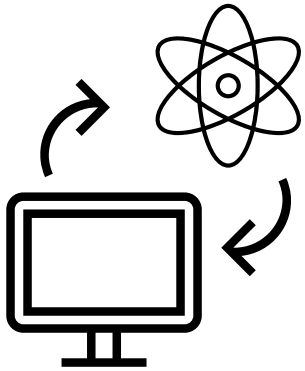
1. **Leverage MBSE** to pull in the same direction
2. **Use real-time models** to derisk physical commissioning
3. **Simplify models** while retaining key behaviour

SIMULATION **DEBUG** **MODELING** **FORMAT** **APPS**

Model Advisor Find Compare To Environment Interface Editor Profile Editor Apply Stereotypes Component Reference Component Variant Component Sequence Diagram Architecture Views Analysis Model Allocation Editor Update Model Stop Time 10.0 Normal Run Stop Fast Restart SIMULATE



How can we **fail fast and learn?**



1. Leverage MBSE to pull in the same direction
2. **Use real-time models** to derisk physical commissioning
3. Simplify models while retaining key behaviour

SIMULATION **DEBUG** **MODELING** **FORMAT** **APPS**

Model Advisor Find Compare To Environment Interface Editor Profile Editor Apply Stereotypes Component Reference Component Variant Component Sequence Diagram Architecture Views Analysis Model Allocation Editor Update Model Stop Time 10.0 Normal Run Stop Fast Restart SIMULATE

PropulsionSystem

PropulsionSystem

PropulsionSystem

ThrustDemand

StructuralMount

Performance

Aerothermal

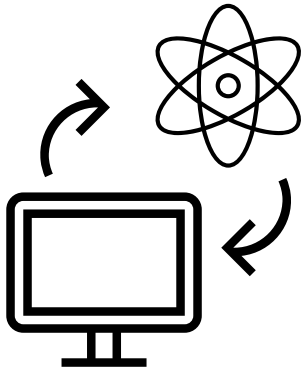
Startup

Life


FuelConsumption


TimeOnWing


How can we **fail fast and learn?**







1. Leverage MBSE to pull in the same direction
2. Use real-time models to derisk physical commissioning
3. **Simplify models** while retaining key behaviour


 Save All
FILE


 Requirements Editor
EDIT


 New Requirement Set
REQUIREMENT SET


 Import
 Open

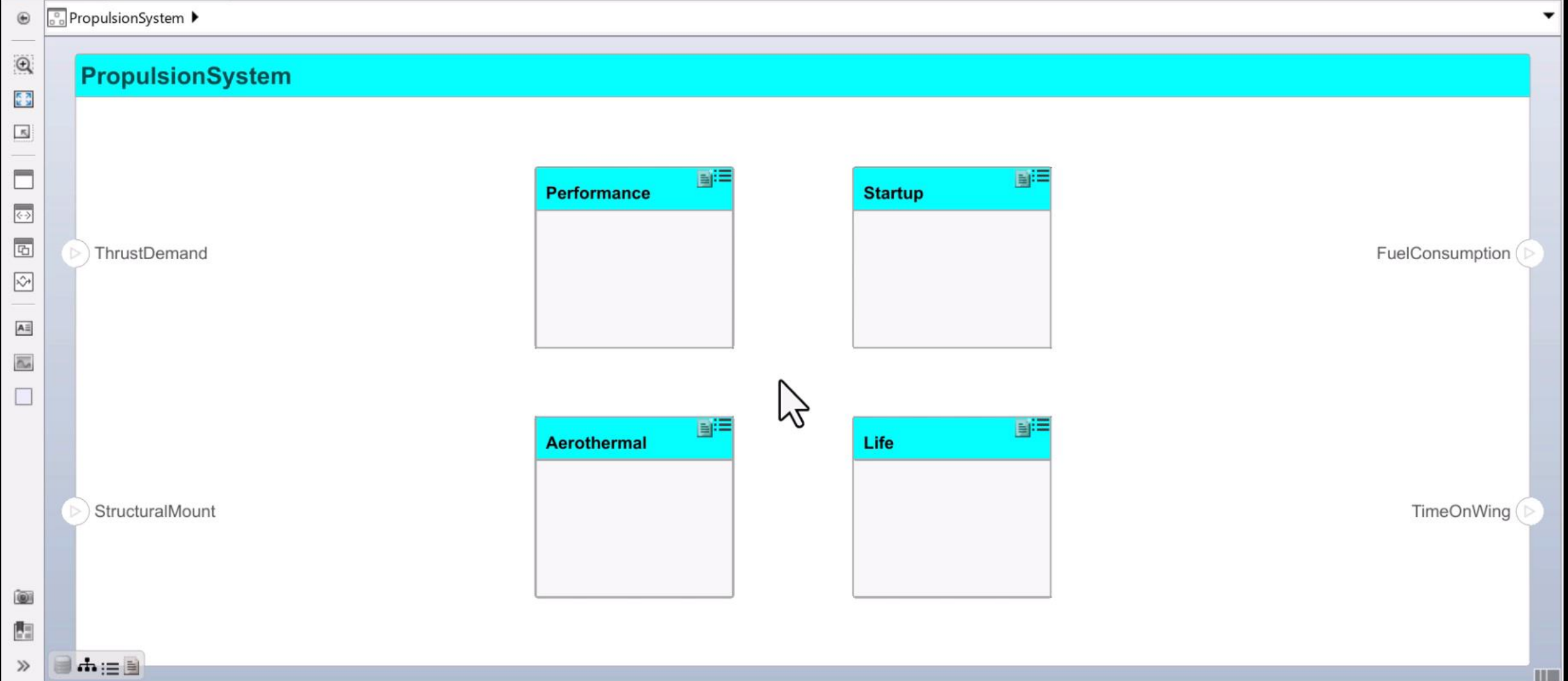
 Highlight Links
 Layout
VISUALIZE

 Check Consistency

 Traceability Matrix
ANALYZE

 Traceability Diagram

 Share
SHARE



Save All
FILE

Requirements Editor
EDIT

New Requirement Set
REQUIREMENT SET

Import
Open

Highlight Links
Layout

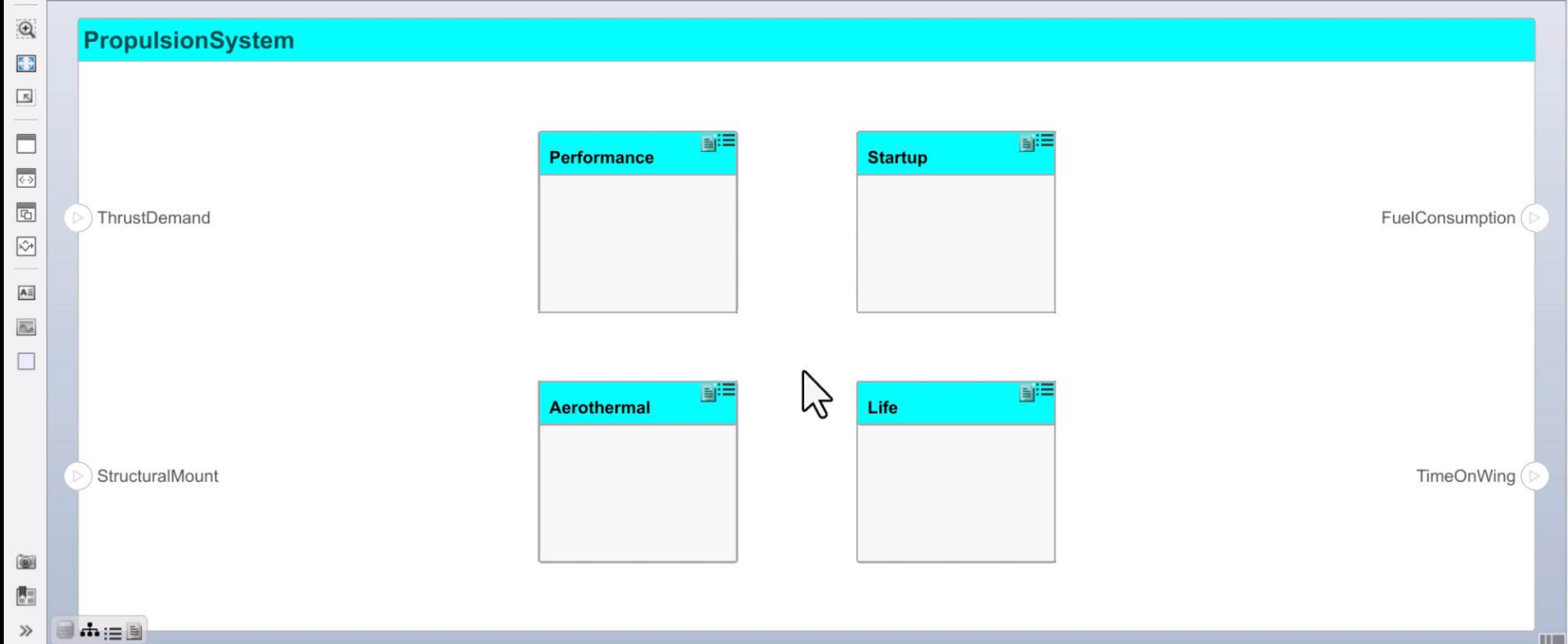
Visualize

Check Consistency

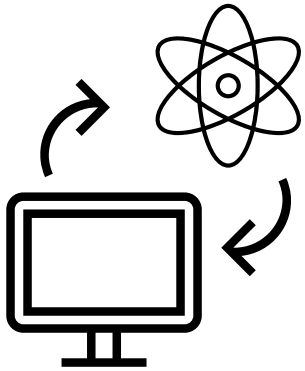
Traceability Matrix
ANALYZE

Traceability Diagram

Share
SHARE



How can we **fail fast and learn?**



1. **Leverage MBSE** to pull in the same direction
2. **Use real-time models** to derisk physical commissioning
3. **Simplify models** while retaining key behaviour

**Designing a Battery System for the VX4
eVTOL Aircraft**

Emma Palin, Felix Diaz-Maroto Rivas,
Vertical Aerospace

**Simulation-Driven Safety Analysis
Through Fault Injection Testing**

Fredrik Håbring

1.

Hardware-in-the-Loop Testing for Electric
Powertrains

6.

Modelling Thermal Management Systems
for an Electric Powertrain using Simscape.



Iterating fast

Simulation of physical systems



Building the plane while flying

Collaboration and sharing your work



Designing for life

Software practices for cyber-physical systems



How can we develop tools a wider group of users can usefully use?

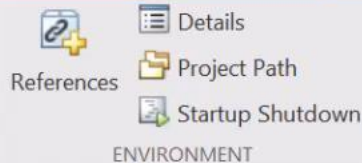
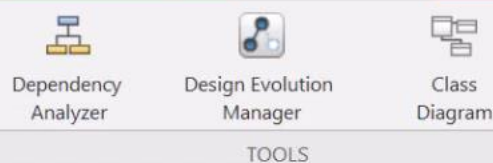
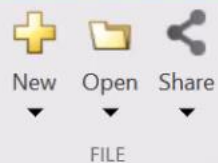
Prototyping models – put together an idea of what you want and iterate

Bridge between science, engineering, technology, to decision makers

How can we bridge between science, engineering, technology, to decision makers



1. **Collaboration** through common ways of working
2. **Prototype and share your work** widely
3. **Continue learning**



C:\Users\ljardine\OneDrive - MathWorks\Presentations\2024 - MATLAB EXPO\Content\Content\expo-simulationarchitecture

Current Folder

Name	Git
models	
pipelines	.
resources	.
.gitattributes	●
.gitignore	●
Digital_default.png	○
Filtered View_default.png	○
tView 1_default.png	○
Undelivered_default.png	○

Details

Workspace

Name ^	Value
--------	-------

Project - expo-simulationarchitecture

Views

Files

Labels

Classification

Git

branch status:
Normal
No remote tracking
branch

All Project (40) Modified (1)

Name ^	Status	Classification	Git
data	✓		.
faults	✓		.
tests	✓		.
AeroplaneArchit...	✓		●
AeroplaneArchit...	✓	Design	□
PropulsionSyste...	✓	Design	●
PropulsionSyste...	✓		●
TestRequiremen...	✓		●

AeroplaneArchitecture.slx (Architecture Model)

1 labels ^

Command Window

fx >>

How can we develop tools a wider group of users can usefully use?



1. Articulate change clearly
2. **Share your work** widely
3. Continue learning new perspectives

SIMULATION **DEBUG** **MODELING** **FORMAT** **APPS**

Project: New, Open, Save, Print

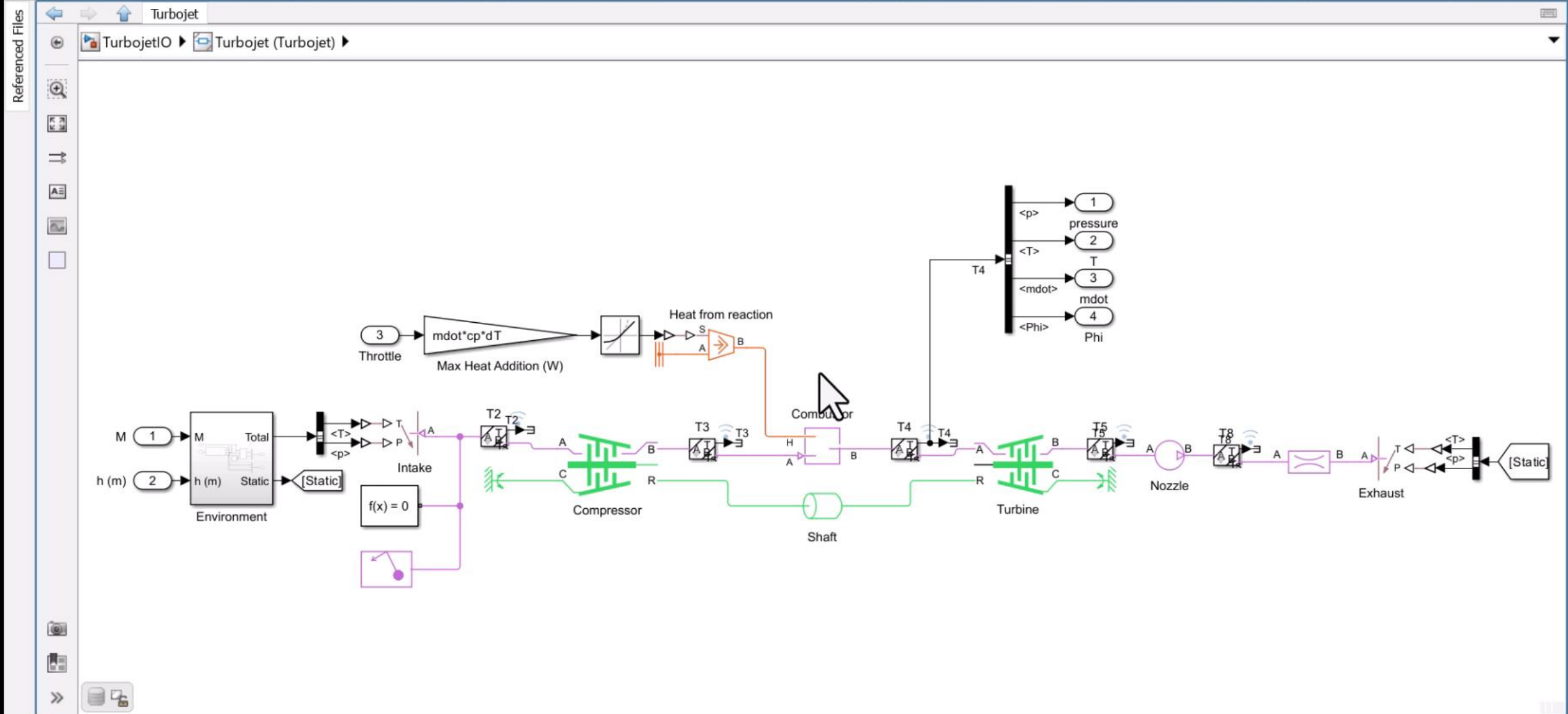
Library: Library Browser

PREPARE: Log Signals, Add Viewer, Signal Table

Stop Time: 10.0
Normal
Fast Restart

SIMULATE: Step Back, Run, Step Forward, Stop

REVIEW RESULTS: Data Inspector, Logic Analyzer, Simulation Manager

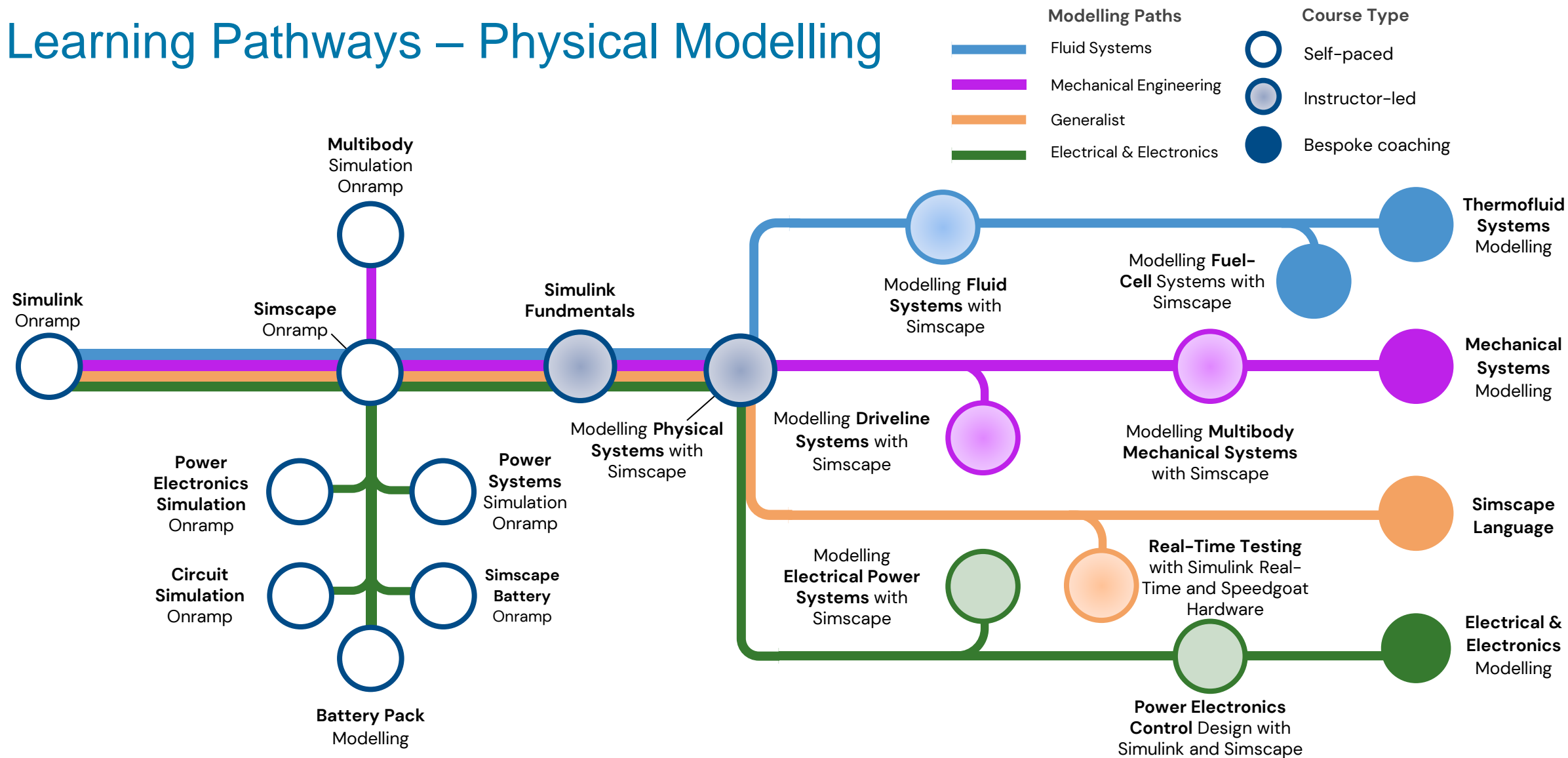


Change is guaranteed – **how can we communicate effectively?**



1. Articulate change clearly
2. Share your work widely
3. **Continue learning** new perspectives

Learning Pathways – Physical Modelling



Fundamental

Technical

Strategic

Change is guaranteed – **how can we communicate effectively?**



1. **Articulate change** clearly
2. **Share your work** widely
3. **Continue learning** new perspectives

2.
Sharing Your Engineering Models and Tools
Using App Development.

7.
Building Architecture Diagrams using
System Composer

14.
Building and Deploying Robust AI Models

15.
Upskilling Zone



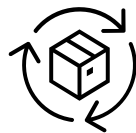
Iterating fast

Simulation of physical systems



Building the plane while flying

Collaboration and sharing your work



Designing for life

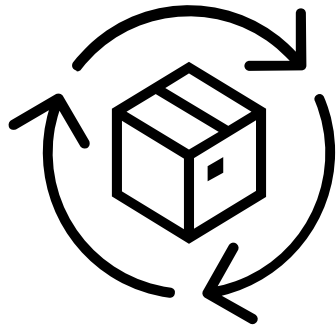
Software practices for cyber-physical systems

How can we handle changesets as the team scales?

*Through Agile, a custom training course, and continuous integration /
continuous deployment pipelines.*

Shortened delivery timescales and increase quality.

How can we **shorten delivery timescales and increase quality**?

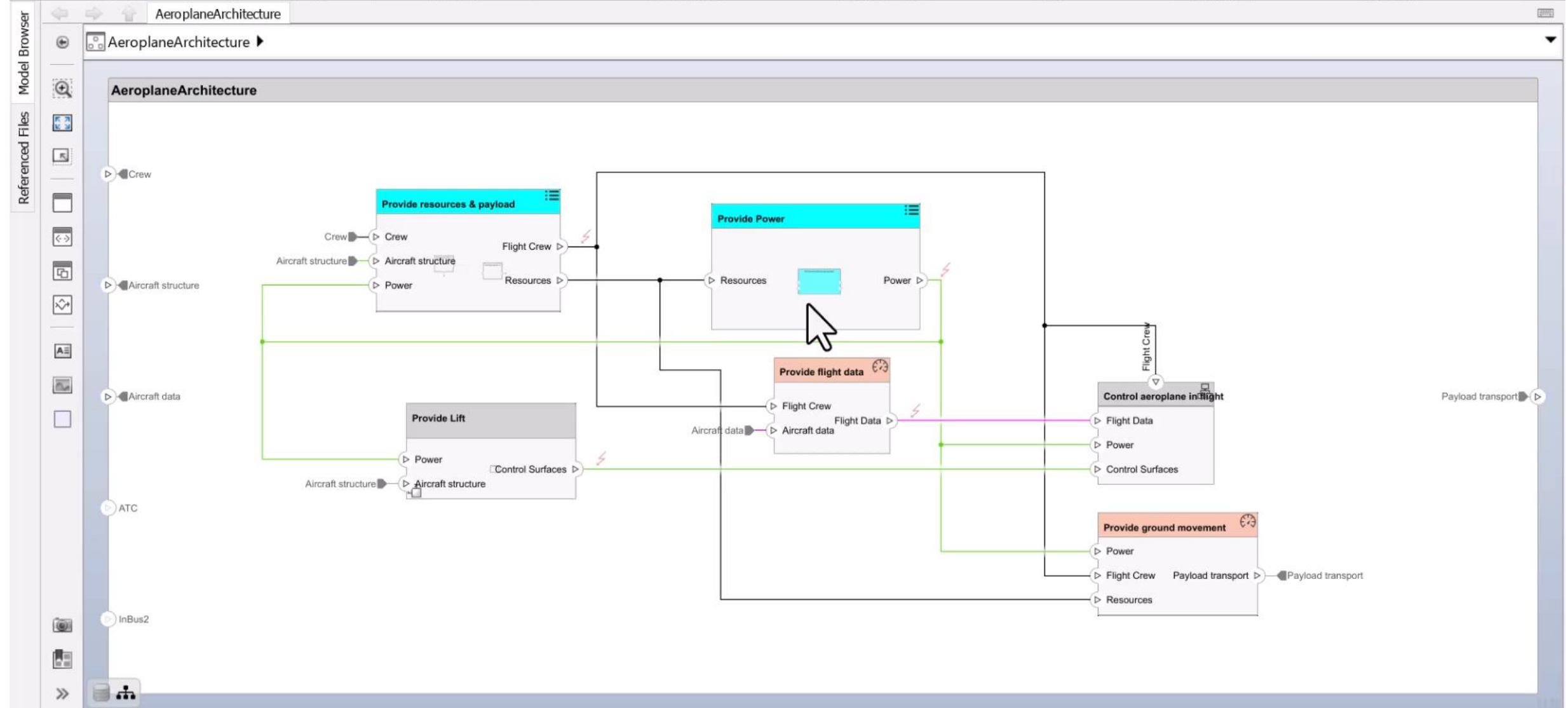


1. **Build in quality** by testing
2. **Automate** through Continuous integration
3. **Discover** with Model Management

AeroplaneArchitecture - Simulink

SIMULATION DEBUG **MODELING** FORMAT APPS

Model Advisor Find Compare To Environment Interface Editor Profile Editor Apply Stereotypes Component Reference Component Variant Component Sequence Diagram Architecture Views Analysis Model Allocation Editor Update Model Stop Time 10.0 Normal Run Stop Fast Restart SIMULATE



New
ScriptNew
Live Script

New



Open



Find Files

Compare

Import
DataClean
Data

Variable

Save Workspace

Clear Workspace



Favorites



Analyze Code

Run and Time

Clear Commands



Simulink



Layout



Preferences

Set Path

Parallel



Add-Ons



Help



Community

Request Support

Learn MATLAB

C:\Users\ljardine\OneDrive - MathWorks\Presentations\2024 - MATLAB EXPO\Content\mass-spring-damper-app-testing

Current Folder

Name	Git
.git	.
Demos	.
derived	.
resources	.
Shortcuts	.
slprj	.
Tests	.
ReadMe.md	.
MassSpringDamper.prj	.

Details

Workspace

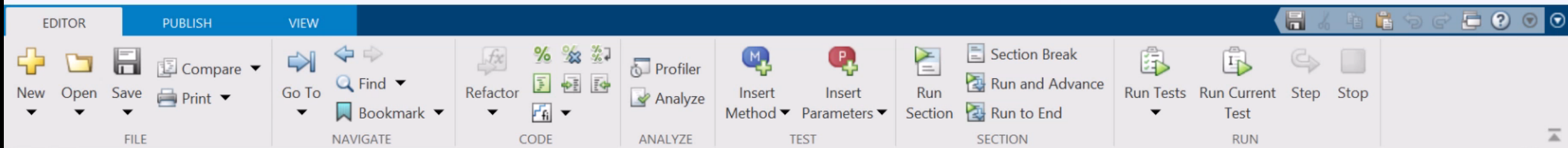
Name ^	Value
ans	1x1 ssc_msd_DoE_
out	1x1 SimulationOu

Command Window

fx >>

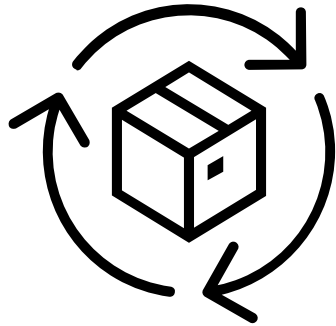


Project - Mass-Spring-Damper



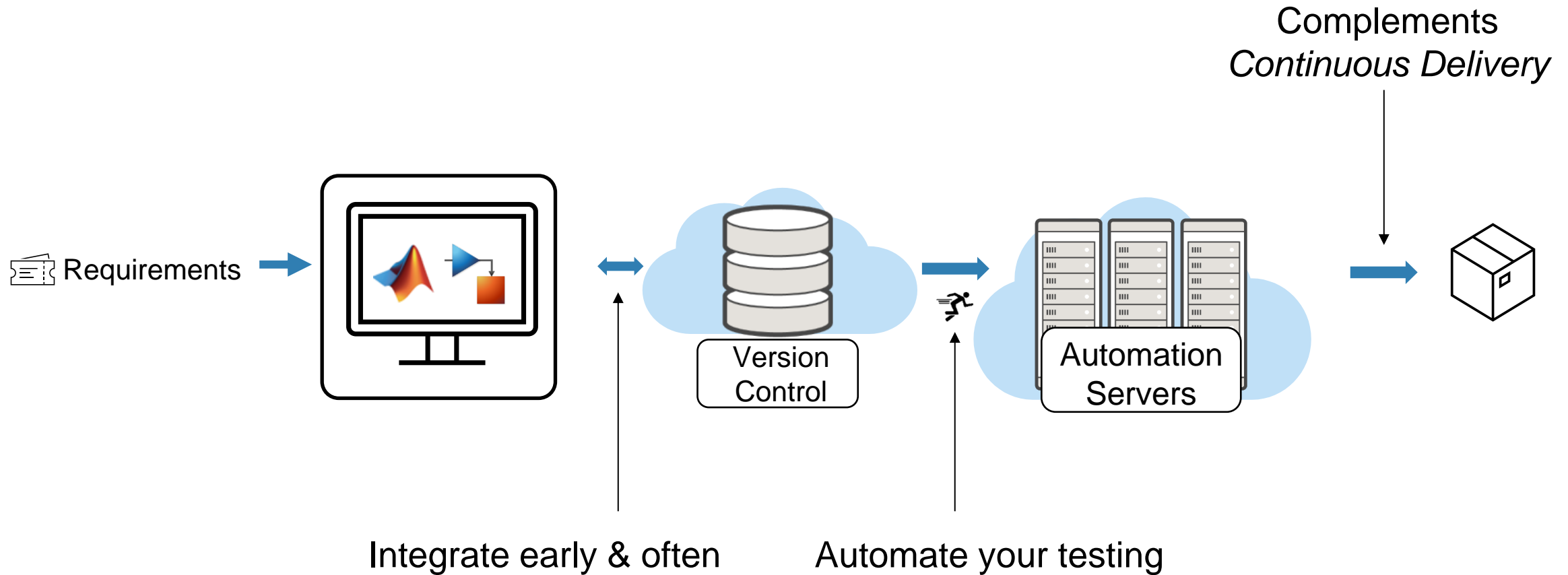
```
1 classdef Test_ROM_Credibility < sltest.TestCase
2
3     properties( TestParameter )
4         ambientNom = { 0.9, 1.1 }
5         coolingNom = { 0.9, 1.1 }
6         pressureNom = { 0.9, 1.1 }
7     end
8
9     properties
10         artifactDir string
11         ambient_val double
12         cooling_val double
13         pressure_val double
14     end
15
16     methods (TestClassSetup)
17         function setupState(testCase)
18             testCase.artifactDir = currentProject().RootFolder + filesep + "artifacts" + filesep;
19             load( "turbineblade_validationData.mat", ...
20                 "ambient_val", "cooling_val", "pressure_val" );
21             testCase.ambient_val = ambient_val;
```


Change is guaranteed – **how can we deliver updates faster?**



1. Build in quality through testing
2. **Automate** through Continuous integration
3. Discover with Model Management

Continuous Integration – what is it?



HOME PLOTS APPS PROJECT PROJECT SHORTCUTS

New Open Share

Dependency Analyzer Design Evolution Manager Class Diagram

References Project Path Startup Shutdown

FILE TOOLS ENVIRONMENT

C: > Users > ljardine > OneDrive - MathWorks > Presentations > 2024 - MATLAB EXPO >

Current Folder

Name	Git
derived	.
models	.
pipelines	.
resources	.
.gitattributes	.
.gitignore	.
simpleRunner.m	.
ExpohighfidelityROM.prj	.

ExpohighfidelityROM.prj (Project)

Workspace

Name	Value
------	-------

Project - expo-highfidelityROM

Views

Files

Dependency Analyzer

Labels

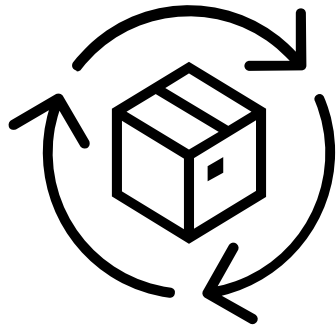
Git

Coincident with /origin/master

fx >>



Change is guaranteed – **how can we deliver updates faster?**



1. Build in quality through testing
2. Automate through Continuous integration
3. **Discover** with Model Management

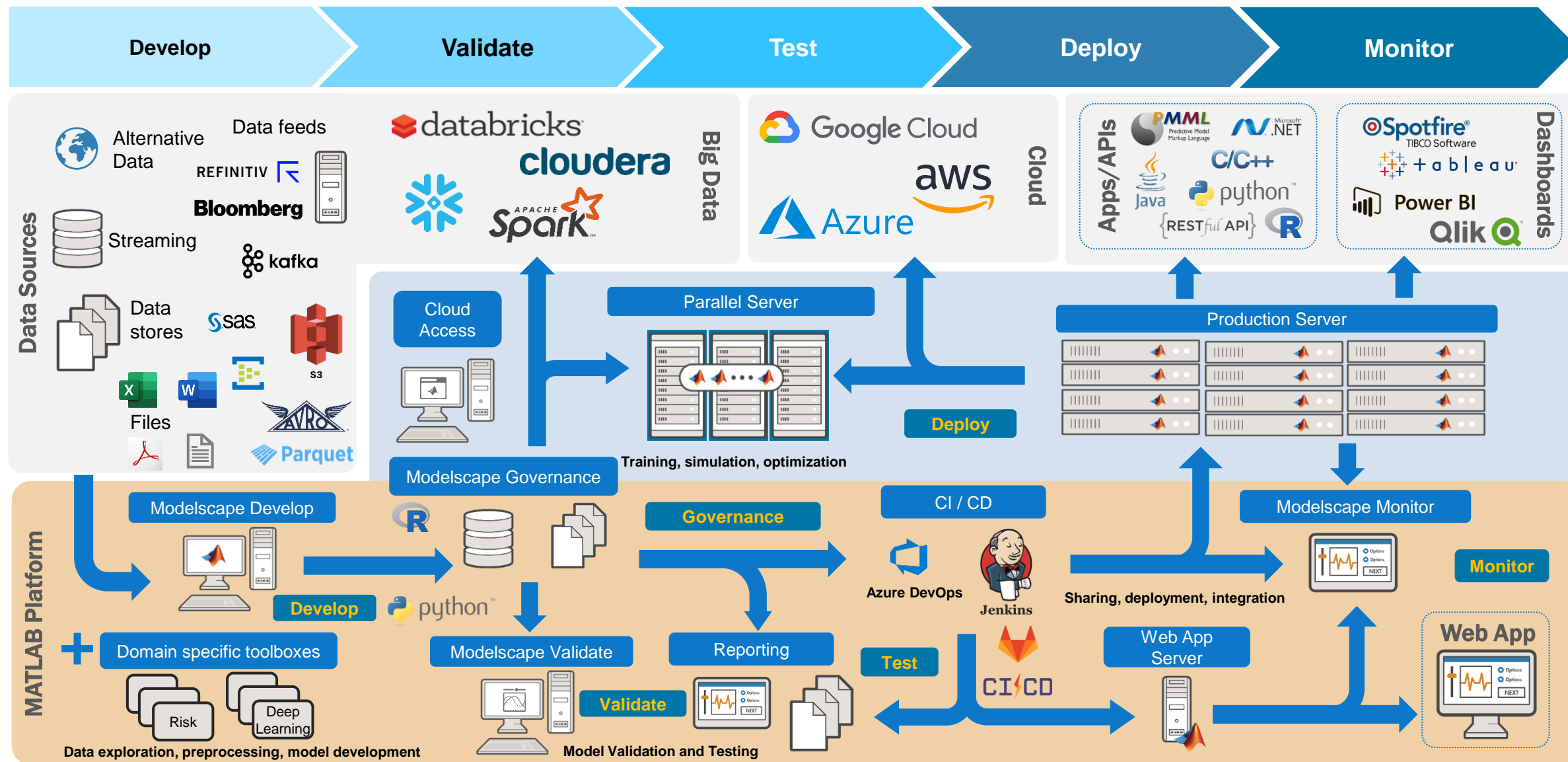
Keep it simple

keep it very simple -- the more models you have, you need inventory, and maturity metadata

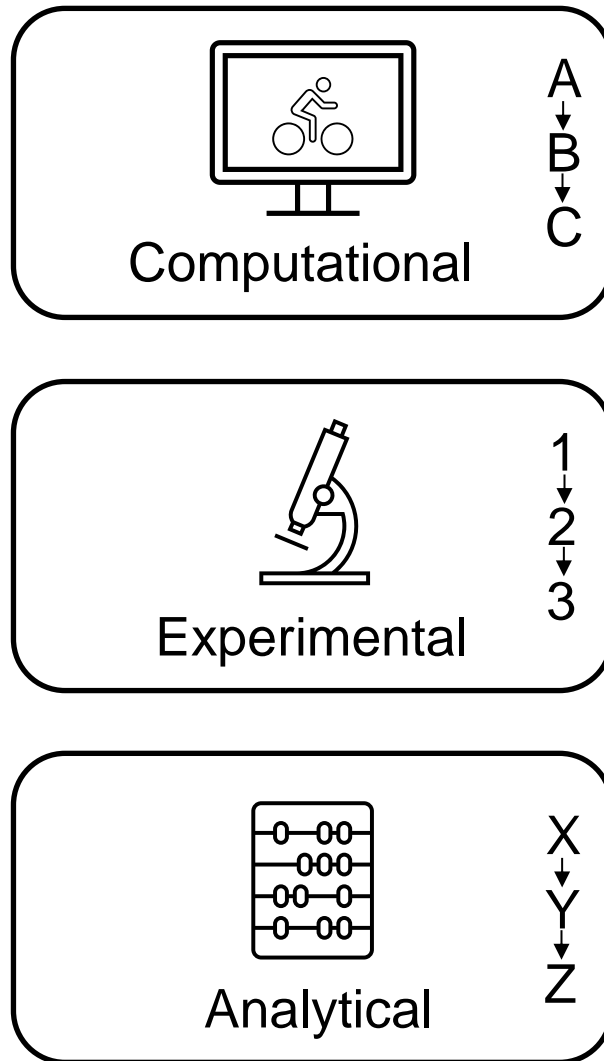
1. Keeping track of models and their artifacts is complex



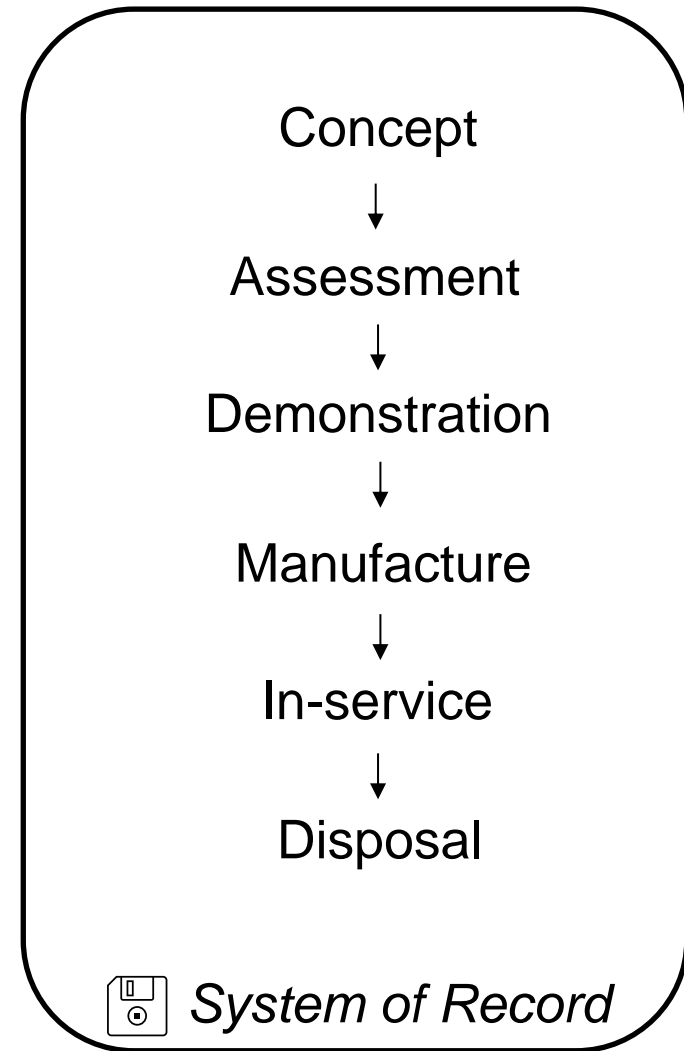
1. Keeping track of models and their artifacts is complex



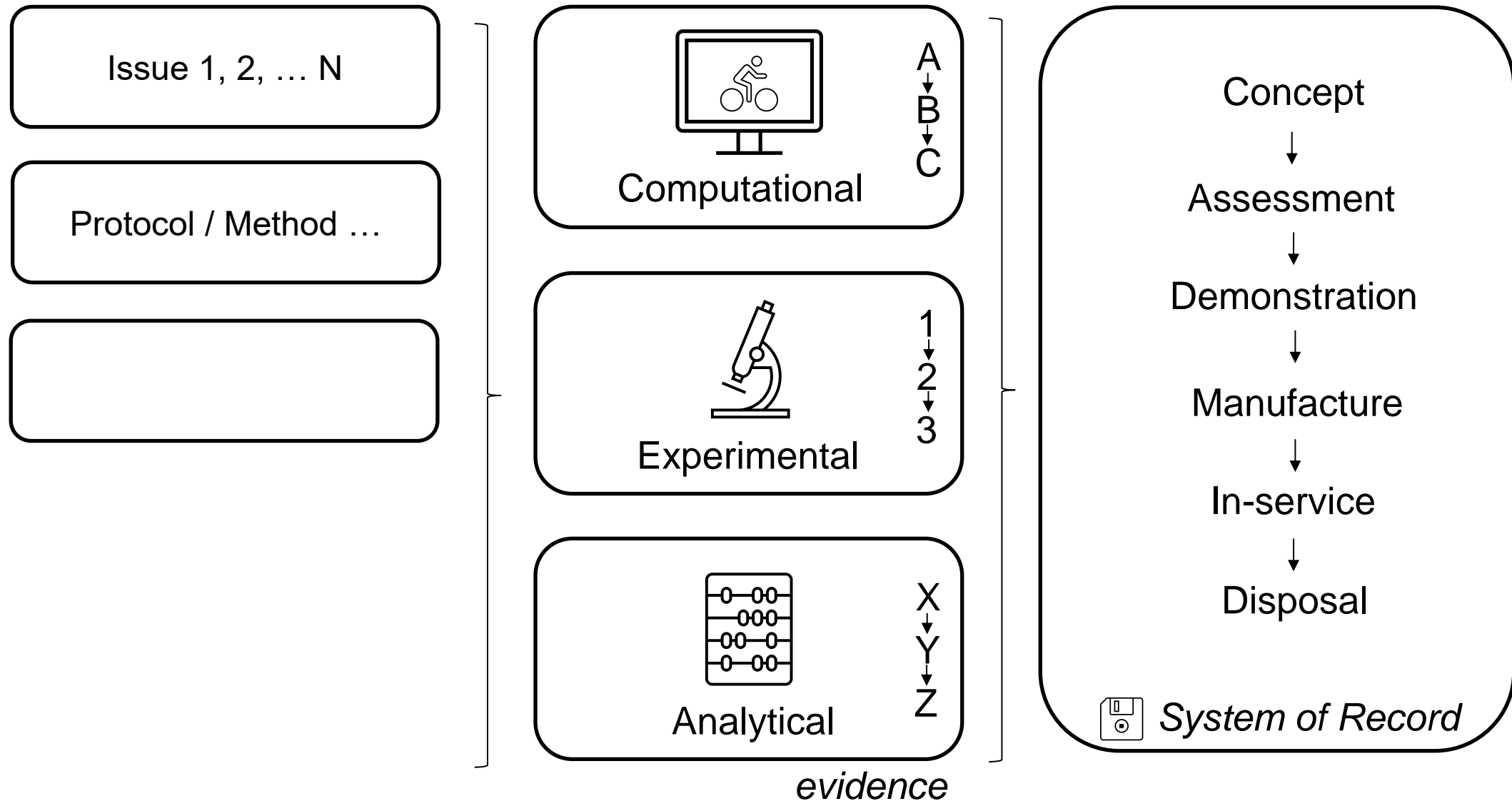
2. Everything has a lifecycle



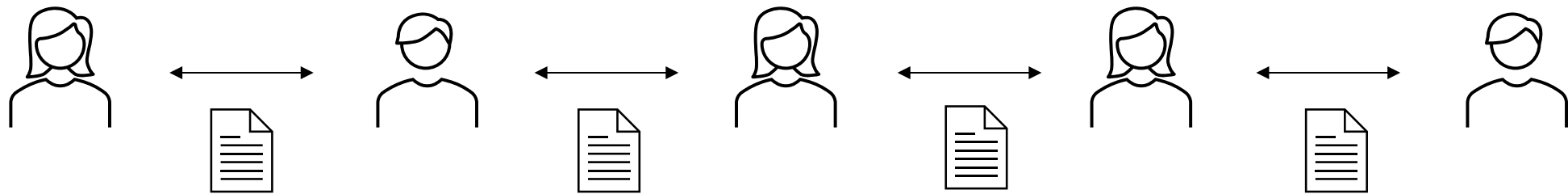
evidence



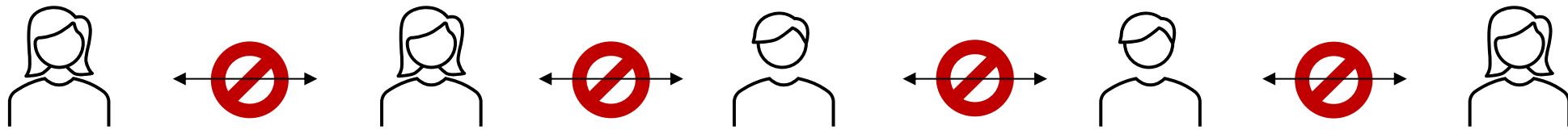
2. Everything has a lifecycle



3. Your framework influences **how** and **what** you communicate



3. Your framework influences **how** and **what** you design











Even for the same standard, the implementation remains company specific



Modelscape Model Management

Lachlan ▾

-  Lifecycles
-  Models
-  Model Versions
-  Reviews
-  Builds
-  Deployment Environm...
-  Deployments
-  Dependencies

Apps

-  Lifecycle Designer
-  Dependency Editor

 Collapse

Quick Access

Getting Started

Get started with Modelscape using the Online Documentation.

Create a New Model

Create a new Modelscape model.

Lifecycle Designer

Use the Lifecycle Designer app to create new lifecycles and edit existing lifecycles in Modelscape.

Quick Search

 Search ...

Filter:

☒ All 30

☐ Models 16

☐ Versions 11

☐ Lifecycles 3

☐ Show my items only

Results per page: 10 ▾

< Results 1 - 10 of 30 >

ACTIONS

NAME

TYPE

CREATED BY

MODIFIED ON

[View Details](#)

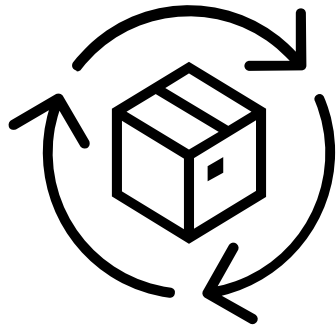
Supplied Component A - v0.1

Model Version

@lachlan

6 Oct 2024

Change is guaranteed – **how can we deliver updates faster?**



1. **Build in quality** through testing
2. **Automate** through Continuous integration
3. **Discover** with Model Management

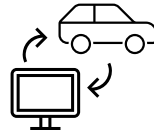
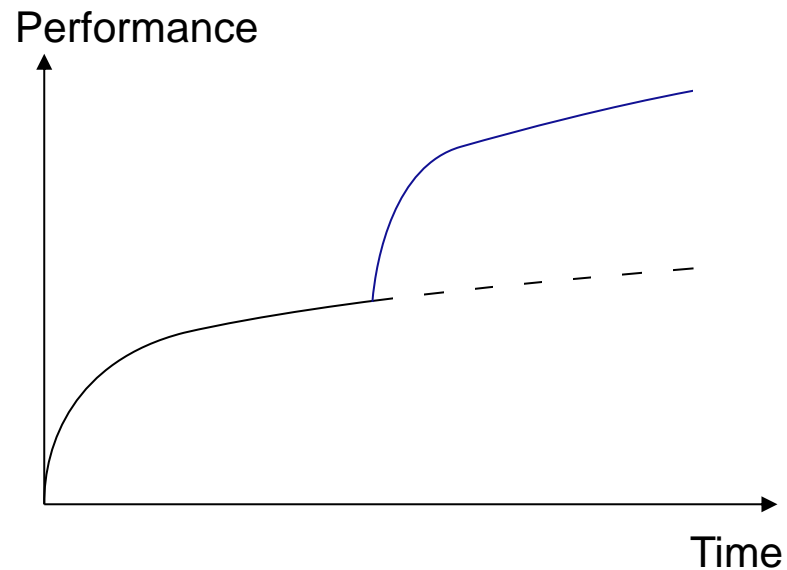
**Engineering in the Cloud: Unlocking
New Possibilities**
Amélie Lamarquette

5.
Testing Models and Code using MATLAB
Test and Simulink Test

3.
Deploying MATLAB and Simulink in the
Cloud.

4.
Using MathWorks Tools with Continuous
Integration Platforms

Disruptive innovation can be seen across every industry



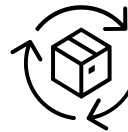
Iterating faster

Simulation of physical systems



Building the plane while flying

Collaboration and sharing your work



Designing for life

Software practices for cyber-physical systems

What advice would you give engineers?

“Your career is a marathon not a sprint”

“Develop skills you are interested in”

“Just talk to people”


MATLAB EXPO

 UNITED KINGDOM

Dr Lachlan Jardine, MathWorks



 ljardine@mathworks.com

 [lachlanjardine](https://www.linkedin.com/in/lachlanjardine)



© 2024 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [mathworks.com/trademarks](https://www.mathworks.com/trademarks) for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

Special Thank You to:

**Martin Bailey,
Rayner Sagers,
Prof. Andrew Wheeler**

In addition:

Bayard Morales