MATLAB EXPO 2019

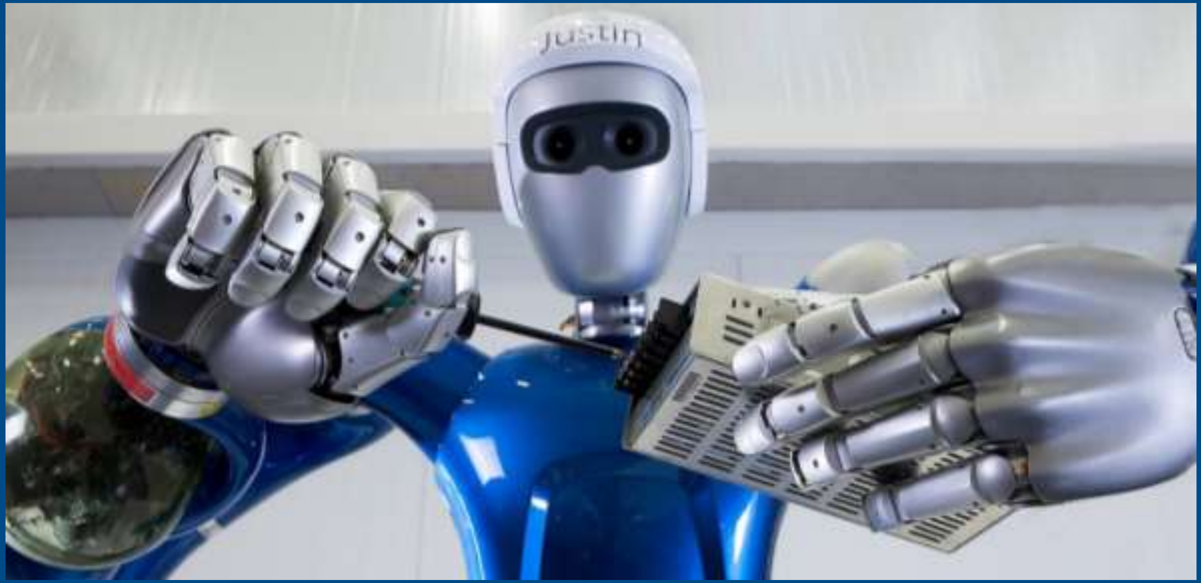
# What’s New in MATLAB and Simulink

**山本 順久**

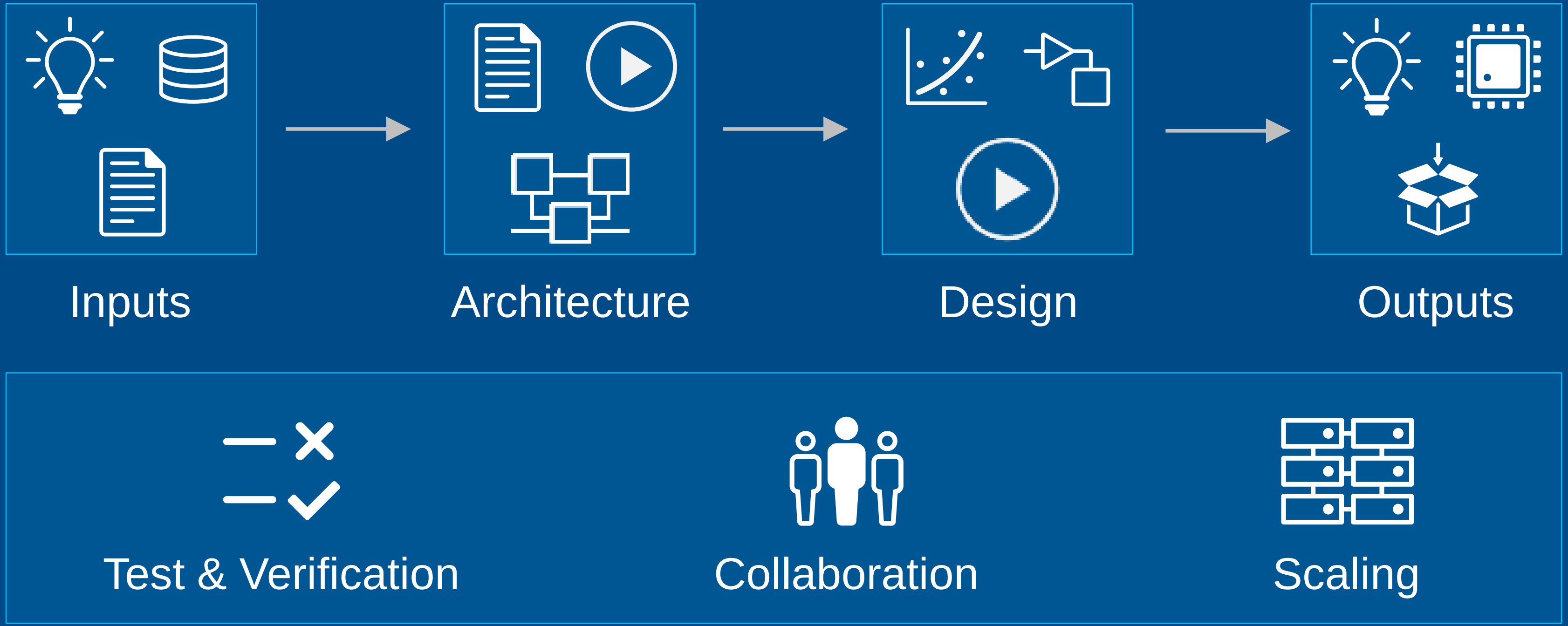
VOYAGE

EarlySense
Proactive Patient Care

# Algorithms in Everything



# Using MATLAB & Simulink to Build Algorithms in Everything



Inputs

Architecture

Design

Outputs

Test & Verification

Collaboration

Scaling

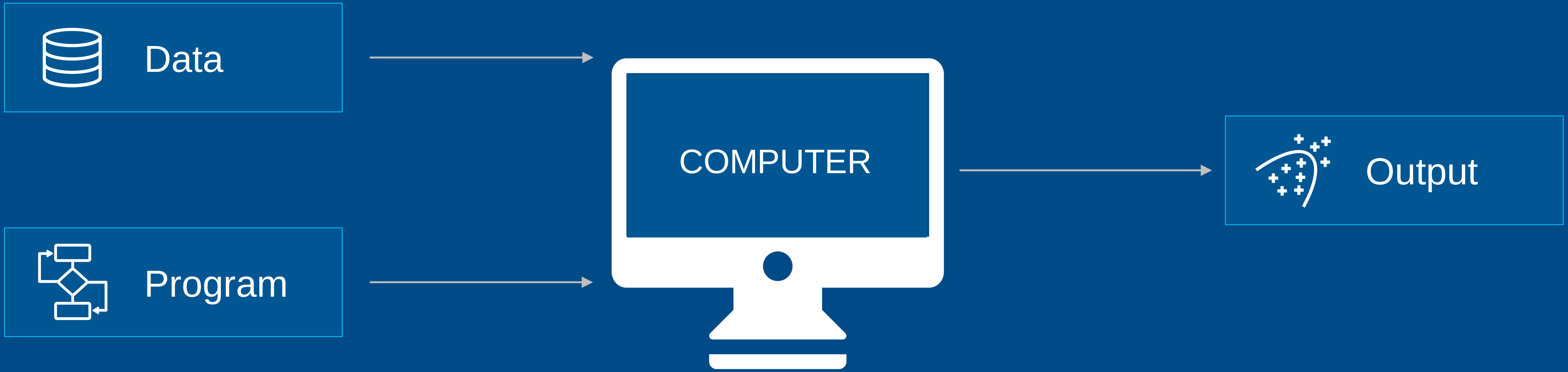
MATLAB®&SIMULINK®

# Artificial Intelligence

*The capability of a machine to*
*match or exceed intelligent human behavior by*
***training a machine***
***to learn the desired behavior***

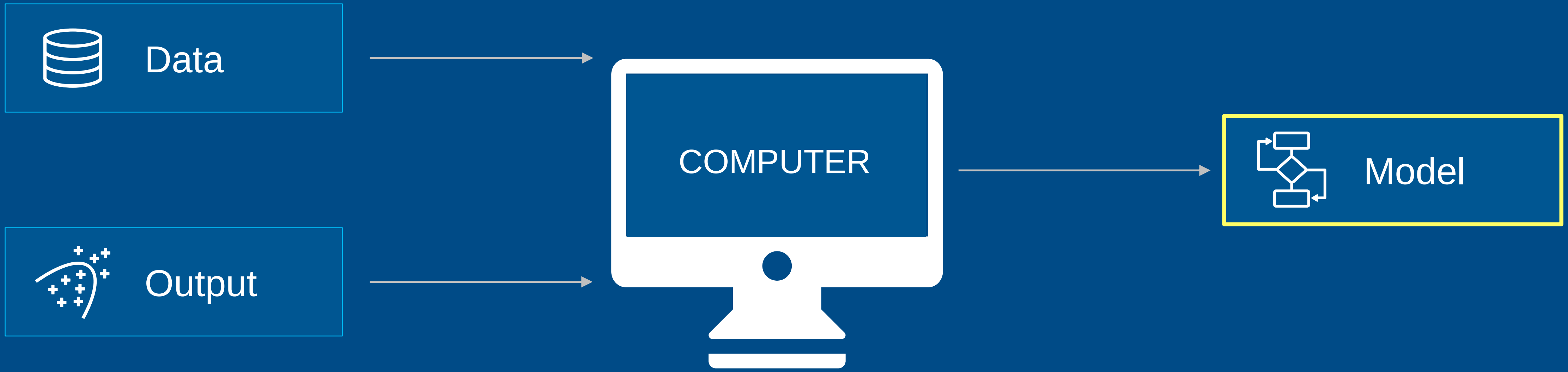
# There are two ways to get a computer to do what you want

## Traditional Programming



# There are two ways to get a computer to do what you want

## Machine Learning



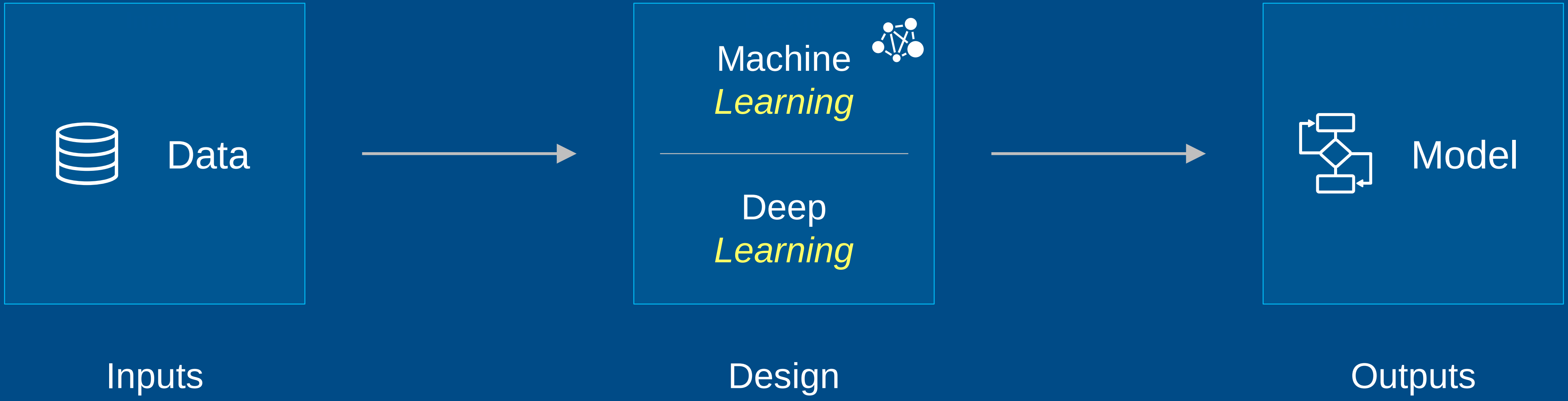
Data

Output

COMPUTER

Model

# Using MATLAB and Simulink to Build Deep Learning Models



Data

Machine *Learning*

Deep *Learning*

Model

Inputs

Design

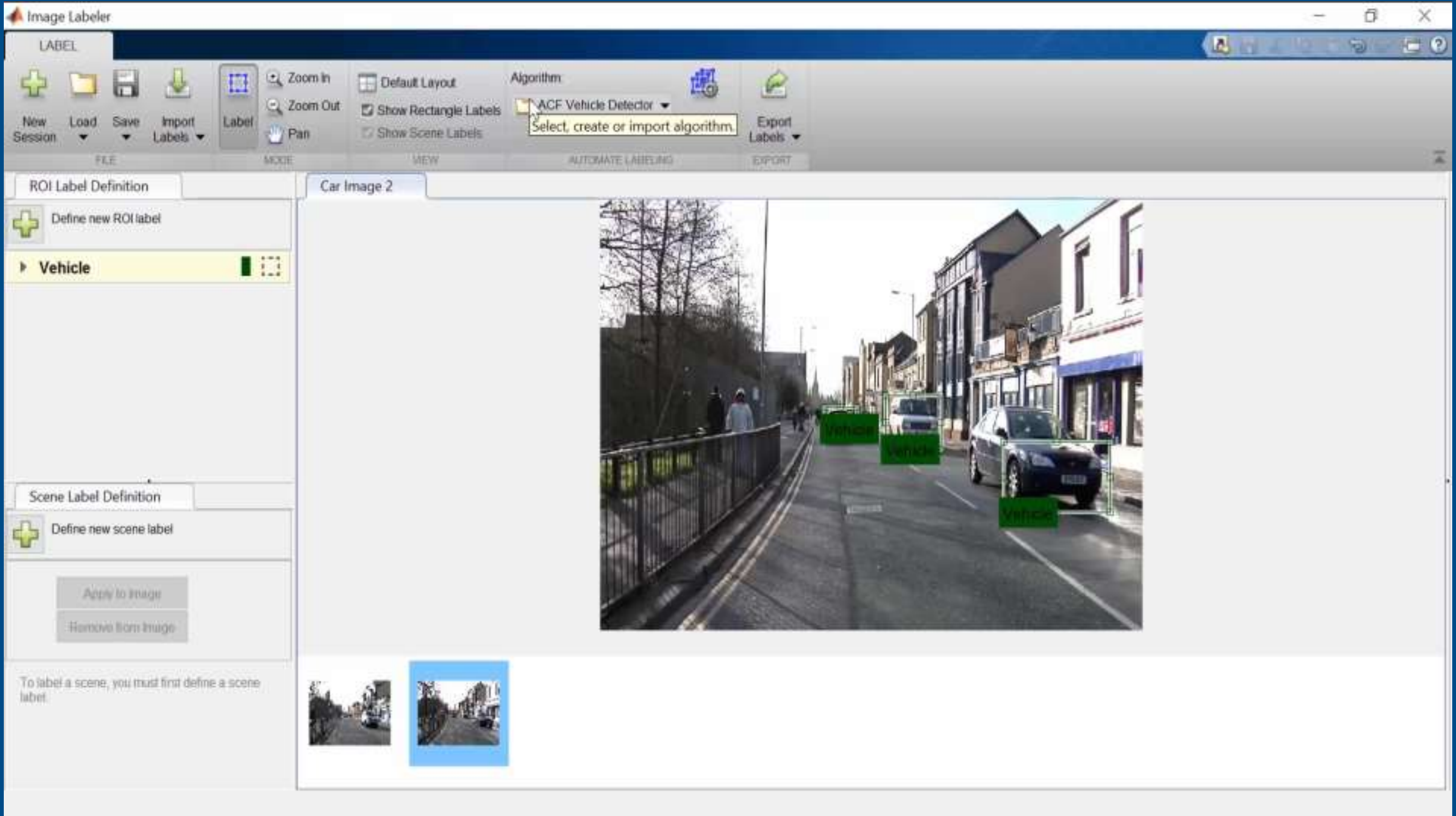
Outputs

MATLAB®&SIMULINK®

Input

# Using Apps for Ground Truth Labeling

## Image and Video Data



Algorithm: ACF Vehicle Detector

Select, create or import algorithm.

☑ Show Rectangle Labels

☑ Show Scene Labels

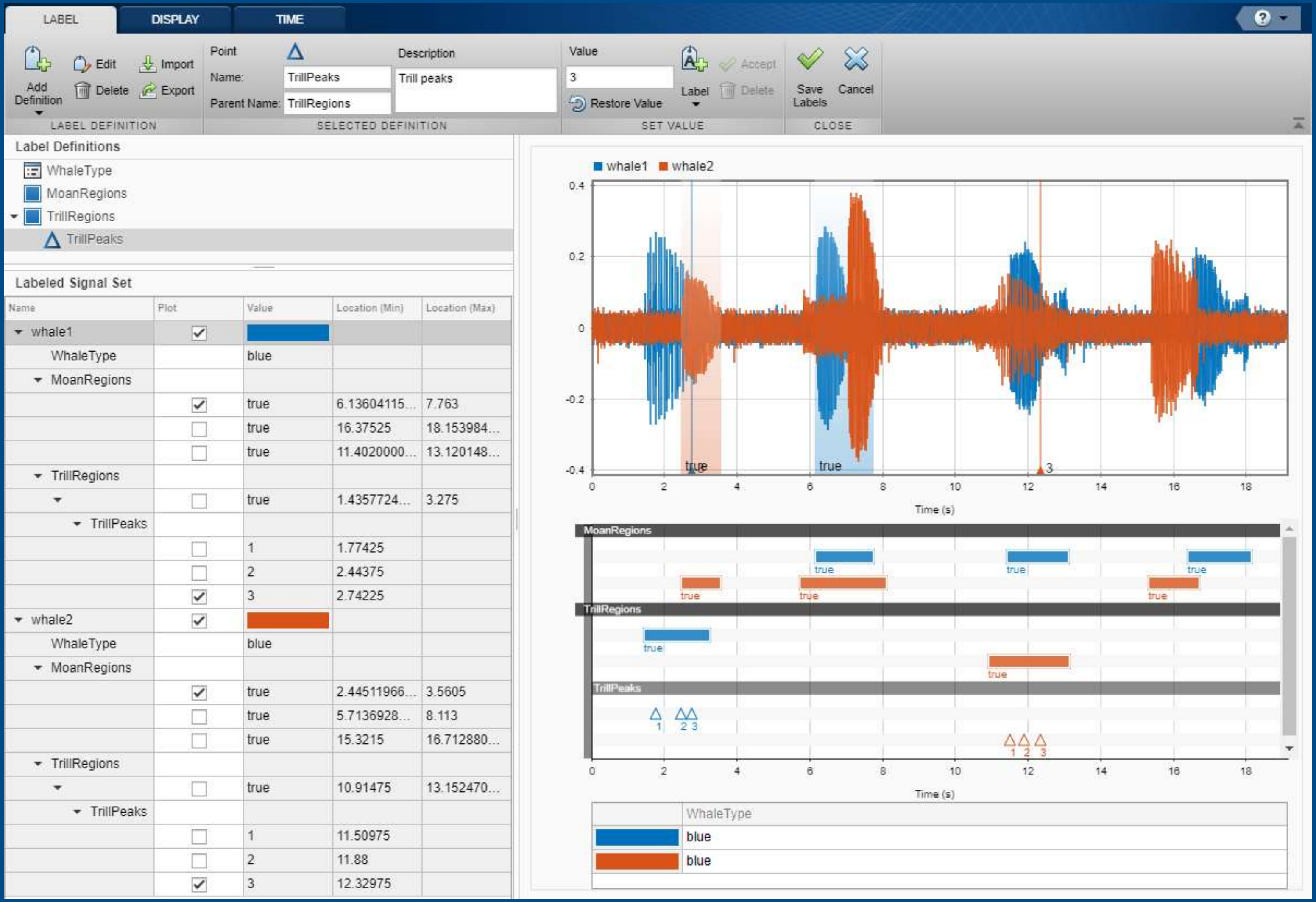
To label a scene, you must first define a scene label.

Computer Vision Toolbox

# Using Apps for Ground Truth Labeling

## Signal Data

Input



Point

Name: TrillPeaks

Parent Name: TrillRegions

Description: Trill peaks

Value: 3

Label Definitions

- WhaleType
- MoanRegions
- TrillRegions
  - TrillPeaks

Labeled Signal Set

| Name | Plot | Value | Location (Min) | Location (Max) |
|---|---|---|---|---|
| whale1 | ☑ | 🟦 | | |
| WhaleType | | blue | | |
| MoanRegions | | | | |
| | ☑ | true | 6.13604115... | 7.763 |
| | □ | true | 16.37525 | 18.153984... |
| | □ | true | 11.4020000... | 13.120148... |
| TrillRegions | | | | |
| | □ | true | 1.4357724... | 3.275 |
| TrillPeaks | | | | |
| | □ | 1 | 1.77425 | |
| | □ | 2 | 2.44375 | |
| | ☑ | 3 | 2.74225 | |
| whale2 | ☑ | 🟧 | | |
| WhaleType | | blue | | |
| MoanRegions | | | | |
| | ☑ | true | 2.44511966... | 3.5605 |
| | □ | true | 5.7136928... | 8.113 |
| | □ | true | 15.3215 | 16.712880... |
| TrillRegions | | | | |
| | □ | true | 10.91475 | 13.152470... |
| TrillPeaks | | | | |
| | □ | 1 | 11.50975 | |
| | □ | 2 | 11.88 | |
| | ☑ | 3 | 12.32975 | |

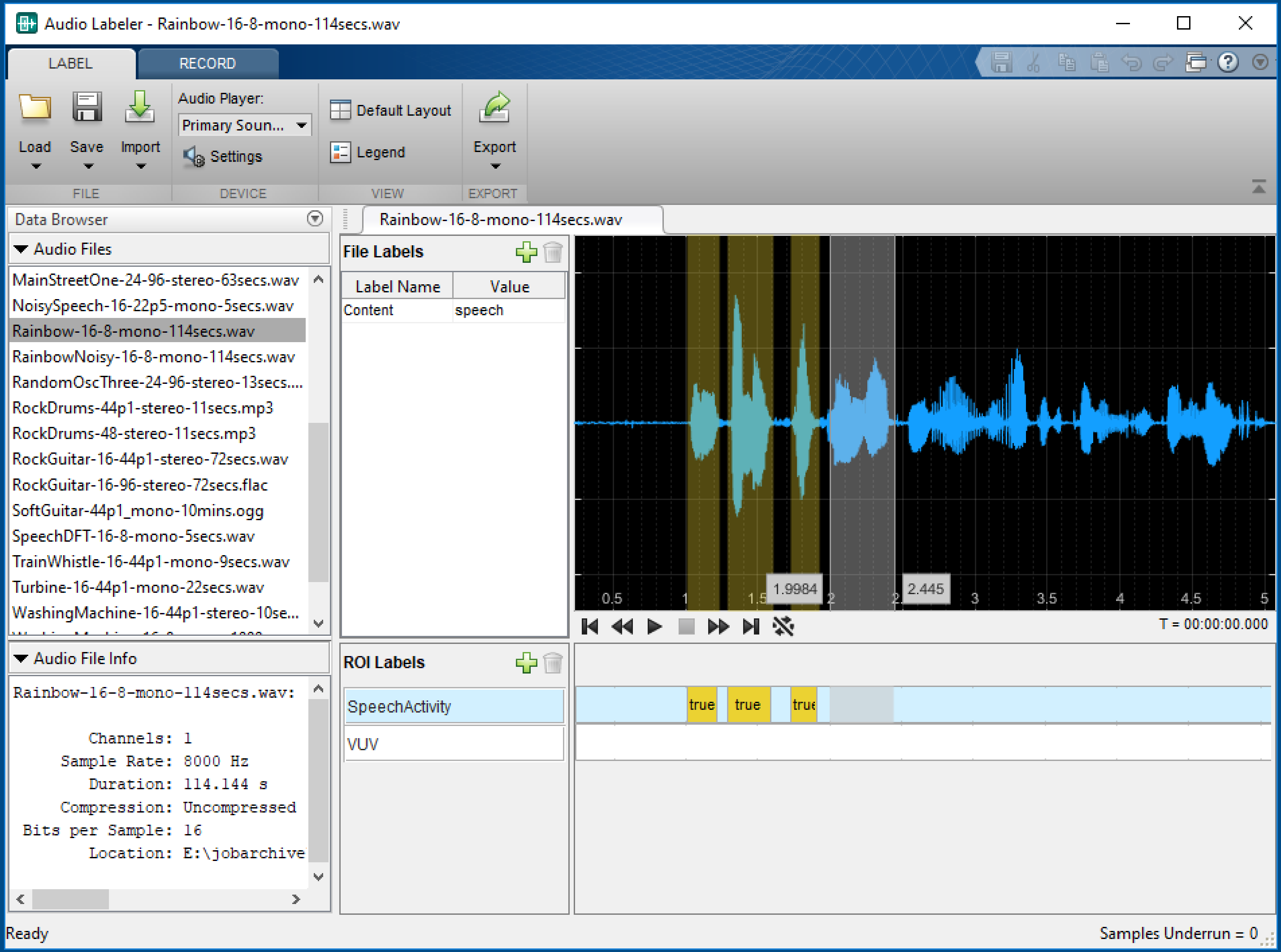
| | WhaleType |
|---|---|
| 🟦 | blue |
| 🟧 | blue |

Signal Processing Toolbox

Input

# Using Apps for Ground Truth Labeling

## Audio Data



Audio Files

- MainStreetOne-24-96-stereo-63secs.wav
- NoisySpeech-16-22p5-mono-5secs.wav
- Rainbow-16-8-mono-114secs.wav
- RainbowNoisy-16-8-mono-114secs.wav
- RandomOscThree-24-96-stereo-13secs....
- RockDrums-44p1-stereo-11secs.mp3
- RockDrums-48-stereo-11secs.mp3
- RockGuitar-16-44p1-stereo-72secs.wav
- RockGuitar-16-96-stereo-72secs.flac
- SoftGuitar-44p1_mono-10mins.ogg
- SpeechDFT-16-8-mono-5secs.wav
- TrainWhistle-16-44p1-mono-9secs.wav
- Turbine-16-44p1-mono-22secs.wav
- WashingMachine-16-44p1-stereo-10se...

Audio File Info

```
Rainbow-16-8-mono-114secs.wav:

        Channels: 1
     Sample Rate: 8000 Hz
        Duration: 114.144 s
     Compression: Uncompressed
 Bits per Sample: 16
        Location: E:\jobarchive
```

| Label Name | Value |
|---|---|
| Content | speech |

ROI Labels: SpeechActivity, VUV

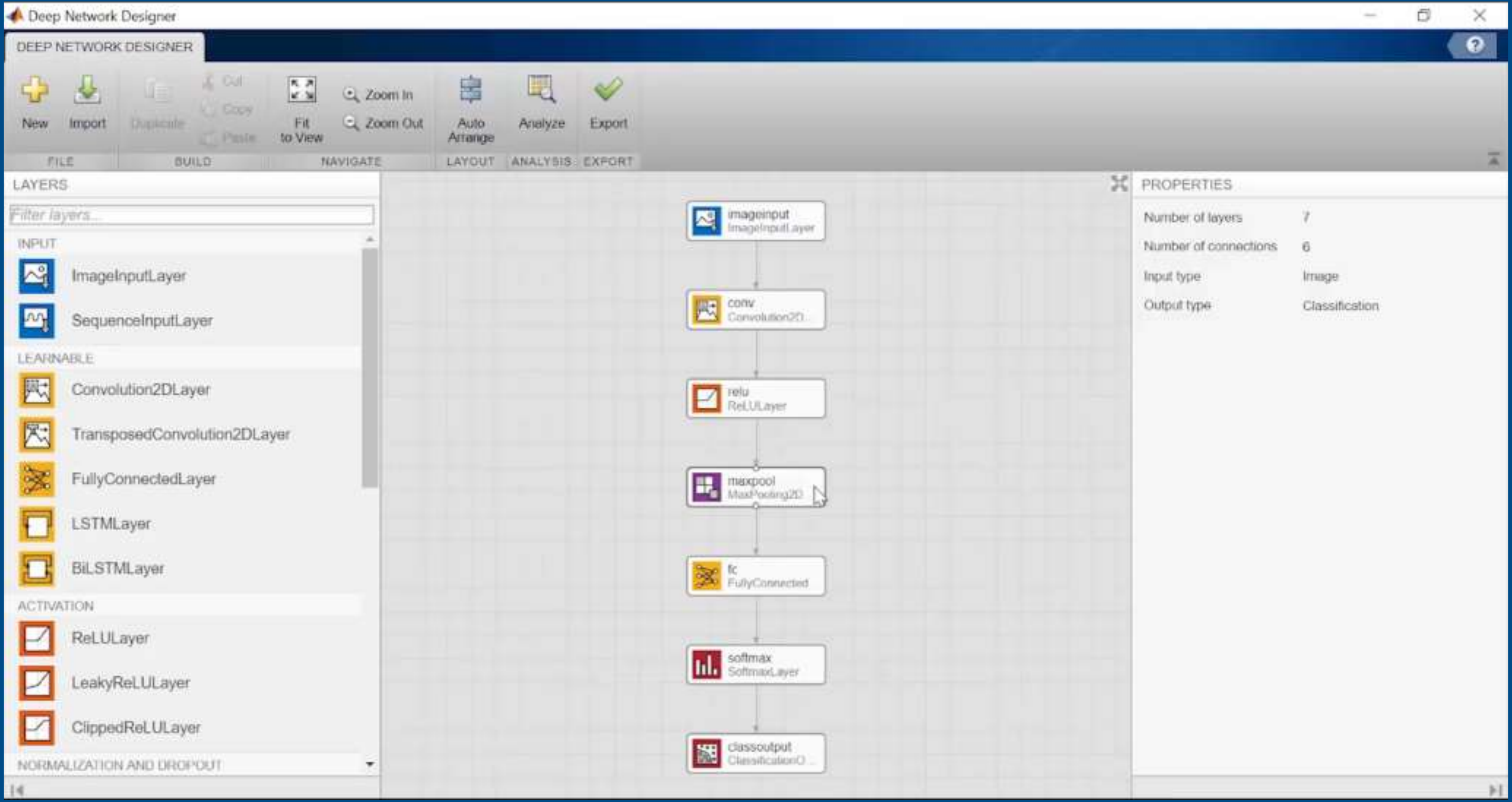
T = 00:00:00.000

Ready

Samples Underrun = 0

Audio Toolbox

# Using Apps for Designing Deep Learning Networks

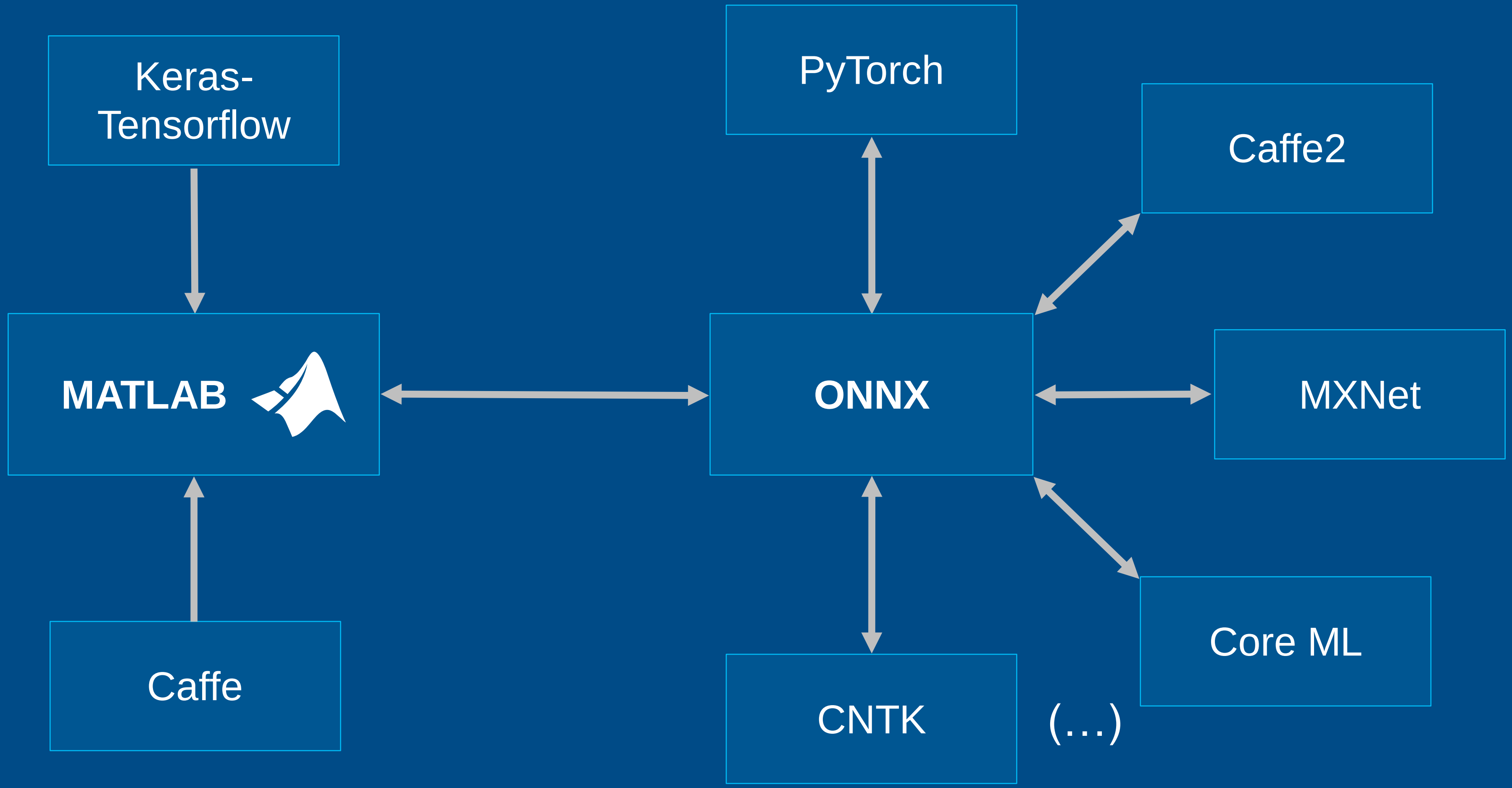


| | |
|---|---|
| Number of layers | 7 |
| Number of connections | 6 |
| Input type | Image |
| Output type | Classification |

Deep Learning Toolbox

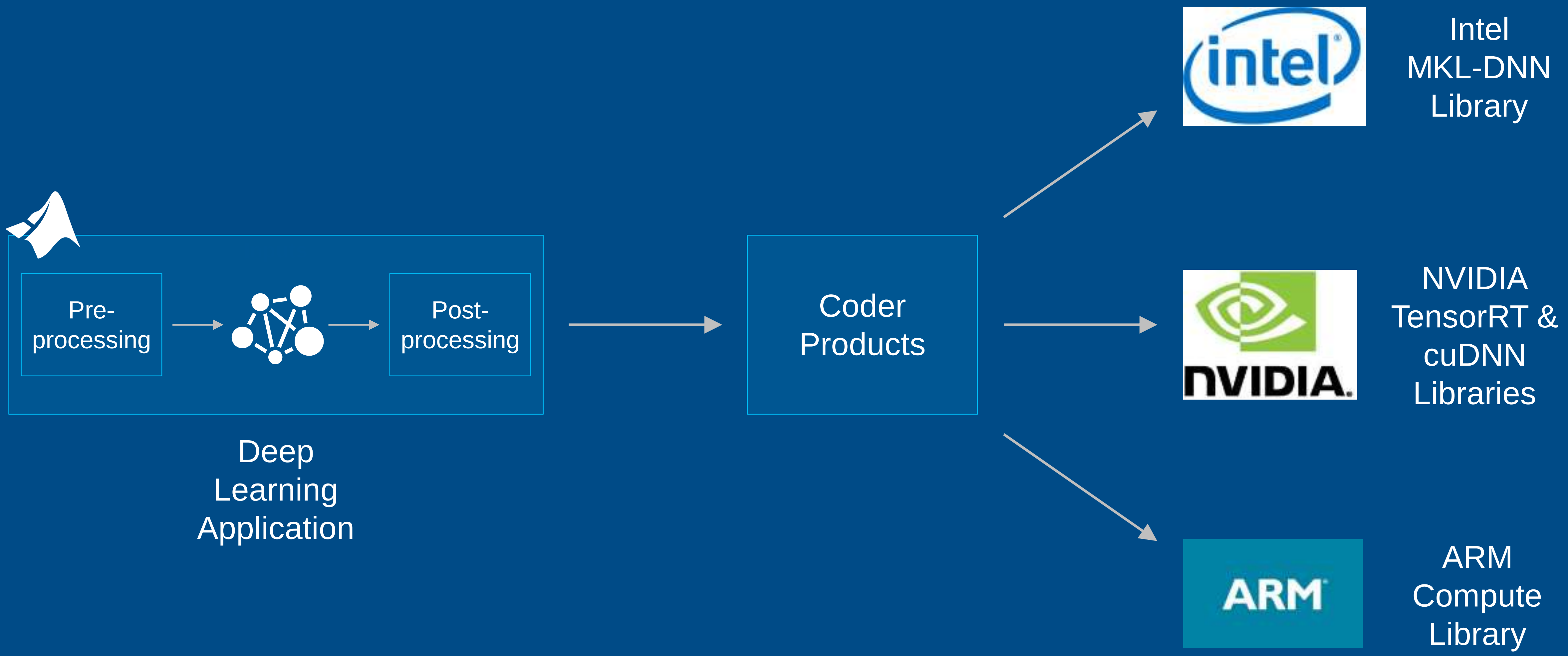
# Using Models from Other Frameworks

Design



Deep Learning Toolbox

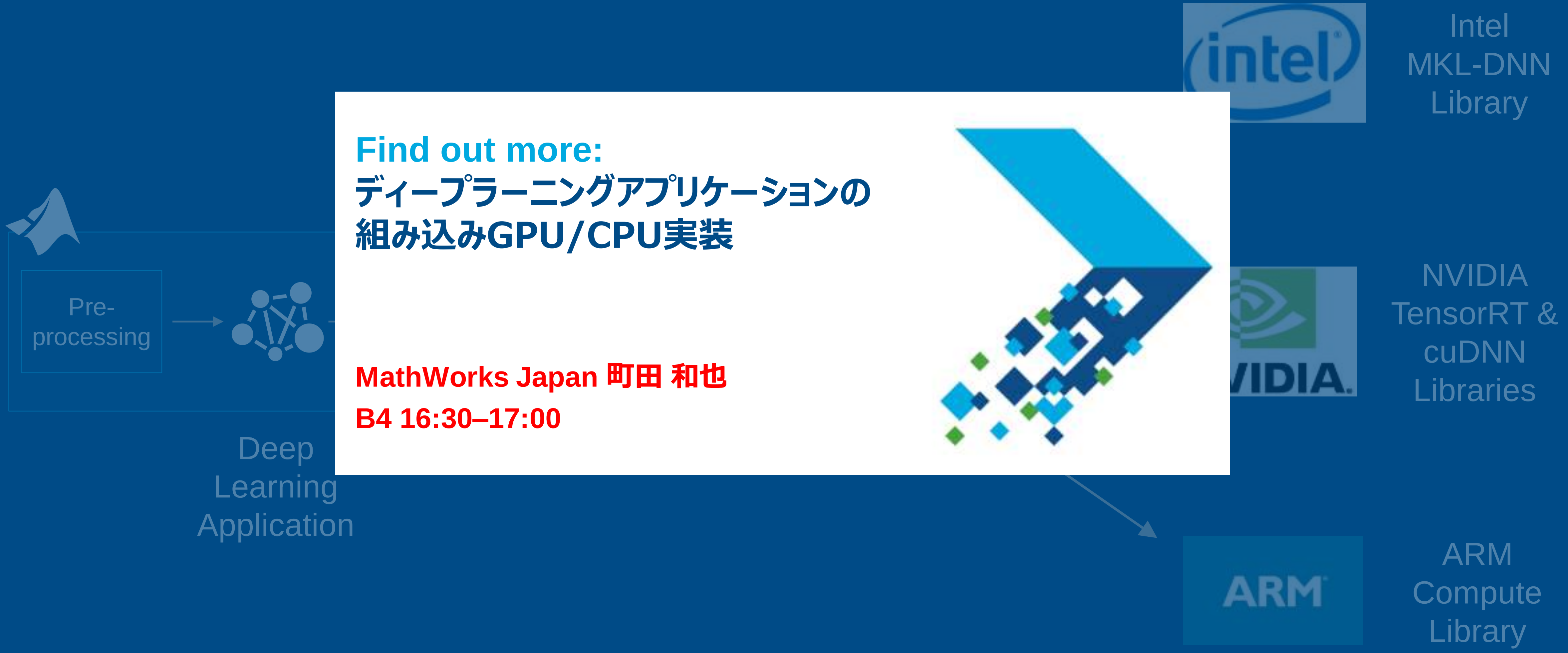
# Deploying Deep Learning Applications



MATLAB Coder
GPU Coder

# Deploying Deep Learning Applications

Output



**Find out more:**

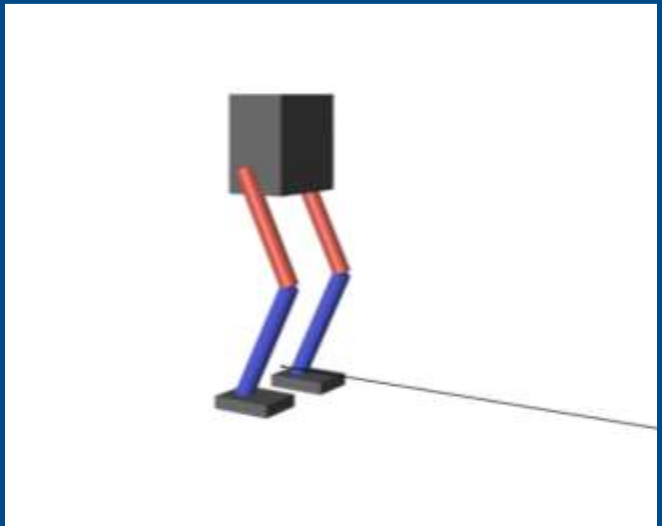
**ディープラーニングアプリケーションの組み込みGPU/CPU実装**

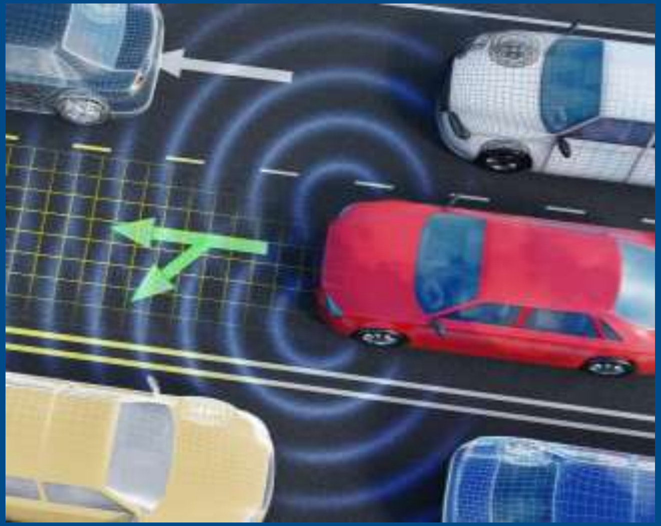
**MathWorks Japan 町田 和也**

**B4 16:30–17:00**

MATLAB Coder
GPU Coder

# Using MATLAB and Simulink for Reinforcement Learning





**Reinforcement Learning Toolbox**

# Using MATLAB and Simulink for Reinforcement Learning

Data

**Find out more:**

**強化学習：最適制御のための**
**ディープラーニングの応用**

**MathWorks Japan 吉田 剛士**
**B2 14:30-15:00**

Model

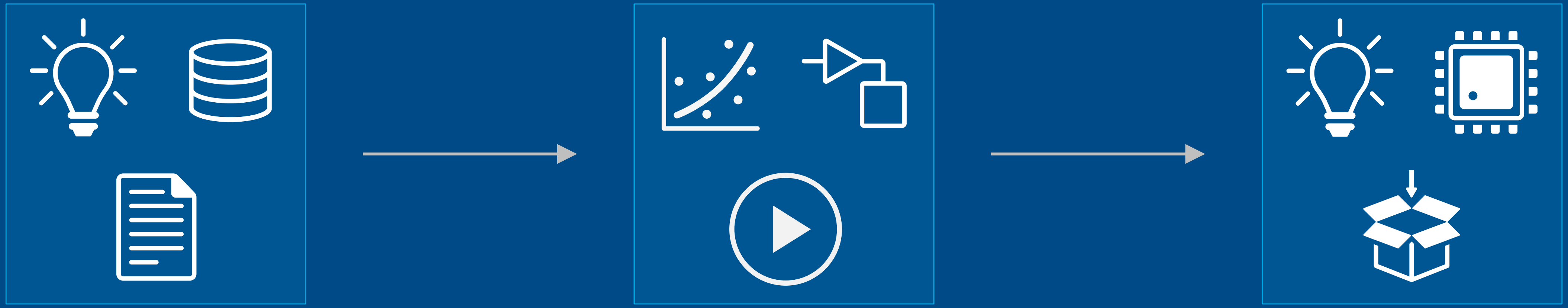
Inputs

Outputs

MATLAB®&SIMULINK®

Reinforcement Learning Toolbox

# Using MATLAB & Simulink to Build Algorithms in Everything



Inputs

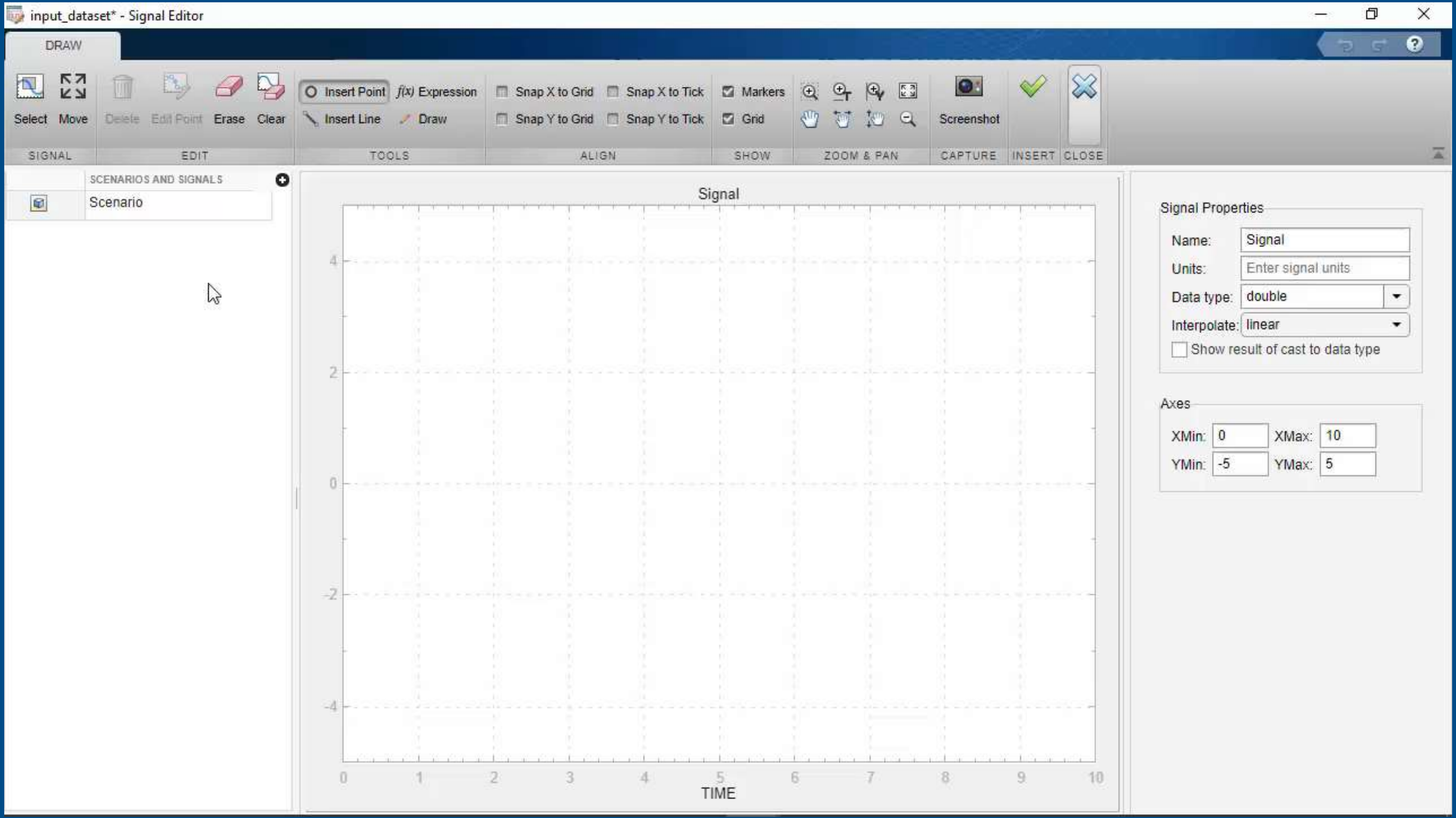
Design

Outputs

MATLAB®&SIMULINK®

# Creating Your Own Data

Input



Signal Properties

Name: Signal

Units: Enter signal units

Data type: double

Interpolate: linear

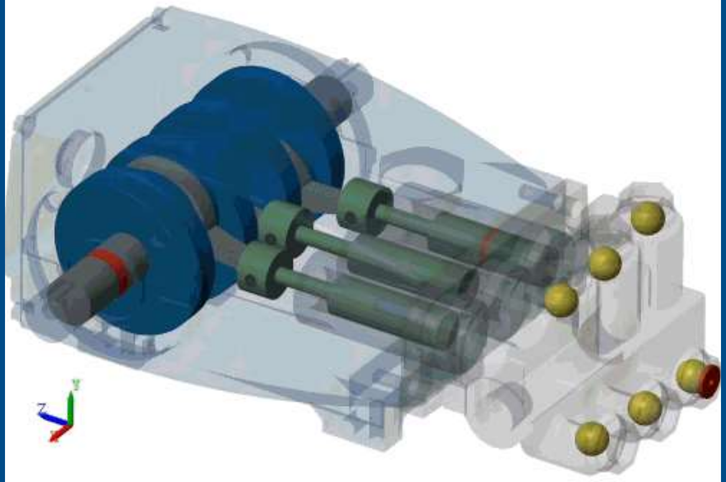
□ Show result of cast to data type

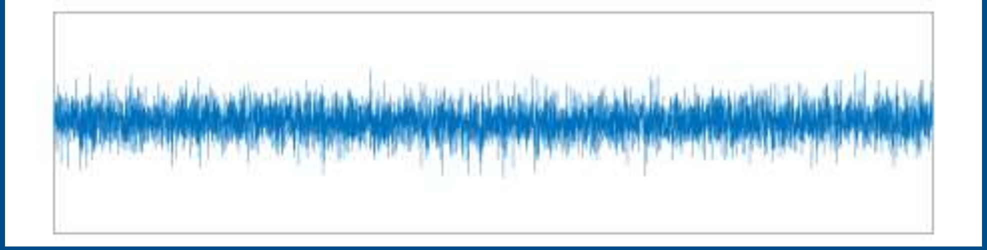
Axes

XMin: 0 XMax: 10

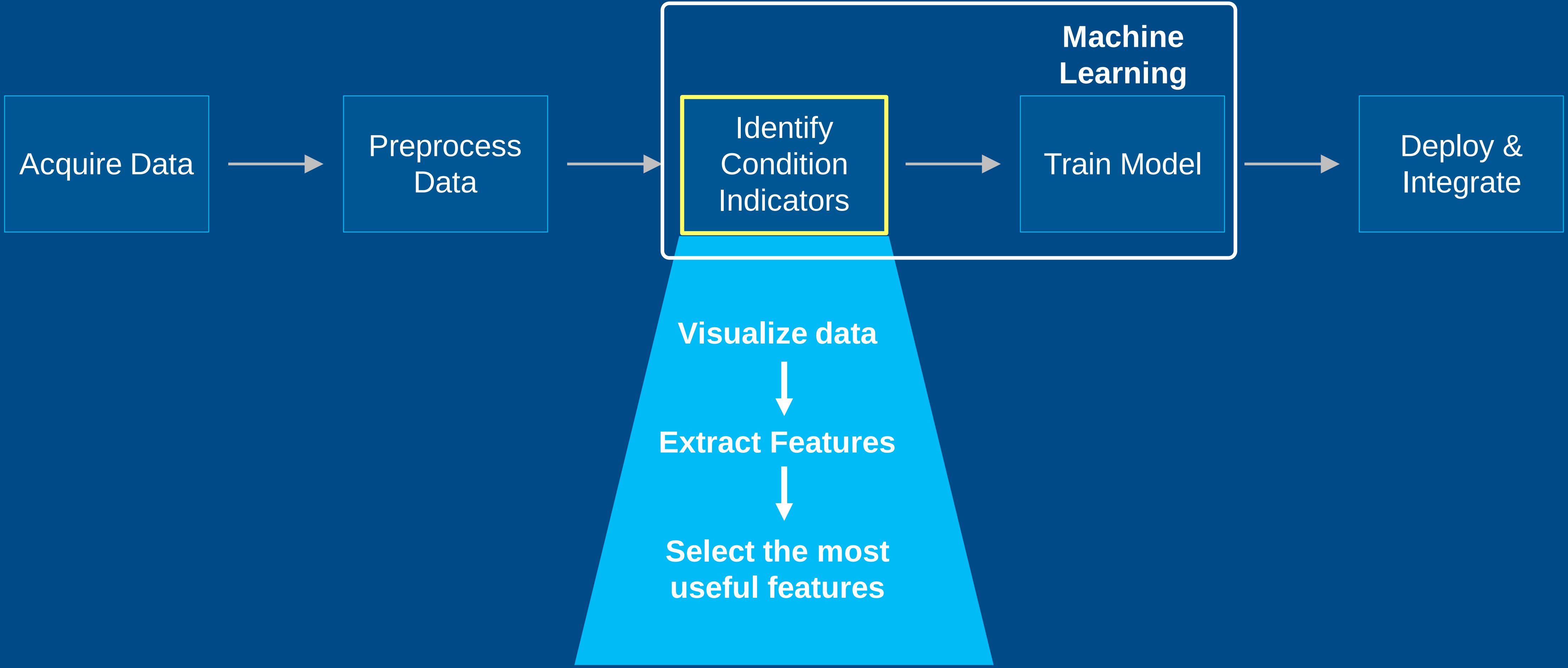
YMin: -5 YMax: 5

Simulink

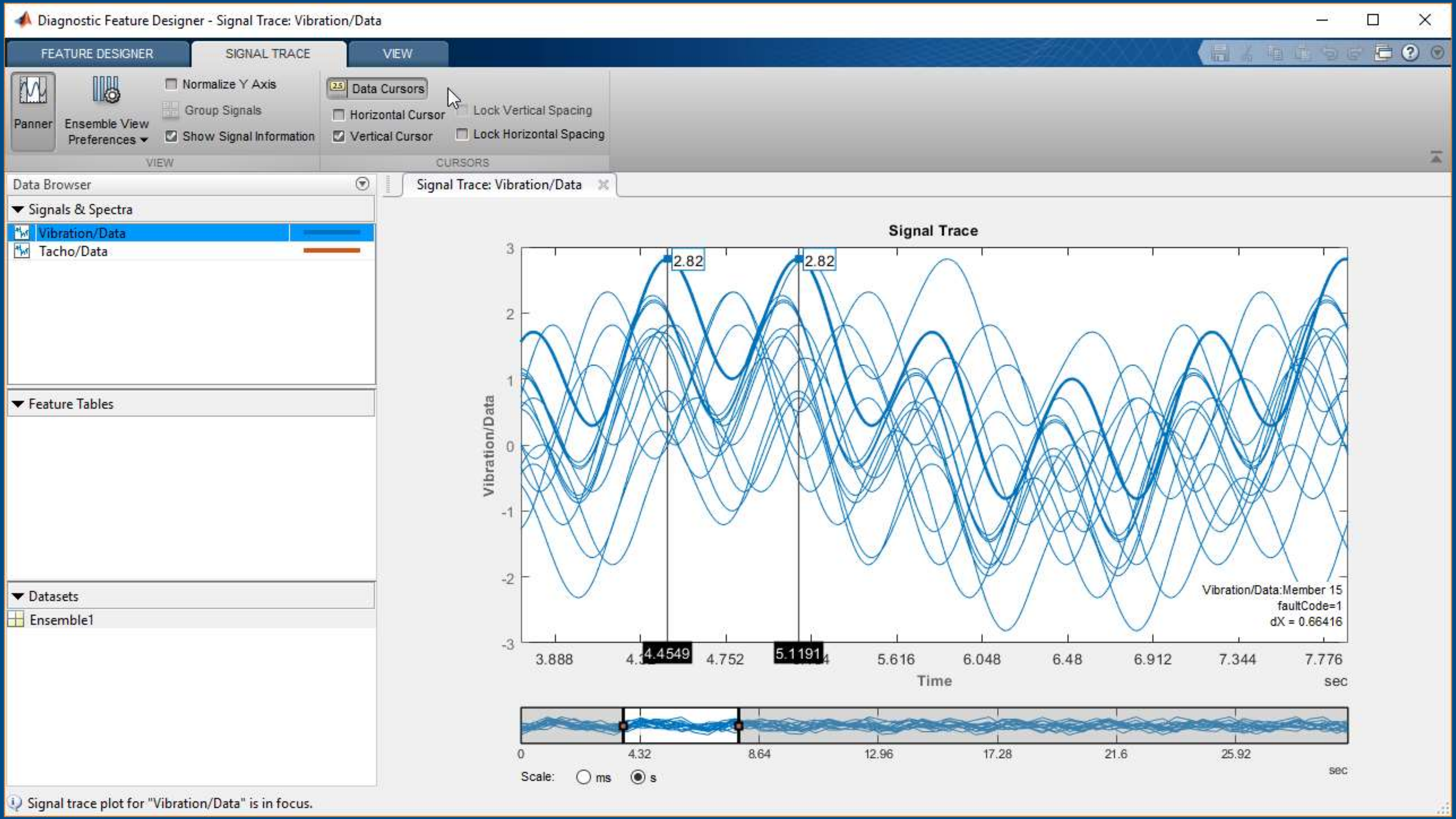




# Identifying the Useful Data



# Identifying the Useful Data



Diagnostic Feature Designer - Signal Trace: Vibration/Data

☐ Normalize Y Axis
Group Signals
☑ Show Signal Information

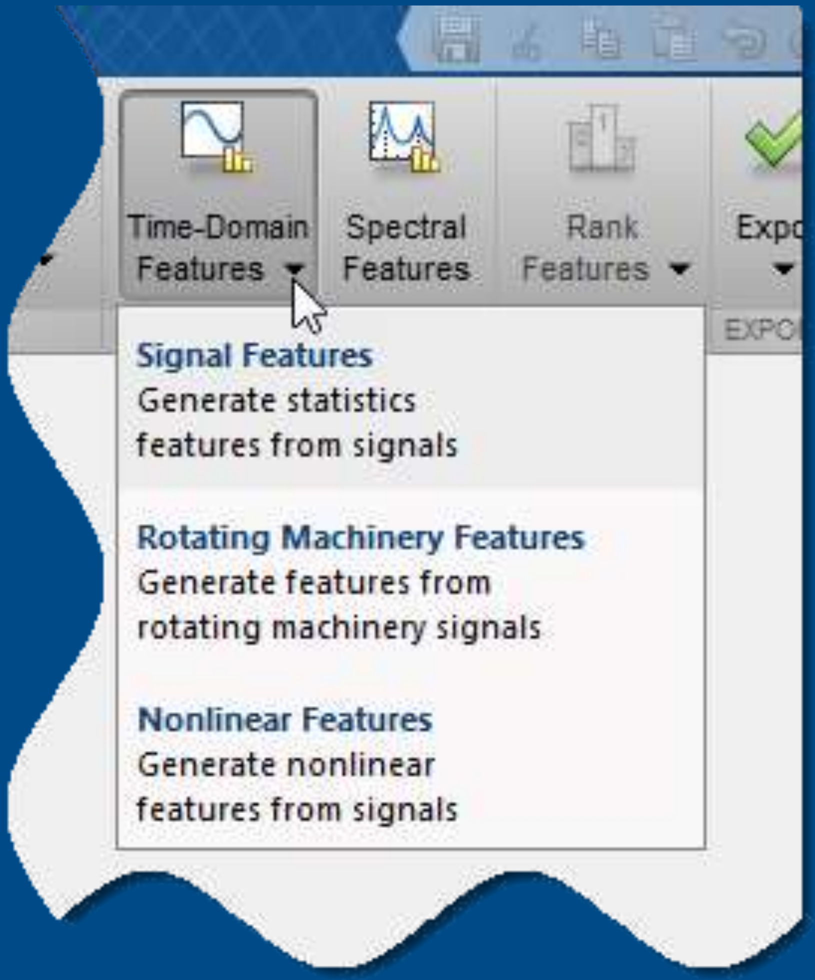
Data Cursors
☐ Horizontal Cursor
☑ Vertical Cursor
☐ Lock Vertical Spacing
☐ Lock Horizontal Spacing

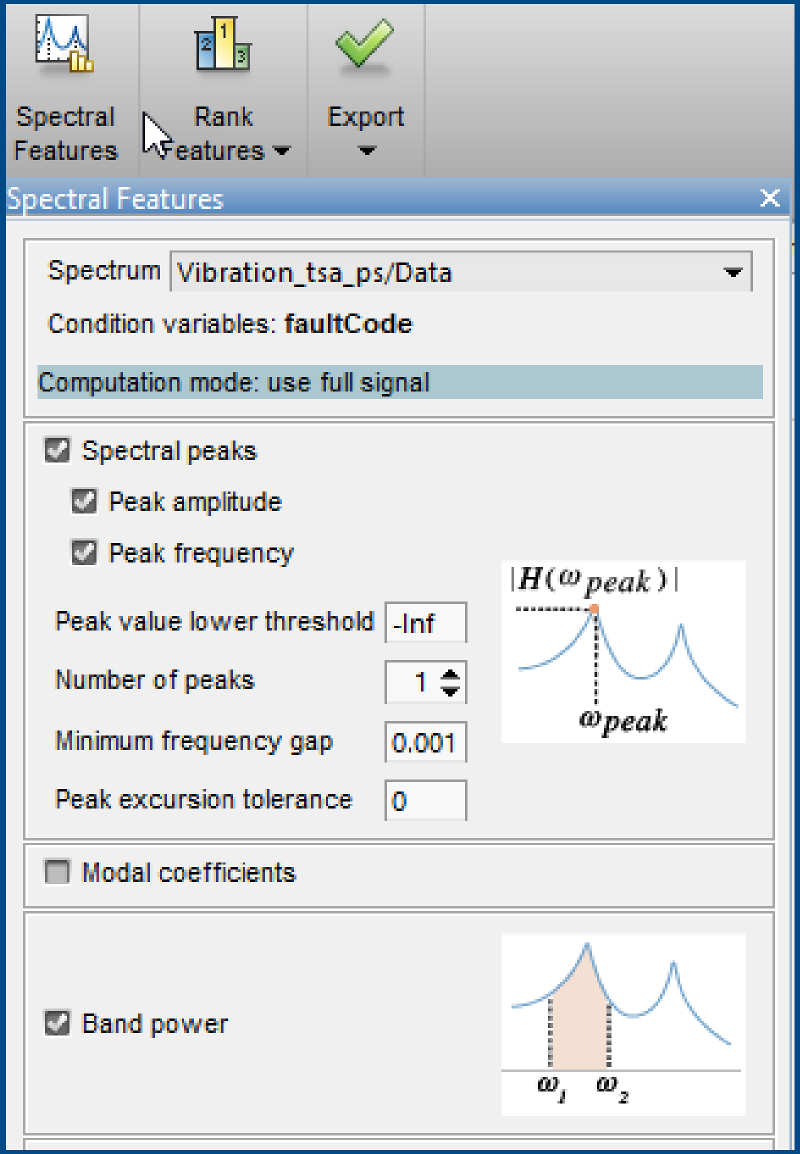
Vibration/Data:Member 15
faultCode=1
dX = 0.66416

Scale: ○ ms ● s

Signal trace plot for "Vibration/Data" is in focus.

# Identifying the Useful Data


Time-Domain Features
Spectral Features
Rank Features
Expo
EXPO
Signal Features
Generate statistics features from signals
Rotating Machinery Features
Generate features from rotating machinery signals
Nonlinear Features
Generate nonlinear features from signals



Spectrum: Vibration_tsa_ps/Data

Condition variables: **faultCode**

Computation mode: use full signal

☑ Spectral peaks

☑ Peak amplitude

☑ Peak frequency

Peak value lower threshold: -Inf

Number of peaks: 1

Minimum frequency gap: 0.001

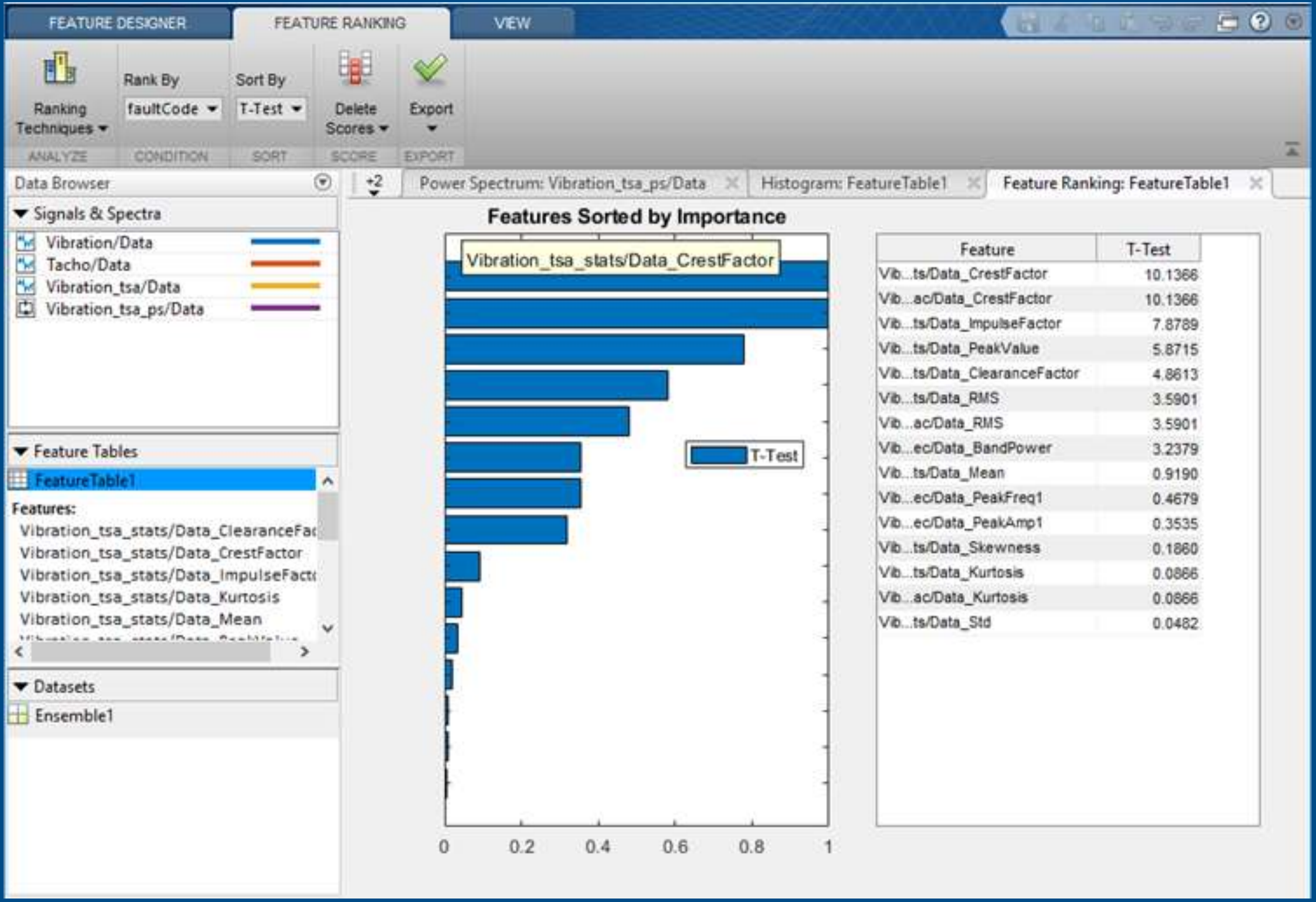
Peak excursion tolerance: 0

□ Modal coefficients

☑ Band power

Design

# Identifying the Useful Data



Rank By: faultCode

Sort By: T-Test

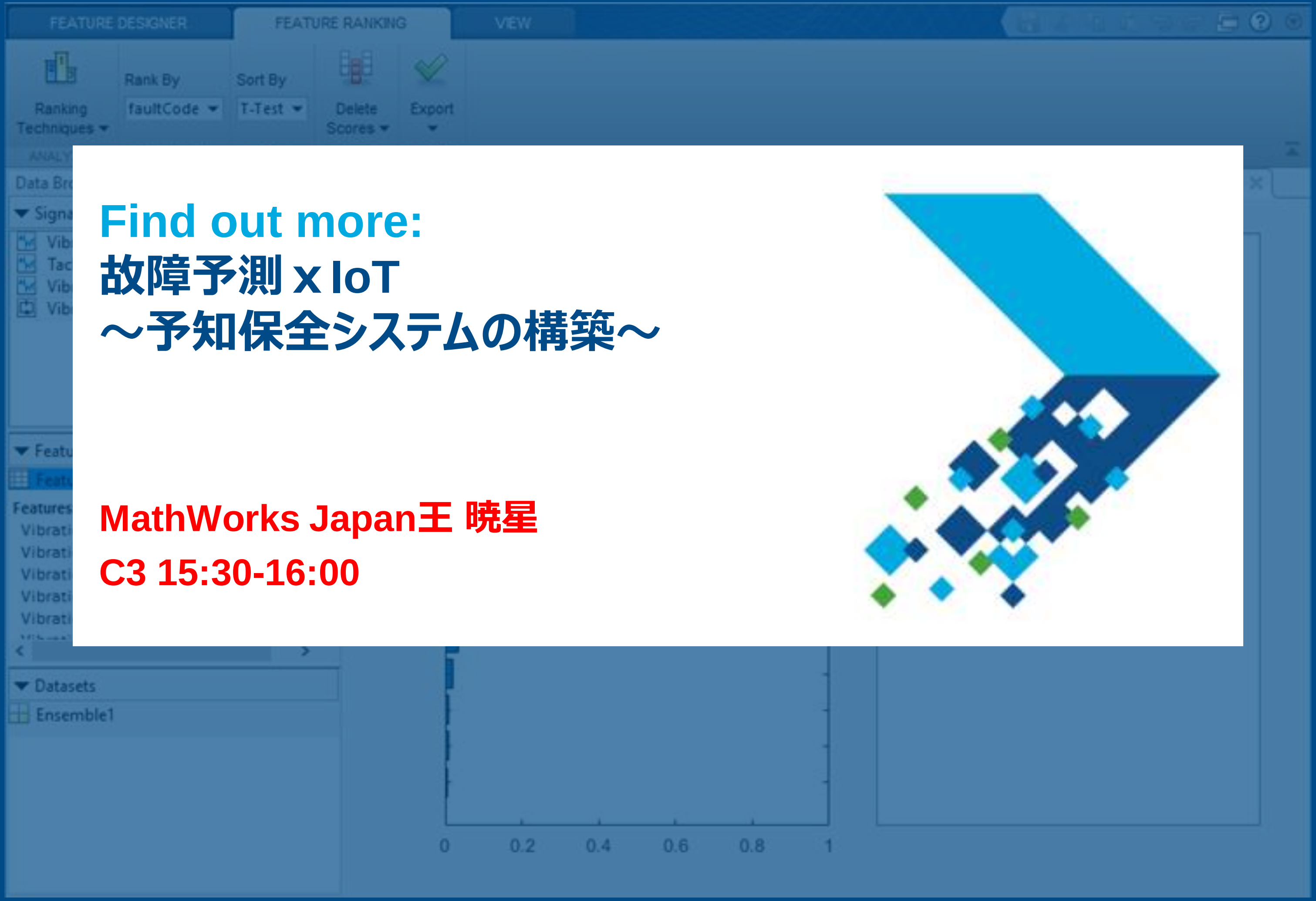
Features:

Vibration_tsa_stats/Data_ClearanceFac
Vibration_tsa_stats/Data_CrestFactor
Vibration_tsa_stats/Data_ImpulseFact
Vibration_tsa_stats/Data_Kurtosis
Vibration_tsa_stats/Data_Mean

| Feature | T-Test |
|---|---|
| Vib...ts/Data_CrestFactor | 10.1366 |
| Vib...ac/Data_CrestFactor | 10.1366 |
| Vib...ts/Data_ImpulseFactor | 7.8789 |
| Vib...ts/Data_PeakValue | 5.8715 |
| Vib...ts/Data_ClearanceFactor | 4.8613 |
| Vib...ts/Data_RMS | 3.5901 |
| Vib...ac/Data_RMS | 3.5901 |
| Vib...ec/Data_BandPower | 3.2379 |
| Vib...ts/Data_Mean | 0.9190 |
| Vib...ec/Data_PeakFreq1 | 0.4679 |
| Vib...ec/Data_PeakAmp1 | 0.3535 |
| Vib...ts/Data_Skewness | 0.1860 |
| Vib...ts/Data_Kurtosis | 0.0866 |
| Vib...ac/Data_Kurtosis | 0.0866 |
| Vib...ts/Data_Std | 0.0482 |

# Identifying the Useful Data

Design



Rank By faultCode

Sort By T-Test

## Find out more:

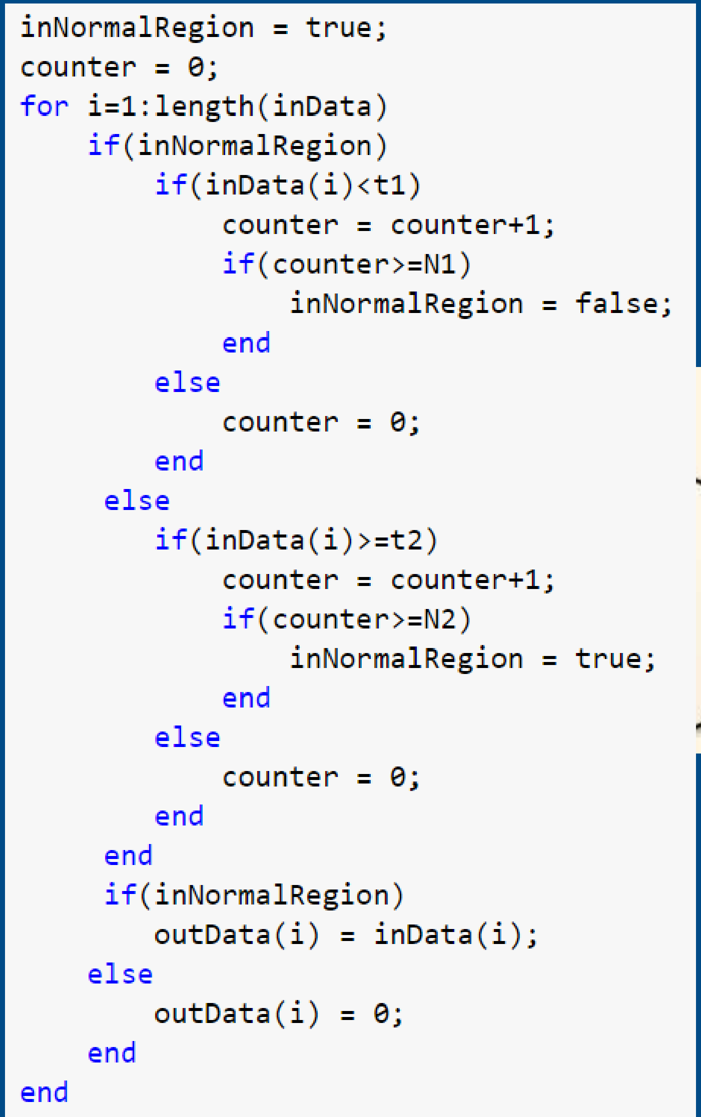
**故障予測 x IoT**
**～予知保全システムの構築～**

**MathWorks Japan王 暁星**

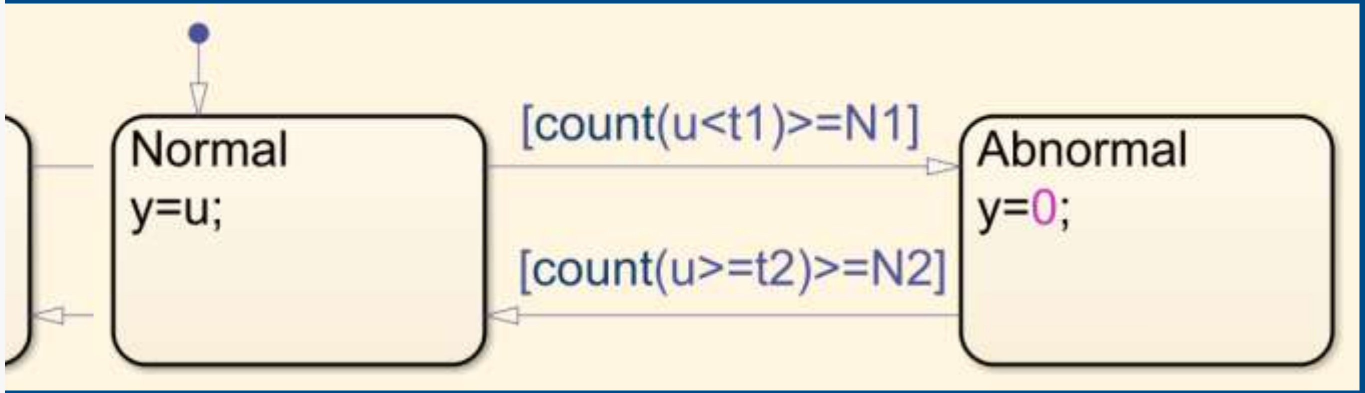
**C3 15:30-16:00**

# Designing Decision Logic with Stateflow

Design

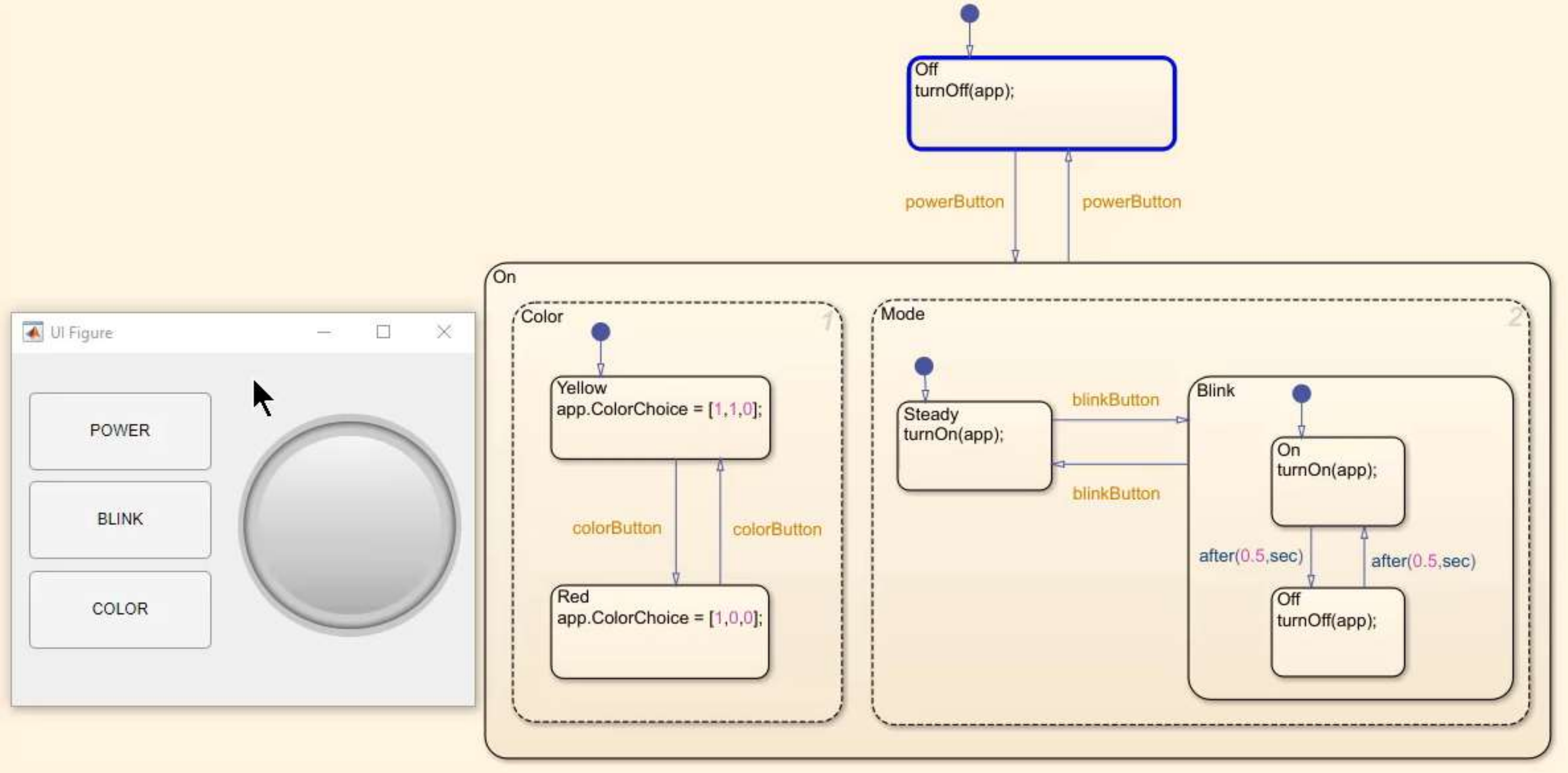


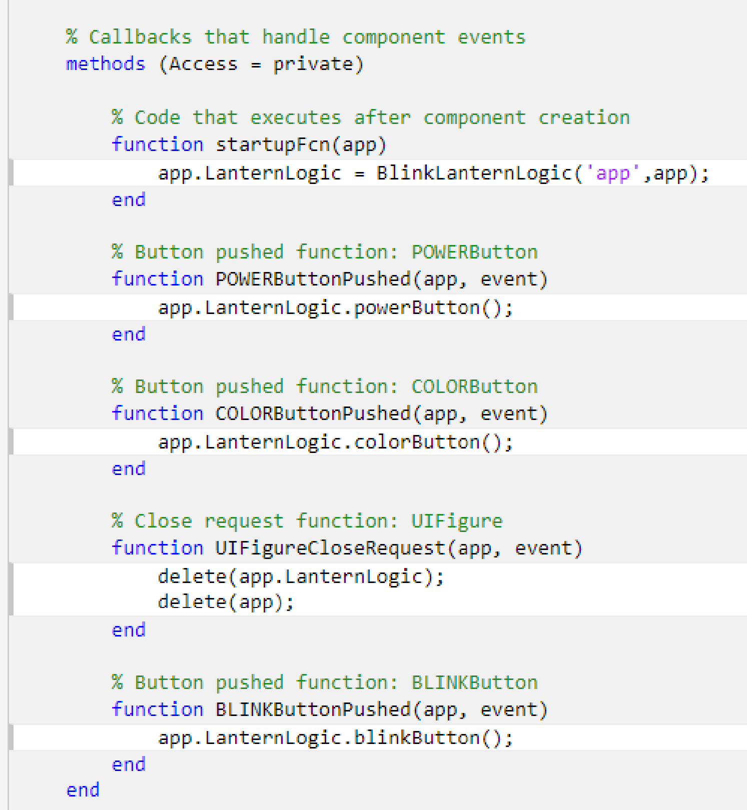
```
inNormalRegion = true;
counter = 0;
for i=1:length(inData)
    if(inNormalRegion)
        if(inData(i)<t1)
            counter = counter+1;
            if(counter>=N1)
                inNormalRegion = false;
            end
        else
            counter = 0;
        end
     else
        if(inData(i)>=t2)
            counter = counter+1;
            if(counter>=N2)
                inNormalRegion = true;
            end
        else
            counter = 0;
        end
     end
     if(inNormalRegion)
        outData(i) = inData(i);
    else
        outData(i) = 0;
    end
end
```



Stateflow
MATLAB

# Using Stateflow in MATLAB



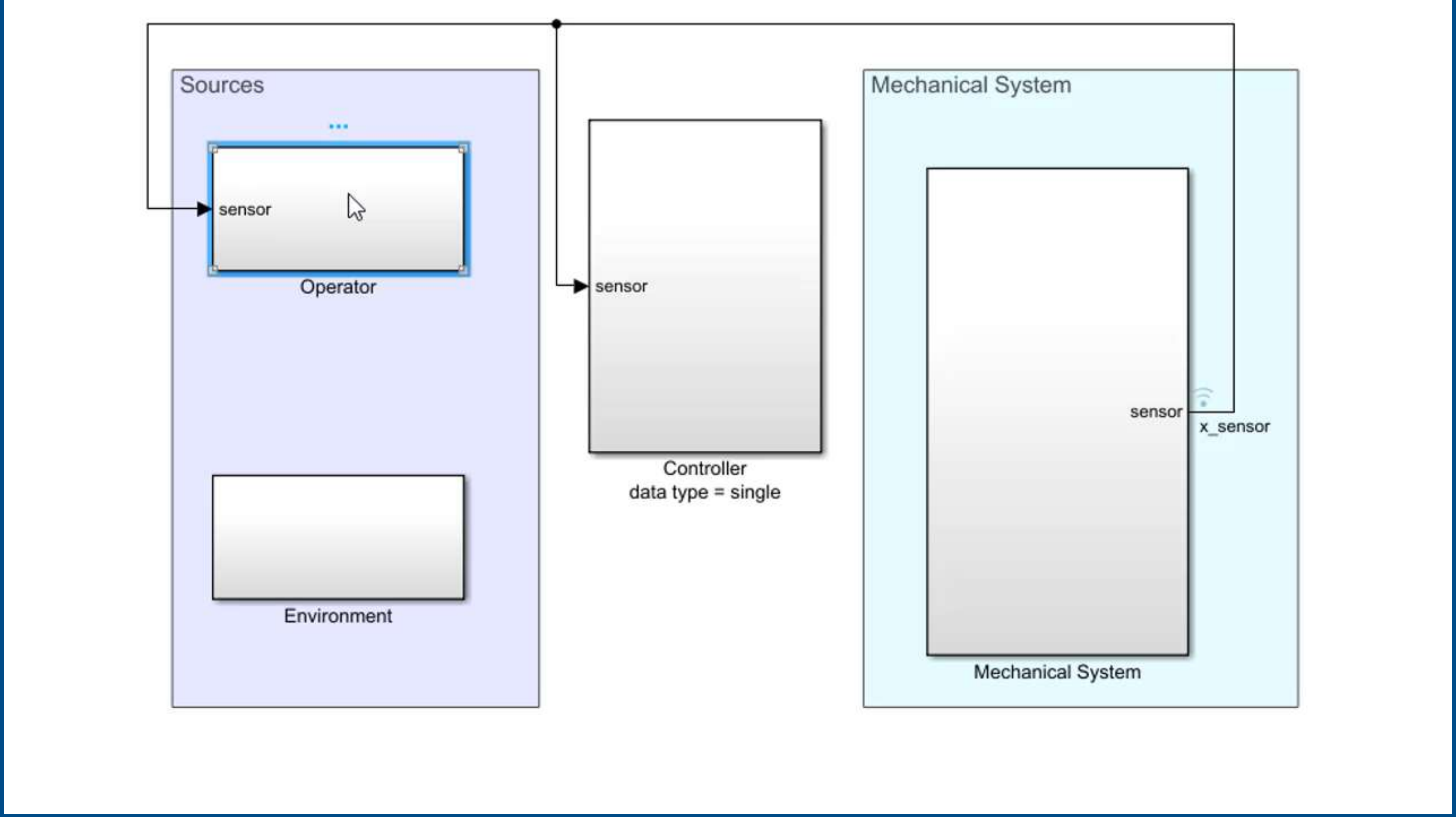


```
% Callbacks that handle component events
methods (Access = private)

    % Code that executes after component creation
    function startupFcn(app)
        app.LanternLogic = BlinkLanternLogic('app',app);
    end

    % Button pushed function: POWERButton
    function POWERButtonPushed(app, event)
        app.LanternLogic.powerButton();
    end

    % Button pushed function: COLORButton
    function COLORButtonPushed(app, event)
        app.LanternLogic.colorButton();
    end

    % Close request function: UIFigure
    function UIFigureCloseRequest(app, event)
        delete(app.LanternLogic);
        delete(app);
    end

    % Button pushed function: BLINKButton
    function BLINKButtonPushed(app, event)
        app.LanternLogic.blinkButton();
    end
end
```

**Stateflow**
**MATLAB**

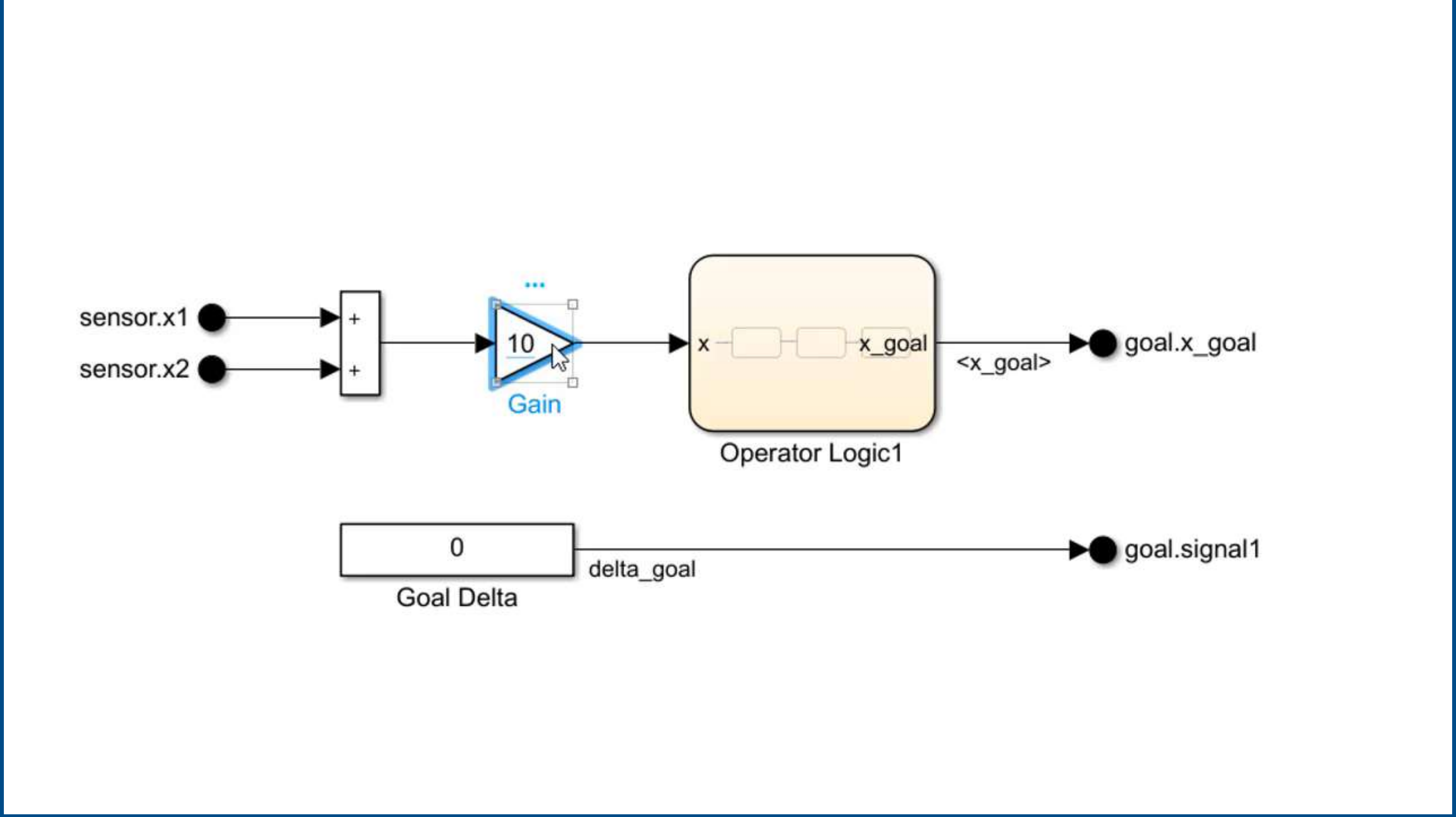
# Editing at the Speed of Thought

Design



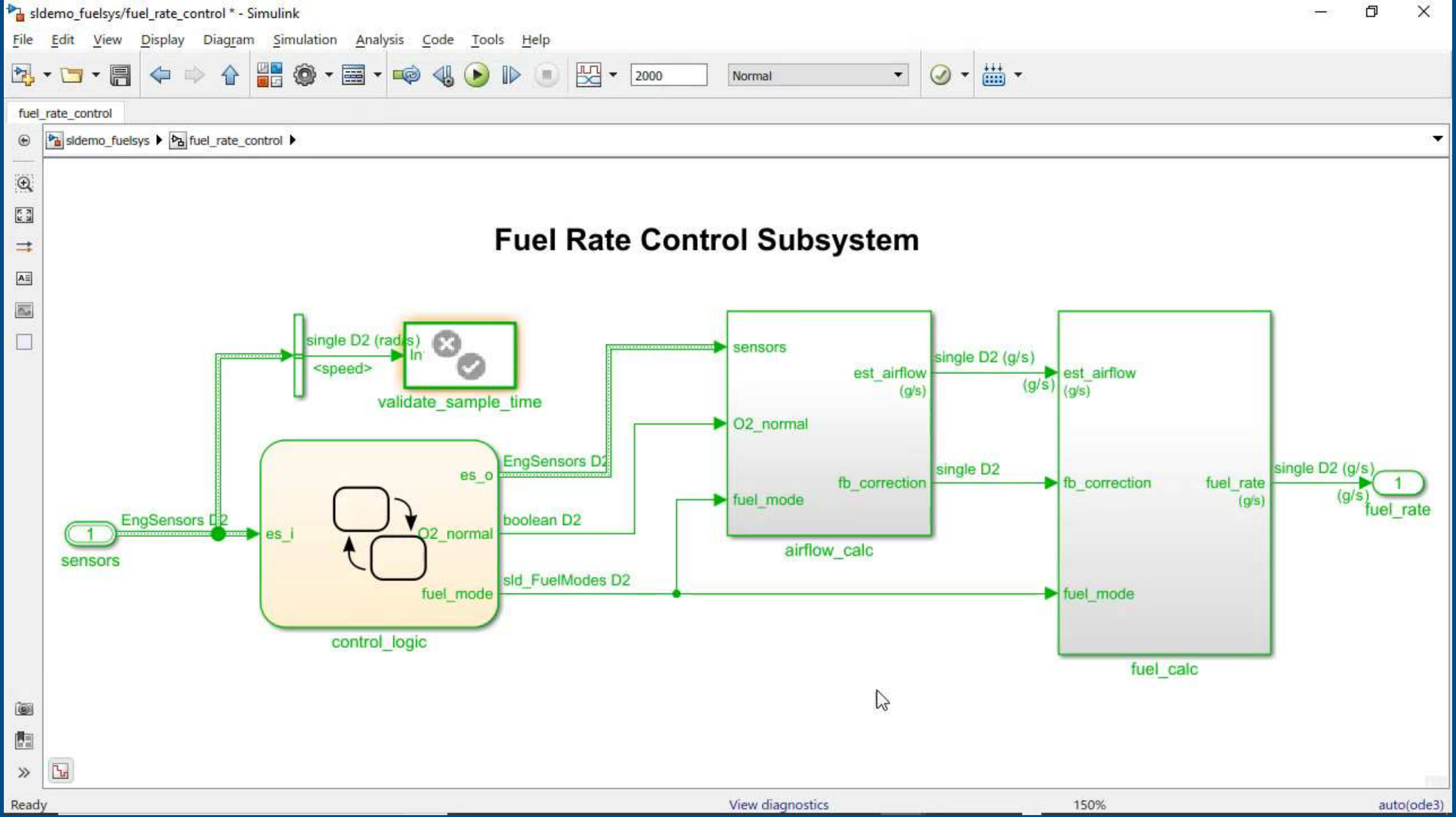
Design

# Editing at the Speed of Thought



# Viewing Generated Code Alongside the Model

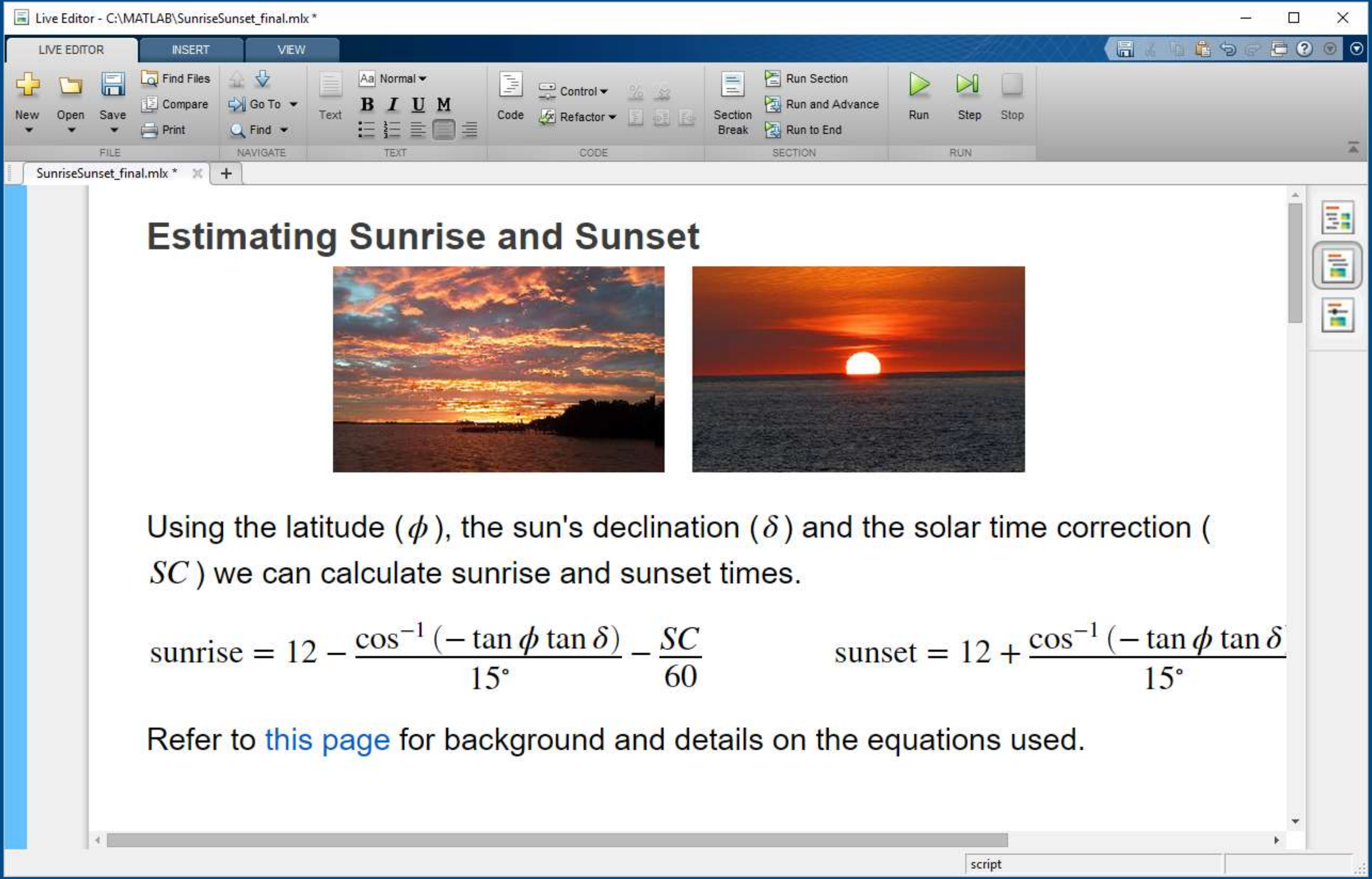
Output



Embedded Coder

# Sharing Live Scripts

Output



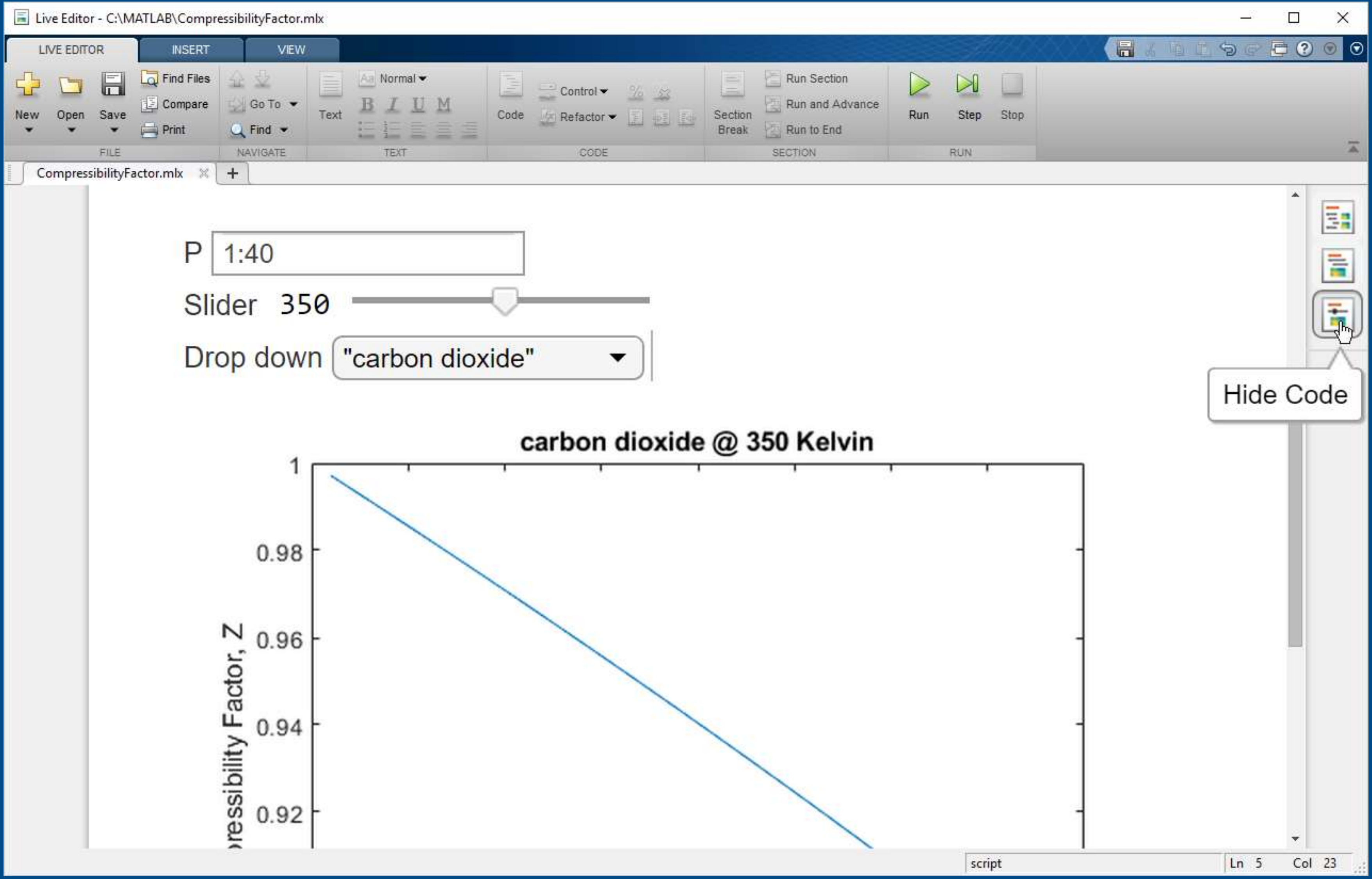
## Estimating Sunrise and Sunset

Using the latitude ($\phi$), the sun's declination ($\delta$) and the solar time correction ( $SC$ ) we can calculate sunrise and sunset times.

$$\text{sunrise} = 12 - \frac{\cos^{-1}(-\tan\phi\tan\delta)}{15^\circ} - \frac{SC}{60} \qquad \text{sunset} = 12 + \frac{\cos^{-1}(-\tan\phi\tan\delta}{15^\circ}$$

Refer to this page for background and details on the equations used.

# Sharing Live Scripts

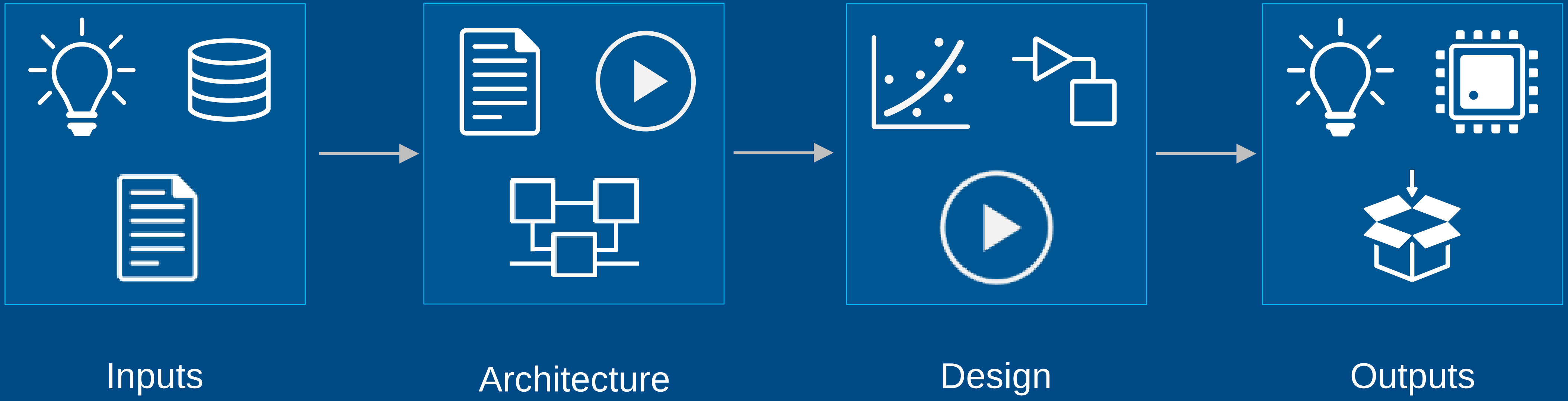


P 1:40

Slider 350

Drop down "carbon dioxide"

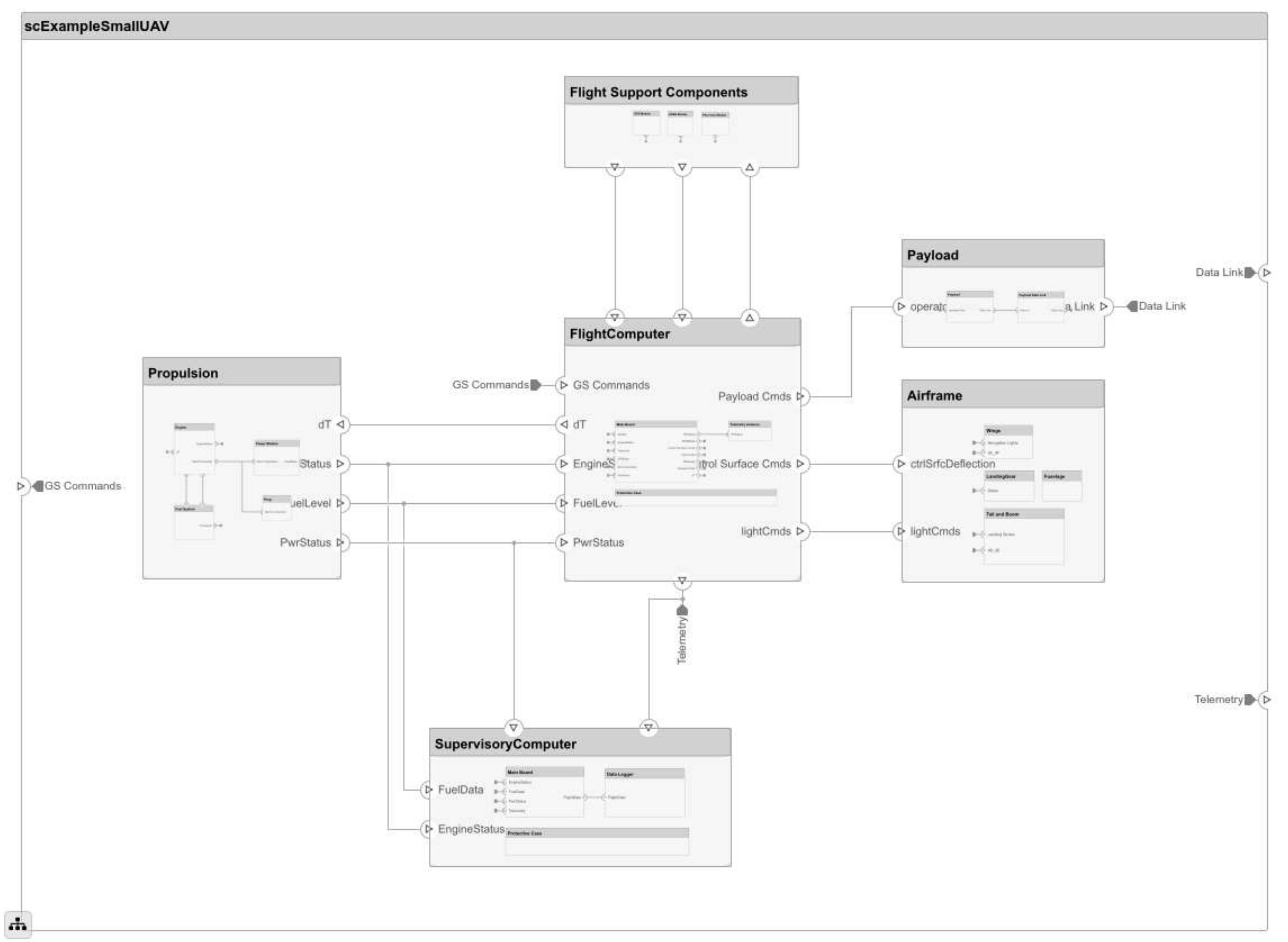
# Evaluating Architectures



MATLAB&SIMULINK

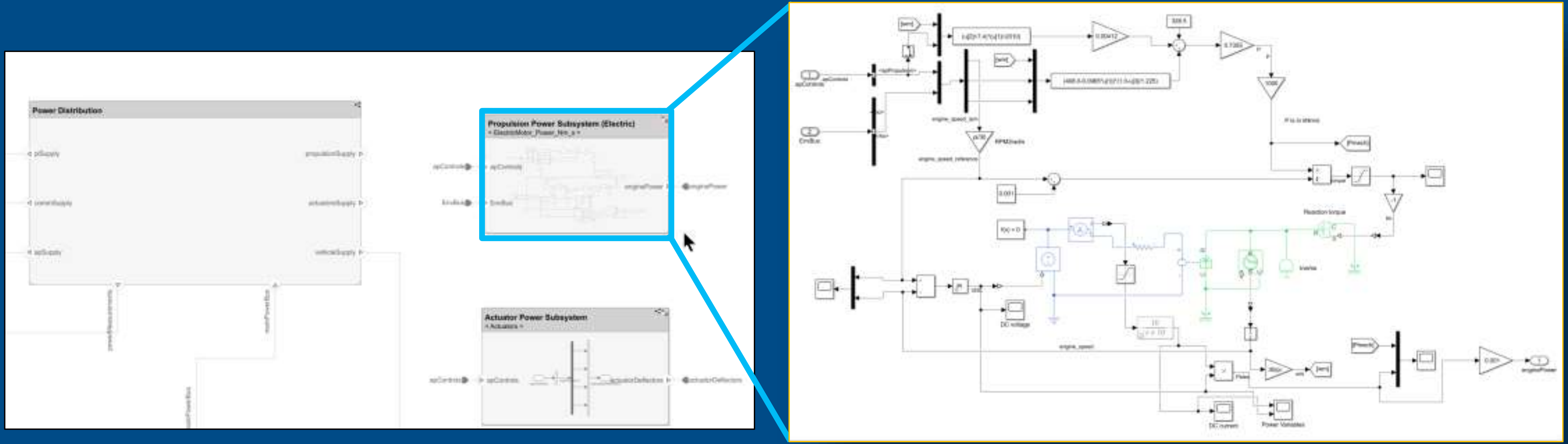
# Designing System and Software Architectures

Arch



System Composer

# Designing System and Software Architectures



System Composer

# Designing System and Software Architectures

Arch

**Find out more:**

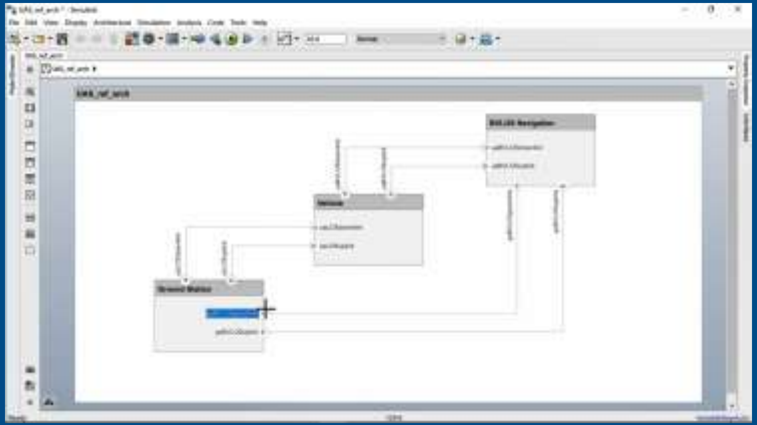
**Simulink Requirementsと新製品 System ComposerによるシステムズエンジニアリSング**

**MathWorks Japan 大越 亮二**

**F2 14:30-15:00**

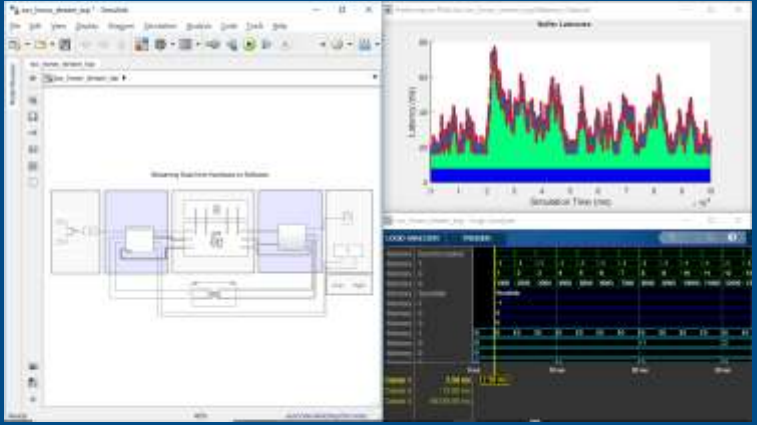
# Designing Beyond System and Software Architectures

## Systems and Software



**System Composer**

## SoC Hardware and Software

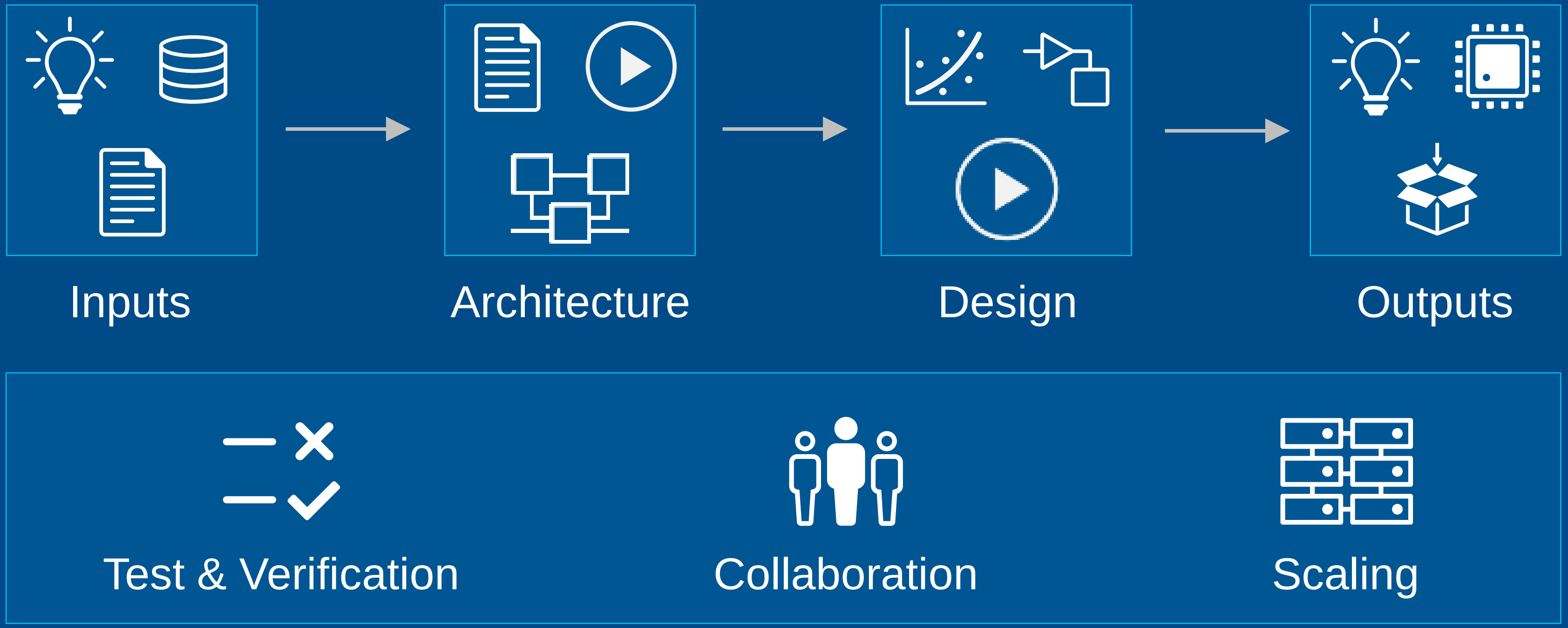


**SoC Blockset**

## AUTOSAR Software

**AUTOSAR Blockset**

# Using MATLAB & Simulink to Build Algorithms in Everything



Inputs

Architecture

Design

Outputs

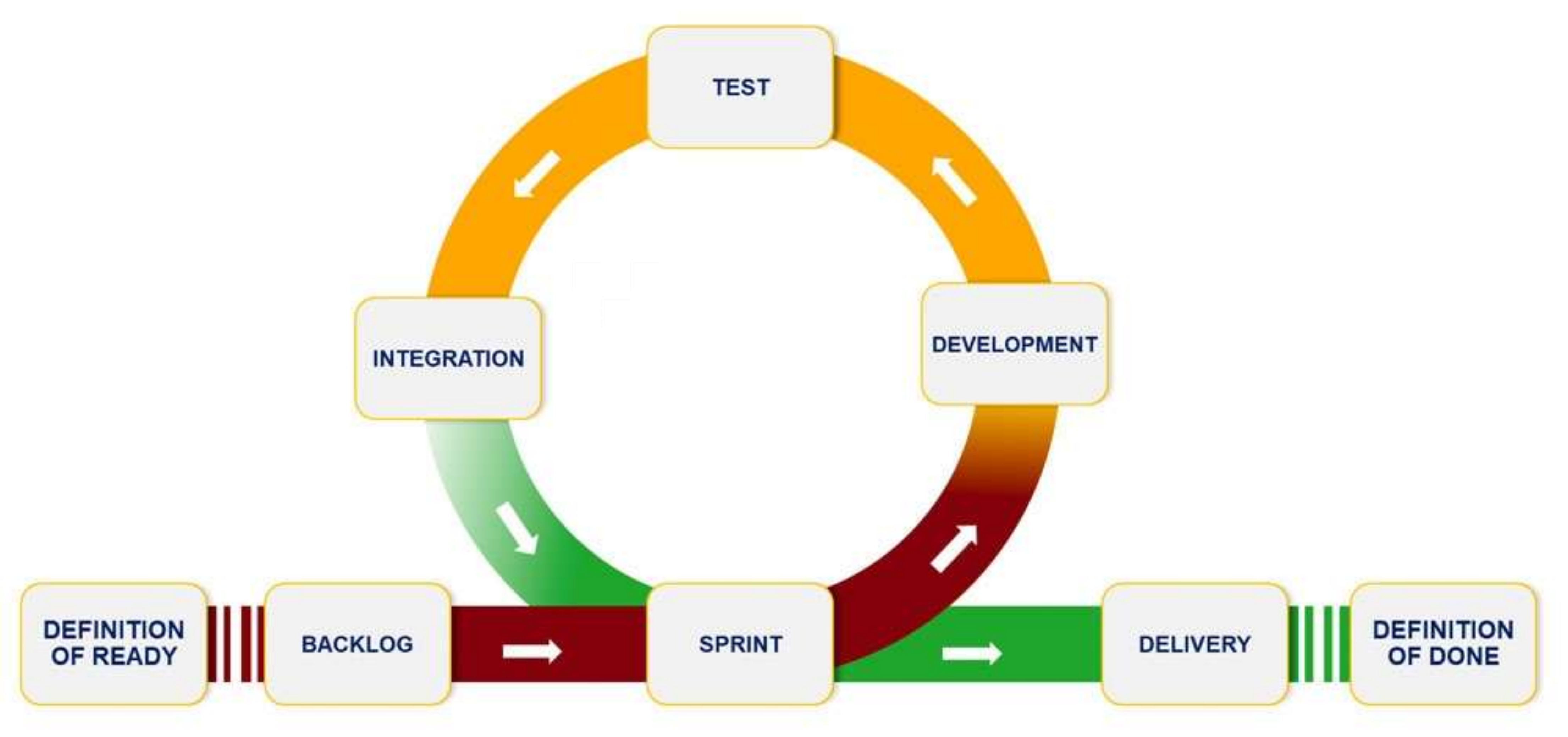
Test & Verification

Collaboration

Scaling

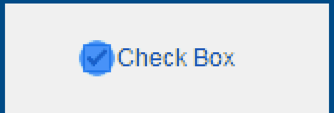
MATLAB®&SIMULINK®

# Using MATLAB & Simulink to Build Algorithms in Everything

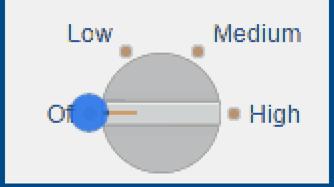


# Using the MATLAB App Testing Framework

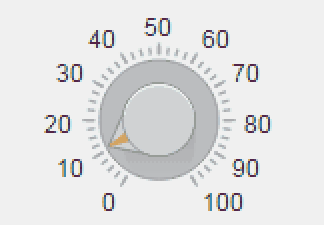
```
testCase.press(myApp.checkbox)
```



```
testCase.choose(myApp.discreteKnob, "Medium")
```



```
testCase.drag(myApp.continuousKnob, 10, 90)
```

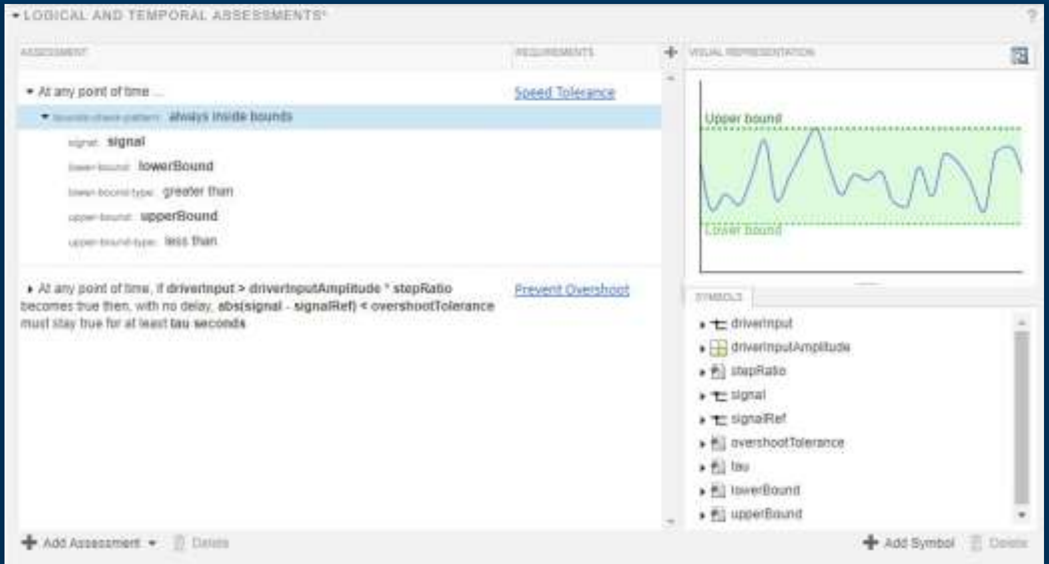


```
testCase.type(myApp.editfield, myTextVar)
```

# Authoring Logical Tests for Simulink Models

Test & Verification

## Textual Requirements Format

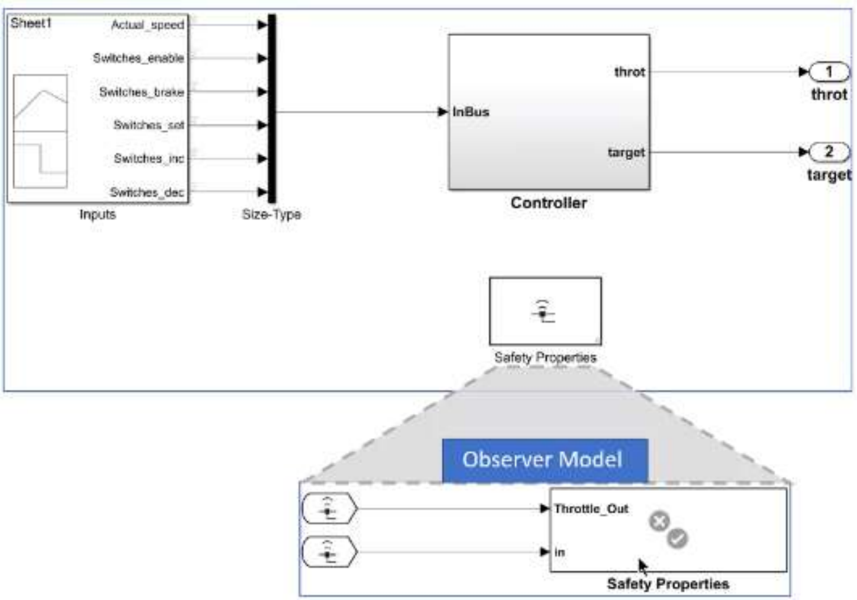


▾ LOGICAL AND TEMPORAL ASSESSMENTS

| ASSESSMENT | REQUIREMENTS |
|---|---|
| ▾ At any point of time ...<br>▾ bounds-check-pattern: always inside bounds<br>signal: signal<br>lower-bound: lowerBound<br>lower-bound-type: greater than<br>upper-bound: upperBound<br>upper-bound-type: less than | Speed Tolerance |
| ▸ At any point of time, if driverInput > driverInputAmplitude * stepRatio becomes true then, with no delay, abs(signal - signalRef) < overshootTolerance must stay true for at least tau seconds | Prevent Overshoot |

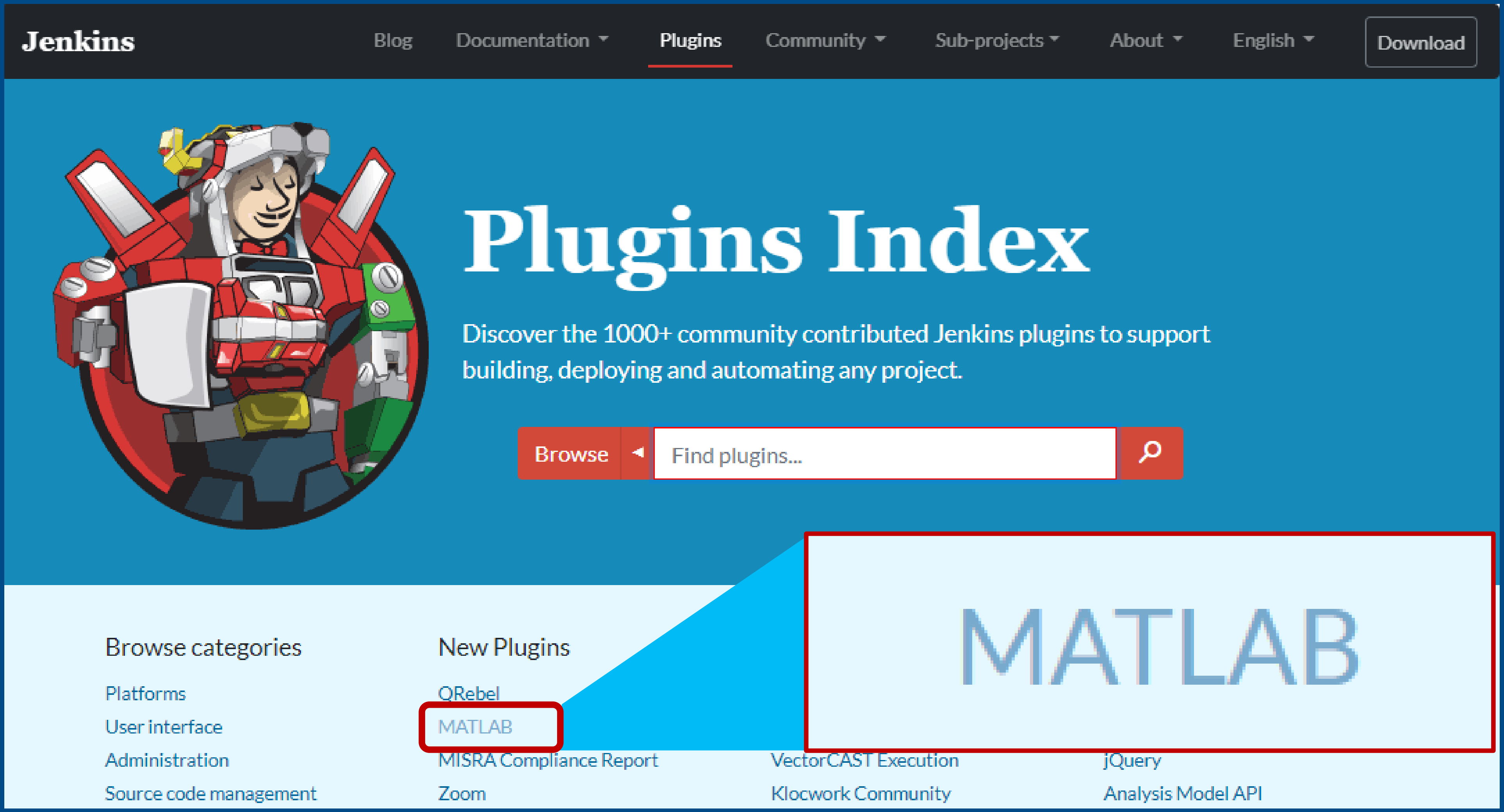
**Temporal Assessments**

## Wireless Test



**Observers**

# Using Continuous Integration



Jenkins

Blog Documentation Community Sub-projects About English Download

Plugins

# Plugins Index

Discover the 1000+ community contributed Jenkins plugins to support building, deploying and automating any project.

Browse Find plugins...

Browse categories

- Platforms
- User interface
- Administration
- Source code management

New Plugins

- QRebel
- MATLAB
- MISRA Compliance Report
- Zoom

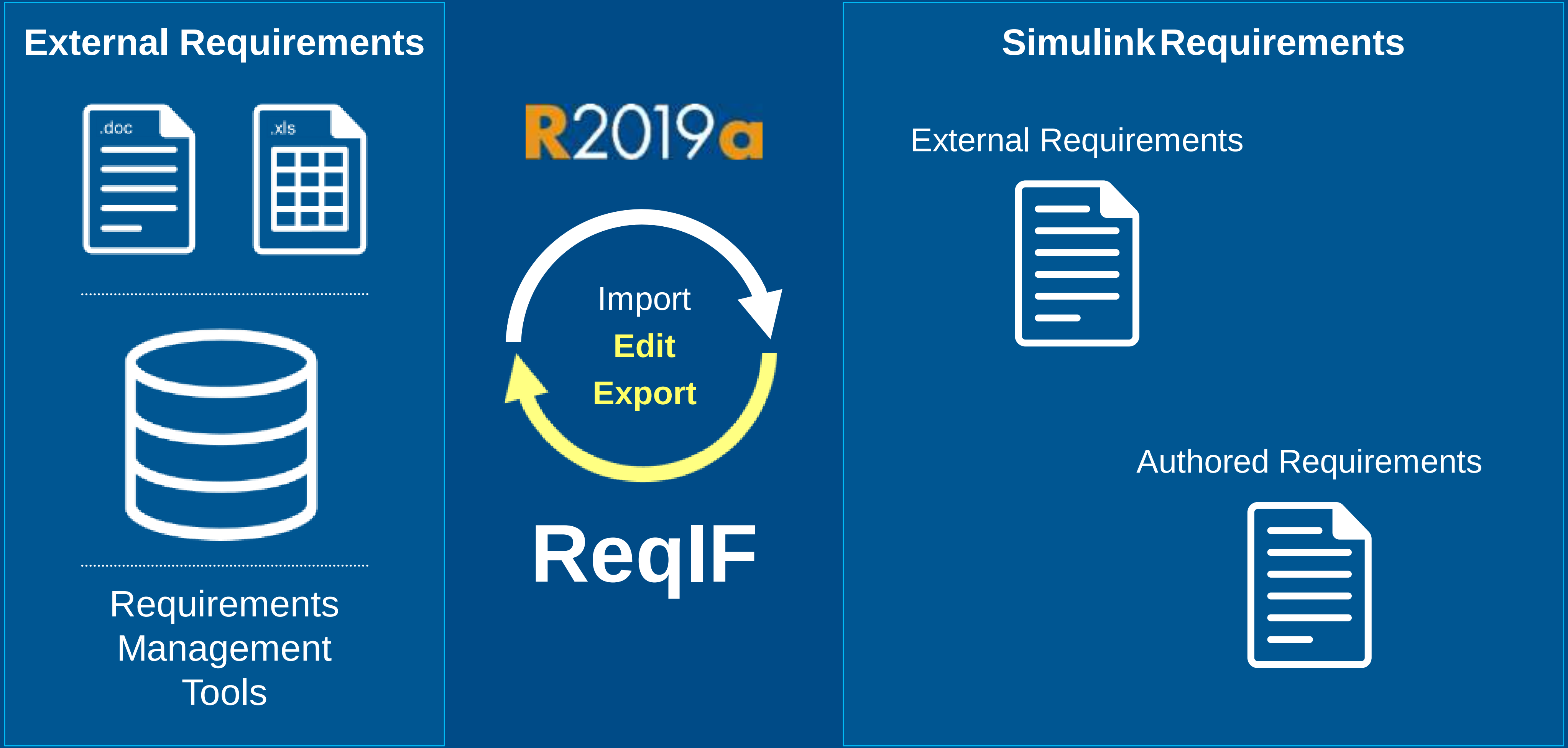
- VectorCAST Execution
- Klocwork Community

- jQuery
- Analysis Model API

MATLAB

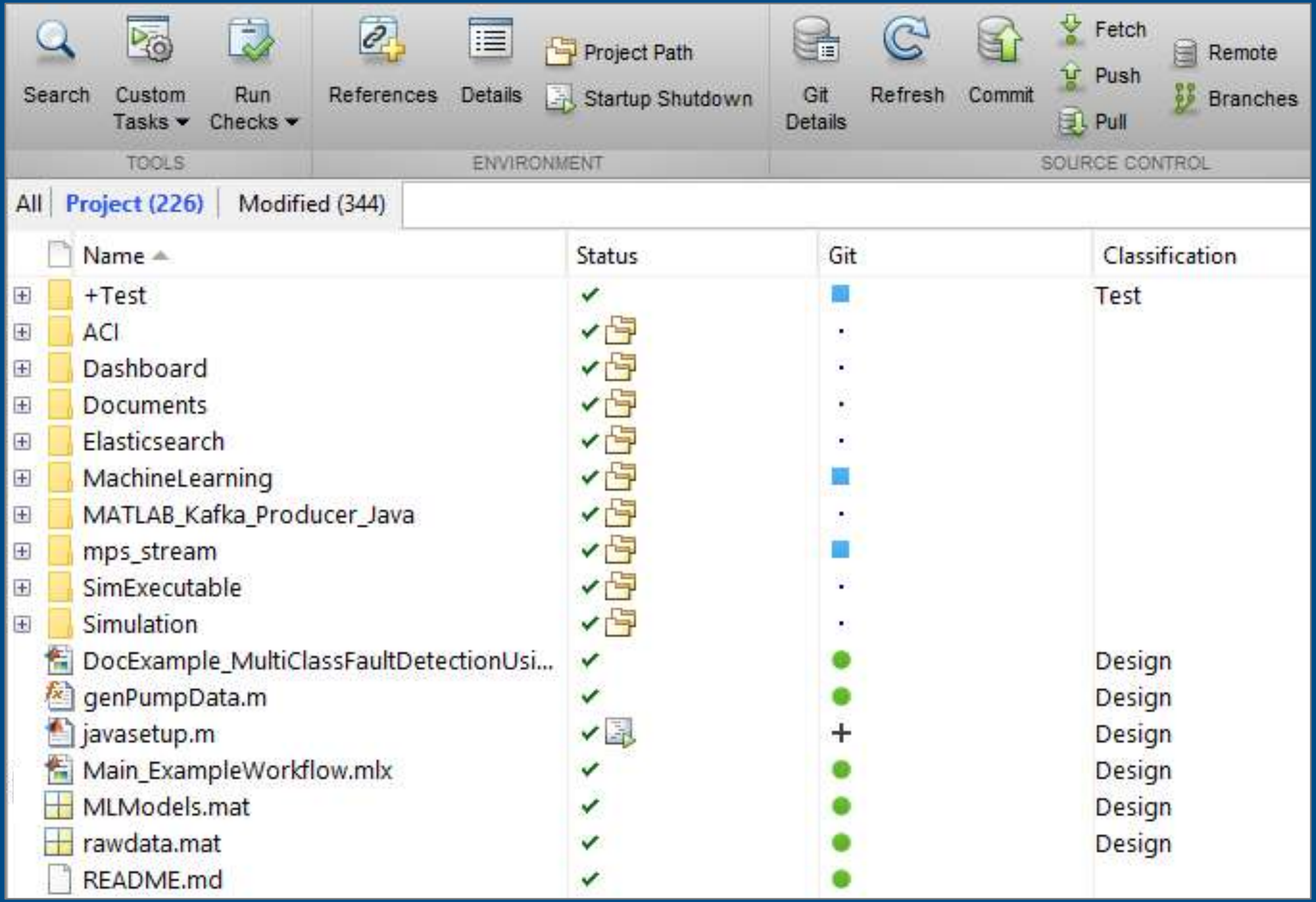
https://plugins.jenkins.io/

# Integrating with Third-party Requirements Tools



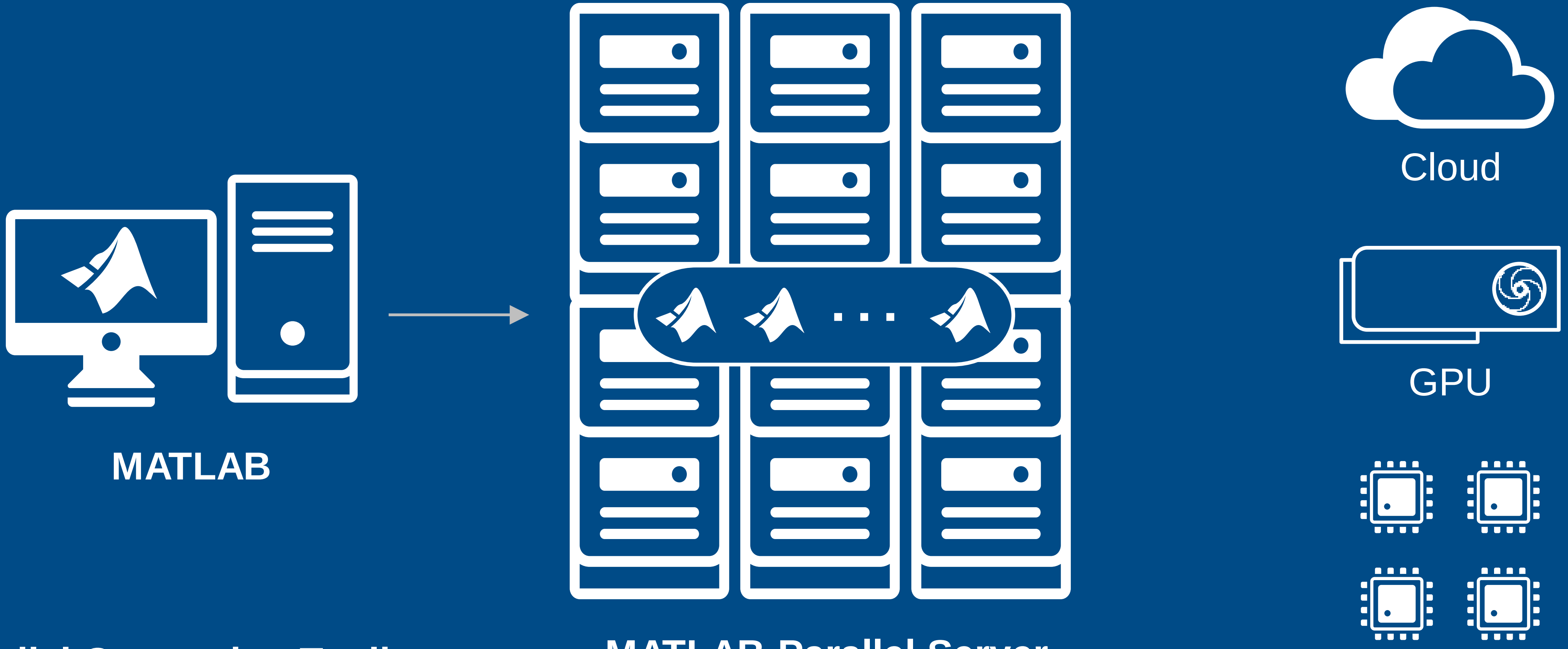
# Using Projects in MATLAB

Collaboration



| Name | Status | Git | Classification |
|---|---|---|---|
| +Test | ✓ | ■ | Test |
| ACI | ✓ | · | |
| Dashboard | ✓ | · | |
| Documents | ✓ | · | |
| Elasticsearch | ✓ | · | |
| MachineLearning | ✓ | ■ | |
| MATLAB_Kafka_Producer_Java | ✓ | · | |
| mps_stream | ✓ | ■ | |
| SimExecutable | ✓ | · | |
| Simulation | ✓ | · | |
| DocExample_MultiClassFaultDetectionUsi... | ✓ | ● | Design |
| genPumpData.m | ✓ | ● | Design |
| javasetup.m | ✓ | + | Design |
| Main_ExampleWorkflow.mlx | ✓ | ● | Design |
| MLModels.mat | ✓ | ● | Design |
| rawdata.mat | ✓ | ● | Design |
| README.md | ✓ | ● | |

# Scaling Computations on Clusters and Clouds

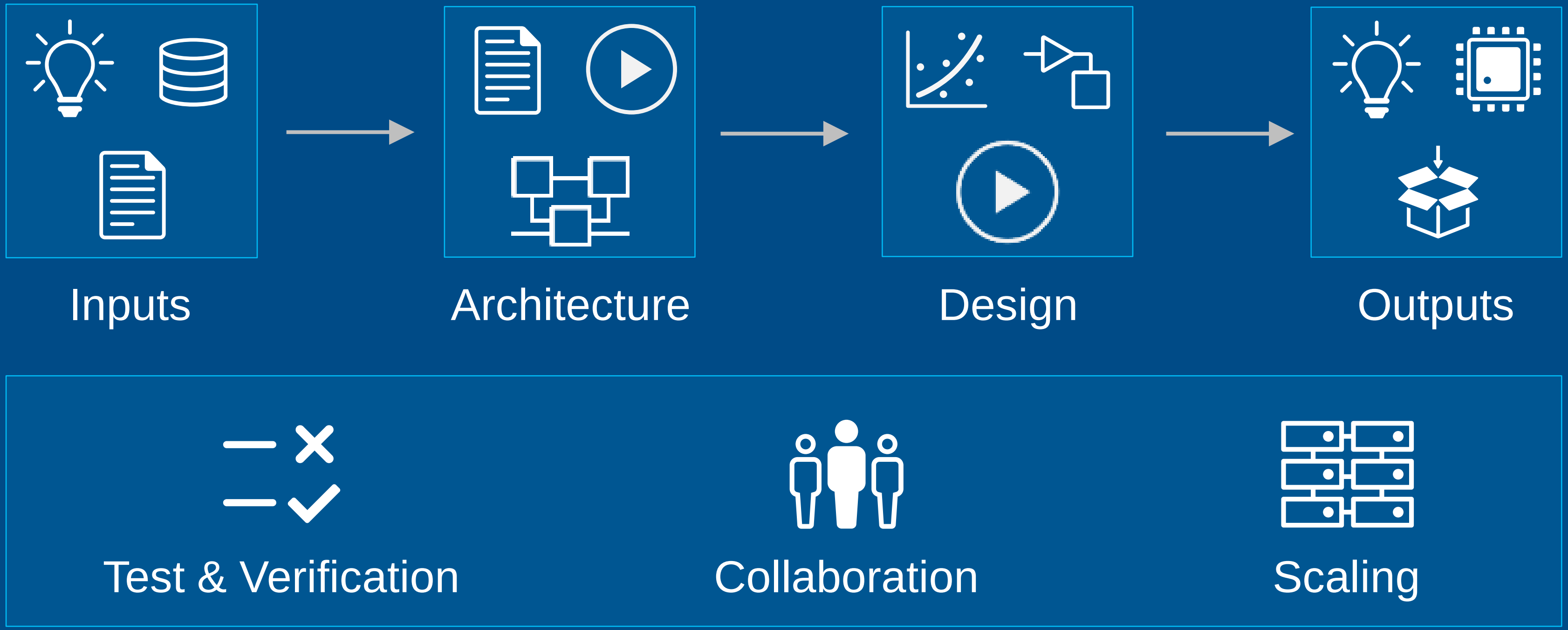


Parallel Computing Toolbox

MATLAB Parallel Server
(MATLAB Distributed
Computing Server)

MATLAB Parallel Server

# Using MATLAB & Simulink to Build Algorithms in Everything



Inputs → Architecture → Design → Outputs

Test & Verification

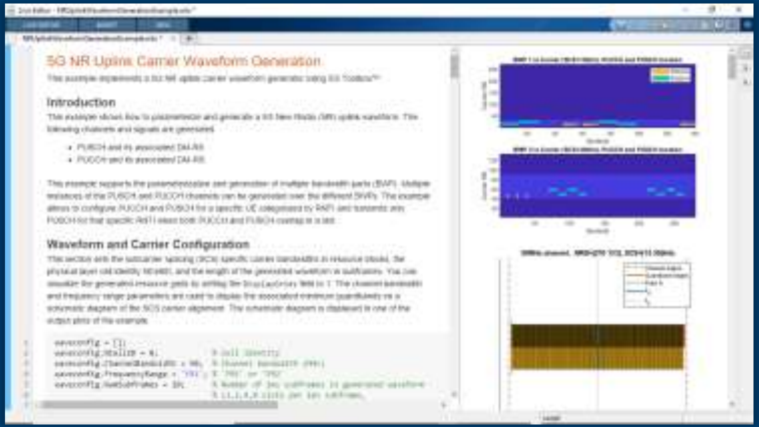
Collaboration

Scaling

MATLAB®&SIMULINK®

# Specialized Tools for Building Algorithms in Everything

## Communications



5G NR Uplink Carrier Waveform Generation

[illegible]

Introduction

[illegible]

- PUSCH and its associated DM-RS
- PUCCH and its associated DM-RS

[illegible]

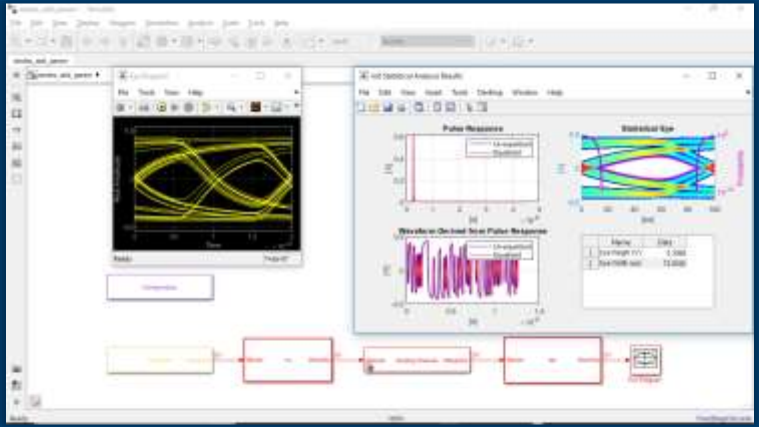
Waveform and Carrier Configuration

[illegible]

```
waveconfig = [];
waveconfig.NCellID = 0;               % Cell identity
waveconfig.ChannelBandwidth = 50;     % Channel bandwidth (MHz)
waveconfig.FrequencyRange = 'FR1';    % 'FR1' or 'FR2'
waveconfig.NumSubframes = 10;         % Number of 1ms subframes in generated waveform
                                      % [illegible]
```

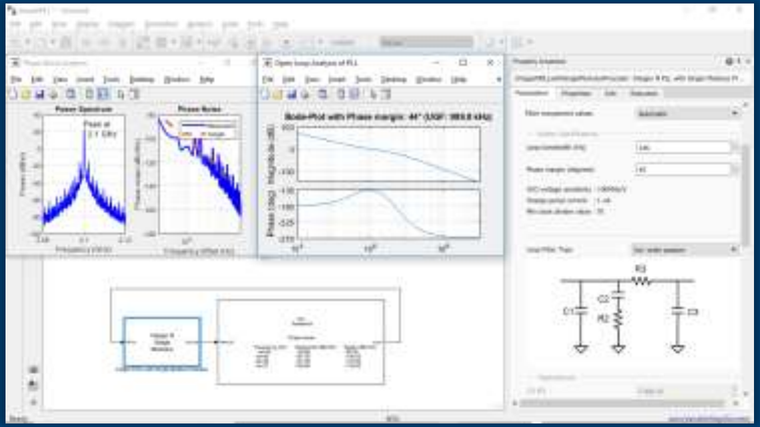
**5G**
**Toolbox**

## Physical interconnects



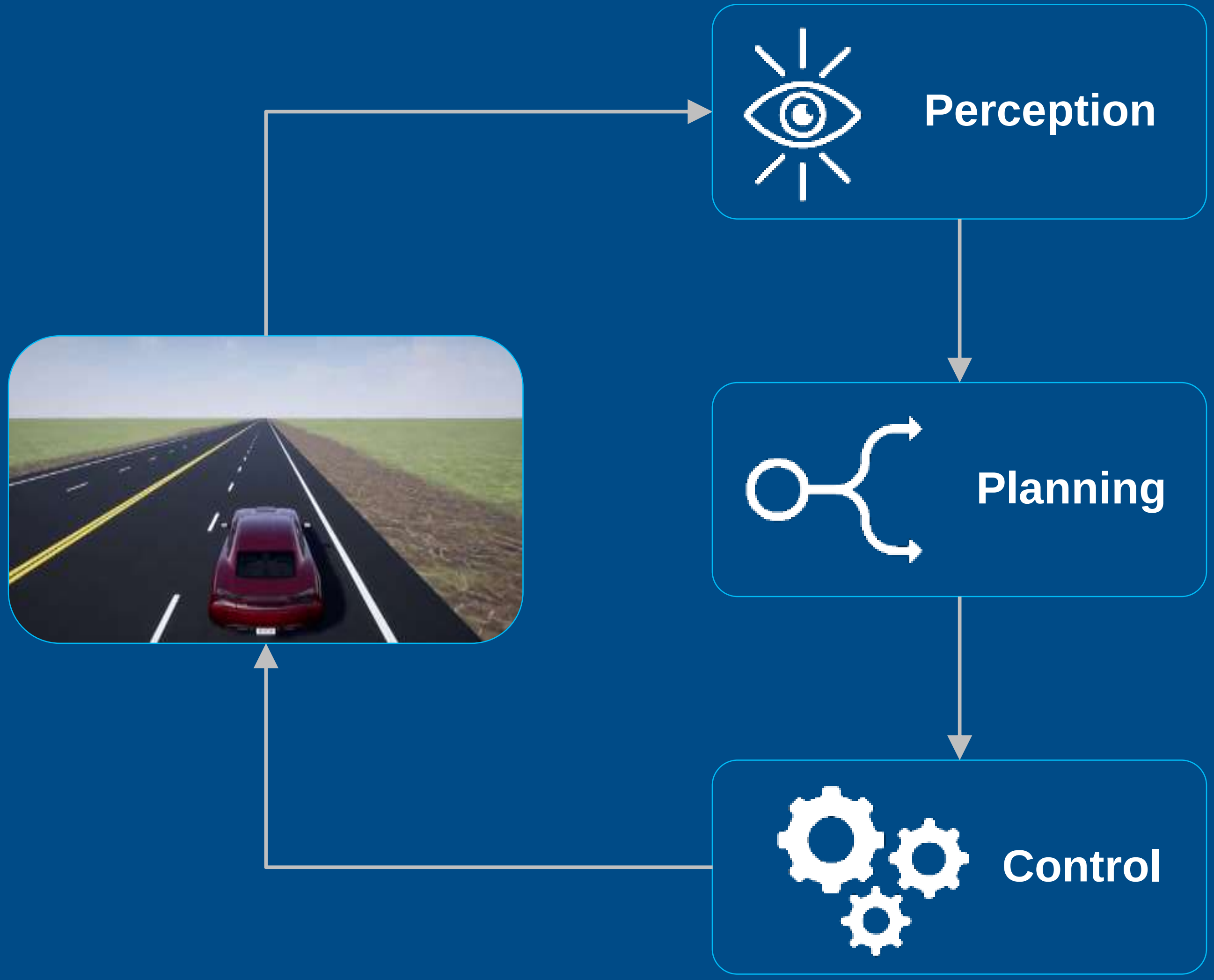
**SerDes**
**Toolbox**

## Analog Mixed-Signal

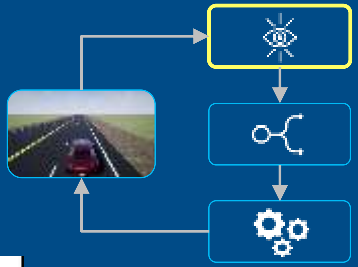


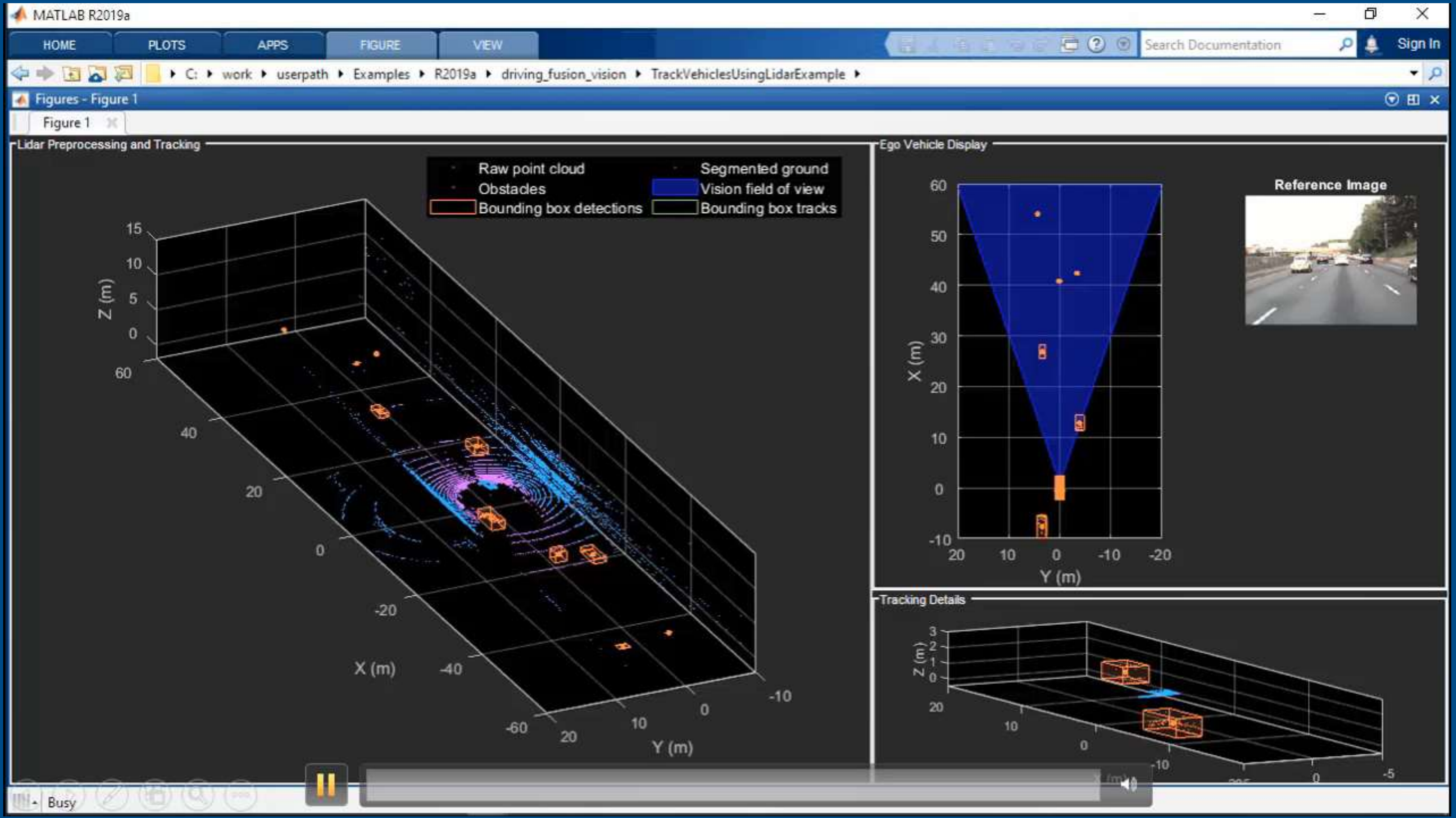
**Mixed-Signal**
**Blockset**

# Developing Autonomous Systems



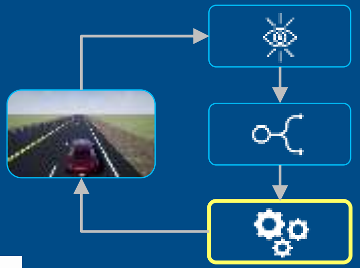
# Evaluate Sensor Fusion Architectures

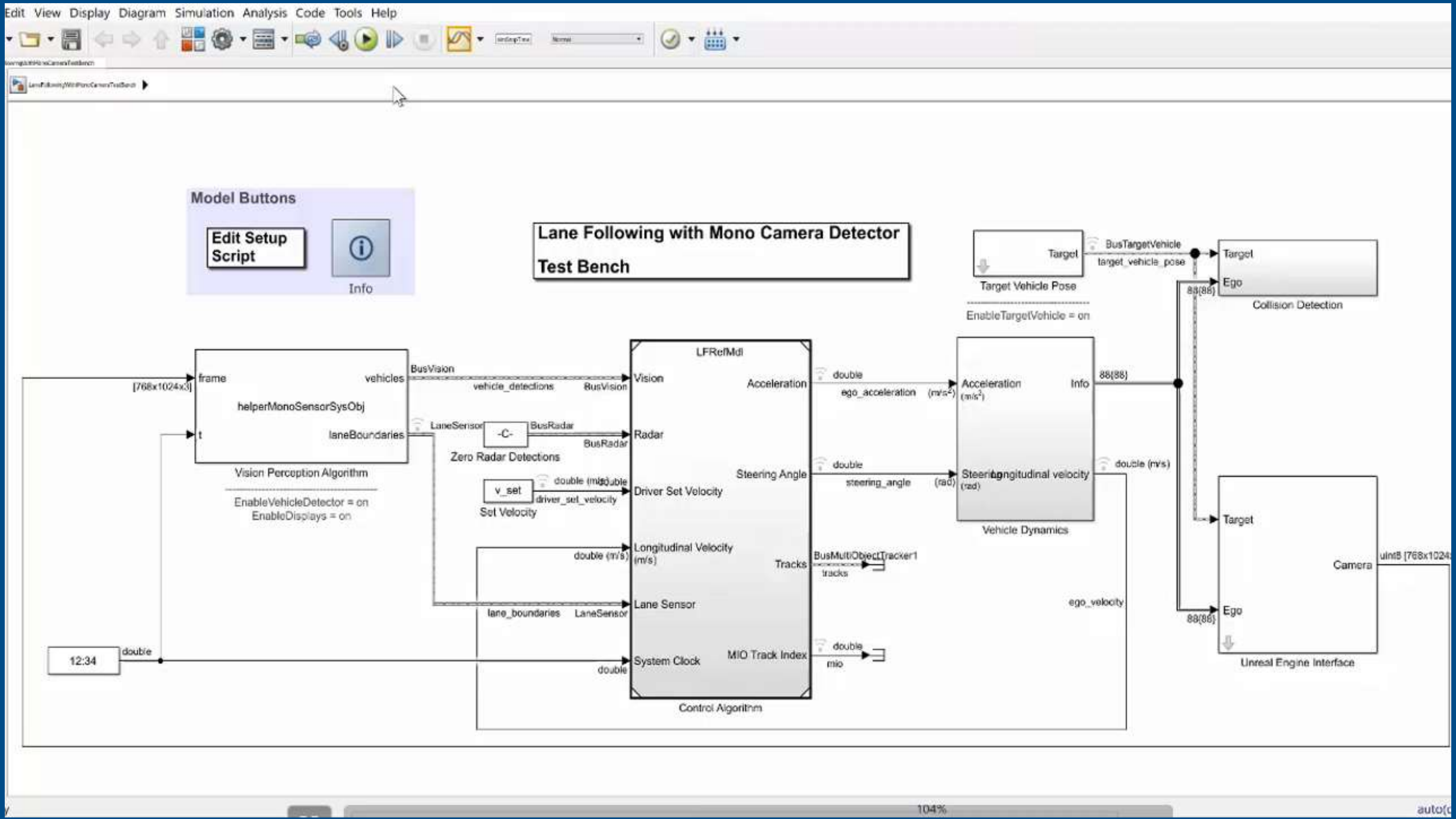




Sensor Fusion and Tracking Toolbox

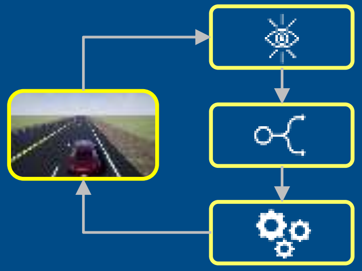
# Design Lane-following and Spacing Control Algorithms





Model Predictive Control Toolbox

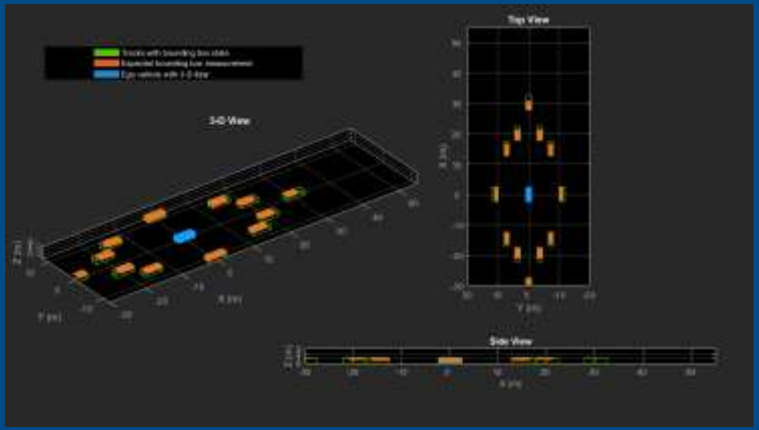
# Developing Autonomous Systems

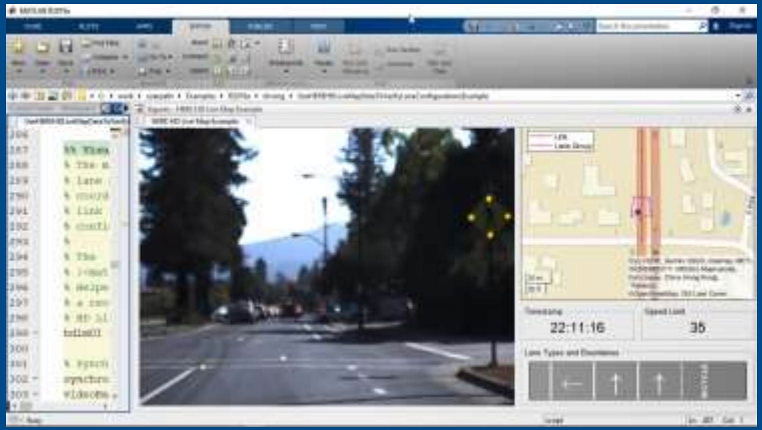


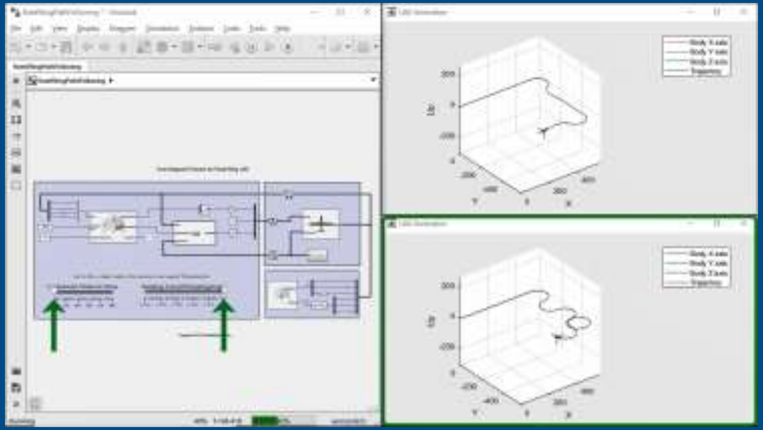
Lidar Processing & Tracking

HERE HD Maps & OpenDRIVE Roads

UAV Algorithms





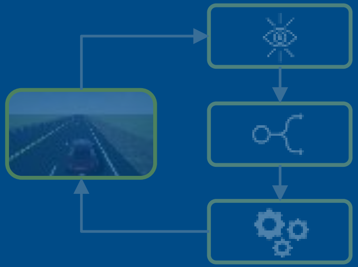


**Computer Vision Toolbox**

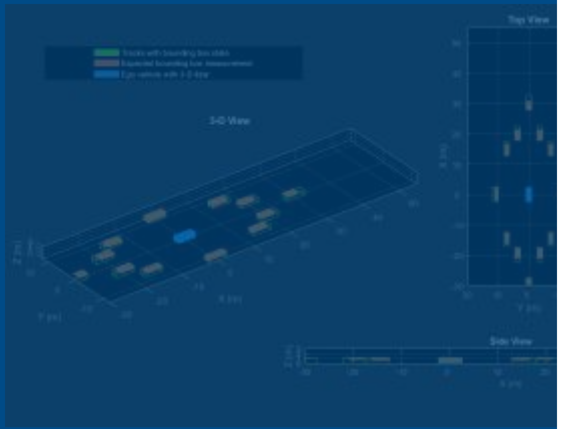
**Automated Driving Toolbox**

**Robotics System Toolbox**

# Developing Autonomous Systems



Lidar Processing & Tracking

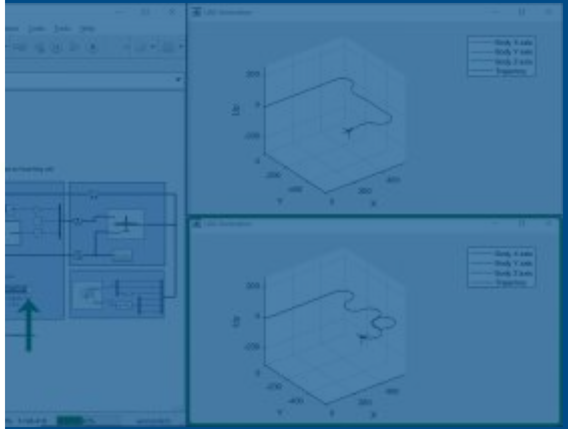


Computer Vision Toolbox

HERE HD Maps &

Automated Driving Toolbox

UAV Algorithms



Robotics System Toolbox

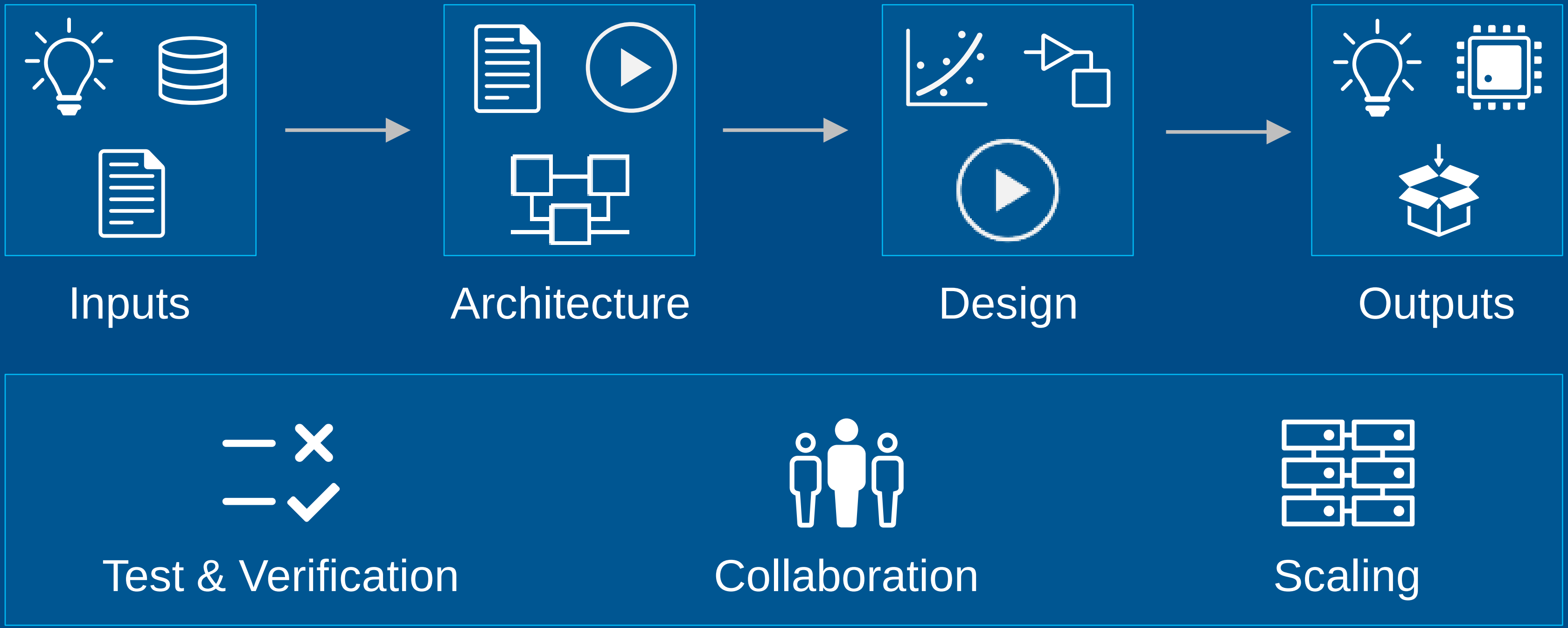
## Find out more:

## ADAS・自動運転アルゴリズム検証のためのシナリオ生成とシミュレーション

MathWorks Japan 大塚 慶太郎

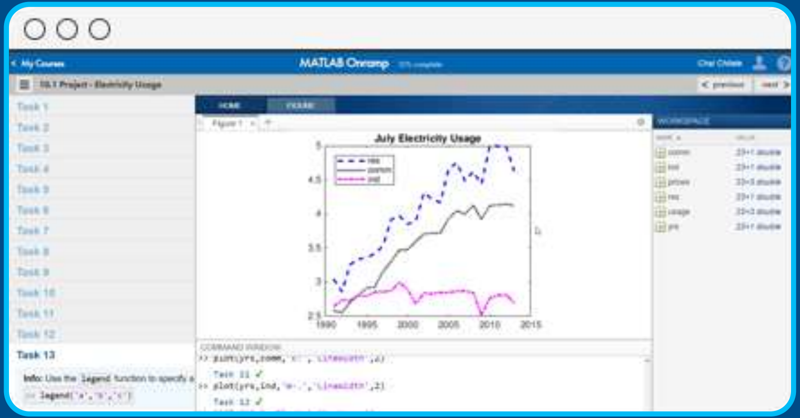
D4 16:30-17:00

# Using MATLAB & Simulink to Build Algorithms in Everything



MATLAB®&SIMULINK®

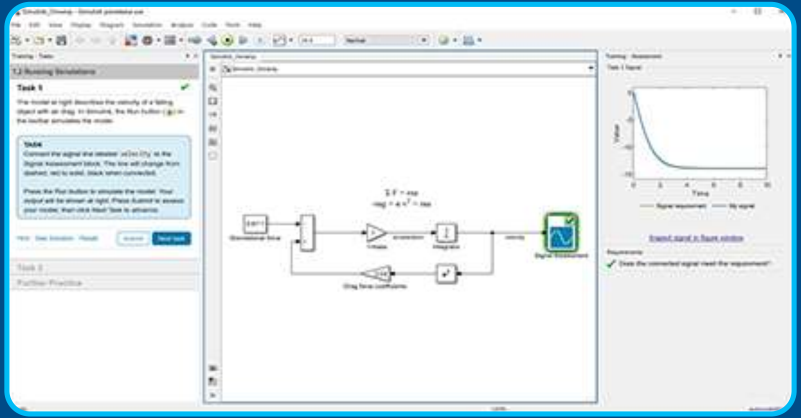
# Get Started

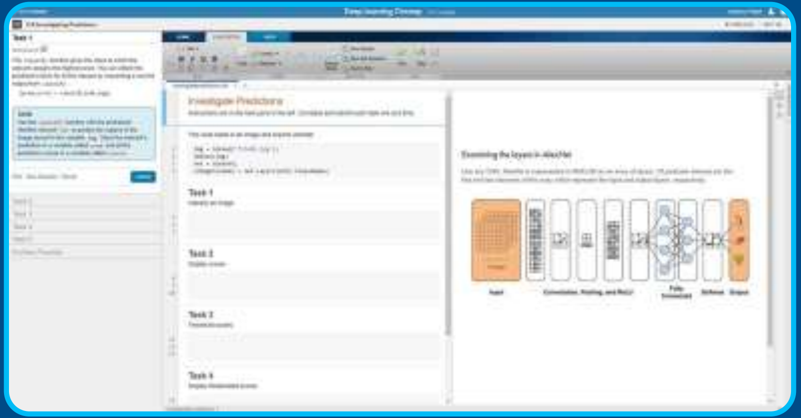


Info: Use the legend function to specify a

```
>> legend('a','b','c')
```

```
>> plot(yrs,comm,'k-','LineWidth',2)
  Task 11 ✓
>> plot(yrs,ind,'m-.','LineWidth',2)
  Task 12 ✓
```





## MATLAB Onramp

Quickly learn the essentials of MATLAB.

## Simulink Onramp

Learn to create, edit, and troubleshoot Simulink models.

## Deep Learning Onramp

Learn to use deep learning techniques in MATLAB for image recognition.

# MATLAB EXPO 2019