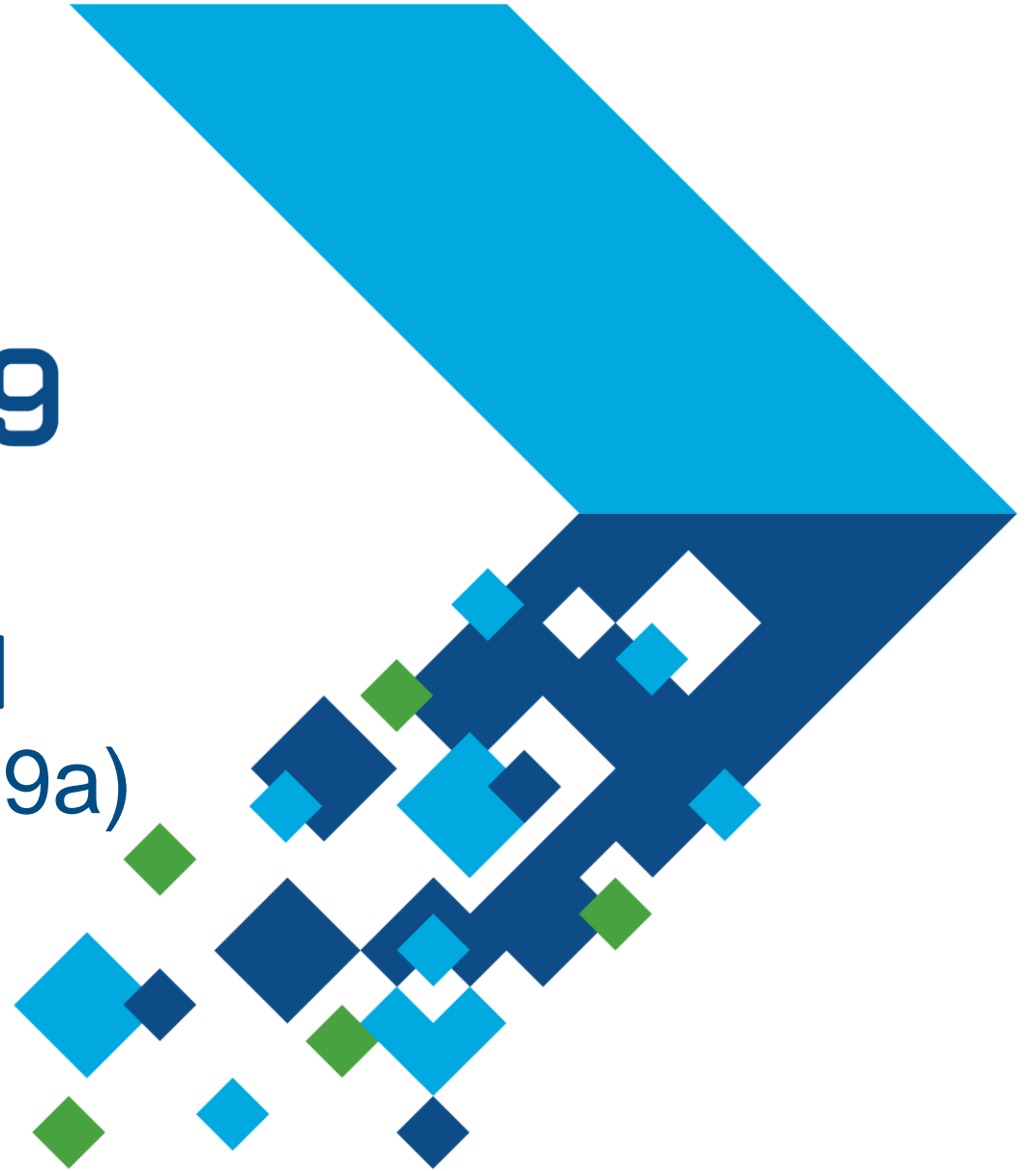
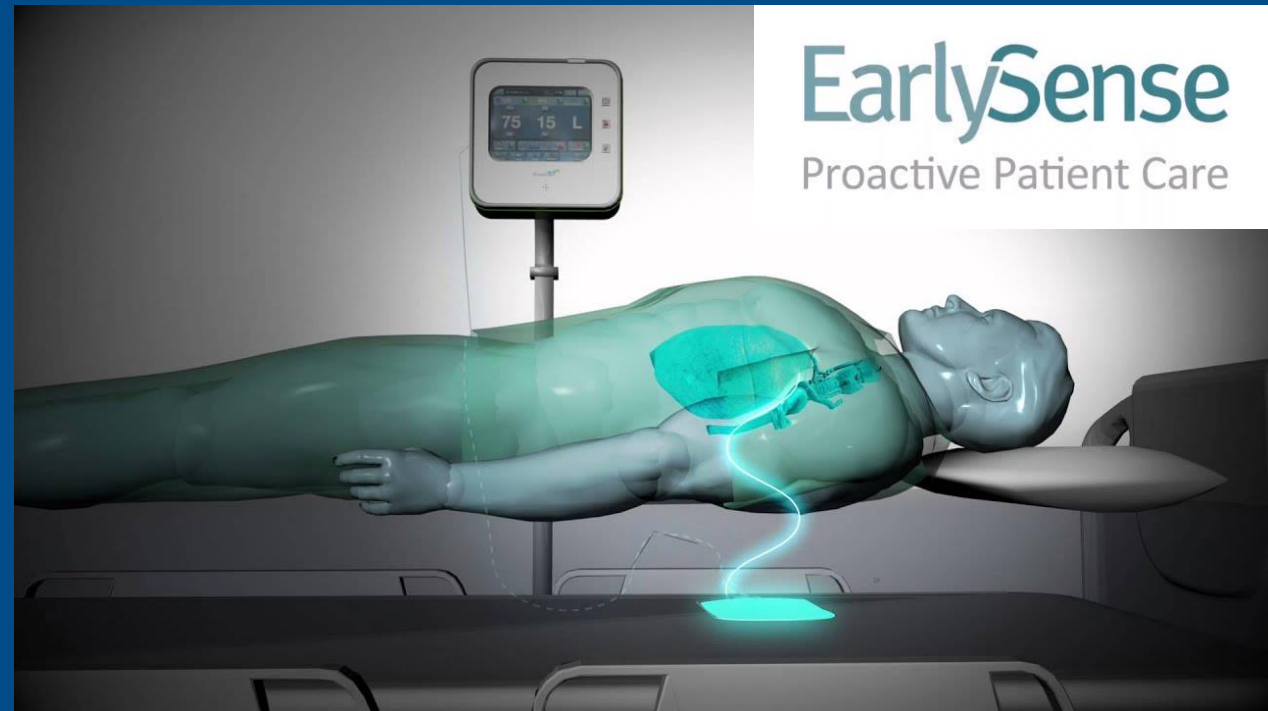


MATLAB EXPO 2019

MATLAB과 Simulink 제품군의
새로운 기능들 (R2018b/R2019a)

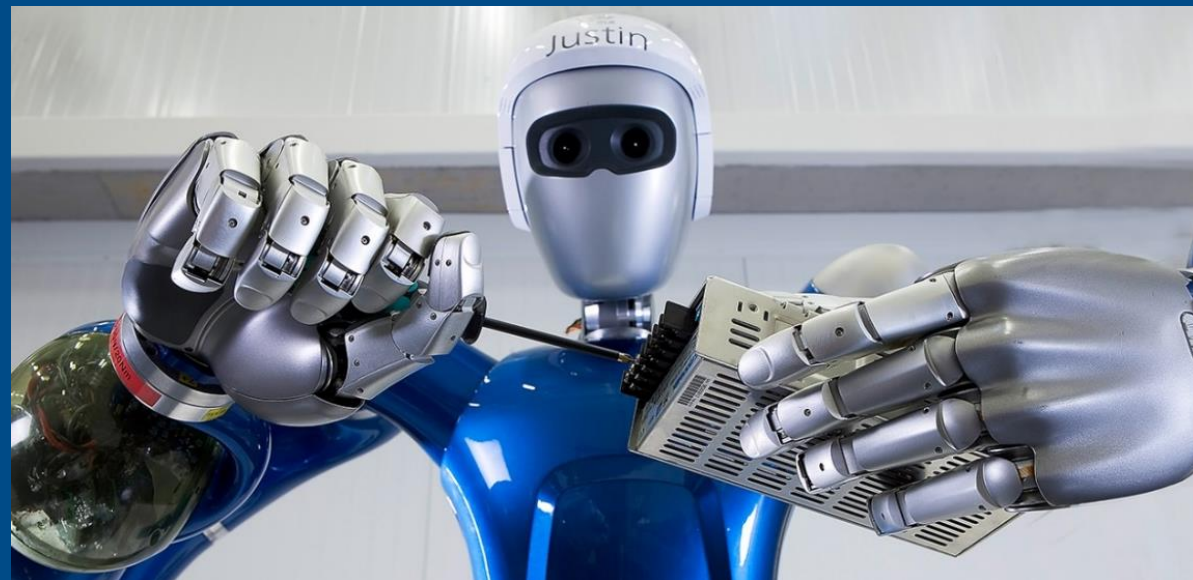
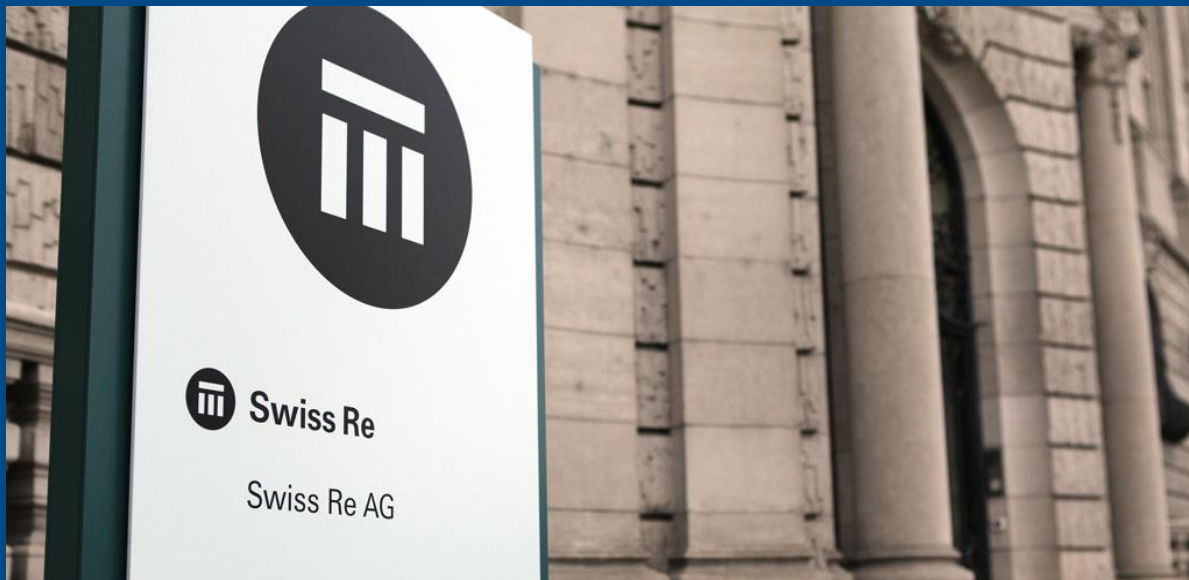
이영준





EarlySense
Proactive Patient Care

Algorithms in Everything



Using MATLAB & Simulink to Build Algorithms in Everything

Simplifying your work...

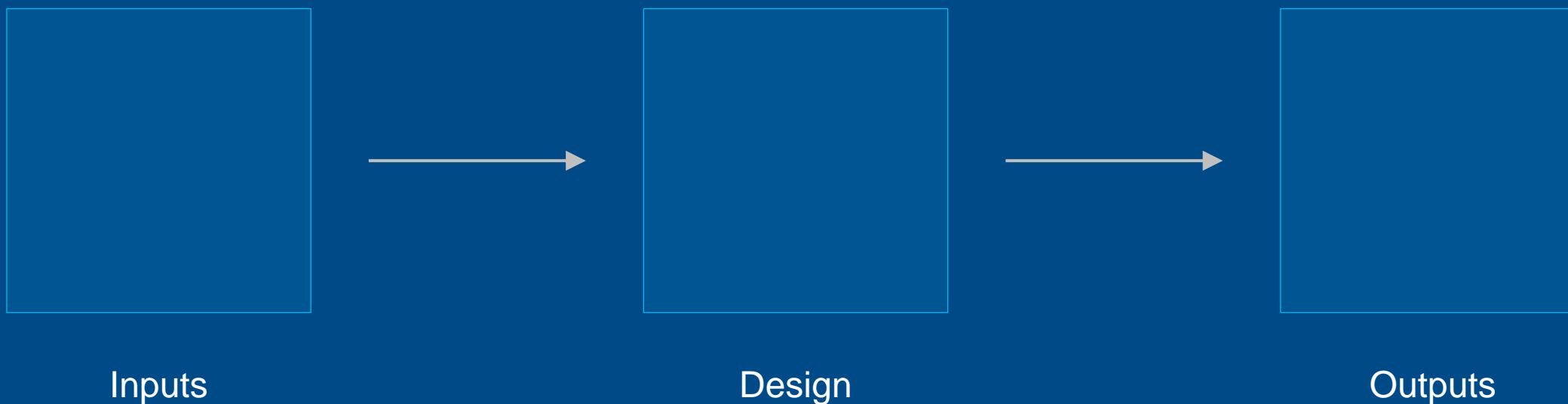
...often at higher levels of abstraction.



MATLAB® & SIMULINK®



Using MATLAB & Simulink to Build Algorithms in Everything



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

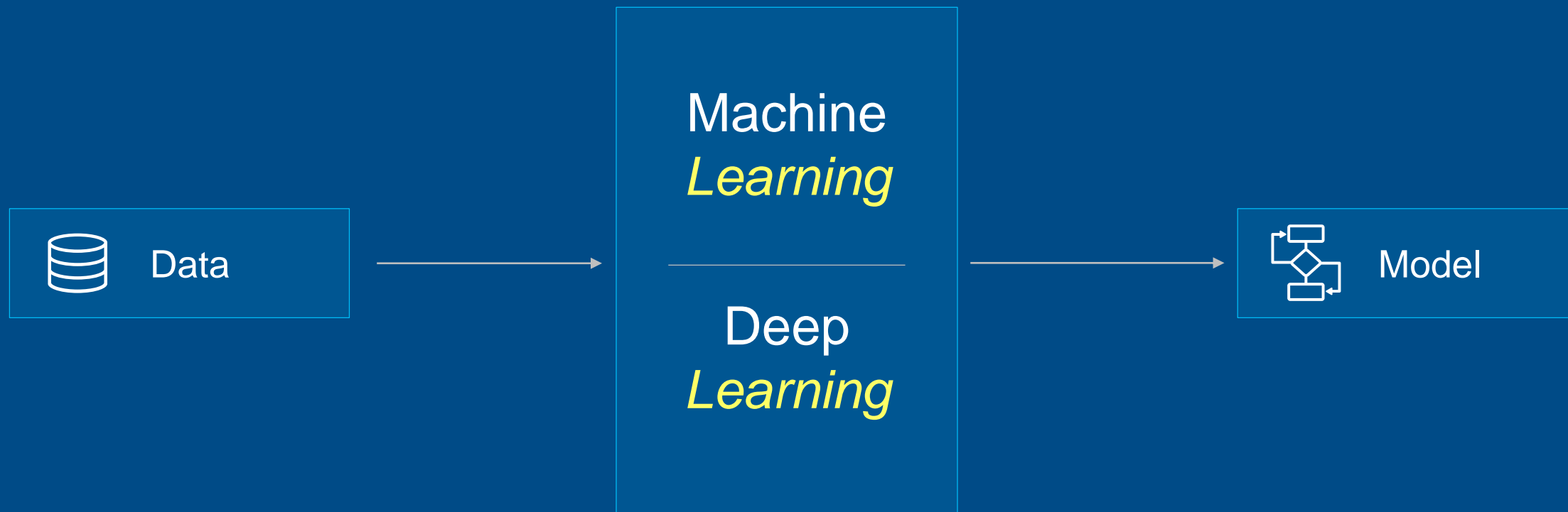


There are two ways to get a computer to do what you want

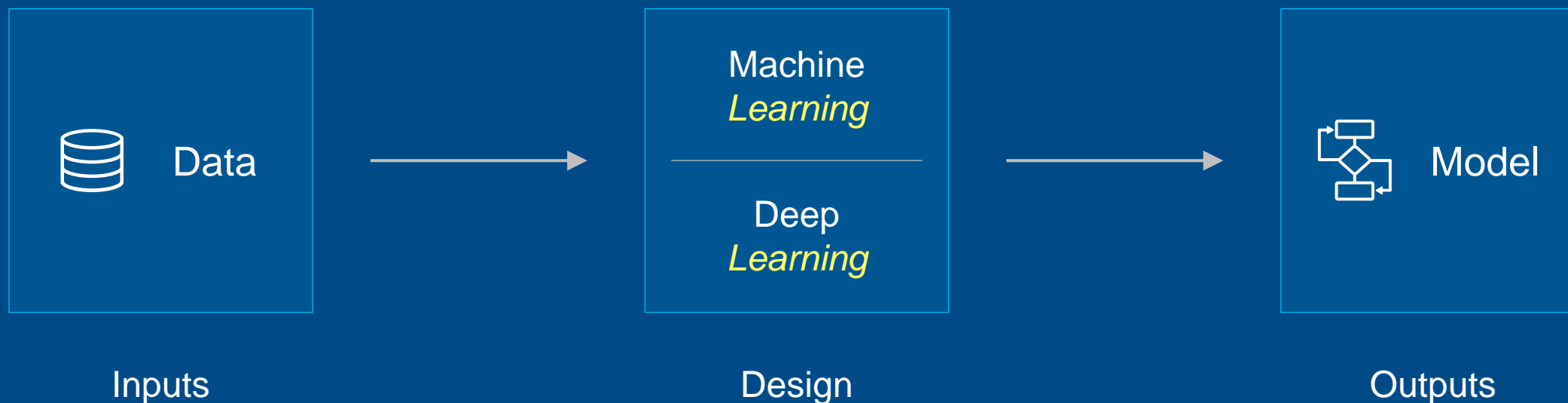
Machine Learning



Artificial Intelligence



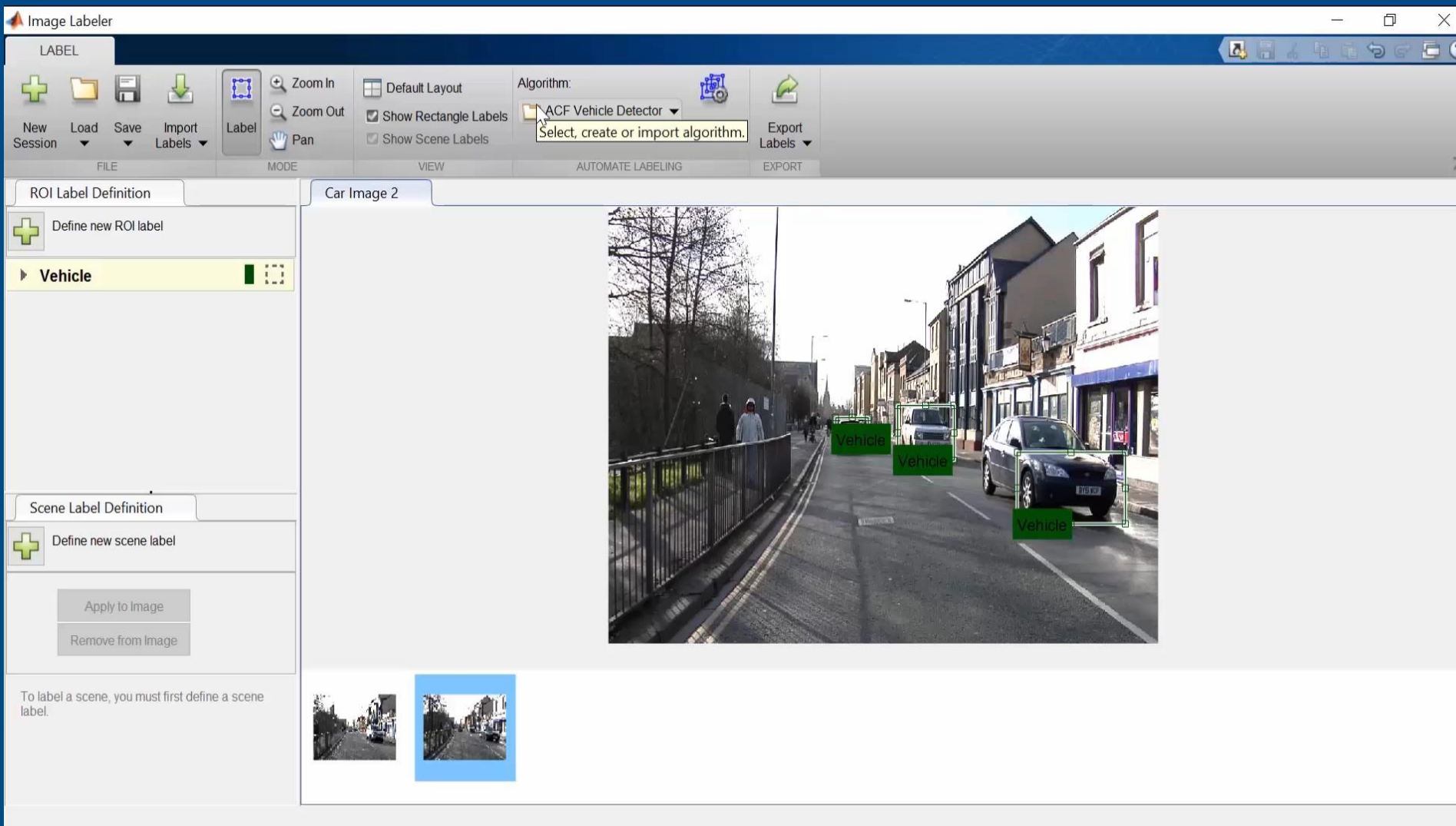
Using MATLAB and Simulink to Build **Deep Learning Models**



MATLAB® & SIMULINK®

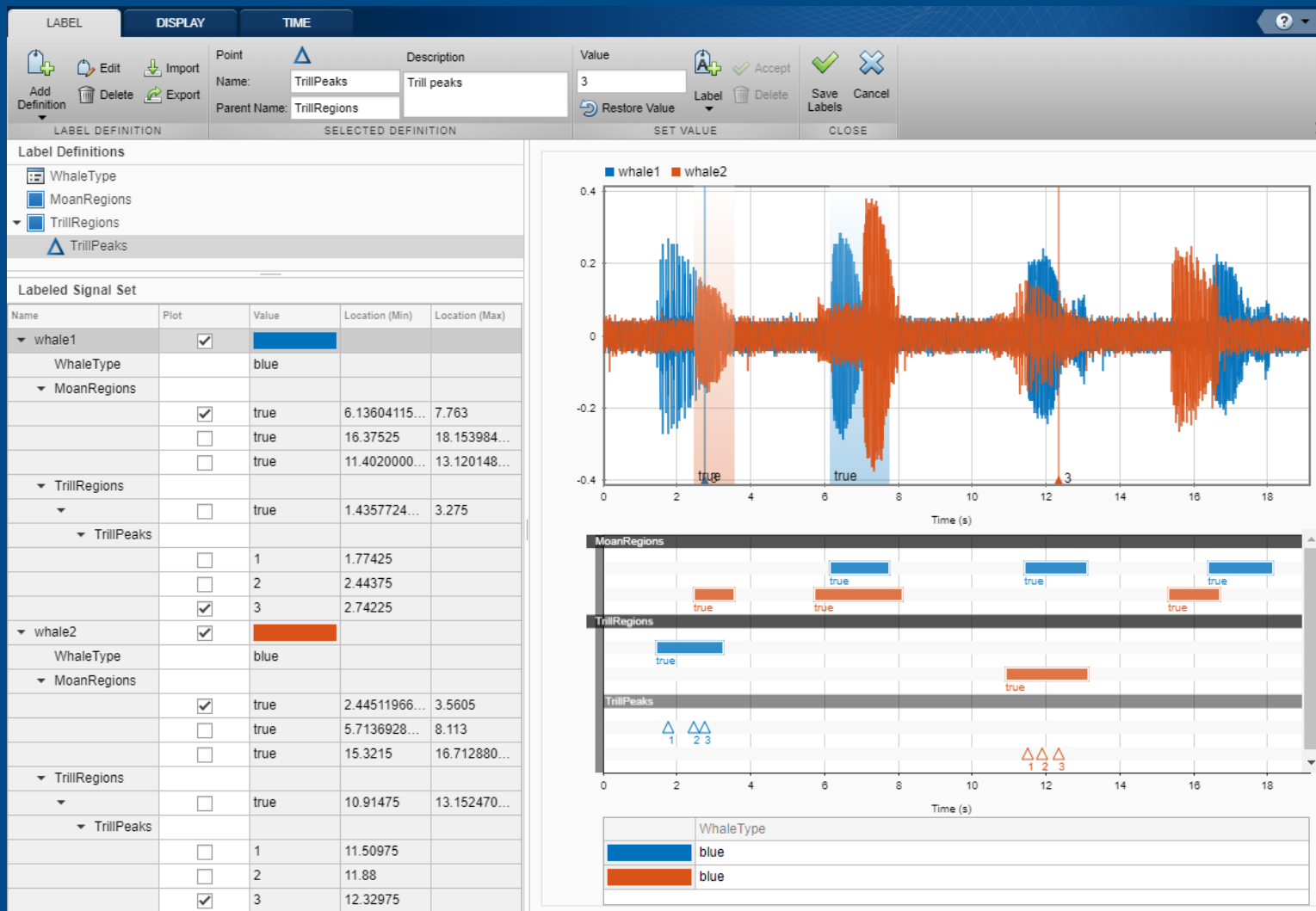


Using Apps for Ground Truth Labeling Image and Video Data



Using Apps for Ground Truth Labeling

Signal Data



Using Apps for Ground Truth Labeling

Audio Data



Audio Labeler - Rainbow-16-8-mono-114secs.wav

LABEL RECORD

Audio Player: Primary Soun...
Settings Legend Export

FILE DEVICE VIEW EXPORT

Data Browser Rainbow-16-8-mono-114secs.wav

▼ 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

File Labels

Label Name	Value
Content	speech

ROI Labels

Label Name	Value
SpeechActivity	true true true
VUV	

Ready Samples Underrun = 0

Using Apps for Designing Deep Learning Networks

The screenshot displays the Deep Network Designer application window. The interface is divided into several sections:

- Toolbar:** Contains icons for New, Import, Duplicate, Cut, Copy, Paste, Fit to View, Zoom In, Zoom Out, Auto Arrange, Analyze, and Export.
- LAYERS Panel:** A sidebar on the left with a search filter and categories: INPUT (ImageInputLayer, SequenceInputLayer), LEARNABLE (Convolution2DLayer, TransposedConvolution2DLayer, FullyConnectedLayer, LSTMLayer, BiLSTMLayer), ACTIVATION (ReLULayer, LeakyReLULayer, ClippedReLULayer), and NORMALIZATION AND DROPOUT.
- Canvas:** The central workspace showing a vertical flow of layers: imageinput (ImageInputLayer), conv (Convolution2D...), relu (ReLULayer), maxpool (MaxPooling2D), fc (FullyConnected), softmax (SoftmaxLayer), and classoutput (ClassificationO...).
- PROPERTIES Panel:** A sidebar on the right showing network statistics: Number of layers (7), Number of connections (6), Input type (Image), and Output type (Classification).

Using Transfer Learning with Pre-trained Models



Inception-v3

ResNet-101

VGG-16

Inception-ResNet-
v2

ResNet-18

GoogLeNet

DenseNet-201

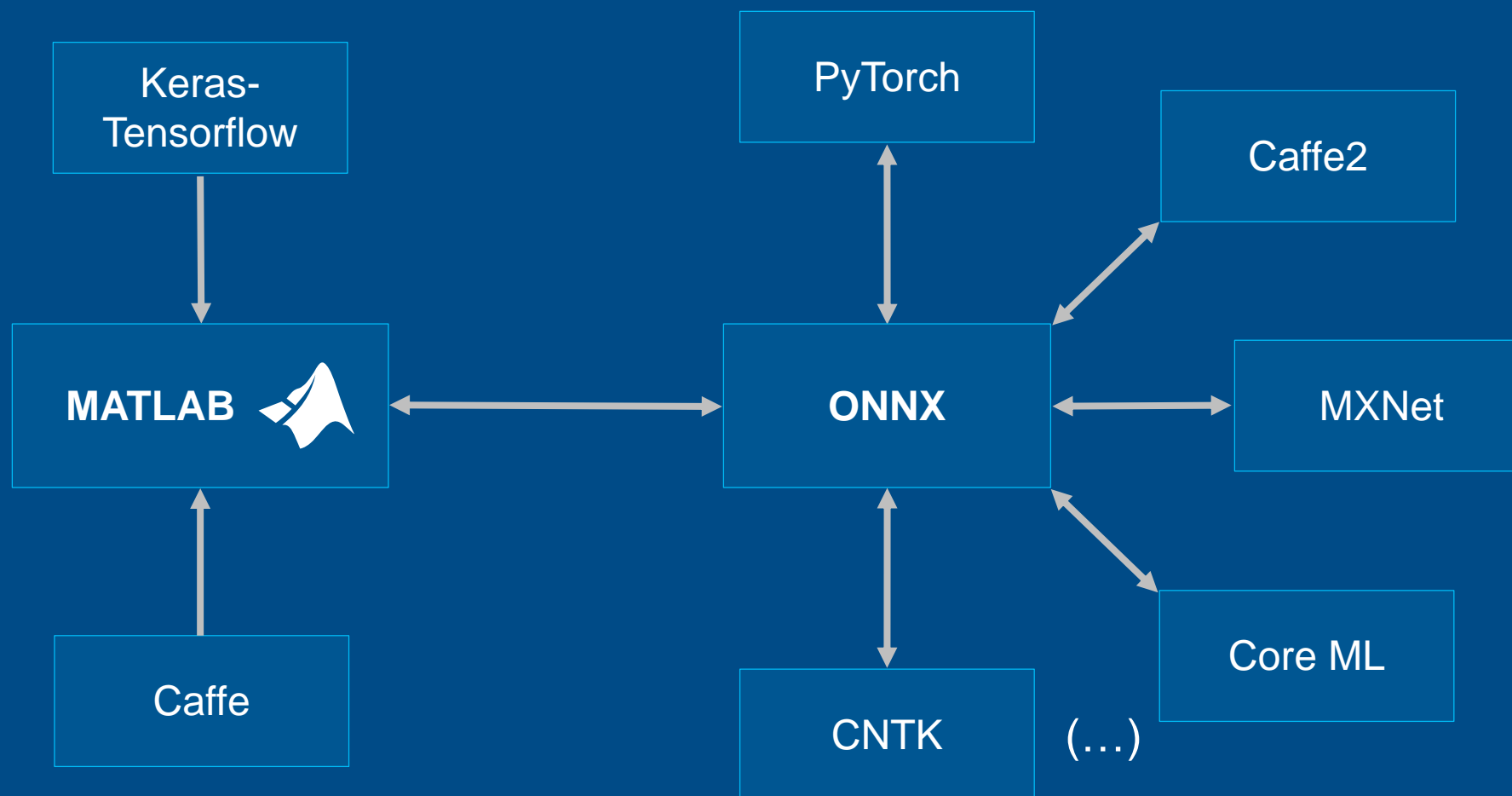
VGG-19

SqueezeNet

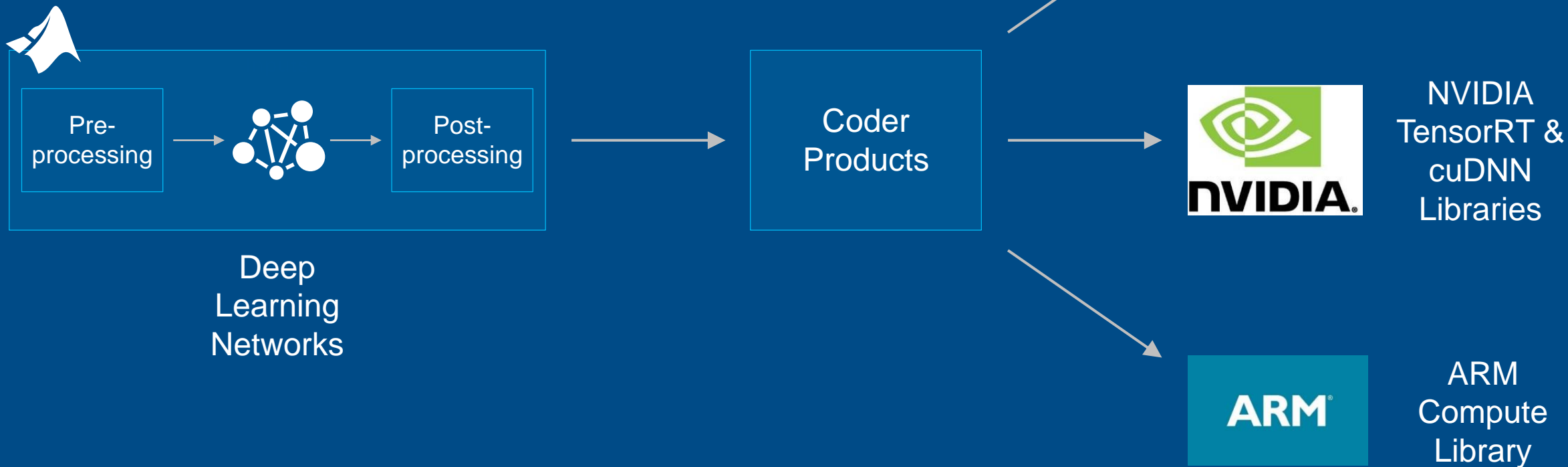
AlexNet

ResNet-50

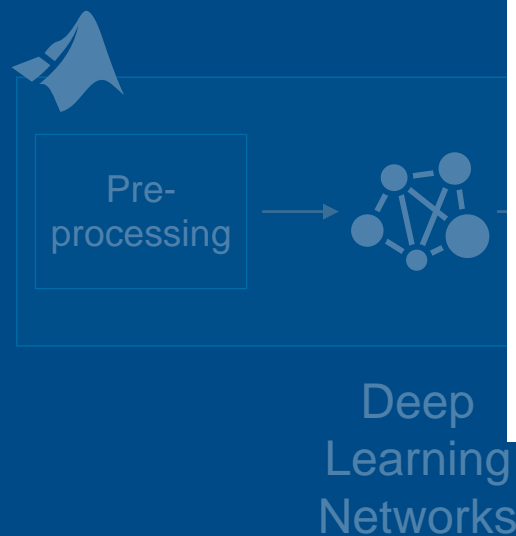
Using Models from Other Frameworks



Deploying Deep Learning Applications



Deploying Deep Learning Applications



Find out more:
**임베디드 하드웨어로의 딥러닝
 응용프로그램 배포**

**인공지능과 딥러닝 트랙
 송완빈**



Intel
MKL-DNN
Library

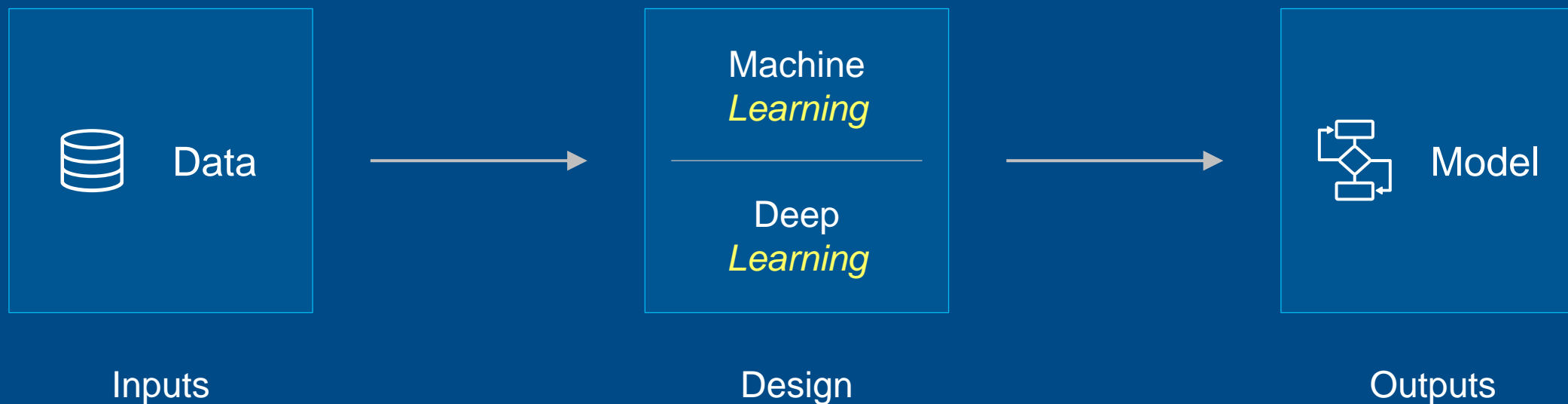


NVIDIA
TensorRT &
cuDNN
Libraries



ARM
Compute
Library

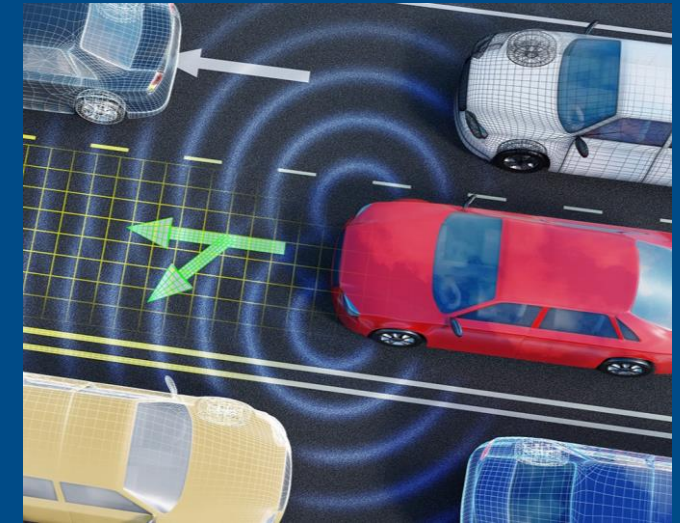
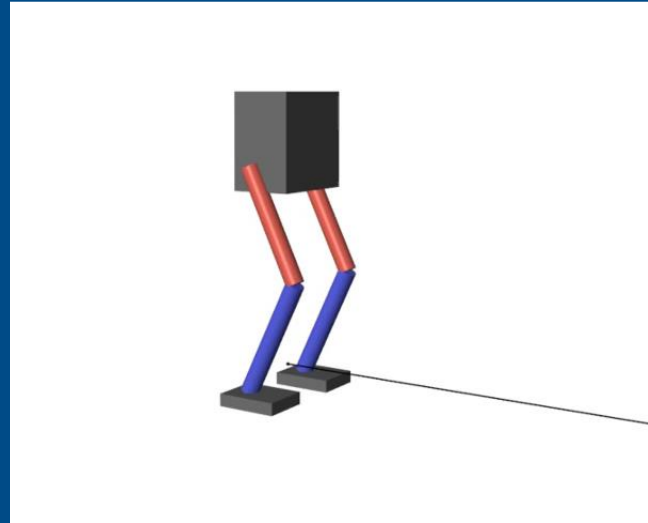
Using MATLAB and Simulink for Reinforcement Learning



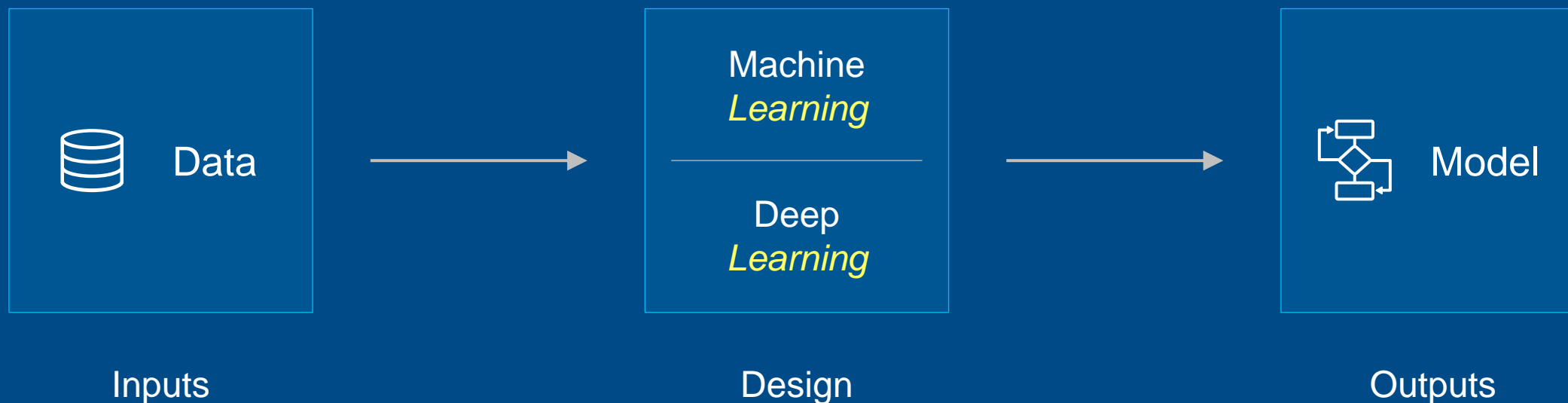
MATLAB® & SIMULINK®



Using MATLAB and Simulink for Reinforcement Learning



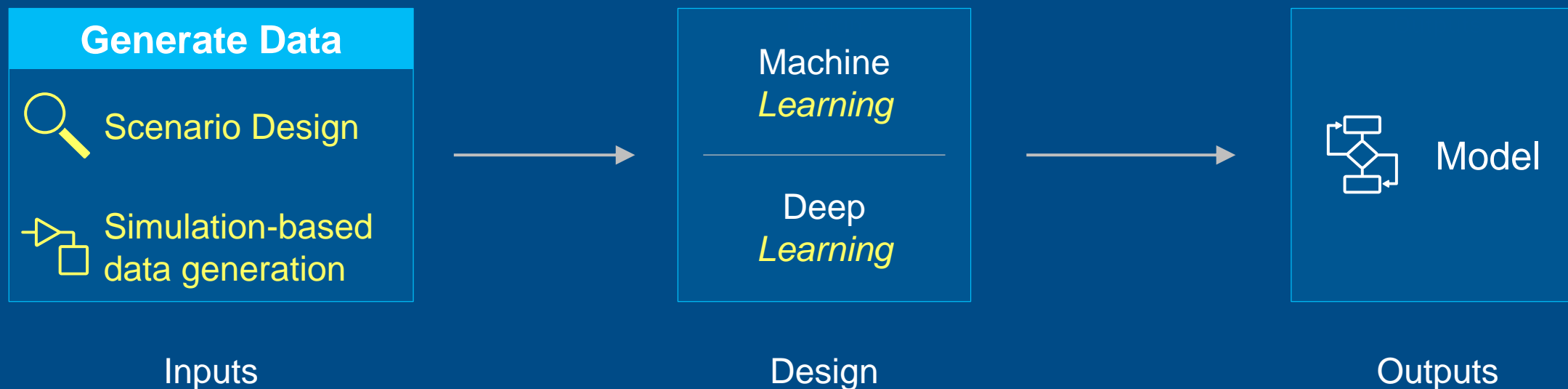
Using MATLAB and Simulink for Reinforcement Learning



MATLAB® & SIMULINK®



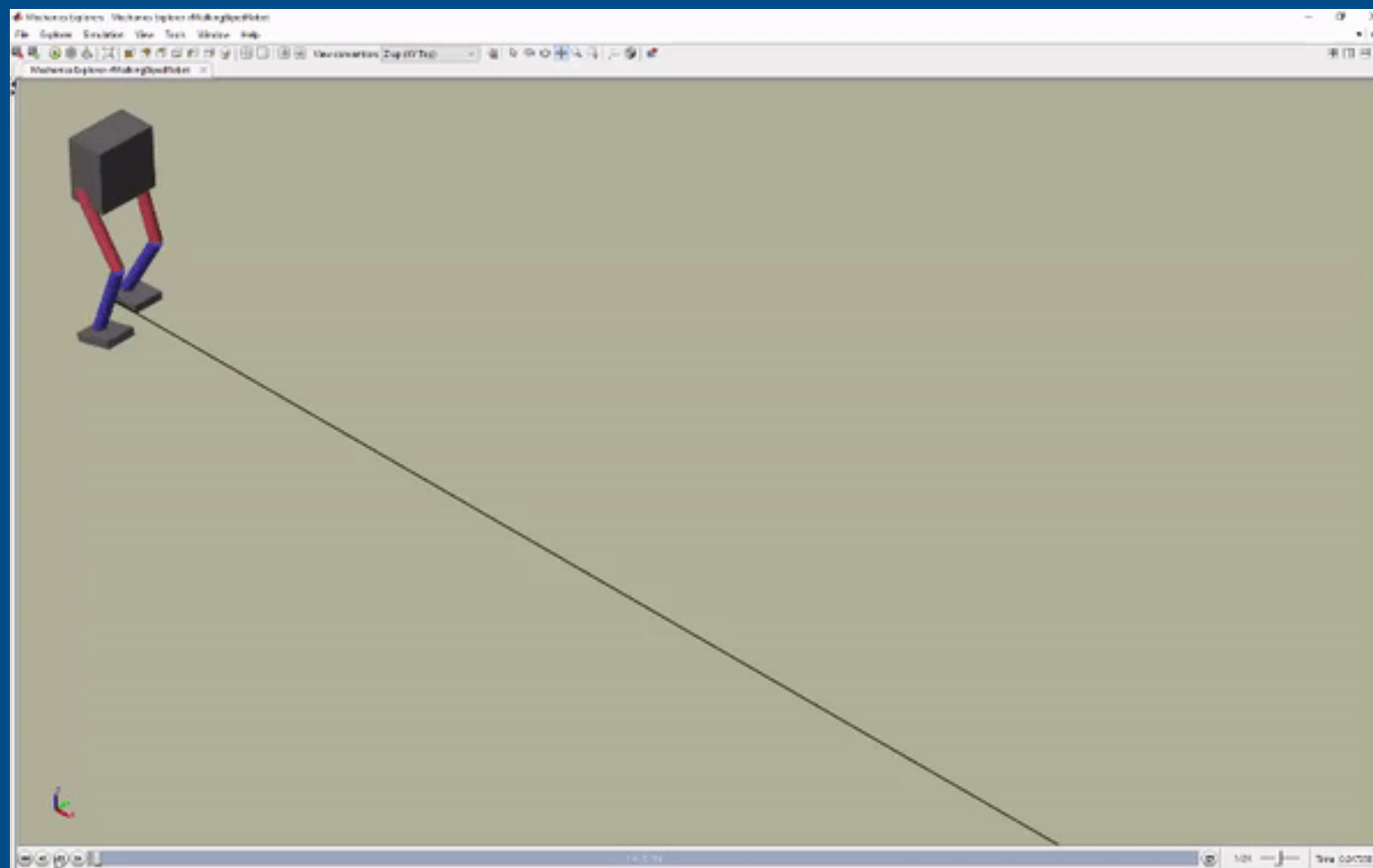
Using MATLAB and Simulink for Reinforcement Learning



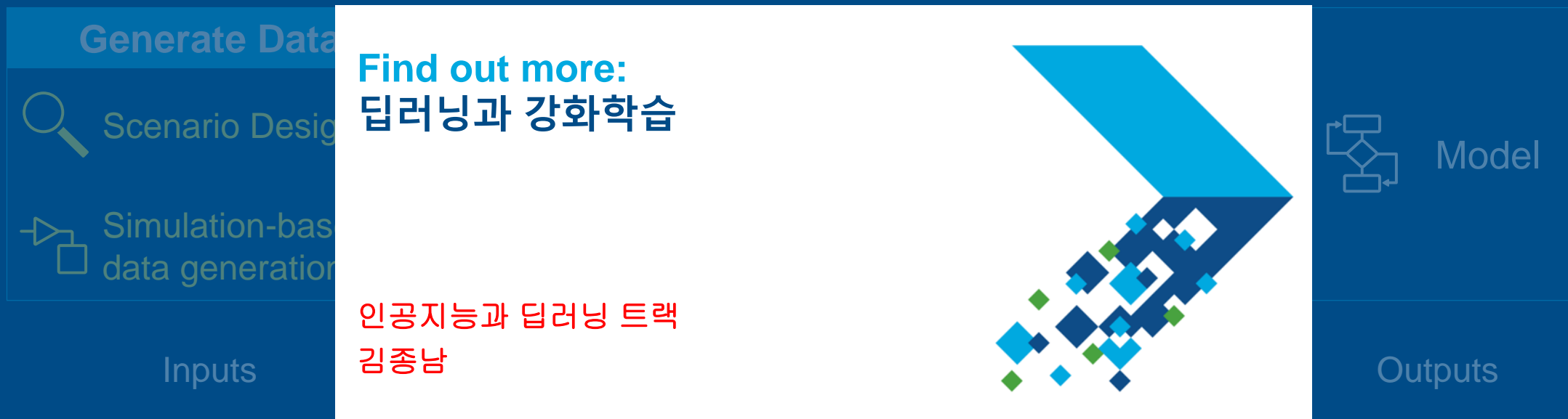
MATLAB® & SIMULINK®



Using MATLAB and Simulink for Reinforcement Learning



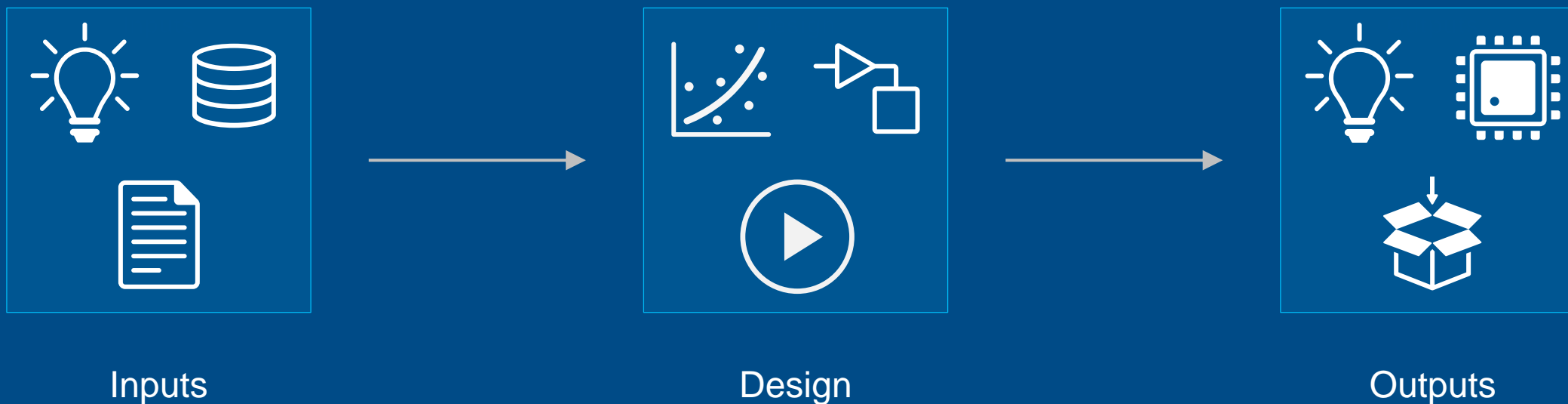
Using MATLAB and Simulink for Reinforcement Learning



MATLAB® & SIMULINK®



Using MATLAB & Simulink to Build Algorithms in Everything



MATLAB® & SIMULINK®



Working with Text Data



```

Vehicle_Repairs.csv x +
Dept,JobDate,jobno,Vehicleid,UnitNo,Reason,Notes,CostParts,CostLabor,CostTotal
1020,01/06/2015 12:00:00 AM,14073,118743,14,04 DRIVER'S REPORT,"PM SERVICE, CHECK TURN SIGNAL, CLUNKING NOISE WHEN DRIVING",493.85,0,493.85
1020,01/14/2015 12:00:00 AM,14232,230973,13,08 PM SERVICE ***,"SERVICEROB,EXT,5604",38.869999999999997,0,38.869999999999997
2111,01/02/2015 12:00:00 AM,14006,1243,116,04 DRIVER'S REPORT,NEED 4 PLOW PINS,45,0,45
2111,01/02/2015 12:00:00 AM,14140,B39109 ,178,04 DRIVER'S REPORT,INSTALL SPINNER ASSY,0,0,0
2111,01/03/2015 12:00:00 AM,14163,574950,215,13 SNOW BREAKDOWN,DONT START,0,0,0
2111,01/05/2015 12:00:00 AM,14169,A00413 ,283,04 DRIVER'S REPORT,DOG BONE PIN BROKEN,20,0,20
2111,01/06/2015 12:00:00 AM,14000,766153,248,08 PM SERVICE ***,"NEED SERVICE, CHECK BRAKES",387.17,0,387.17
2111,01/06/2015 12:00:00 AM,14155,525670,232,04 DRIVER'S REPORT,HYD CAP CHECK ENGINE LIGHT ON,12.95,0,12.95
2111,01/06/2015 12:00:00 AM,14157,621909,213,40 NEGLIGENCE,TARP VALVE STICKINGRIGHT SIDE MIRROR BRACKET BROKEN,50.02,0,50.02
2111,01/06/2015 12:00:00 AM,14164,1226,117,13 SNOW BREAKDOWN,HANDLES IN CAB LOOSE,0,0,0
2111,01/06/2015 12:00:00 AM,14165,525999,114,04 DRIVER'S REPORT,NO PLOW LIGHTS,0,0,0
2111,01/06/2015 12:00:00 AM,14172,B34632 ,276,10 ROADCALL,WILL NOT START,0,0,0
2111,01/06/2015 12:00:00 AM,14174,1469,122,10 ROADCALL,WILL NOT START,0,0,0
2111,01/06/2015 12:00:00 AM,14175,68932,147,10 ROADCALL,WILL NOT START,0,0,0
2111,01/06/2015 12:00:00 AM,14176,68933,148,10 ROADCALL,WILL NOT START,0,0,0
2111,01/06/2015 12:00:00 AM,14177,621907,208,10 ROADCALL,WILL NOT START,0,0,0
2111,01/06/2015 12:00:00 AM,14181,337657,218,04 DRIVER'S REPORT,CONVEORY NOT WORKING,0,0,0
2111,01/06/2015 12:00:00 AM,14182,D-1920 ,164,10 ROADCALL,DONT START,0,0,0
2111,01/06/2015 12:00:00 AM,14183,525998,217,10 ROADCALL,DONT START,0,0,0
2111,01/06/2015 12:00:00 AM,14184,526000,225,10 ROADCALL,DONT START,0,0,0
2111,01/06/2015 12:00:00 AM,14185,621921,214,04 DRIVER'S REPORT,CONVORY NOT WORKING,0,0,0
2111,01/07/2015 12:00:00 AM,14188,001469 ,201,04 DRIVER'S REPORT,needs def/jim f,0,0,0
2111,01/07/2015 12:00:00 AM,14190,337656,219,04 DRIVER'S REPORT,NEEDS FLOOR MATTS,65.069999999999993,0,65.069999999999993
2111,01/07/2015 12:00:00 AM,14191,B34632 ,276,10 ROADCALL,DONT START,0,0,0
2111,01/07/2015 12:00:00 AM,14196,1222,118,04 DRIVER'S REPORT,HARDWARE FOR REAR SPRINGS,14.32,0,14.32
2111,01/07/2015 12:00:00 AM,14199,52565,626,04 DRIVER'S REPORT,WASHER FLUIDDEF,28.88,0,28.88
2111,01/09/2015 12:00:00 AM,14107,1467,121,08 PM SERVICE ***,"REMOVE & REPLACE REAR SPRINGS, CHECK COOLANT TUBESPM SERVICE",4697.55,0,

```

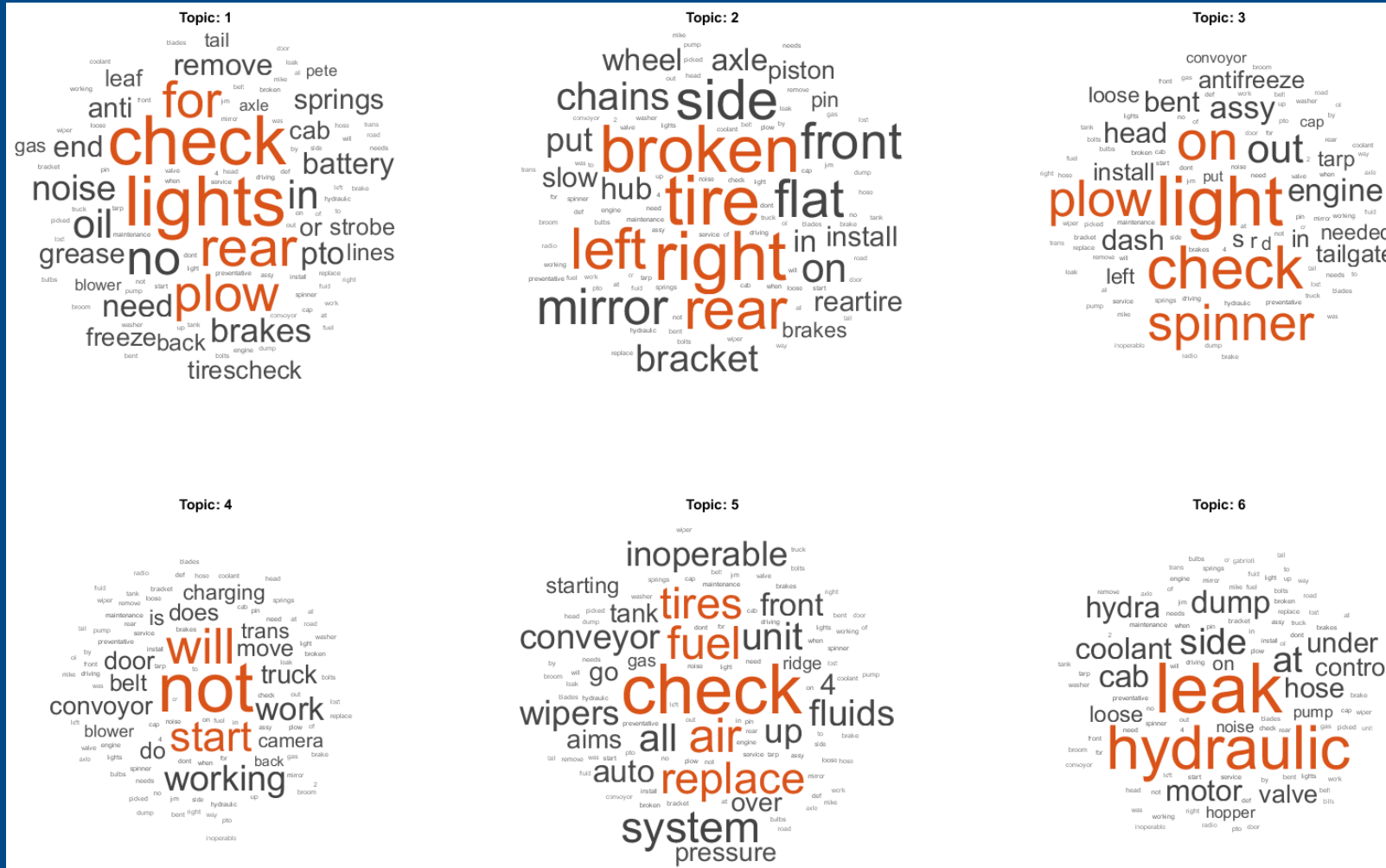
Working with Text Data



```
t = readtable(filename, 'TextType', 'string');
disp(t(1:20,6:7))
```

	Reason		Notes
"04	DRIVER'S REPORT"		"PM SERVICE, CHECK TURN SIGNAL, CLUNKING NOISE WHEN DRIVING"
"08	PM SERVICE	***"	"SERVICEROB,EXT,5604"
"04	DRIVER'S REPORT"		"NEED 4 PLOW PINS"
"04	DRIVER'S REPORT"		"INSTALL SPINNER ASSY"
"13	SNOW BREAKDOWN"		"DONT START"
"04	DRIVER'S REPORT"		"DOG BONE PIN BROKEN"
"08	PM SERVICE	***"	"NEED SERVICE, CHECK BRAKES"
"04	DRIVER'S REPORT"		"HYD CAP CHECK ENGINE LIGHT ON"
"40	NEGLIGENCE"		"TARP VALVE STICKINGRIGHT SIDE MIRROR BRACKET BROKEN"
"13	SNOW BREAKDOWN"		"HANDLES IN CAB LOOSE"
"04	DRIVER'S REPORT"		"NO PLOW LIGHTS"
"10	ROADCALL"		"WILL NOT START"
"10	ROADCALL"		"WILL NOT START"
"10	ROADCALL"		"WILL NOT START"
"10	ROADCALL"		"WILL NOT START"
"10	ROADCALL"		"WILL NOT START"
"04	DRIVER'S REPORT"		"CONVEORY NOT WORKING"
"10	ROADCALL"		"DONT START"
"10	ROADCALL"		"DONT START"
"10	ROADCALL"		"DONT START"

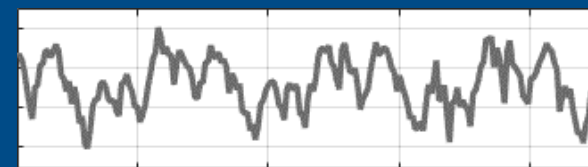
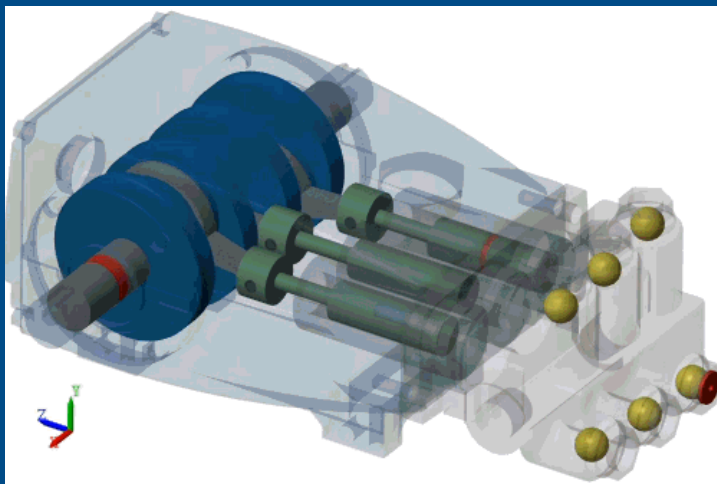
Working with Text Data



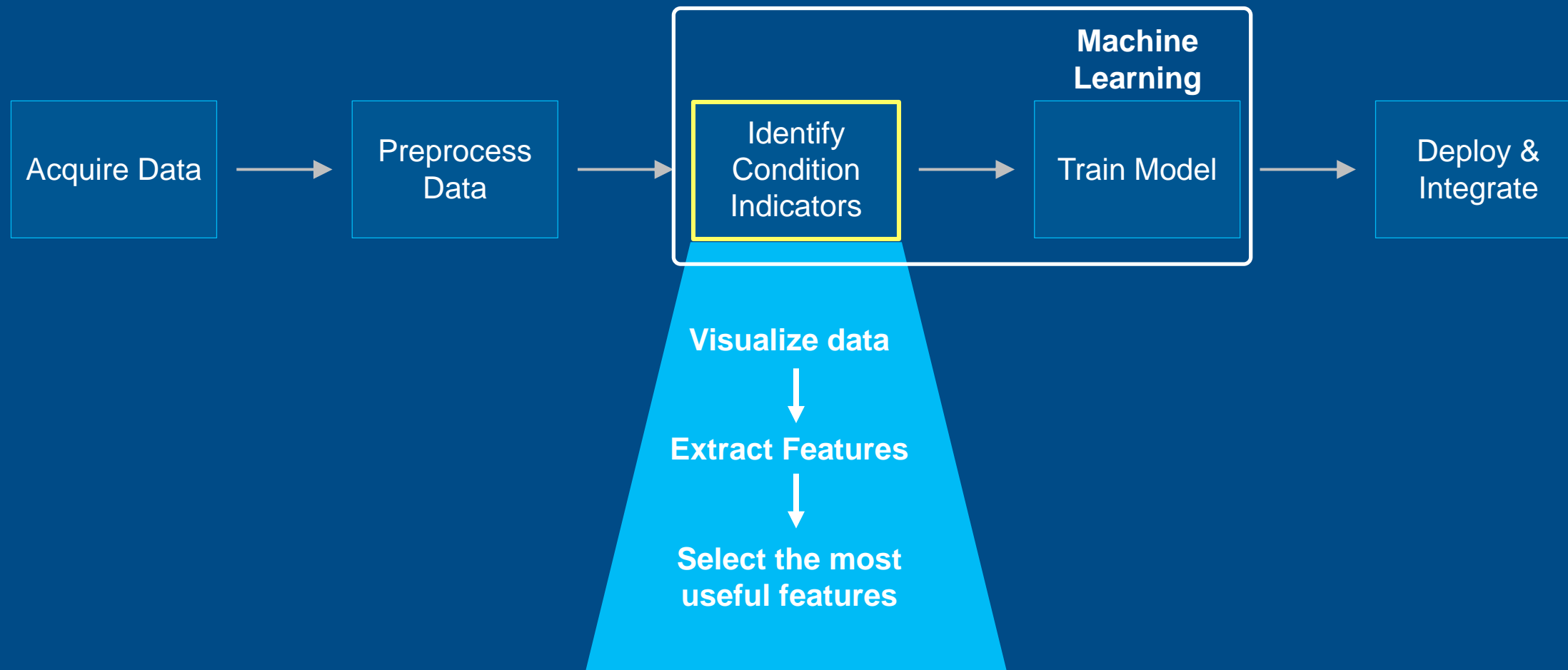
Creating Your Own Data

The screenshot displays the Signal Editor window for an input dataset. The interface is divided into several sections:

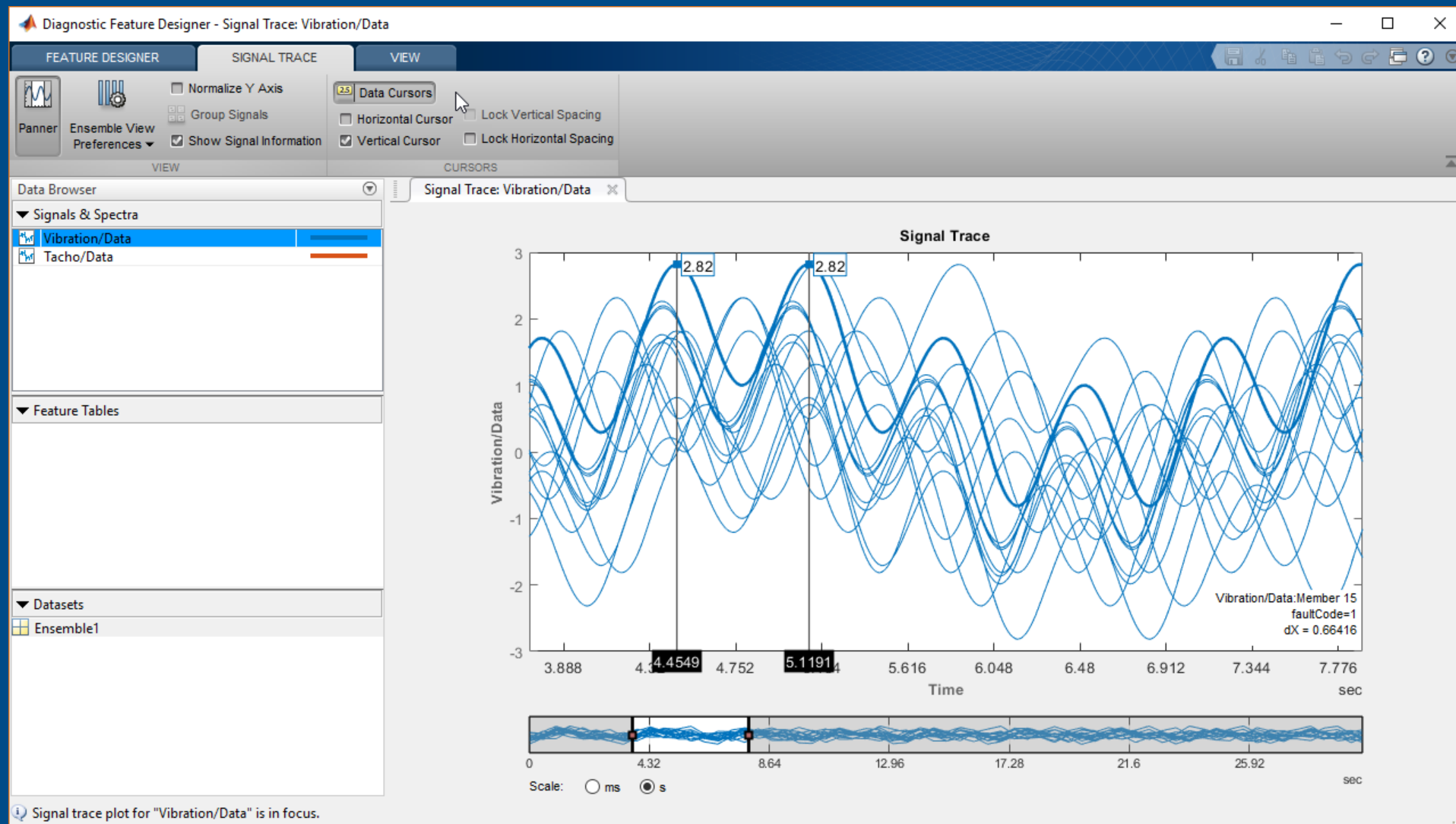
- Toolbar:** Includes 'DRAW' (Select, Move, Delete, Edit Point, Erase, Clear), 'TOOLS' (Insert Point, $f(x)$ Expression, Insert Line, Draw), 'ALIGN' (Snap X to Grid, Snap X to Tick, Snap Y to Grid, Snap Y to Tick), 'SHOW' (Markers, Grid), 'ZOOM & PAN' (Zoom In, Zoom Out, Pan), 'CAPTURE' (Screenshot), 'INSERT', and 'CLOSE'.
- SCENARIOS AND SIGNALS:** A list containing 'Scenario'.
- Signal Plot:** A graph titled 'Signal' with 'TIME' on the x-axis (0 to 10) and y-axis values from -4 to 4. A dashed grid is visible.
- 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



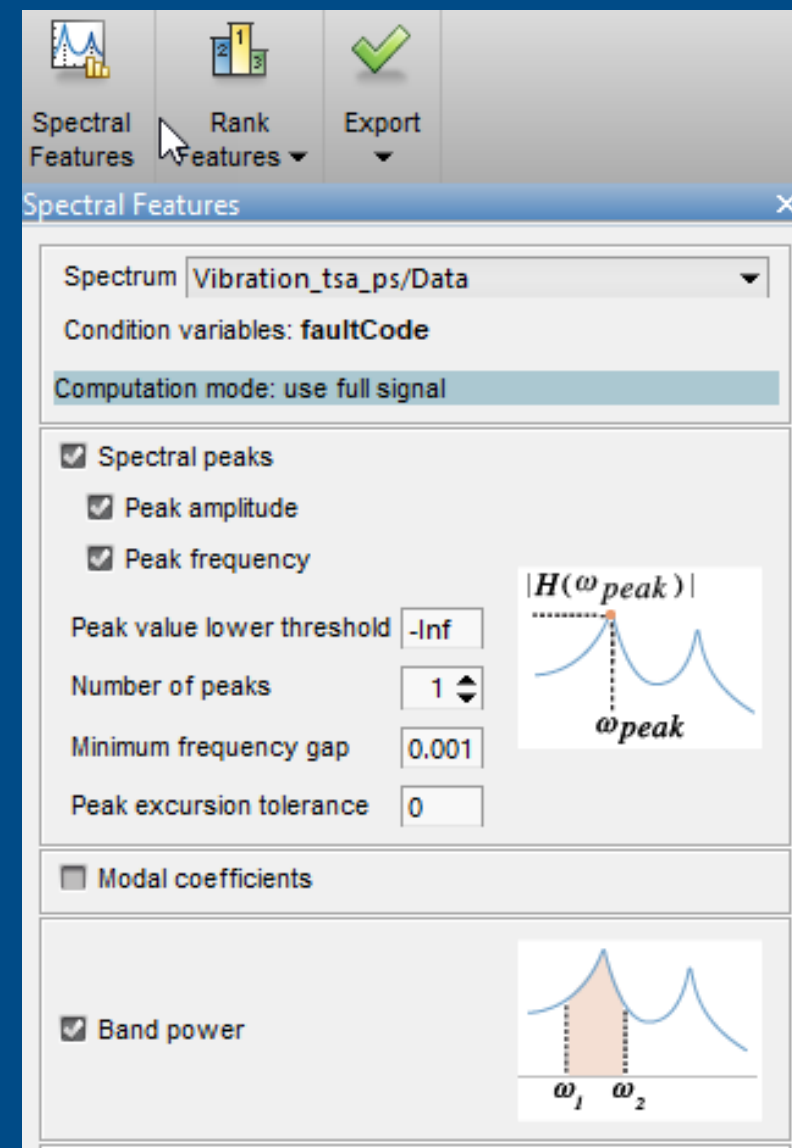
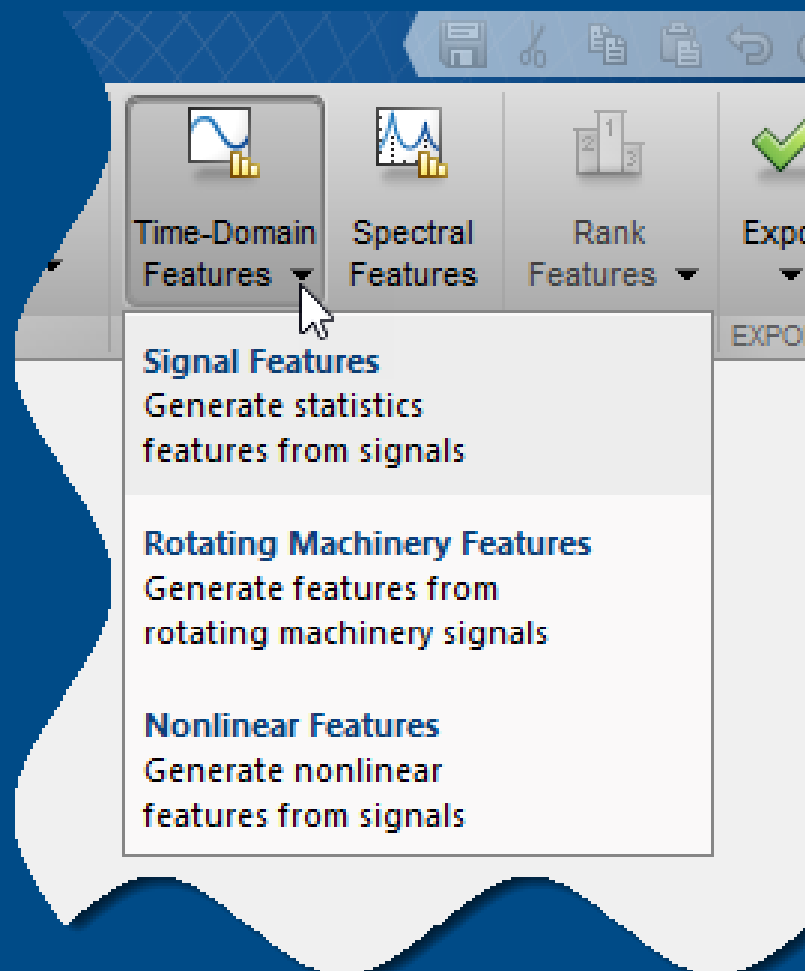
Identifying the Useful Data



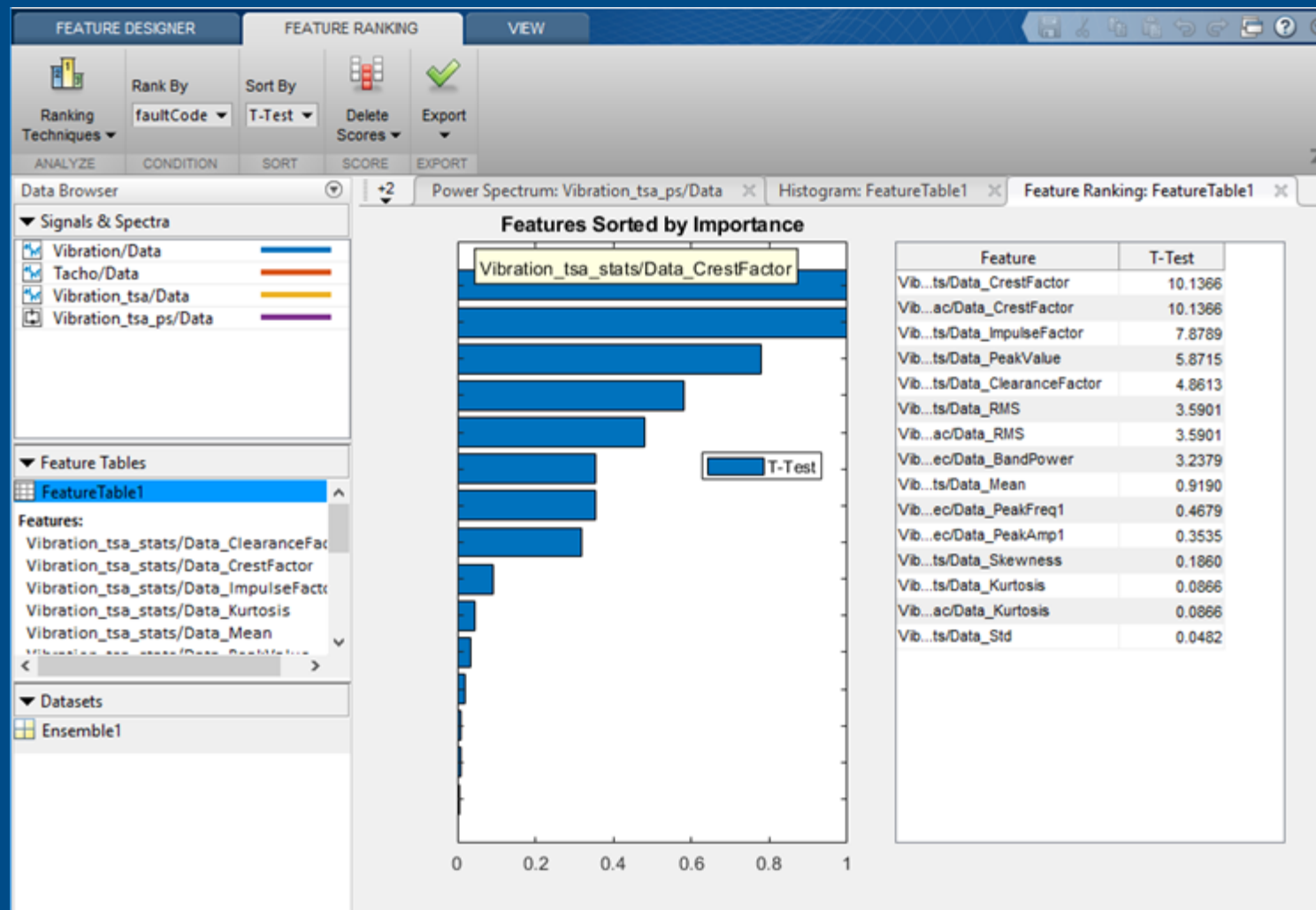
Identifying the Useful Data



Identifying the Useful Data



Identifying the Useful Data



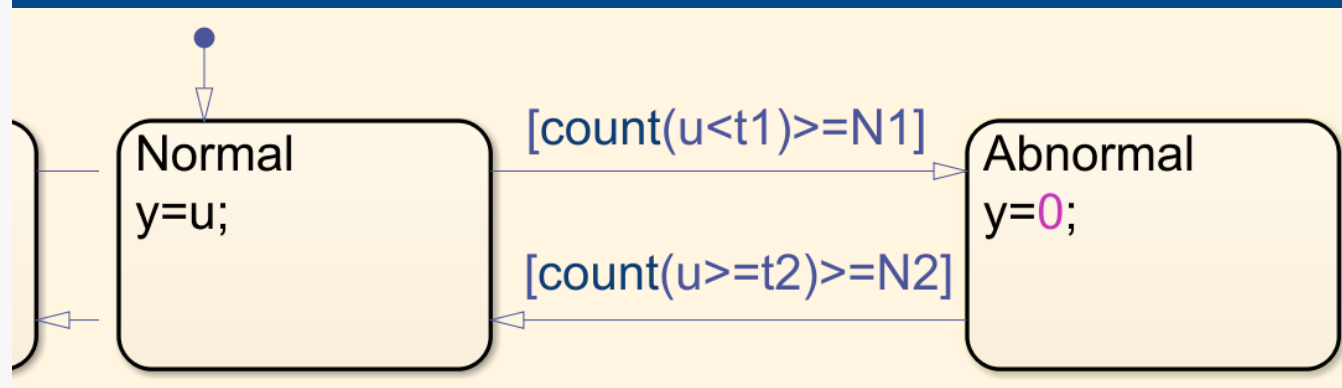
Designing Decision Logic with Stateflow



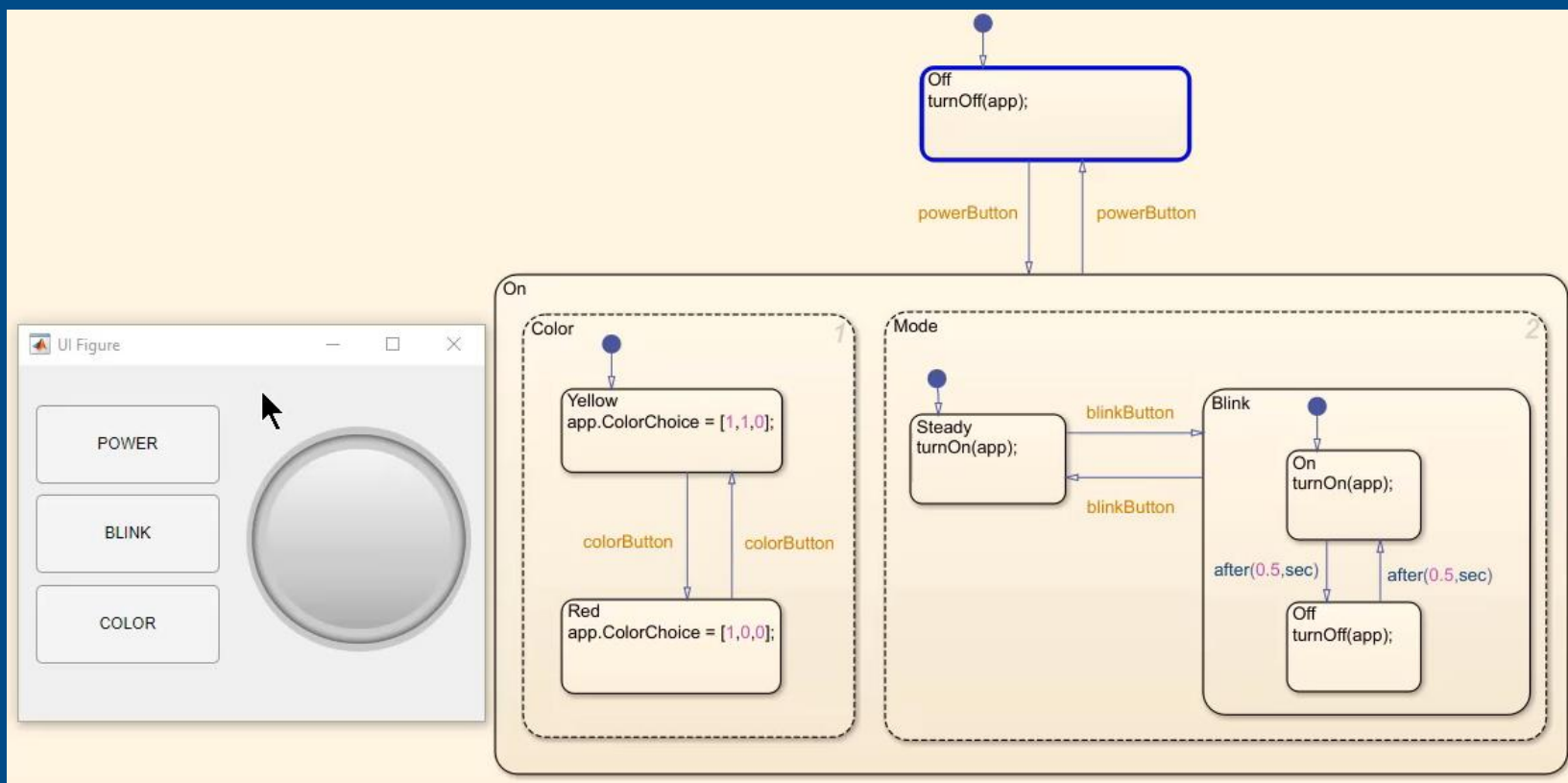
```

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
end

```



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

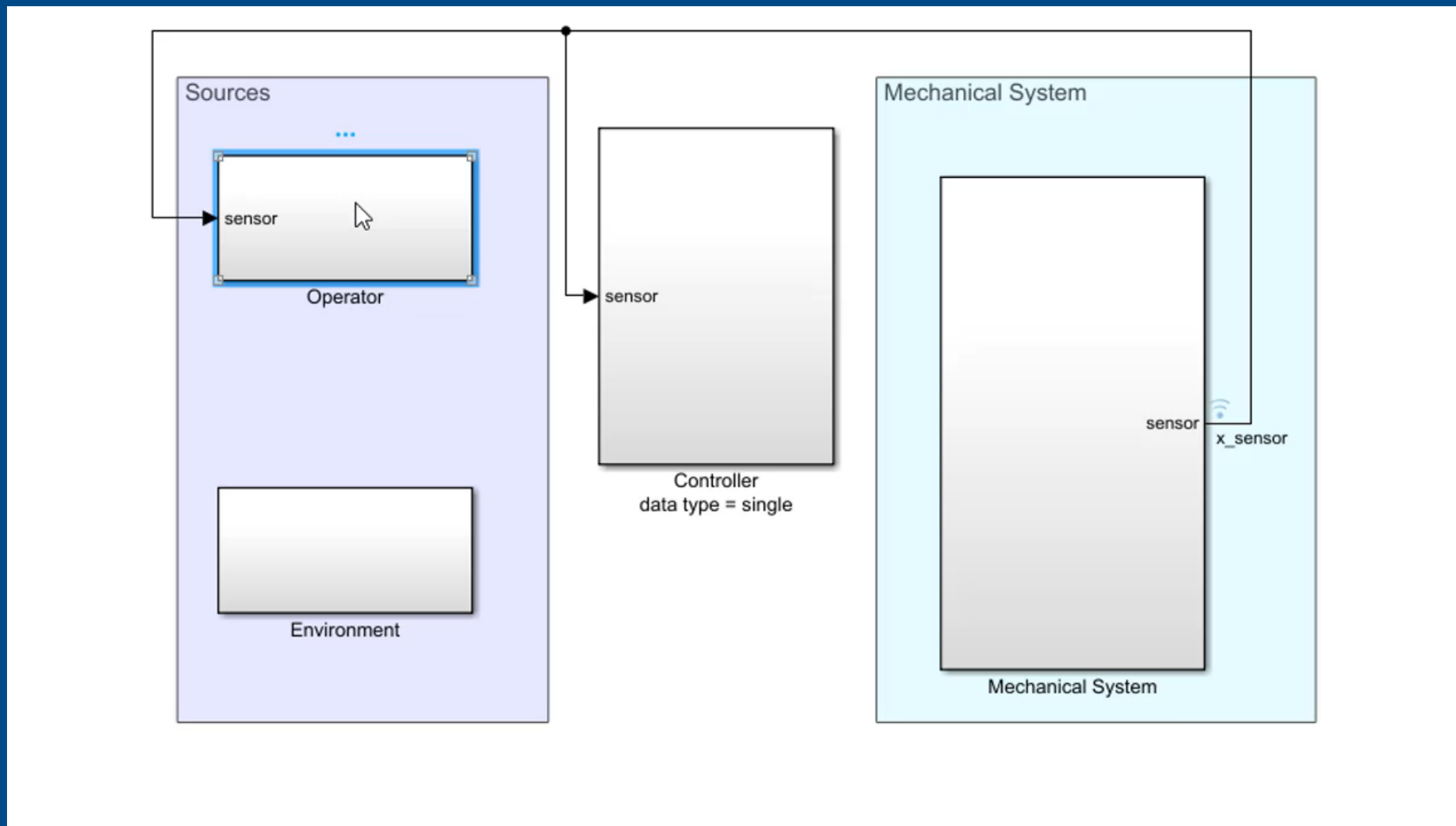
% Button pushed function: COLORButton
function COLORButtonPushed(app, event)
    app.LanternLogic.colorButton();
end

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

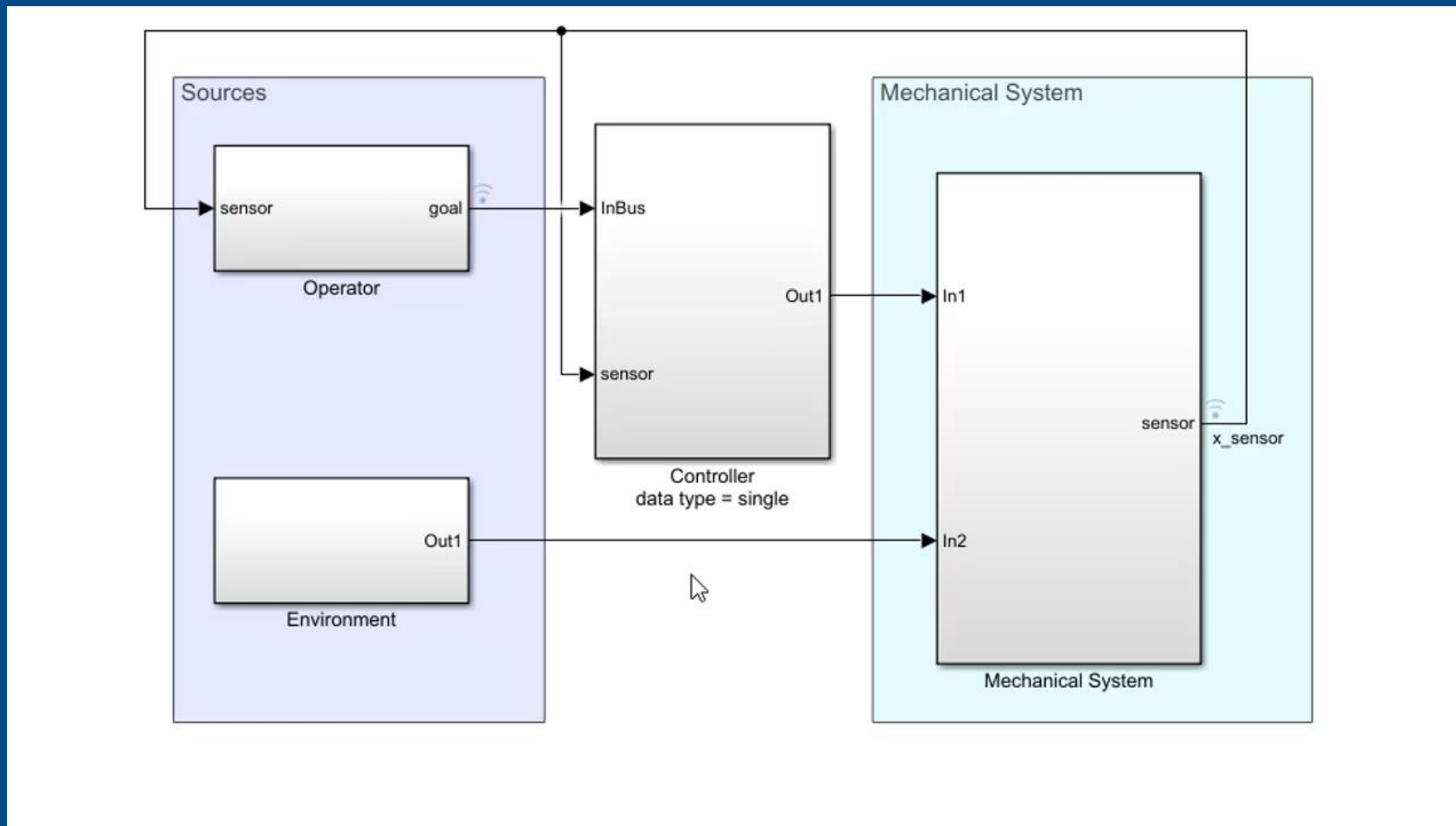
% Button pushed function: BLINKButton
function BLINKButtonPushed(app, event)
    app.LanternLogic.blinkButton();
end
end

```

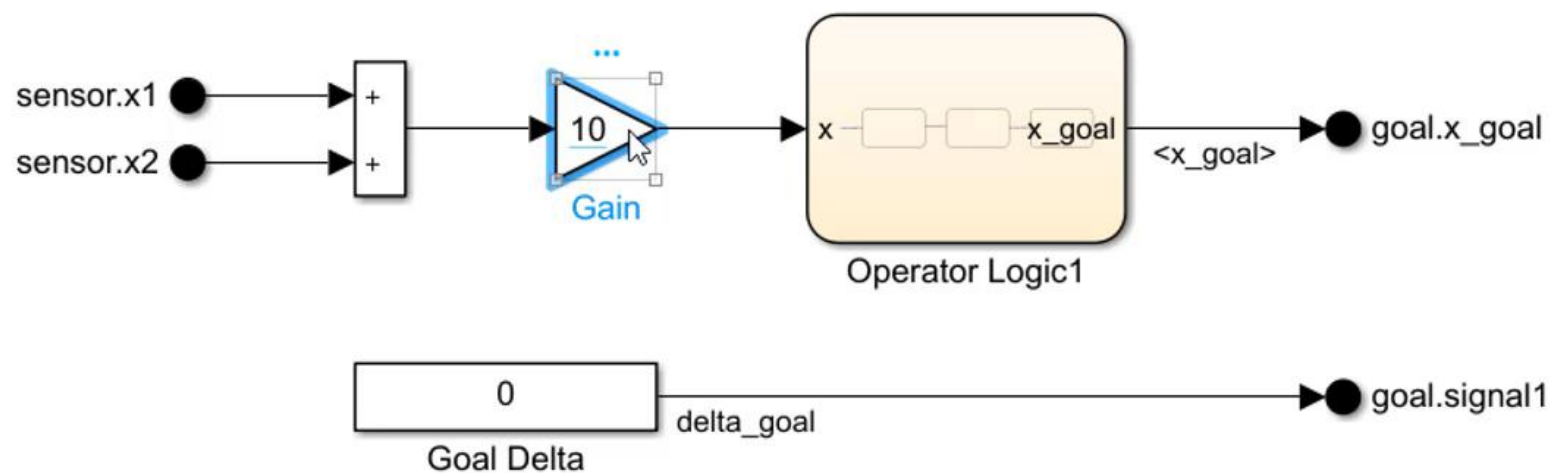
Editing at the Speed of Thought



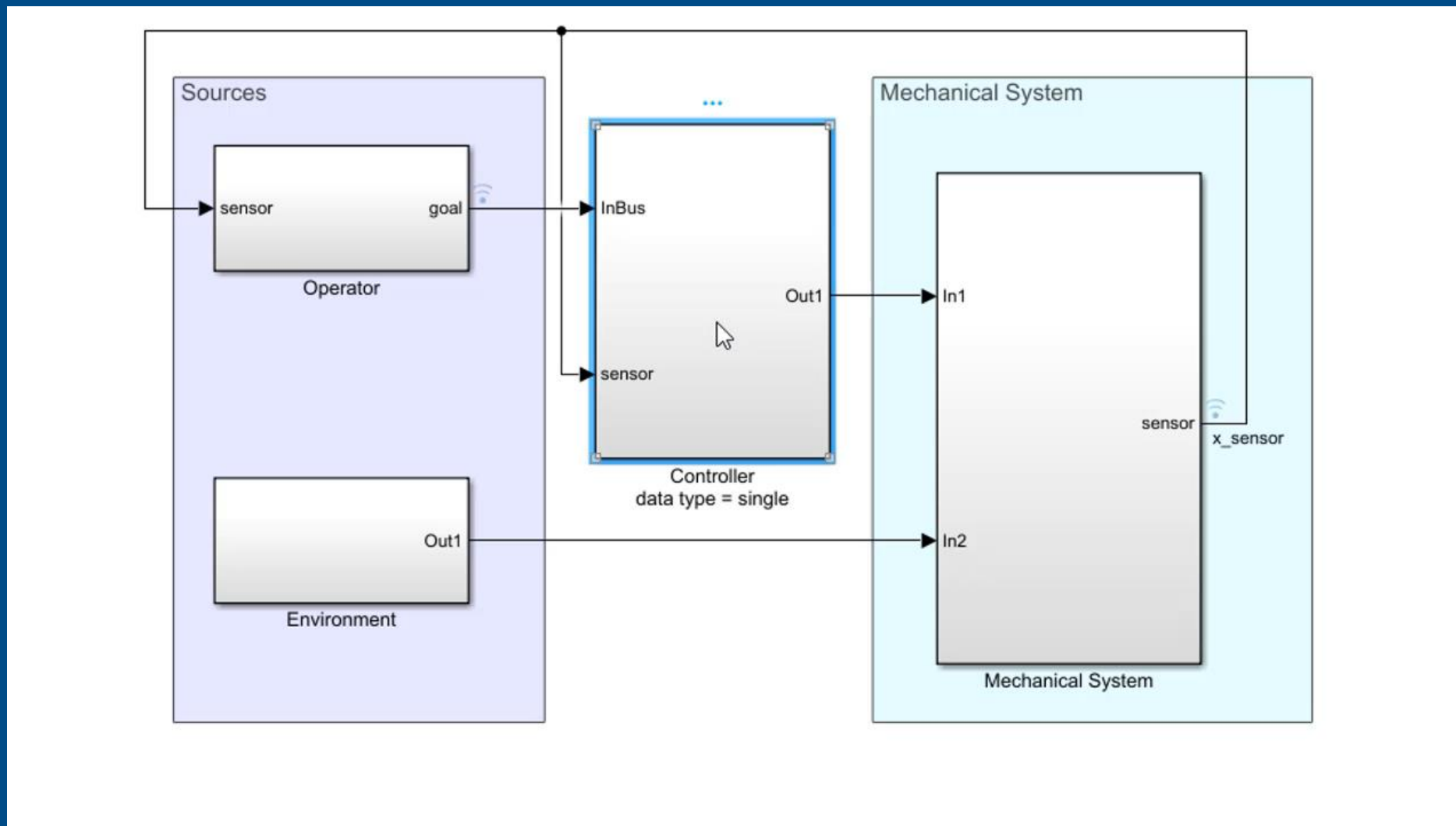
Editing at the Speed of Thought



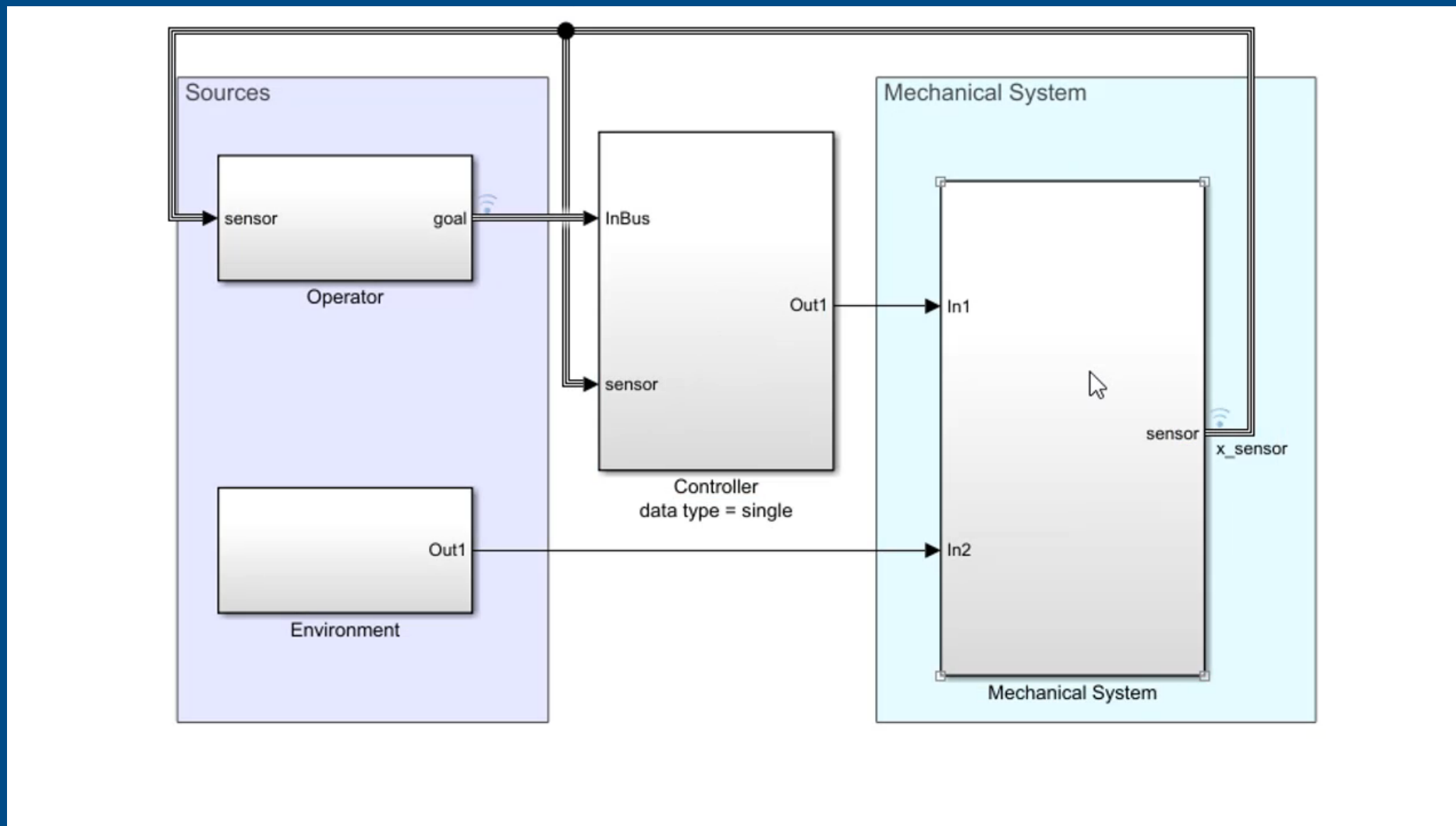
Editing at the Speed of Thought



Editing at the Speed of Thought



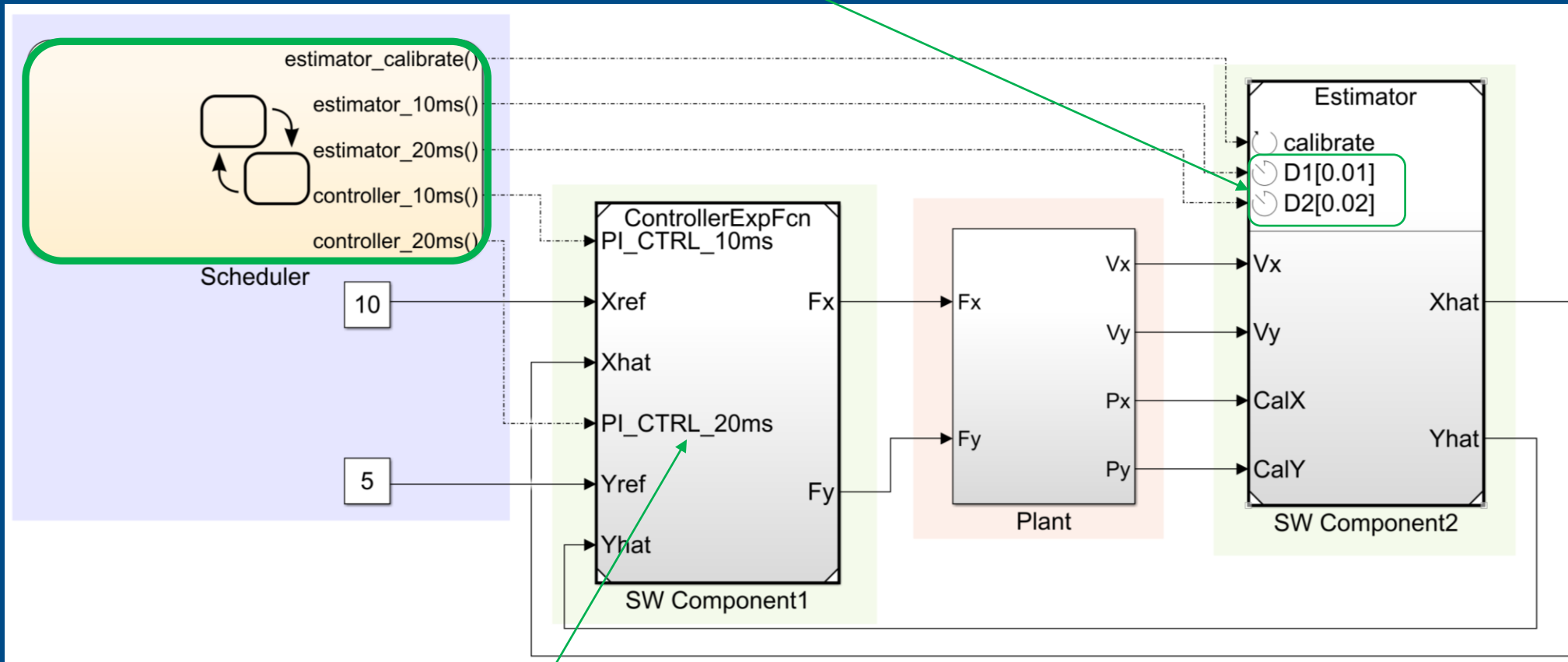
Editing at the Speed of Thought



Controlling the Execution of Model Components



Schedulable Rate-Based Model



Export Function Model

Controlling the Execution of Model Components

SCHEDULE EDITOR

MANAGE PARTITIONS EXECUTION MODEL DISPLAY VIEW

LEGEND

EXECUTION ORDER

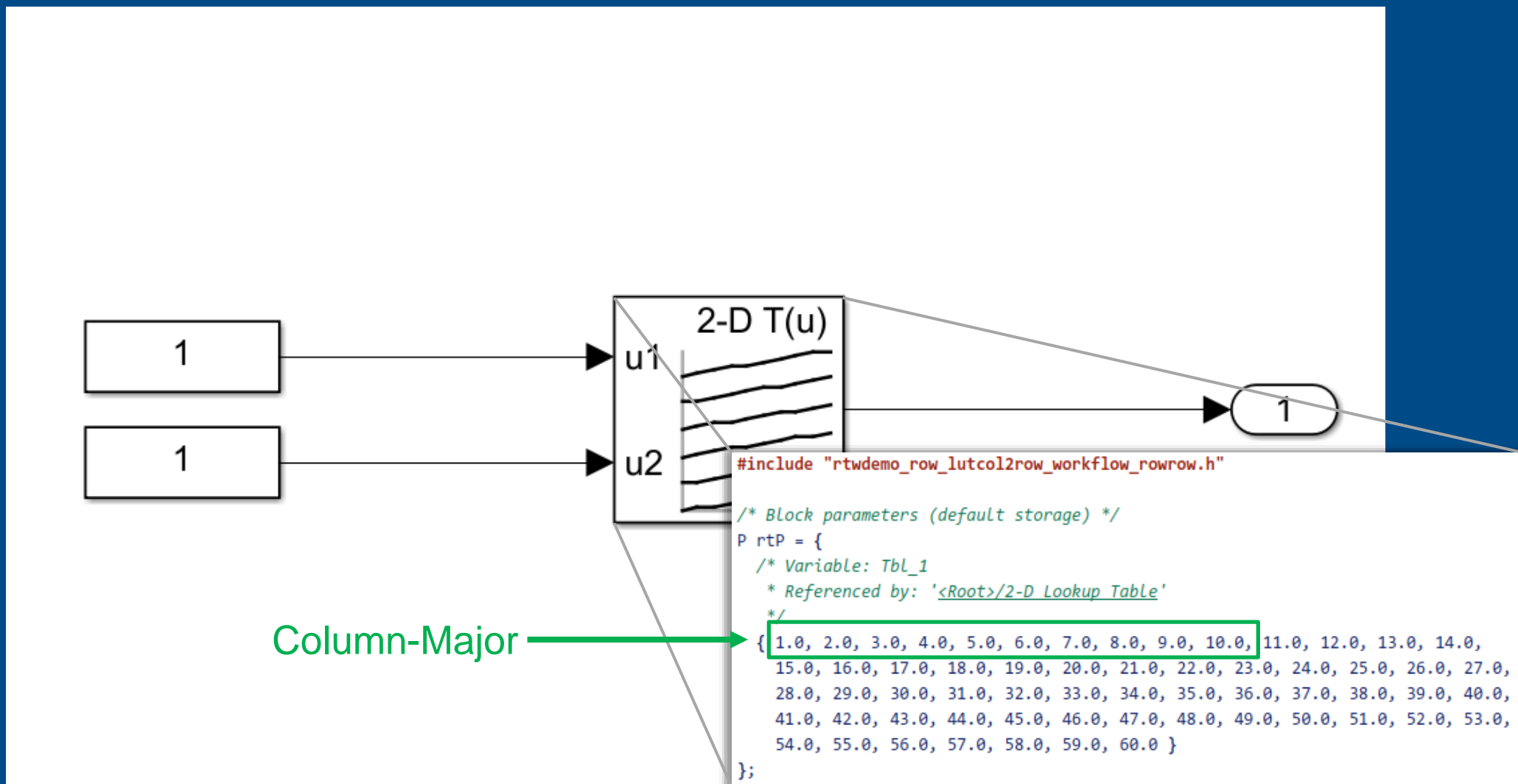
Order	Name	Rate
1	Cont	0
2	D1	0.001
3	ThrottleControl.APPSnsrRun	-1
4	ThrottleControl.ActuatorRun5ms	0.005
5	ThrottleControl.TPSSecondaryRun5ms	0.005
6	ThrottleControl.MonitorRun5ms	0.005
7	ThrottleControl.ControllerRun5ms	0.005
8	D3	0.01
9	ThrottleControl.TPSPrimaryRun10ms	0.01

PROPERTY INSPECTOR

Partition	
Name	D3
Rate	0.01
Type	Implicit periodic partition

MANAGE PARTITIONS

Simplifying Integration with External C/C++ Code



Simplifying Integration with External C/C++ Code



rtwdemo_row_lutcol2row_workflow_rowrow

2-D T(u)

u1

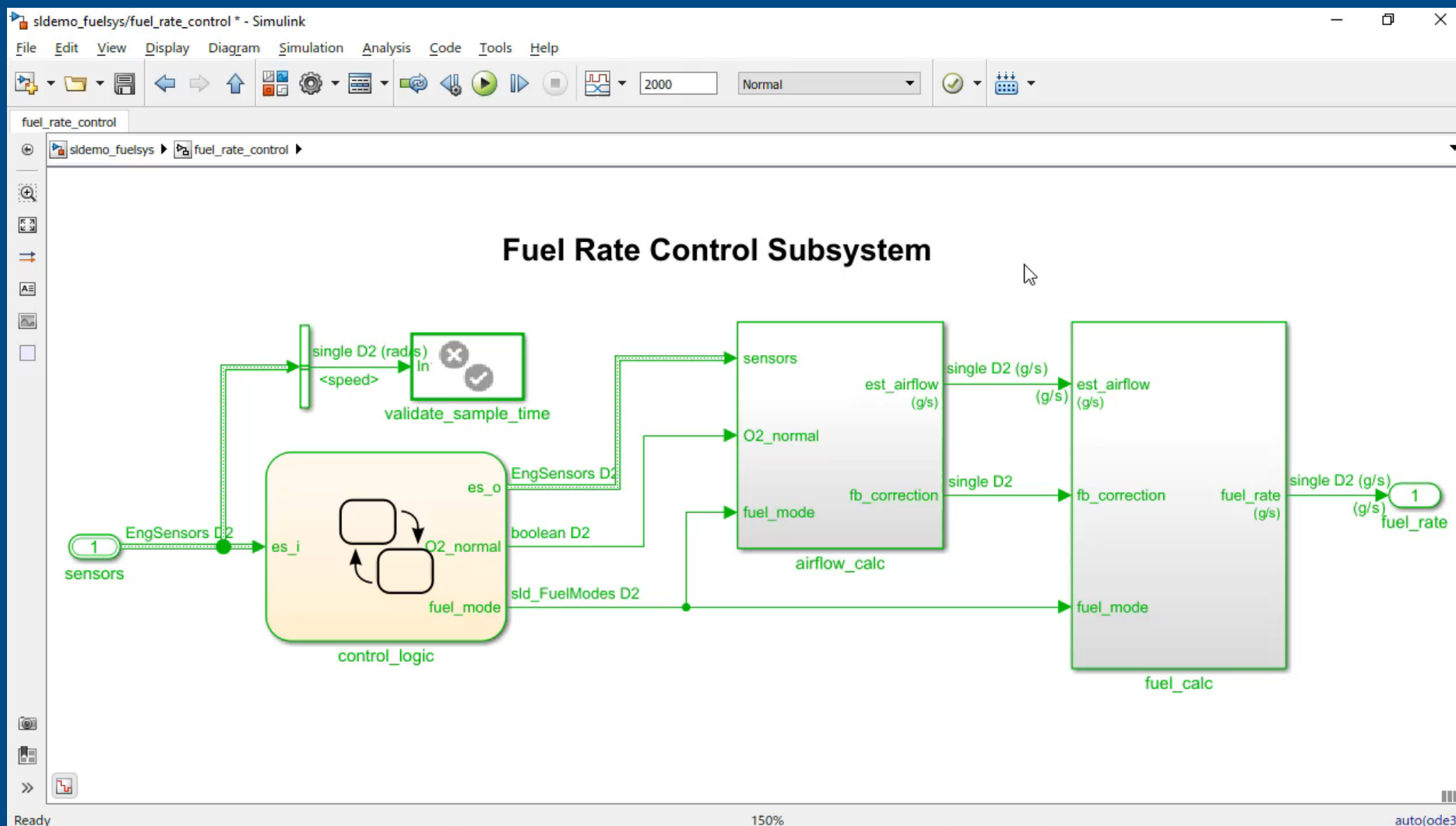
u2

Row-Major

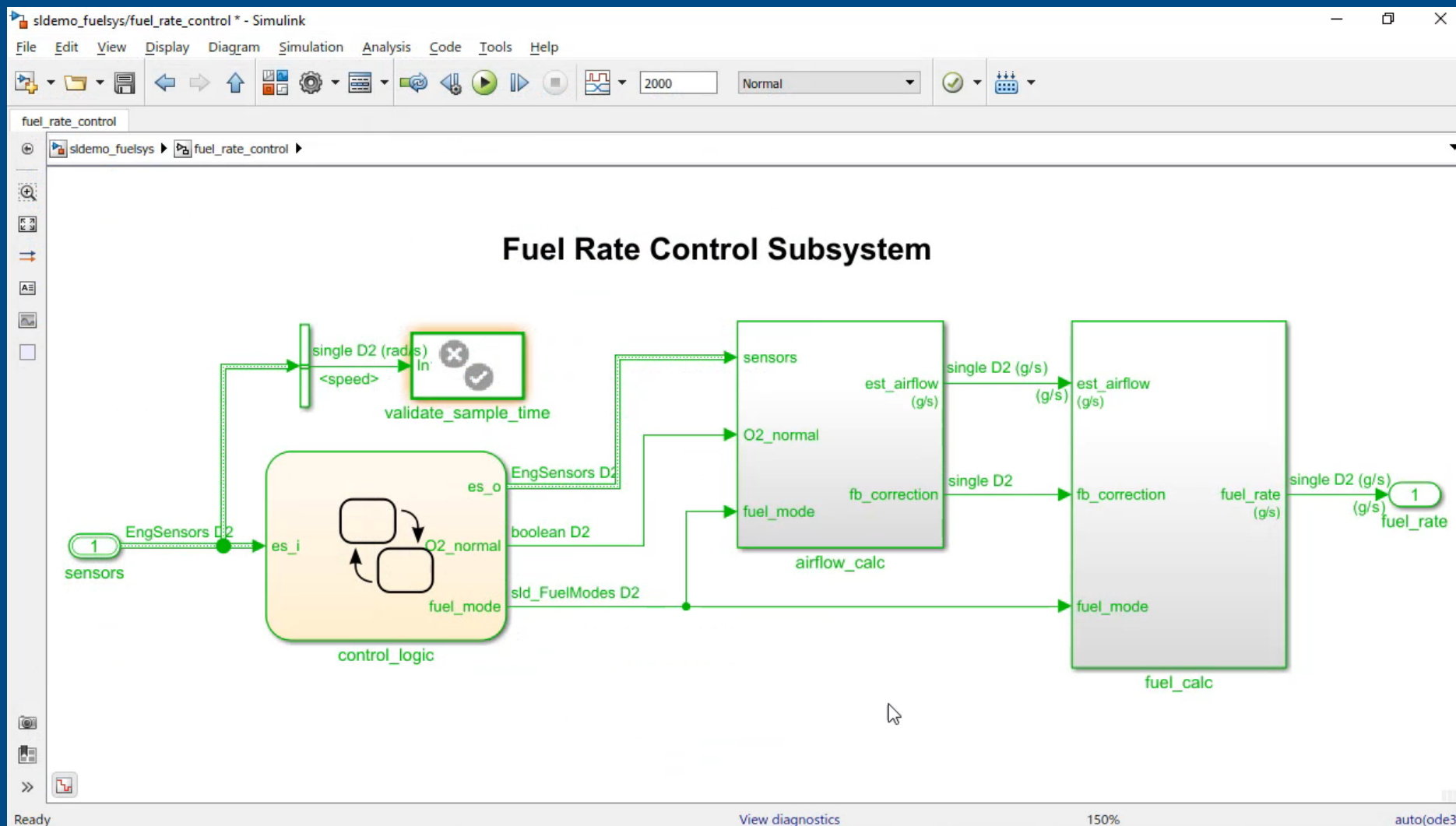
```
#include "rtwdemo_row_lutcol2row_workflow_rowrow.h"

/* Block parameters (default storage) */
P rtP = {
  /* Variable: Tbl_1
   * Referenced by: '<Root>/2-D Lookup Table'
   */
  { 1.0, 11.0, 21.0, 31.0, 41.0, 51.0, 2.0, 12.0, 22.0, 32.0, 42.0, 52.0, 3.0,
    13.0, 23.0, 33.0, 43.0, 53.0, 4.0, 14.0, 24.0, 34.0, 44.0, 54.0, 5.0, 15.0,
    25.0, 35.0, 45.0, 55.0, 6.0, 16.0, 26.0, 36.0, 46.0, 56.0, 7.0, 17.0, 27.0,
    37.0, 47.0, 57.0, 8.0, 18.0, 28.0, 38.0, 48.0, 58.0, 9.0, 19.0, 29.0, 39.0,
    49.0, 59.0, 10.0, 20.0, 30.0, 40.0, 50.0, 60.0 }
};
```

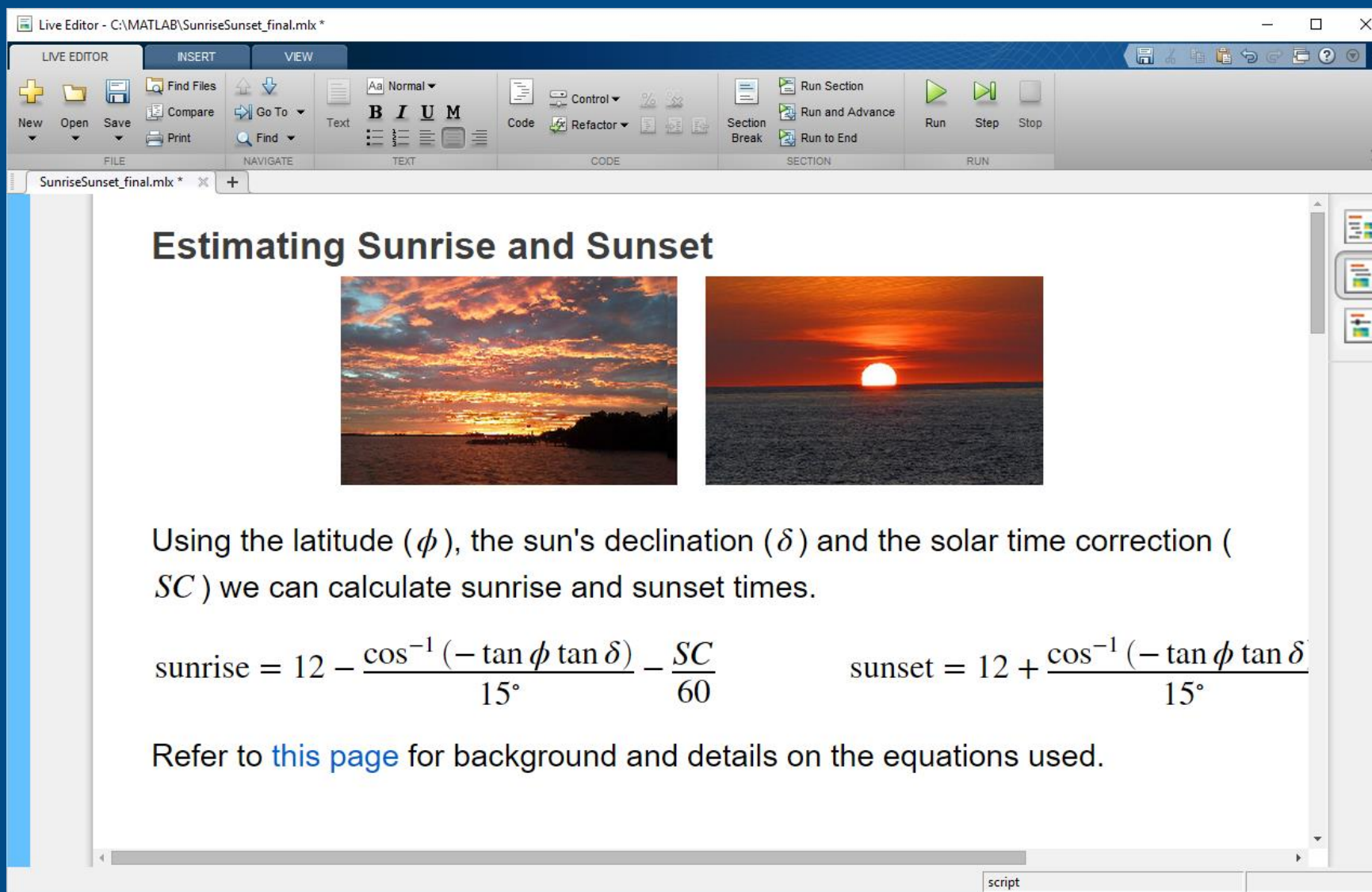
Viewing Generated Code Alongside the Model



Viewing Generated Code Alongside the Model



Sharing Live Scripts



Live Editor - C:\MATLAB\SunriseSunset_final.mlx*

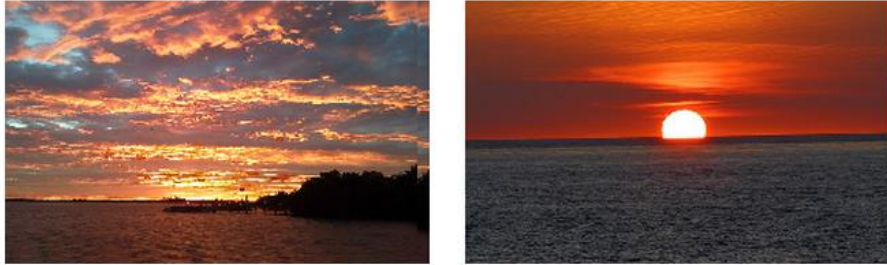
LIVE EDITOR INSERT VIEW

Find Files Go To Text Code Section Break Run Section Run and Advance Run to End Run Step Stop

FILE NAVIGATE TEXT CODE SECTION RUN

SunriseSunset_final.mlx * +

Estimating Sunrise and Sunset



Using the latitude (ϕ), the sun's declination (δ) 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.

script

Sharing Live Scripts

AutoSave [OFF] ExploringExoplanets.docx - Compatibility Mode David Garrison

File Home Insert Design Layout References Mailings Review View Help Tell me what you want to do Share

Clipboard Font Paragraph Styles Editing

Exploring Exoplanets

In this example we will explore some data on exoplanets - planets outside our own solar system. The data used here is a subset of data from the [NASA Exoplanet Archive](#). We will start by using the data to answer some questions about the set of exoplanets in the archive. Then we will do some calculations to try to identify planets in the archive that might be capable of supporting life.

```
exoplanets = readtable("exoplanets.xlsx");
exoplanets(1:10,:);
```

How Far Away Are these Planets?

There are 90 exoplanets within 50 light-years of earth and 450 exoplanets within 200 light-years.

```
histogram(3.26*exoplanets.st_distance,'BinWidth', 50)
xlim([0 1000])
ylabel 'Number of Planets'
xlabel 'Light Years from Earth'
```

Where is the nearest exoplanet?

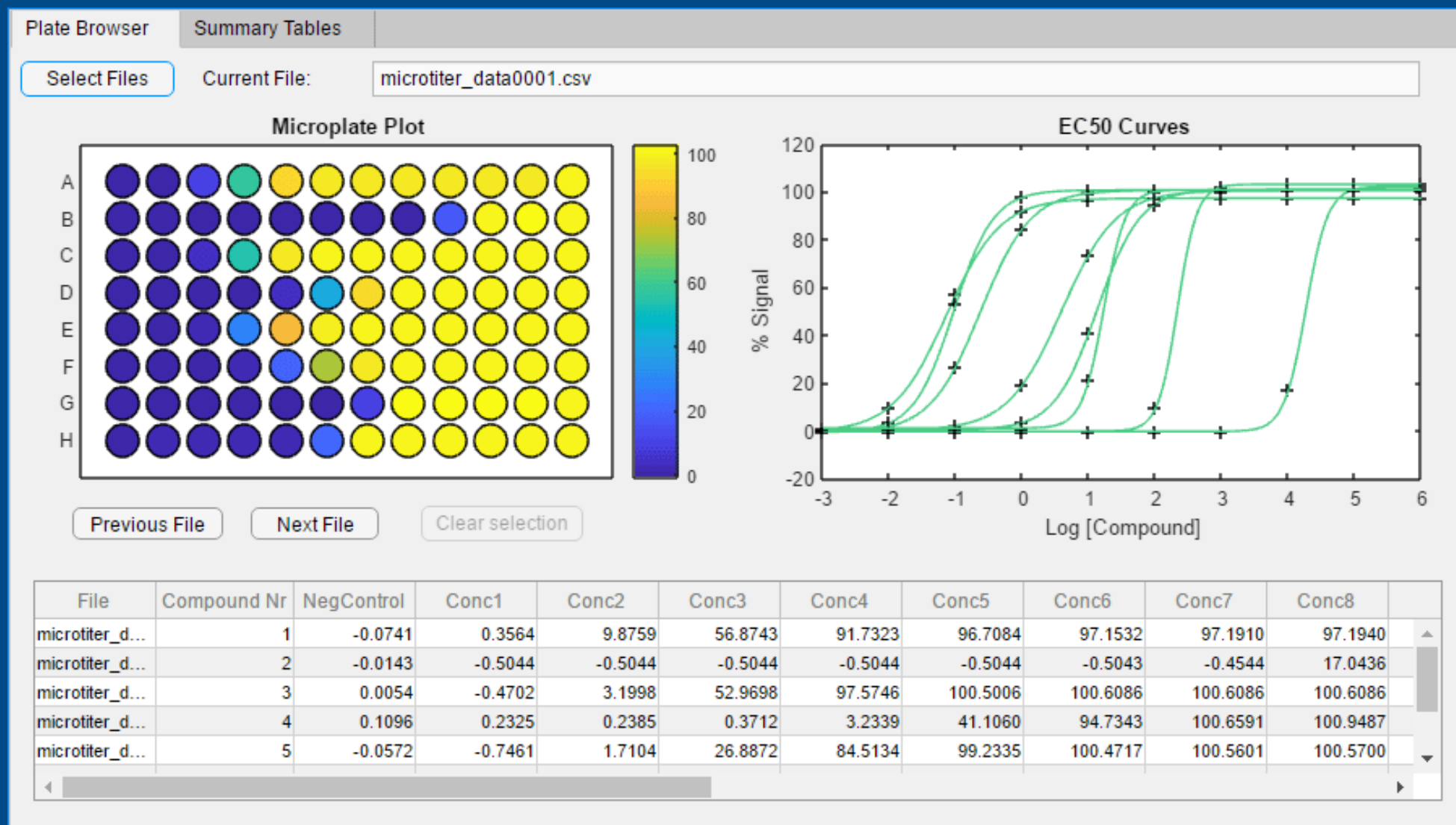
```
idx = find(exoplanets.st_distance == min(exoplanets.st_distance));
name = char(exoplanets(idx,'st_name'));
```

Page 1 of 7 1468 words

Sharing Live Scripts

The screenshot displays the MATLAB Live Editor window for a file named 'CompressibilityFactor.mlx'. The interface includes a menu bar with 'LIVE EDITOR', 'INSERT', and 'VIEW' tabs. Below the menu is a toolbar with various icons for file operations (New, Open, Save, Find Files, Compare, Print), navigation (Go To, Find), text formatting (Normal, Bold, Italic, Underline, Monospace), code management (Code, Refactor), and execution (Run Section, Run and Advance, Run to End, Run, Step, Stop). The main workspace contains three interactive elements: a text input field labeled 'P' with the value '1:40', a slider labeled 'Slider' with a value of '350', and a drop-down menu labeled 'Drop down' with the selected value '"carbon dioxide"'. Below these controls is a plot titled 'carbon dioxide @ 350 Kelvin'. The plot shows the 'Compressibility Factor, Z' on the y-axis (ranging from 0.92 to 1.0) versus an unlabeled x-axis. A single blue line starts at (0, 1.0) and decreases linearly to approximately (1.0, 0.91). A 'Hide Code' tooltip is visible on the right side of the plot area. The status bar at the bottom indicates 'Ln 5 Col 23'.

Creating Apps



Deploying Web Apps

The screenshot shows a web browser window with the URL `ah-dgarriso:9988/webapps/home/session.html?app=transientConduction`. The application interface is titled "MATLAB Web Apps" and "Transient Heat Conduction".

Initial and Boundary Conditions:

- Initial T (C): 10
- Top T (C): 0
- Bottom T (C): 50
- Left T (C): 25
- Right T (C): 25

Geometry:

- x (m): 0.05
- y (m): 0.05
- dx (m): 0.0025
- dy (m): 0.0025

Note: Numerical stability requires F...
Current Fo = 0.0608

Thermal Diffusivity:

- Alpha (m²/s): 1e-4
- Material: Air (selected)

Time and Convergence:

- dt (s): 0.01
- Total Time (s): 50
- Convergence Criterion: 1e-4

Buttons: Start, Stop

Time = 39 s

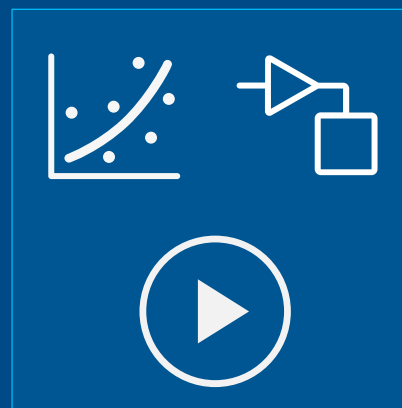
The top plot is a 2D contour plot of temperature distribution in the x-y plane. The x-axis ranges from 0 to 0.04 m, and the y-axis ranges from 0 to 0.05 m. The temperature is highest at the bottom (red) and lowest at the top (blue).

The bottom plot is a line graph of the center temperature T_{center} versus time t (s). The x-axis ranges from 0 to 50 s, and the y-axis ranges from 10 to 25. The temperature starts at 10°C at $t=0$ and rises to approximately 25°C by $t=40$ s.

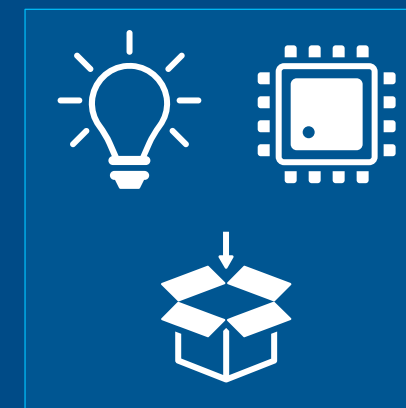
Using MATLAB & Simulink to Build Algorithms in Everything



Inputs



Design



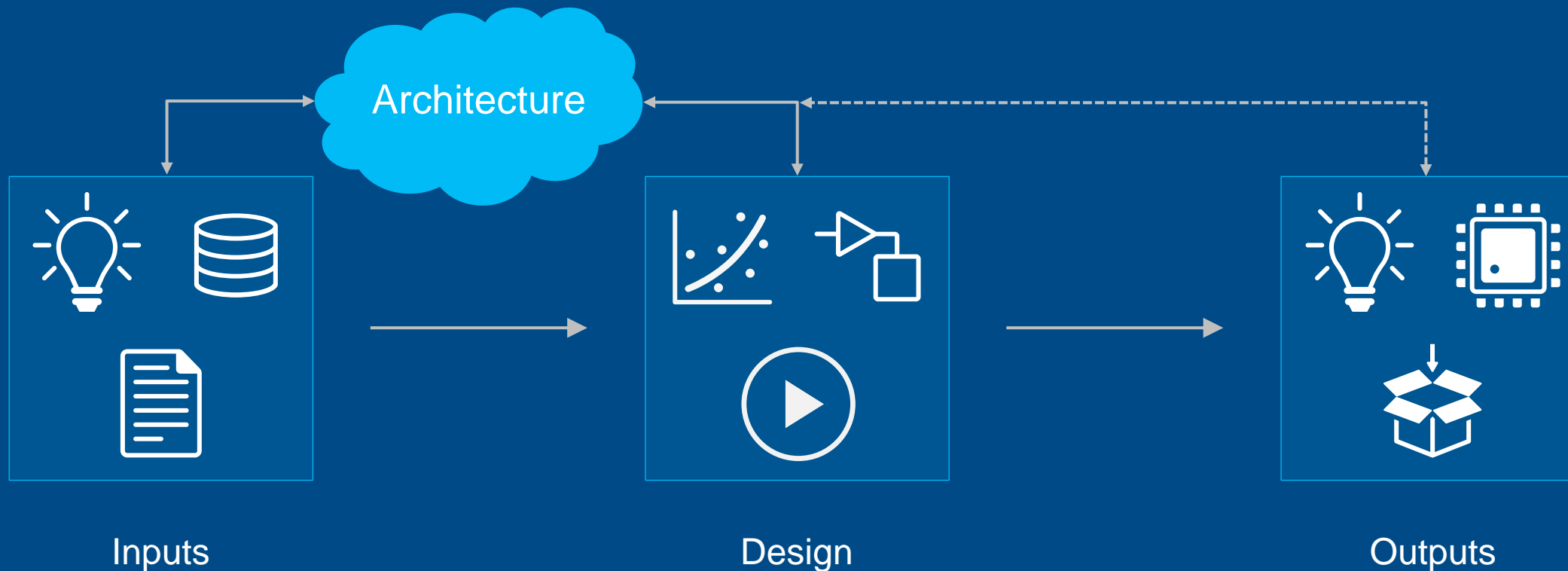
Outputs



MATLAB® & SIMULINK®



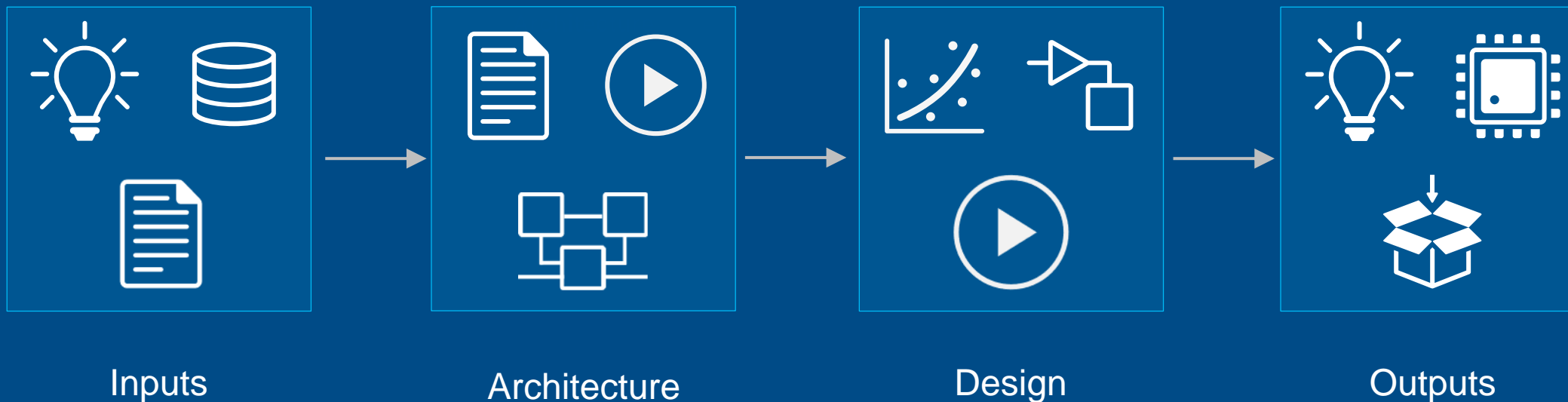
Evaluating Architectures



MATLAB® & SIMULINK®



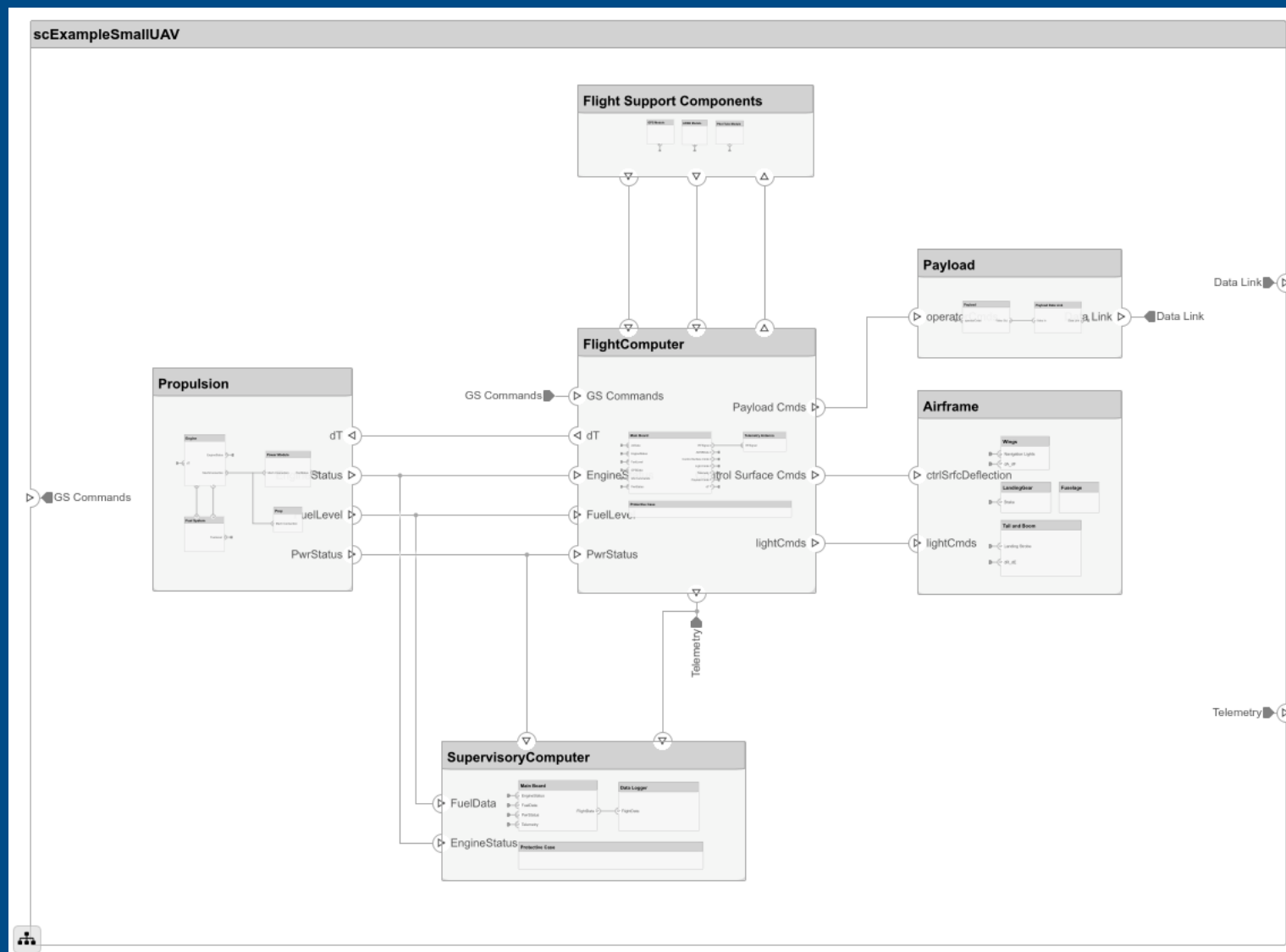
Evaluating Architectures



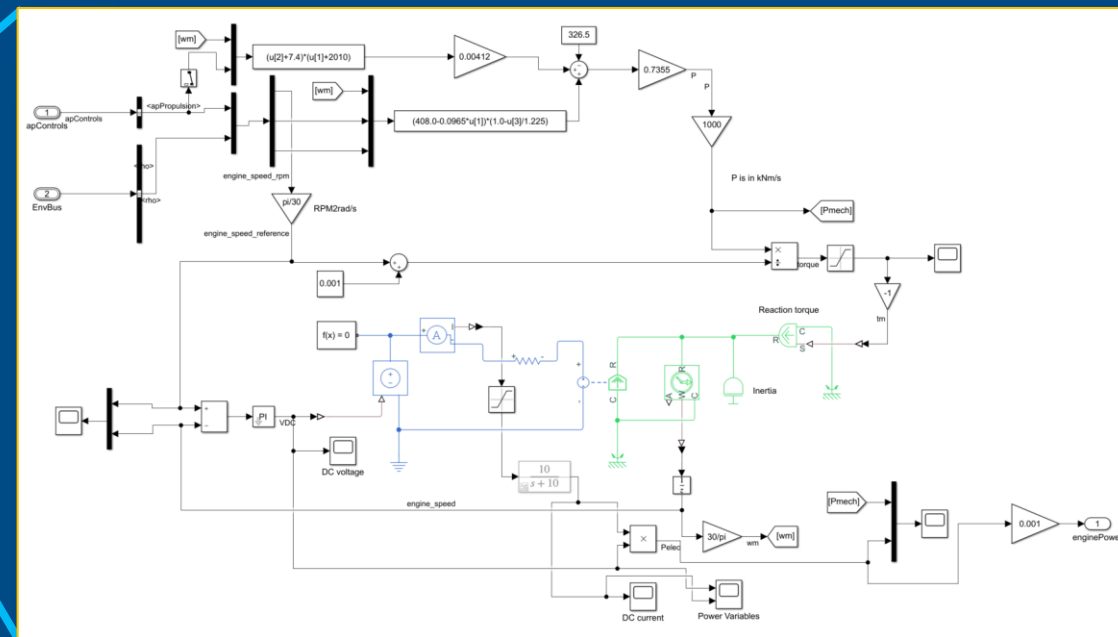
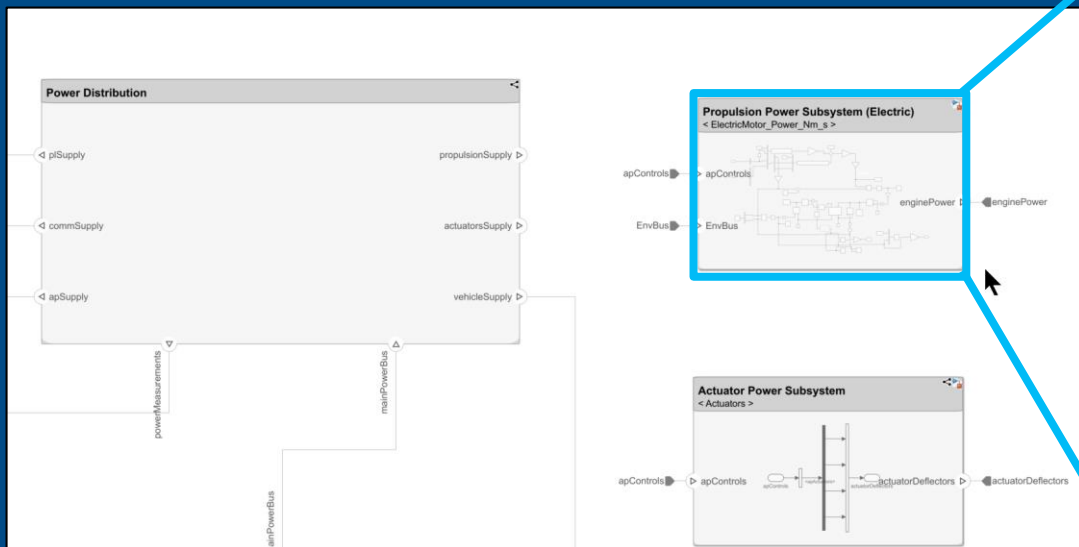
MATLAB® & SIMULINK®



Designing System and Software Architectures



Designing System and Software Architectures



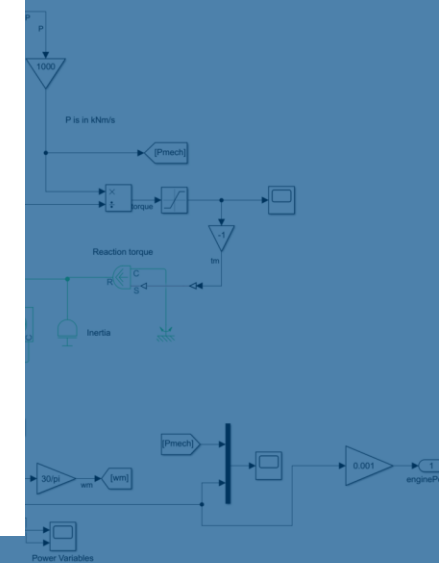
Designing System and Software Architectures



Find out more:

요구사항부터 아키텍처 설계와
시뮬레이션까지 시스템 엔지니어링을
위한 방안

제어 및 임베디드 시스템 트랙
류성연



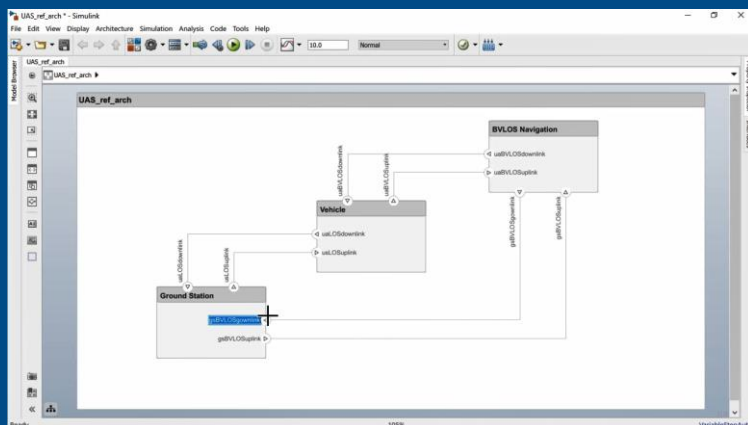
Designing **Beyond** System and Software Architectures



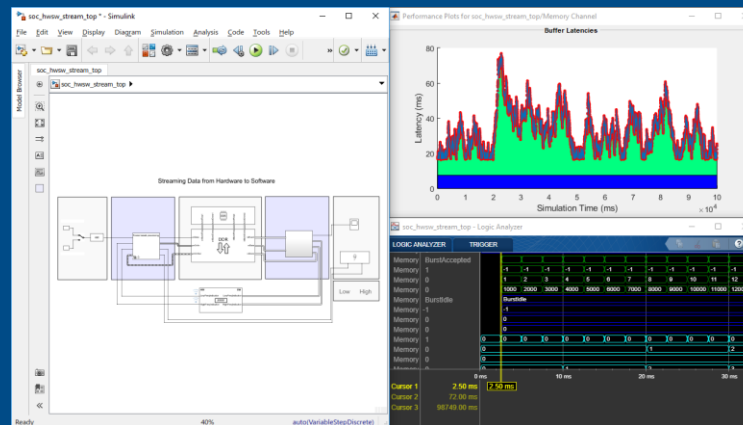
Systems and Software

SoC Hardware and Software

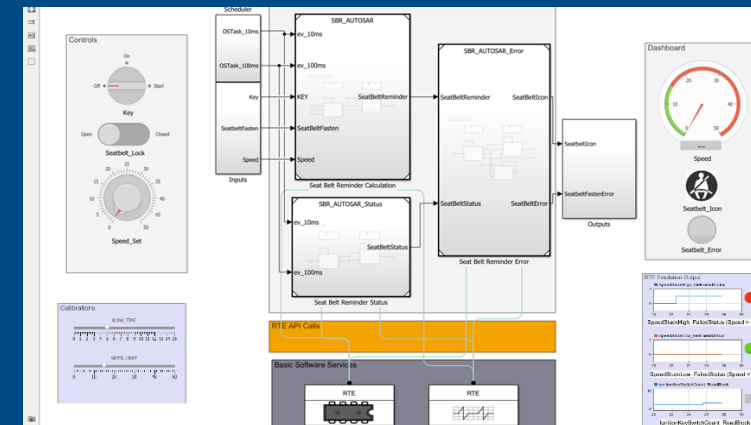
AUTOSAR Software



System Composer

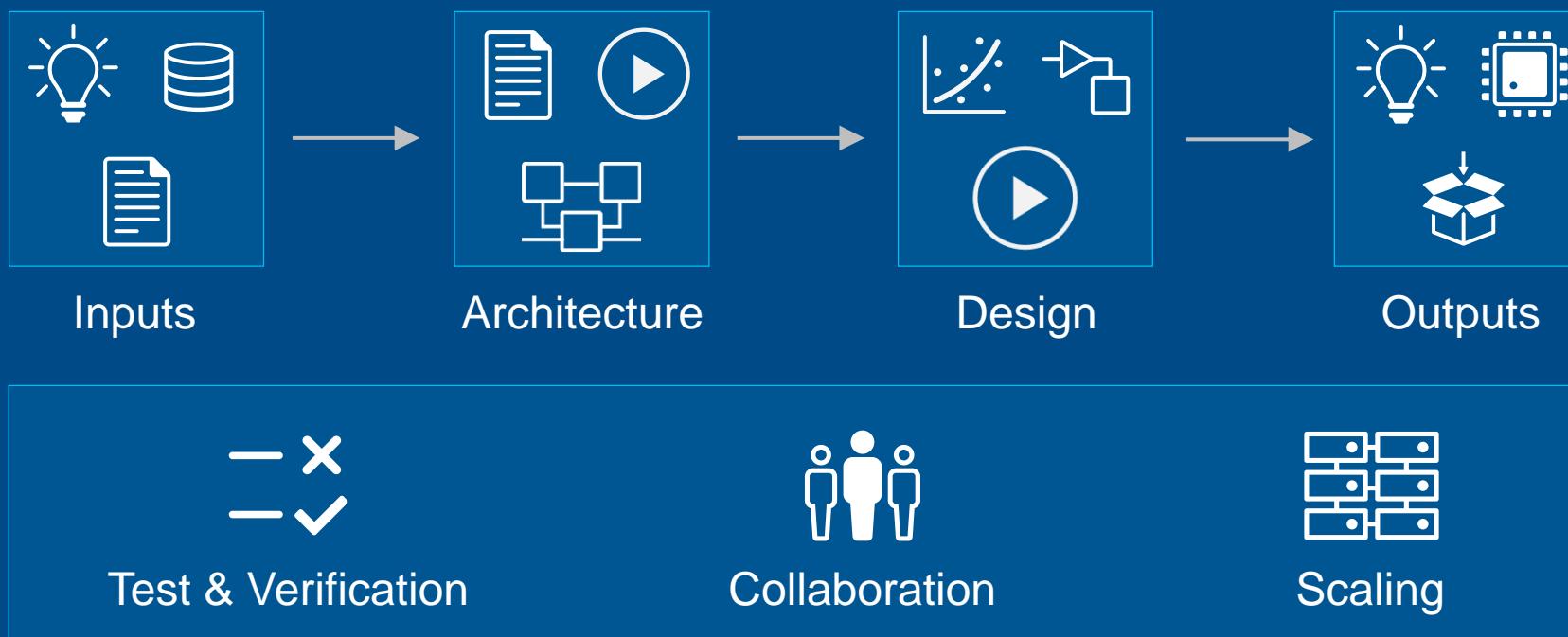


SoC Blockset



AUTOSAR Blockset

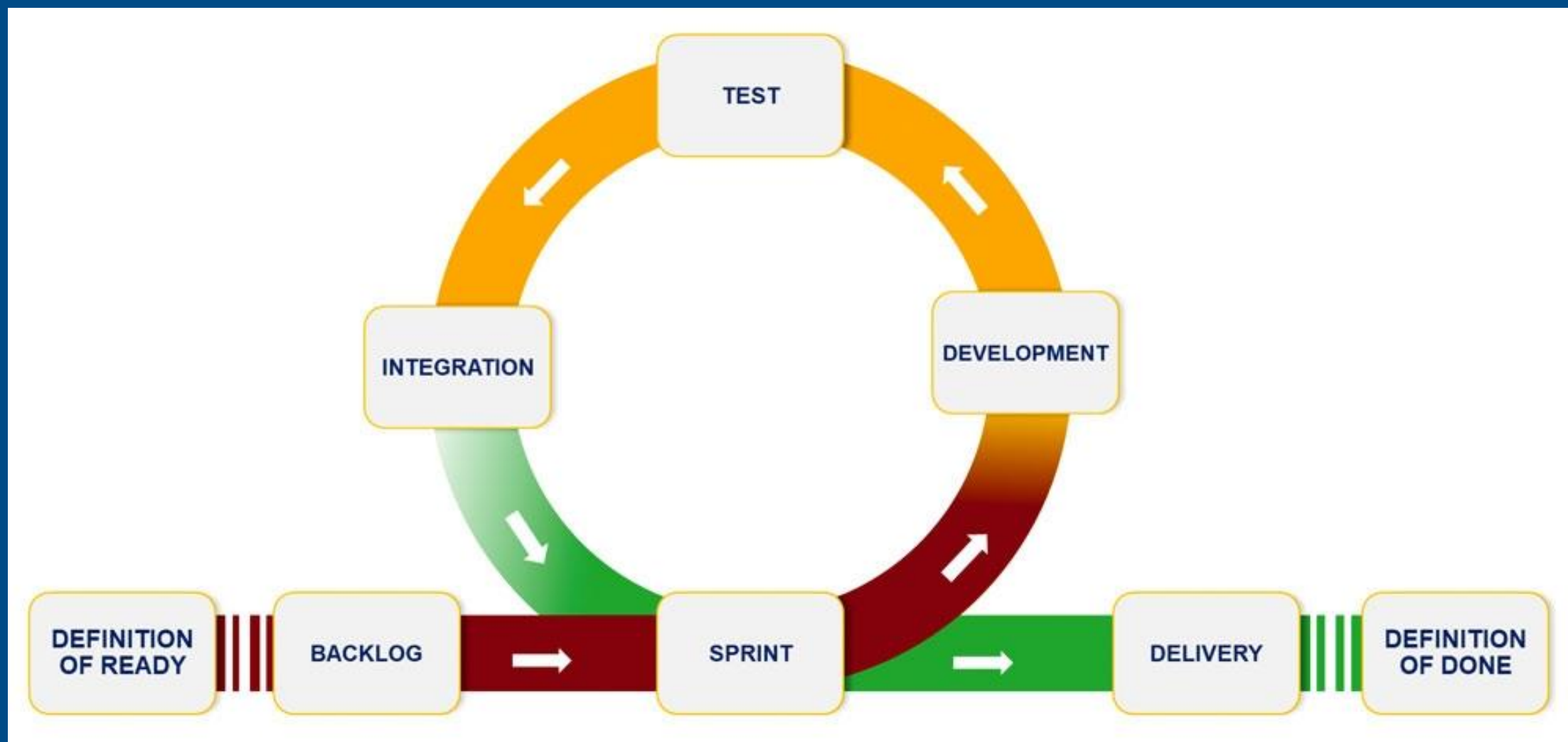
Using MATLAB & Simulink to Build Algorithms in Everything



MATLAB® & SIMULINK®



Using MATLAB & Simulink to Build Algorithms in Everything



Integrating with Third-party Requirements Tools



External Requirements





Requirements Management Tools


R2019a




ReqIF

Simulink Requirements

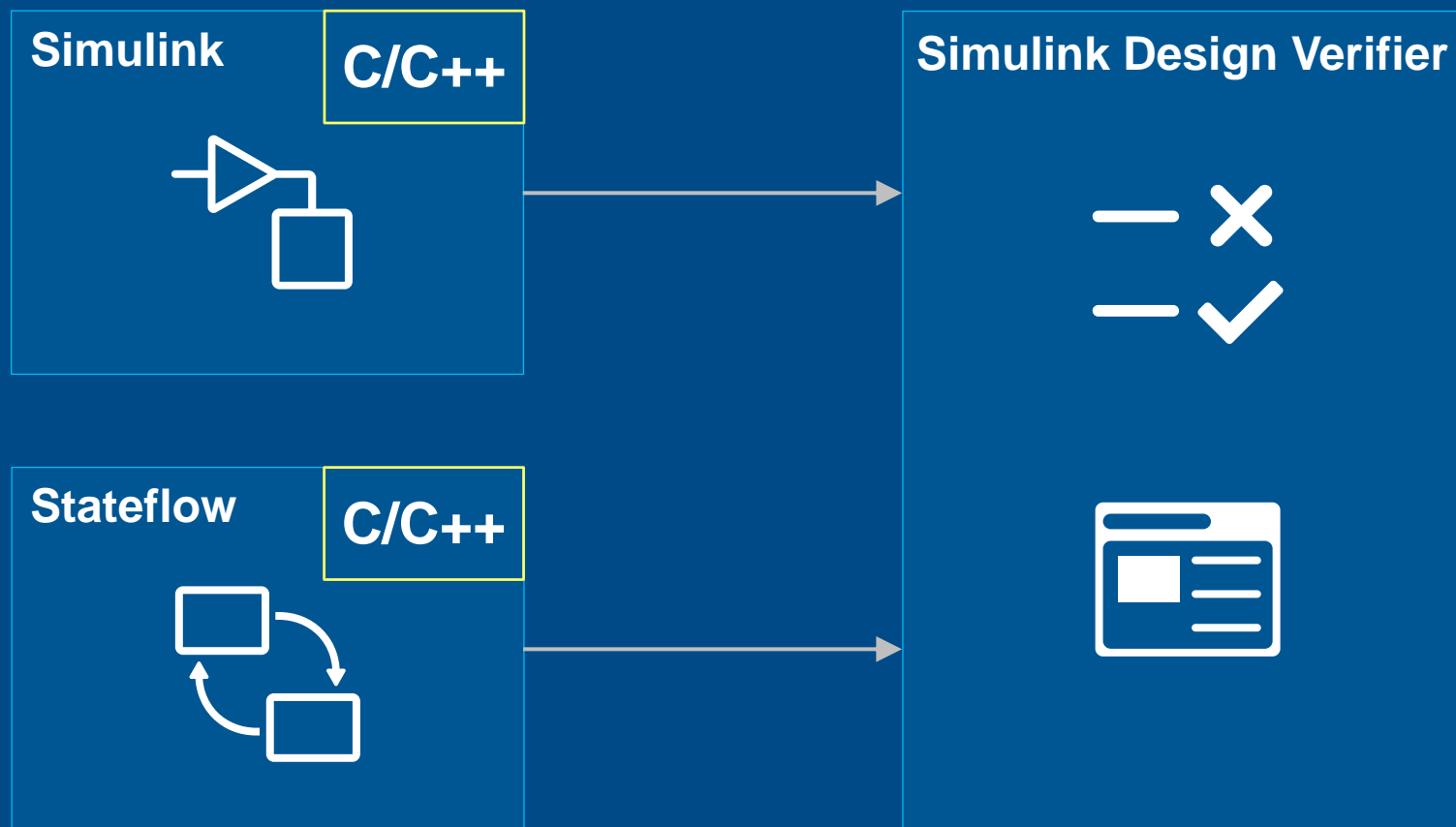
External Requirements



Authored Requirements



Include Custom Code in Test & Verification



Include Custom Code in Test & Verification



Simulink

C/C++

Simulink Design Verifier

Find out more:

모델기반설계를 이용한 요구사항 기반
검증의 단순화

제어 및 임베디드 시스템
홍혁기



Using the MATLAB Unit Test Framework



```
>> result.table
```

```
ans =
```

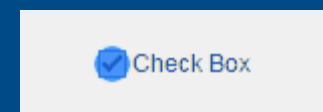
```
2x6 table
```

Name	Passed	Failed	Incomplete	Duration	Details
'test_Predictions/Test_ModelType'	true	false	false	0.12241	[1x1 struct]
'test_Predictions/Test_Prediction'	false	true	true	0.11542	[1x1 struct]

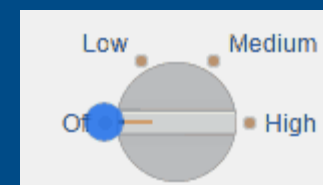
Using the MATLAB App Testing Framework



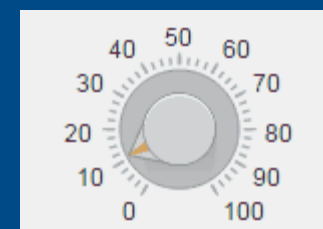
```
testCase.press(myApp.checkbox)
```



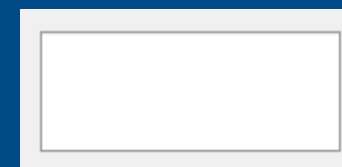
```
testCase.choose(myApp.discreteKnob, "Medium")
```



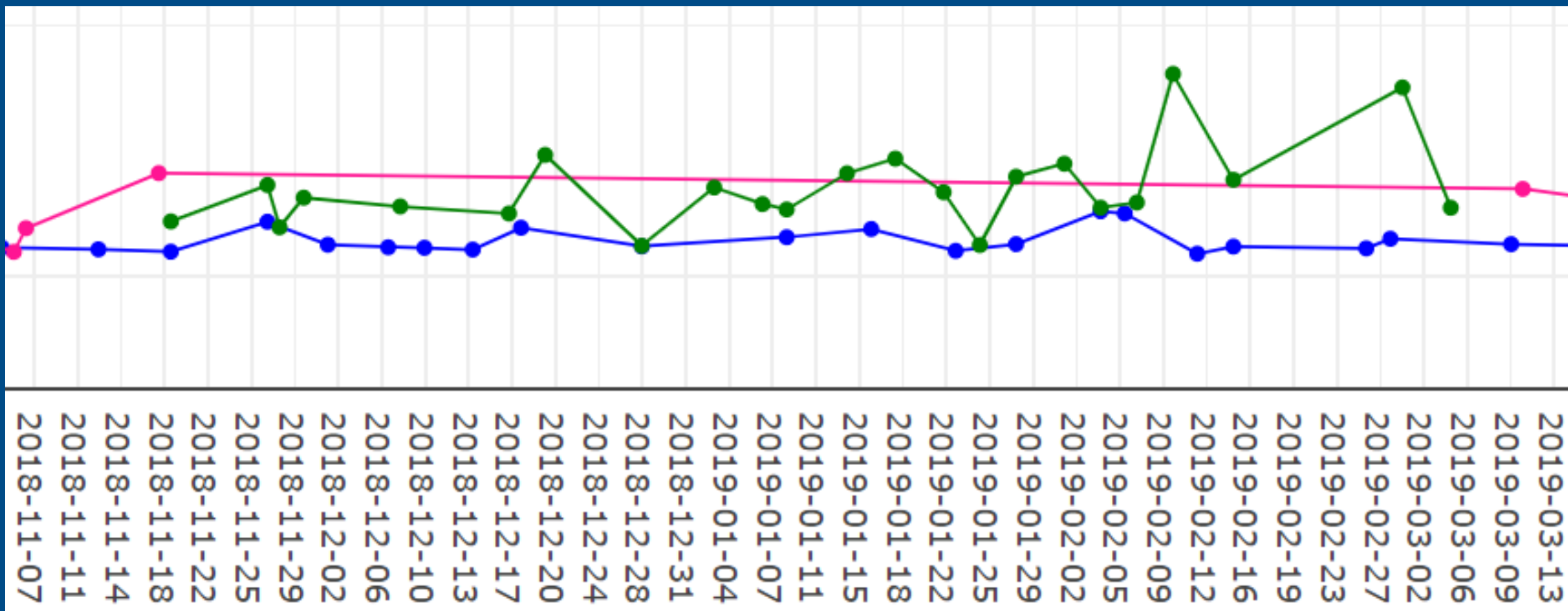
```
testCase.drag(myApp.continuousKnob, 10, 90)
```



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



Using the MATLAB Performance Testing Framework



Using Continuous Integration



Jenkins Blog Documentation Plugins Community Sub-projects About English Download

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

- MATLAB**
- MISRA Compliance Report
- Zoom
- VectorCAST Execution
- Klocwork Community
- jQuery
- Analysis Model API

MATLAB

Using Continuous Integration



Jenkins

Blog Documentation ▾

[← Find plugins](#)

MATLAB ^{1.0.0}

Minimum Jenkins requirement: 2.7.3
ID: matlab

Installs: No usage data available GitHub → Last released: 2 days ago	Maintainers MathWorks	Dependencies bouncycastle API v.2.16.0 (implied) (what's this?) Command Agent Launcher v.1.0 (implied) (what's this?) JDK Tool v.1.0 (implied) (what's this?) JAXB v.2.3.0 (implied) (what's this?)
--	--------------------------	---

The Jenkins plugin for MATLAB® enables you to easily run your MATLAB tests and generate test artifacts in formats such as JUnit, TAP, and Cobertura code coverage reports.

Features

- Support to run MATLAB tests, present in the Jenkins workspace automatically. (This also includes the tests present in .prj files)
- Generate tests artifacts in JUnit, TAP & Cobertura code coverage formats.
- Support to run tests, using custom MATLAB command or custom MATLAB script file.

Using Projects in MATLAB



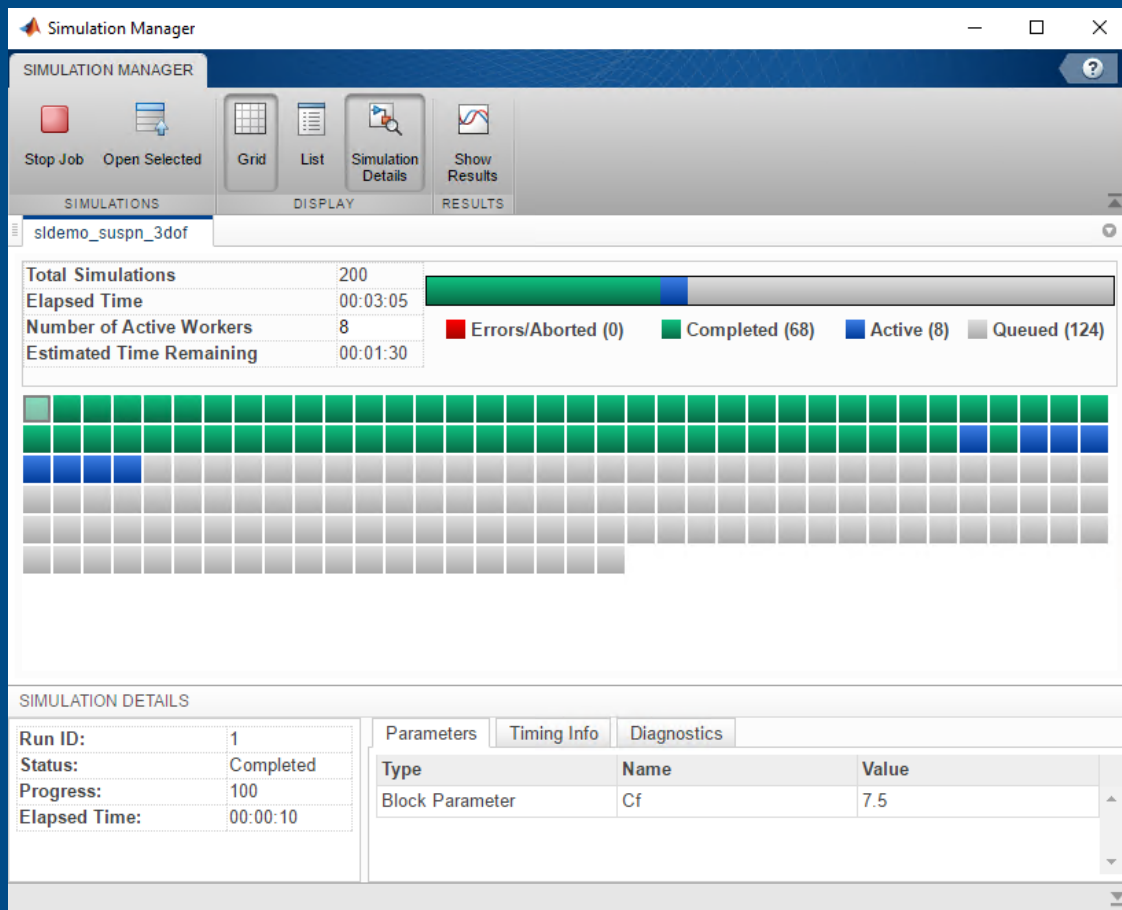
TOOLS		ENVIRONMENT		SOURCE CONTROL										
Search	Custom Tasks	Run Checks	References	Details	Project Path	Startup Shutdown	Git Details	Refresh	Commit	Fetch	Push	Pull	Remote	Branches

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	✓	●	

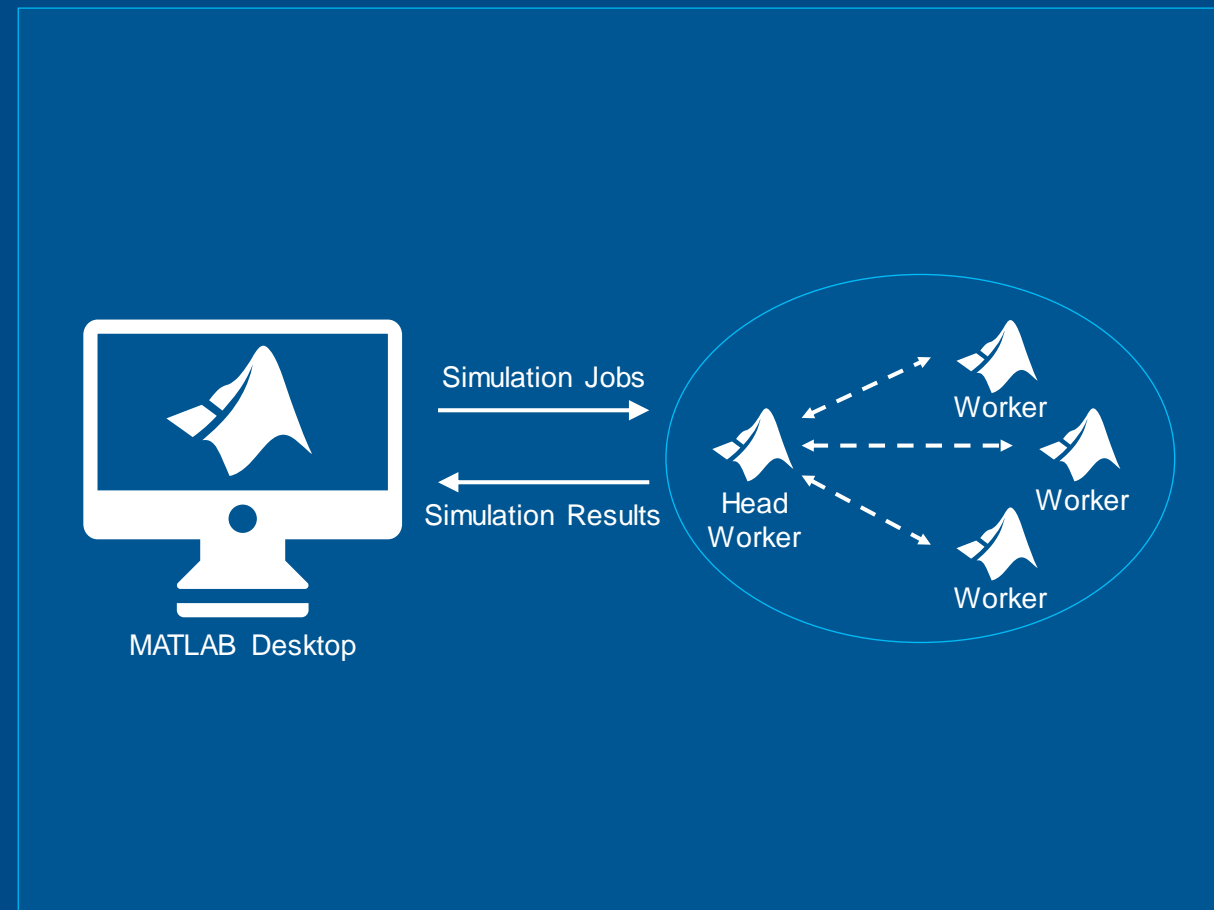
Parallel Simulations in Simulink



Simulation Manager



batchsim



Scaling Computations on Clusters and Clouds



MATLAB



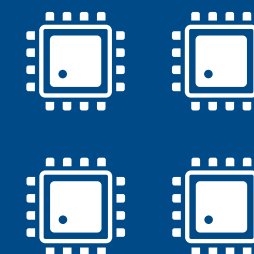
MATLAB Parallel Server



Cloud



GPU



Multi-core CPU

Parallel Computing Toolbox

Team-Based Collaboration for Code Verification and Review

- Web browser-based view of results directly in the code
- Navigation tools for investigating code analysis and proving results
- Ability to triage, assign, and justify code analysis results
- Create and assign tickets in bug-tracking systems such as Jira

The screenshot displays the Polyspace web interface. At the top, there's a navigation bar with tabs for Dashboard, Run-time Checks, Defects, Coding Standards, Code Metrics, and Global Variables. Below this is a filter section with 'Showing: 109 / 395' and filters for 'In Progress', 'OR To Do', 'AND', and 'Unassigned'. The main area is divided into a 'Results List' table and a 'Result Details' panel.

Family	ID	Type	Group	Check	Information
• *	27366	Red Check	Static memory	Illegally dereferenced p...	
• *	27431	Red Check	Other	Invalid use of standard l...	
• *	27514	Red Check	Control flow	Non-terminating loop	
• *	27529	Red Check	Static memory	Out of bounds array index	
• *	27673	Red Check	Control flow	Non-terminating call	
× *	27362	Gray Check	Data flow	Unreachable code	
× *	27455	Gray Check	Data flow	Unreachable code	
× *	27509	Gray Check	Data flow	Unreachable code	
× *	27553	Gray Check	Data flow	Unreachable code	
× *	27595	Gray Check	Data flow	Unreachable code	
× *	27675	Gray Check	Data flow	Unreachable code	
? *	27371	Orange Check	Static memory	Illegally dereferenced p...	Origin: Pos
? *	27398	Orange Check	Numerical	Division by zero	
? *	27410	Orange Check	Numerical	Overflow	Origin: Pos
? *	27413	Orange Check	Numerical	Overflow	Origin: Pos
? *	27417	Orange Check	Numerical	Overflow	Origin: Pos
? *	27425	Orange Check	Numerical	Overflow	Origin: Pos
? *	27427	Orange Check	Data flow	Non-initialized local vari...	Origin: Pos
? *	27429	Orange Check	Other	User assertion	Origin: Pos
? *	27454	Orange Check	Data flow	Non-initialized local vari...	Origin: Pat
? *	27502	Orange Check	Data flow	Non-initialized local vari...	Origin: Pos
? *	27503	Orange Check	Data flow	Non-initialized local vari...	Origin: Pos

The 'Result Details' panel shows an error: 'Illegally dereferenced pointer'. The error message states: 'Error: pointer is outside its bounds. Dereference of local pointer 'p' (pointer to int 32, size: 32 bits): Pointer is not null. Points to 4 bytes at offset 400 in buffer of 400 bytes, so is outside bounds. Pointer may point to variable or field of variable: "get_oil_pressure"'. Below the error, there's a 'Source Code' section showing a C code snippet:

```

example.c
...
97
98
99
100 if (get_bus_status() >= 0) {
101     if (get_oil_pressure() >= 0) {
102         *p = 5; /* Out of bounds */
103     } else {
104         1++;
105     }
106 }
107 i = get_bus_status();
108
109 if (i >= 0) {*(p + i) = 10;}
110
111 if ((0 <= i) && (i <= 100)) {
112     p = p - i;
113     *p = 5; /* Safe pointer access */

```

At the bottom of the interface, there are buttons for 'Track issue' and 'Create Ticket', which are circled in blue in the image.

Team-Based Collaboration for Code Verification and Review

- Web browser-based view of results directly in the browser
- Navigation tools for investigating code and improving results
- Ability to triage, justify code analysis
- Create and assign tickets in bug-tracking systems such as Jira

Find out more:

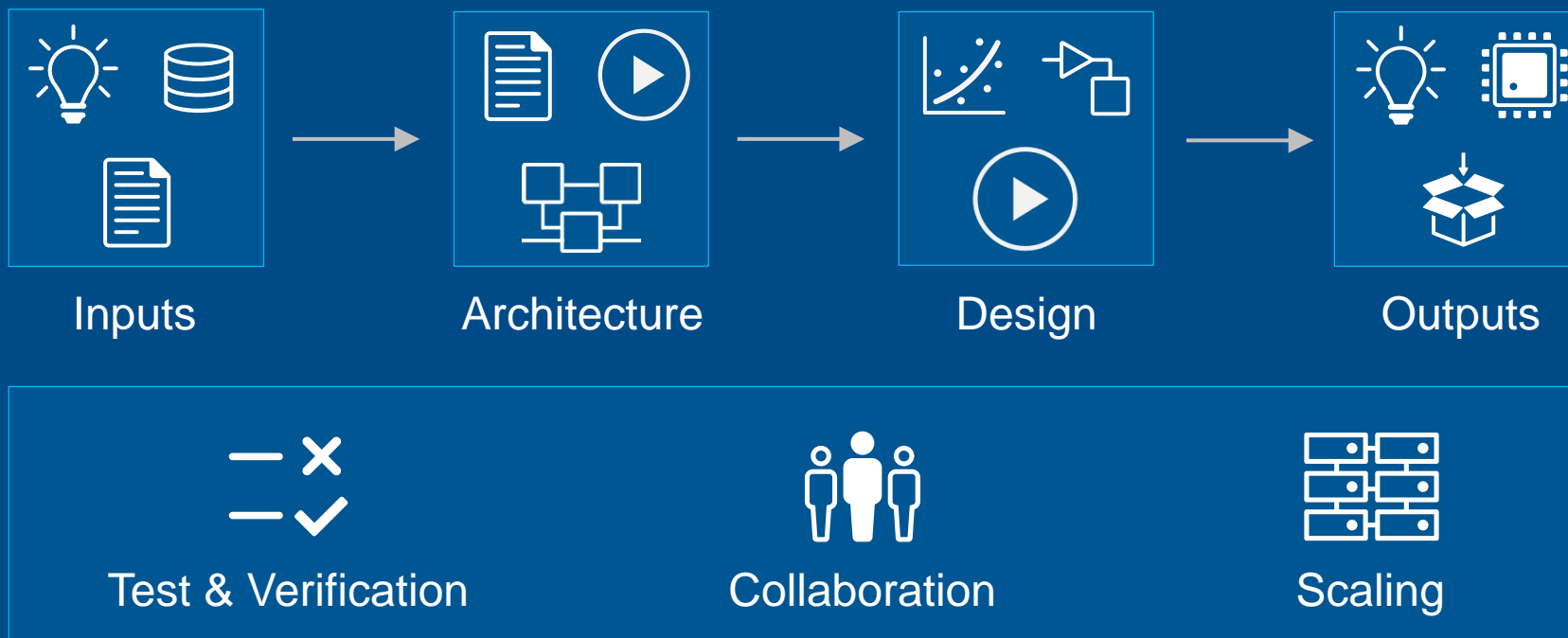
팀 협업을 통한 소프트웨어 안전성 및
보안성 확보 방안

제어 및 임베디드 시스템 트랙
Jay Abraham



Issue ID	Severity	Category	Description	Origin
27427	Orange Check	Data flow	Non-initialized local vari...	Origin: Pos
27429	Orange Check	Other	User assertion	Origin: Pos
27454	Orange Check	Data flow	Non-initialized local vari...	Origin: Pat
27502	Orange Check	Data flow	Non-initialized local vari...	Origin: Pos
27503	Orange Check	Data flow	Non-initialized local vari...	Origin: Pos

Using MATLAB & Simulink to Build Algorithms in Everything



MATLAB® & SIMULINK®



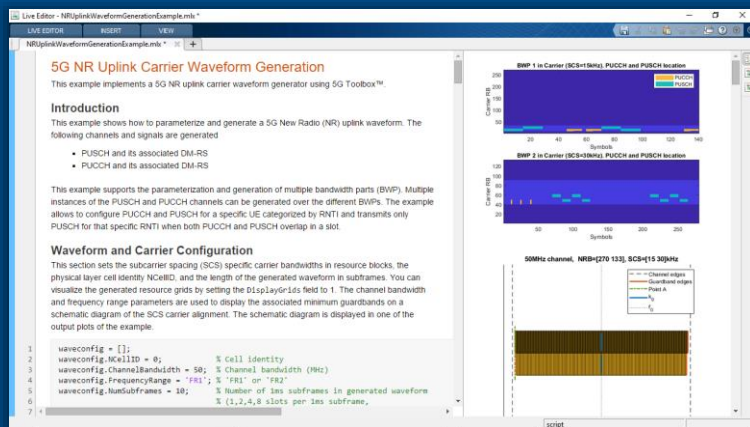
Specialized Tools for Building Algorithms in Everything



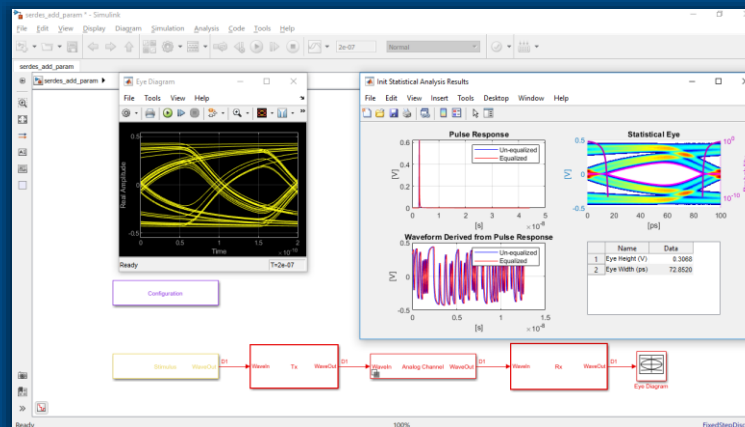
Communications

Physical interconnects

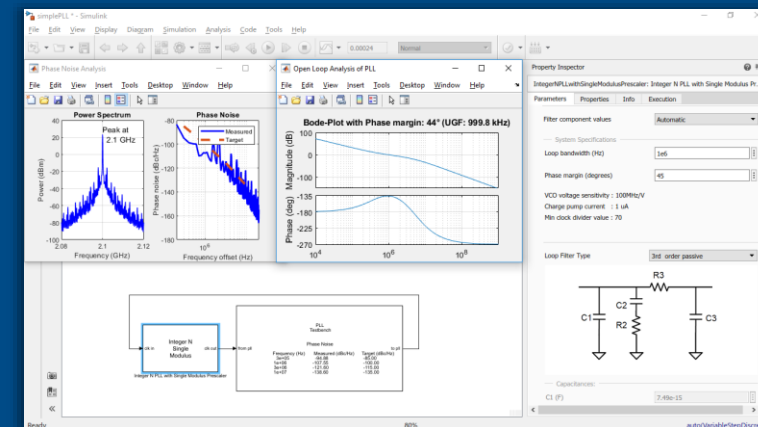
Analog Mixed-Signal



5G Toolbox

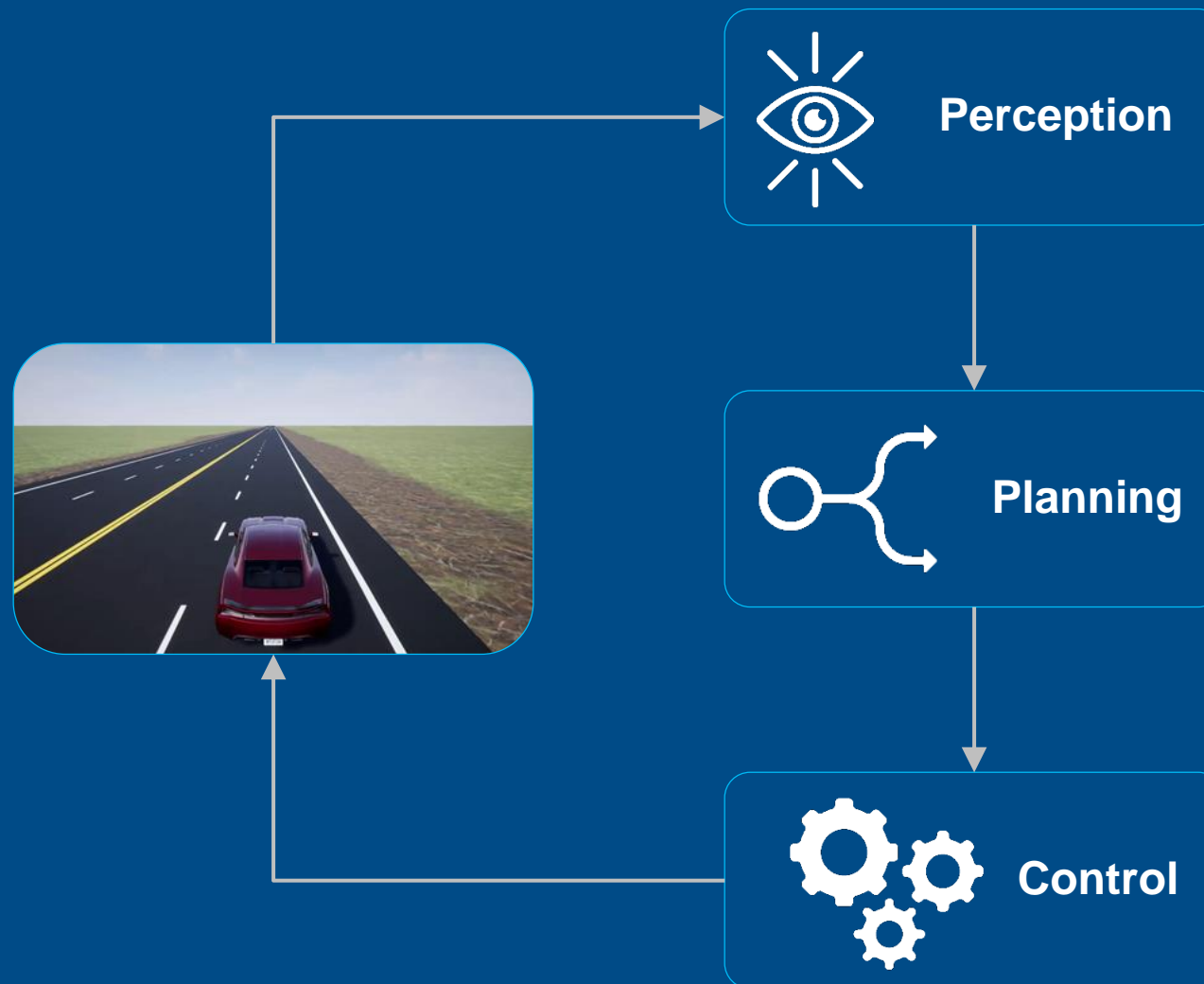


SerDes Toolbox

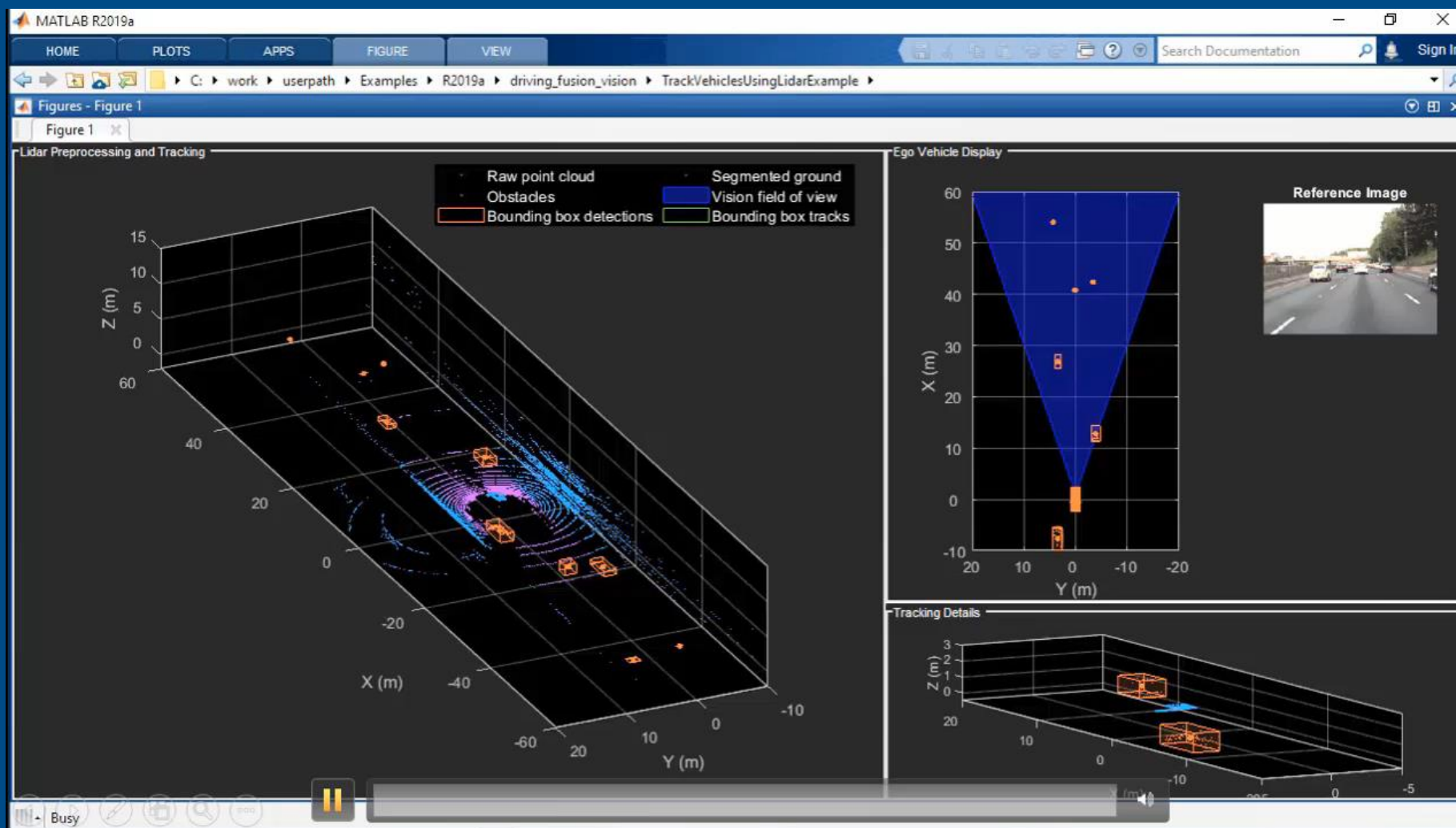
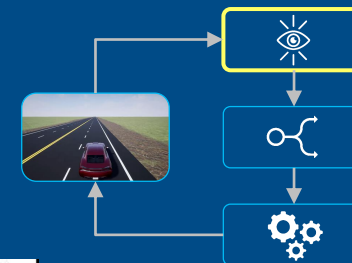


Mixed-Signal Blockset

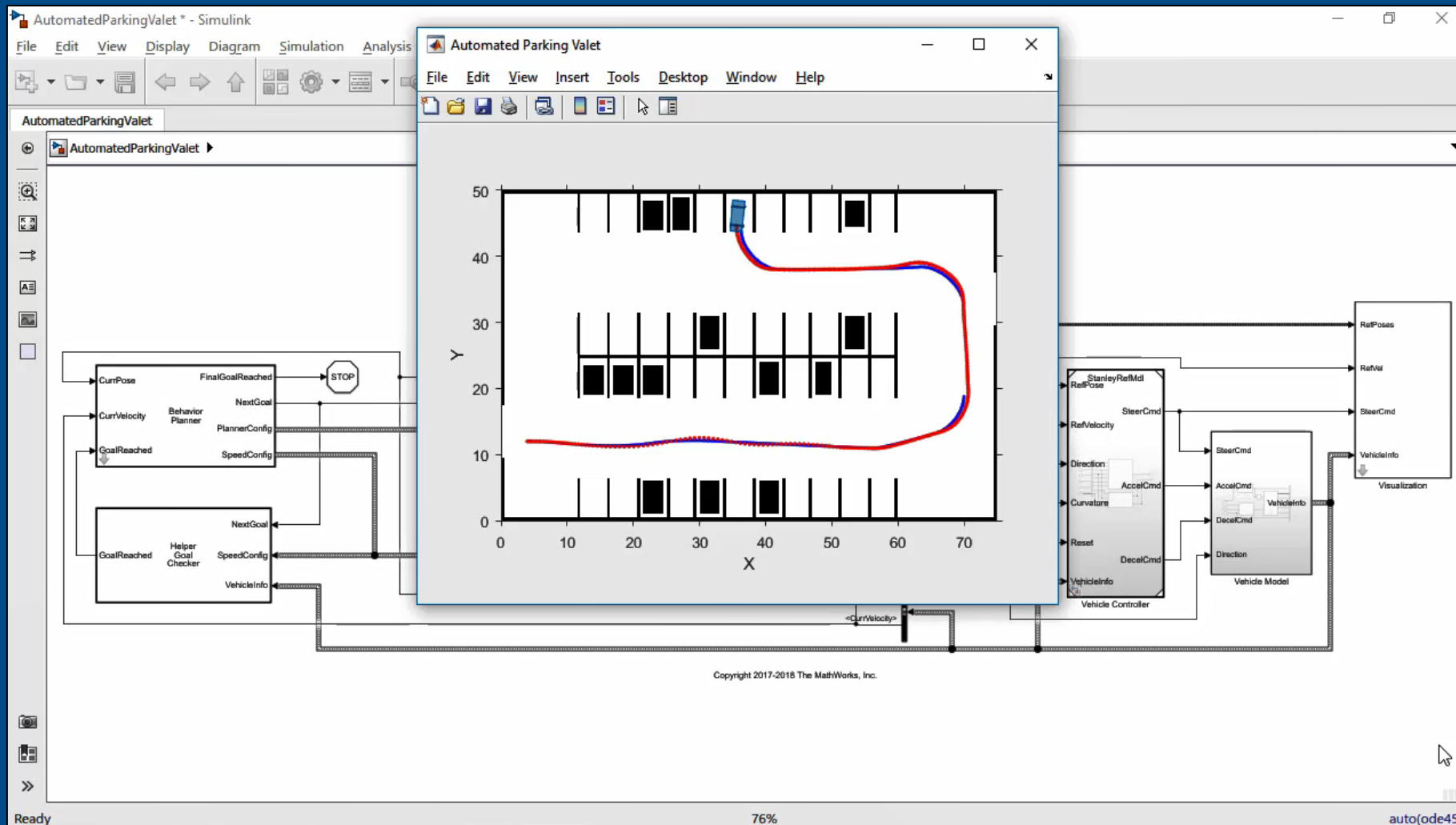
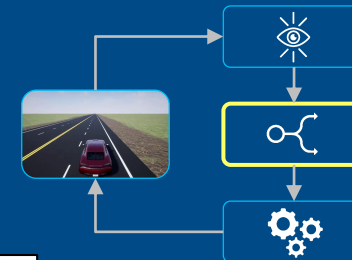
Developing Autonomous Systems



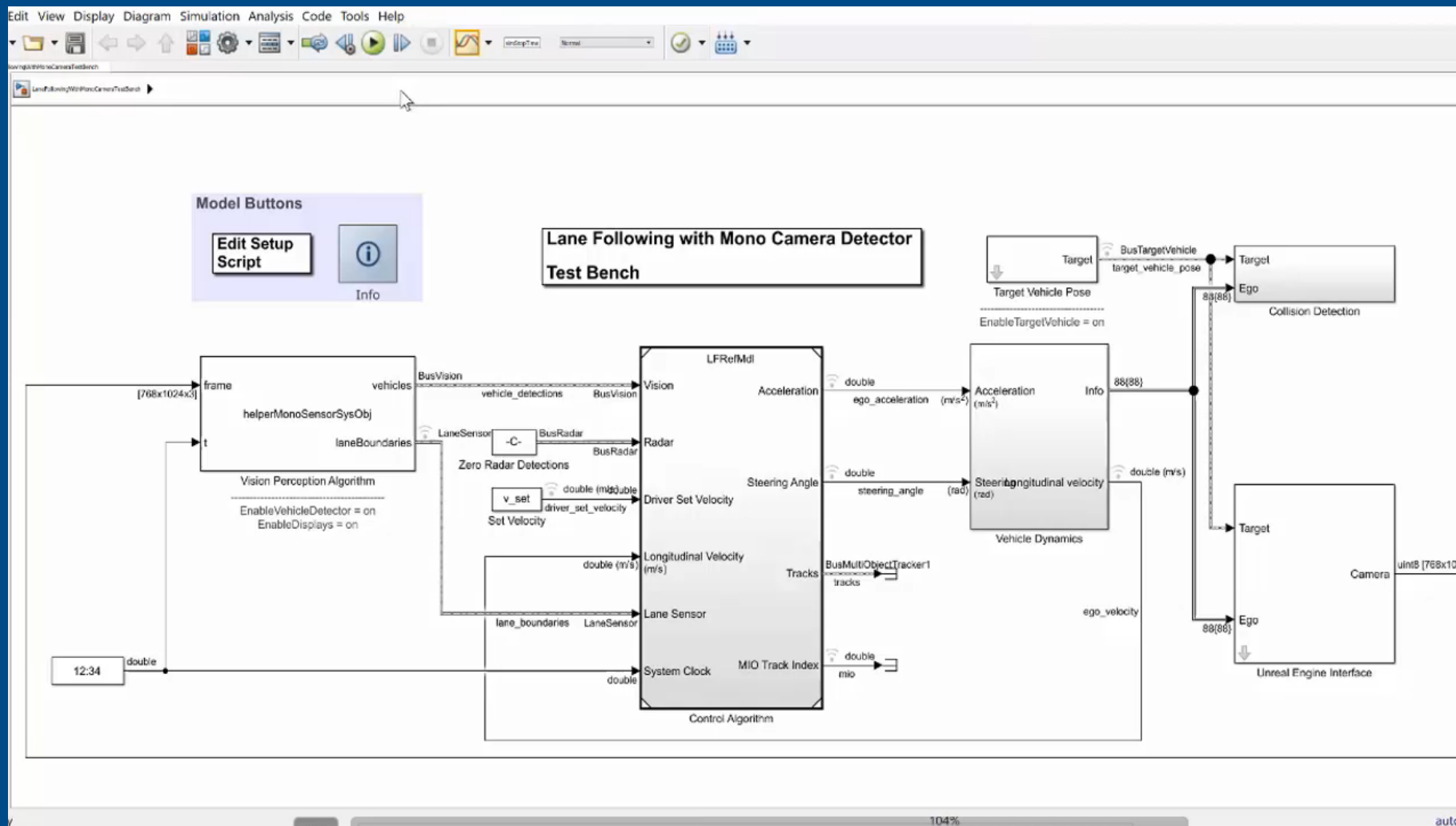
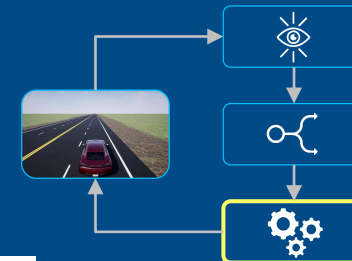
Evaluate Sensor Fusion Architectures



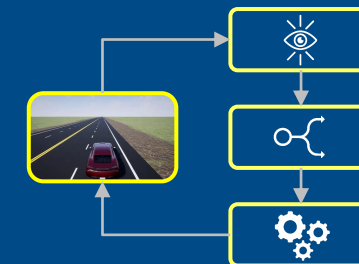
Simulate Path Planning Algorithms



Design Lane-following and Spacing Control Algorithms



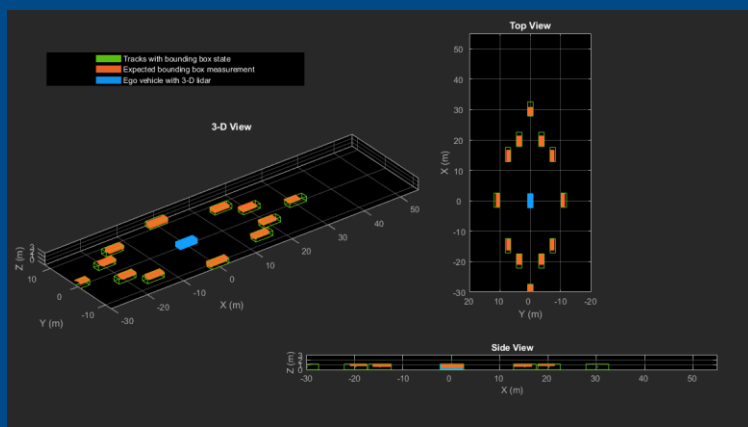
Developing Autonomous Systems



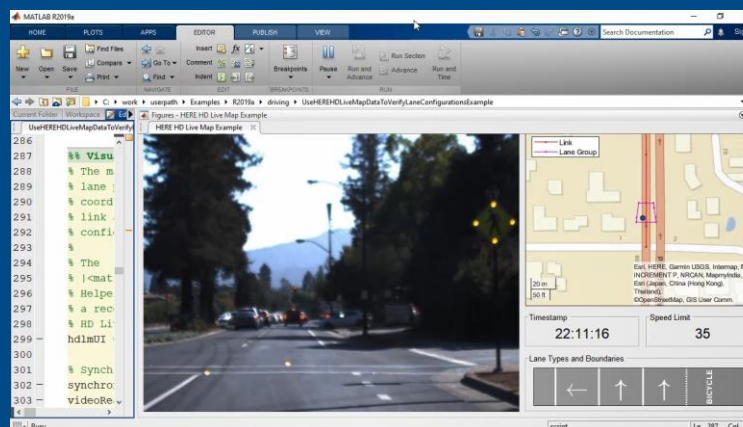
Lidar Processing
& Tracking

HERE HD Maps &
OpenDRIVE Roads

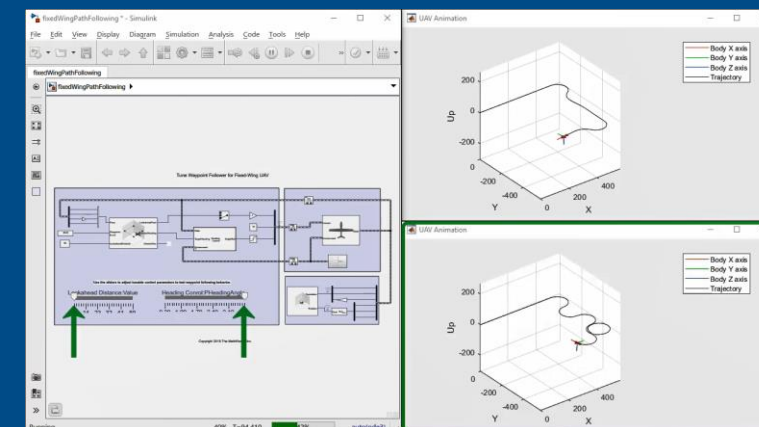
UAV Algorithms



Computer Vision Toolbox

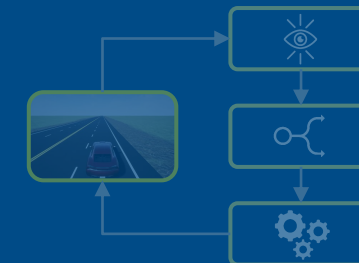


Automated Driving Toolbox

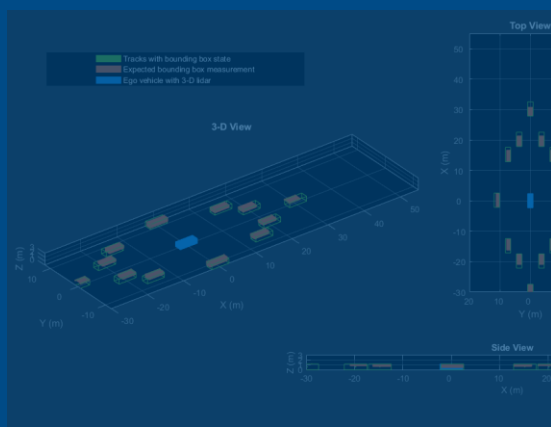


Robotics System Toolbox

Developing Autonomous Systems



Lidar Processing
& Tracking



Computer Vision Toolbox

HERE HD Maps &

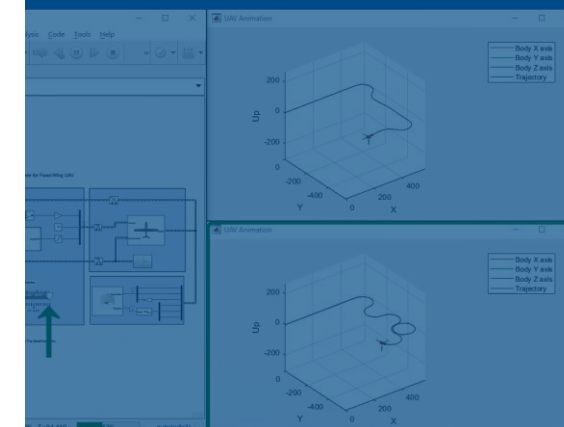
Find out more:
**MATLAB 및 Simulink를 이용한
 자율주행 시스템 설계 및 시뮬레이션**

제어 및 임베디드 시스템 트랙
 김종헌



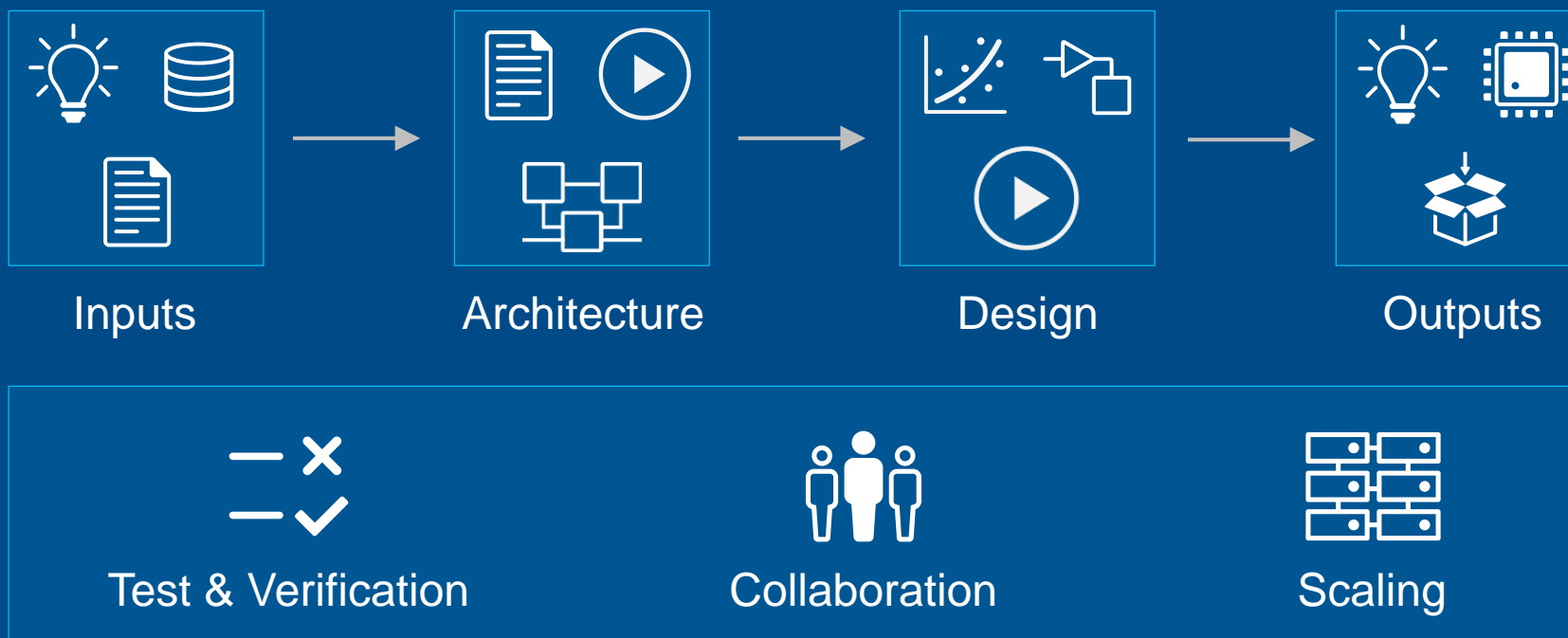
Automated Driving Toolbox

UAV Algorithms



Robotics System Toolbox

Using MATLAB & Simulink to Build Algorithms in Everything



MATLAB® & SIMULINK®



Attend Sessions this Afternoon

	엔지니어링 데이터 애널리틱스	인공지능과 딥러닝	제어 및 임베디드 시스템	센서 신호처리 및 무선기술	Tech Talk Special
11:40	인공지능 및 엔터프라이즈 환경 개발을 위한 빅데이터 개발 프레임워크 성호현 부장, MathWorks Korea	딥러닝과 강화학습 김종남 부장, MathWorks Korea	팀 협업을 통한 소프트웨어 안전성 및 보안성 확보 방안 Jay Abraham, MathWorks		
12:10	점심 식사				
13:10	APM기반의 자기진단: 반도체 공정용 건식 진공 펌프의 상태 진단의 새로운 지평 정완섭 박사, 한국표준과학연구원	GPU 기반 임베디드 딥러닝 코드 적용을 위한 MATLAB 솔루션 활용 김영태 팀장, SFA 스마트연구센터	Simscape모델의 HDL 변환을 통한 전력 전자 HIL 테스트 환경 Joel Van Sickle, MathWorks	5G New Radio 기본: 차세대 무선 통신 기술 이해하기 서기환 차장, MathWorks Korea	Simulink 기반 Legacy C/C++ Code 통합, 결과 시각화 및 검증 방안 유성재 부장, MathWorks Korea
13:40	휴식				
13:50	건정성 관리 예측 모델 개발을 위한 MATLAB 활용 방안 엄준상 차장, MathWorks Korea	신호 및 시계열 데이터를 위한 딥러닝 송완빈 대리, MathWorks Korea	모델기반설계를 이용한 FPGA 모터제어기 개발 적용 사례 나태용 선임연구원, LG전자 L&A 센터	MATLAB을 활용한 5G RF 중계기 TSYNC 모듈의 개발 김형우 책임연구원, (주)씨에스	에너지 최적화를 위한 에너지 관리 시스템(EMS) 강효석 차장, MathWorks Korea
14:20	휴식				
14:30	MATLAB을 이용한 원전기기 금속 이물질 충격신호 분석 문성인 박사, 한국원자력연구원	철도 궤도 결함 탐지를 위한 영역 기반 및 픽셀 기반 딥러닝 기법 적용 사례 황성호 선임연구원, 한국철도기술연구원	MATLAB 및 Simulink 를 이용한 자율주행 시스템 설계 및 시뮬레이션 김종현 부장, MathWorks Korea	자율시스템을 위한 센서 융합 및 추적 서기환 차장, MathWorks Korea	Simulink를 이용한 배터리 관리 시스템(Battery Management System) 개발 강효석 차장, MathWorks Korea
15:00	전시부스 관람				
15:40	제조 생산 현장에서 관리 시스템까지 빠른 인공지능 기반 시스템 구축 엄준상 차장, MathWorks Korea	임베디드 하드웨어로의 딥러닝 응용프로그램 배포 송완빈 대리, MathWorks Korea	모델기반설계를 이용한 요구사항 기반 검증의 단순화 홍혁기 부장, MathWorks Korea	FPGA, ASIC, SoC 개발을 위한 모델 기반 설계 도입 방안 정승혁 과장, MathWorks Korea	Powertrain Blockset을 이용한 전동식 파워트레인 설계 및 검증 방안 류성연 차장, MathWorks Korea
16:10	휴식				
16:20	MATLAB을 활용한 머신러닝 기반 가상발전소 운영 시스템 구축 백승엽 대표, 브이젠㈜	MATLAB을 사용한 영화에서 사용되는 색채심리학 분류 한영수 박사, 한국외국어대학교	요구사항 부터 아키텍처 설계와 시뮬레이션까지 시스템 엔지니어링을 위한 방안 류성연 차장	SoC Blockset 소개 정승혁 과장, MathWorks Korea	간편해진 C/C++코드 생성 설정 방법 소개 유재홍 부장, MathWorks Korea
16:50	경품 추첨 및 맺음말				

Read the Release Notes

R2019a at a Glance

Search MathWorks.com



Explore What's New

Get more out of MATLAB and Simulink by downloading the latest release.

[Download release now](#)



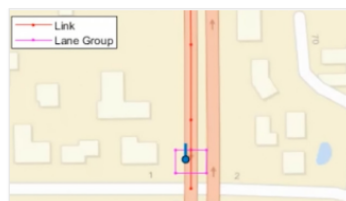
Release Highlights



Deep Learning

Develop controllers and decision making systems using reinforcement learning, train deep learning models on NVIDIA DGX and cloud platforms, and apply deep learning to 3-D data.

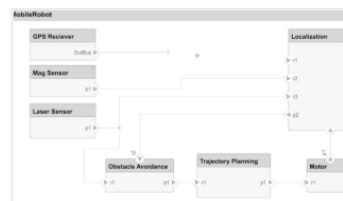
» [Learn more](#)



Automotive

Design and simulate AUTOSAR software, interface with HERE HD maps, and generate energy balance reports.

» [Learn more](#)



Systems Engineering

Design and analyze system and software architectures with System Composer.

» [Learn more](#)

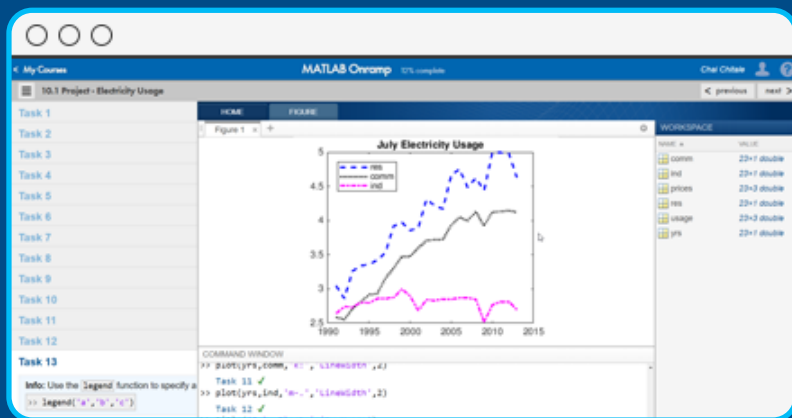


Projects

Use projects in MATLAB and Simulink to organize, manage, and share your work.

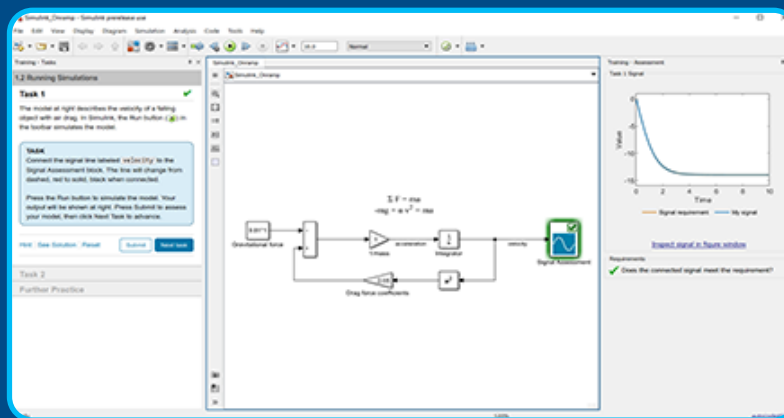
» [Learn more](#)

Get Started



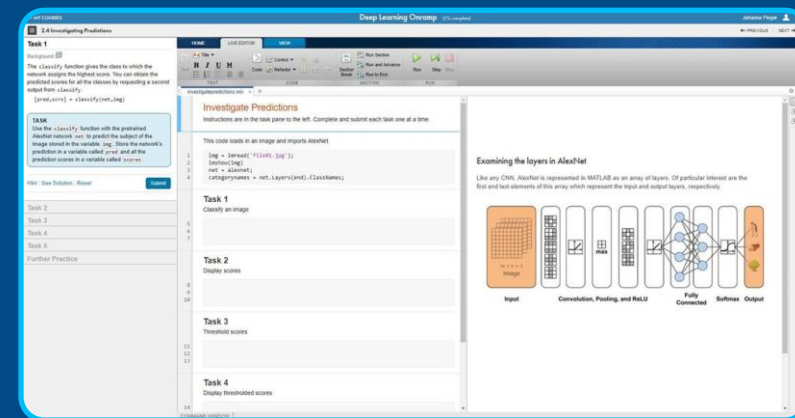
MATLAB Onramp

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

데모 부스와 상담부스로 질문 하시기 바랍니다.

감사합니다

