MATLAB EXPO 2016

What’s New in MATLAB and Simulink

Fabrizio Sara
Engineers and scientists...
Engineers and scientists…

Develop algorithms

write MATLAB code.

Analyze data
Engineers and scientists...

deploy algorithms and applications within web, enterprise, and production systems.

MATLAB EXPO 2016
Engineers and scientists... build Simulink models.

Model systems

Run simulations
Engineers and scientists…

combine MATLAB code and Simulink models together.
Engineers and scientists…

generate code.
Engineers and scientists…

connect software to hardware.
And it’s all easier to do in the latest releases.
Analysis and Visualization

Modeling and Simulation

Testing and Verification

Sharing and Collaboration

Performance
Analysis and Visualization

Modeling and Simulation

Testing and Verification

Sharing and Collaboration

Performance
MATLAB Live Editor

Change the way you work in MATLAB

- See results together with the code that produced them, accelerating exploratory programming and analysis
- Add equations, images, hyperlinks, and formatted text to create interactive narratives
- Create lectures that combine explanatory text, mathematical equations, code and results
Analysis and Visualization

Testing and Verification

Modeling and Simulation

Sharing and Collaboration

Performance
Deep Learning

Perform fast, accurate image classification

- Enables recognition workflows in autonomous robotics and ADAS
- Convolutional neural network (CNN) algorithm added to Neural Network Toolbox
- Uses cuDNN (a GPU-accelerated library from NVIDIA) (requires Parallel Computing Toolbox)
3D Vision

Enables autonomous systems to map and measure the world

- Supports workflows for ADAS, autonomous driving, and robotics
- New functionality to support:
  - 3D point cloud processing
  - Structure from motion
Get started or resume work faster by accessing templates, recent models, and featured examples

- Create new Simulink models using templates as starting points to common modeling approaches
- Use fully developed example models as a reference as you set out to build your own models
- Access most recent Simulink models right from the start page
Automatic Solver Option

Set up and simulate your model more quickly with automatically selected solver settings

- Simulink will select a solver and step size that is optimized for your specific model
- Considers factors such as model stiffness and simulation performance
- All new Simulink models use the automatic solver option
- Can optionally lock down solver so that it does not change from one simulation to another
Simulink Units

Specify, visualize, and check consistency of units on interfaces

- Specify physical units for Simulink signals and bus elements at the interfaces of components such as subsystems, model references, Stateflow charts and MATLAB function blocks

- Identify unit mismatches at the component interfaces

- Enforce consistency is by restricting the unit systems for certain components using the configuration parameter, ‘Allowed unit systems’
Messages

Model asynchronous operations in state charts using objects that carry data and can be queued

- New message object and queue
- Message Viewer block to visualize lifetime of a message
- Signal lines in Simulink to transfer messages between charts
Audio System Toolbox and WLAN System Toolbox

Design and test audio processing and WLAN (WiFi) communications systems

- Audio System Toolbox enables real-time audio processing in MATLAB and Simulink
- WLAN System Toolbox enables design and verification of evolving WLAN systems and WiFi devices
- Use together with LTE System Toolbox to design and test wireless systems
MATLAB Unit Testing Framework

Write and run unit tests, and analyze test results

- xUnit-style testing framework for the MATLAB language
- Includes a set of readily available qualification methods, and supports automation, providing easy reuse of test-cases
- Includes script-based, function-based, and object-oriented interfaces

function tests = solverTest
tests = functionTests(localFunctions);
end

function testRealSolution(testCase)
actSolution = quadraticSolver(1,-3,2);
expSolution = [-0.5; 2];
verifyEqual(TestCase, actSolution, expSolution);
end

%% Test Class Definition
classdef MyComponentTest < matlab.unittest.TestCase

%% Test Method Block
methods (Test)

%% Test Function
function testASolution(testCase)

%% Exercise function under test
% act - the value from the function under test

%% Verify using test qualification
% exp - your expected value
% testCase.<qualification method>(act,exp);
end
end
Simulink Test

Author, execute and manage simulation-based testing

- Build synchronized executable test environments
- Create inputs and assessments based on logic or temporal conditions
One-Click Display

Click a signal line when the simulation is running to view the current value

- Display port value for a signal by clicking it during simulation for easy debugging
- For bus signals, select the signals of interest before simulation
New Interface for Scopes

View and debug signals with cursors and measurements

- Scope, Floating Scope, and Viewers all upgraded with new UI
- Includes simulation data analysis and debugging tools
  - Cursors
  - Measurements
  - Triggers
Test Generation for C Code

Automatically generate tests for C code
S-functions

- Test generation automates a difficult task
- Generated tests lets you gain insight into the simulation of your design containing S-functions
Deploying to Hardware

Run your models on low-cost hardware and stream data into MATLAB

- Acquire images from Raspberry Pi and Kinect V2 into MATLAB and Simulink
- Run Simulink models on Lego EV3, Raspberry Pi 3, Raspberry Pi 3, and Arduino Yun
- Adds to existing support for Arduino, Lego, and Raspberry Pi platforms
Analysis and Visualization

Modeling and Simulation

Testing and Verification

Sharing and Collaboration

Performance
App Designer

Develop MATLAB applications with an enhanced design environment and expanded UI component set

- Choose from standard components (buttons, check boxes, panels, etc.), as well as gauges, lamps, knobs and switches
- Quickly move between visual design and code development
- New object-based code format makes it easier to share data between parts of the app
Add-On Explorer

Extend the capabilities of MATLAB by providing additional functionality for specific tasks and applications

- Browse, search, and install add-ons directly from MATLAB
- Add-ons include community-authored and MathWorks toolboxes, apps, functions, models, and hardware support
Using MATLAB with Other Languages

Integrate MATLAB with other programming languages, including C/C++, Java, .NET, and Python

- Call MATLAB from another language
- Reuse legacy code written in another programming language within MATLAB
- Package MATLAB programs into language-specific software components to integrate with other programming languages
  - Python support added in R2015b
Analysis and Visualization

Modeling and Simulation

Testing and Verification

Sharing and Collaboration

Performance
MATLAB Execution Engine

Redesigned execution engine runs MATLAB code faster

- All MATLAB code can now be JIT compiled
- Average performance improvement of 40% on 76 performance-sensitive user applications
- A platform for future improvements
- Performance testing framework R2016a
  - Measure MATLAB code performance
  - Interface leverages the unit testing framework