What’s New in Simulink

Ed Marquez – Simulink Product Marketing
Model and simulate your system
- Use one multi-domain environment
- Model the system under test and the plant
- Simulate closed-loop system behavior
Model and simulate your system
- Use one multi-domain environment
- Model the system under test and the plant
- Simulate closed-loop system behavior

Test early and often
- Test your system under all conditions
- Validate your design with real-time testing
- Trace from requirements to design to code
Core MathWorks Products

**Model and simulate your system**
- Use one multi-domain environment
- Model the system under test and the plant
- Simulate closed-loop system behavior

**Test early and often**
- Test your system under all conditions
- Validate your design with real-time testing
- Trace from requirements to design to code

**Automatically generate code**
- Generate production-quality C and HDL code
- Deploy directly to embedded processors or FPGA’s/ASIC’s
In this session, you will learn about new major Simulink capabilities to help you work more effectively and efficiently.
Simulink Toolstrip
Access and discover Simulink capabilities when you need them
Edit at the Speed of Thought

Locate ports on any side, in any order, on Subsystems, Subsystem References, Model References and Stateflow charts

Flexible port placement
Edit at the Speed of Thought

Use the keyboard to select any combination of blocks and annotations in a diagram

Flexible port placement

Keyboard shortcuts
Edit at the Speed of Thought
Drag or paste text into Simulink Editor canvas to create annotations

Flexible port placement

Keyboard shortcuts

Drag text to annotate
Edit at the Speed of Thought

Optionally resize blocks to fit parameter values

Flexible port placement

Keyboard shortcuts

Drag text to annotate

Fit block size to content
Edit at the Speed of Thought

Create a port by clicking or dragging the block outline

Flexible port placement

Keyboard shortcuts

Drag text to annotate

Fit block size to content

Automatic port creation
Edit at the Speed of Thought

Modify block parameters without opening a dialog box

Flexible port placement

Keyboard shortcuts

Drag text to annotate

Fit block size to content

Automatic port creation

Edit on block icon
Edit at the Speed of Thought

Improve the speed and accuracy of block parameter editing by selecting from variable or function names as you type

Flexible port placement

Keyboard shortcuts

Drag text to annotate

Fit block size to content

Automatic port creation

Edit on block icon

Block parameter autocomplete
Edit at the Speed of Thought

Open referenced models in the same window as their parent models

Flexible port placement

Keyboard shortcuts

Drag text to annotate

Fit block size to content

Automatic port creation

Edit on block icon

Block parameter autocomplete

Model reference in same window
Edit at the Speed of Thought

Connect a recommended block to an existing block in your model, sorted by frequency of use

Flexible port placement

Keyboard shortcuts

Drag text to annotate

Fit block size to content

Automatic port creation

Edit on block icon

Block parameter autocomplete

Model reference in same window

Predictive quick insert
Edit at the Speed of Thought

Highlight compatible ports as you draw a signal line

Port connection cues
Edit at the Speed of Thought
Get instant notifications about missing variables while editing your model

Port connection cues

Missing variable detection
Edit at the Speed of Thought

Rotate selected blocks as a group and evenly place blocks in suggested zones

... Port connection cues

Missing variable detection

Group rotation and smart guides
Edit at the Speed of Thought
Model Run-Time Software
Componentize Your Design
New and enhanced ways to **model software components**

Scoped Simulink Functions

Call Simulink Function blocks within a subsystem hierarchy

Create Simulink Functions that can cross model boundaries for reusable software components

>> slexMultiInstanceFunctionsExample
New and enhanced ways to **model software components**

Call a Simulink function from repeated executions (For Each subsystem)

Use Function Caller block inside a For Each subsystem block to call a utility

Utility can be defined in a Simulink function outside the subsystem (parent model)

Enables sending and receiving messages from within a For Each iteration

>> openExample('simulink/TemperatureControlSystemCommunicatingWithMessagesExample')
Simulink Messages **simplify software component integration**

Messages: Model and generate C++ code for software compositions with message-based communication

Integration of software components through middleware is an important and common task

Middleware uses messages (Send/Receive/Async Comm)

New library in Simulink

Messages in Stateflow
Can send Messages to model root
Simulink Messages **simplify software component integration**

Messages: Model and generate C++ code for software compositions with message-based communication

Stateflow and SimEvents work seamlessly with Messages

SimEvents used for routing, delaying, broadcasting, replication, gating

Stateflow used for handshaking, state management, logic
Simulink Messages **simplify software component integration**

Messages: Model and generate C++ code for software compositions with message-based communication

Examples

>> QueueSortingPoliciesModel

>> slexMessagePatterns
New interfaces to **visualize** and **control** component execution and communication

Schedule Editor: Explicitly schedule the execution of your model components
New interfaces to **visualize** and **control** component execution and communication

Sequence Viewer: Visualize function calls, message communication, and Stateflow state changes and event activity
Edit at the Speed of Thought

Model Run-Time Software

Componentize Your Design
Create an `.mlproj` that contains:
- `.slxp`
- Enumerated types
- Needed variables
- Data dictionaries
- Harness model
- ...
Sharing protected models is easy with the latest release

From: Oscar
To: Susan

Hi Susan,
The protected Simulink model and all dependencies are attached.

Good luck.

Supplier (Susan)

From: Susan
To: Oscar

Awesome!
I’m able to run it.

Create an .mlproj that contains:
- .slxp
- Enumerated types
- Needed variables
- Data dictionaries
- Harness model
- …

The recipient gets one file that contains everything needed to simulate the protected model
Projects

Organize, manage, and share your work

- Configure the project environment and manage files
- Share, reuse, upgrade
- Automate tasks, processes
- Componentize development
- Analyze dependencies, impact of changes
- Integrate with source control
Share, reuse, and upgrade your project files

**Project Compatibility:** Export a complete Project to a previous release with just one function

Additionally converts project metadata for previous versions
- Including "resources/project" → "SimulinkProject"

Accessed via same API as Simulink model export
- Clear alignment with Model-Based Design workflows

Supports 7 years of previous releases

Exports the currently loaded top-level project. Does not export referenced projects
Share, reuse, and upgrade your project files

Project Compatibility: Export a complete Project to a previous release with just one function
Integrate with source control

Source Control Integration

- Configure environment, manage files
- Automate tasks, processes
- Analyze dependencies, impact of changes
- Integrate with source control
- Componentize development
- Share, reuse, upgrade
Automatically merge branches that contain changes in different subsystems in the same SLX file.
The need for **simulation deployment**

- Enabling collaborators, suppliers and clients to access insights from simulation
- Leveraging simulations for in-operation usage, such as digital twin
- Interchanging with other simulation environments
Simulink Compiler is an out-of-the-box solution to share Simulink simulations

Support a variety of Simulink simulation features including variable-step solvers

Support flexible simulation inputs / parameter tuning

Royalty-free distribution
Simulink Compiler supports multiple deployment scenarios.

Share simulations as standalone executables, web apps, and Functional Mockup Units (FMUs).
Simulation deployment users play different roles in different scenarios

**Simulation Author:** They define, build, edit and compile Simulink simulations

**Simulation User:** They run, tune, and analyze the deployed simulations

**IT:** They support integrating deployed simulations with IT systems
Scenario 1

Package simulations as standalone desktop apps

Standalone Apps

- Runs on PC
- Can use App Designer GUI
- Needs local installation
Scenario 1

Package simulations as standalone desktop apps

Standalone Apps

Author
Simulation Author
Create simulation app with App Designer
App Designer

Package
Simulation Author
Use Simulink Compiler to package
Standalone application

Execute
Simulation User
.exe
MATLAB Runtime
.exe
MATLAB Runtime
.exe
MATLAB Runtime
.exe
Scenario 2

Package simulations as web apps

Runs on MATLAB Web App Server

Uses App Designer GUI

Browser-based access, no local installation needed
Scenario 2

Package simulations as web apps

Web Apps

Author
Simulation Author
Create simulation app with App Designer
App Designer

Package
Simulation Author
Use Simulink Compiler to package
MATLAB Web App

Host
IT Admin
Host on MATLAB Web App Server

Execute
Simulation User
Browser-based access
Scenario 3
Package simulations as a service API

Service APIs

Runs on MATLAB Production Server*

Supports customer developed client-server App and web apps (e.g. HTML/JavaScript)

Centrally hosted, no local installation needed
Scenario 3

Package simulations as a service API

**Service APIs**

**Author**

- Simulation Author

  Create simulation function in MATLAB

  MATLAB function

**Package**

- Simulation Author

  Use Simulink Compiler / Compiler SDK to package

  Deployed Archive

**Host**

- IT Admin

  Host on MATLAB Production Server

  RESTful interface

**Execute**

- Simulation User

  Call API via web apps or enterprise applications
Scenario 4

Package simulations as standalone FMU

**Standalone FMUs**

Runs in 3rd party simulation tools (which support FMI import)

Has no dependency on MATLAB Runtime

Supports standalone co-simulation FMU with fixed-step solver only
Scenario 4
Package simulations as standalone FMU

Standalone FMUs

Author
- Simulation Author
- Create and configure model for export
- Simulink Model

Package
- Simulation Author
- Use Simulink Compiler to package
- Standalone FMU

Execute
- Simulation User
- Run simulation in 3rd party environments
Simulink Compiler is the latest addition to MathWorks deployment options.
Simulink Compiler is an out-of-the-box solution to share Simulink simulations.
Edit at the Speed of Thought

Model Run-Time Software

Componentize Your Design

Protected Models
Projects
Simulink Compiler

Standalone Apps
Work more **effectively** and **efficiently** with the latest release, and check out:

- Edit at the Speed of Thought
- Model Run-Time Software
- Componentize Your Design

**Simulink What’s New Page**

**Release Notes**

**New Ways to Work in Simulink** (Video Series)

**Blogs**

**Software Updates**
Access and **try the latest release** with MATLAB and **Simulink Online**
Access and try the latest release with MATLAB and Simulink Online
Get **started** and **learn** with **Onramps**

- MATLAB Onramp
- Simulink Onramp
- Stateflow Onramp
- Deep Learning Onramp
- Machine Learning Onramp
- Control Design Onramp
Upgrade to the latest release **today**

Get Latest Release

**R2020b**

» Learn More

Download

Click here to download any MathWorks release