

Simulink® Coverage™ Release Notes



MATLAB® & SIMULINK®



## How to Contact MathWorks



Latest news: [www.mathworks.com](http://www.mathworks.com)  
Sales and services: [www.mathworks.com/sales\\_and\\_services](http://www.mathworks.com/sales_and_services)  
User community: [www.mathworks.com/matlabcentral](http://www.mathworks.com/matlabcentral)  
Technical support: [www.mathworks.com/support/contact\\_us](http://www.mathworks.com/support/contact_us)



Phone: 508-647-7000



The MathWorks, Inc.  
1 Apple Hill Drive  
Natick, MA 01760-2098

### *Simulink® Coverage™ Release Notes*

© COPYRIGHT 2017–2020 by The MathWorks, Inc.

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from The MathWorks, Inc.

FEDERAL ACQUISITION: This provision applies to all acquisitions of the Program and Documentation by, for, or through the federal government of the United States. By accepting delivery of the Program or Documentation, the government hereby agrees that this software or documentation qualifies as commercial computer software or commercial computer software documentation as such terms are used or defined in FAR 12.212, DFARS Part 227.72, and DFARS 252.227-7014. Accordingly, the terms and conditions of this Agreement and only those rights specified in this Agreement, shall pertain to and govern the use, modification, reproduction, release, performance, display, and disclosure of the Program and Documentation by the federal government (or other entity acquiring for or through the federal government) and shall supersede any conflicting contractual terms or conditions. If this License fails to meet the government's needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to The MathWorks, Inc.

### **Trademarks**

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [www.mathworks.com/trademarks](http://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.

### **Patents**

MathWorks products are protected by one or more U.S. patents. Please see [www.mathworks.com/patents](http://www.mathworks.com/patents) for more information.

## R2020b

<b>Load and view coverage data from previous releases</b> .....	1-2
<b>Justify missing coverage for individual MCDC objectives</b> .....	1-2
<b>Collect and view requirements testing data by using the Model Testing Dashboard</b> .....	1-3
<b>Simplified coverage configuration parameters</b> .....	1-3
<b>Functionality being removed or changed</b> .....	1-3
CovShowResultsExplorer, CovHighlightResults, and CovHtmlReporting are removed .....	1-3

## R2020a

<b>Scope model coverage to requirements-based tests</b> .....	2-2
<b>Manage and view multiple coverage filters</b> .....	2-2
<b>Coverage support for Stateflow variant transitions</b> .....	2-3
<b>Enhanced calculation of cyclomatic complexity</b> .....	2-3

## R2019b

<b>Unit-to-System Test Coverage Aggregation: View system test coverage achieved from unit tests in new Aggregated Tests section of coverage report</b> .....	3-2
<b>Requirements-to-Test-Case Traceability in Coverage Report: View Simulink Requirements links and coverage details for each Simulink block when generating coverage reports from Simulink Test Manager</b> .....	3-2

<b>Test Case Traceability of Coverage Results: Trace coverage results to relevant simulations in Simulink Test Manager and Coverage Results Explorer</b> .....	3-2
Simulink Test Integration .....	3-2
<b>Coverage Toolstrip: Access common coverage features from the new Simulink Toolstrip, including model highlighting, coverage details, and report generation</b> .....	3-2
<b>View code coverage information in code view</b> .....	3-3
<b>Extract subsystem coverage data from system-level coverage data</b> .....	3-3

## R2019a

<b>Lookup table breakpoint value changes in coverage data</b> .....	4-2
<b>Simulink Coverage contextual tabs in the Simulink Toolstrip Tech Preview</b> .....	4-2

## R2018b

<b>Model Coverage Visualization: Gain enhanced perspective of coverage results through model highlighting and pop-ups within the Simulink Editor</b> .....	5-2
<b>Parallel Simulation Support: Accelerate coverage analysis through use of parsim</b> .....	5-2
<b>Stateflow Custom Code Support: Collect coverage on elements of Stateflow charts where C/C++ code is used</b> .....	5-2
<b>C Caller Block Support: Perform code coverage analysis for custom C/C++ code in Simulink models</b> .....	5-2
<b>Coverage Filtering API: Create filter rules for custom C/C++ code in normal mode and generated code in SIL or PIL modes</b> .....	5-2

## R2018a

<b>Fine-grained filtering for relational boundary metrics: Control coverage results for individual design elements</b> .....	6-2
--	-----

<b>Stateflow Just-In-Time (JIT) Compilation Mode: Reduce model update time when recording coverage .....</b>	<b>6-2</b>
--	------------

**R2017b**

---

<b>Simulink Verification and Validation Packaging: Moved model and generated code coverage functionality and component verification functions such as slvnmakeharness to Simulink Coverage .....</b>	<b>7-2</b>
<b>Coverage Filtering API: Filtering choices for coverage justifications that include specified decisions, conditions, and outcomes .....</b>	<b>7-2</b>
<b>Logical Expressions in Assignment Statements: Record Condition and MCDC coverage for logical expressions in assignments in Stateflow and MATLAB Function Blocks .....</b>	<b>7-2</b>
<b>Function and Function Call Coverage: Collect SIL &amp; PIL coverage as required by ISO 26262 .....</b>	<b>7-2</b>



# R2020b

---

**Version: 5.1**

**New Features**

**Bug Fixes**

**Compatibility Considerations**

## Load and view coverage data from previous releases

In R2020b, you can load and view coverage data from previous releases as far back as R2017b.

The `cvdata` object now contains a `dbVersion` property that identifies the origin release of the coverage data.

```
[~, cvd] = cvload('myCvData_19b');
cvd = cvd{1};
cvd.dbVersion
ans =
    '(R2019b)'
```

The coverage report contains a new section about the coverage data, which includes a field titled **Collected in version**.

After you import coverage data from a previous release, you can extract information by using the same API that you would use normally. Model highlighting and filtering continue to work as expected. You can aggregate coverage data from two or more `cvdata` objects if the `dbVersion` properties match.

The Test Manager in Simulink® Test™ can show coverage reports for imported coverage data from a previous release. Simulink Design Verifier™ can use coverage data collected in a previous release to generate test cases to achieve missing coverage. For more information, see “Achieve Missing Coverage in Referenced Model” (Simulink Design Verifier).

It is not possible to export coverage data from the current release to a previous release format.

## Justify missing coverage for individual MCDC objectives

In R2020b, you can justify missing coverage for individual MCDC objectives.

The following image shows the model coverage report with a missing coverage objective justified after MCDC analysis.

**MC/DC analysis (combinations in parentheses did not occur)**

Decision/Condition	True Out	False Out
expression for output		
input port 1	<u>1</u>	
input port 2	TTTT	TFTT
input port 3	TTTT	TTFT
input port 4	TTTT	(TTTF)



---

The input port 1 expression is a missing coverage objective that has been justified after MCDC analysis. The input port 4 expression shows a missing coverage objective and the **Add justification rule** icon.

To justify an MCDC outcome, follow the same procedure as justifying a decision or condition outcome. For a coverage filtering example, see “Creating and Using Coverage Filters”. For an example that justifies an unsatisfied MCDC objective outcome, see “Filter Coverage Results Using a Script”.

## Collect and view requirements testing data by using the Model Testing Dashboard

The Model Testing Dashboard collects and displays metric data on the status and quality of your requirements-based testing. If you have Simulink Check™ and Simulink Test licenses, you can assess the testing status of a model by using the dashboard to view:

- Summary data on requirements, tests, and traceability between requirements and tests
- Status and results for the latest test runs
- Model coverage measurements achieved by tests and justifications
- A list of the latest artifacts of the project, organized by the associated models

For more information, see “Model Testing Dashboard: Track completeness of requirements-based testing for compliance to standards such as ISO 26262” (Simulink Check).

## Simplified coverage configuration parameters

In R2020b, in the Configuration Parameters window, the **Coverage > Results** tab is removed. The parameters previously located under **Results** are now listed under **Coverage**.

## Functionality being removed or changed

### CovShowResultsExplorer, CovHighlightResults, and CovHtmlReporting are removed

In R2020b, three Simulink Coverage™ configuration parameters are removed:

- `CovShowResultsExplorer` — To open the Coverage Results Explorer, in the **Apps** tab, click **Coverage Analyzer**. Then click **Results Explorer**.
- `CovHighlightResults` — To highlight coverage results in your model, use `cvmodelview`.
- `CovHtmlReporting` — To create a coverage report, use `cvhtml`.

## Compatibility Considerations

Using `CovShowResultsExplorer`, `CovHighlightResults`, or `CovHtmlReporting` does not cause errors in your simulation, but they are ignored and a warning is generated.

This change affects only models run using `sim`. These parameters were already ignored by `cvsim` and simulations run using the **Run** button.



# R2020a

---

**Version: 5.0**

**New Features**

**Bug Fixes**

**Compatibility Considerations**

## Scope model coverage to requirements-based tests

Starting in R2020a, you can scope coverage results to linked requirements-based tests. From the Simulink Test Manager, select **Scope coverage results to linked requirements**. This setting scopes the aggregated coverage results such that each test only contributes coverage for the corresponding model elements that implement the requirements verified by that test. This improves confidence that model elements are covered by the intended test cases.

For more information on scoping coverage results to linked requirements, see [Assess Coverage Results from Requirements-Based Tests](#).

For more information on collecting coverage by using the Simulink Test Manager, see [Test Coverage for Requirements-Based Testing \(Simulink Test\)](#). For more information on creating requirements links, see [Link Blocks and Requirements \(Simulink Requirements\)](#).

## Manage and view multiple coverage filters

You can apply multiple coverage filters to coverage data:

- Programmatically. For more information, see [Use Multiple Filter Files for a Simulation](#).
- From the **Results Explorer**. For more information, see [Creating and Using Coverage Filters](#).
- From the **Results and Artifacts** pane of the Simulink Test Manager. You can add or remove existing coverage filters from the **Applied Coverage Filters** section or create new filter rules directly from a report created by using the **Report** button in the **Aggregated Coverage Results** section.

The screenshot displays the Simulink Test Manager interface. On the left, the 'TESTS' pane shows a 'Test Browser' and 'Results and Artifacts' section. The 'Results and Artifacts' section lists 'Results: 2019-Oct-02 14:48:55' with a status of '1' and a green checkmark, and 'New Test Case 1' with a green checkmark. Below this is a table with the following data:

PROPERTY	VALUE
Name	Results: 2019-Oct-02 1...
Status	1 ✓
Start Time	10/02/2019 14:48:55
End Time	10/02/2019 14:48:57

The main pane shows 'Results: 2019-Oct-02 14:48:55'. It includes a 'SUMMARY' section and an 'AGGREGATED COVERAGE RESULTS' table. The table has columns for 'ANALYZED MODEL', 'REPORT', 'COMPLEXI...', 'DECISION', and 'EXECUTION'. The row for 'slvndemo\_covfilt' shows a 'REPORT' button (highlighted with a red box), a complexity of 25, a decision of 83%, and an execution of 69%. Below the table is a checkbox for 'Scope coverage results to linked requirements' and buttons for '+ Add Tests for Missing Coverage' and 'Export'.

The 'APPLIED COVERAGE FILTERS' section is highlighted with a red box and contains a list of filter names:

- CompanyStandardFilter\_DivBy0\_lib.cvf
- CompanyStandardFilter\_tick.cvf
- myModel\_Justifications.cvf

Buttons for '+ Add' and 'Remove' are located at the bottom right of this section.

- From coverage results. You can create new filter rules from a coverage report by right-clicking a block with model coverage highlighting or from the **Coverage Details** pane. For more information, see [View Coverage Results in a Model](#).

---

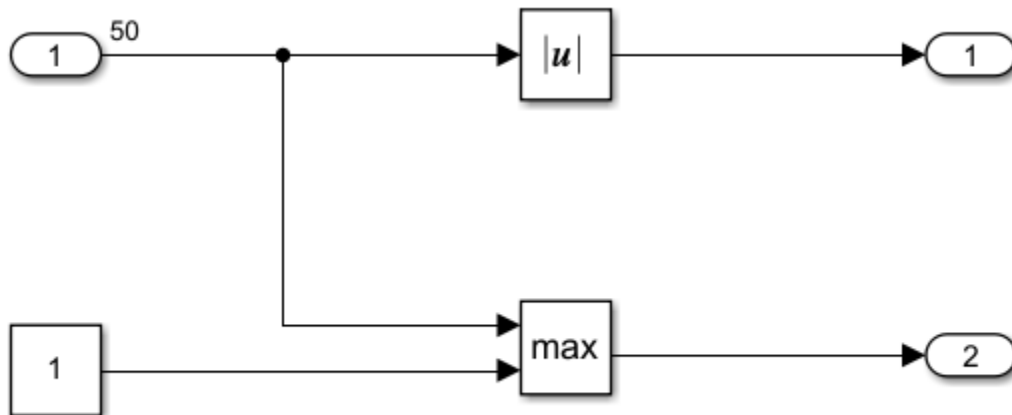
## Coverage support for Stateflow variant transitions

Simulink Coverage records coverage for active Stateflow® variants. For more information, see [Software Configurations Using Variant Transitions \(Stateflow\)](#).

## Enhanced calculation of cyclomatic complexity

In R2019b, the cyclomatic complexity metric calculation counted each element of a vectorized operation as a separate decision point. As a result, a simple model with a signal that is a vector or matrix could have a high cyclomatic complexity result. In R2020a, the cyclomatic complexity calculation considers a vectorized operation as a single decision point, thereby lowering the cyclomatic complexity result for some models.

For example, consider the following model which has a vector signal of size 50. This signal branches to two signals. Two blocks perform calculations on each signal.



In R2019b, the coverage report stated that this model had a cyclomatic complexity of 101. In R2020a, the coverage report states this model has a cyclomatic complexity of 3. This is a more reasonable value for a simple model. For more information, see [Cyclomatic Complexity](#).

## Compatibility Considerations

Previously, models with vectorized operations had higher cyclomatic complexity values than they have in R2020a.



# R2019b

---

**Version: 4.4**

**New Features**

**Bug Fixes**

## **Unit-to-System Test Coverage Aggregation: View system test coverage achieved from unit tests in new Aggregated Tests section of coverage report**

If you use Simulink Test to record coverage for multiple subsystems from the same model, you can aggregate the coverage data for the subsystems into top-level model coverage. You access the aggregated results in the Simulink Test Manager results tab for a given test run.

For more information on recording coverage for test cases in Simulink Test, see [Trace Coverage Results to Associated Test Cases and Aggregated Tests](#).

## **Requirements-to-Test-Case Traceability in Coverage Report: View Simulink Requirements links and coverage details for each Simulink block when generating coverage reports from Simulink Test Manager**

When you collect aggregated coverage from the Simulink Test Manager for a model with associated Simulink Requirements™ links, the coverage report now includes a summary of the associated requirements links for each model element.

For more information, see [Trace Coverage Results to Requirements by Using Simulink Test and Simulink Requirements and Requirement Testing Details](#).

## **Test Case Traceability of Coverage Results: Trace coverage results to relevant simulations in Simulink Test Manager and Coverage Results Explorer**

When you collect aggregated coverage for a model, the coverage report now includes a summary of the associated tests in the Results Explorer or the Simulink Test Manager and links to the first test case which executed each coverage outcome for model elements.

For more information, see [Trace Coverage Results to Associated Test Cases and Aggregated Tests](#).

### **Simulink Test Integration**

If you record coverage from the Simulink Test Manager, the coverage report includes the associated Simulink Test test cases in the **Aggregated Test** summary. The coverage details section for each model element also links to Simulink Test test cases associated with each coverage outcome.

## **Coverage Toolstrip: Access common coverage features from the new Simulink Toolstrip, including model highlighting, coverage details, and report generation**

The Simulink toolstrip includes contextual tabs, which appear when you open the **Coverage Analyzer** app, under **Verification, Validation, and Test**. The Simulink Coverage contextual tab includes options for completing actions that apply only to Simulink Coverage.

For more information, see [“Simulink Toolstrip: Access and discover Simulink capabilities when you need them”](#).



---

## **View code coverage information in code view**

If you have an Embedded Coder® license, you can view code coverage information in the Code view. For more information, see “Code coverage information in Code view” (Embedded Coder).

## **Extract subsystem coverage data from system-level coverage data**

You can now use `extract` to extract the coverage data for a subsystem. For an example, see [Create HTML Coverage Report for a Subsystem from Model Coverage Data](#).



# R2019a

---

**Version: 4.3**

**New Features**

**Bug Fixes**

## Lookup table breakpoint value changes in coverage data

In R2019a, when you record coverage for multiple runs of a model with a lookup table, and then change the breakpoint values for the lookup table between runs, the coverage data from the previous runs is compatible with the updated model if the dimensions of the lookup table are the same as in the previous runs.

## Simulink Coverage contextual tabs in the Simulink Toolstrip Tech Preview

In R2019a, you can turn on the Simulink toolstrip. See “Simulink Toolstrip Tech Preview replaces menus and toolbars in the Simulink Desktop” for more details.

The Simulink toolstrip includes contextual tabs, which appear when you open the **Coverage Analyzer** app, under **Verification, Validation, and Test**. The Simulink Coverage contextual tab includes options for completing actions that apply only to Simulink Coverage. Documentation does not reflect the addition of the Simulink Coverage contextual tabs.

# R2018b

---

**Version: 4.2**

**New Features**

## **Model Coverage Visualization: Gain enhanced perspective of coverage results through model highlighting and pop-ups within the Simulink Editor**

The model coverage perspective allows you to hover over a model element and view a coverage summary for the model element in a tool tip. The **Coverage Details** window allows you to view detailed coverage information for model elements without leaving the Simulink Editor. For more information, see [View Coverage Results in a Model](#).

## **Parallel Simulation Support: Accelerate coverage analysis through use of parsim**

You can now leverage the `parsim` function to record model coverage for multiple simulation runs in parallel. For more information, see [Record Coverage in Parallel Simulations by Using Parsim and parsim](#).

## **Stateflow Custom Code Support: Collect coverage on elements of Stateflow charts where C/C++ code is used**

Simulink Coverage records code coverage for elements of Stateflow charts where C/C++ code is used. For more information on how to enable custom code support, see [Coverage for Custom C/C++ Code in Simulink Models](#).

## **C Caller Block Support: Perform code coverage analysis for custom C/C++ code in Simulink models**

Simulink Coverage records code coverage for custom C/C++ code in C Caller blocks. For more information on how to enable custom code support, see [Coverage for Custom C/C++ Code in Simulink Models](#).

## **Coverage Filtering API: Create filter rules for custom C/C++ code in normal mode and generated code in SIL or PIL modes**

You can use model coverage commands to filter custom C/C++ code in normal mode and generated code in SIL or PIL modes. For more information, see `slcoverage.CodeSelector` and [Automate Coverage Workflows](#).

# R2018a

---

**Version: 4.1**

**New Features**

## **Fine-grained filtering for relational boundary metrics: Control coverage results for individual design elements**

To achieve complete coverage when you record saturate on integer overflow coverage or relational boundary coverage for a model, you can exclude or justify incomplete coverage outcomes from the coverage report.

For more information on coverage filtering, see [Create, Edit, and View Coverage Filter Rules](#). For more information on saturate on integer overflow coverage and relational boundary model coverage, see [Types of Model Coverage](#).

## **Stateflow Just-In-Time (JIT) Compilation Mode: Reduce model update time when recording coverage**

When you record coverage for Stateflow charts, Stateflow uses just-in-time (JIT) compilation technology to improve model update performance. For more information on JIT compilation technology, see [Speed Up Simulation](#).



# R2017b

---

**Version: 4.0**

**New Features**

**Compatibility Considerations**

## **Simulink Verification and Validation Packaging: Moved model and generated code coverage functionality and component verification functions such as slvnmakeharness to Simulink Coverage**

The model and generated code coverage functionalities and component verification functions of Simulink Verification and Validation™ have been transitioned to Simulink Coverage. For an introduction to the product, a basic Simulink Coverage workflow, and an outline of how Simulink Coverage fits into a systematic, end-to-end verification workflow, see the Getting Started with Simulink Coverage category. For coverage-related release notes for Simulink Verification and Validation prior to R2017b, see <https://www.mathworks.com/help/releases/R2017a/slvnv/release-notes.html>.

## **Coverage Filtering API: Filtering choices for coverage justifications that include specified decisions, conditions, and outcomes**

You can use a command-line API to create filtering rules for blocks. Selection criteria for filtering includes filtering by individual block ID, filtering for all blocks of the same type, filtering certain decisions, conditions, and outcomes of a block, and more. You can also filter S-Function C++ code by code coverage outcome. For more information and examples, see Automate Coverage Workflows.

## **Logical Expressions in Assignment Statements: Record Condition and MCDC coverage for logical expressions in assignments in Stateflow and MATLAB Function Blocks**

In models where logical expressions are assigned to variables – to break up complicated logic, to reuse common subexpressions, etc. – Simulink Coverage now records Condition and MCDC coverage for the logical expressions in assignment statements. For a detailed example of how Simulink Coverage records Condition and MCDC coverage for models where logical expressions are assigned to variables, see Coverage for MATLAB® Function Blocks.

## **Compatibility Considerations**

Models that use logical expressions in assignment statements in Stateflow and MATLAB® Function blocks record an increased number of Condition and MCDC objectives than previously recorded.

## **Function and Function Call Coverage: Collect SIL & PIL coverage as required by ISO 26262**

Simulink Coverage introduces two new metrics for measuring Statement coverage for code.

- **Function Coverage:** Function coverage determines whether all the functions of your code have been called during simulation.
- **Function Call Coverage:** Function call coverage determines whether all function calls in your code have been executed.

The new metrics are reported in the top-level Summary and in the Details section of the HTML Coverage Report when you record code coverage.