

Requirements Toolbox™ Release Notes



MATLAB® & SIMULINK®



How to Contact MathWorks



Latest news: www.mathworks.com
Sales and services: www.mathworks.com/sales_and_services
User community: www.mathworks.com/matlabcentral
Technical support: www.mathworks.com/support/contact_us



Phone: 508-647-7000



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

Requirements Toolbox™ Release Notes

© COPYRIGHT 2017–2023 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 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 for more information.

R2023a

Link from sequence diagrams to requirements	1-2
Use stereotypes when importing requirements and links from ReqIF files	1-2
Generate report to analyze comparison results for requirement and link sets	1-2
Prove properties with the Requirements Table block	1-2
Debug Requirements Table blocks with the debugger	1-2
Define custom document domain types for importing requirements from third-party applications	1-3
Enhancements to programmatically accessing requirement and link information	1-3
Programmatic interface enhancements	1-4

R2022b

Use profiles and stereotypes with requirements and links	2-2
Choose when to run change tracking and implementation and verification status analyses	2-2
Include requirements comments in code generated from MATLAB code	2-2
Track changes to MATLAB code lines	2-2
Create selection-based links from C code and other plain text files	2-3
Link with MATLAB code programmatically and adjust linked code range	2-3
New link set file naming standard	2-3

Filter requirements and links in the Requirements Editor and Requirements Perspective	2-3
Compare requirement and link set files in Comparison Tool	2-3
View verification status of test steps in Requirements Perspective	2-4
String support for Requirements Table blocks	2-4

R2022a

Author, link, and validate requirements implemented by MATLAB algorithms	3-2
Use slreq.refreshCustomizations to register custom requirement and link types in MATLAB	3-2
Formalize requirements in Simulink Models by using Requirements Table Block	3-2
Run, link, and view verification status of test cases created in MATLAB Unit Test	3-2
Track changes to Simulink Test test cases	3-3
Run favorite commands from the Requirements Editor quick access toolbar	3-3
Use callbacks to execute code when you load and save requirement set files	3-3
Use callbacks to execute code when you import or update requirements	3-3
Customize requirement index numbering	3-3
Enhanced support for IBM DOORS Next integration	3-3
Support for hierarchical merge changes	3-4
Programmatic interface enhancements	3-4
Functionality being removed or changed	3-4
Changes to Type property and index numbering for imported IBM DOORS Next requirements	3-4
Changes to update behavior for requirements imported from ReqIF files	3-4

R2021b

Visualize traceability between Model-Based Design items with a traceability diagram	4-2
Open the Requirements Editor from the MATLAB and Simulink apps galleries	4-2
Customize the Requirements Editor quick access toolbar	4-2
Merge changes to requirement set and link set files from multiple users	4-2
Enhanced support for importing and exporting ReqIF files	4-3
Register a custom callback function to authenticate IBM DOORS Next or other web-based servers	4-3
Import a subset of requirements from an IBM Rational DOORS module	4-3

R2021a

Manage requirements with updated Requirements Editor	5-2
Traceability Matrix: Create matrix with multiple artifacts manually or by API	5-4
Enhanced support for importing and exporting ReqIF files	5-4
Enhancements for linking with requirements in IBM DOORS Next	5-4
Improved loading of link information	5-5
oslc Object: Access OSLC services with command-line client	5-5
Programmatic interface enhancements	5-5

R2020b

Traceability Matrix: Select multiple cells, create and delete links in batch, track changes, and export	6-2
--	-----

Customize link sets and links with custom attributes programmatically or in the Requirements Editor	6-2
Verify requirements with MATLAB-based Simulink Tests	6-3
User interface enhancements	6-3
Programmatic interface enhancements	6-3
Model Testing Dashboard	6-3

R2020a

Traceability Matrix: Review and analyze links in a matrix format	7-2
Include results from external sources in verification status	7-2
View, link, and sync IBM Rational DOORS Next Generation requirements	7-2
Link from ports in System Composer	7-3
User interface enhancements	7-3
Programmatic interface enhancements	7-3

R2019b

Contextual Tabs in the Simulink Toolstrip for Simulink Requirements	8-2
Improved Requirements Exchange with ReqIF	8-2
Navigation Support for Baselines in IBM Rational DOORS 9	8-2
Enhanced Web View Reports for Reviewing Requirements and Links ...	8-2
Programmatically Modify Link Source and Destination	8-2
External Source Document Change Notification	8-3

Round Trip Workflow with Requirements Tools: Exchange requirements with third-party requirements tools through ReqIF	9-2
Navigation Backlinks for Third-Party Requirements: Manage backlinks in external requirement documents that enable you to navigate to Simulink model elements	9-2
Integration with System Composer: Link requirements to architecture models in System Composer	9-2
Usability Enhancements	9-2
Simulink Requirements contextual tab in the Simulink Toolstrip	9-3
Code Traceability: Include outgoing traceability links from external MATLAB code referenced by MATLAB Function Blocks in code generation report	9-3
Report Generation Enhancements: Include change information in requirements report	9-3

External Rich Text Editor: Use text editing and formatting features of Microsoft Word to author and edit rich text requirements	10-2
Configurable Requirement and Link Types: Define custom requirement and link types for improved classification and analysis	10-2
Incoming Link Resolution: Automatically load and display incoming links when opening artifacts stored in Simulink Projects	10-2
IBM Rational DOORS Next Generation Integration: Establish traceability to requirements stored in DOORS Next Generation project areas with configuration management enabled	10-2
Extended Requirements Management Capabilities: Organize, view and manage requirements content with improved indexing and updating of imported requirements content	10-3

Requirements Import with ReqIF: Import requirements information from third-party tools supporting the ReqIF standard such as IBM Rational DOORS Next Generation or Siemens Polarion	11-2
IBM Rational DOORS Import Wizard: Work with and manage requirements from DOORS by importing them as read-only references	11-2
Batch Test Execution: Verify requirements by launching linked tests	11-2
Link Set Comparison: Track changes to requirements links by comparing Link Set files	11-2
Simulink and Stateflow Annotation Conversion: Convert existing annotations to requirements to improve traceability, analysis, and management	11-2
Test Harness Support: Work with Simulink Test test harnesses from the requirements perspective	11-2
Requirements Report Generation: Generate requirements reports in PDF and HTML formats	11-3

Introduction to Simulink Requirements: Author, analyze, and manage requirements	12-2
Requirements Editor and Browser: Author, edit, and organize requirements within Simulink	12-2
Requirements Import Wizard: Import requirements content from Microsoft Word and Microsoft Excel documents	12-2
Requirements Perspective: Establish requirements - design traceability and annotate your model with requirements information in the Simulink Editor	12-2
Requirements Change Tracking and Comparison: Automatically identify and manage changing requirements and impacts on design, models, and tests	12-3
Implementation and Verification Status Summary: Review implementation and verification status metrics for your requirements	12-3

Requirements Reporting: Archive and document your requirements traceability, implementation status, and change history	12-3
Simulink Verification and Validation Packaging: Moved Requirements Management Interface (RMI) features from Simulink Verification and Validation to Simulink Requirements	12-3

R2023a

Version: 2.2

New Features

Bug Fixes

Link from sequence diagrams to requirements

You can now link to requirements from sequence diagrams in System Composer™ and AUTOSAR Blockset. You can link sequence diagrams and elements in a sequence diagram:

- Lifelines
- Messages
- Gates
- Fragments

For more information, see “Link from Sequence Diagrams”.

Use stereotypes when importing requirements and links from ReqIF files

When you import requirements and links from ReqIF™ files, you can map the ReqIF requirement types, link types, and attributes to Requirements Toolbox stereotypes and stereotype properties. To map the requirements and link types, enable the **Requirements Editor** to use profiles during import, then create a new profile. The **Requirements Editor** creates a profile with a stereotype for each requirement type and link type in the ReqIF file. Each stereotype has a property for the attributes defined by the requirement and link types. Alternatively, you can use a profile that you created during a previous import or by using the **Profile Editor**.

For more information, see “Use Stereotypes when Importing from ReqIF Files”.

Generate report to analyze comparison results for requirement and link sets

You can now generate a report to analyze comparison results for requirement sets and link set files. You can generate the report in the HTML, DOCX, or PDF formats from the comparison user interface or by using the `visdiff` command. You can generate a custom report by applying filters in the comparison user interface or by using the `visdiff` command. For more information, see “Compare Requirement Sets Using Comparison Tool” and “Publish and Save Printable Report of Comparison Results”.

Prove properties with the Requirements Table block

If you have Simulink® Design Verifier™, you can prove the properties of your model by using a Requirements Table block. When you create requirements in the block, open the **Design Verifier** tab, then click **Prove Properties**. Simulink Design Verifier produces proof objectives for the requirements in the requirement set. For more information, see “Prove Properties with Requirements Table Blocks”.

Debug Requirements Table blocks with the debugger

You can now debug Requirements Table blocks with the debugger. Use the debugger by setting breakpoints in requirements that stop before or after the block evaluates the preconditions. When you simulate the model, the simulation pauses at the breakpoints. For more information, see “Debug Requirements Table Blocks”.

Define custom document domain types for importing requirements from third-party applications

Prior to R2023a, to import requirements from third-party applications that Requirements Toolbox does not support, you had to use the Requirements Interchange Format (ReqIF).

Starting in R2023a, you can define custom document domain types to import requirements from third-party applications that Requirements Toolbox does not support. You can also create a custom version of a built-in Requirements Toolbox domain type for a supported products, such as Microsoft® Word.

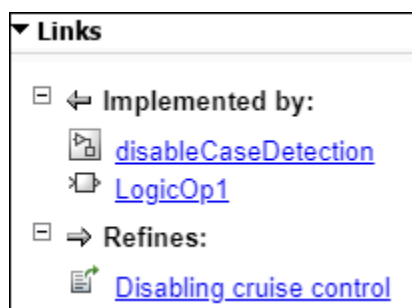
For more information, see “Define Custom Document Interface for Importing Requirements”.

Enhancements to programmatically accessing requirement and link information

In addition to the accessing requirement and link information via property and custom attribute values, you can now programmatically access some requirement and link information from the **Requirements Editor**.

To get link information provided in the **Links** section of the **Requirements Editor**:

- Use `getOutgoingTypeLabel` and `getIncomingTypeLabel` to get the label classifying the link type at its source and destination, respectively.
- Use `getSourceLabel` and `getDestinationLabel` to get the label for the source and destination of the link.
- Use `getSourceOpenCommand` and `getDestinationOpenCommand` to get a command to navigate to the source and destination from within MATLAB®.
- Use `getSourceURL` and `getDestinationURL` to get a URL to navigate to the link source or destination from within a web browser.
- Use `slreq.getIncomingLinkTypeLabel` and `slreq.getOutgoingLinkTypeLabel` to get the forward or backward name for the specified link type that is displayed in the **Requirements Editor**.



To get content displayed in the **Requirements Editor** for requirements and referenced requirements:

- Use `getDescriptionAsText` to get the requirement description as plain text.
- Use `getRationaleAsText` to get the requirement rationale as plain text.

Programmatic interface enhancements

- You can now export requirements and links to ReqIF files by using `slreq.export`.
- You can now get the incoming link sets for a requirement set by using `slreq.getDependentLinkSets`. For example, suppose requirements in a requirement set called `myReqSet` have incoming links from Simulink models and MATLAB code. To get the link sets that are stored with the Simulink models and MATLAB code files, pass `myReqSet` as an input to `slreq.getDependentLinkSets`.

R2022b

Version: 2.1

New Features

Bug Fixes

Use profiles and stereotypes with requirements and links

You can use stereotypes to define custom requirement or link types with custom properties. You can use stereotypes to apply properties to a subset of requirements or links within a requirement set or link set. To define a stereotype, create a profile to contain the stereotype, assign the profile to a requirement set or link set, then use the stereotypes with the requirements or links. You can reuse stereotypes by assigning profiles to multiple requirement sets or link sets. For more information, see [Customize Requirements and Links by Using Stereotypes](#).


For a comparison between stereotypes, custom types defined in `sl_customization` files, and custom attributes, see [Define Custom Requirement and Link Types and Properties](#).

Choose when to run change tracking and implementation and verification status analyses

Prior to R2022b, when you opened requirements that have links, Requirements Toolbox performed change tracking, implementation status, and verification status analyses. For more information about these analyses, see:

- [Track Changes to Requirement Links](#)
- [Review Requirements Implementation Status](#)
- [Review Requirements Verification Status](#)

Starting in R2022b, Requirements Toolbox does not perform these analyses until you run them. A banner in the **Requirements Editor** and Requirements Perspective indicates when the results are pending.

To perform the analyses, click **Analyze now** in the banner or the refresh icon  in the **Requirements Editor**. The analyses continuously run in the background until you use `slreq.clear`.

Include requirements comments in code generated from MATLAB code

When you generate C/C++ code from MATLAB code that has links to requirements, you can include comments that contain information about the requirements and the linked MATLAB code ranges. When you view the generated code from the code generation report, the comments are hyperlinks that you can use to navigate to the requirement or the linked MATLAB code range. For more information, see [Requirements Traceability for Code Generated from MATLAB Code](#).

Track changes to MATLAB code lines

You can now track the changes to MATLAB code lines and view the impacted requirements in the **Requirements Editor**. When you view requirements by clicking **Show Requirements** or **Show Links**, the **Requirements Editor** highlights the impacted requirements. You can batch run all changed test cases linked with the specific requirement in a single run. You can also review changes in MATLAB code or tests and clear the associated change issues. See [Review Changes to MATLAB Code Lines](#).

For an example, see [Track Changes to MATLAB® Code Using Requirements Editor](#).

Create selection-based links from C code and other plain text files

You can now create selection-based links from lines of code in plain text files, such as C and H files. To create the link, open the plain text file in the MATLAB Editor. For more information, see [Link Requirements to MATLAB or Plain Text Code and Create and Store Links](#). By default, the link type for links from plain text code is `Relate`.

Link with MATLAB code programmatically and adjust linked code range

Starting in R2022b, you can create links to MATLAB code or plain-text external code programmatically. Additionally, you can adjust the linked code range programmatically or in the MATLAB Editor. For more information, see [Requirements Traceability for MATLAB Code](#).

New link set file naming standard

Prior to R2022b, when you created a link from a design artifact, such as a Simulink model, the link set file name was the same as the base name of the design artifact. For example, Requirements Toolbox stored outgoing links from the `vdp.slx` model in the `vdp.slmx` link set.

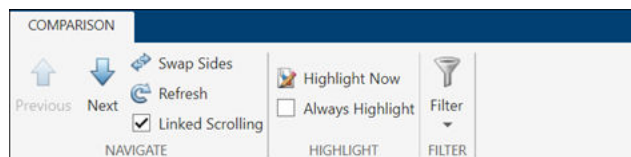
Starting in R2022b, when you create a link from a design artifact, the link set file name includes both the artifact base name and the artifact extension, separated by a tilde. This naming standard avoids conflicts when multiple design artifacts use the same base name. For example, Requirements Toolbox stores links from a MATLAB function called `myFunction.m` in a link set file called `myFunction~m.slmx`. For more information, see [Link Storage](#).

Filter requirements and links in the Requirements Editor and Requirements Perspective

In R2022b, you can filter requirements and links in the **Requirements Editor** and Requirements Perspective by creating a view that filters based on properties and attributes. You can save the view in your preferences folder or in a requirement set. For more information, see [Filter Requirements and Links in the Requirements Editor](#).

Compare requirement and link set files in Comparison Tool

In R2022b, Requirements Toolbox has a redesigned comparison tool user interface to compare requirement and link set files. You can compare requirement and link set files belonging to any version in the comparison tool. The tool displays the requirement and link set contents in a hierarchical manner.



You can perform these actions in the comparison tool user interface:

- **Navigate:** Navigate requirements and link sets using the **Next**, **Previous**, **Swap Sides**, and **Refresh** options.

- **Highlight:** Highlight changes in the requirement and link set comparison tree using the **Highlight Now** and **Always Highlight** buttons.
- **Filter:** Filter to show only changed items using **Show Only Changed** button.

For an example, see Compare Requirement Sets Using Comparison Tool.

View verification status of test steps in Requirements Perspective

In R2022b, Requirements Toolbox by default mark links with `step` in Test Sequence and Test Assessment block that contain `verify` statement as verification links. You can run the tests in batches and view the verification status by using the Requirements Perspective tool and the **Requirements Editor**. You can also identify stale results from Test Sequence blocks. For more information, see Verify Functional Requirements.

String support for Requirements Table blocks

You can create and manipulate string data in Requirements Table blocks. The string data type is compatible with strings in MATLAB and Simulink. To see a list of supported operators, see the **Requirements Table Block Operators** section in Formalize Requirements in Simulink Models.

R2022a

Version: 2.0

New Features

Bug Fixes

Compatibility Considerations

Author, link, and validate requirements implemented by MATLAB algorithms

Simulink Requirements™ is now called Requirements Toolbox and does not require Simulink. You can author rich text requirements in MATLAB with custom attributes or import them from other requirements management tools and create links to MATLAB code and tests. See [Verify a MATLAB Algorithm by Using Requirements-Based Tests](#).

Use `slreq.refreshCustomizations` to register custom requirement and link types in MATLAB

Starting in R2022a, you can use `slreq.refreshCustomizations` to register custom requirement or link types in MATLAB. This function does not require Simulink. If you have Simulink, `slreq.refreshCustomizations` behaves the same as `sl_refresh_customizations` (Simulink). For more information, see [Define Custom Requirement and Link Types](#).

Formalize requirements in Simulink Models by using Requirements Table Block

You can formalize your requirements in Simulink models by using the new Requirements Table block. You establish requirements with logical expressions that use simulation data to check that the actual model behavior matches required model behavior. Requirements defined in the block associate with a requirement set in Requirements Toolbox. For more information about specifying requirements in Requirements Table blocks, see [Use a Requirements Table Block to Create Formal Requirements](#).

This table shows an example of a requirement in the block. The requirement checks if the input data u is greater than or equal to 0 . If this condition is satisfied, the block checks if the input data $y1$ and $y2$ are greater than or equal to 0 .

Requirements		Assumptions			
Index	Summary	Precondition	Postcondition		Action
1	Requirement 1	$u > 0$	$y1 \geq 0$	$y2 \geq 0$	[1,2,3]

For more information on additional Requirements Table block features, see [Formalize Requirements in Simulink Models](#).

Run, link, and view verification status of test cases created in MATLAB Unit Test

You can now verify requirements by authoring MATLAB unit tests, creating links to the requirements, and running the tests by using the **Requirements Editor** or `slreq.ReqSet.runTests`. You can view the verification status by using the Requirements Perspective and the **Requirements Editor**. For more information, see [View Verification Status for MATLAB Code Lines](#).

For an example, see [Verify a MATLAB Algorithm by Using Requirements-Based Tests](#).

Track changes to Simulink Test test cases

You can track test changes in the Test Manager in Simulink Test™ and view the impacted requirements in the **Requirements Editor**. When you view requirements or links by clicking **Show Requirements** or **Show Links**, the **Requirements Editor** highlights the impacted requirements. You can run multiple test cases that have changed or unchanged requirements in a single run. For more information, see [Review Changes to Requirements and Test Objects](#).

For an example, see [Track Changes to Test Cases in Requirements Editor](#).

Run favorite commands from the Requirements Editor quick access toolbar

In R2022a, you can create favorite commands to run a group of MATLAB language statements from the Favorite Command menu in the **Requirements Editor** quick access toolbar. For more information, see [Access Frequently Used Features and Commands from the Requirements Editor](#).

Use callbacks to execute code when you load and save requirement set files

You can now use Requirements Toolbox callbacks to execute code when you load and save requirement sets. The `PostLoadFcn` callback executes when you load the requirement set and the `PreSaveFcn` callback executes when you save the requirement set. You can assign code to the callbacks in the **Requirements Editor** or at the MATLAB command line. For more information, see [Execute Code When Loading and Saving Requirement Sets](#).

Use callbacks to execute code when you import or update requirements

In R2022a, you can add code that automatically executes when you import requirements from third-party tools. The `PreImportFcn` callback executes before you import requirements and the `PostImportFcn` executes after you import requirements. The callbacks also execute before and after you update requirements. You can assign code to these callbacks to, for example, apply attribute mappings and make changes to the requirements hierarchy and indices. For more information, see [Use Callbacks to Customize Requirement Import Behavior](#).

Customize requirement index numbering

You can now customize requirement index numbering for requirements, referenced requirements, or justifications. You can disable requirement index numbering to omit requirements from the numbered hierarchy list. However, the requirements remain in the same place in the hierarchy. You can also specify an integer value to set the index to. For more information, see [Customize Requirement Index Numbering](#).

Enhanced support for IBM DOORS Next integration

In R2022a, when you import requirements from IBM® DOORS® Next, requirements that are not functional import as referenced requirements with `Type` set to `Container` or `Informational`.

Additionally, when you import requirements from IBM DOORS Next, Requirements Toolbox disables index numbering for requirements that are not headings. For more information, see [Import Requirements from IBM DOORS Next](#)

You can also use a global configuration when configuring a MATLAB session for integration with IBM DOORS Next. For more information, see [Configure IBM DOORS Next Session](#).

Support for hierarchical merge changes

You can now use the `mlAutoMerge` tool to merge non-conflicting changes between two versions of a requirement set when you promote or demote a requirement, or move a requirement under a different parent requirement. However, you cannot merge changes when you change the requirement index but keep it under its current parent requirement.

For more information, see [Three-Way AutoMerge Solution for Requirement Set and Link Set](#).

Programmatic interface enhancements

- You can now programmatically open and close the **Requirements Manager** app in a Simulink model by using `slreq.openRequirementsManager` and `slreq.closeRequirementsManager`.
- You can now programmatically synchronize backlinks in external documents to match links in a Requirements Toolbox link set by using `updateBacklinks`. For more information about backlinks, see [Manage Navigation Backlinks in External Requirements Documents](#).

Functionality being removed or changed

Changes to Type property and index numbering for imported IBM DOORS Next requirements

Behavior change

If you imported requirements from IBM DOORS Next prior to R2022a and then update the requirements using R2022a, Requirements Toolbox sets the `Type` property to `Container` or `Informational` for requirements with the external type `Heading` and `Information`, respectively. Additionally, index numbering is disabled except for requirements with the external type `Heading`. For more information, see [Import Requirements from IBM DOORS Next](#).

Changes to the index numbering can cause change issues for requirements that have links. For more information, see [Track Changes to Requirement Links](#).

Changes to update behavior for requirements imported from ReqIF files

Behavior change

If you imported requirements from a ReqIF file using version R2018b or prior and then update the requirements using R2022a, additional ReqIF attributes might be imported as Requirements Toolbox custom attributes. For more information, see [Import Requirements from ReqIF Files](#).

Adding custom attributes can cause change issues for requirements that have links. For more information, see [Track Changes to Requirement Links](#).

R2021b

Version: 1.8

New Features

Bug Fixes

Visualize traceability between Model-Based Design items with a traceability diagram

You can visualize the traceability structure of requirements and other Model-Based Design items by using the Traceability Diagram. A traceability diagram graphically displays the links between an originating Model-Based Design item, such as a requirement, and the items linked to it, such as other requirements or Simulink blocks or tests. For more information, see [Visualize Links with a Traceability Diagram](#).

Traceability diagrams also allow you to visually inspect the requirements allocation in a requirement set, which helps you analyze whether the design and tests implement and verify high-level requirements. You can also visualize indirect links to assess how a change impacts and propagates between Model-Based Design items. For more information, see [Assess Allocation and Impact](#).

Open the Requirements Editor from the MATLAB and Simulink apps galleries

You can now open the Requirements Editor from the MATLAB and Simulink apps galleries.

In MATLAB, select the **Apps** tab. In the **Verification, Validation, and Test** section, click **Requirements Editor**.

In Simulink, select the **Apps** tab. In the **Model Verification, Validation, and Test** section, click **Requirements Editor**.

You can also add the Requirements Editor to the MATLAB or Simulink quick access toolbars from the **Apps** tab by right-clicking the **Requirements Editor** app and selecting **Add to Quick Access Toolbar**. For more information, see [Customize MATLAB Toolbars and Access Frequently Used Features and Commands in Simulink \(Simulink\)](#).

Customize the Requirements Editor quick access toolbar

In R2021b, you can use the Requirements Editor quick access toolbar to quickly perform actions. By default, you can cut, copy, and paste items and access relevant documentation. Additionally, you can add actions to the quick access toolbar by right-clicking a toolstrip button and selecting **Add to Quick Access Toolbar**. You can remove non-default actions from the quick access toolbar by right-clicking the toolbar and selecting **Remove from Quick Access Toolbar**. The quick access toolbar preferences are persistent across sessions.

For more information, see [Customize the Requirements Editor Quick Access Toolbar](#).

Merge changes to requirement set and link set files from multiple users

In R2021b, you can now merge changes from multiple users working on the same requirement set and link set files in different branches in Git™. For more information, see [Three-way AutoMerge Solution for Requirement Set and Link Set](#). If the changes do not conflict, the files are merged when you click **Merge** in the Branches dialog box. For more information, see [Branch and Merge with Git](#). If the changes conflict, you can view and resolve the conflicts manually, and then merge the files.

Enhanced support for importing and exporting ReqIF files

In R2021b, you can export the Simulink Requirements requirement type for requirements that use a custom type when you export requirements to a ReqIF file. You must define and use custom requirement types and use the generic mapping during export. For more information, see [Using the Generic Mapping During Export](#).

Additionally, you can link Simulink Requirements requirements to items in MATLAB or Simulink, export the requirements to a ReqIF file, and then re-import the file to reconstruct the links between the requirements and the original MATLAB or Simulink objects. When you link a requirement to an item in MATLAB and Simulink and then export the requirements to a ReqIF file, Simulink Requirements creates a proxy object for the linked item in the exported file. If the linked item is one of the supported types, the proxy object also has a type value that describes the linked object type. For a list of supported types, see [Exporting Links](#). In R2021b, if you re-import this ReqIF file, the software reconstructs the links between the requirements and the items specified by the proxy objects of the supported types. For more information, see [Importing Links from a ReqIF File Generated by Simulink Requirements](#).

Register a custom callback function to authenticate IBM DOORS Next or other web-based servers

You now authorize web-based services such as IBM DOORS Next that require advanced authentication that the default authentication process does not support by registering custom callback functions. Use `rmipref(LoginProvider="authenticationFunction")` to register the authentication callback function `authenticationFunction`. For more information, see `rmipref` and the 'LoginProvider' input argument. When you authenticate your MATLAB session, you can only create direct links to requirements in IBM DOORS Next to items in MATLAB and Simulink and cannot import the requirements. For more information, see [Directly Linking DOORS Next Requirements](#).

Similarly, you can authenticate an `oslc.Client` object that requires advanced authentication by using `setCustomLoginProvider`. You can then use `oslc.Client` object functions to integrate third-party OSLC server-based requirements management tools with Simulink Requirements. You can also get the registered callback function for an `oslc.Client` object by using `getCustomLoginProvider`.

Import a subset of requirements from an IBM Rational DOORS module

You can now import a subset of requirements from a requirements module in IBM Rational® DOORS by applying a view filter to the module. When you import the requirements to Simulink Requirements, the import process imports only the requirements that satisfy the currently applied filter and their parents. You can store the filter so that you can apply it if you update the requirement set.

For more information, see [Import Requirements from IBM Rational DOORS](#) and [Import Requirements from IBM Rational DOORS](#) by using the API.

R2021a


Version: 1.7

New Features
































Bug Fixes

Manage requirements with updated Requirements Editor

The Requirements Editor now contains a toolstrip to help you complete common requirements tasks more easily. The Requirements Editor toolstrip contains components that were previously available in menus and toolbars.

Some toolstrip actions are contextual and only become enabled when you select a relevant item. For example, the  **Delete** button is dimmed unless you select a requirement that can be deleted.

Menu Bar Item	Toolstrip Equivalent
File > <ul style="list-style-type: none"> • New Requirement Set • Import • Open (Ctrl+O) • Save (Ctrl+S) • Save As • Export to Previous • Save All • Export to ReqIF 	<ul style="list-style-type: none"> •  New Requirement Set •  Import •  Open (Ctrl+O) •  Save (Ctrl+S) •  Save > Save As •  Save > Export to Previous •  Save > Save All •  Export > ReqIF
Edit > <ul style="list-style-type: none"> • Cut (Ctrl+X) • Copy (Ctrl+C) • Paste (Ctrl+V) • Delete (Del) 	<ul style="list-style-type: none"> •  Cut (Ctrl+X) •  Copy (Ctrl+C) •  Paste (Ctrl+V) •  Delete (Del) for requirements or  Delete (Del) for links
Display > <ul style="list-style-type: none"> • Comments • Implementation Status • Verification Status • Change Information 	<ul style="list-style-type: none"> •  Information > Comments •  Columns > Implementation Status •  Columns > Verification Status •  Information > Change Information
Analysis > <ul style="list-style-type: none"> • Show code traceability • Refresh 	<ul style="list-style-type: none"> •  Information > Code Traceability •  Refresh

Menu Bar Item	Toolstrip Equivalent
Report > <ul style="list-style-type: none"> • Generate Report • Generate Traceability Matrix 	<ul style="list-style-type: none"> •  Export > Generate Report •  Traceability Matrix
Help > <ul style="list-style-type: none"> • Authoring Requirements • Simulink Requirements Documentation 	 Help > <ul style="list-style-type: none"> • Authoring Requirements • Help
View drop-down menu: <ul style="list-style-type: none"> • Requirements • Links 	<ul style="list-style-type: none"> •  Show Requirements •  Show Links
Toolbar <ul style="list-style-type: none"> • New Requirement Set button () • Open Requirement Set button () • Save Requirement Set/Link Set button () • Add Requirement () • Promote Requirement () • Demote Requirement () • Cut Requirement () • Copy Requirement () • Paste Requirement () • Delete Requirement/Link () • Add Justification () • Refresh () 	<ul style="list-style-type: none"> •  New Requirement Set •  Open (Ctrl+O) •  Save (Ctrl+S) •  Add Requirement and  Add Requirement > Add Child Requirement •  Promote Requirement •  Demote Requirement •  Cut (Ctrl+X) •  Copy (Ctrl+C) •  Paste (Ctrl+V) •  Delete (Del) or  Delete (Del) •  Add Requirement > Add Justification •  Refresh

You can view more requirement or link properties by adding columns to the Requirements Editor view when you select  **Columns > Select Attributes**.

Traceability Matrix: Create matrix with multiple artifacts manually or by API

In R2021a, you can create a traceability matrix with more than two artifacts when, in the **Select Artifacts** dialog, you select **Select Multiple Artifacts**. You can drag and drop artifacts to the top and left artifacts list and then generate the matrix. After you generate the matrix, you can add, remove, or reorder the artifacts. To read more about creating a traceability matrix with more than two artifacts, see [Track Requirement Links with a Traceability Matrix](#).

You can also create a traceability matrix by using `slreq.generateTraceabilityMatrix`. You can choose the artifacts manually in the Traceability Matrix window or programmatically by creating a template options structure using `slreq.getTraceabilityMatrixOptions` and entering the options as an argument when you use `slreq.generateTraceabilityMatrix`. See [Programmatically Generate a Traceability Matrix](#). Additionally, you can use `slreq.getTraceabilityMatrixOptions` to get the artifacts used in the current generated Traceability Matrix. See [Get Artifacts from the Selected Traceability Matrix](#).

Enhanced support for importing and exporting ReqIF files

In R2021a, after you import requirements from a ReqIF file, you can map the `SpecObjectTypes` to requirement types in Simulink Requirements. See [Map SpecObjectTypes to Requirement Types](#).

When you import requirements from a ReqIF file, link them to Model-Based Design items such as Simulink blocks, and then export the requirement set to a ReqIF file, the exported `SpecObject` items have a `SpecObjectType` value that corresponds to the object type in MATLAB or Simulink. For more information, see [Exporting Links](#).

You can now export a ReqIF file using a generic mapping regardless of which import mapping was used. The generic ReqIF mapping export now includes links. See [Using the Generic Mapping During Export](#).

Enhancements for linking with requirements in IBM DOORS Next

In R2021a, when you create a direct link from an item in MATLAB or Simulink to a requirement in IBM DOORS Next, you can choose to create the link in the module context.

Additionally, when you create direct links or link to an imported requirement that references a requirement in IBM DOORS Next, you can insert a backlink in IBM DOORS Next to navigate from the requirement in DOORS to the linked item in MATLAB or Simulink. If you created the direct link in the module context, or you imported the requirements from a module, then the backlink is also inserted in the module context. If you did not insert a backlink when you created the forward link or if the backlink was deleted, you can insert backlinks in your IBM DOORS Next project by right-clicking the link set in the Requirements Editor or Requirements Browser and selecting **Update Backlinks**.

See [Requirements Traceability with IBM DOORS Next](#) and [Manage Navigation Backlinks in External Requirements Documents](#).

Improved loading of link information

In R2021a, in order to improve link loading performance, loading one of the artifacts or a requirement set will correspond to the explicitly requested requirements data. For more information, see Load Link Information.

oslc Object: Access OSLC services with command-line client

You can now use a command-line OSLC client in MATLAB to communicate with OSLC providers. You can use the client to integrate third-party requirements management, quality management, and change management tools with Simulink Requirements.

After you instantiate a client and provide authentication credentials, you can query a service provider for existing OSLC resources and fetch all properties of the resource. You can commit changes to the provider from MATLAB when you add, update, or delete resources.

For more information, see the documentation for these classes:

- `oslc.Client`
- `oslc.core.CreationFactory`
- `oslc.core.Dialog`
- `oslc.core.QueryCapability`
- `oslc.rm.Requirement`
- `oslc.rm.RequirementCollection`
- `oslc.cm.ChangeRequest`
- `oslc.qm.TestCase`
- `oslc.qm.TestExecutionRecord`
- `oslc.qm.TestPlan`
- `oslc.qm.TestResult`
- `oslc.qm.TestScript`

Programmatic interface enhancements

- Get an object for the currently selected item in the Requirements Editor or Requirements Browser by using `slreq.getCurrentObject`.
- Use `slreq.getExternalURL` to get a navigation URL and object label for a linkable item to navigate from an external browser to a link source or destination, requirement, test, or Simulink model element. In the Requirements Editor or Simulink Editor, you can copy the navigation URL to your clipboard by right-clicking a requirement, model element, or Simulink data dictionary entry and selecting **Copy URL to Clipboard** or **Requirements > Copy URL to Clipboard**.

R2020b

Version: 1.6

New Features

Bug Fixes

Traceability Matrix: Select multiple cells, create and delete links in batch, track changes, and export

You can select multiple cells in the Traceability Matrix by clicking and dragging, or pressing **Shift** and clicking the cells. You can toggle cells by pressing **Ctrl** and clicking the cells.

You can also create or delete multiple links at a time by selecting multiple cells and clicking **Create Link** or **Remove Link**.

You can track changes in the Traceability Matrix when requirements change. When a requirement changes, you can highlight the cell and the link artifacts in the row and column of the matrix that correspond to the changed link by clicking **Highlight Missing Links > Highlight Changed Links**. Additionally, you can:

- Clear change issues directly from the Traceability Matrix by clicking **Clear Change Issue**.
- Clear change issues for multiple cells at a time.
- Filter the matrix to only show items that have a change issue by clicking **Highlight Missing Links > Show Changed Links Only**.

You can export the current state of the Traceability Matrix to an HTML report or to a MATLAB variable that contains table data or by using **Export > Generate HTML Report** or **Export > Create MATLAB Variable**, respectively.

Additionally, when you create a Traceability Matrix with a System Composer model as an artifact, the Traceability Matrix hierarchy now contains any System Composer views associated with the model.

For more information, see [Track Requirement Links with a Traceability Matrix](#).

Customize link sets and links with custom attributes programmatically or in the Requirements Editor

You can now create custom attributes and set their corresponding values for link sets and links. You manage custom attributes for link sets and links in the same way you manage custom attributes for requirement sets and requirements. In the Requirements Editor, use the **Custom Attribute Registry** and **Custom Attributes** panes to manage custom attributes for link sets and links, respectively.

Additionally, you can now manage custom attributes for links and requirements programmatically:

- View, add, update, and remove attributes for a LinkSet or a ReqSet using `inspectAttribute` or `inspectAttribute`, `addAttribute` or `addAttribute`, `updateAttribute` or `updateAttribute`, and `deleteAttribute` or `deleteAttribute`, respectively.
- View custom attribute names for a LinkSet with the `CustomAttributeNames` property. Previously, you could view custom attribute names for a ReqSet with the `CustomAttributeNames` property.
- Find a Link that matches a custom attribute value with `slreq.find` or the `find` method. Previously, you could find a Requirement that matches a custom attribute value with `slreq.find` or the `find` method.
- View and change the custom attribute value of a Link with `getAttribute` and `setAttribute`. Previously, you could view and change the custom attribute value of a Requirement using `getAttribute` and `setAttribute`.

For more information on managing custom attributes in the Requirements Editor, see [Customize Requirements with Custom Attributes](#) and [Customize Links with Custom Attributes](#).

For more information on managing custom attributes programmatically, see [Manage Custom Attributes for Requirements by Using the Simulink® Requirements™ API](#) and [Manage Custom Attributes for Links by Using the Simulink® Requirements™ API](#).

Verify requirements with MATLAB-based Simulink Tests

In R2020b, you can link requirements to Simulink Test test cases written in the MATLAB Editor when you create a MATLAB-based Simulink test. See [Test Models Using MATLAB-Based Simulink Tests \(Simulink Test\)](#). You can use the Traceability Matrix, MATLAB Editor, and Simulink Test Test Manager to create these links. Additionally, you can view the verification status of requirements in the Requirements Editor and Requirements Browser when you link requirements to these test cases.

User interface enhancements

- You can clear more than one change issue at a time in the Requirements Editor when you select multiple links, right-click them, and select **Clear Issue** from the context menu.

Programmatic interface enhancements

- Move and copy a requirement or justification: You can move a single Requirement or Justification programmatically by using `move` or `moveDown` or `moveUp`, and `moveUp` or `moveDown`, respectively. You can copy a single Requirement or Justification by using `copy` or `copy`, respectively.
- View Index property: You can now view Index, a Requirement property, programmatically.

Model Testing Dashboard

The Model Testing Dashboard collects and displays metric data on the status and quality of your requirements-based testing. 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 in 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” \(Simulink Check\)](#). To use this feature, you must have a Simulink Check™ license.

R2020a

Version: 1.5

New Features

Bug Fixes

Traceability Matrix: Review and analyze links in a matrix format

You can review and analyze requirements, links, and model entities by creating a traceability matrix. A traceability matrix summarizes traceability links and their sources and destinations, such as requirements, models, or test cases. You can:

- Identify missing or incorrect links.
- Create missing links.
- Fix incorrect links.
- Display requirements that are linked to model elements.
- Inspect links by navigating to their sources or destinations.
- Filter by requirement type, link type, or model hierarchy.
- Confirm model and requirement completeness.

To create a traceability matrix,

- 1 In the **Requirements** tab, select **Share > Open Requirements Traceability Matrix**.
- 2 Click **Add** to create a matrix.

For more information, see [Track Requirement Links with a Traceability Matrix](#).

Include results from external sources in verification status

Requirements-based tests can run outside of the Simulink environment, such as using MATLAB scripts or continuous integration servers, such as Jenkins®. You can include external test results in the requirements verification status by linking to scripts or results files. To include external results in requirements verification status:

- 1 Create a link type to your test script or result file. Use plug-ins for the format, such as TAP, JUnit®, or Excel®.
- 2 Link from the requirement to the test script or result file.
- 3 View the verification status in the in the **Verified** column of the Requirements Editor and Requirements Browser.

For more information, see [Include Results from External Sources in Verification Status](#).

View, link, and sync IBM Rational DOORS Next Generation requirements

You can now capture a requirements module or subset of requirements directly from IBM Rational DOORS Next Generation (DNG). In Simulink Requirements, you can link these requirements to your model or MATLAB code, and refresh the requirement set if the source in DNG changes. You can capture requirements by specifying:

- A DNG module
- A DNG query

Capturing DNG items from a module or query enables you to directly use DNG requirements as a source, without intermediate ReqIF conversions. You can work with captured requirements without an active connection to the DNG server.

For an example, see the section **Capture DNG Collections into Simulink Requirements Set** in Link with Requirements in DOORS Next Generation Project.

Link from ports in System Composer

You can link from requirements to ports in a System Composer architecture model.

User interface enhancements

- You can select multiple requirements and links in the Requirements Browser and Requirements Editor. You can link and delete multiple selected items.
- You can display a requirements description and rationale in the Requirements Browser.

Programmatic interface enhancements

- Determine links in a link set: You can return the links in a `LinkSet` using `getLinks`.
- Navigate to link source and destination: You can navigate to a `Link` source or destination using `slreq.show`.
- `slreq.find` has been re-implemented to improve performance when operating on large sets of requirements and links. For example, this command to find a single link in a set of approximately 1200 links runs approximately 2 orders of magnitude faster:

```
function timingTest
    slreq.find('type', 'Link', 'SID', 63)
end
```

The approximate execution times are:

- R2019b: 1.09 s
- R2020a: 0.01 s

The code was timed on a Windows® 10, Intel Core® i7-8650U CPU @ 1.90 GHz system by calling the function `timingTest`.

R2019b

Version: 1.4

New Features

Bug Fixes

Contextual Tabs in the Simulink Toolstrip for Simulink Requirements

In R2019b, the Simulink Toolstrip replaces Simulink menu bar. See “Simulink Toolstrip: Access and discover Simulink capabilities when you need them” (Simulink) for more details. The location of several Simulink Requirements features and buttons are located in contextual tabs that appear when you need them.

- 1 Click on **Apps**.
- 2 Under **Verification, Validation, and Test**, you can open the **Requirements Manager** tab or the **Requirements Viewer** tab.

The **Requirements Manager** tab also appears when you open the Requirements Perspective.

Improved Requirements Exchange with ReqIF

You can import multiple source specifications from Requirements Interchange Format (ReqIF) files. When you import ReqIF files that contain multiple source specifications, you can choose to:

- Import only a single ReqIF source specification into a requirement set.
- Combine ReqIF source specifications into one requirement set. After importing, each specification corresponds to a node in the requirement set. You can update nodes independently if the content of the ReqIF file changes.
- Import each ReqIF source specification into a separate requirement set. You select a folder, and the import operation creates multiple requirement set files in that folder.

Also, import supports exchange links in ReqIF files, which are represented in link sets.

For more information, see Import Requirements from ReqIF Files.

Navigation Support for Baselines in IBM Rational DOORS 9

You can now link requirements to a baseline version of a DOORS 9 module. A baseline version remains consistent if the most recent DOORS 9 module changes. If your model or code reflects baseline requirements, link to the baseline DOORS 9 version.

Set the baseline version using `slreq.cmConfigureVersion`, and query the active baseline version using `slreq.cmGetVersion`.

Enhanced Web View Reports for Reviewing Requirements and Links

With Simulink Report Generator™, you can generate a Simulink Web View of your model that now includes requirements and requirements property information. In the Simulink model, open the Requirements Perspective. In the **Requirements** tab, click **Share > Export Model to Web View**.

For more information on Simulink Web Views, see Web Views (Simulink Report Generator).

Programmatically Modify Link Source and Destination

In R2019b, you can use the new `setSource` and `setDestination` methods to modify link source and destination targets.

External Source Document Change Notification

If you have requirement references to external source documents, the Requirements Editor now notifies you of updates to the source documents. If a source document has changed since the most recent import or update operation:

- The top import node displays a notification icon.
- The **Data Exchange** pane displays a status message.

For more information, see Update Imported Requirements.

R2019a

Version: 1.3

New Features

Bug Fixes

Round Trip Workflow with Requirements Tools: Exchange requirements with third-party requirements tools through ReqIF

Simulink Requirements now supports round trip importing and exporting using the open standard ReqIF (Requirements Interchange Format) file format. The round trip workflow can be broadly subdivided into three categories:

- **Import:** Use attribute mappings to define how the attributes of a requirement in Simulink Requirements relate to the attributes of an external requirement represented by a different third-party requirements management application. When you import requirements from a ReqIF file, Simulink Requirements detects the source tool for the ReqIF file and applies a pre-configured attribute mapping. You can also customize and save attribute mapping templates.
- **Edit:** Provide supplementary information to imported requirements by adding custom attributes and editing requirements content.
- **Export:** Export locally authored requirements and referenced requirements with local edits and additional attributes to ReqIF files.

For more information, see Round Trip Workflows with ReqIF Files.

Navigation Backlinks for Third-Party Requirements: Manage backlinks in external requirement documents that enable you to navigate to Simulink model elements

When you work with requirements from an external requirements document, you can insert, delete, and modify navigation backlinks in the external requirements document that match the set of links stored by Simulink Requirements. Use these navigation backlinks to navigate from the external document to Simulink and MATLAB artifacts. For more information, see Manage Navigation Backlinks in External Requirements Documents.

Integration with System Composer: Link requirements to architecture models in System Composer

Link and trace requirements to architecture components in System Composer models.

Usability Enhancements

- Reorder requirements in the Requirements Spreadsheet by using drag-and-drop functionality.
- Link requirements to area annotations on the Simulink Editor by dragging and dropping. A badge appears on the annotation to show that a link was created. Click the badge and click **Show** to display a requirement annotation on the Simulink Editor.
- All link sets on the MATLAB or project path with incoming links to a requirement set are loaded automatically when you load the source and destination artifacts. When you close all source and destination artifacts related to loaded link sets, the link sets are unloaded when no requirements reference any of the links in the link sets. For more information, see Review Requirement Links.
- Simulink Requirements now uses short names on the MATLAB path as the default path setting in the **Document file reference** option on the **Selection Linking** tab of the Requirements Settings dialog box.

-
- Starting from R2019a, MATLAB uses HTTPS as the default connection protocol. HTTP connections are disabled by default. Run `rmi('httpLink')` at the MATLAB command prompt to enable HTTP to navigate from third-party tools and web reports to artifacts in Simulink Requirements.

Simulink Requirements contextual tab in the Simulink Toolstrip

In R2019a, you have the option to turn on the Simulink Toolstrip. For more information, see Simulink Toolstrip Tech Preview replaces menus and toolbars in the Simulink Desktop (Simulink).

The Simulink Toolstrip includes contextual tabs — they appear only when you need them. The Simulink Requirements contextual tabs include options for completing actions that apply only to Simulink Requirements.

- Access the **Requirements Manager** tab by entering the Requirements Perspective View.
- Access the **Requirements Viewer** tab from the **Apps** tab on the Simulink Toolstrip.

Code Traceability: Include outgoing traceability links from external MATLAB code referenced by MATLAB Function Blocks in code generation report

The code generation report from Embedded Coder® now includes requirement traceability links for MATLAB files saved externally from the Simulink model and referenced from MATLAB Function blocks in Simulink. For more information, see Traceability Links in Code Generation Reports.

Report Generation Enhancements: Include change information in requirements report

You can include requirements and links change information in the generated requirements report. All change issues are summarized in a table in the Requirements Report Appendix.

R2018b

Version: 1.2

New Features

Bug Fixes

External Rich Text Editor: Use text editing and formatting features of Microsoft Word to author and edit rich text requirements

To edit rich text content in the **Description** and **Rationale** fields of your requirements, open Microsoft Word from within the Requirements Editor or the Requirements Perspective View. You can:

- Spell-check requirements content.
- Resize images.
- Insert and edit equations.
- Insert and edit tables.

For more information, see [Author and Edit Requirements Content by Using Microsoft Word](#).

Configurable Requirement and Link Types: Define custom requirement and link types for improved classification and analysis

With Simulink Requirements, you can now classify requirements by type. Define your own custom requirement types or use these types:

- **Functional:** Classify requirements that are meant to be implemented or verified in your Model-Based Design workflow. Functional requirements contribute to the Implementation and Verification status metrics of the requirement set that they are in.
- **Container:** Group requirements. Container requirements themselves do not contribute to the Implementation and Verification status metrics of the requirement set that they are in. However, all the Functional requirements under a Container requirement contribute to the status metrics.
- **Informational:** Provide supplemental information. Informational requirements and all requirements under them do not contribute to the Implementation and Verification status metrics of the requirement set that they are in.

You can also define custom requirement and link types by using the `sl_customization.m` file. See [Define Custom Requirement and Link Types](#).

Incoming Link Resolution: Automatically load and display incoming links when opening artifacts stored in Simulink Projects

When you work with requirements sets in Simulink Projects, all the link sets related to your requirements sets are loaded automatically when you load the source and destination artifacts. When you close all source and destination artifacts related to loaded link sets, the link sets are automatically unloaded when no requirements reference any of the links in the link sets. For more information, see [Manage Requirements Across a Team](#).

IBM Rational DOORS Next Generation Integration: Establish traceability to requirements stored in DOORS Next Generation project areas with configuration management enabled

If you are working with IBM Rational DOORS Next Generation projects that have configuration management enabled, you can create links from Simulink model elements to requirements stored in streams or changesets. View link summaries and batch redirect existing links from a stream or a

changeset in an IBM Rational DOORS Next Generation project to another stream or changeset in the same project. For more information, see Requirements Traceability with IBM Rational DOORS Next Generation.

Extended Requirements Management Capabilities: Organize, view and manage requirements content with improved indexing and updating of imported requirements content

You can now update requirements imported from Microsoft Excel spreadsheets with non-unique **Custom Id** fields. When you work with the Requirements Editor or in the Requirements Perspective view, your window settings are saved across sessions. The saved settings are:

- Displayed columns (Requirements and Links view)
- Item sort order
- Current view (Requirements or Links view)

For more information on view settings, see View or Hide Columns in the Requirements Editor.

R2018a

Version: 1.1

New Features

Bug Fixes

Requirements Import with ReqIF: Import requirements information from third-party tools supporting the ReqIF standard such as IBM Rational DOORS Next Generation or Siemens Polarion

Use the Document Import Wizard to import and work with requirements from third-party tools supporting the Requirements Interchange Format (ReqIF) file format. For more information, see [Import Requirements from ReqIF Files](#).

IBM Rational DOORS Import Wizard: Work with and manage requirements from DOORS by importing them as read-only references

Use the Document Import Wizard to import and work with requirements from IBM Rational DOORS from within Simulink Requirements. For more information, see [Import Requirements from IBM Rational DOORS](#).

Batch Test Execution: Verify requirements by launching linked tests

If you have Simulink Test test cases or Simulink Design Verifier Proof Objective blocks linked to requirements, you can run tests from the Requirements Editor. Right-click the requirement node and select **Run Tests**. In the dialog box, select the tests to run. The Verification Status metric updates after the tests complete. For more information, see [Review Requirement Verification Status Metrics Data](#).

Link Set Comparison: Track changes to requirements links by comparing Link Set files

Use the Link Set Comparison tool to compare link set files in your project and track changes to requirements links. For more information, see [Track Changes to Requirements Links](#).

Simulink and Stateflow Annotation Conversion: Convert existing annotations to requirements to improve traceability, analysis, and management

Convert annotations in your Simulink models and Stateflow® charts to requirement objects that you can manage from within Simulink Requirements. For more information, see [Convert Simulink Annotations to Requirements](#).

Test Harness Support: Work with Simulink Test test harnesses from the requirements perspective

You can now enter and work with Simulink Test test harnesses from within the requirements perspective.

Requirements Report Generation: Generate requirements reports in PDF and HTML formats

You can now generate requirements reports in PDF and HTML formats. For more information, see Report Requirements Information.

R2017b

Version: 1.0

New Features

Introduction to Simulink Requirements: Author, analyze, and manage requirements

With Simulink Requirements, you can author and manage requirements within Simulink. Create requirement sets, organize requirements in hierarchies, and link requirements to model entities, tests, or other requirements. You can author requirements in the Simulink Editor or the Requirements Editor. You can also import requirements from external Microsoft Word or Microsoft Excel documents.

Quickly create implementation-type links by dragging requirements from the Requirements Browser to your model. Annotate your model diagrams with requirements information and navigate to the source requirement. Trace requirements to associated model entities and tests. To help you quantify completeness of your development activities in Model-Based Design, summarize implemented and verified requirements in your set and include this information in a report. For more information, see *Getting Started with Simulink Requirements*.

Requirements Editor and Browser: Author, edit, and organize requirements within Simulink

You can author, edit, and organize requirements within Simulink from the Requirements Editor or the Requirements Browser in the Simulink Editor. Describe requirements with rich text and images, and organize requirements in hierarchies. The requirements view displays the hierarchy and the detailed information for each requirement. The links view displays the source, destination, and type of relationship. To perform impact analysis for your requirements, use the links view. For more information, see *Author Requirements in Simulink*.

Requirements Import Wizard: Import requirements content from Microsoft Word and Microsoft Excel documents

You can import requirements stored in external Microsoft Word and Microsoft Excel documents into Simulink Requirements. You can import requirements in two ways:

- As editable requirements objects, in which you make subsequent updates by using Simulink Requirements.
- As read-only requirements objects that reference the external document, in which you use Word or Excel to update your requirements. You can navigate to the external document content from a requirement object.


For more information, see *Import Requirements from Third-Party Tools*.

Requirements Perspective: Establish requirements - design traceability and annotate your model with requirements information in the Simulink Editor

With a new requirements user interface in the Simulink Editor, you can:

- Create links by dragging requirements to model entities.
- Annotate your model block diagram with requirements information.
- Show and hide requirements information associated with your model.

-
- Understand traceability gaps by identifying blocks without requirements badges.

To enable the Requirements Perspective, click the  icon in the lower right of the Simulink canvas, and then click the Requirements graphic. For more information, see Author Requirements in Simulink.

Requirements Change Tracking and Comparison: Automatically identify and manage changing requirements and impacts on design, models, and tests

The Requirements Editor highlights changes to requirements or links in your requirement set. You can use this information to assess potential impacts to your design, and take action. Changes to the requirement set appear in the **Change Information** section of the Requirements Editor. If a requirement change impacts your model or tests, update the related model element to meet the modified requirement. If an issue has no impact, you can clear the issue to reflect the latest revision of the requirement. For more information, see Track Changes to Requirements Links.

Implementation and Verification Status Summary: Review implementation and verification status metrics for your requirements

The Requirements Editor summarizes your requirement set implementation links in the **Implementation status** column. For a particular requirement, implementation status shows whether the requirement has an implementation link. For a parent requirement or requirement set, implementation status aggregates the implementation metrics for child requirements. For more information, see Review Requirement Implementation Status Metrics Data.

The Requirements Editor also summarizes the results of verification type links in the **Verification status** column. With a Simulink Test license, verification status summarizes the results of linked test cases. With a Simulink Design Verifier license, verification status summarizes the results of linked proof objectives. For more information, see Review Requirement Verification Status Metrics Data.

Requirements Reporting: Archive and document your requirements traceability, implementation status, and change history

From the Requirements Editor, you can generate a detailed report of your requirement set and links. You can include detail such as requirement description, rationale, implementation status, verification status, and revision history. Reports use Microsoft Word format. For more information, see Report Requirements Information.

Simulink Verification and Validation Packaging: Moved Requirements Management Interface (RMI) features from Simulink Verification and Validation to Simulink Requirements

Linking and management features that use the Requirements Management Interface from Simulink Verification and Validation™ are now included in Simulink Requirements. You can update your RMI links to use the Simulink Requirements interface. For an example, see Migrating Requirements Management Interface Data to Simulink® Requirements™.

