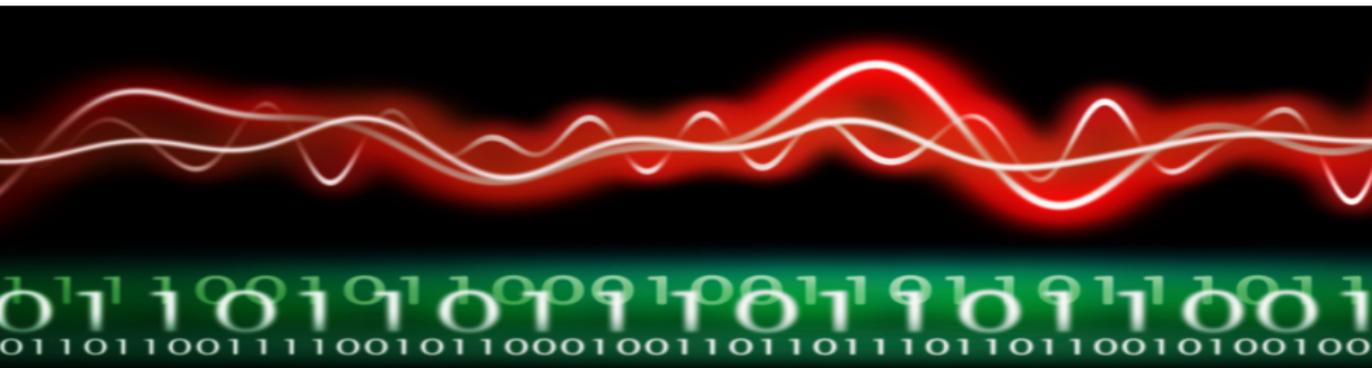


Reactis V2013

Released August 14, 2013



New Simulink Support

- ▶ Initial support for R2013a
- ▶ Support for .slx Format
 - ▶ Introduced in R2012a to replace .mdl format
 - ▶ XML-based format
- ▶ Support Simulink *configuration set reference* concept

Multiple Condition Coverage (MCC)

MCC metric tracks if all combinations of condition outcomes for a decision have been exercised.

Multiple Condition Coverage (MCC)

MCC metric tracks if all combinations of condition outcomes for a decision have been exercised. For decision $A \ \&\& \ B \ \&\& \ C$, MCC targets are:

Multiple Condition Coverage (MCC)

MCC metric tracks if all combinations of condition outcomes for a decision have been exercised. For decision $A \ \&\& \ B \ \&\& \ C$, MCC targets are:

No Short-Circuiting

A	B	C	Decision
F	F	F	F
F	F	T	F
F	T	F	F
F	T	T	F
T	F	F	F
T	F	T	F
T	T	F	F
T	T	T	T

Multiple Condition Coverage (MCC)

MCC metric tracks if all combinations of condition outcomes for a decision have been exercised. For decision $A \ \&\& \ B \ \&\& \ C$, MCC targets are:

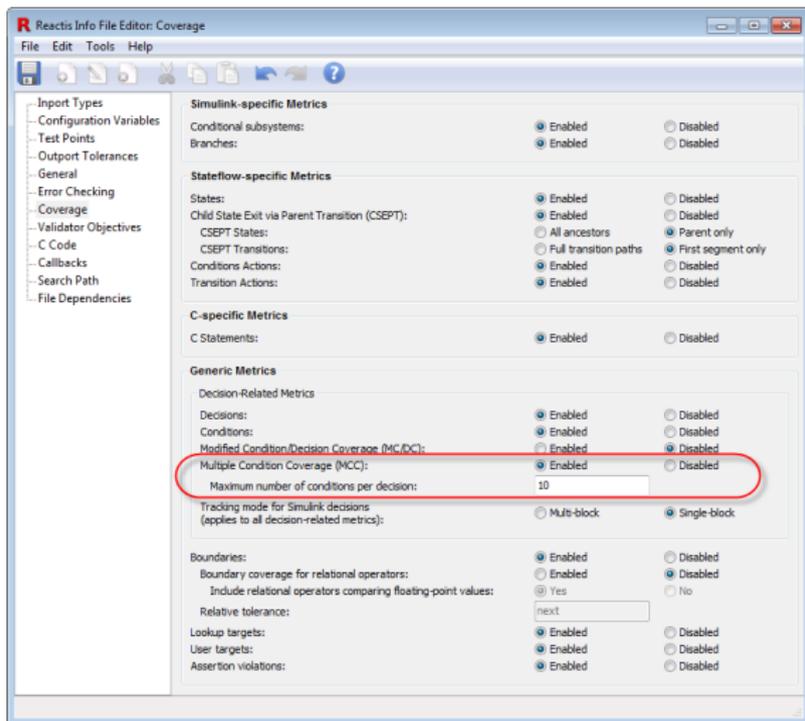
No Short-Circuiting

A	B	C	Decision
F	F	F	F
F	F	T	F
F	T	F	F
F	T	T	F
T	F	F	F
T	F	T	F
T	T	F	F
T	T	T	T

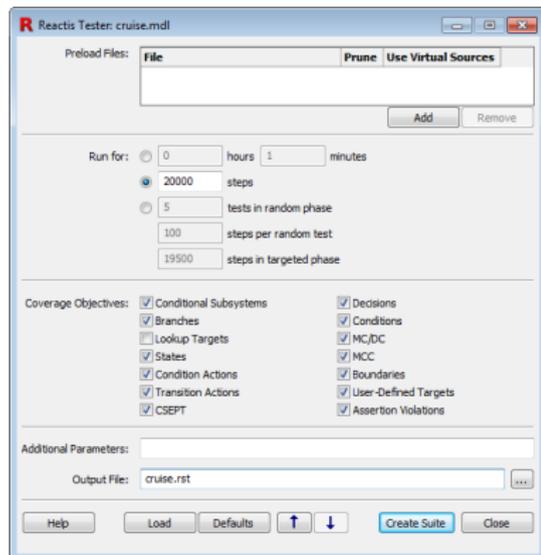
Short-Circuiting

A	B	C	Decision
F	x	x	F
T	F	x	F
T	T	F	F
T	T	T	T

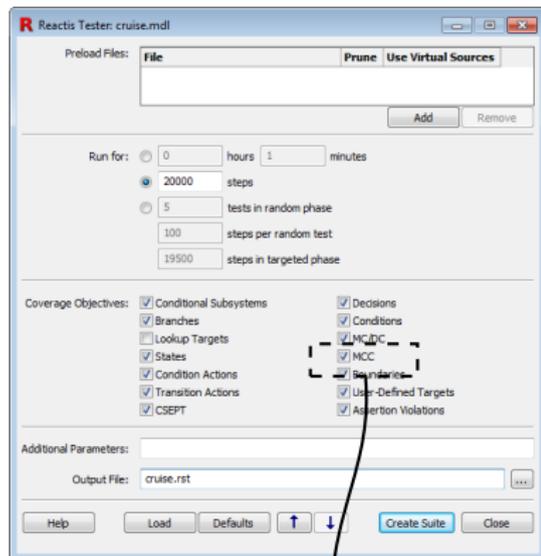
MCC Now Supported by Reactis



MCC Support in Reactis Tester

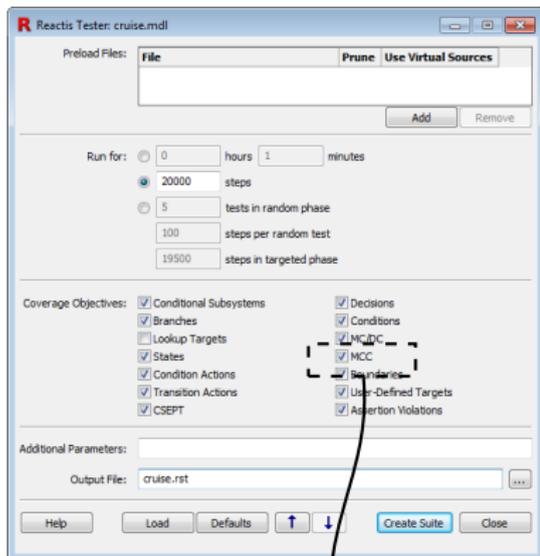


MCC Support in Reactis Tester

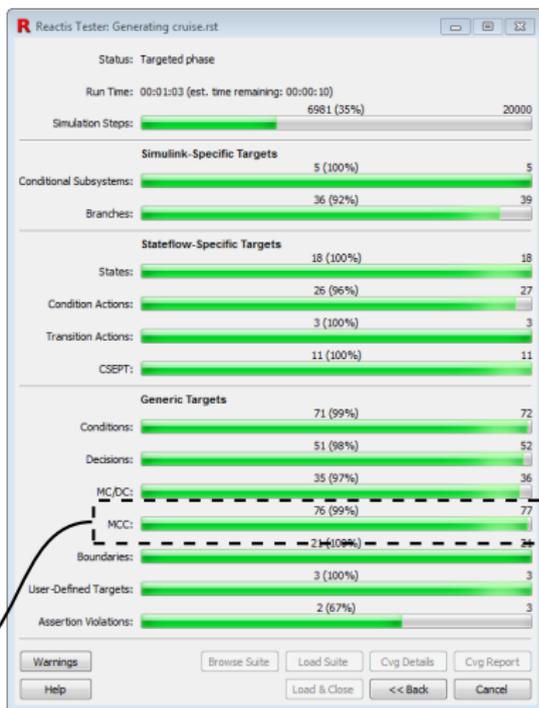


Tester attempts to cover MCC targets

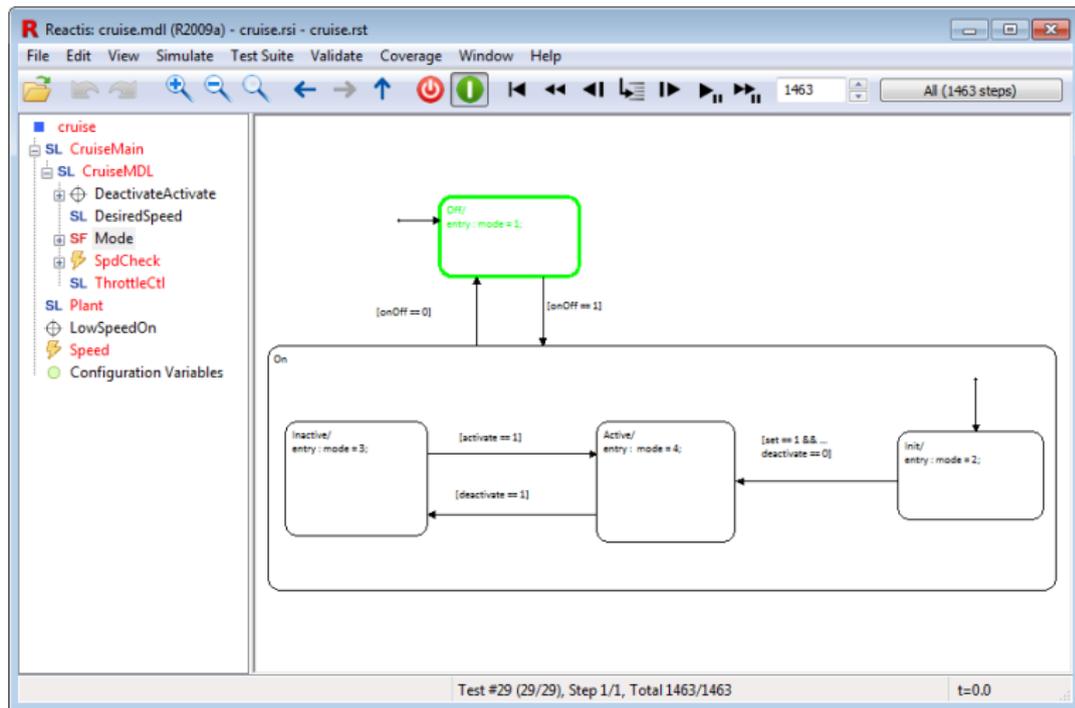
MCC Support in Reactis Tester



Tester attempts to cover MCC targets



Tracking MCC in Simulator



Tracking MCC in Simulator

The screenshot displays the Reactis simulator window for a file named 'cruise.mdl (R2009a)'. The interface includes a menu bar (File, Edit, View, Simulate, Test Suite, Validate, Coverage, Window, Help), a toolbar with navigation and simulation controls, and a status bar at the bottom showing 'Test #29 (29/29), Step 1/1, Total 1463/1463' and 't=0.0'.

On the left, a project tree lists the following components:

- cruise
 - SL CruiseMain
 - SL CruiseMDL
 - DeactivateActivate
 - SL DesiredSpeed
 - SF Mode
 - SpdCheck
 - SL ThrottleCtl
 - SL Plant
 - LowSpeedOn
 - Speed
 - Configuration Variables

The main workspace shows a state machine diagram. A state box at the top is labeled 'Off/ entry : mode = 1;'. Below it is a large state box labeled 'On'. Inside the 'On' state, there is a central state box labeled 'Decision with MCC targets'. To its right is another state box labeled 'init/ entry : mode = 2;'. Transitions are as follows:

- From 'Off' to 'On' (labeled 'On').
- From 'On' to 'Off' (labeled '[onOff == 1]').
- From 'On' to 'Decision with MCC targets' (labeled 'mode == 1;').
- From 'Decision with MCC targets' to 'On' (labeled 'mode == 0;').
- From 'init/ entry : mode = 2;' to 'Decision with MCC targets' (labeled '[set == 1 S.S. ... deactivate == 0]').
- From 'Decision with MCC targets' to 'init/ entry : mode = 2;' (labeled 'mode == 1;').
- From 'init/ entry : mode = 2;' to 'On' (labeled 'On').

The 'Decision with MCC targets' state has a self-loop transition labeled 'mode == 0;'. The 'init/ entry : mode = 2;' state has a self-loop transition labeled 'mode == 1;'. The 'On' state has a self-loop transition labeled 'On'.

Tracking MCC in Simulator

The screenshot displays the Reactis simulator window for a file named 'cruise.mdl (R2009a)'. The interface includes a menu bar (File, Edit, View, Simulate, Test Suite, Validate, Coverage, Window, Help) and a toolbar with various simulation controls. A progress bar at the top right shows '1463' and 'All (1463 steps)'. On the left, a project tree lists components: 'cruise', 'SL CruiseMain', 'SL CruiseMDL', 'DeactivateActivate', 'SL DesiredSpeed', 'SF Mode', 'SpdCheck', 'SL ThrottleCtl', 'SL Plant', 'LowSpeedOn', 'Speed', and 'Configuration Variables'. The main workspace shows a state machine diagram with a central state box labeled 'Decision with MCC targets'. This state has an outgoing transition to a state box labeled 'Off/ entry : mode = 1;' with the guard '[onOff == 0]'. The 'Off' state has an outgoing transition back to 'Decision with MCC targets' with the guard '[onOff == 1]'. The 'Decision with MCC targets' state has a self-loop transition with the guard 'mode == 0;'. It also has an outgoing transition to another state box labeled 'init/ entry : mode = 2;' with the guard '[set == 1 && ... deactivate == 0]'. A 'Toggle Breakpoint' button is visible, with a 'View Coverage Details' button below it. The status bar at the bottom indicates 'Test #29 (29/29), Step 1/1, Total 1463/1463' and 't=0.0'.

Tracking MCC in Simulator

The screenshot displays the Reactis simulator interface. A 'Coverage Details' window is open, showing a table of MCC targets. The main window shows a state machine diagram with a highlighted state 'On' containing a 'Decision with MCC targets' box. A 'Toggle Breakpoint' and 'View Coverage Details' button is visible in the diagram. The status bar at the bottom indicates 'Test #29 (29/29), Step 1/1, Total 1463/1463' and 't=0.0'.

Coverage Details

Decision		Condition	Condition		MC/DC		MC/DC	
True	False		True	False	True	False	True	False
1/30	1/4	set==1.0 deactivate==0.0	1/20	1/4	TT: 1/30	Fx: 1/4	TT: 1/30	TF: 1/20

Decision with MCC targets

On

Toggle Breakpoint
View Coverage Details

Test #29 (29/29), Step 1/1, Total 1463/1463 t=0.0

Tracking MCC in Simulator

The image displays a simulator interface with two 'Coverage Details' windows and a state transition diagram. The diagram features a state labeled 'Decision with MCC targets' and a 'Toggle Breakpoint View Coverage Details' button. A green box highlights a state transition labeled 'OH/ entry : mode = 1;'. The Coverage Details windows show MCC data for various conditions and decisions.

Left Coverage Details Window:

Decision	Decision	Condition	Condition	Condition	MC/DC	MC/DC
True	False		True	False	True	False
1/30	1/4	set==1.0	1/20	1/4	TT: 1/30	Fx: 1/4
		deactivate==0.0	1/30	1/20	TT: 1/30	TF: 1/20

Right Coverage Details Window:

Click on column headers to enable filters Clear Filter

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

State Transition Diagram:

- State: **Decision with MCC targets**
- Transitions:
 - From 'inactive/ entry : mod...' to 'Decision with MCC targets' (labeled 'mode = 1;')
 - From 'Decision with MCC targets' to 'Toggle Breakpoint View Coverage Details' (labeled 'mode = 1;')
 - From 'Toggle Breakpoint View Coverage Details' to 'Decision with MCC targets' (labeled 'mode = 1;')
 - From 'Decision with MCC targets' to 'init/ entry : mode = 2;' (labeled 'mode = 1;')
 - From 'init/ entry : mode = 2;' to 'Decision with MCC targets' (labeled 'mode = 2;')
- Condition: `[set == 1.0 ... deactivate == 0]`

Test #29 (29/29), Step 1/1, Total 1463/1463 t=0.0

Tracking MCC in Simulator

The image displays a simulator interface with two 'Coverage Details' windows and a 'Decision with MCC targets' block. The top window shows a table with columns for Decision, Condition, and MC/DC. The bottom window shows a table with columns for set, deactivate, Decision, and Covered. The decision block contains a state machine diagram with a 'Toggle Breakpoint' button and a 'View Coverage Details' button. A yellow box highlights the 'View Coverage Details' button, and a green box highlights the 'entry : mode = 1;' state.

Row for each MCC target

Decision	Condition	Condition	Condition	MC/DC	MC/DC
True	False	True	False	True	False
1/30	1/4	set==1.0	1/20	1/4	TT: 1/30
		deactivate==0.0	1/30	1/20	Fx: 1/4
					TT: 1/30
					TF: 1/20

Decision with MCC targets

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

Test #29 (29/29), Step 1/1, Total 1463/1463 t=0.0

MCC Filtering

Decision MCC

Click on column headers to enable filters Clear Filter

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

Help Close

MCC Filtering

Click column header

The image shows two side-by-side screenshots of a software window titled "Coverage Details". The window has a tab labeled "Decision" and "MCC". Below the tab, there is a text prompt "Click on column headers to enable filters" and a "Clear Filter" button. The main content is a table with columns: "set==1.0", "deactivate==0.0", "Decision", and "Covered".

In the left screenshot, the table contains three rows of data:

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

In the right screenshot, the "Decision" column header is highlighted with a mouse cursor. The table now only displays the row where "Decision" is "True":

set==1.0	deactivate==0.0	T: Decision	Covered
True	True	True	1/30

A yellow callout box with blue text says "Only display rows with *True* in column". A red arrow points from the text "Click column header" to the "Decision" header in the right screenshot. Another red arrow points from the callout box to the "Decision" header in the right screenshot.

MCC Filtering

Click column header

Click on column headers to enable filters

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

Only display rows with *True* in column

Click column header again

Click on column headers to enable filters

set==1.0	deactivate==0.0	T: Decision	Covered
True	True	True	1/30

Help Close

Click on column headers to enable filters

set==1.0	deactivate==0.0	F: Decision	Covered
False	x	False	1/4
True	False	False	1/20

Only display rows with *False* in column

Click on column headers to enable filters

set==1.0	deactivate==0.0	F: Decision	Covered
False	x	False	1/4
True	False	False	1/20

Help Close

MCC Considerations

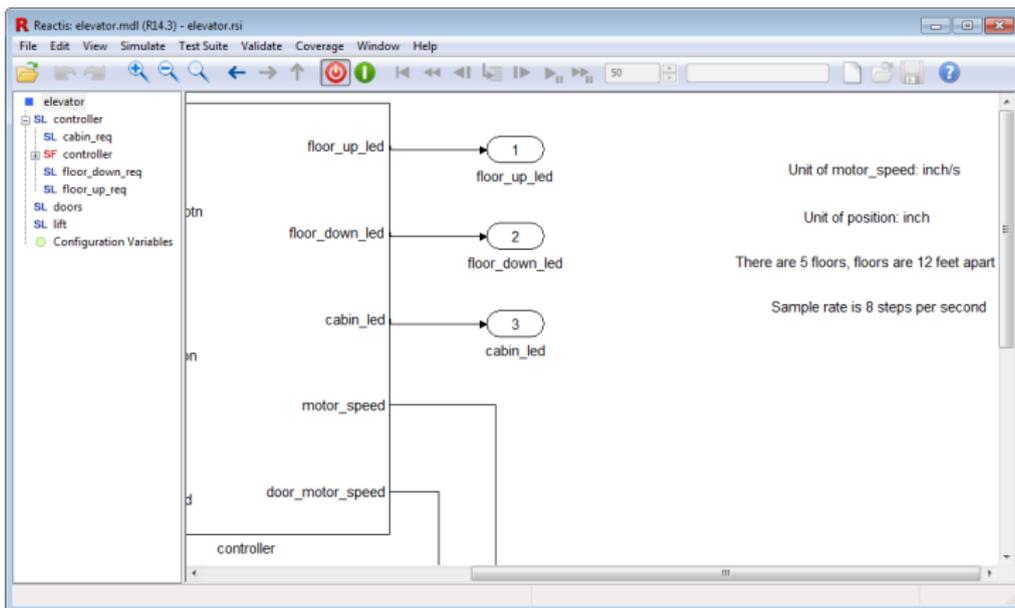
- ▶ Decision with n conditions has 2^n MCC targets.
- ▶ When short-circuiting is enabled, many fewer MCC targets. Number is between MC/DC ($n+1$) and 2^n .
- ▶ Even if 100% MCC is not goal, MCC coverage details can be helpful in obtaining MC/DC.
- ▶ In model-specific settings (*Edit* → *Coverage...*), can set upper-bound on number of conditions in a decision; if exceeded MCC will not be tracked.

Text Search Enhancement

Text search now searches Simulink annotations

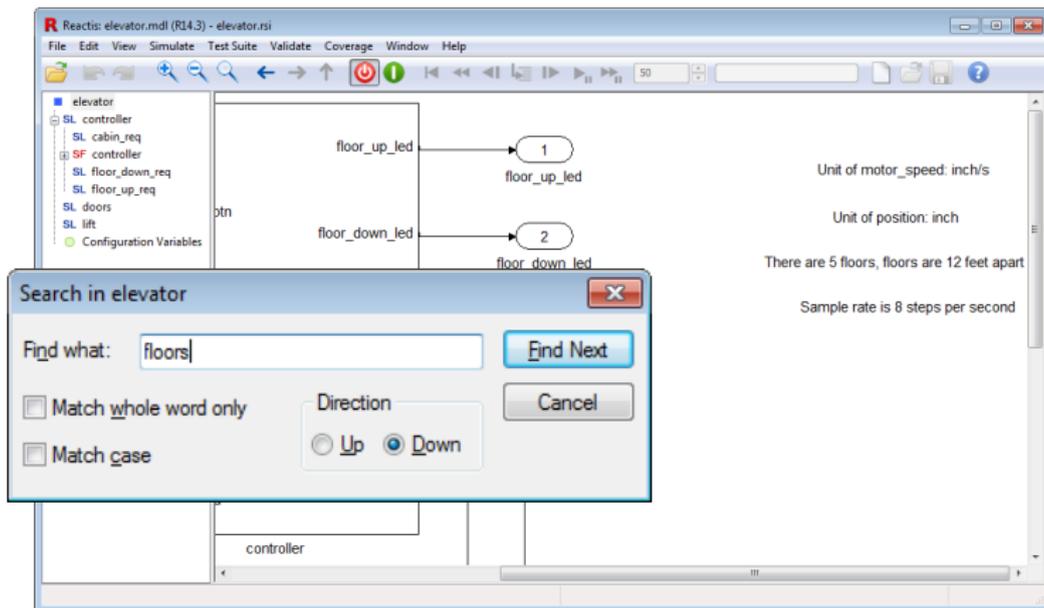
Text Search Enhancement

Text search now searches Simulink annotations



Text Search Enhancement

Text search now searches Simulink annotations



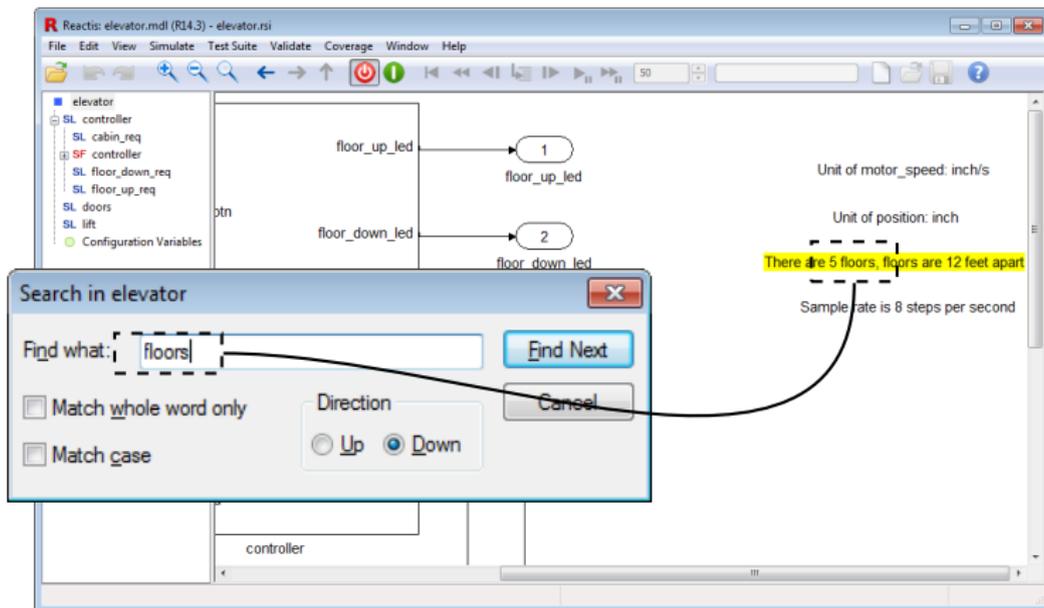
Text Search Enhancement

Text search now searches Simulink annotations

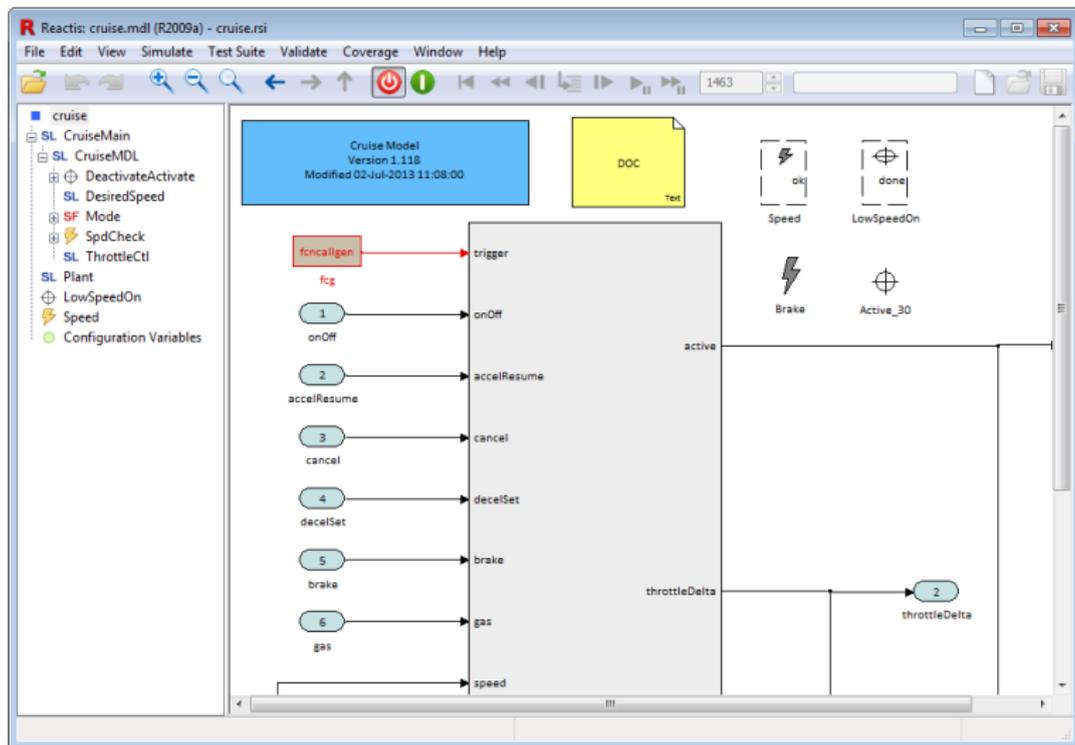
The screenshot displays the Simulink environment for a model named 'elevator.mdl'. A search dialog box titled 'Search in elevator' is open, with the search term 'floors' entered in the 'Find what:' field. The dialog includes options for 'Match whole word only' and 'Match case', and a 'Direction' section with radio buttons for 'Up' and 'Down'. The 'Find Next' button is highlighted. In the background, a Simulink diagram shows a 'controller' block connected to 'floor_up_led' and 'floor_down_led' blocks, which are in turn connected to '1' and '2' blocks. To the right of the diagram, a text annotation reads: 'Unit of motor_speed: inch/s', 'Unit of position: inch', 'There are 5 floors, floors are 12 feet apart', and 'Sample rate is 8 steps per second'. The text 'There are 5 floors, floors are 12 feet apart' is highlighted in yellow, indicating the search result.

Text Search Enhancement

Text search now searches Simulink annotations



Display Simulink Block Colors



Display Model Info and Doc Block Content

Display contents of Model Info and Doc blocks

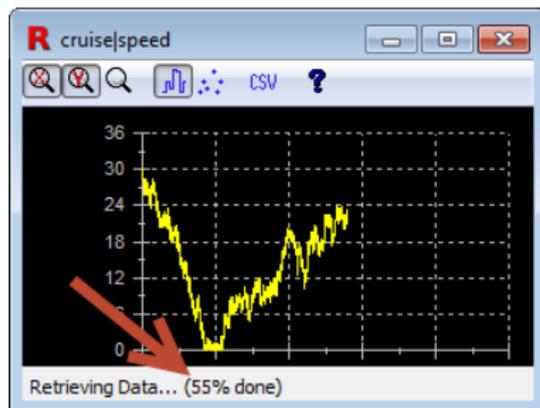
The screenshot displays the Reactis software interface for a model named 'cruise.mdl'. The main window shows a block diagram with a 'Cruise Model' block (Version 1.113, Modified 10-Apr-2013 15:34:52) and a 'DOC' block (Text). A 'DOC block' dialog box is open, displaying the following text:

R DOC block

This model implements a cruise control to demonstrate the different capabilities offered by Reactis to test and validate models. Chapter 3 of the Reactis User's Guide describes how to use Reactis with this model.

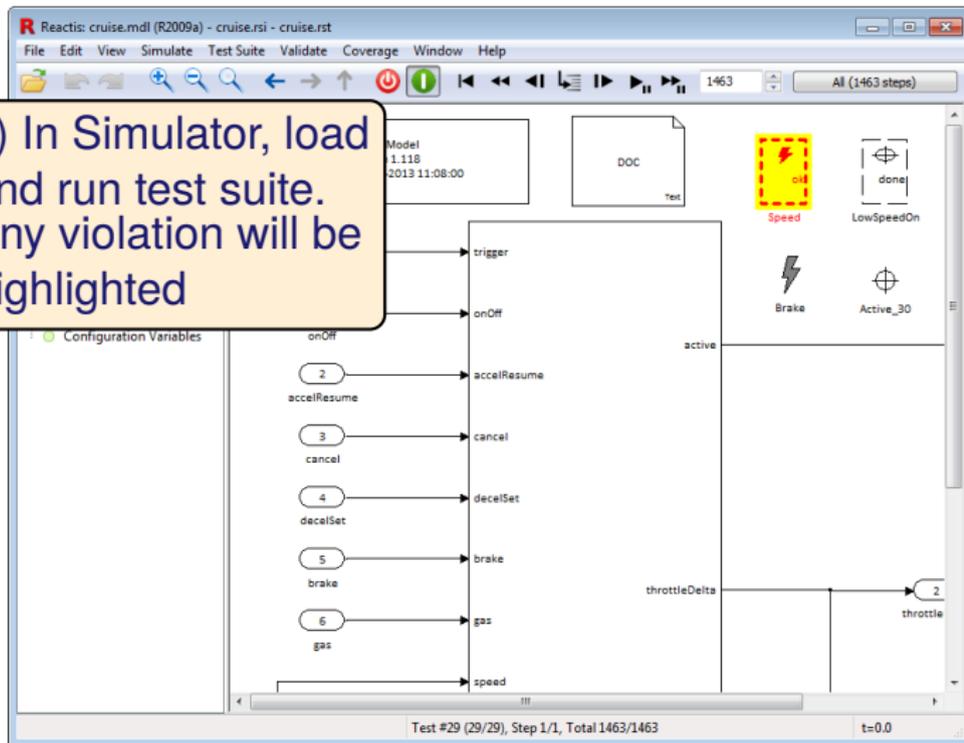
The dialog box includes a 'Close' button. The background interface shows a tree view on the left with 'cruise' expanded to show 'SL CruiseMain', 'SL Plant', 'LowSpeedOn', 'Speed', and 'Configuration Variables'. The main workspace contains a block diagram with 'fncallgen' connected to 'trigger', and a vertical stack of blocks labeled 'fcg', '1 onOff', '2 accelResume', '3 cancel', and '4 deccelSet'.

Show Progress when Opening Scope



Run-to-Violation

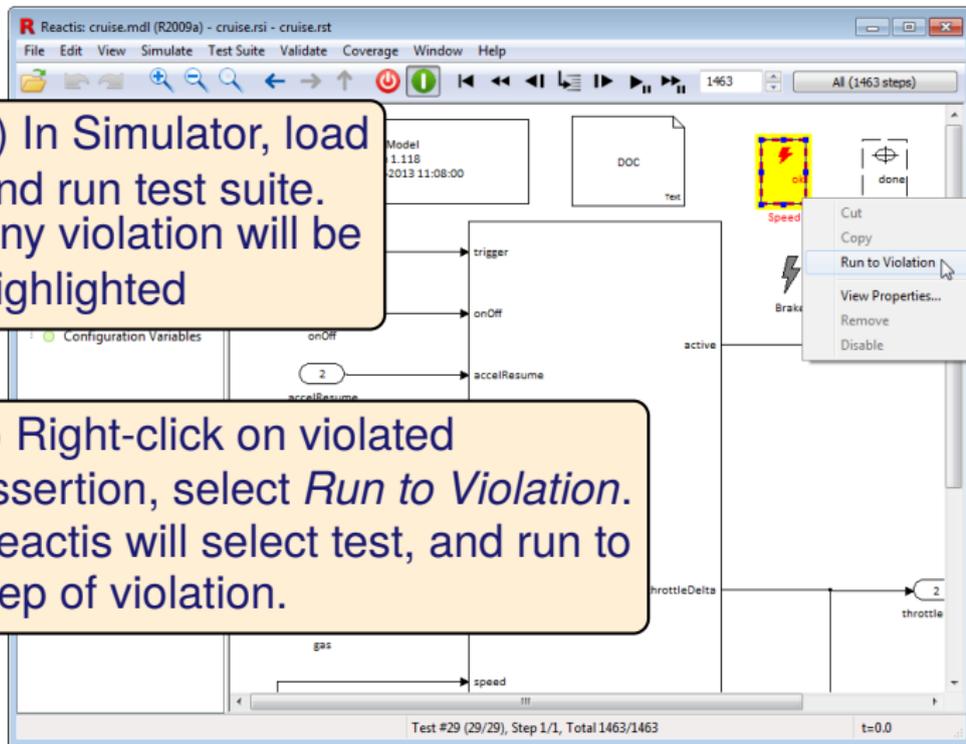
1) In Simulator, load and run test suite. Any violation will be highlighted



Run-to-Violation

1) In Simulator, load and run test suite. Any violation will be highlighted

2) Right-click on violated assertion, select *Run to Violation*. Reactis will select test, and run to step of violation.



Lookup Table Enhancements

- ▶ Enhancements to N-d lookup tables, Pre Lookup, Interpolation
 - ▶ Track coverage
 - ▶ Support more type combinations
- ▶ In coverage reporting, show breakpoint values

	<code>[-inf, c1]</code>	<code>[c1, c2]</code>	<code>[c2, c3]</code>	<code>[c3, +inf]</code>
<code>[-inf, r1]</code>	1/4	2/3	5/1	1/6
<code>[r1, r2]</code>	1/5	2/2	3/11	2/1
<code>[r2, +inf]</code>	1/1	1/3	1/2	1/7

Old

	<code>u2 < 1.0</code>	<code>u2 >= 1.0</code>	<code>u2 >= 10.0</code>	<code>u2 >= 100.0</code>
<code>u1 < 1.0</code>	1/4	2/3	5/1	1/6
<code>u1 >= 1.0</code>	1/5	2/2	3/11	2/1
<code>u1 >= 10.0</code>	1/1	1/3	1/2	1/7

New

- ▶ Tester gets better coverage for lookup tables

API Functions to Export Coverage Reports

- ▶ *rsTesterWithReport* produces a coverage report after a Tester run
- ▶ *rsSimRunSuiteWithReport* produces a coverage report after running a test suite in Simulator