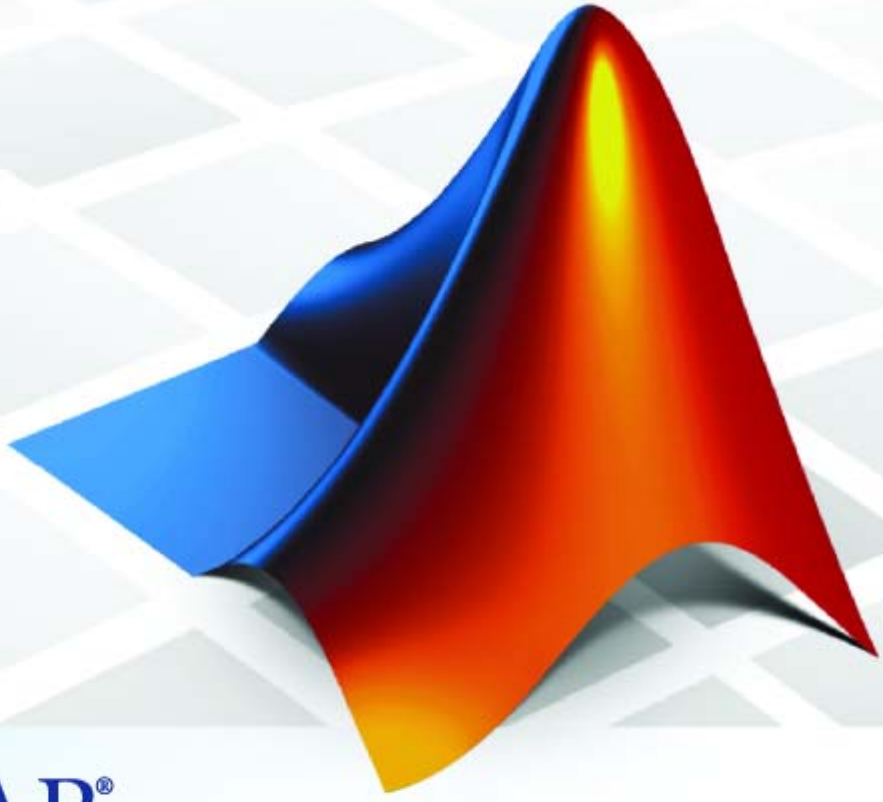
Report Generator 3

User's Guide



MATLAB®
&SIMULINK®

The MathWorks
Accelerating the pace of engineering and science

How to Contact The MathWorks

`www.mathworks.com`	Web
`comp.soft-sys.matlab`	Newsgroup
`www.mathworks.com/contact_TS.html`	Technical Support

`suggest@mathworks.com`	Product enhancement suggestions
`bugs@mathworks.com`	Bug reports
`doc@mathworks.com`	Documentation error reports
`service@mathworks.com`	Order status, license renewals, passcodes
`info@mathworks.com`	Sales, pricing, and general information

508-647-7000 (Phone)

508-647-7001 (Fax)

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

For contact information about worldwide offices, see the MathWorks Web site.

Report Generator User's Guide

Revision History

January 1999	First printing	New (Release 11)
December 2000	Second printing	Revised (Release 12)
June 2004	Third printing	Revised for Version 2.02 (Release 14)
August 2004	Online only	Revised for Version 2.1
October 2004	Online only	Revised for Version 2.1.1 (Release 14SP1)
December 2004	Online only	Revised for Version 2.2 (Release 14SP1+)
April 2005	Online only	Revised for Version 2.2.1 (Release 14SP2+)
September 2005	Online only	Revised for Version 2.3.1 (Release 14SP3)
March 2006	Online only	Revised for Version 3.0 (Release 2006a)
September 2006	Online only	Revised for Version 3.1 (Release 2006b)
March 2007	Fourth printing	Revised for Version 3.2 (Release 2007a)

Contents

1 What Is Report Generator?

2 Creating Reports in MATLAB

3 Creating Reports in Simulink

4 Creating a Web View of a Simulink Model

5 Generating Reports

6 Exporting Simulink Models to Web Views

7 Working with Components

8 Building Custom Components

9 Creating and Using Customized Styles

10 Components — By Category

Index

1

What Is Report Generator?

This chapter covers the following topics, which apply to both MATLAB ® and Simulink® users.

To find information specific to MATLAB Report Generator and Simulink Report Generator, see “How to Use This Documentation” on page 1-9.

Report Generator Overview

Report Generator creates documentation (called *reports*) about your MATLAB, Simulink, and Stateflow® data and workflow. You can configure your report templates to include:

- MATLAB code and its output
- Workspace data
- Graphics created in MATLAB, Simulink models, Stateflow diagrams
- Logical components like `IF`, `THEN`, `ELSE`, and `WHILE` that allow for conditional report generation
- Loops that perform a series of operations on multiple objects
- User-defined text, styles, and components

Using Report Generator features, you can do the following:

- Document tasks you perform in MATLAB such as analyzing and visualizing data and developing algorithms.
- Create a report template that can be reused and that reflects your own styles and standards.
- Keep your documentation and specifications up to date with your workflow.
- Create user manuals that accurately capture your application capabilities and specifications.
- Document code generated by Real-Time Workshop® in Adobe Acrobat PDF and Microsoft Word formats.

Report Generator also lets you create interactive Web views of Simulink models and Stateflow diagrams. These Web views are viewable in a browser, so you can share them with colleagues who do not have MathWorks products installed. The Web view's model hierarchy allows you to navigate to a specific subsystem and view the properties of blocks, signals, and scopes.

Report Generator for MATLAB and Simulink

Report Generator for MATLAB provides report components and stylesheets for documenting the MATLAB environment and workflow. Report Generator for Simulink is installed on top of Report Generator for MATLAB, and it lets you:

- Add report components for snapshots of Simulink models (and their properties) and Stateflow diagrams.
- Create Web views of Simulink models and Stateflow diagrams.
- Create reports of code generated by Real-Time Workshop.
- Simulate Simulink models as reports are generated, and document results of the simulation in the reports.
- Run checks on your Simulink model using Model Advisor, and document the results of the checks in PDF and Word formats.

Report Generator Workflow

Most people execute a workflow similar to the one described in this section when using Report Generator to create reports. Report Generator interacts with MATLAB and Simulink to give you access to Report Generator capabilities.

1. To get started with a report, open the Report Explorer, the graphical user interface (GUI) that helps you design and lay out your report. To open the Report Explorer, enter `report` at the MATLAB command line, or if you are using Simulink, select **Tools > Report Generator** from the Simulink model window.

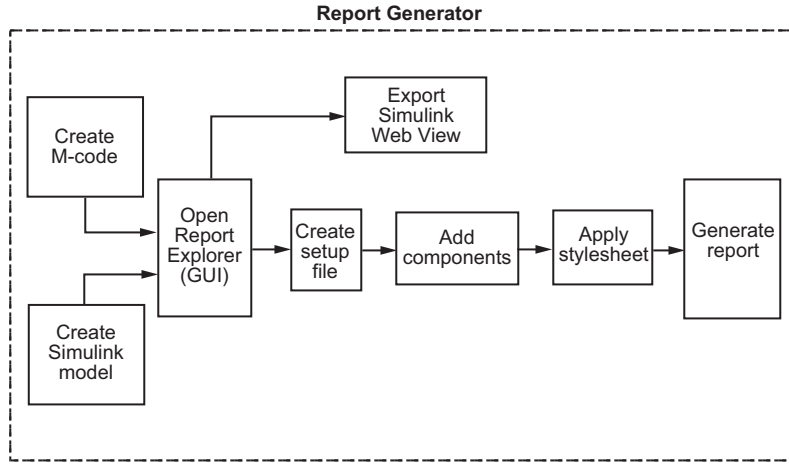
2. Create a new report template. For details about report templates, see "Setting Report Options in the Report Template" on page 3-3.

3. Add existing components to the report template or create your own custom components. For details on using components, see Chapter 7, "Working with Components".

4. Choose an existing stylesheet or create an new stylesheet to apply desired styles and standards to the report. For details on stylesheets and attributes, see Chapter 9, "Creating and Using Customized Styles".

5. Generate the report.

The following figure illustrates a typical workflow for Report Generator.



How MATLAB and Simulink Interact with Report Generator

MATLAB and Simulink interact with Report Generator to create reports and Web views. You can access the Report Explorer GUI from the MATLAB command line or from the Simulink model window. The following table describes these interactions in detail.

User Interface	Interaction with Report Generator	Purpose
Report Explorer	The Report Explorer is the graphical user interface (GUI) for Report Generator that allows you to: • Create and modify report templates. • Apply stylesheets to the report format. • Choose output file format. • Generate reports. • Export Web views.	Use the Report Explorer to edit existing report templates, components, stylesheets, and attributes, or to customize your own.

User Interface	Interaction with Report Generator	Purpose
MATLAB command line	From the MATLAB command line, you can: • Generate reports. • Start the Report Explorer. • Export a Simulink model to a Web view without using the Report Explorer. • Create and modify report template files. • Apply stylesheets. • Specify output formats for reports.	The following MATLAB commands work with Report Generator: • `report` — Start the Report Explorer. • `setedit` — Edit a report template with the Report Explorer. • `rptconvert` — Convert a Report Generator source file to the desired output format. • `rptlist` — List `.rpt` files in the current path. • `slwebview` — Export a Simulink model to a Web view.
Simulink model window	From a Simulink model window, you can: • Export the model to a Web view without using the Report Explorer. • Start the Report Explorer to export the model to a Web view or generate a report.	With a Simulink model open, you can export the model to a Web view directly by selecting **File > Export to Web** or by selecting **Tools > Report Generator** and clicking **Export Simulink model to Web viewer**. Both methods open a small dialog box with a few options in it. For details about exporting Simulink models to a Web view, see Chapter 4, “Creating a Web View of a Simulink Model”. You can also create a report that incorporates the current Simulink model.

Report Generator also interacts with Real-Time Workshop and Stateflow. You can use Report Generator to create PDF and Word documents for code generated by Real-Time Workshop. You can use Report Generator with Stateflow to take snapshots of diagrams, describe truth tables, document diagram hierarchy, and document Stateflow object properties.

Multiple Report Formats

When Report Generator first creates a report, it generates a DocBook XML source file. You can customize this XML as needed; for more information on how to customize DocBook XML, see the OASIS DocBook TC Web page at `http://www.oasis-open.org`.

Next, Report Generator converts the XML source to a user-specified report format that can be one of the following:

- Rich Text Format (RTF)
- Hypertext Markup Language (HTML)
- Microsoft Word (`.doc`)
- Adobe Acrobat PDF

Note Only bitmap (`.bmp`) and jpeg (`.jpg`) images are supported in PDF reports.

Browser Requirements for Web Views

To use Web views, you need a Web browser that supports Scalable Vector Graphics (SVG). Web views use SVG to render and navigate models.

The MathWorks recommends using these Web browsers:

- Firefox version 1.5 or later, which has native support for SVG. To obtain the Firefox browser, visit `http://www.mozilla.com/`.

  Note At this time, Web views are not supported in Firefox on Mac OS.

- Internet Explorer with the Adobe SVG Viewer plug-in. To obtain the Adobe SVG Viewer plug-in, visit `http://www.adobe.com/svg/`.

How to Use This Documentation

This table shows recommended reading paths for MATLAB users and Simulink users.

If you use ...	Read the following chapters ...
MATLAB	Chapter 2, “Creating Reports in MATLAB” Chapter 5, “Generating Reports” Chapter 7, “Working with Components” Chapter 8, “Building Custom Components” Chapter 9, “Creating and Using Customized Styles”
Simulink	Chapter 3, “Creating Reports in Simulink” Chapter 4, “Creating a Web View of a Simulink Model” Chapter 5, “Generating Reports” Chapter 6, “Exporting Simulink Models to Web Views” Chapter 7, “Working with Components” Chapter 8, “Building Custom Components” Chapter 9, “Creating and Using Customized Styles”

2

Creating Reports in MATLAB

This tutorial covers the following tasks, which apply to MATLAB users.

About This Tutorial

Report Generator features demonstrated in this tutorial include the following:

- Evaluating MATLAB expressions
- Formatting reports with multiple chapters
- Running loops and flow control
- Handling errors
- Inserting workspace data
- Inserting images created by MATLAB

This tutorial takes you through the steps required to design a report template file and generate a report from it in MATLAB, using the Report Explorer GUI.

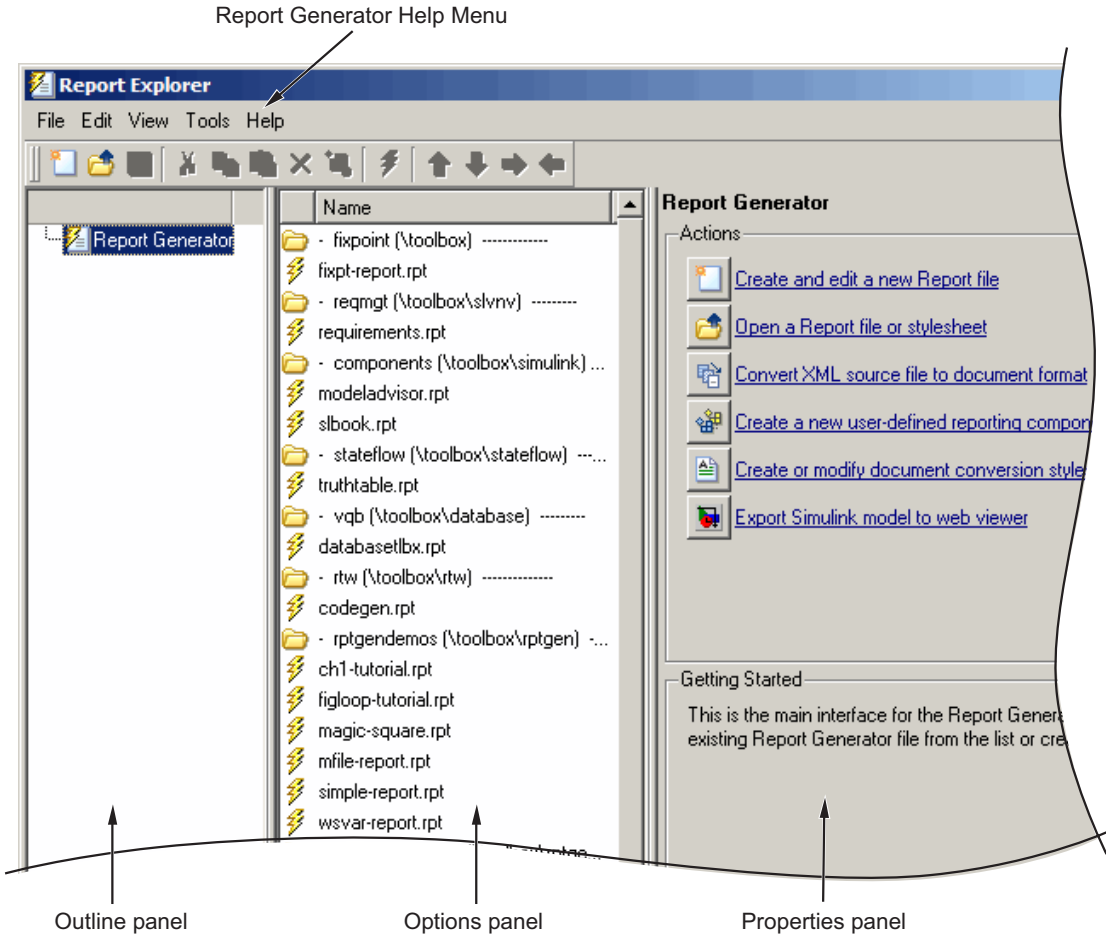
The report template that you design creates several magic squares using M-code and displays them as matrices or images, depending on their size. (A magic square is a matrix in which the columns, rows, and diagonal all add up to the same number.) For details about magic squares, see “Matrices and Magic Squares”.

Note You do not need to know MATLAB to execute this tutorial, but it might be helpful for understanding the M-code that executes during report generation.

Setting Report Options in the Report Template

Follow these steps to create and configure your report template:

1. Start MATLAB.
2. Type `report` in the MATLAB command window. The Report Explorer window opens.



Tip If the Report Explorer window opens with only two panes, one of the panes is hidden. The vertical boundaries can be moved to reveal any hidden pane.

The Report Explorer has three panes:

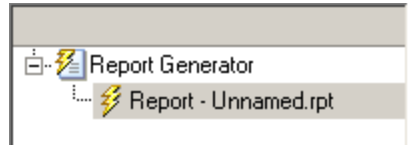
- The Outline pane on the left shows the hierarchy of components in all currently opened reports. Report components can reside within other report components, creating parent, child, and sibling relationships.
- The Options pane in the middle lists the options available in the context of the Outline pane. Because no report is open, the Options pane lists the available reports. When a report is open, the Options pane lists the available components. When a stylesheet is open, the Options pane lists the available attributes.
- If no report is open, the Properties pane on the right displays the tasks the Report Explorer can perform. If a report is open, the Properties pane displays the properties for the element currently selected in the Options pane.

These three panes can be made wider or narrower by dragging the vertical boundaries between the panes.

3 Select **File > New** to create a new report. Your new report will be created with the default name `Unnamed.rpt`.

The three Report Explorer panes have the following contents:

The Outline pane initially contains the top level of the report.



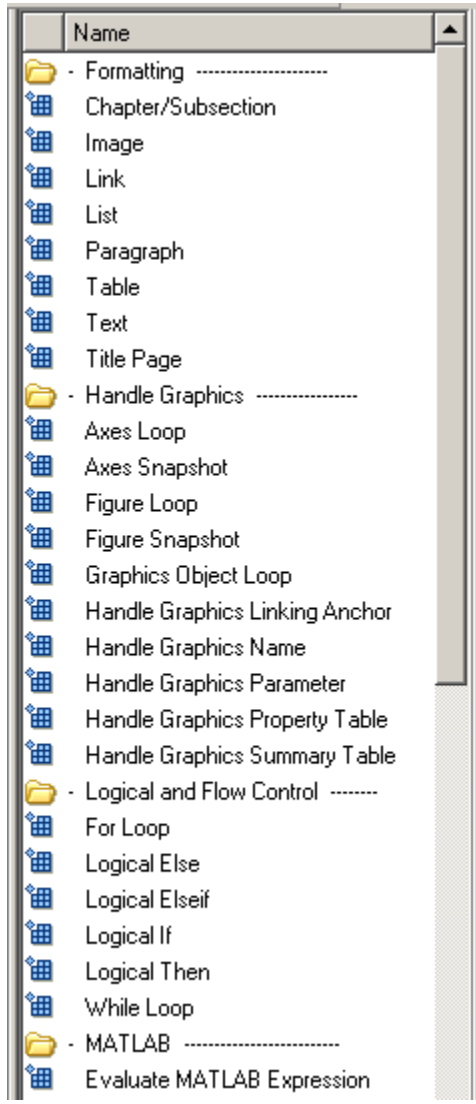
Outline Pane

The Options pane lists the component categories and components available for adding to reports. In this figure, the visible component categories are

`Formatting`, `Handle Graphics`, `Logical and Flow Control`, and `MATLAB`. For details about report components, see Chapter 10, “Components — By Category”.

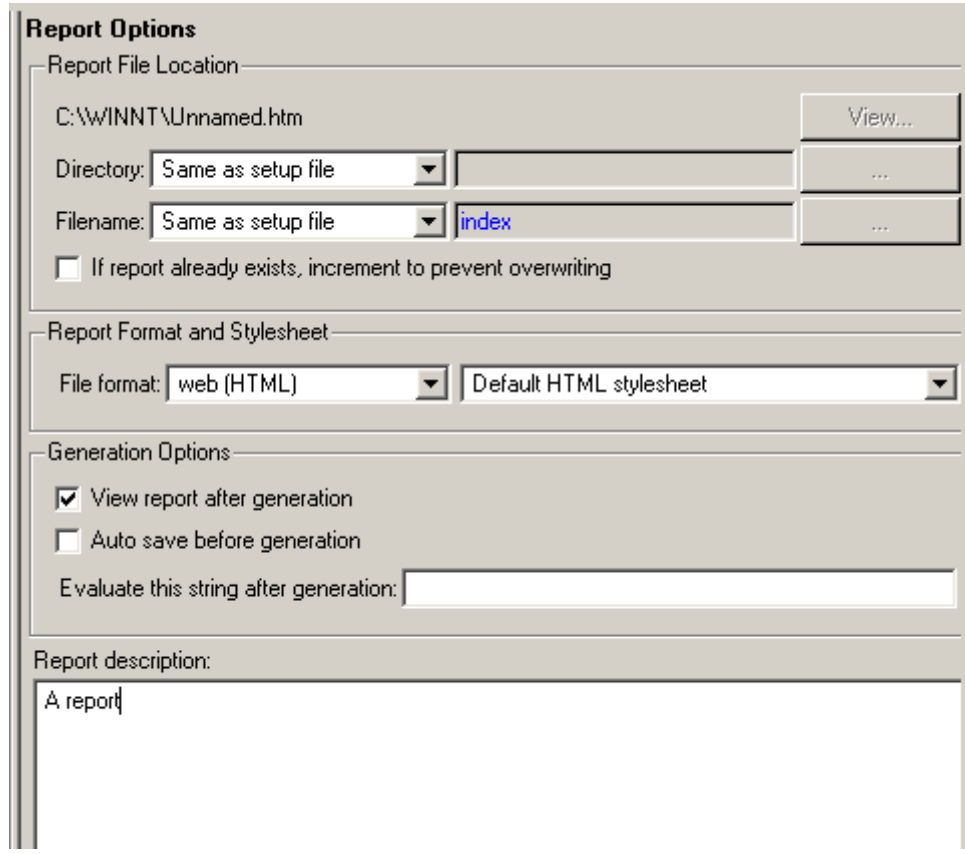
The folder icon indicates a component category.

The blue square icon indicates a component.



Options Pane

After you create a new report, the Properties pane initially displays the properties for the report as a whole.



Properties Pane

4 In the Properties pane on the right, do the following:

- To save the report in the current working directory, select `Present Working Directory` from the **Directory** drop-down list.
- Set **File format** to `web (HTML)` to create the report as an HTML file.
- In the **Report description** text box, replace the existing text with the following text.

Tip You can copy and paste this text from the HTML documentation into the Report Explorer.

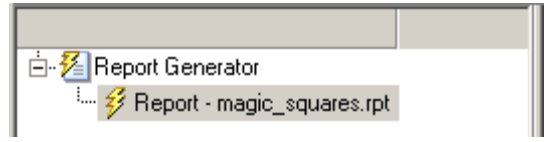
```
This report creates a series of magic squares
and displays them as images.

A magic square is a matrix in which the
columns, rows, and diagonal all add up to the
same number.
```

Note When you change a Properties pane field, the background changes to a different color (the default is a cream color), meaning that there are unapplied changes to that field. As soon as you perform any action with another component, the changes are applied and the background color becomes white again.

5 Select **File > Save As** and name your report's report template `magic_squares.rpt`.

The Outline pane on the left reflects the new file name.

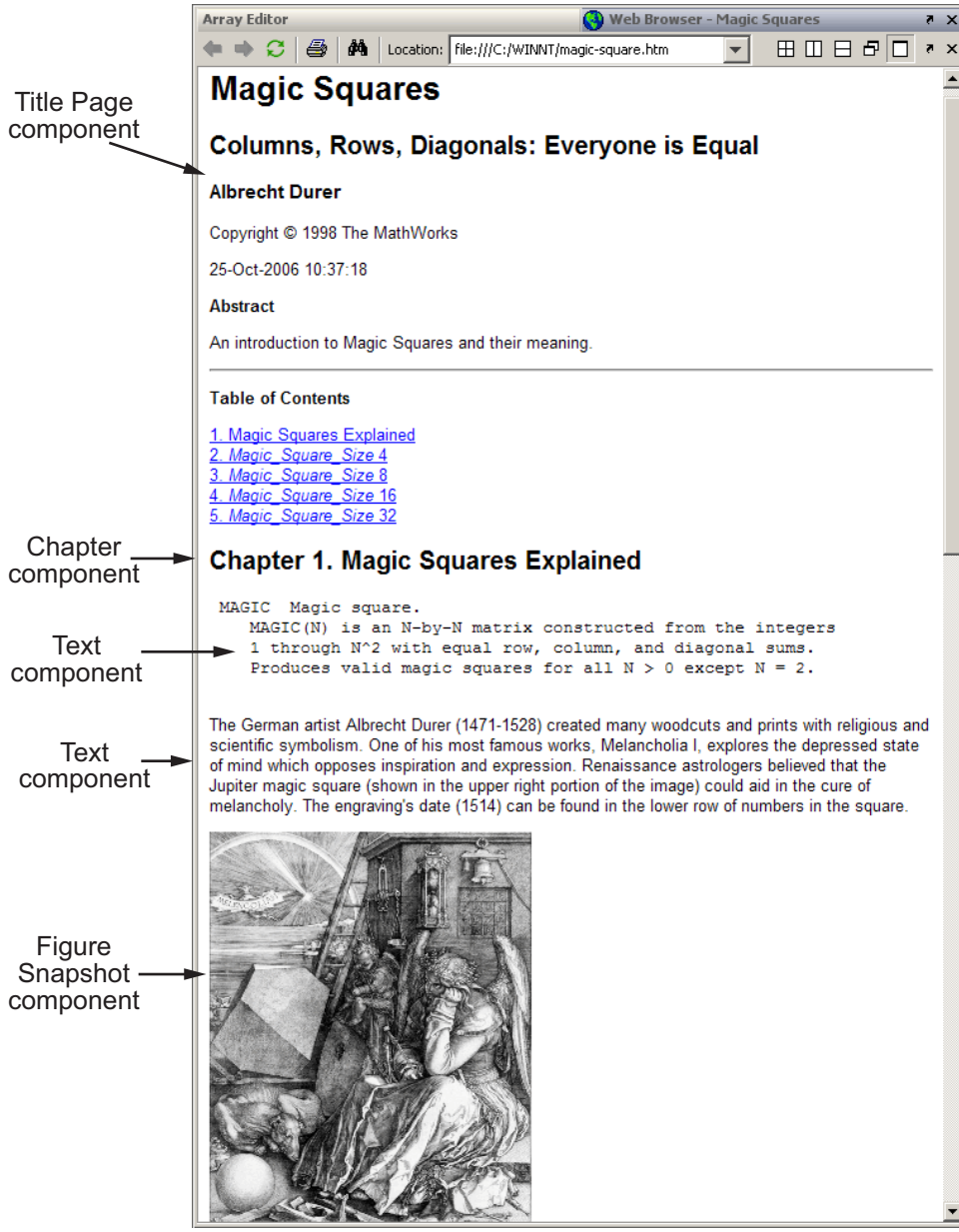


Adding Components to the Report Template

The following topics are covered in this section:

Report components specify what information you want included in the report. The following figure shows a sample page from the report you create in this tutorial and what components are used to produce this output.

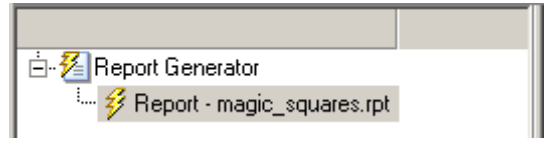
Note Report components added to the report template must not be deactivated in order for the report to be correctly generated.


Array Editor
Web Browser - Magic Squares
Location:
file:///C:/WINNT/magic-square.htm
Title Page component
Chapter component
Text component
Text component
Figure Snapshot component
Magic Squares
Columns, Rows, Diagonals: Everyone is Equal
Albrecht Durer
Copyright © 1998 The MathWorks
25-Oct-2006 10:37:18
Abstract
An introduction to Magic Squares and their meaning.
Table of Contents
1. Magic Squares Explained
2. Magic_Square_Size 4
3. Magic_Square_Size 8
4. Magic_Square_Size 16
5. Magic_Square_Size 32
Chapter 1. Magic Squares Explained
MAGIC Magic square.
MAGIC(N) is an N-by-N matrix constructed from the integers
1 through N^2 with equal row, column, and diagonal sums.
Produces valid magic squares for all N > 0 except N = 2.
The German artist Albrecht Durer (1471-1528) created many woodcuts and prints with religious and scientific symbolism. One of his most famous works, Melancholia I, explores the depressed state of mind which opposes inspiration and expression. Renaissance astrologers believed that the Jupiter magic square (shown in the upper right portion of the image) could aid in the cure of melancholy. The engraving's date (1514) can be found in the lower row of numbers in the square.

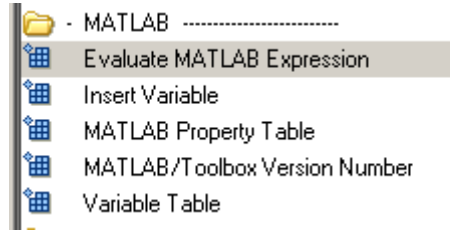
Setting Initial Values

To specify the sizes of the magic squares to be included in the report, use the Evaluate MATLAB Expression component.

1. In the Outline pane on the left, select the top level of the report template.



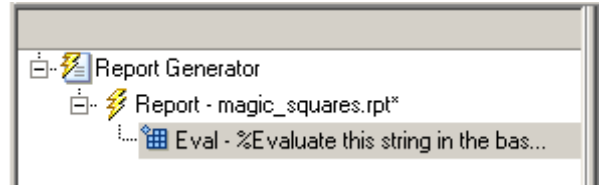
2. In the Options pane in the middle, under the `MATLAB` category, select `Evaluate MATLAB Expression`.



3. In the Properties pane on the right, click the icon next to **Add component to current report** to insert that component into the report.

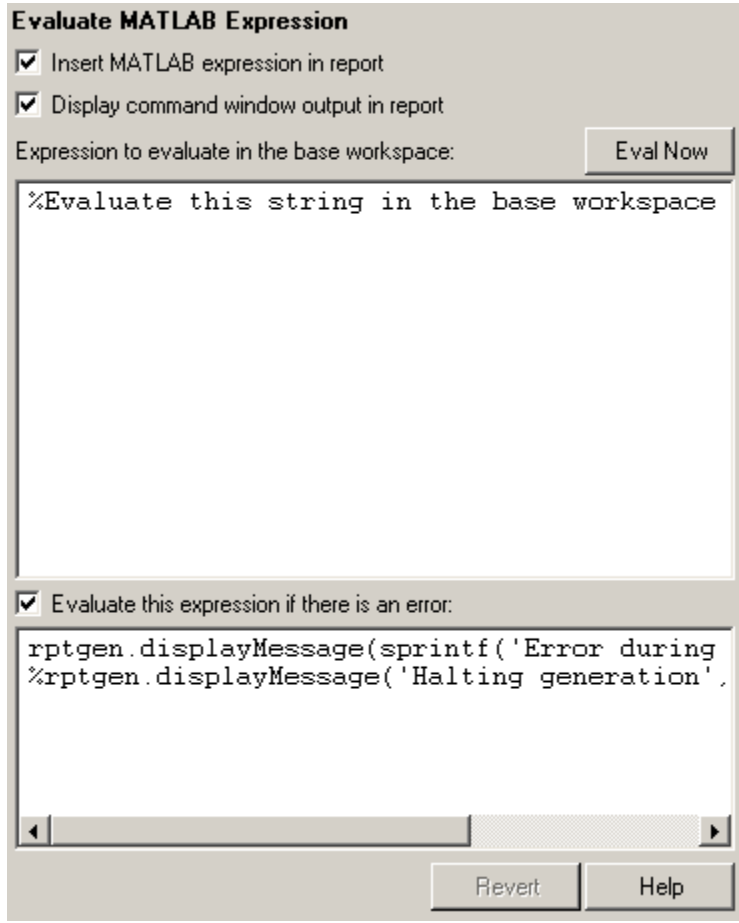
   You cannot edit the component information in the Properties pane until you have added the component to the report.

   In the Outline pane on the left, the `Eval` component is listed under the `magic_squares` report.



The icon next to the `Eval` component means that this component cannot have any child components. By default, any components you add with the `Eval` component selected are added as siblings to this component.

The options for the `Evaluate MATLAB Expression` component appear in the Properties pane on the right.



4 Clear the **Insert MATLAB expression in report** and **Display command window output in report** check boxes.

You do not want to include the MATLAB code details or its output in this report.

5 In the **Expression to evaluate in the base workspace** text box, replace the existing text with the following M-code.

Tip You can copy and paste this text from the HTML documentation into the Report Explorer.

```
%This M-code sets up two variables
%that define how the report runs.
%magicSizeVector is a list of MxM
%Magic Square sizes to insert into
%the report. Note that magic
%squares cannot be 2x2.

magicSizeVector=[4 8 16 32];

%largestDisplayedArray sets the
%limit of array size that will be
%inserted into the report with the
%Insert Variable component.

largestDisplayedArray=15;
```

6 In the **Evaluate this expression if there is an error** text box, replace the existing text with the following text:

```
disp(['Error during eval: ', lasterr])
```

This causes an error to display if the M-code fails.

Tip You can click the **Eval Now** button at the top right corner of the Report Explorer window to execute the commands that you specified in your MATLAB expression immediately. This is an easy way to check if your commands are correct and will not result in problems at run time.

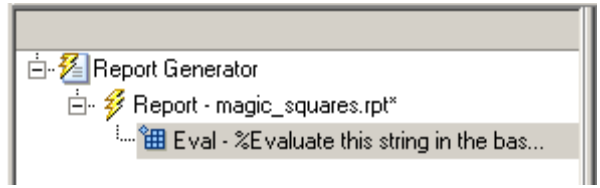
---

7 Save the report by clicking **File > Save**.

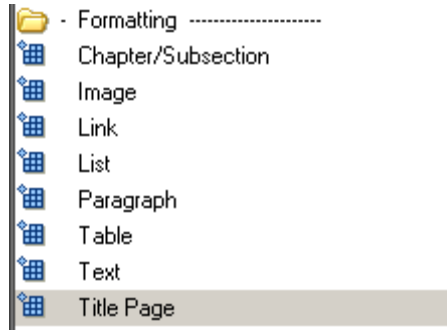
Creating a Title Page

To create a title page for the report, use the Title Page component:

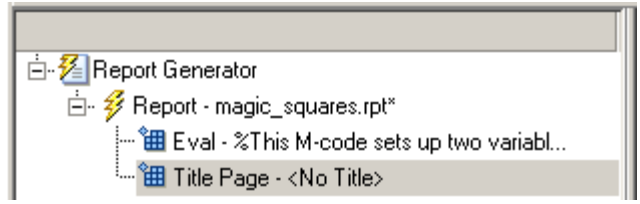
1 In the Outline pane on the left, select the `Eval` component.



2 In the Options pane in the middle, under the `Formatting` category, double-click `Title Page` to add the component to the report in one step.

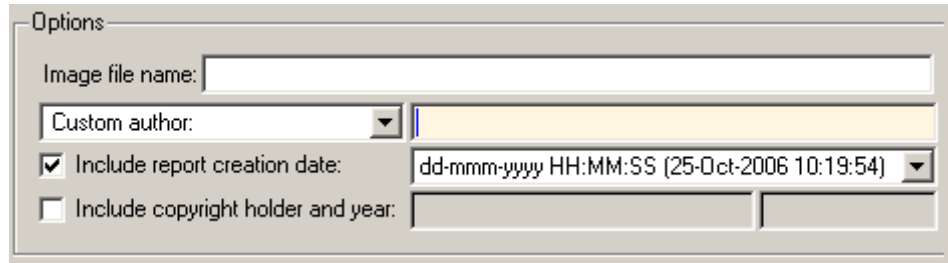


Since the icon next to the `Eval` component indicates that it cannot have children, the `Title Page` component is added as a sibling of the `Eval` component. Likewise, the `Title Page` component cannot have children either.



3 In the Properties pane on the right, do the following:

- In the **Title** text box, enter `Magic Squares`.
- In the **Subtitle** text box, enter `Columns, Rows, Diagonals:  Everyone is Equal`.
- Under **Options**, you do not need to enter an image file name, but make sure the drop-down list is set to `Custom author` as in this figure.



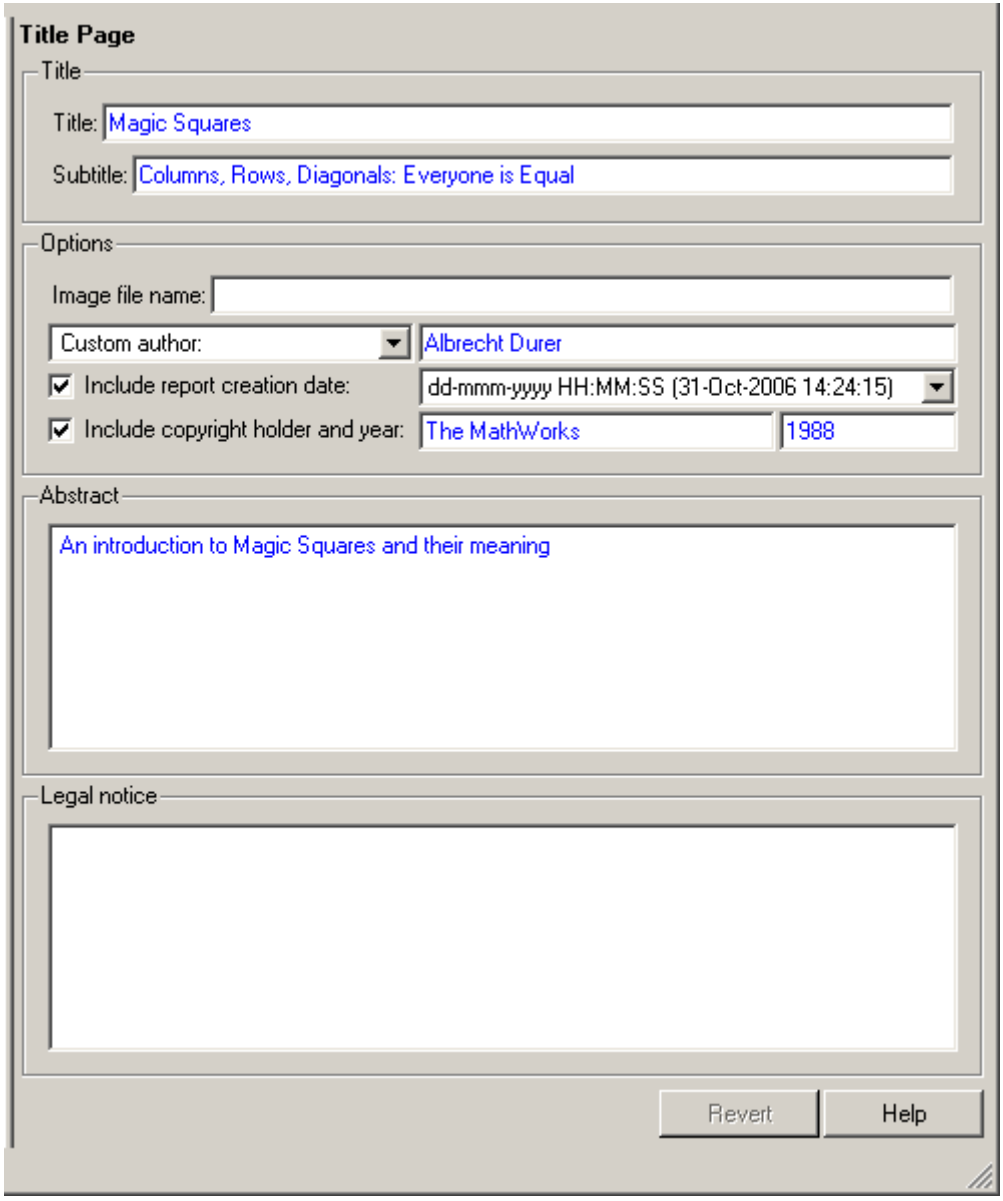
- Next to `Custom author`, enter `Albrecht Durer`.

  Albrecht Dürer created the etching that contains a magic square. Your final report includes an image of that etching.

- Select the **Include copyright holder and year** check box.
- In the next text box, enter `The MathWorks`.
- In the second text box, enter `1998`.
- In the **Abstract** text box, enter

  ```
  An introduction to Magic Squares and their meaning.
  ```

  The Properties pane on the right should look as follows.

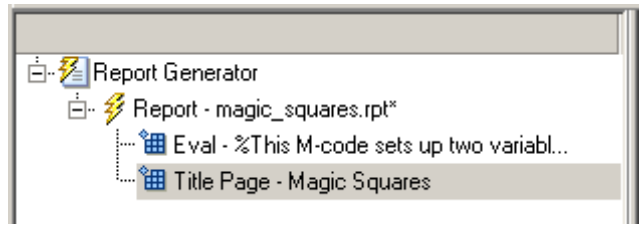


4 Save the report.

Adding the First Chapter

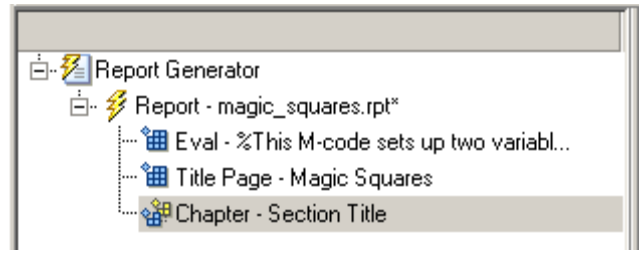
Add a chapter to the report by using the Chapter/Subsection component.

1. In the Outline pane on the left, select the `Title Page` component.



2. In the Options pane in the middle, under the `Formatting` category, double-click `Chapter/Subsection`.

   The Outline pane should look as follows.

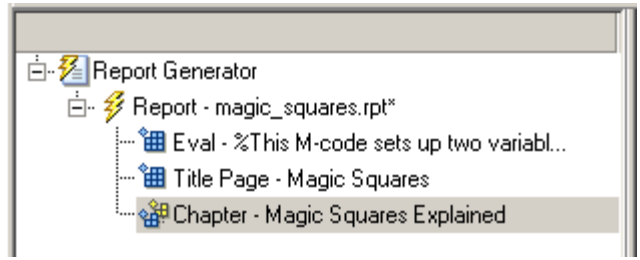


   Note that the `Eval`, `Title Page`, and `Chapter` components are all child components of the report's top level, but siblings to each other.

   The icon next to the `Chapter` component indicates that it can have child components. The next section explains how to add child components to this `Chapter` component.

3. In the Properties pane on the right, enter `Magic Squares Explained` for the custom chapter title.

   The Outline pane on the left changes to reflect the chapter title.



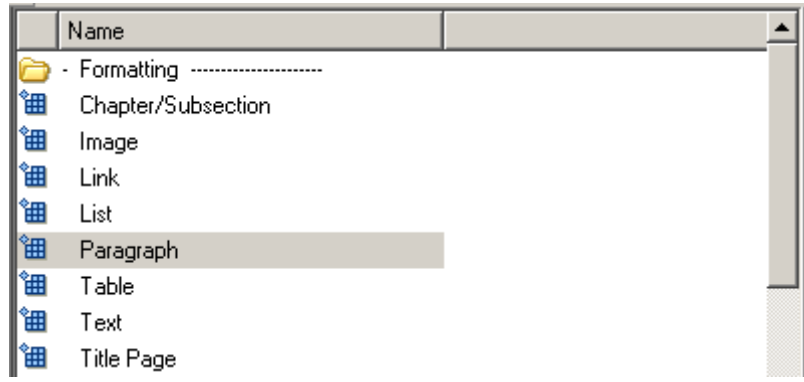
4 Save the report.

Adding Components to the First Chapter

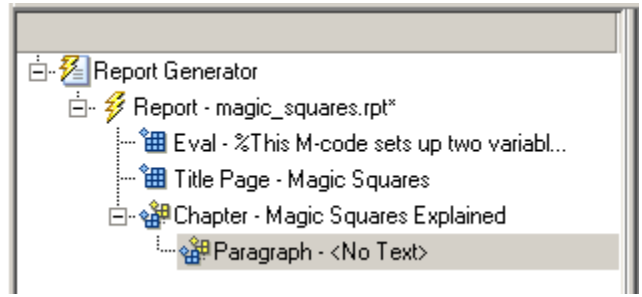
Add some introductory text to the first chapter using the Paragraph and Text components.

1 In the Outline pane on the left, select the `Chapter` component.

2 In the Options pane in the middle, under the `Formatting` category, double-click `Paragraph`.



In the Outline pane on the left, the new component is added as a child to the `Chapter` component.

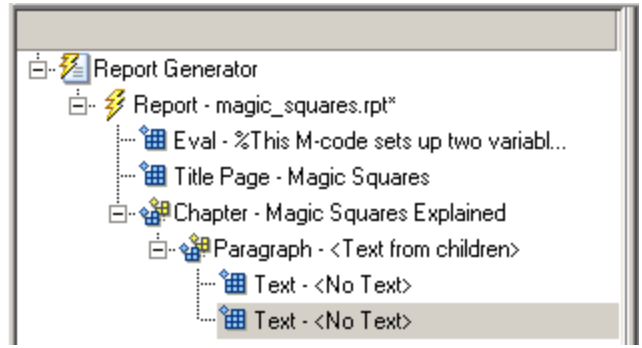


By default, the Paragraph component inherits its text from its child components, so add two `Text` components.

3 In the Options pane in the middle, under the `Formatting` category, double-click `Text`.

4 Double-click `Text` again to add a second component.

The Outline pane should look as follows.



5 In the Outline pane on the left, select the first `Text` component.

6 In the **Text to include in report** text box, enter `%<help('magic')>`.

The `%` sign and angle brackets `<>` indicate to Report Generator that this is MATLAB code to be evaluated. The command `help('magic')` displays information about the MATLAB function `magic`.

7 In the Outline pane on the left, select the second `Text` component.

8 In the **Text to include in report** text box, enter the following text.

Tip You can copy and paste this text from the HTML documentation into the Report Explorer.

```
The German artist Albrecht Durer (1471-1528)
created many woodcuts and prints with religious
and scientific symbolism.  One of his most famous
works, Melancholia I, explores the depressed state
of mind that opposes inspiration and expression.
Renaissance astrologers believed that the Jupiter magic
square (shown in the upper right portion of the image)
could aid in the cure of melancholy. The engraving's
date (1514) can be found in the lower row of numbers
in the square.
```

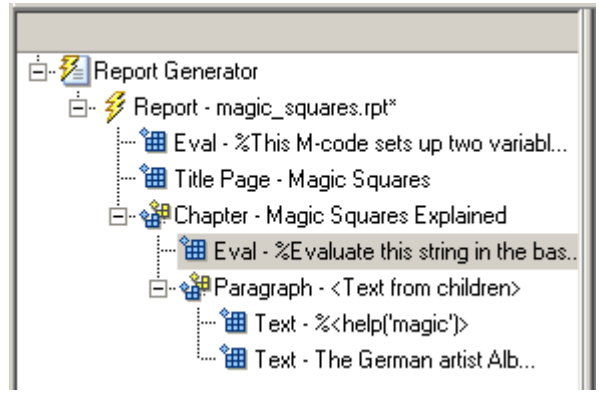
9 Save the report.

The next step is to include an image of the etching shown at the beginning of this section.

10 In the Outline pane on the left, select the `Chapter` component.

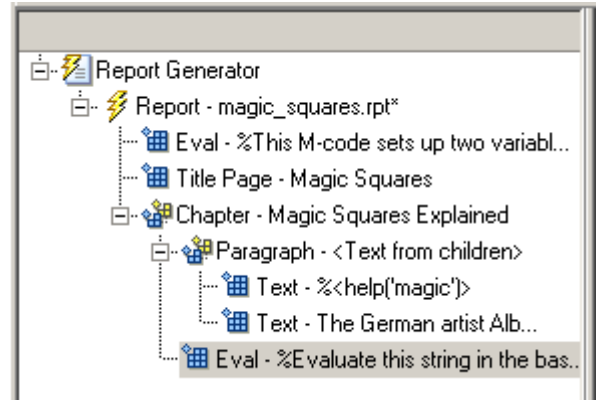
11 In the Options pane in the middle, under the `MATLAB` category, double-click `Evaluate MATLAB Expression`.

The new component becomes a child under the `Chapter` component.



You need to move the `Eval` component below the `Paragraph` component so that the image follows the introductory text.

12 Click the **down** arrow on the toolbar to move the `Eval` component below the `Paragraph` component.



13 With the `Eval` component still selected, do the following in the Properties pane on the right:

a Clear the **Insert MATLAB expression in report** and **Display command window output in report** check boxes.

You do not want to include the code or its output in the report.

b In the **Expression to evaluate in the base workspace** text box, replace the existing text with the following M-code.

Tip You can copy and paste this text from the HTML documentation into the Report Explorer.

```
%This loads a self-portrait of Albrecht
%Durer, a German artist. There is a
%magic square in the upper right corner
%of the image.

durerData=load('durer.mat','-mat');
figure('Units','Pixels',...
'Position',[200 200 size(durerData.X,2)*.5 size(durerData.X,1)*.5 ]);

image(durerData.X);
colormap(durerData.map);
axis('image');
set(gca,...
    'Xtick',[],...
    'Ytick',[],...
    'Units','normal',...
    'Position',[0 0 1 1]);

clear durerData
```

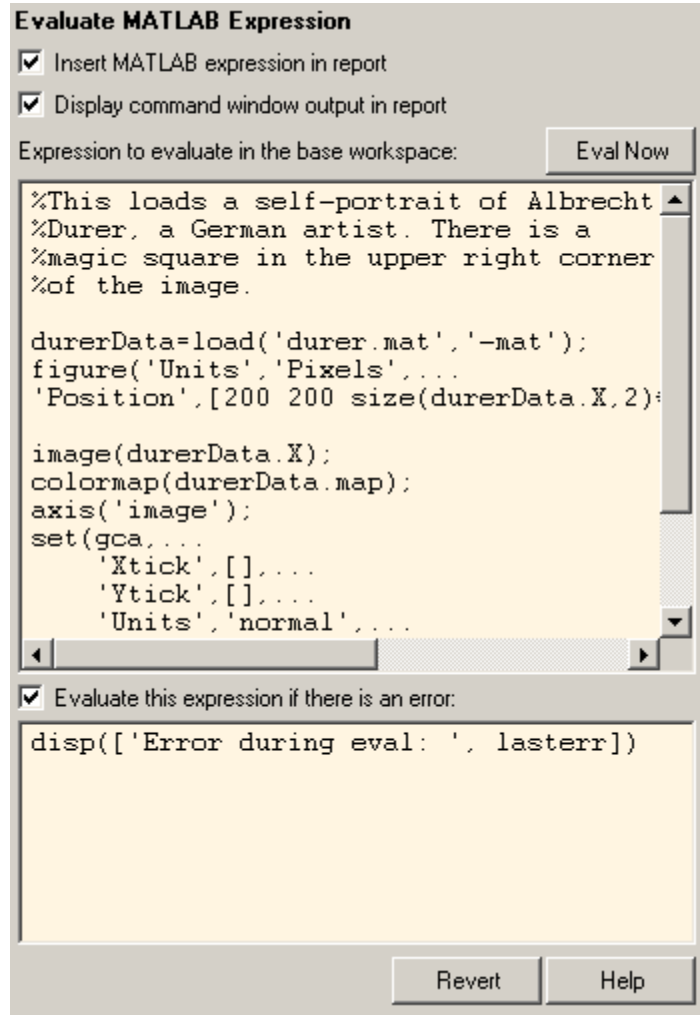
This M-code displays the Dürer etching in a MATLAB figure window.

c In the **Evaluate expression if there is an error** text box, replace the existing text with the following text:

```
disp(['Error during eval: ', lasterr])
```

This code executes if an error occurs while loading the Dürer etching.

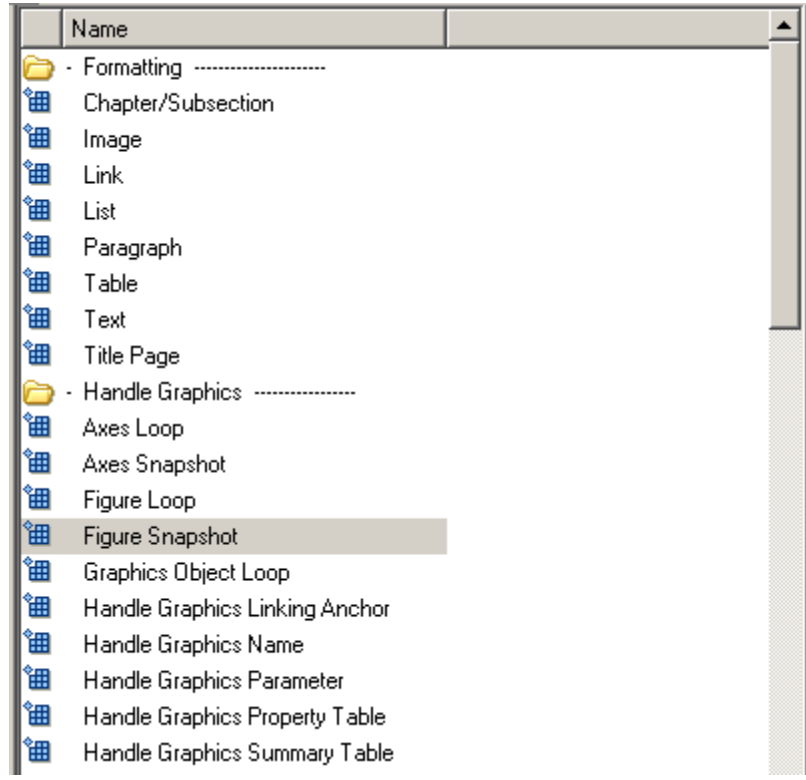
The Properties pane on the right should look as follows.



Now that the image of the Dürer etching is loaded in the MATLAB workspace, include it in the report using the Figure Snapshot component.

14 In the Outline pane on the left, select the `Eval` component.

15 In the Options pane in the middle, under the `Handle Graphics` category, double-click `Figure Snapshot`.



16 In the Properties pane on the right, do the following:

- In the **Paper** orientation drop-down list, select `Portrait`.
- In the **Invert hardcopy** drop-down list, select `Don't invert`.

  Selecting this option specifies not to change the image's on-screen colors for printing.

Now that the image has been added to the report, you can delete it.

17 In the Outline pane on the left, select the `Figure Snapshot` component.

18 In the Options pane in the middle, under the `MATLAB` category, double-click `Evaluate MATLAB Expression`.

19 In the Properties pane on the right, do the following:

- Clear the **Insert MATLAB expression in report** and **Display command window output in report** check boxes.

  You do not want to include the code or its output in the report.

- In the **Expression to evaluate in the base workspace** text box, replace the existing text with the following text:

  ```
  %This command deletes the D rer image
  delete(gcf);
  ```

  The `delete(gcf)` command deletes the current image in the MATLAB workspace, in this case, the Dürer etching.

- In the **Evaluate expression if there is an error** text box, replace the existing text with the following text:

  ```
  disp(['Error during eval: ', lasterr])
  ```

  This code executes if an error occurs while deleting the Dürer etching.

20 Save the report.

The contents of the first chapter are now complete.

Creating the Magic Squares and Their Images

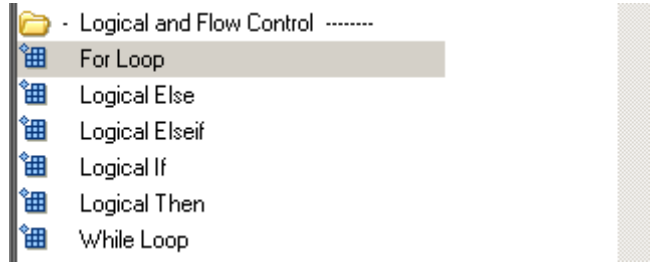
In this section, the components create several magic squares and insert the contents of the squares, or representative images, into the report.

Creating the For Loop

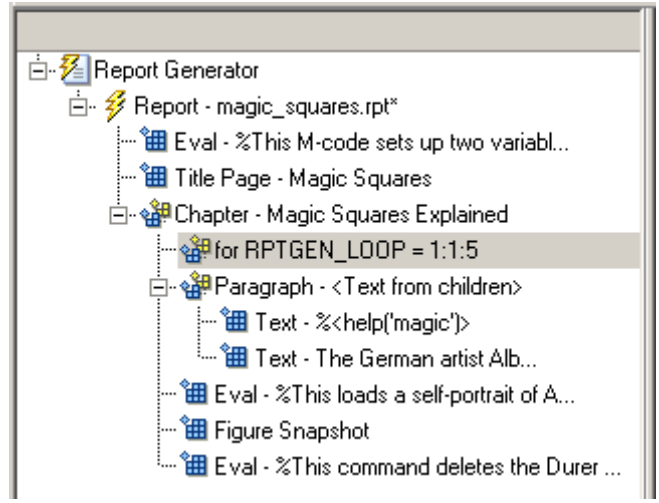
Each square has its own chapter. A For Loop component performs the desired actions for each square.

1 In the Outline pane on the left, select the `Chapter` component.

2 In the Options pane in the middle, under the `Logical and Flow Control` category, double-click `For Loop`.



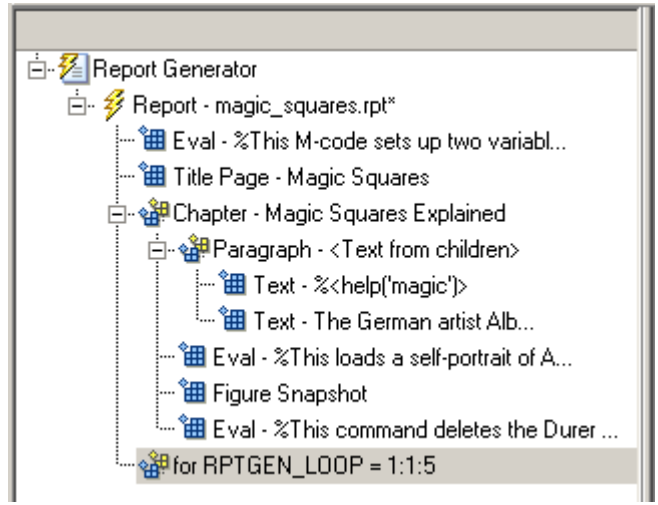
The Outline pane on the left should look as follows.



This component is added inside the `Chapter` component. However, the magic squares should be processed *after* the first chapter, so the `for` component should be a sibling of the `Chapter` component, not a child.

3 In the Outline pane on the left, select the `for` component.

4 Click the **left** arrow to make the `for` component a sibling, not a child, of the `Chapter` component.



5 In the Properties pane, do the following:

- In the **End** text box, replace the existing text with the following text:

  ```
  length(magicSizeVector)
  ```

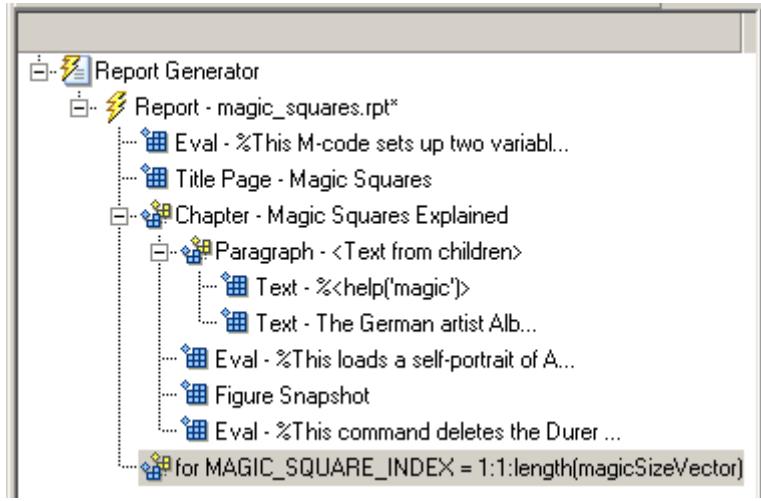
  This is the length of the vector that contains the various sizes for the magic square matrices.

- In the **Variable name** text box, replace the existing text with the following text:

  ```
  MAGIC_SQUARE_INDEX
  ```

  This variable will act as a loop index.

The Outline pane on the left should look as follows.



6 Save the report.

Adding a Chapter for Each Square

Add a chapter for each square processed using the Chapter/Subsection component.

1 In the Outline pane on the left, select the `for` component.

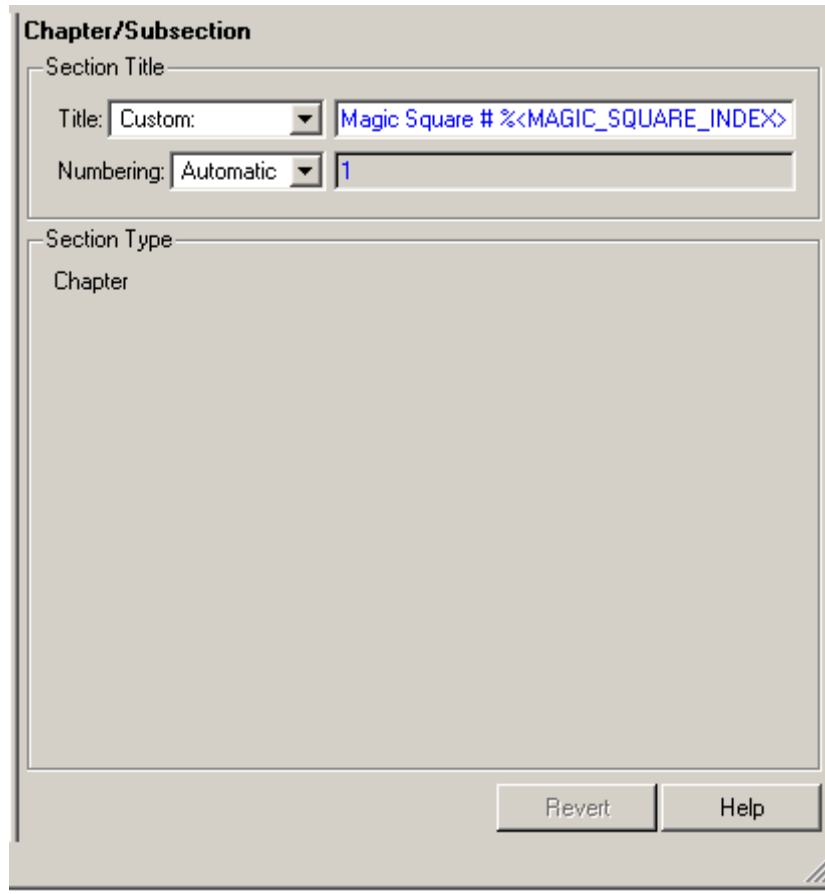
2 In the Options pane in the middle, under the `Formatting` category, double-click `Chapter/Subsection`.

It becomes a child of the `for` component.

3 In the Properties pane on the right, select `Custom` from the **Title** drop-down list and enter the following for the chapter title:

```
Magic Square # %<MAGIC_SQUARE_INDEX>
```

The Properties pane should look as follows.



4 Save the report.

Determining the Matrix Size

You need to extract the size of each magic square matrix from `magicSizeVector` using an Evaluate MATLAB Expression component.

1 In the Outline pane on the left, select the `Chapter` component.

2 In the Options pane in the middle, under the `MATLAB` category, double-click `Evaluate MATLAB Expression`.

3 In the Properties pane on the right, do the following:

- Clear the **Insert MATLAB expression in report** and **Display command window output in report** check boxes.
- In the **Expression to evaluate in the base workspace** text box, replace the existing text with the following text:

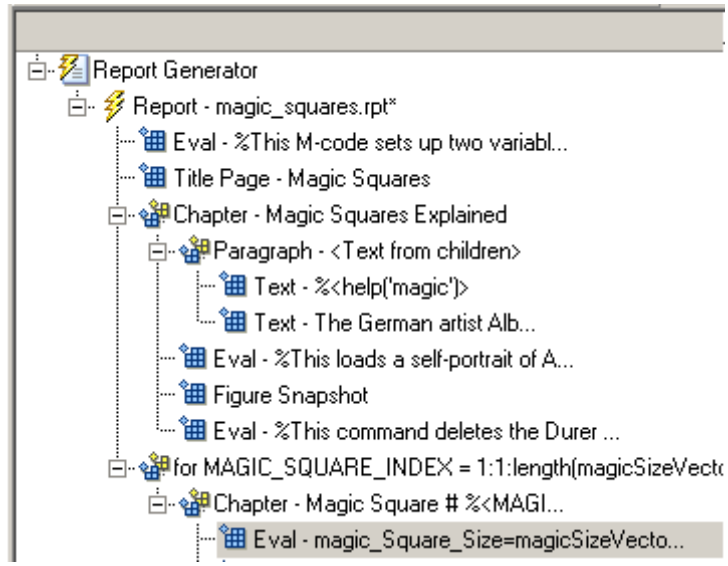
  ```
  magic_Square_Size=magicSizeVector(MAGIC_SQUARE_INDEX);
  ```

  This command extracts the next size for the magic square from the vector of sizes initialized in the first `Eval` component of the report. The variable `magic_Square_Size` represents the size of the current magic square being processed.
- In the **Evaluate expression if there is an error** text box, replace the existing text with the following:

  ```
  disp(['Error during eval: ', lasterr])
  ```

  This code executes if an error occurs while attempting to extract a value from `magicSizeVector`.

The Outline pane on the left should look as follows.



4 Save the report.

Inserting the Magic Square Size into the Report

Insert the size of the magic square into the report using the Paragraph and Insert Variable components.

1 In the Outline pane on the left, select the `Eval` component.

2 In the Options pane in the middle, under the `Formatting` category, double-click `Paragraph`.

Do not change the properties. The variable that contains the size of the magic square will go in this paragraph.

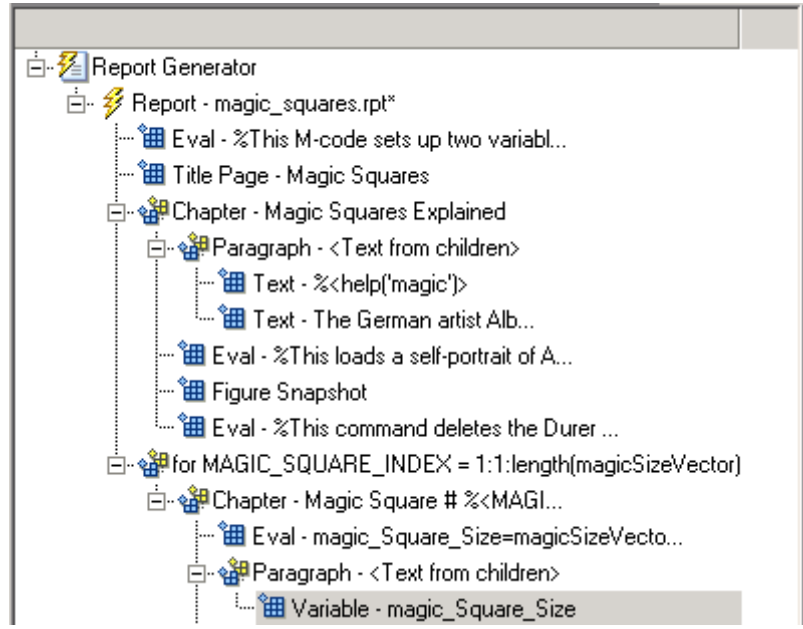
3 In the Outline pane on the left, select the `Paragraph` component.

4 In the Options pane in the middle, under the `MATLAB` category, double-click `Insert Variable`.

5 In the Properties pane on the right, do the following:

- In the **Variable name** text box, enter `magic_Square_Size`.
- In the **Display as** drop-down list, select `Inline text`.

The Outline pane on the left should look as follows.



6 Save the report.

Displaying the Magic Square

To create the magic square and display the associated matrix or image, use the Evaluate MATLAB Expression component.

1 In the Outline pane on the left, select the `Paragraph` component.

2 In the Options pane in the middle, under the `MATLAB` category, double-click `Evaluate MATLAB Expression`.

   You need to make this component a sibling of the `Paragraph` component, not a child.

3 In the Outline pane on the left, select the `Eval` component.

4 Click the left arrow on the toolbar to make the `Eval` component a sibling of the previous `Paragraph` component.

5 In the Properties pane on the right, do the following:

- Clear the **Insert MATLAB expression in report** and **Display command window output in report** check boxes.
- In the **Expression to evaluate in the base workspace** text box, replace the existing text with the following M-code.

Tip You can copy and paste this text from the HTML documentation into the Report Explorer.

```
%This m-script produces a magic
%square of size magic_Square_Size
%and creates an image of that square.

mySquare=magic(magic_Square_Size);
clf
imagesc(mySquare);
title(sprintf('Magic Square N=%i',magic_Square_Size))
set(gca,'Ydir','normal');
axis equal;
axis tight;
```

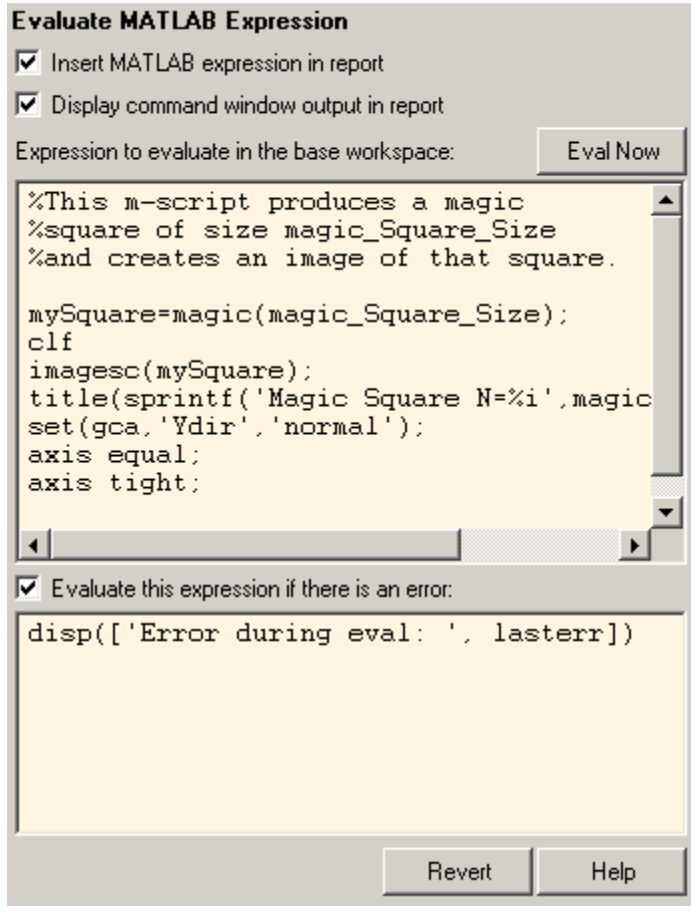
This code creates a magic square matrix `mySquare` of size `magic_Square_Size`, and opens an image of that matrix in the MATLAB Figure window.

- In the **Evaluate expression if there is an error** text box, replace the existing text with the following:

```
disp(['Error during eval: ', lasterr])
```

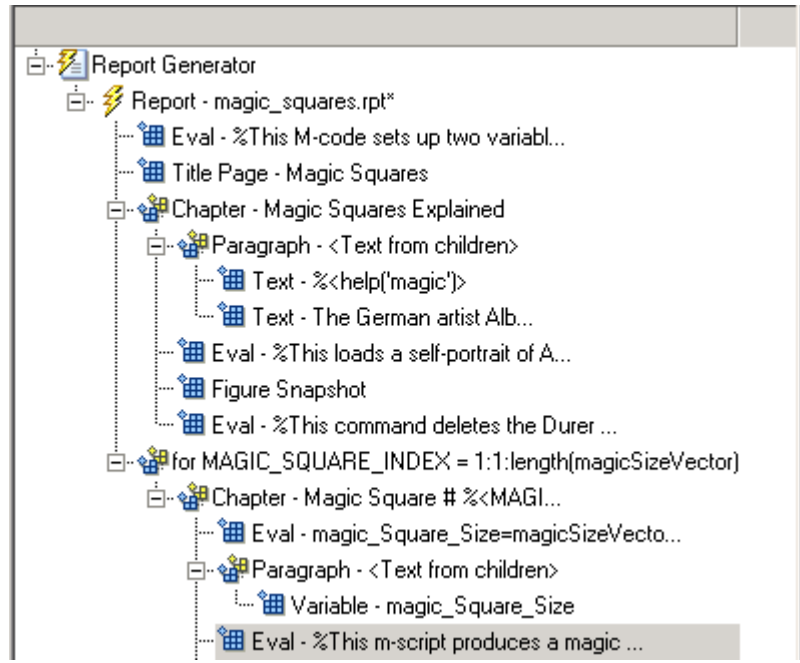
This code executes if an error occurs while creating and displaying the magic square.

The Properties pane on the right should look as follows.



6 Save the report.

7 In the Outline pane on the left, select the `Eval` component.



8 On the Options pane in the middle, under the `Logical and Flow Control` category, double-click `Logical If`.

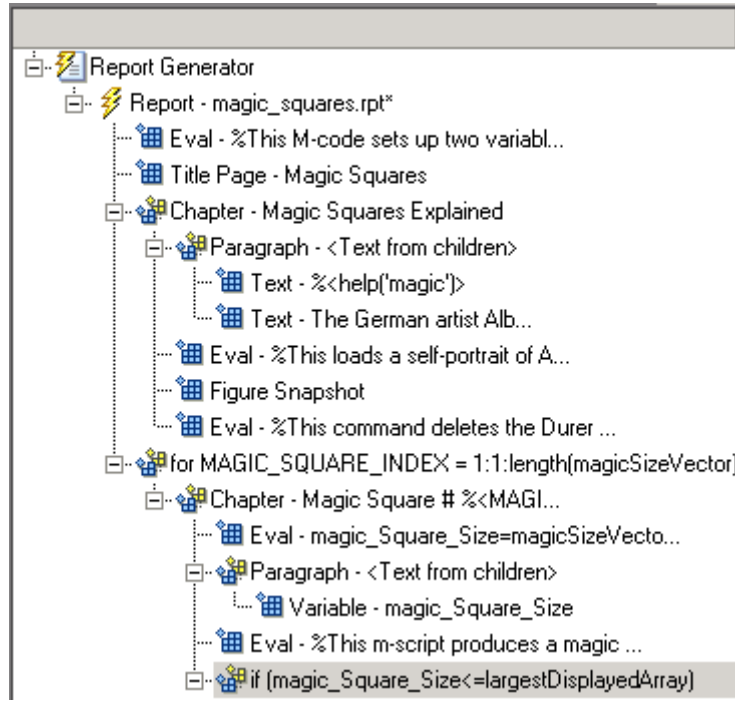
9 On the Properties pane on the right, in the **Test Expression** text box, replace the existing text with the following text:

```
magic_Square_Size<=largestDisplayedArray
```

This command tests if the current matrix size (`magic_Square_Size`) is less than or equal to the value assigned in the first `Eval` component of the report (`largestDisplayedArray=15`).

To deal with the result of this `Logical If` component, you will create two child components—`Logical Then` and `Logical Else`. If `magic_Square_Size` is less than or equal to 15, the matrix variable is inserted into the report. If `magic_Square_Size` is greater than 15, the matrix image is inserted into the report.

10 On the Outline pane on the left, select the `if` component.

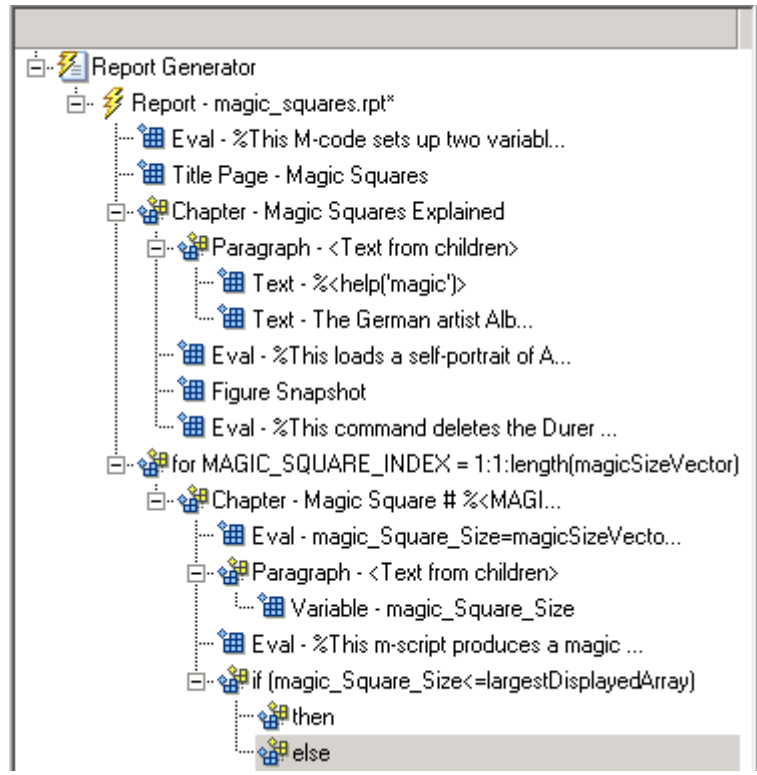


11 On the Options pane in the middle, under `Logical and Flow Control`, double-click `Logical Else`.

12 On the Outline pane on the left, select the `if` component again.

13 On the Options pane in the middle, under `Logical and Flow Control`, double-click `Logical Then`.

The `then` component is added above the `else` component.



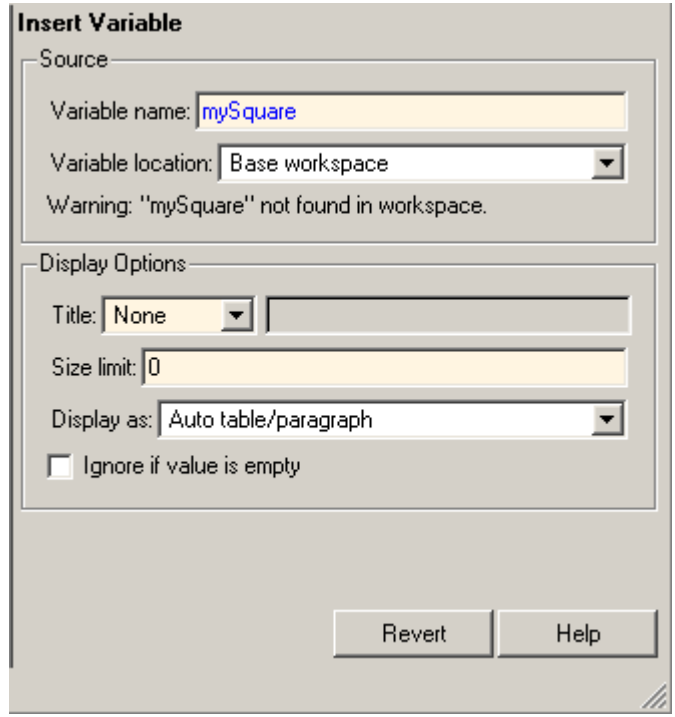
14 In the Outline pane on the left, select the `then` component.

15 In the Options pane in the middle, under the `MATLAB` category, double-click `Insert Variable`.

16 In the Properties pane on the right, do the following:

- In the **Variable name** text box, enter `mySquare`, which is the variable that contains the magic square of the specified size.
- In the **Title** drop-down list, select `None`.
- In the **Size Limit** text box, enter `0`.

  The Properties pane on the right should look as follows.



This `Variable` component displays the magic square matrix, stored in the variable `mySquare`.

17 In the Outline pane on the left, select the `else` component.

18 In the Options pane in the middle, under the `Handle Graphics` category, double-click `Figure Loop`.

Do not change the properties.

19 In the Outline pane on the left, select the `Figure Loop` component.

20 In the Options pane in the middle, under the `Handle Graphics` category, double-click `Figure Snapshot`.

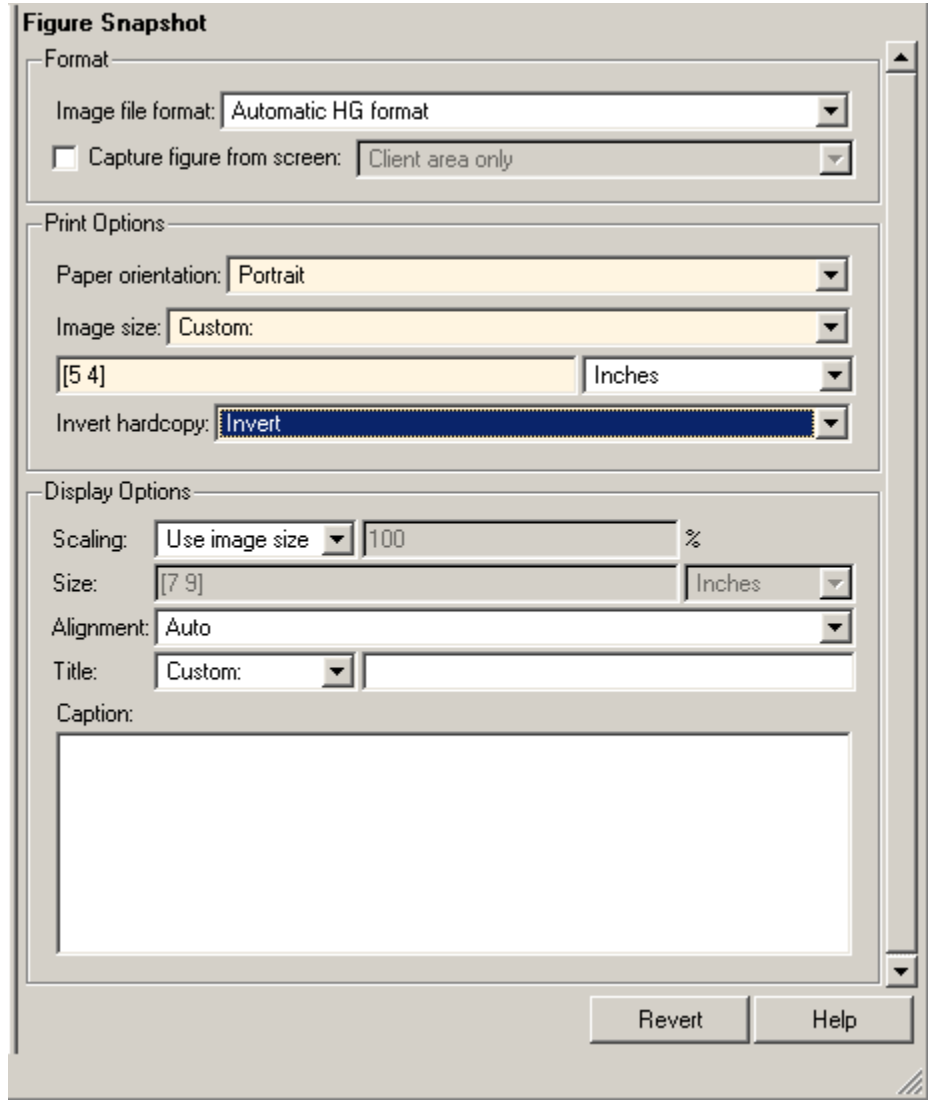
21 In the Properties pane on the right, do the following:

- In the **Paper orientation** drop-down list, select `Portrait`.

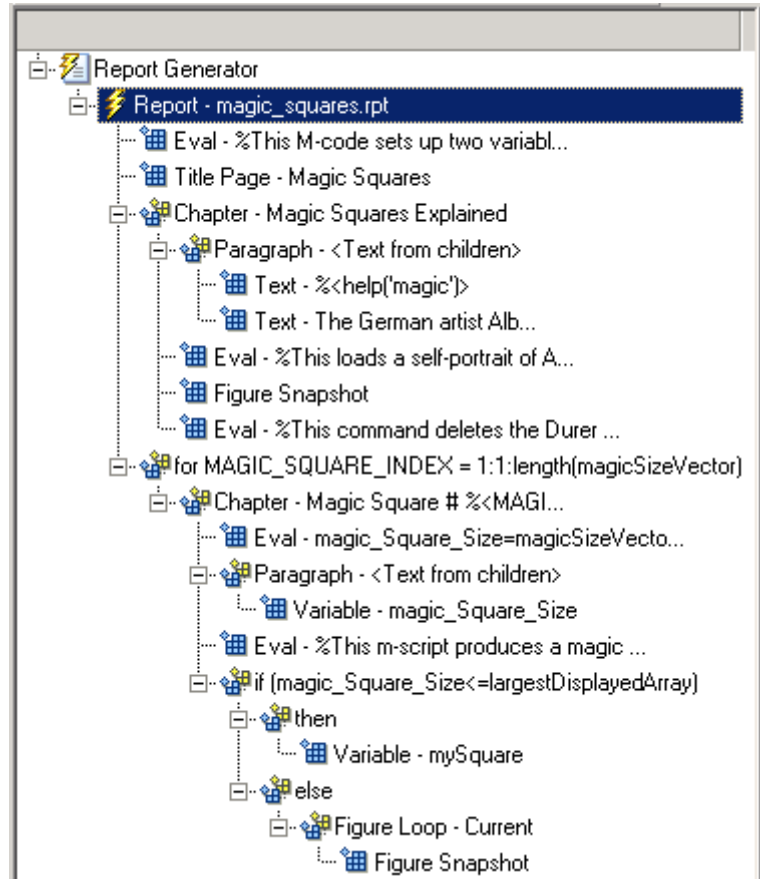
- In the **Image size** drop-down list, select `Custom`.
- Below the **Image size** drop-down list, enter `[5 4]` for the custom image size.
- In the **Invert hardcopy** drop-down list, select `Invert`.

  This option changes dark axes colors to light axes colors, and vice versa.

  The Properties pane on the right should look as follows.



If your Outline pane on the left looks like the following figure, you can generate the report, as described in “Generating the Report” on page 3-52.

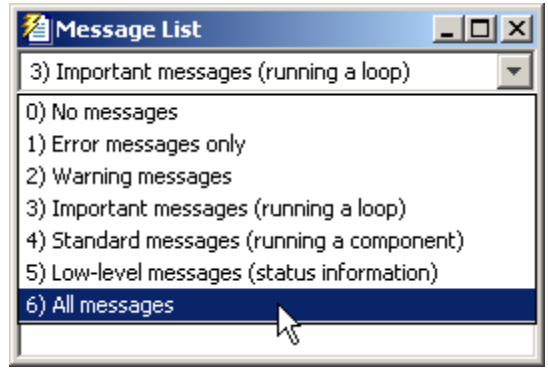

Report Generator
Report - magic_squares.rpt
Eval - %This M-code sets up two variabl...
Title Page - Magic Squares
Chapter - Magic Squares Explained
Paragraph - <Text from children>
Text - %<help('magic')>
Text - The German artist Alb...
Eval - %This loads a self-portrait of A...
Figure Snapshot
Eval - %This command deletes the Durer ...
for MAGIC_SQUARE_INDEX = 1:1:length(magicSizeVector)
Chapter - Magic Square # %<MAGI...
Eval - magic_Square_Size=magicSizeVecto...
Paragraph - <Text from children>
Variable - magic_Square_Size
Eval - %This m-script produces a magic ...
if (magic_Square_Size<=largestDisplayedArray)
then
Variable - mySquare
else
Figure Loop - Current
Figure Snapshot

Generating the Report

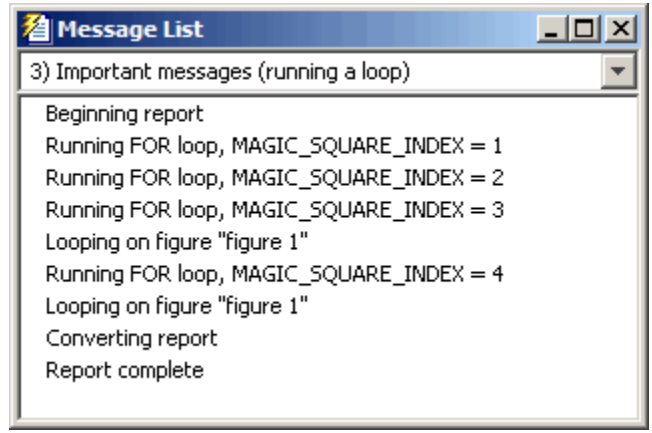
Now the report contains all the components it needs.

Click the **Report** icon on the toolbar to generate the report. You will see the following displayed on your screen:

- A Message List window appears, displaying informational and error messages as Report Generator is processing the report. While the report is being generated, you can specify the level of detail you would like the Message List window to display. Options range from 0 (least detail) to 6 (most detail). Click the drop-down list located under the title bar of the Message List window to choose an option, as shown in the figure below.

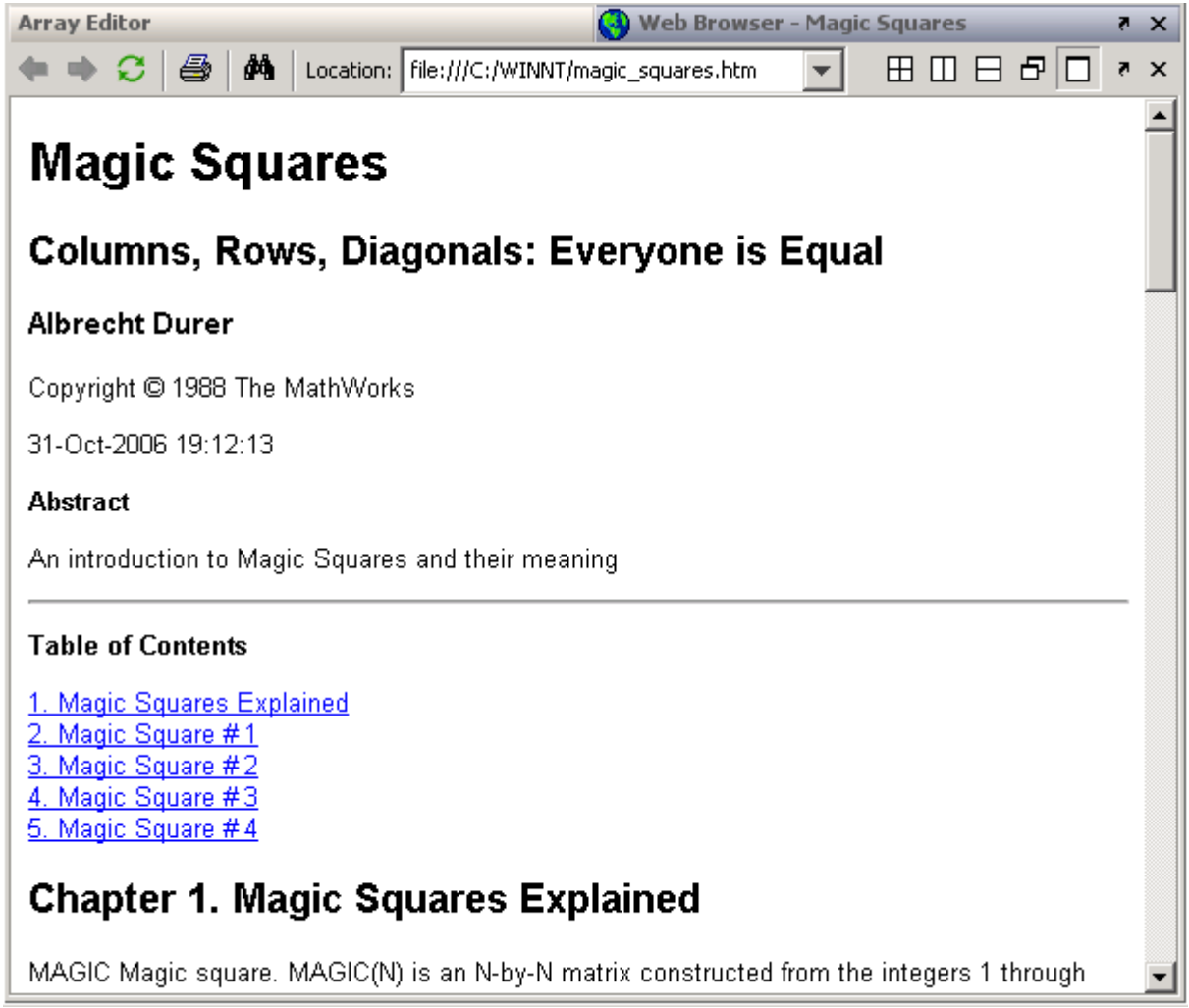


Message level 3 (Important messages) is used for the remainder of this example.



- An image of the etching appears briefly.
- Images of two magic square images of sizes 16 and 32 appear briefly.
- If you watch the Outline pane on the left of your Report Explorer window, each component of the report template is highlighted as it is executed.

Recall in the beginning of this tutorial you specified HTML as the output format of this report. When Report Generator is finished processing, the MATLAB Web browser opens with the report's HTML file.


Array Editor
Web Browser - Magic Squares
Location:
file:///C:/WINNT/magic_squares.htm
Magic Squares
Columns, Rows, Diagonals: Everyone is Equal
Albrecht Durer
Copyright © 1988 The MathWorks
31-Oct-2006 19:12:13
Abstract
An introduction to Magic Squares and their meaning
Table of Contents
1. Magic Squares Explained
2. Magic Square #1
3. Magic Square #2
4. Magic Square #3
5. Magic Square #4
Chapter 1. Magic Squares Explained
MAGIC Magic square. MAGIC(N) is an N-by-N matrix constructed from the integers 1 through

3

Creating Reports in Simulink

This tutorial covers the following tasks, which apply to Simulink users.

About This Tutorial (p. 3-2)	Describes Simulink Report Generator features that are demonstrated in this tutorial
Setting Report Options in the Report Template (p. 3-3)	How to create a report template and specify options for the generated report
Adding Components to the Report Template (p. 3-8)	How to build a report by adding components to its template
Generating the Report (p. 3-52)	How to generate a report after defining its components

About This Tutorial

Report Generator features demonstrated in this tutorial include the following:

- Evaluating MATLAB expressions
- Inserting the values of workspace variables
- Creating a report with chapters and subsections
- Running loops and flow control
- Handling errors
- Inserting Simulink models and scopes
- Creating a table using workspace data

This tutorial takes you through the steps required to design a report template and generate a report in Simulink using the Report Explorer GUI. This report opens a Simulink model for the van der Pol equation, sets the `Gain` parameter for the Mu block to five different values, simulates the model each time, and collects the results. Results that fall within a specified range appear in a table in the final report.

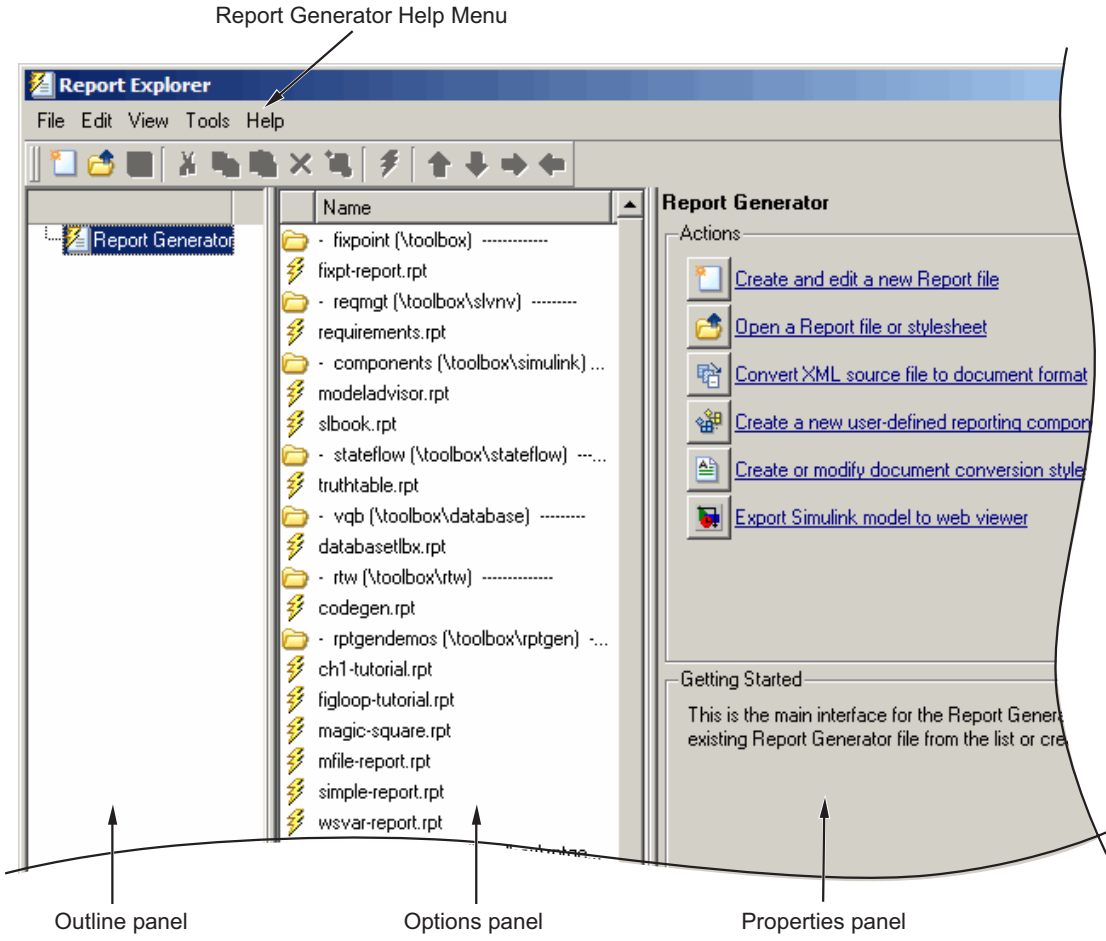
Note The directions for this tutorial refer to this model as the *vdp model*.

Note You do not need to know MATLAB or Simulink to execute this tutorial, but it might be helpful for understanding the M-code and model simulation that executes.

Setting Report Options in the Report Template

To create and configure your report template:

1. Start MATLAB.
2. Type `report` in the MATLAB command window. The Report Explorer window opens.



Tip If the Report Explorer window opens with only two panes, one of the panes is hidden. The vertical boundaries can be moved to reveal any hidden pane.

The Report Explorer has three panes:

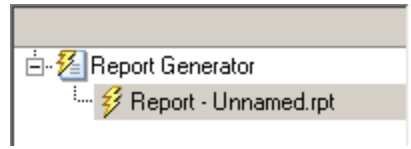
- The Outline pane on the left shows the hierarchy of components in all currently opened reports. Report components can reside within other report components, creating parent, child, and sibling relationships.
- The Options pane in the middle lists the options available in the context of the Outline pane. Because no report is open, the Options pane lists the available reports. When a report is open, the Options pane lists the available components. When a stylesheet is open, the Options pane lists the available attributes.
- If no report is open, the Properties pane on the right displays the tasks the Report Explorer can perform. If a report is open, the Properties pane displays the properties for the element currently selected in the Options pane.

These three panes can be made wider or narrower by dragging the vertical boundaries between the panes.

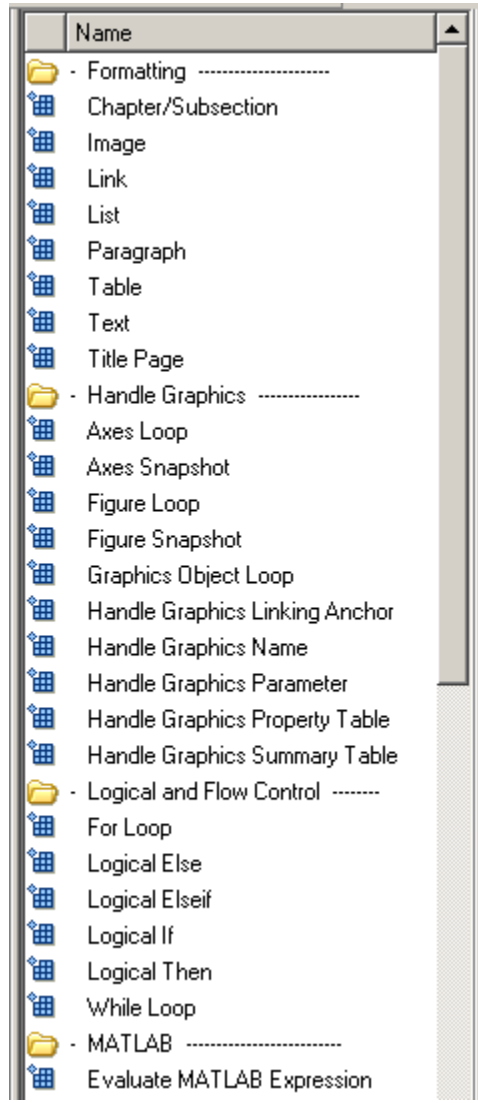
3 Select **File > New** to create a new report.

The three Report Explorer panes have the following contents.

The Outline pane initially contains the top level of the report.



Outline Pane



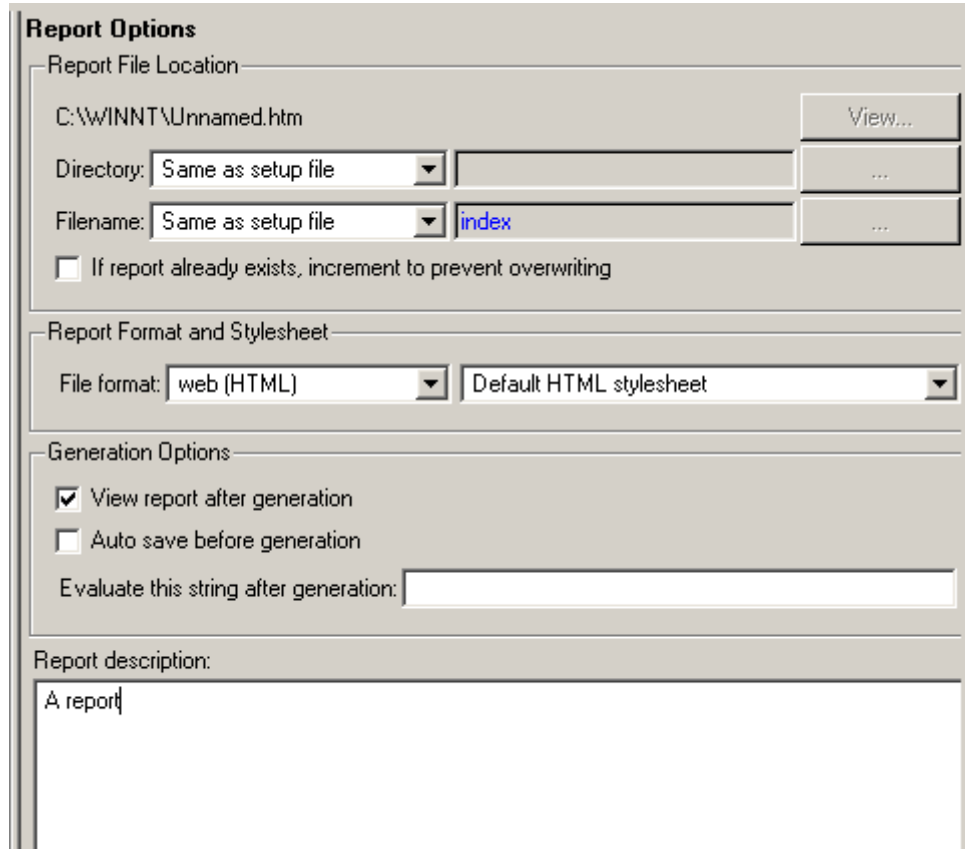
Options Pane

The Options pane lists the component categories and components available for adding to reports. In this figure, the categories shown are `Formatting`,

`Handle Graphics`, `Logical and Flow Control`, and `MATLAB`. For details about report components, see Chapter 10, “Components — By Category”.

The folder icon indicates a component category.

The blue square icon indicates a component.



Properties Pane

After creating a new report, the Properties pane initially displays the properties for the report as a whole.

4 In the Properties pane on the right, do the following:

- To save the report in the current working directory, select `Present Working Directory` from the **Directory** drop-down list.
- In the **File format** drop-down list, select `Acrobat (PDF)`.
- In the **Report description** text box, replace the existing contents with the following text.

Tip You can copy and paste this code from the HTML documentation into the Report Explorer.

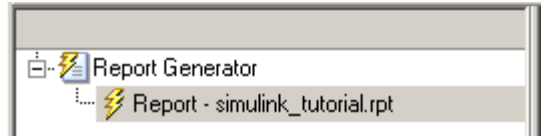
```
Simulink Dynamic Report

This report opens up a model, sets a block parameter
several times, simulates the model, and collects the
results. Results that fall within a specified range are
displayed in a table after the test is complete.

The report is configured to test the vdp model only.
By selecting the Eval String component immediately
below the Report component, you can modify
* model
* block
* parameter
* tested values
```

5 Select **File > Save As** and name your report's report template `simulink_tutorial.rpt`.

The Outline pane on the left reflects the new file name.



Adding Components to the Report Template

The following topics are covered in this section:

Report components specify what information you want included in the report. The following figure shows a sample page from the report you create in this tutorial and what components are used to produce this output.

Note Report components added to the report template must not be deactivated in order for the report to be correctly generated.

Chapter/ Subsection component →

Chapter 1. Model - vdp

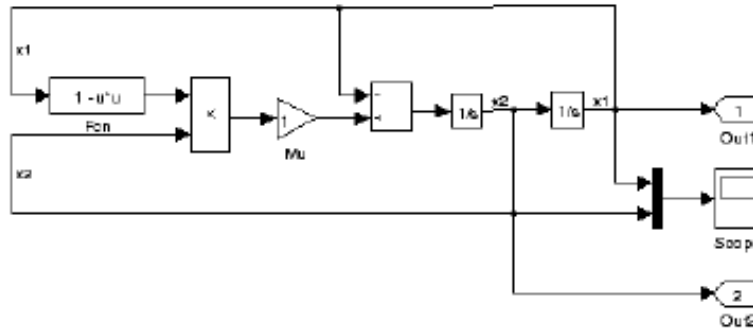
Paragraph component →

This report demonstrates Report Generator's ability to experiment with Simulink systems and auto-document the results. In this report, you load the model vdp and simulate it length times. This report modifies the vdp/Mu block's "Gain" value, setting it to the values [-1 0 0.5 1 2] . Each iteration of the test includes a set of scope snapsnots in the report.

?

van der Pol Equation

System Snapshot component →



Chapter/ Subsection component →

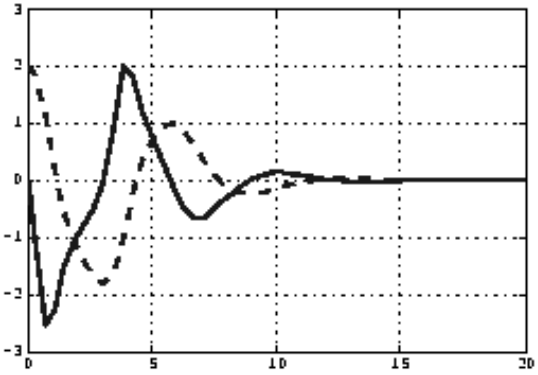
Processing the vdp model

Insert Variable component →

Iteration_Value -1

Figure 1.1. Scope

Scope Snapshot component →

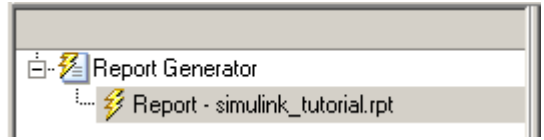


Time offset: 0

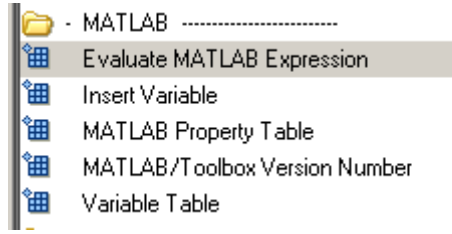
Adding MATLAB Code

The first component to add is the Evaluate MATLAB Expression component, which evaluates MATLAB commands in the workspace. For this tutorial, the code in this component assigns some initial values.

1 In the Outline pane on the left, select the top level of the report template.



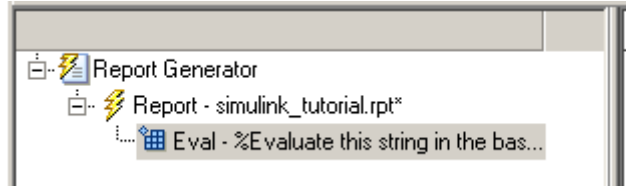
2 In the Options pane in the middle, under the `MATLAB` category, select `Evaluate MATLAB Expression`.



3 In the Properties pane on the right, click the icon next to **Add component to current report** to insert that component into the report.

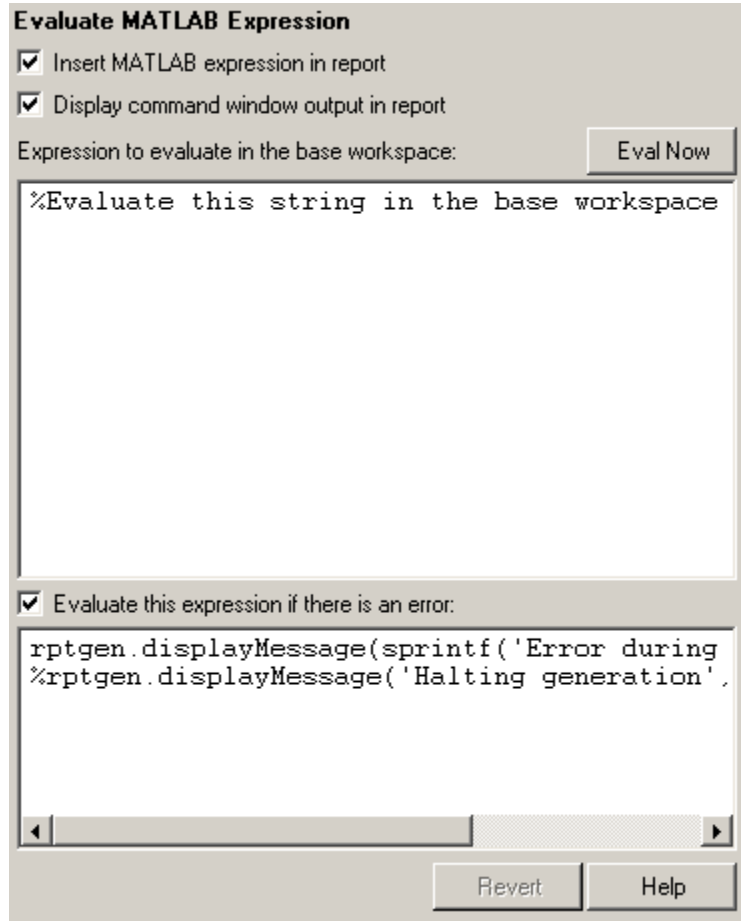
You cannot edit the component information in the Properties pane on the right until you have added the component to the report.

In the Outline pane on the left, the `Evaluate MATLAB Expression` component is listed under the `simulink_tutorial` report template and the component name is abbreviated to `Eval` in the Outline pane.



The icon next to the `Eval` component means that this component cannot have any child components. By default, any components you add with the `Eval` component selected are added as siblings to this component.

The options for the `Evaluate MATLAB Expression` component appear in the Properties pane on the right.



4 Clear the **Insert MATLAB expression in report** and the **Display command window output in report** check boxes.

You do not want to include the MATLAB code details or its output in this report.

5 In the **Expression to evaluate in the base workspace** text box, replace the existing text with the following M-code.

Tip You can copy and paste this code from the HTML documentation into the Report Explorer.

```
%The name of the model
%that will be changed
expModel='vdp';

%The name of the block in the model
%that will be changed
expBlock='vdp/Mu';

%The name of the block parameter
%that will be changed
expParam='Gain';

%The values that will be set
%during experimentation
expValue=[-1 0 .5 1 2];

%expValue can be either a vector
%or a cell array

testMin=2.1;
testMax=3;

%---- do not change code below line ---

try
   open_system(expModel);
end

expOkValues=cell(0,2);
```

This code assigns values to specify the model name, block name, block parameter, parameter values, and other initial values required for processing the vdp model.

Note When you change a field in the Properties pane on the right, the field background changes color (the default is a cream color), meaning there are unapplied changes to that field. As soon as you perform any action with another component, the changes are applied and the background color becomes white again.

---

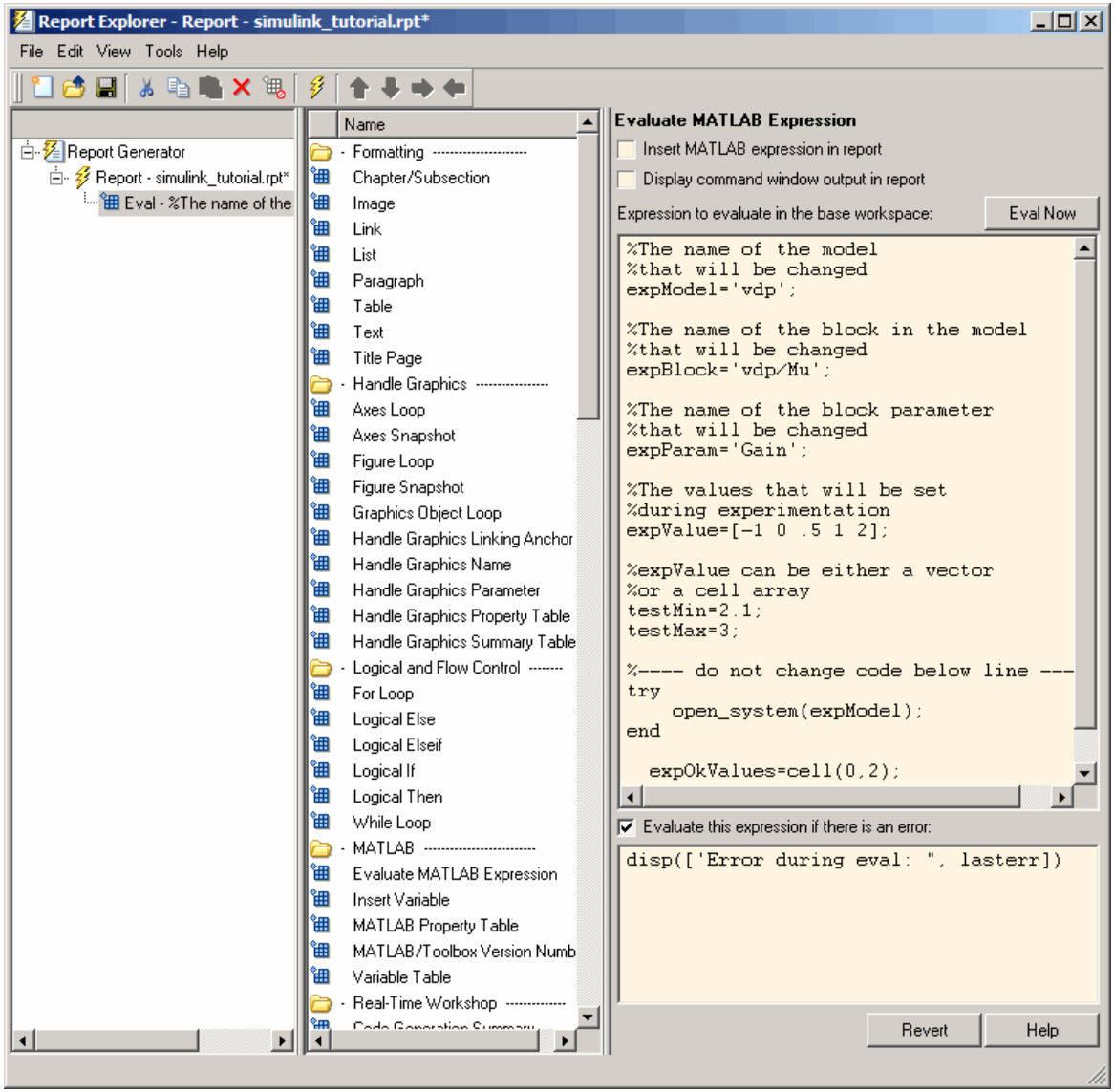
6 Make sure the **Evaluate this expression if there is an error** check box is selected.

7 Under the check box, replace the existing text with the following:

```
disp(['Error during eval: ', lasterr])
```

If the MATLAB code fails, a message appears.

Your Report Explorer window should look as follows.


Report Explorer - Report - simulink_tutorial.rpt*
File Edit View Tools Help
Report Generator
Report - simulink_tutorial.rpt*
Eval - %The name of the
Name
Formatting
Chapter/Subsection
Image
Link
List
Paragraph
Table
Text
Title Page
Handle Graphics
Axes Loop
Axes Snapshot
Figure Loop
Figure Snapshot
Graphics Object Loop
Handle Graphics Linking Anchor
Handle Graphics Name
Handle Graphics Parameter
Handle Graphics Property Table
Handle Graphics Summary Table
Logical and Flow Control
For Loop
Logical Else
Logical Elseif
Logical If
Logical Then
While Loop
MATLAB
Evaluate MATLAB Expression
Insert Variable
MATLAB Property Table
MATLAB/Toolbox Version Numb
Variable Table
Real-Time Workshop
Evaluate MATLAB Expression
Insert MATLAB expression in report
Display command window output in report
Expression to evaluate in the base workspace:
Eval Now
%The name of the model
%that will be changed
expModel='vdp';
%The name of the block in the model
%that will be changed
expBlock='vdp/Mu';
%The name of the block parameter
%that will be changed
expParam='Gain';
%The values that will be set
%during experimentation
expValue=[-1 0 .5 1 2];
%expValue can be either a vector
%or a cell array
testMin=2.1;
testMax=3;
%---- do not change code below line ---
try
open_system(expModel);
end
expOkValues=cell(0,2);
Evaluate this expression if there is an error:
disp(['Error during eval: ', lasterr])
Revert
Help

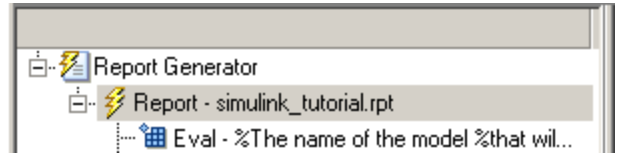
Note You can click the **Eval Now** button, located at the top right corner of the Report Explorer window, to execute the commands that you specified in your MATLAB expression immediately. This is an easy way to check if your commands are correct and will not result in problems at runtime.

8 Save your report.

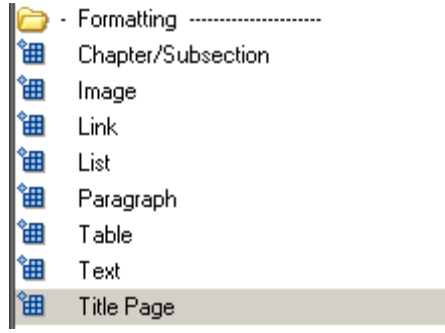
Adding a Title Page

Do the following to create a custom title page for your report using the Title Page component:

1 In the Outline pane on the left, select the `Eval` component so that the next component is added directly below it.



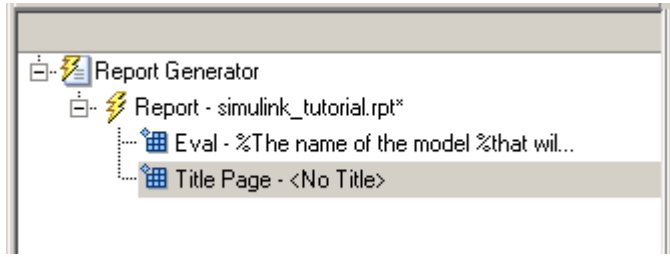
2 In the Options pane in the middle, under the `Formatting` category, click `Title Page`.



In the Properties pane on the right, properties are not editable until a component has been added to a report.

3 Click the icon next to **Add component to current report**.

The properties are now editable, and the `Title Page` component appears in the Outline pane.



The icon in the Outline pane next to the `Title Page` component indicates that it cannot have child components.

4 In the Properties pane on the right, do the following:

- In the **Title** text box, enter:

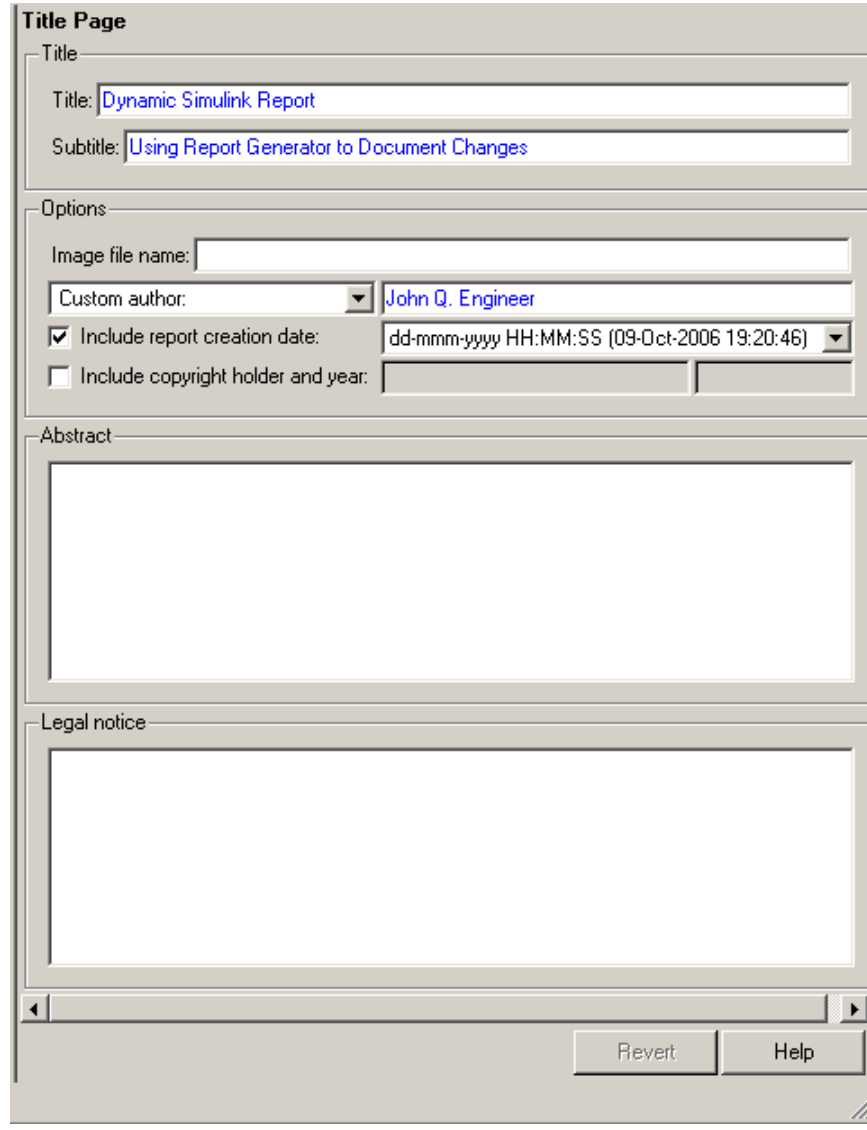
  ```
  Dynamic Simulink Report
  ```

- In the **Subtitle** text box, enter:

  ```
  Using Report Generator to Document Changes
  ```

- In the **Options** section, in the second field, select `Custom Author` from the drop-down list.
- Enter your name in the text box.
- Make sure **Include report creation date** is selected, and select the default date and time format from the drop-down list.

  The Properties pane on the right should look as follows:



5 Save the report by clicking **File > Save**.

Opening the Simulink Model

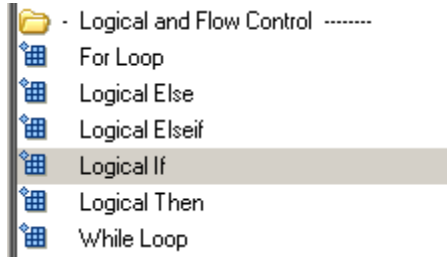
Recall the following statement in the Evaluate MATLAB Expression component that tries to open the vdp model:

```
try
   open_system(expModel);
end
```

Tip Select the `Eval` component in the Outline pane on the left to look at this code again.

To see if the vdp model was successfully opened, test the result of the `open_system` command with a Logical If component:

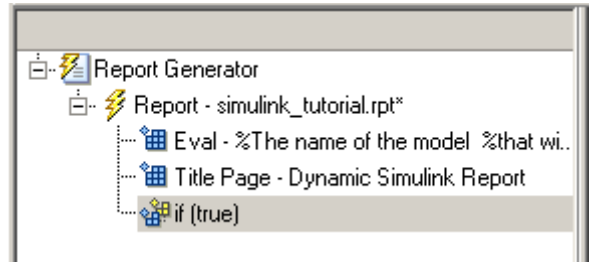
1. In the Outline pane on the left, select the `Title Page` component.
2. In the Options pane in the middle, under the `Logical and Flow Control` category, select `Logical If`.



   The Logical If component checks to see if a given condition is true or false, in this case, if the model opened successfully.

3. In the Properties pane on the right, click the icon next to **Add component to current report**.

   This component appears as `if` in the Outline pane.



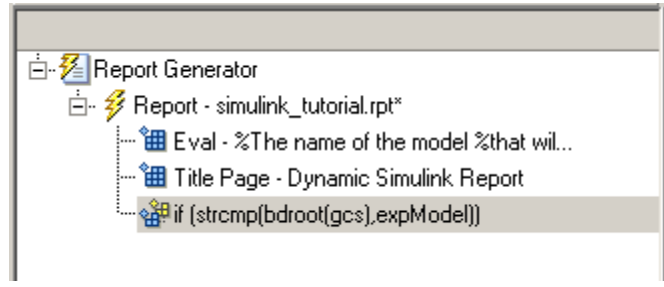
All three components you have added to the report are child components in the report. They are siblings with each other. Components can have parent, child, and sibling relationships.

The icon next to the `if` component indicates it can have child components. "Adding Logical Then and Logical Else Components" on page 3-20 explains how to add two child components to the `if` component.

4 In the Properties pane on the right, in the **Test expression** text box, replace the default text, `true`, with the following text:

```
strcmp(bdroot(gcs),expModel)
```

The `if` component name in the Outline pane changes to include this expression.



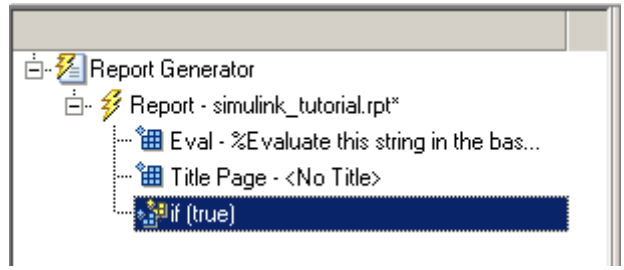
The `strcmp` function compares the name of the open Simulink model and the value of `expModel`, which was set to `'vdp'`. It tests to see if the vdp model opened successfully. `strcmp` returns 1 (true) if the two strings match, and 0 (false) if not.

5 Save the report.

Adding Logical Then and Logical Else Components

The `if strcmp(bdroot(gcs), expModel))` component has two results. To add two child components to the report template to handle those cases:

1. In the Outline pane on the left, select the `if` component.

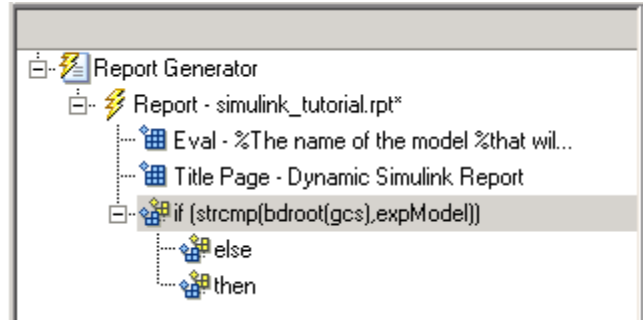


2. In the Options pane in the middle, under the `Logical and Flow Control` category, double-click `Logical Then`.

3. In the Outline pane on the left, select the `if` component again.

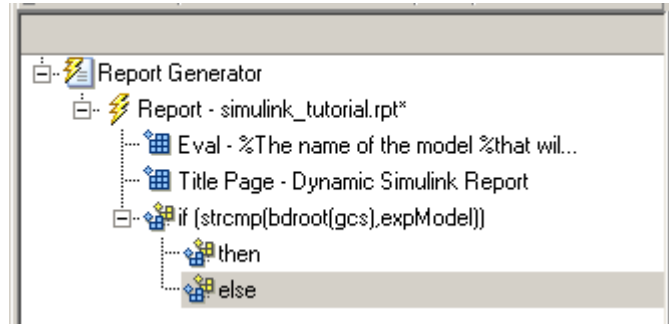
4. In the Options pane in the middle, under the `Logical and Flow Control` category, this time double-click `Logical Else`.

   Both elements are added as child components to the if component, as you can see in the Outline pane.



5. To move the else component below the then component, select the else component and click the **down** arrow on the toolbar once.

The Outline pane on the left looks as follows:



6 Save the report.

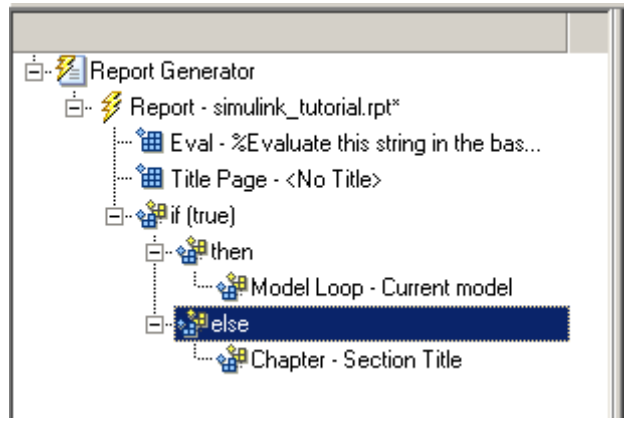
Displaying an Error When the Model Cannot Be Opened

If the `if strcmp(bdroot(gcs), expModel))` component fails (the vdp model could not open), the else component executes. Display an error message prints in the report using the Chapter/Subsection component.

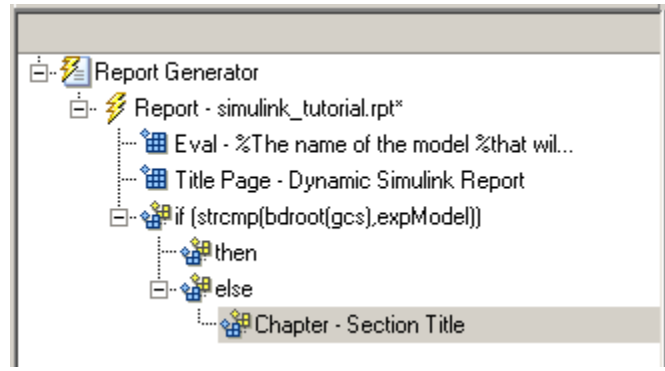
Tip You can also insert plain text into reports using the Paragraph and Text components.

---

1 In the Outline pane on the left, select the `else` component.



2 In the Options pane in the middle, under the `Formatting` category, double-click `Chapter/Subsection` so it is added as a child to the `else` component.



This component displays an error message if an error occurred opening the vdp model.

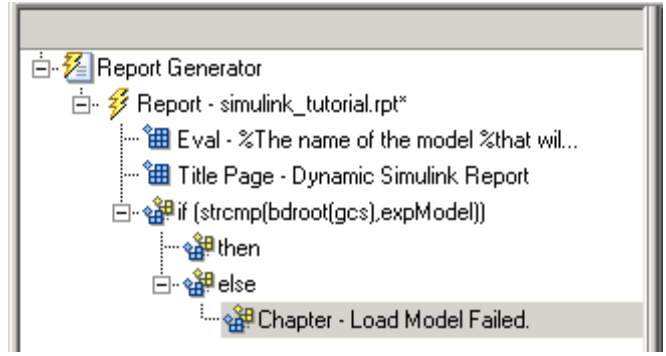
Note When you add a component to a report, it is added by default as a child component unless the selected component cannot have child components.

---

3 In the Properties pane on the right, make sure the **Title** drop-down list is set to `Custom` and enter the following text in the text box:

```
Load Model Failed.
```

The Outline pane should look as follows:



4 In the Outline pane on the left, select the `Chapter` component.

5 In the Options pane in the middle, under `Formatting`, double-click `Paragraph`.

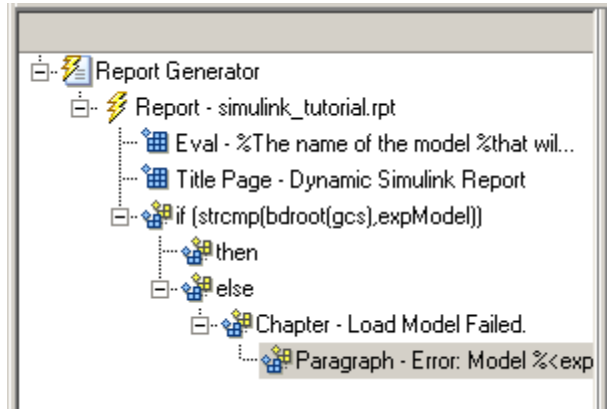
6 In the Properties pane on the right, enter the following text in the **Paragraph Text** text box to display the error message:

```
Error: Model %<expModel> could not be opened.
```

The expression `%<expMode>` indicates to insert the value of the workspace variable `expModel` into the text, as in the following example:

```
Error: Model vdp could not be opened.
```

The Outline pane on the left should look as follows:



7 Save the report.

Creating the Body of the Report

There are several tasks the report needs to perform if the vdp model opens successfully. Each action requires a separate component under the then component.

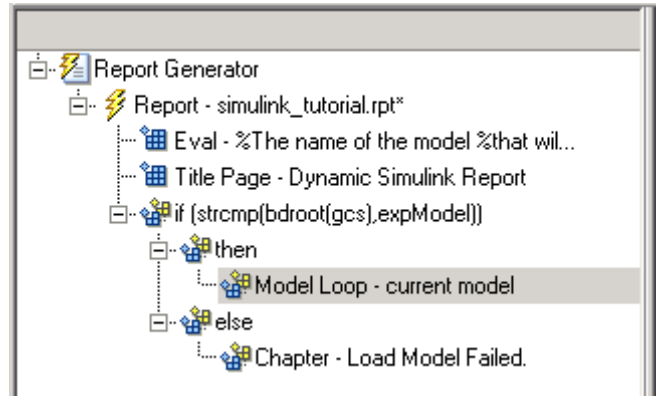
Processing the Model with a Model Loop Component

With the vdp model, the report changes the Gain parameter for the Mu block several times. For that, you need a Model Loop component.

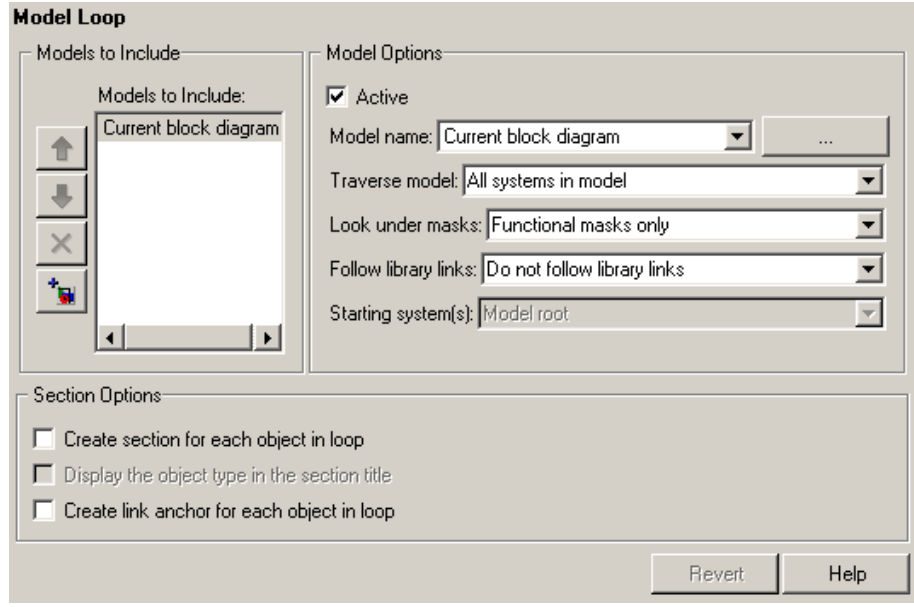
1 In the Outline pane on the left, select the `then` component.

2 In the Options pane in the middle, scroll down to the `Simulink` category.

3 Double-click `Model Loop`. That component is added as a child to the `then` component.



The Properties pane on the right should look as follows.



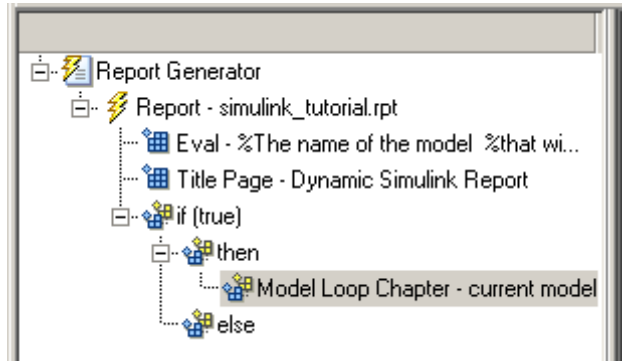
4 In the Properties pane on the right, do the following:

- Select the **Active** check box to process the vdp model.
- In the **Traverse model** drop-down list, select `Selected system(s) only` to traverse only the vdp model.

The next two drop-down lists become disabled.

- Make sure the **Starting system(s)** drop-down list is set to `Model root`.
- At the bottom of the Properties pane on the right, select the **Create section for each object in loop** check box to create a new chapter or section for each model.

  When you select this check box, the component name in the Outline pane on the left changes to `Model Loop Chapter`.



- Select the **Display the object type in the section title** check box to include the object type (in this example, model) in the title name.
- Clear the **Create link anchor for each object in loop** check box.

5 Save the report.

Adding a Paragraph for Each Model

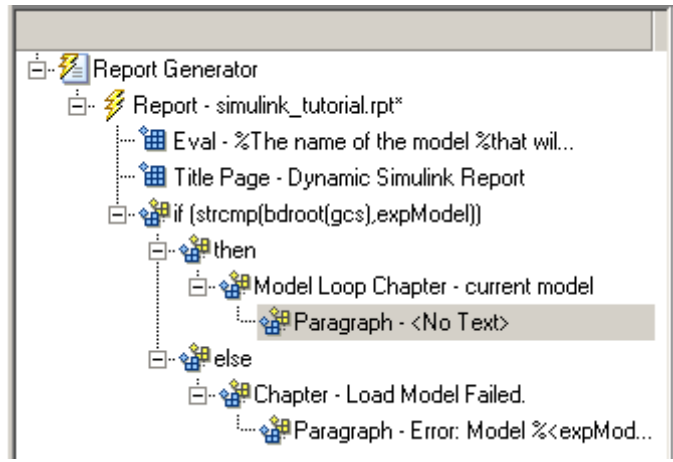
In each Model Loop Chapter, add an explanation using the Paragraph component.

1 In the Outline pane on the left, select the `Model Loop Chapter` component.

2 In the Options pane in the middle, scroll up to the `Formatting` category.

3 Double-click `Paragraph`.

The icon next to the `Model Loop Chapter` component indicates that it can have child components, so the `Paragraph` component is added as a child.



4 In the Properties pane on the right, in the **Paragraph Text** text box, enter the following text.

Tip You can copy and paste this text from the HTML documentation into the Report Explorer.

```
This report demonstrates Report Generator's ability
to experiment with Simulink systems and auto-document
the results. In this report, you load the model %<expModel>
and simulate it %<length> times. This report modifies the
%<expBlock> block's "%<expParam>" value, setting it to the
values %<expValue>. Each iteration of the test includes
a set of scope snapsnots in the report.
```

When this report is generated, the variable names preceded by percent signs (%) and enclosed in brackets (<>) are replaced with the actual values of those variables in the MATLAB workspace.

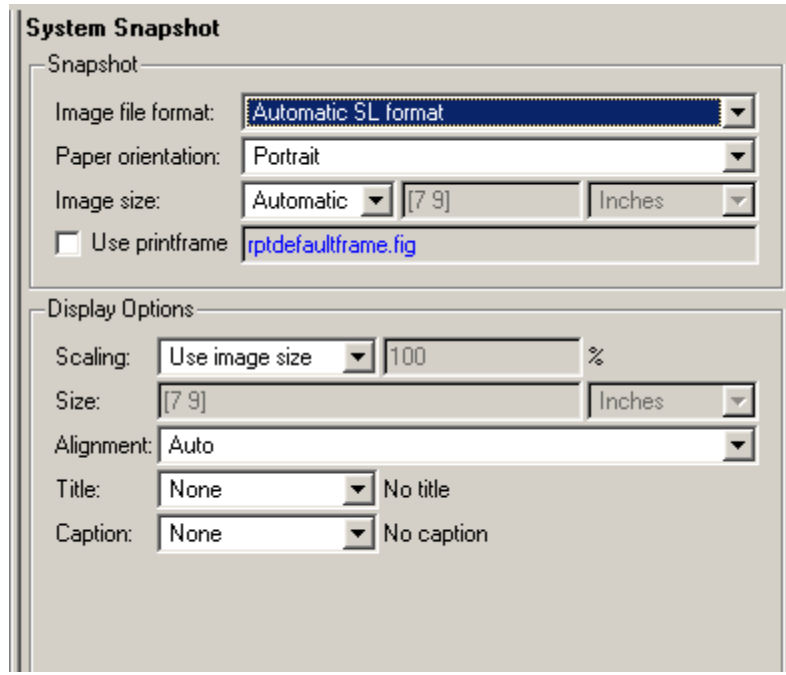
5 Save the report.

Inserting a Snapshot of the Model into the Report

Inside each Model Loop Chapter component, include a snapshot of the current model using the System Snapshot component.

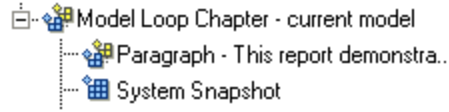
1. In the Outline pane on the left, select the `Model Loop Chapter` component.
2. In the Options pane in the middle, scroll down to the `Simulink` category.
3. Double-click the `System Snapshot` component.

   This component inserts an image of the current model into your report. The Properties pane on the right should look as follows:



4. In the Properties pane on the right, from the **Scaling** drop-down list, select `Zoom`, and enter `70` as the % value.
5. In the Outline pane on the left, select the `System Snapshot` component.

6 Click the **down** arrow on the toolbar once to move it below the `Paragraph` component.



7 Save the report.

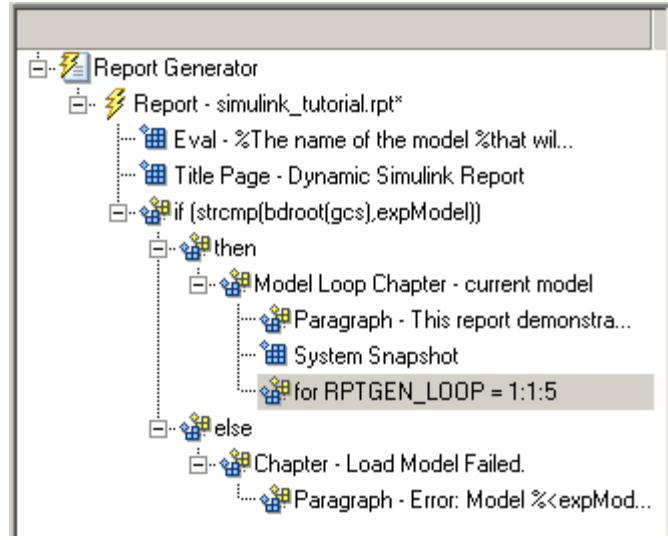
Adding a Loop for Processing the Model

Create a loop to process the model `%length` times using the For Loop component.

1 In the Outline pane on the left, select the `System Snapshot` component.

2 In the Options pane in the middle, under the `Logical and Flow Control` category, double-click `For Loop`.

Since the System Snapshot component cannot have child components, the `For Loop` component is added as a sibling of the `System Snapshot` component.



3 In the Properties pane on the right, do the following:

- In the **End** text box, replace the existing text with the following text:

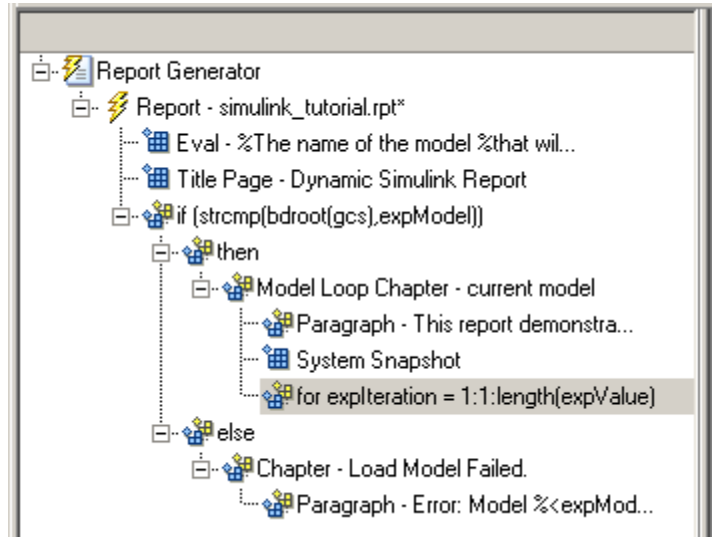
  ```
  length(expValue)
  ```

  `expValue` is the array of Gain parameter values assigned in the `Eval` component with the command `expValue=[-1 0 .5 1 2];`. The expression `length(expValue)` evaluates to 5 for this example.

- In the **Variable name** text box, replace the existing text with the name of the for loop variable, in this case,

  ```
  expIteration
  ```

  The name of the `for` component in the Outline pane on the left changes to reflect the loop variable and the termination value.

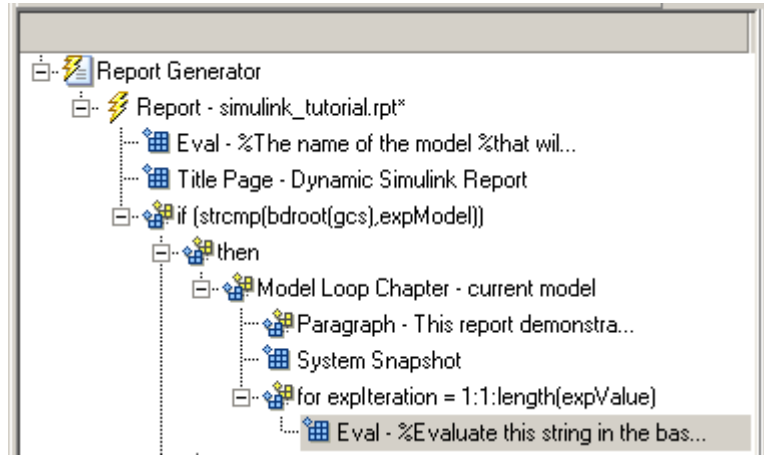


4 Save the report.

Getting the Gain Parameter Value from the expValue Array

On each iteration, get a value from the `expValue` array to use as the Gain parameter value. For this you need an Evaluate MATLAB Expression component.

1. In the Outline pane on the left, select the `for` component.
2. In the Options pane in the middle, under the `MATLAB` category, double-click `Evaluate MATLAB Expression`. In the Outline pane, the component name is shortened to `Eval`.



3. On the Properties pane on the right, do the following:
   - Clear the **Insert MATLAB expression in report** and **Display command window output in report** check boxes.
   - Enter the following text in the **Expression to evaluate in the base workspace** text box.

Tip You can copy and paste this text from the HTML documentation into the Report Explorer.

```
%Evaluate this string in the base workspace
```

```
if iscell(expValue)
   Iteration_Value=expValue{expIteration};
else
  Iteration_Value=...
     num2str(expValue(expIteration));
end
```

The `Iteration_Value` variable represents the designated array element.

- Clear the **Evaluate expression if there is an error** check box.

4 Save the report.

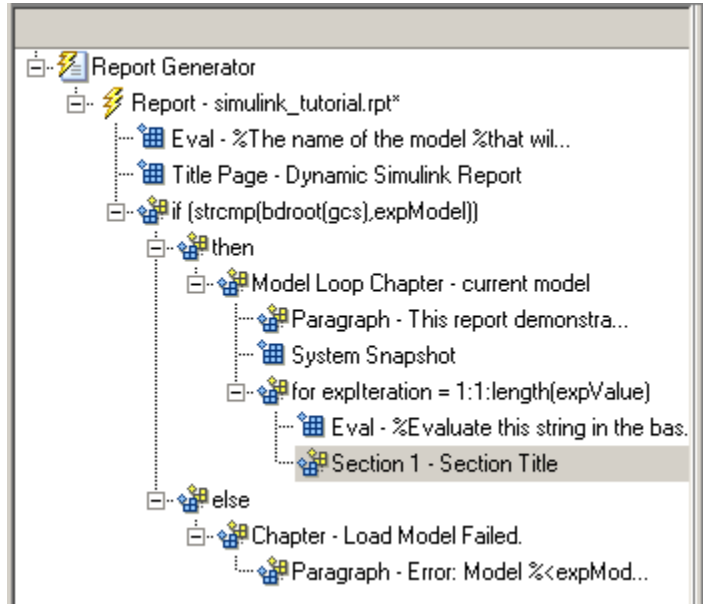
Creating a Section for Each Iteration

Create a separate section for each iteration of the loop that includes the data using the Chapter/Subsection component.

1 In the Outline pane on the left, select the `Eval` component.

2 In the Options pane in the middle, under the `Formatting` category, double-click the `Chapter/Subsection` component to add it as a sibling.

This component is automatically added as `Section 1` because it is inside a `Chapter` component (the Model Loop Chapter component).



3 In the Properties pane on the right, do the following:

- In the **Title** drop-down list, select `Custom`.
- In the text box, enter the following title:

  ```
  Processing the vdp model
  ```

This indicates that the section title comes from the first child component. Do not change any other properties.

4 Save the report.

Inserting the Gain Value into the Report

Insert the Gain value being used for each simulation.

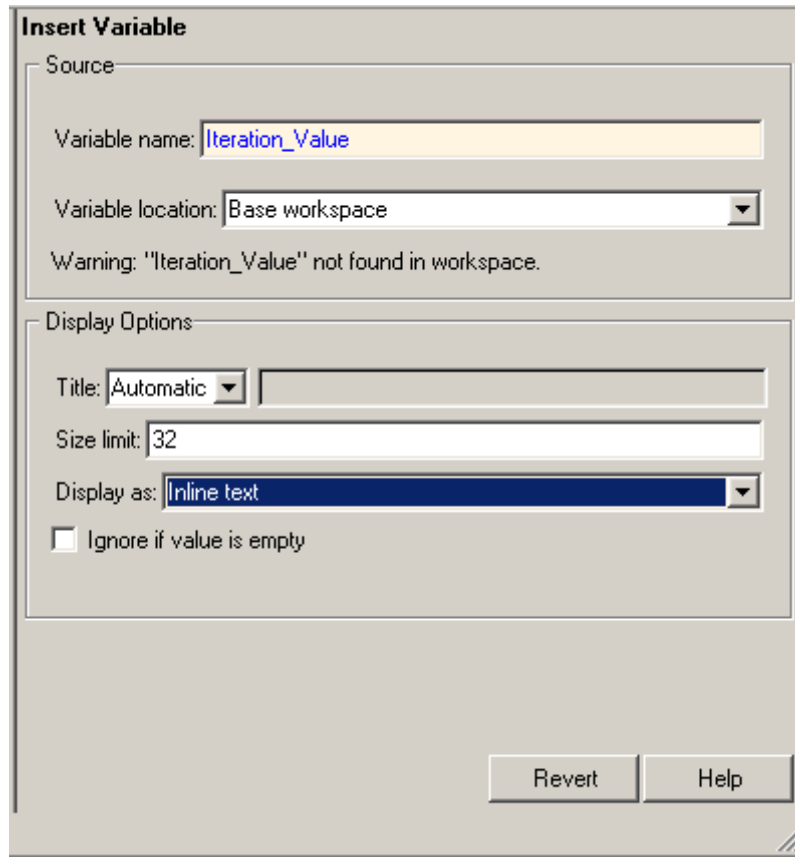
1 In the Outline pane on the left, select the `Section 1` component.

2 In the Options pane in the middle, under the `MATLAB` category, double-click `Insert Variable`.

3 In the Properties pane on the right, do the following:

- In the **Variable name** text box, enter `Iteration_Value`.
- In the **Display as** drop-down list, select `Inline text`.

The Outline pane on the left should look as follows:



4 Save the report.

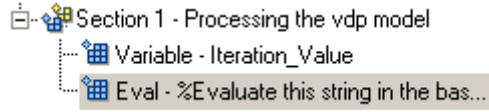
Setting the Gain Parameter

For each iteration, set the Gain parameter to the value you extracted from the `expValue` array.

1. In the Outline pane on the left, select the `Variable` component.

2. In the Options pane in the middle, under the `MATLAB` category, double-click `Evaluate MATLAB Expression`.

   This component is added as a sibling of the `Variable` component.



3. In the Properties pane on the right, clear the **Insert MATLAB expression in report** and **Display command window output in report** check boxes.

4. In the **Expression to evaluate in the base workspace** text box, replace the existing text with the following:

   ```
   set_param(expBlock,expParam,Iteration_Value);
   okSetValue=(1);
   ```

   The `set_param` command sets the value of the Gain parameter for the Mu block in the `vdp` model to the value of `Iteration_Value`.

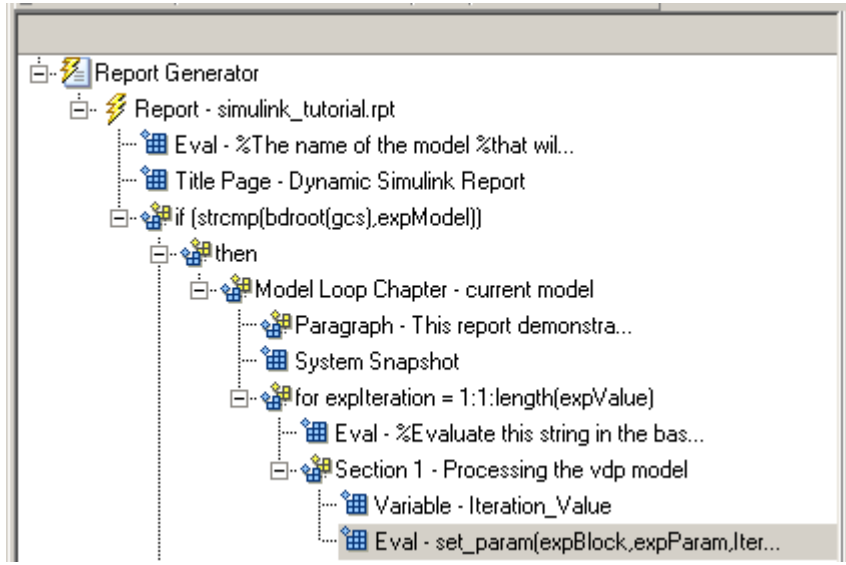
5. Make sure **Evaluate expression if there is an error** is selected, and enter the following text in the text box:

   ```
   okSetValue=logical(0);
   ```

   If the `set_param` command works, `okSetValue` is set to 1. If an error occurs, `okSetValue` is set to 0. The next component then reports the error and terminates processing.

6. Save the report.

   The Outline pane on the left looks as follows:

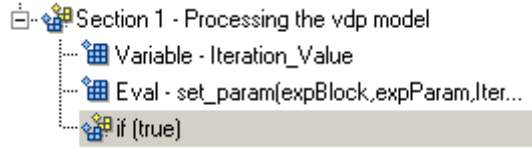


Checking okSetValue

Check the value of `okSetValue` using a Logical If component. If the value is 0, the simulation cannot proceed because the Gain parameter could not be set.

1. In the Outline pane on the left, select the `Eval` component for the `set_param` command.
2. In the Options pane in the middle, under the `Logical and Flow Control` category, double-click `Logical If`.

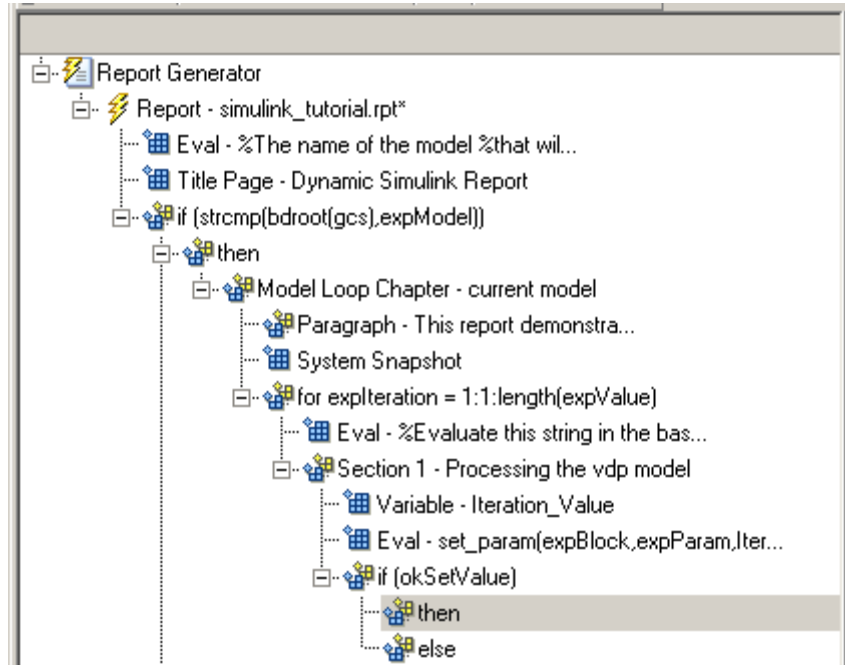
   The component is added as another sibling.



3. In the Properties pane on the right, in the **Test expression** text box, replace `true` with `okSetValue`.

`okSetValue` can be 1 (true) or 0 (false), so insert two components—`Logical Then` and `Logical Else`—to deal with those conditions:

1. In the Outline pane on the left, select the `if(okSetValue)` component.
2. To insert these two components in the correct order, first, in the Options pane in the middle, double-click the `Logical Else` component.
3. Select the `if(okSetValue)` component again.
4. Double-click the `Logical Then` component. The Outline pane on the left should look as follows:



5. In the Outline pane on the right, select the `else` component.
6. In the Options pane in the middle, double-click `Paragraph`.

   If `okSetValue` = 0, the Gain parameter value was not set, so the report should display an error.

7 In the Properties pane on the right, do the following:

- In the **Title Options** drop-down list, select `Custom title`.
- Enter `Error` in the text box next to the drop-down list.
- Enter the following in the **Paragraph Text** text box:

```
Could not set %<expBlock> "%<expParam>" to value
%<Iteration_Value>.
```

8 Save the report.

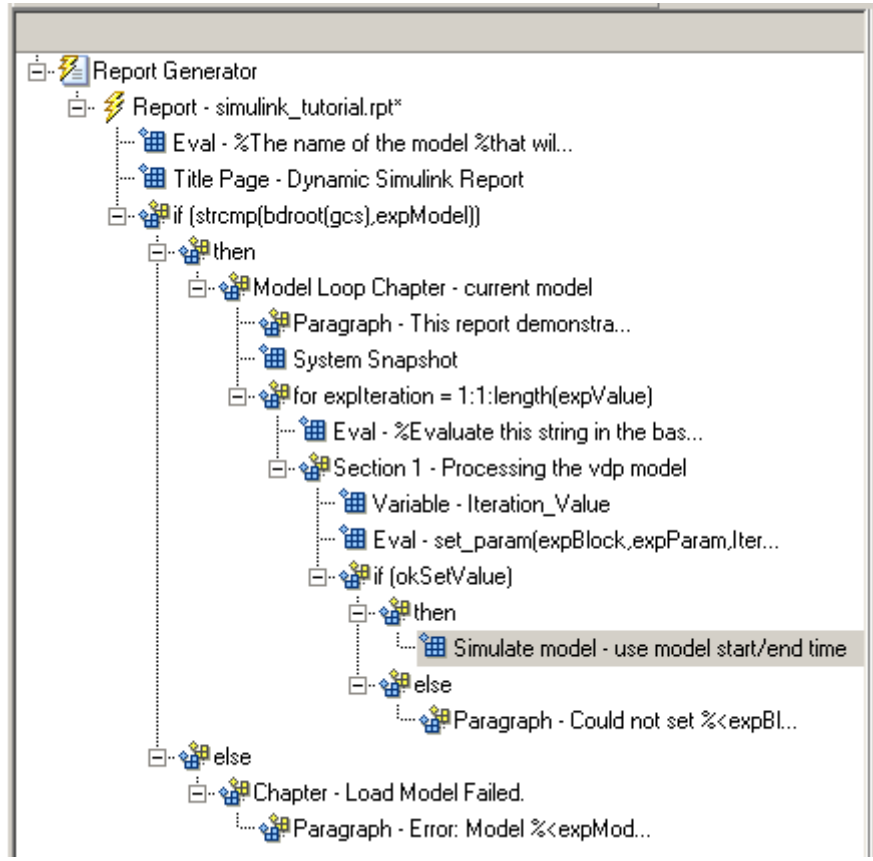
Simulating the Model, and Capturing the Scope and Data

Now that the model is open and the Gain parameter is set, use the Model Simulation component to simulate the vdp model.

1 In the Outline pane on the left, select the `then` component under the `if (okSetValue)` component.

2 In the Options pane, under the `Simulink` category, double-click `Model Simulation`.

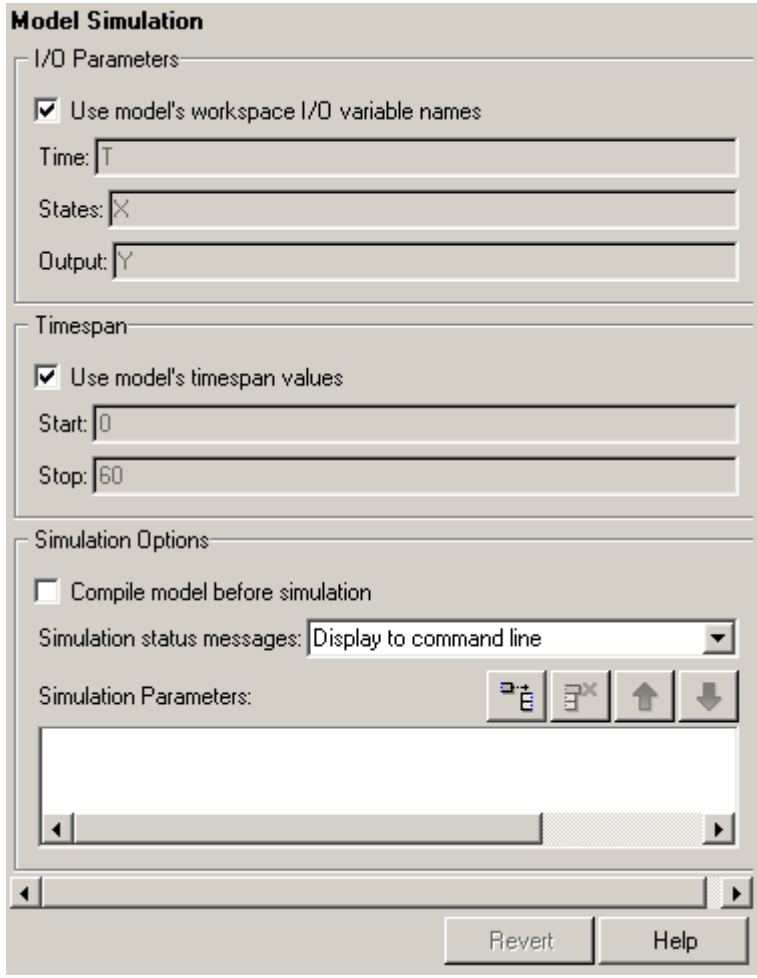
In the Outline pane on the left, this component is renamed to `Simulate model`.



3 In the Properties pane on the right, do the following:

- Clear the **Use model's workspace I/O variable names** check box.
- In the **Time** text box, enter `dynamicT`.
- In the **States** text box, enter `dynamicX`.
- In the **Output** text box, enter `dynamicY`.

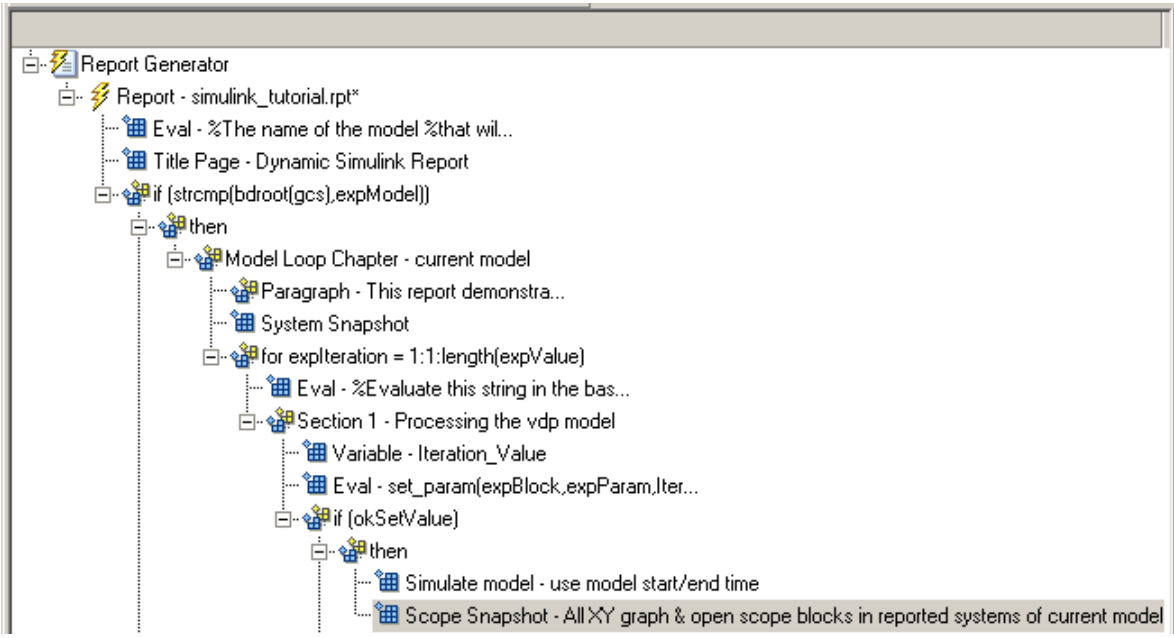
The Properties pane on the right should look as follows:



4 In the Outline pane on the left, select the `Simulate model` component.

5 In the Options pane in the middle, scroll down to the `Simulink Blocks` category and double-click `Scope Snapshot` to add it as a sibling to the `Simulink Model` component.

This component captures the scope for each iteration.



6. On the Properties pane on the right, do the following:
   - In the **Paper orientation** drop-down list, select `Portrait`.
   - For the **Image size**, enter `[5 4]`.
   - In the **Scaling** drop-down list, select `Zoom`.
   - Enter `75` for the % value.
7. Save the report.

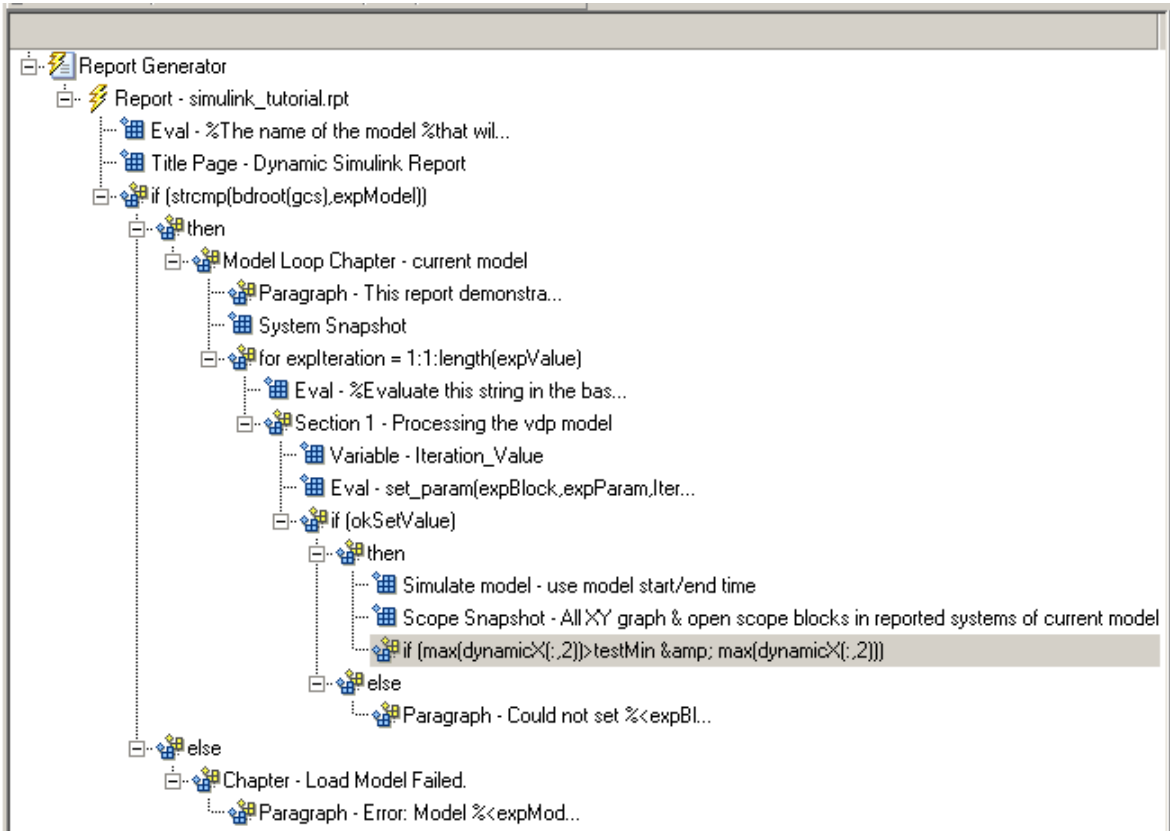
   To test to see if the signal data falls within a specified range, you need another Logical If component.
8. In the Outline pane on the left, select the `Scope Snapshot` component.
9. In the Options pane in the middle, scroll up to the `Logical and Flow Control` category.
10. Double-click the `Logical If` component.

11 To test the signal data, replace `true` in the **Test expression** text box with the following in the Properties pane on the right:

```
max(dynamicX(:,2))>testMin & max(dynamicX(:,2))
```

12 Save the report.

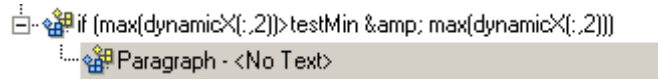
The Outline pane should look as follows:



13 In the Outline pane on the left, select the `if` component you just added.

If this condition is true, the signal data falls within the desired range, so add a `Paragraph` component to print this in the report.

14 In the Options pane in the middle, under the `Formatting` category, double-click `Paragraph` so that it becomes a child of the `if` component.



15 In the Properties pane on the right, do the following:

- From the **Title Options** drop-down list, select `Custom title`.
- Type `Success` in the text box.
- Enter the following text in the **Paragraph text** text box.

Tip You can copy and paste this text from the HTML documentation into the Report Explorer.

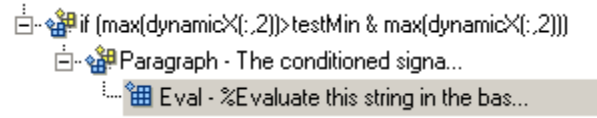
```
The conditioned signal has a maximum value
of %<max(dynamicX(:,2))>, which lies in the
desired range of greater than %<testMin> and
less than %<testMax>.
```

To save the success values to insert into a table at the end of the iterations, use an `Evaluate MATLAB Expression` component.

16 In the Outline pane on the left, select the `Paragraph` component.

17 In the Options pane in the middle, under the `MATLAB` category, double-click `Evaluate MATLAB Expression`.

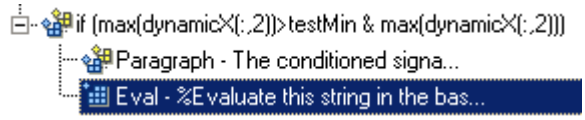
An unintended result occurs: the new component is a child of the `Paragraph` component.



18 To make the new component a *sibling* of the `Paragraph` component, in the Outline pane on the left, select the `Eval` component.

19 Click the left arrow on the toolbar.

The `Eval` component becomes a sibling of the `Paragraph` component.



20 In the Properties pane on the right, for the `Eval` component, do the following:

- Clear the **Insert MATLAB expression in report** and **Display command window output in report** check boxes.
- In the **Expression to evaluate in the base workspace** text box, enter the following to save the desired signal values in the `expOkValues` array:

```
expOkValues=[expOkValues;...
     {Iteration_Value,max(dynamicX(:,2))}];
```

- Make sure the **Evaluate this expression if there is an error** check box is selected and insert the following text in the text box:

```
disp(['Error during eval: ', lasterr])
```

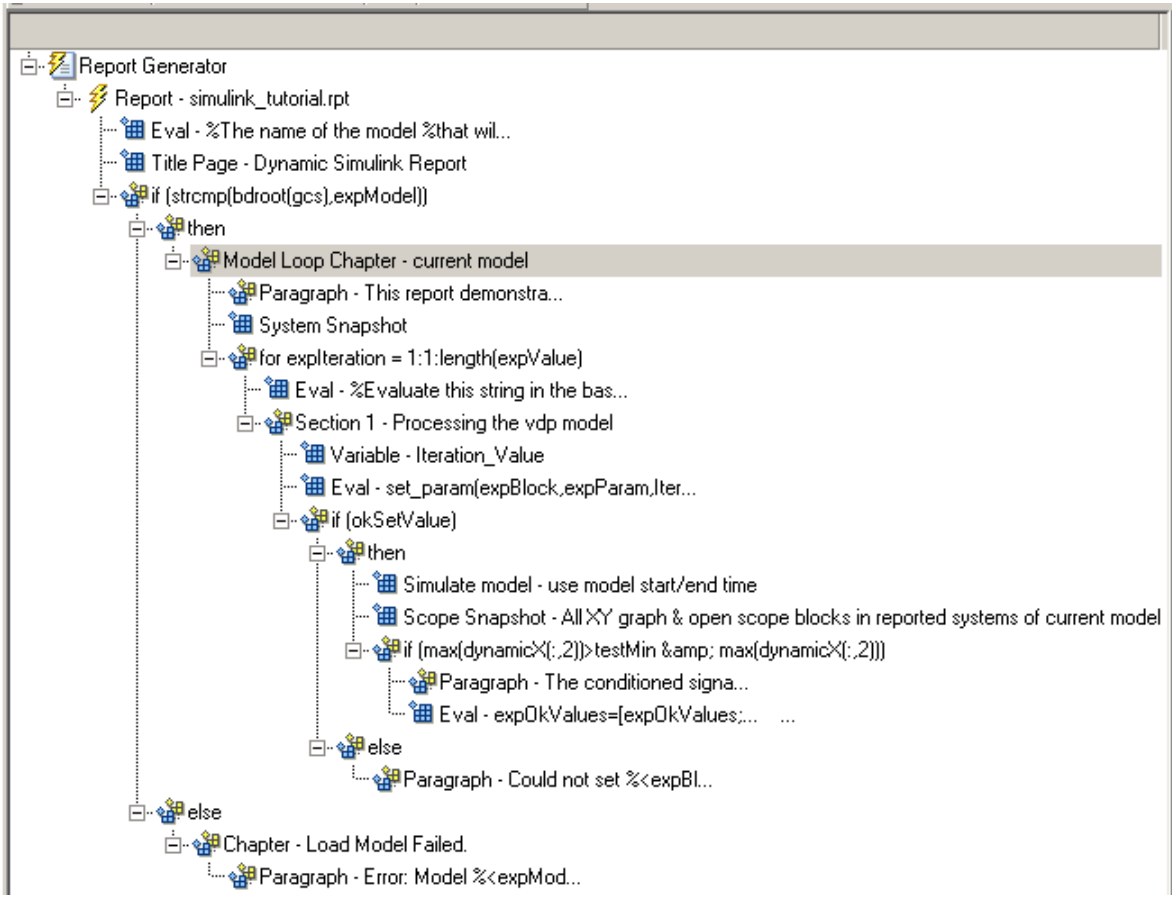
21 Save the report.

The only task left is creating the post-test analysis section.

Creating the Post-Test Analysis Section

Now that you have collected all the desired values, create a table and insert the table into your report.

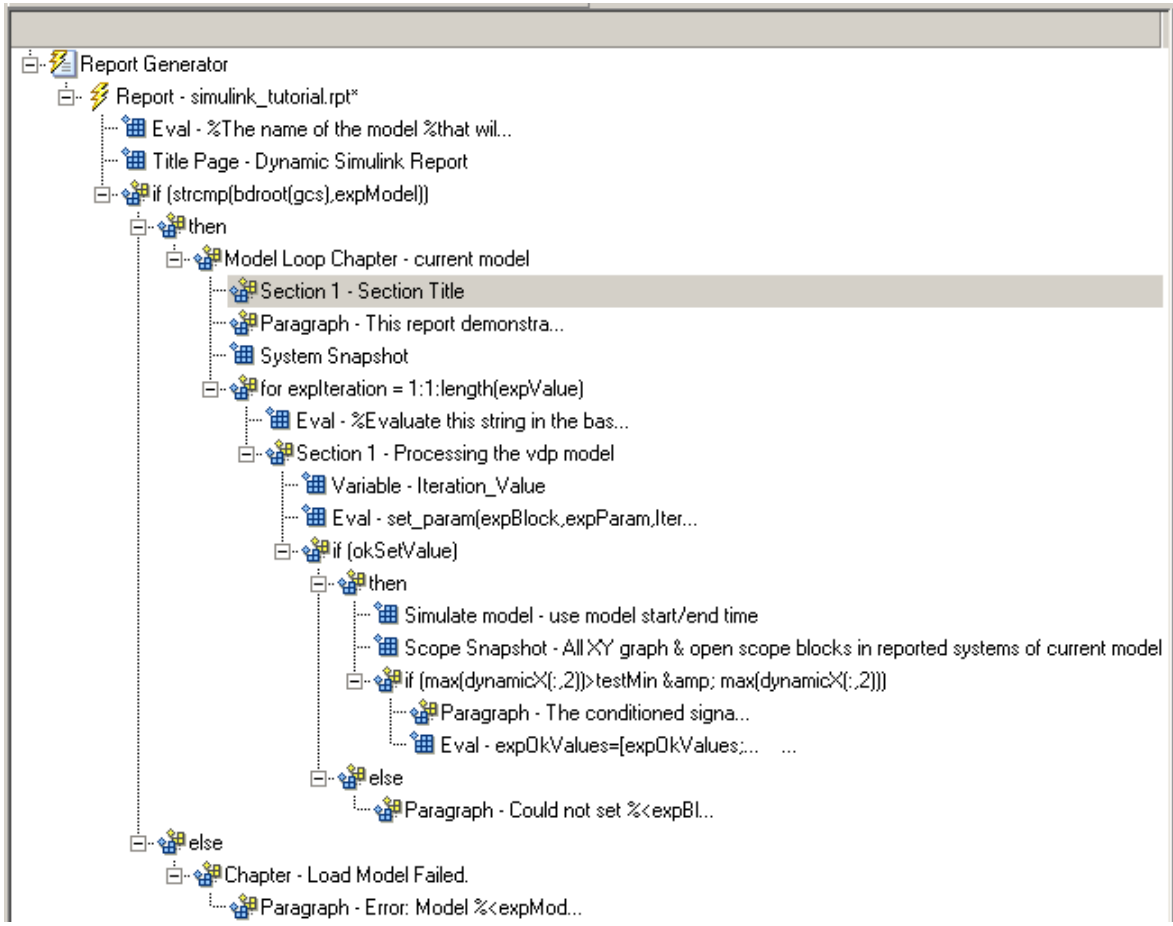
1 In the Outline pane on the left, select the `Model Loop Chapter` component.



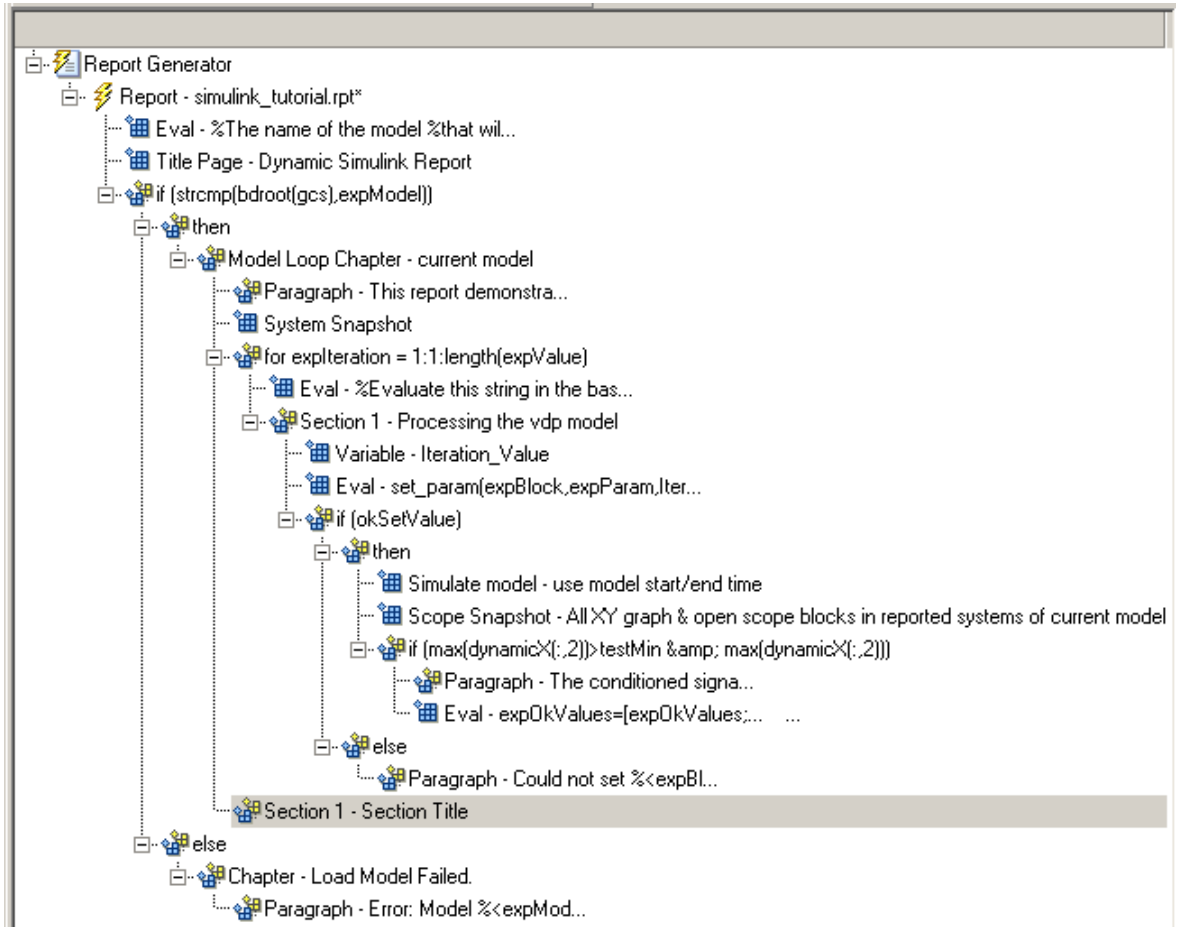
The post-test analysis section should go at the end of this chapter.

2 In the Options pane in the middle, under the `Formatting` category, double-click `Chapter/Subsection`.

The new section appears at the beginning of the chapter.



3 Click the **down** arrow three times so `Section 1` moves to the end of the `Model Loop Chapter` component.



4 In the Properties pane on the right, make sure the **Title** drop-down list is set to `Custom` and enter `Post-Test Analysis` in the text box.

5 In the Outline pane on the left, select the new `Section 1` component.

6 In the Options pane in the middle, under the `Formatting` category, double-click `Paragraph`.

   Do not change the properties.

To see if there are any desired signal values, check the array `expOkValues` with a Logical If component; if `expOkValues` is empty, there were no signal values in the desired range, so report that as well.

7 In the Outline pane on the left, select the `Paragraph` component and add a `Logical If` child component.

8 In the Properties pane on the right, enter the expression to evaluate in the **Test expression** text box:

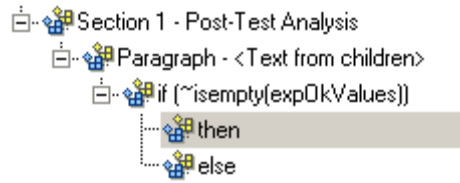
```
~isempty(expOkValues)
```

This expression evaluates to 0 (false) if `expOkValues` is empty; otherwise, it evaluates to 1 (true).

9 In the Outline pane on the left, select the `if(~isempty(expOkValue))` component and add the `Logical Else` component as a child.

10 Select the `if(~isempty(expOkValue))` component again and add the `Logical Then` component as a child.

The two components should be siblings in the Outline pane on the left.



11 Save the report.

Now, insert report components to handle the case where `expOkValues` is empty, that is, no signal values fell within the designated range.

12 In the Outline pane on the left, select the `else` component.

13 In the Options pane in the middle, double-click the `Text` component to add it as a child to the `else` component.

14 In the Properties pane on the right, in the **Text to include in report** text box, enter the following:

```
None of the selected iteration values had
a maximum signal value between %<testMin> and %<testMax>.
```

Now handle the case where `expOkValues` is not empty and you want to insert a table of the acceptable signal values.

15 In the Outline pane on the left, select the `then` component.

16 Add a `Text` component as a child to the `then` component.

17 In the Properties pane on the right, in the **Text to include in report** text box, enter the following text.

Tip You can copy and paste this text from the HTML documentation into the Report Explorer.

```
%<size(expOkValues, 1)> values for %<expBlock> were
found that resulted in a maximum signal value greater
than %<testMin> but less than %<testMax>. The following
table shows those values and their resulting signal maximum.
```

18 Save the report.

19 In the Properties pane on the right, select the `Text` component in the Outline pane on the left below the `then` component of the `if(~isempty(expOkValues)` component.

To create an array for use when formatting the table, use the `Evaluate MATLAB Expression` component.

20 In the Options pane in the middle, double-click `Evaluate MATLAB Expression`.

21 In the Properties pane on the right, do the following:

- Clear the **Insert MATLAB expression in report** and **Display command window output in report** check boxes.
- Insert the following in the **Expression to evaluate in the base workspace** text box:

```
expOkValues=[{'Mu Value','Signal Maximum'} expOkValues];
```

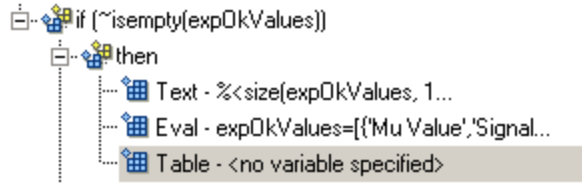
This adds the strings `Mu Value` and `Signal Maximum` to the front of the `expOkValues` cell array. They are used as the table headers in the next component of the report.

- Make sure the **Evaluate this expression if there is an error** check box is selected and insert the following text in the text box:

```
disp(['Error during eval: ', lasterr])
```

22 In the Outline pane on the left, select the `Eval` component.

23 In the Options pane in the middle, under the `Formatting` category, double-click the `Table` component so it becomes a sibling of the `Text` and `Eval` components.

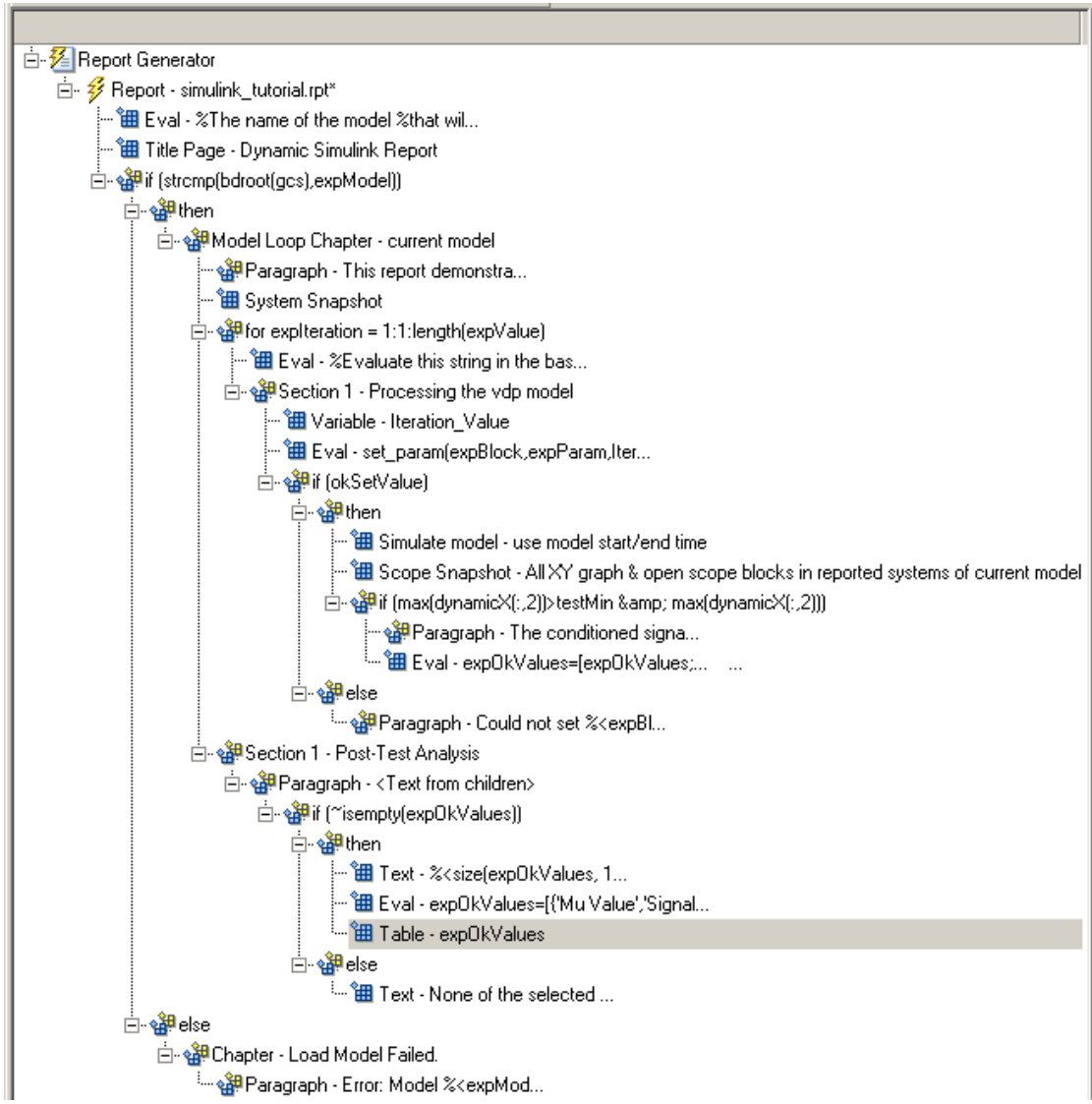


24 In the Properties pane on the right, do the following:

- In the **Workspace variable name** text box, enter `expOkValues`, whose contents are used to construct the table.
- In the **Table title** text box, enter `Valid Iteration Values`.

25 Save the report.

The Outline pane on the left should look as follows:

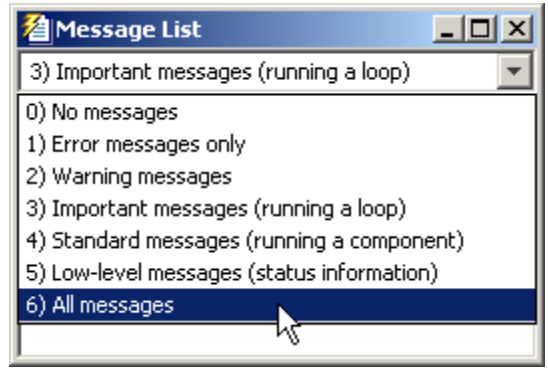

Report Generator
Report - simulink_tutorial.rpt*
Eval - %The name of the model %that wil...
Title Page - Dynamic Simulink Report
if (strcmp(bdroot(gcs),expModel))
then
Model Loop Chapter - current model
Paragraph - This report demonstra...
System Snapshot
for expIteration = 1:1:length(expValue)
Eval - %Evaluate this string in the bas...
Section 1 - Processing the vdp model
Variable - Iteration_Value
Eval - set_param(expBlock,expParam,Iter...
if (okSetValue)
then
Simulate model - use model start/end time
Scope Snapshot - All XY graph & open scope blocks in reported systems of current model
if (max(dynamicX(:,2))>testMin & max(dynamicX(:,2)))
Paragraph - The conditioned signa...
Eval - expOkValues=[expOkValues;... ...
else
Paragraph - Could not set %<expBl...
Section 1 - Post-Test Analysis
Paragraph - <Text from children>
if (~isempty(expOkValues))
then
Text - %<size(expOkValues, 1...
Eval - expOkValues=[{'Mu Value','Signal...
Table - expOkValues
else
Text - None of the selected ...
else
Chapter - Load Model Failed.
Paragraph - Error: Model %<expMod...

Generating the Report

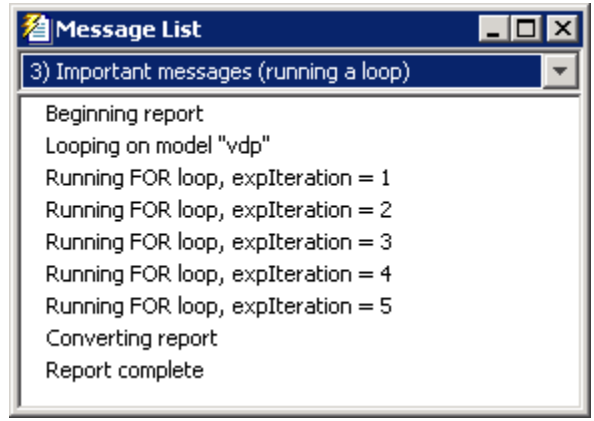
Now the report contains all the components it needs.

Click the **Report** icon on the toolbar to generate the report. The following occurs.

- A Message List window appears, displaying informational and error messages as Report Generator is processing the report. While the report is being generated, you can specify the level of detail you would like the Message List window to display. Options range from 0 (least detail) to 6 (most detail). Click the drop-down list located under the title bar of the Message List window to choose an option, as shown in the figure below.

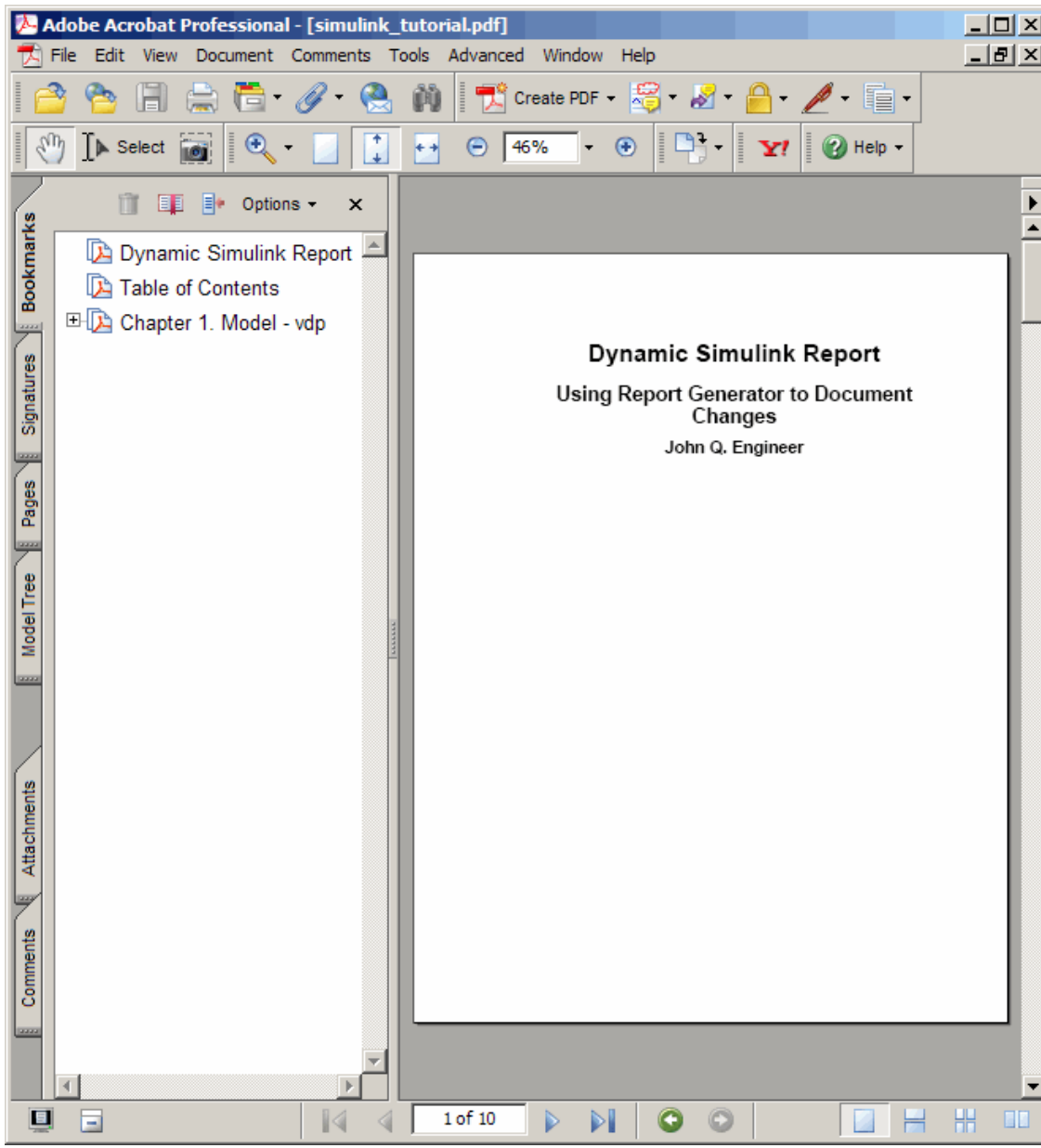


Message level 3 (Important messages) is used for the remainder of this example.



- The vdp model appears, and you can see each time it is simulated.
- The scope window appears. The scope graph changes each time the parameter value changes.
- If you watch the Outline pane on the left of your Report Explorer window, each component of the report is highlighted as it executes.

When the report is complete, Adobe Reader opens your report in PDF format. If you specified another format for your output, the appropriate application opens; for example, a browser window opens if you specified HTML as the output format.


Adobe Acrobat Professional - [simulink_tutorial.pdf]
File Edit View Document Comments Tools Advanced Window Help
Create PDF
Select
46%
Help
Options
Bookmarks
Signatures
Pages
Model Tree
Attachments
Comments
Dynamic Simulink Report
Table of Contents
Chapter 1. Model - vdp
Dynamic Simulink Report
Using Report Generator to Document Changes
John Q. Engineer
1 of 10

4

Creating a Web View of a Simulink Model

This chapter covers the following topics, which apply to Simulink users.

What Is a Web View? (p. 4-2)	Overview of Web views and their key features
Exporting a Model to a Web View (p. 4-3)	An example of how to set Web view export options and export a model to a Web view
Navigating the Web View (p. 4-8)	Describes how to navigate layers in a Web view

What Is a Web View?

A Web view is an interactive rendition of a Simulink model or Stateflow model that you can view in a Web browser. You can use Web views to hierarchically navigate to specific subsystems and see properties of blocks, signals and scopes. You can also zoom in and out, pan, and fit your model to your view using the same keyboard shortcuts that you use in Simulink and Stateflow editors.

You need Report Generator to create a Web view. Using a Web view in a Web browser requires SVG support in the browser, implemented natively or through a plug-in, but it does not require MathWorks products to be installed. Because they require only Web browsers with widely available SVG support, Web views are useful for presenting models to audiences and for sharing models with colleagues who may not have MathWorks products installed.

For more details about Web views, see Chapter 6, “Exporting Simulink Models to Web Views”.

Exporting a Model to a Web View

This tutorial takes you through the steps required to export a Simulink model to a Web view. If you have a detailed Simulink model of your own, feel free to use it for this example; otherwise, use the example provided in this chapter.

There are several ways to access Report Generator to create a Web view of a Simulink model. They use similar steps, but this tutorial explains how to create a Web view from the Simulink model window.

The Simulink model used in this tutorial is `sldemo_auto_climatecontrol`, which uses Simulink and Stateflow to simulate the working of an automatic climate control system in a car.

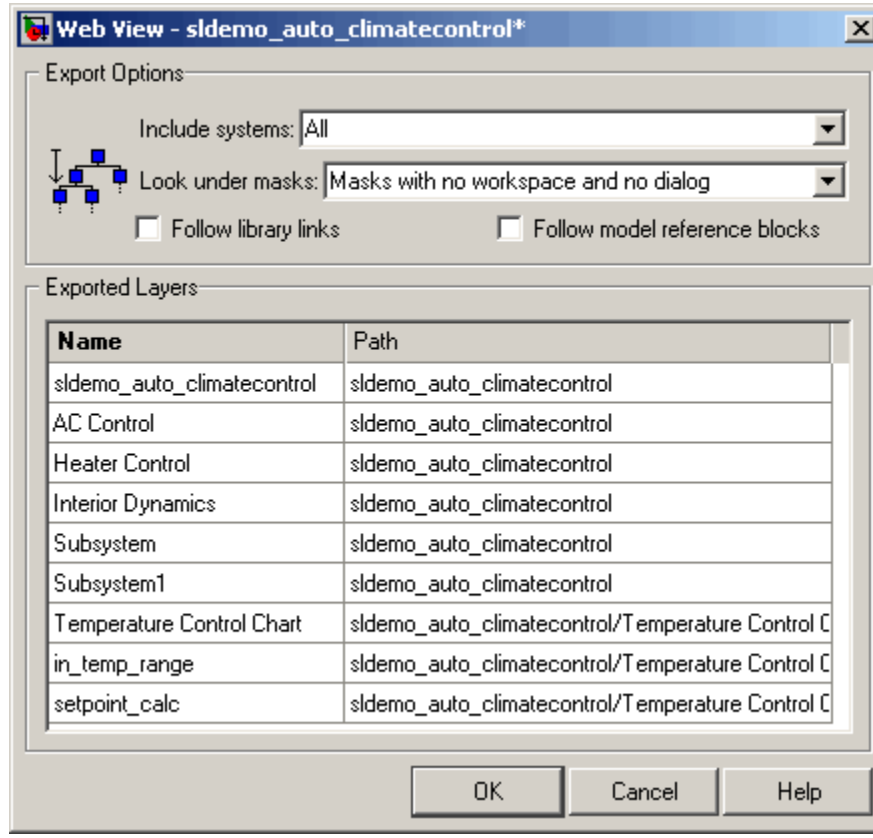
Note You do not need to know Simulink to follow this tutorial.

Specifying Web View Export Options

Before creating the Web view, you must specify export options.

1. Start MATLAB.
2. At the command prompt, enter `sldemo_auto_climatecontrol` to open the Simulink model.
3. Select **File > Export to Web**.

   The Web view window appears with data about the current model.



4. In the **Include systems** drop-down list, select `All` to export all systems in this model.

5. In the **Look under masks** drop-down list, select `Masks with no workspace and no dialog` to export only graphically masked blocks.

6. Leave the **Follow library links** and **Follow model reference blocks** check boxes cleared.

The lower part of the Web View window is the **Exported Layers** area. This area lists the names of the systems to be exported to the Web view, along with their Simulink path names. Because you selected `All` in the **Include**

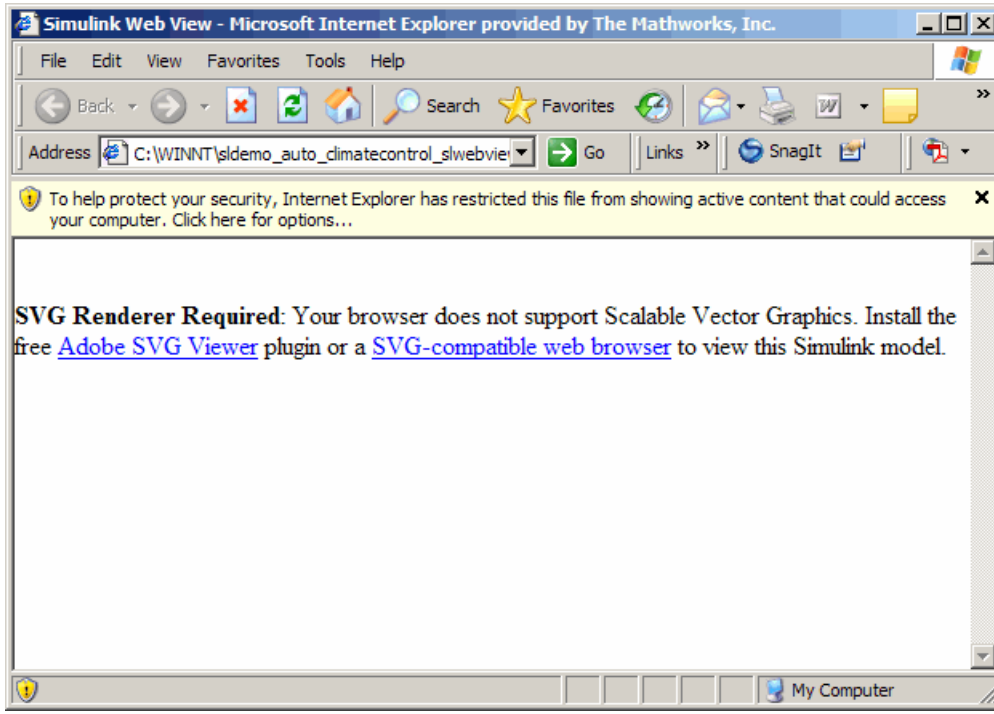
systems drop-down list, all layers in the `sldemo_auto_climatecontrol` model are listed.

Exporting the Model

Now that the export options set, you can export the model to a Web view.

1 Click **OK** in the Web View window.

After a few seconds, a browser window opens.

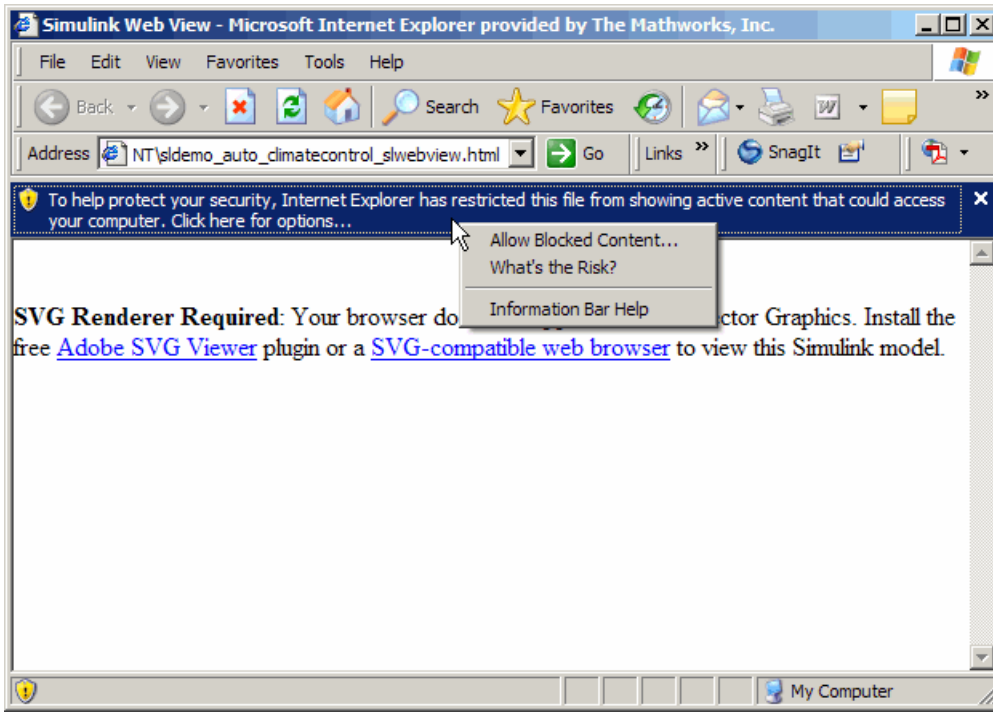


Note Scalable Vector Graphics (SVG) is required for Web views. The screen shots in this tutorial use Internet Explorer with the SVG plug-in installed. Mozilla Firefox has native support for SVG. For details on SVG, see “Browser Requirements for Web Views” on page 1-8.

2 Hover over the error message at the top of the screen.

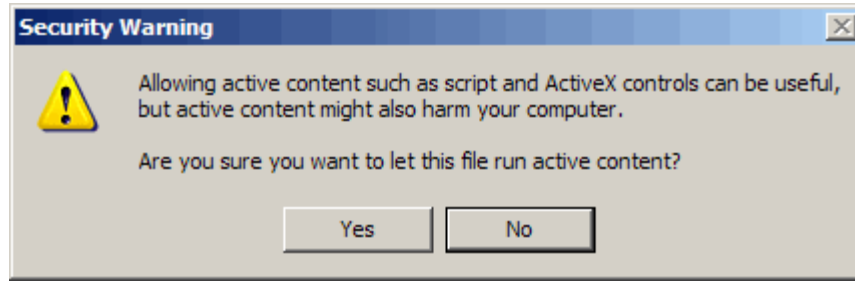
The error message background changes color.

3 Left-click to display a context menu.



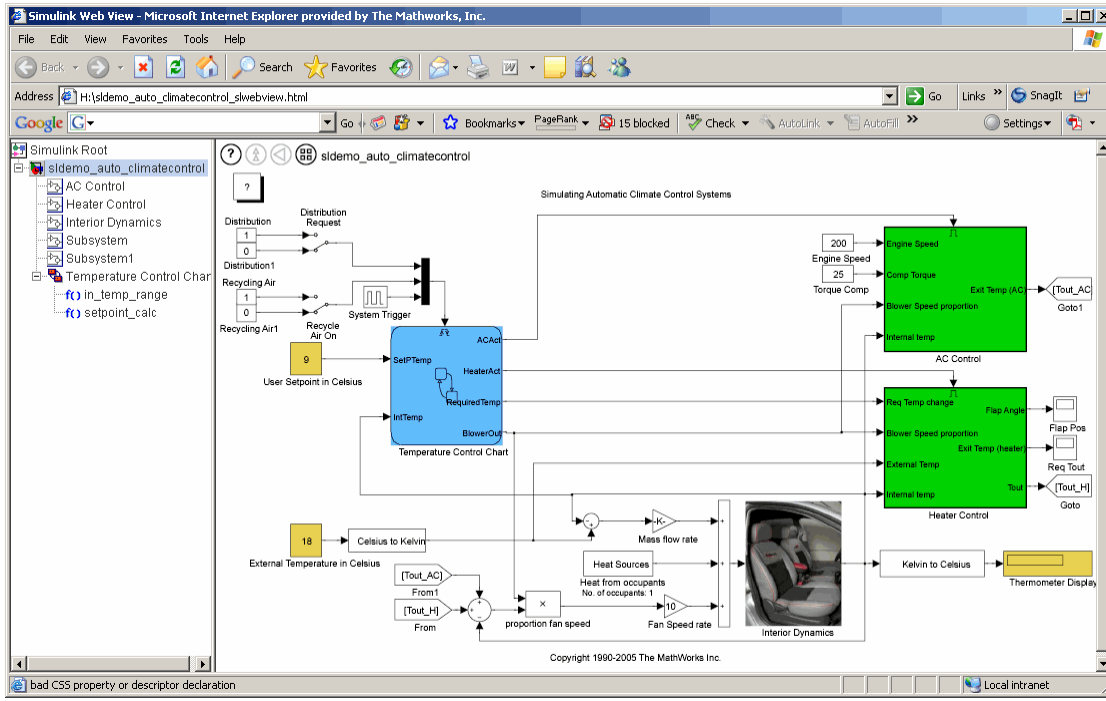
4 Select **Allow Blocked Content** from the context menu.

The following security warning appears.



5 Click **Yes**.

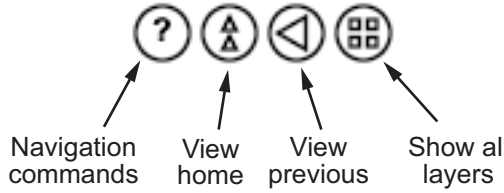
The Web view of the Simulink model appears in the browser window.



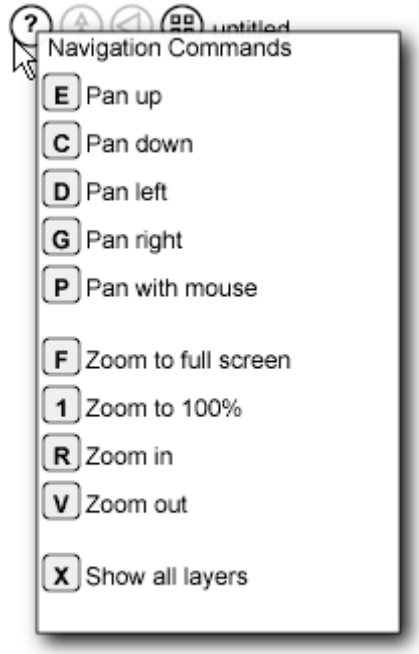
Navigating the Web View

You can navigate the Web view of the model in ways that are similar to how you navigate a model in Simulink. The following steps let you experiment with the types of navigation that are available:

1. Notice the four navigation icons in the upper-left corner of the browser window.

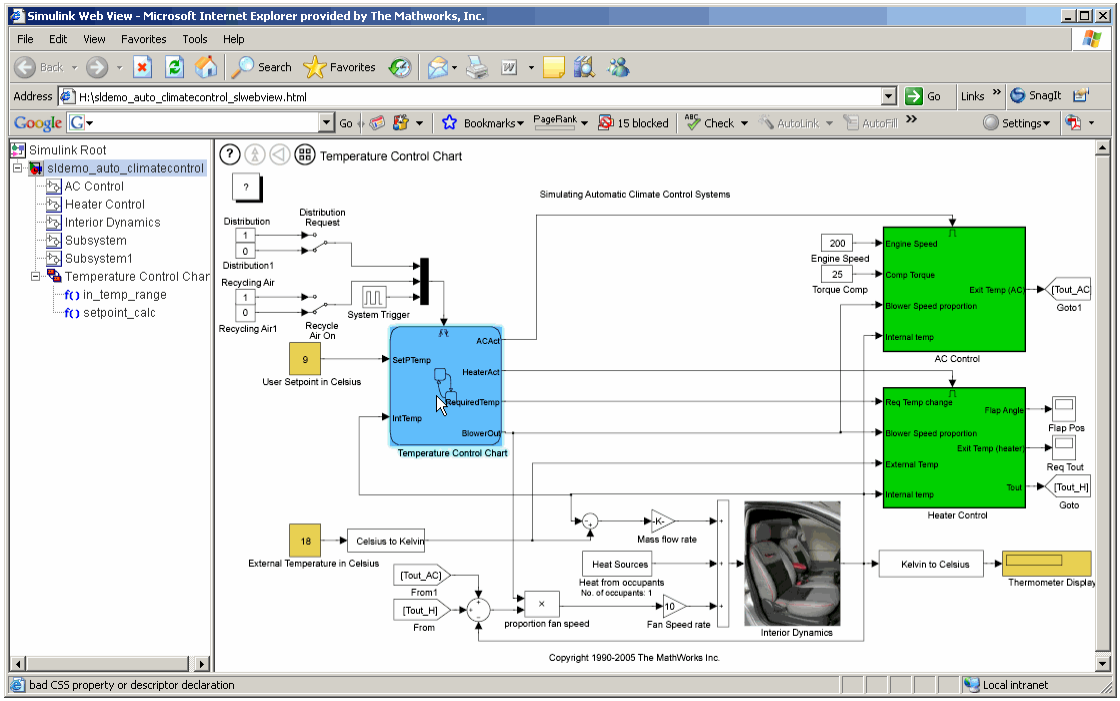


2. Hover over the Navigation Commands icon  to see a list of keyboard keys that allow you to move around the current window.

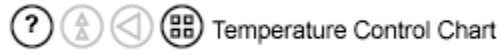


3 Press some of the keyboard keys listed here to see how the model in the Web view reacts.

4 Hover over the Temperature Control Chart diagram.



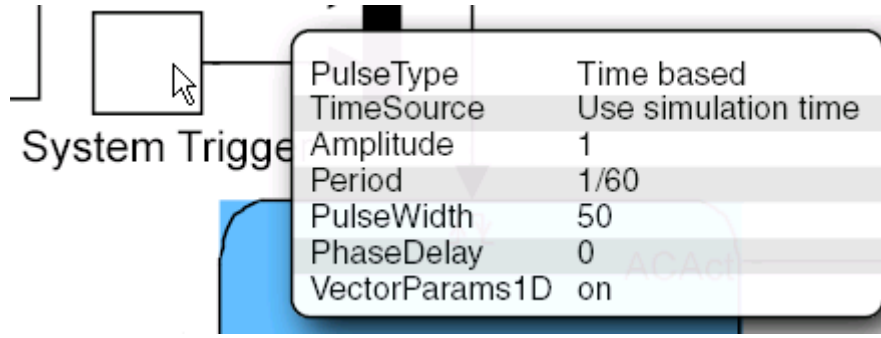
The name next to the navigation icons becomes `Temperature Control Chart`.



5 Hover over the System Trigger block.

The name next to the navigation icons changes to `System Trigger`.

In addition, the block parameter values for the System Trigger block appear.

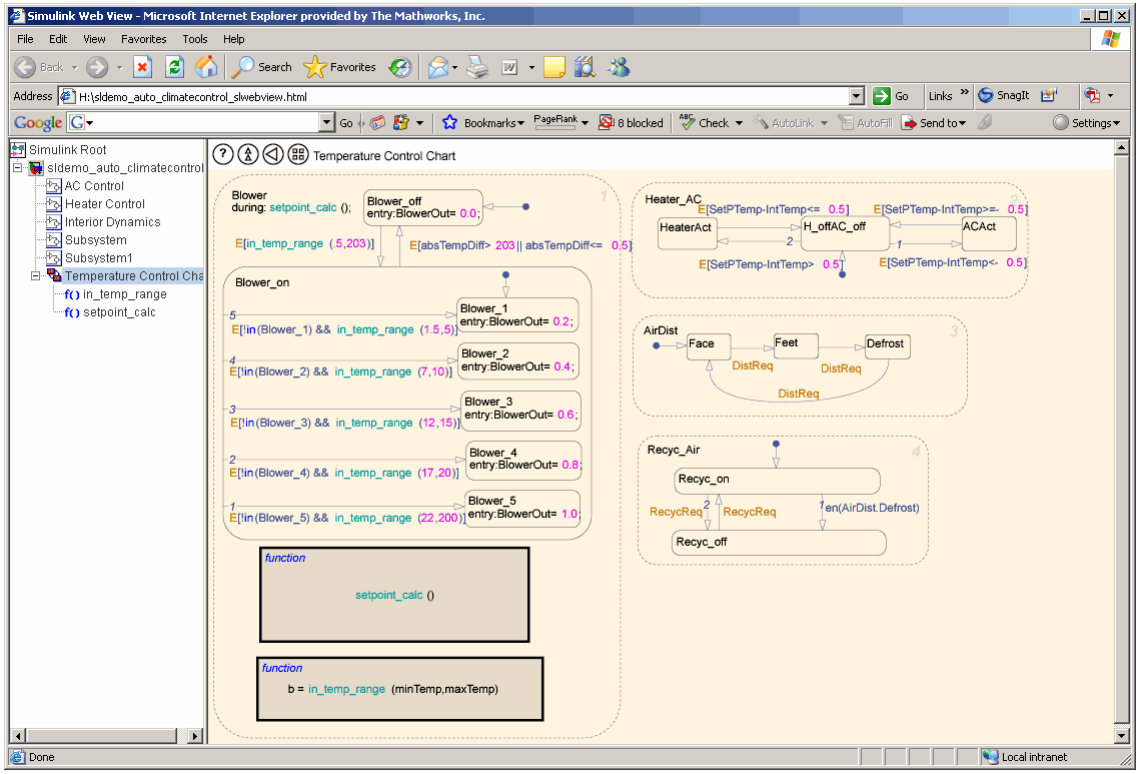


6 Hover over other blocks and subsystems in the model and see what results you get.

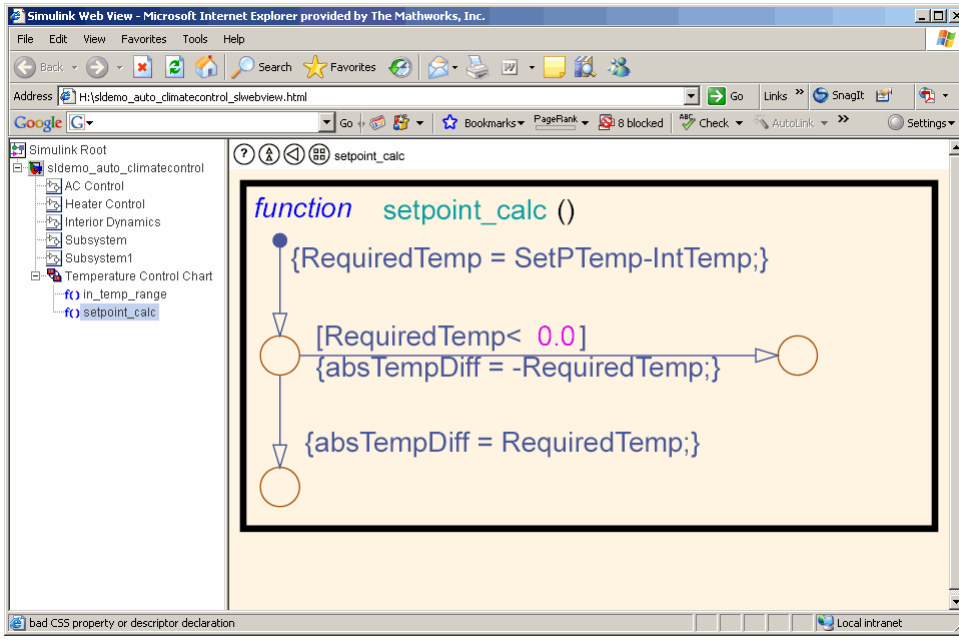
7 Click the Temperature Control Chart.

The browser displays the details of this Stateflow diagram and the middle two navigation icons are now enabled:

- The **View Home** icon takes you up to the top level of the model.
- The **View Previous** icon takes you up a level in the model.

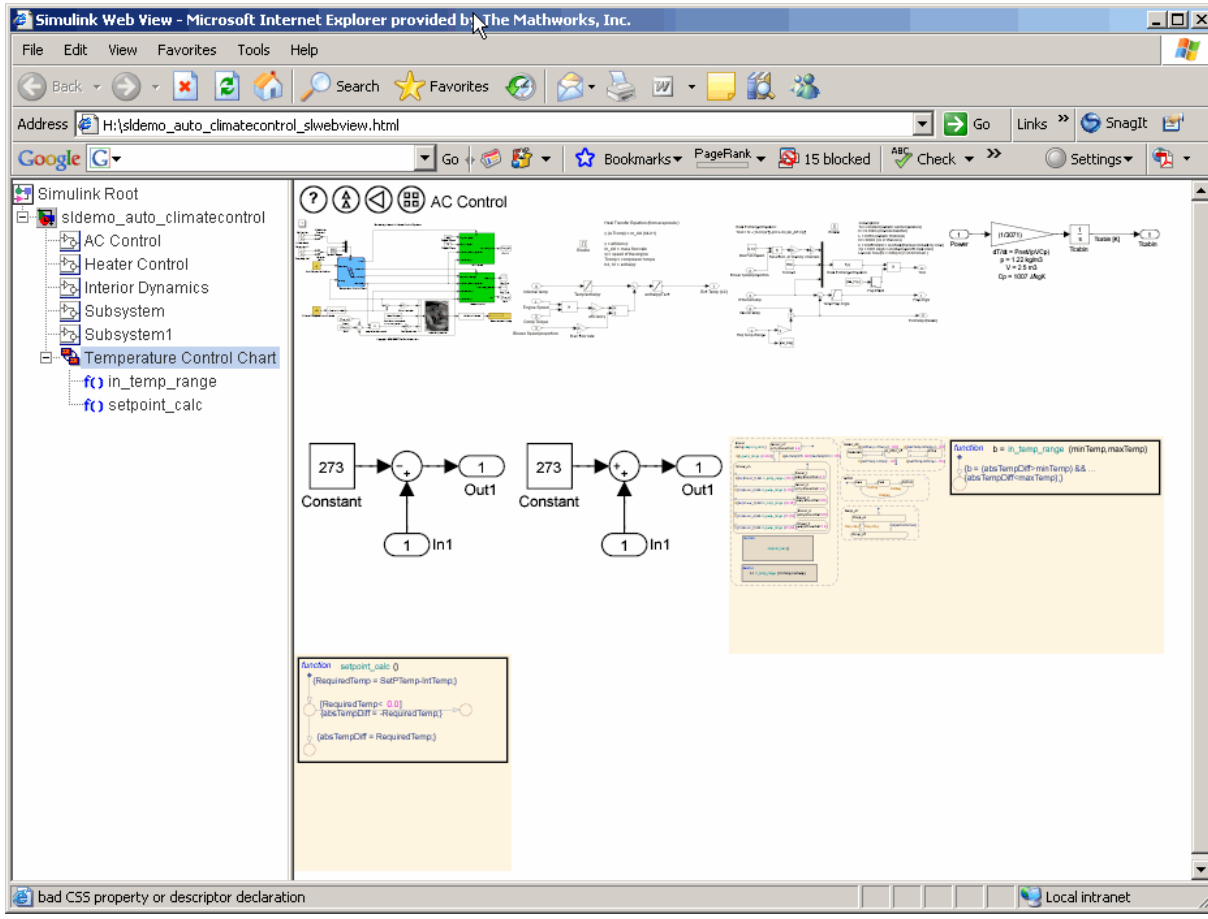


8 Click the `setpoint_calc` box in the Stateflow diagram to see the details of that block.



9 Click the **View home** icon to return to the model's top layer.

10 Click the **Show all layers** icon to display an overall view of the model.



This high-level view of the model shows all the layers of the model. In this figure, the arrow is hovering over the AC Control layer.

Try opening other Simulink models and exporting them to Web views using Report Generator to experiment with the results.

5

Generating Reports

This chapter covers the following topics, which apply to both MATLAB and Simulink users.

Setting Report Output Options (p. 5-2)	Describes report generation options
Converting XML Documents (p. 5-10)	Demonstrates how to convert XML source files with the Report Explorer
Creating Log Files (p. 5-14)	Shows how to create log files for report generation
Generating M-code from Report Templates (p. 5-15)	Describes how to generate M-code versions of report templates
Working with Legacy Report Templates (p. 5-16)	Describes how to work with report templates created in a previous version of Report Generator
Troubleshooting (p. 5-18)	Dealing with problems when using Report Generator

Setting Report Output Options

Report Generator enables you to control most aspects of report output. For details, see the following sections:

Setting the Report File Directory

Choose the appropriate directory for the report file according to the table below. Make sure you have write privileges for that directory.

Directory	Option
The same directory as the report template	`Same as report template`
The current working directory	`Present working directory`
Temporary directory	`Temporary directory`
Another directory	`Custom`. Specify a directory in the **Custom** text box (you can use the **Browse** button (...) to select from a list of directories).

You can use the `%<VariableName>` notation to specify a directory in the **Custom** text box. For more details about this notation, see "%<VariableName> Notation" on page 11-269 on the Text component reference page.

Setting the Report File Name

Choose the appropriate file name for the report file according to the table below.

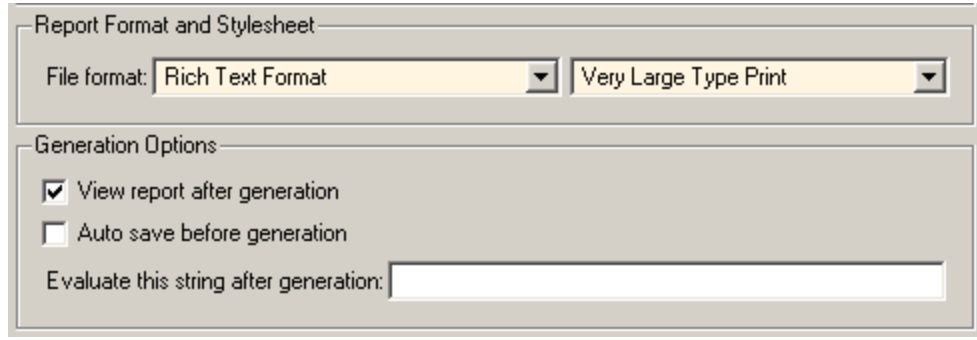
Images are placed in a directory with the same name as the report file. For example, `testreport.html` images are placed in a directory named `testreport_files`.

File Name	Option
Your report is to have the same file name as the report template.	`Same as report template` (default)
Your report is to have a file name that is different from the report template name.	`Custom`. Enter the name of the report in the text box.

You can use the `%<VariableName>` notation to specify a file name in the **Custom** text box. For more details about this notation, see "%<VariableName> Notation" on page 11-269 on the Text component reference page.

Setting the Report Format

You can choose the report output format in the **File format** text box. You should choose the output format according to how you want to view the report. For example, if you want to use Microsoft Word, you should choose `Word document` format or `Rich Text Format`.



Each output format has a default stylesheet. Specify the stylesheet in the text box next to the **File format** text box.

The table below shows which output format to use with different viewers or software as well as which stylesheets are available for use with each output format.

Viewer	Format	Purpose	Stylesheet Options
Adobe Acrobat Reader	Acrobat (PDF)	Produces a PDF that can be viewed with Acrobat Reader	PDF (see "PDF Stylesheets" on page 5-5)
Web browser	`Web (HTML)` (default)	Use for publishing on the World Wide Web	Web (see "Web Stylesheets" on page 5-6)
Word processor	`Rich Text Format (RTF)` or `Word Document`	Compatible with most word-processing packages, including Microsoft Word	Print (see "RTF (DSSSL Print) and Word Stylesheets" on page 5-7)
DocBook	DocBook (XML)	Produces a report in DocBook format	N/A

Note You can create and use customized styles, in addition to the built-in stylesheets. See Chapter 9, "Creating and Using Customized Styles".

PDF Stylesheets

PDF Stylesheet	Description
`Standard Print`	Displays title page, table of contents, list of titles.
`Simple Print`	Suppresses title page, table of contents, list of titles.
`Compact Simple Print`	Minimizes page count, suppresses title, table of contents, list of titles.
`Large Type Print`	Uses 12 point font (slightly larger than Standard Print).
`Very Large Type Print`	Uses 24 point font and landscape paper orientation.
`Compact Print`	Minimizes white space to reduce page count.
`Unnumbered Chapters & Sections`	Chapters and sections are not numbered.
`Numbered Chapters & Sections`	Chapters and sections are both numbered.
`Paginated Sections`	Sections are printed with page breaks.
`Custom Header`	Lets you specify custom headers and footers.
`Custom Titlepage`	Lets you specify custom title page content and presentation.
`Logo stylesheet for PDF`	Lets you specify a logo, such as your company logo, in the header.
`Verbose Print`	Lets you specify advanced print options.

Web Stylesheets

Web Stylesheet	Description
`Default HTML stylesheet`	HTML on a single page.
`Simulink book HTML stylesheet`	HTML on multiple pages; suppresses chapter headings and table of contents.
`Truth Table HTML stylesheet`	HTML on multiple pages; suppresses chapter headings and table of contents.
`Multi-page Web`	HTML, with each chapter on a separate page.
`Single-page Web`	HTML on a single page.
`Single-page Unnumbered Chapters & Sections`	HTML on a single page; chapters and sections are not numbered.
`Single-page Numbered Chapters & Sections`	HTML on a single page; chapters and sections are numbered.
`Single-page Simple`	HTML on a single page; suppresses title page and table of contents.
`Multi-page Simple`	HTML on multiple pages; suppresses title page and table of contents.
`Multi-page Unnumbered Chapters & Sections`	HTML on multiple pages; chapters and sections are not numbered.
`Multi-page Numbered Chapters & Sections`	HTML on multiple pages; chapters and sections are numbered.

RTF (DSSSL Print) and Word Stylesheets

RTF or Word Stylesheet	Description
`Standard Print`	Displays title page, table of contents, list of titles.
`Simple Print`	Suppresses title page, table of contents, list of titles.
`Compact Simple Print`	Minimizes page count, suppresses title, table of contents, list of titles.
`Large Type Print`	Uses 12 point font (slightly larger than Standard Print).
`Very Large Type Print`	Uses 24 point font and landscape paper orientation.
`Compact Print`	Minimizes white space to reduce page count.
`Unnumbered Chapters & Sections`	Chapters and sections are not numbered.
`Numbered Chapters & Sections`	Chapters and sections are both numbered.

Note Some Web and print stylesheets include an automatically generated list of titles. The list of titles includes a list of table titles and a list of figures with titles.

Viewing and Printing Your Report

To view your report after generation, you have the following options:

- To view the report automatically, select the **View report after generation** check box in the **Generation Options** section in the Properties pane on the right. When report generation is complete, the viewer associated with the report output format displays the report.
- To view the report from the MATLAB command line, use the `rptviewfile` command, such as `rptviewfile ch1-tutorial.`

- To view the report manually, browse to the location specified in the **Report File Location** section in the Properties pane on the right and open the file.

If you want to print your report, select the print option from the viewer.

Converting English Strings to Other Languages

Report Generator 2.0 and later uses the system language settings through the Java interface; therefore, if you change your system language, Report Generator should use the language you specify.

Alternatively, you can change the language directly in Java from the MATLAB command line. In the following example, the language is set to Italian:

```
java.util.Locale.setDefault(java.util.Locale.ITALY)
```

You can also set the preferred language directly in your `.rpt` file. To do so, follow these steps:

1. Right-click the **Report** component and select **Send to Workspace**.

   This displays the properties of the report, which are stored in the variable *ans*. You can access the report's `Language` property from the command line through this variable. By default, `Language` is set to `auto`, which indicates that the system's default language is in use.

2. Override the default value of `Language` by setting this property to your desired language; for example, `en` for English or `it` for Italian.

Autosaving the Report Template

To automatically save the report template before you generate a report, select **Auto save before generation**, which is located in the **Generation Options** section of the Properties pane on the right.

Regenerating Images

By default, the **Regenerate Simulink and Stateflow Images** option in the **Generation Options** section of the Properties pane on the right is not selected. Previously generated images are not regenerated each time you

generate a report, which results in faster report generation. If you have made any changes that could affect the image, you should select this option.

Including a Report Description

Use the **Report Description** field in the Properties pane on the right to make notes and comments on your report template. This text appears in the Properties pane when you select a report template in the Outline pane on the left.

Converting XML Documents

The following topics are covered in this section:

If you want to generate the same report in a different output file format without regenerating the report, you can use either the user interface (Report Explorer File Converter) or the command line (`rptconvert` command). The File Converter transforms a DocBook XML source file generated in Report Generator into a formatted document, such as HTML, RTF, or PDF.

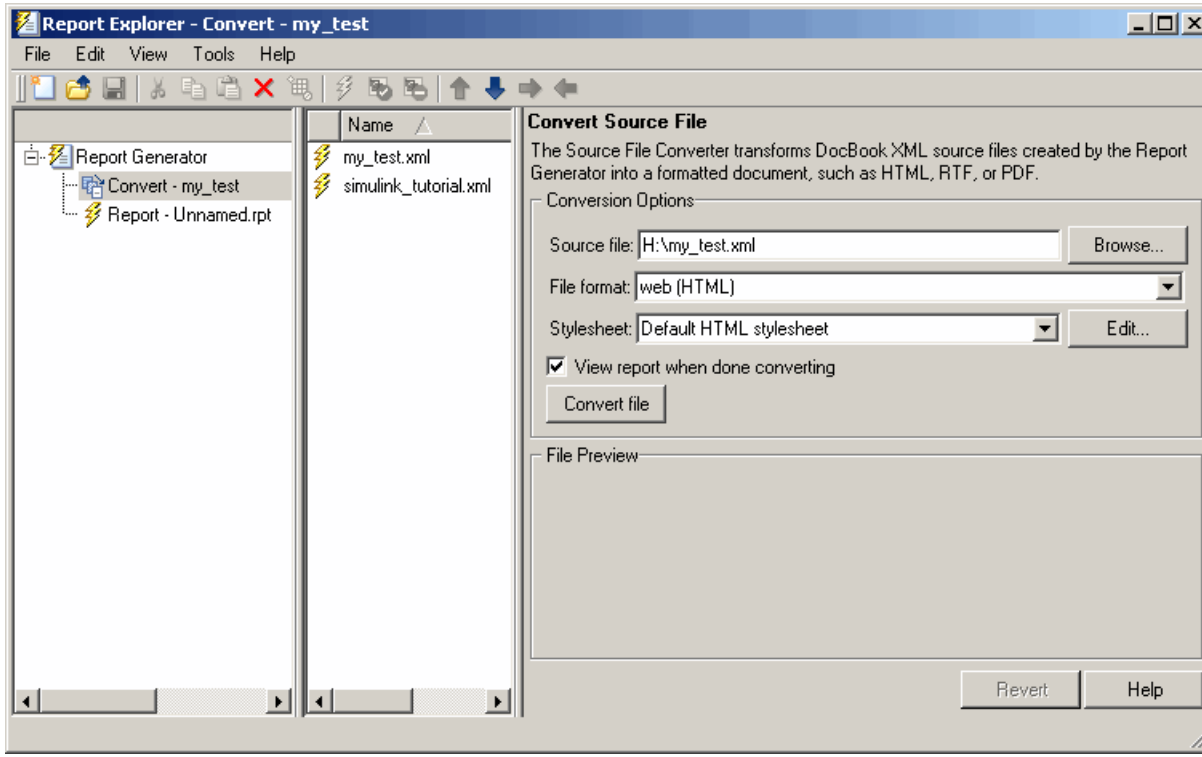
Note Report Generator can convert only XML source files created by the latest version of Report Generator.

Using the Report Explorer to Convert XML Documents

To open the **Convert** Properties pane:

1. In the Report Explorer, select **Tools > Convert source file**.

   The **Convert Source File** Properties pane appears. All XML files in your current directory are listed in the Options pane in the middle.



2 Select your XML source file by using one of the following methods:

- Click **Browse** in the Properties pane on the right to browse to the location of your XML source.
- Double-click a file name in the Options pane in the middle. This automatically enters the file name into the **Source file** field in the Properties pane.

3 Select your output format and stylesheet as follows:

a In the **File format** text box, select an output format.

b In the **Stylesheet** text box, select a stylesheet. The stylesheet choice depends on the selected output format.

c Optionally, click **Edit** if you want to customize the stylesheet you have selected.

For details about available formats and stylesheets, see “Setting the Report Format” on page 5-3.

For details about editing stylesheets, see Chapter 9, “Creating and Using Customized Styles”.

4. Use the **View Report when done converting** check box to indicate whether or not you want to view the report after it has been converted.

5. To begin the conversion, click **Convert file**.

Editing XML Source Files

Before you send a source file to the converter, you can edit it as text in the Report Explorer. To do so:

1. In the Outline pane on the left, open the File Converter. One way to do this is to right-click **Report Generator** and select **Convert source file**.

2. In the Options pane in the middle, select the source file that you want to edit.

3. In the Properties pane on the right, click **Edit as text**.

4. Use the MATLAB Editor to edit and save the text.

Converting XML Documents Using the Command Line

To convert files outside the Report Explorer interface, you can use `rptconvert` from the command line with varying levels of definition:

- `<ReportName>=rptconvert('<DocBookfilename>')`, where `<ReportName>` is the name of your report, and `DocBookfilename` is the name of the DocBook file (with or without the `.xml` extension). Report Generator uses the default format (HTML) and stylesheet (Single-Page Web).
- `<ReportName>=rptconvert('<DocBookfilename>',format)`, where the valid `format` arguments are
  - `Web(HTML)`
  - `Rich Text Format`
  - `PDF`

- `Word Document`

  Enter the format name in single quotation marks, for example, `'HTML'`. Report Generator uses the default stylesheet for the selected option.

- `<ReportName>=rptconvert('<DocBookfilename>',format,stylesheet);`

  For a discussion of valid `stylesheet` options for each output format, see "Setting the Report Format" on page 5-3. Type `rptconvert #STYLESHEETLIST` for a list of valid `stylesheet` arguments. Enclose the stylesheet name in single quotation marks.

Creating Log Files

In Report Generator, a log file describes a report template, including its report generation settings and components. A log file can be used for many purposes, including:

- As a debugging tool
- As a reference to a report template
- To share information about a report template through e-mail

A log file includes the following:

- report template outline
- Components and their attributes
- Generation status messages currently displayed in the **Generation Status** tab
- All MATLAB licenses

To generate a log file, select **File > Log File**. An HTML version of the log file is saved in the same directory as the report template and is named `<report_template_file_name_log>.html`. The log contains the following:

- Report template table
- Report template table of contents
- Table of details for each component in the report template
- Product version table

Generating M-code from Report Templates

Using Report Generator, you can generate M-code versions of report templates in the form of an M-file (`*.m`). An M-file of a report template is useful for a variety of purposes, including generating reports and modifying report templates programmatically.

To generate an M-file from Report Generator, a report template must already be loaded. Select **File > Generate M-File**. After generation, the M-file opens in the MATLAB Editor. The M-file is saved in the same location as the report template.

Working with Legacy Report Templates

Legacy report templates are files that you created in a previous version of Report Generator. You can use such legacy report templates in the latest version in either of two ways. You can load legacy report templates in the Report Explorer, or you can use the previous versions of Report Generator interfaces to work with legacy report templates.

The following topics are covered in this section:

Enabling Legacy Report Generator Interfaces

You enable the previous versions of interfaces at the MATLAB command line. To enable legacy interfaces, enter

```
RptgenML.v1mode(true)
```

at the MATLAB command line. MATLAB responds with

```
ans =

     1
```

A logical `1` indicates that Version 1 mode is on. To turn Version 1 mode off, enter

```
RptgenML.v1mode(false)
```

at the MATLAB command line. MATLAB responds with

```
ans =

     0
```

A logical `0` indicates that Version 1 mode is off.

Using Legacy Report Generator Interfaces

When Version 1 mode is enabled, you can use the previous versions of Report Generator interfaces. The MATLAB functions related to Report Generator also load the old interfaces.

Keep the following points in mind when using the previous versions of Report Generator interfaces:

- You cannot use a report template created in the latest version of Report Generator with the old interfaces. If you save a legacy report template in the latest version of Report Generator, you will not be able to work with the same report template in the old interface.
- You cannot use the old interfaces with the latest version of Report Generator. While the Setup File Editor can coexist with the Report Explorer, MathWorks does not recommend using the interfaces in this way.

Troubleshooting

This section contains information about how to deal with problems that you may have when using Report Generator. Topics covered include:

Managing Report Generator Memory Usage

By default, MATLAB sets a limit of 100 MB on the amount of memory the Java Virtual Machine (JVM) can allocate. The memory used by Report Generator to build a document must fit within this 100 MB limit. If you are having trouble processing large reports, you might need to increase the amount of memory Report Generator can allocate.

Running MATLAB Without a Desktop

One way to increase the amount of JVM memory available to Report Generator is to run MATLAB with `-nodesktop` mode enabled.

Note This option is available on UNIX platforms only.

Increasing the MATLAB JVM Memory Allocation Limit

You can also increase the amount of JVM memory available by increasing the MATLAB JVM memory allocation limit.

To increase this limit, create a file named `java.opts` and put it in your MATLAB startup directory. In this file, include the `-Xmx` option, specifying the amount of memory you want to give the JVM.

For example, to increase the JVM memory allocation limit to 128 MB, use the following syntax in the `java.opts` file:

```
-Xmx128m
```

Caution To avoid virtual memory thrashing, never set the `-Xmx` option to more than 66% of available physical RAM.

On UNIX systems, create the `java.opts` file in a directory where you intend to start MATLAB and move to that directory before starting MATLAB.

On Windows systems:

1. Create the `java.opts` file in a directory where you intend to start MATLAB.
2. Create a shortcut to MATLAB.
3. Right-click the shortcut and select **Properties**.
4. In the Properties dialog box, specify the name of the directory in which you created the `java.opts` file as the MATLAB startup directory.

6

Exporting Simulink Models to Web Views

This chapter covers the following topics, which apply to Simulink users.

Exporting a Model to the Web (p. 6-2)	Describes how to export a Simulink or Stateflow model to a Web view
Navigating a Web View (p. 6-8)	Describes how to use a Web view

Exporting a Model to the Web

The following topics are covered in this section:

Opening the Export Interface

The Report Generator interface for exporting a model to a Web view is either a dialog box or a Report Explorer pane. The two interfaces have the same capabilities and a similar appearance, with minor differences due to the different contexts in which the interfaces appear.

To Open the export interface from ...	Use ...	Export interface opens as...
Report Explorer	**Export Simulink to Web** command on **Tools** menu	Report Explorer pane
Simulink model window	**Export to Web** command on **File** menu	Dialog box
Stateflow diagram editor	**Export to Web** command on the**File** menu	Dialog box
Command line	`slwebview` command with no arguments	Report Explorer pane

Setting Export Options

The following topics are covered in this section:

In exporting a Web view, you can set options that, together, specify which Simulink systems or Stateflow charts to export:

- Whether to export Simulink systems or Stateflow charts that contain or are contained by the current system or chart
- Which masked blocks, if any, to include
- Whether to follow links to library blocks
- Whether to follow model reference blocks

Setting Options in the Export Interface

The **Include Systems** option specifies which Simulink systems or Stateflow charts to export.

Include Systems Option Value	Meaning
Current	Export only the Simulink system that is open in the model window or the Stateflow chart that is open in the diagram editor.
Current and above	Export the current Simulink system or Stateflow chart and all systems or charts that contain it.
Current and below	Export the current Simulink system or Stateflow chart and all subsystems or subcharts that it contains.
All	Export all Simulink systems or Stateflow charts in the model that contains the current system or chart (the default value).

The **Look Under Masks** option is available only when you are exporting Simulink subsystems or Stateflow subcharts that the current system or chart contains. This option specifies which masked blocks, if any, in subsystems or subcharts to export.

Look Under Masks Option Value	Meaning
No masked blocks	Do not export any masked blocks.
Masks with no workspace and no dialog	Export only graphical masked blocks—those that have no workspaces and no dialog boxes (the default value).
Masks with no dialog	Export only functional masked blocks—those that have no dialog boxes.
All masked blocks	Export all masked blocks.

The **Follow Library Links** option specifies whether or not to follow links into library blocks.

Follow Library Links Option Value	Meaning
Selected	Follow links into library blocks.
Unselected	Do not follow links into library blocks (the default value).

The **Follow model reference blocks** option specifies whether or not to follow links into referenced models.

Follow model reference blocks Option Value	Meaning
Selected	Follow links into referenced models.
Unselected	Do not follow links into referenced models (the default value).

Setting Options on the Command Line

The first argument to the `slwebview` command is the name (a string) or a handle identifying the Simulink system or Stateflow chart to export. Remaining arguments are alternating name and value pairs (strings) setting export options.

The option named `'SearchScope'` specifies which Simulink systems or Stateflow charts to export.

'SearchScope' Option Value	Meaning
`'Current'`	Export only the Simulink system or the Stateflow chart specified by the first argument.
`'CurrentAndAbove'`	Export the Simulink system or the Stateflow chart specified by the first argument and all systems or charts that contain it.
`'CurrentAndBelow'`	Export the Simulink system or the Stateflow chart specified by the first argument and all systems or charts that it contains (the default value).
`'All'`	Export all Simulink systems or Stateflow charts in the model that contains the system or chart specified by the first argument.

The option named `'LookUnderMasks'` is meaningful only when you are exporting Simulink subsystems or Stateflow subcharts that the specified system or chart contains. This option specifies which masked blocks, if any, in subsystems or subcharts to export.

'LookUnderMasks' Option Value	Meaning
`'none'`	Do not export any masked blocks.
`'graphical'`	Export only masked blocks that have no workspaces and no dialog boxes (the default value).
`'functional'`	Export only masked blocks that have no dialog boxes.
`'all'`	Export all masked blocks.

The option named `'FollowLinks'` specifies whether or not to follow links into library blocks.

'FollowLinks' Option Value	Meaning
`'on'`	Follow links into library blocks.
`'off'`	Do not follow links into library blocks (the default value).

'FollowModelReferenceBlocks' Option Value	Meaning
`'on'`	Follow links into referenced models.
`'off'`	Do not follow links into referenced models (the default value).

The option named `'ViewFile'` specifies whether or not to display the Web view in a Web browser after generating it.

'ViewFile' Option Value	Meaning
`true`	Display the Web view in a Web browser after generating it (the default value).
`false`	Do not display the Web view in a Web browser after generating it.

Reviewing and Generating a Web View

Each Web view export dialog box is associated with a particular Simulink model window or Stateflow diagram editor window, and is therefore also associated with the system or chart displayed in that window.

The Report Explorer Web view export pane shows information about the Simulink system or Stateflow chart that is current at the time you display the pane from the Report Explorer or the `slwebview` command. However, it exports the system or chart that is current at the time you click **Export Model**. If you change the current system or chart while the Report Explorer Web view export pane is displayed, the Report Explorer pane does not

automatically change to show information about the newly current system. To update the Report Explorer pane and display information about the current system or chart, click **Refresh**.

The Web view export dialog box or the Report Explorer pane displays a list of layers—that is, Simulink systems or Stateflow charts—to be exported. The list of layers depends on the system or chart associated with the dialog box or the Report Explorer pane. It also depends on the export options you choose.

To generate a Web view, click **OK** in the export dialog box or click **Export Model** in the Report Explorer pane. This action generates two sets of files in the current MATLAB directory:

- An HTML file to be displayed in a Web browser
- A directory with the same name as the HTML file, that contains supporting files

To send the Web view to another computer, you must send both the HTML file and the directory of supporting files, and these must remain together. You can package the HTML file and the supporting directory into a zip file for transfer.

When you generate a Web view from the export dialog box or the Report Explorer pane, the Web view is also displayed in a Web browser. The `slwebview` command displays the Web view in a Web browser by default, but you can suppress the display by setting the `'ViewFile'` option to `false`.

Navigating a Web View

The following topics are covered in this section:

To display a Web view, you need a Web browser with support for Scalable Vector Graphics (SVG), implemented natively or through a plug-in. For detailed information on software requirements and limitations for using Web views, see *Report Generator Release Notes*.

A Web view consists of a top-level HTML file and a folder, in the same directory, containing support files. To display the Web view, open the top-level HTML file in a Web browser.

The Web view display has these components:

- Graphical display of the current layer (Simulink system or Stateflow chart)
- A hierarchical treeview browser for easy navigation through layers
- Four navigation buttons at top left
- Menu of navigation commands, displayed when you move the mouse over the leftmost button
- Shortcut menu, displayed in some browsers when you right-click the current layer

If you have Simulink and Report Generator installed, you can display an example Web view of the Simulink F-14 demo model by typing these commands at the MATLAB command prompt:

```
sldemo_f14; slwebview(gcs)
```

Viewing the Current Layer

As you move the mouse over the current layer, parameter names and values are displayed for each system, chart, block, and line.

You can use the mouse, menu commands, keyboard accelerators, and the arrow keys on your keyboard to pan and zoom within the current layer.

Action	Navigation Menu Command	Shortcut Menu Command	Keyboard Accelerator
Zoom in	**Zoom in**	**Zoom in**	**R**
Zoom out	**Zoom out**	**Zoom out**	**V**
Zoom to normal size (100%)	**Zoom to 100%**	None	**1**
Zoom to largest size that fits in browser window	**Zoom to full screen**	**Zoom to full screen**	**F**
Pan up	**Pan up**		**E**
Pan down	**Pan down**		**C**
Pan left	**Pan left**		**D**
Pan right	**Pan right**		**G**
Pan with mouse (drag layer while in panning mode)	**Pan with mouse** (click to toggle panning mode)	**Pan with mouse** (click to toggle panning mode)	**P**,**Q**, or **SPACEBAR** (press to enter panning mode; release to exit panning mode)

The shortcut menu command **Find text** searches for a text string in the current layer.

Note **Find text** is only available in Internet Explorer.

Viewing Other Layers

If the current layer contains an exported subsystem or subchart, you can navigate to that subsystem or subchart and make it the current layer by clicking it. The current layer is also selected in the treeview browser, and its

subsystems or subcharts are displayed in the treeview browser hierarchically. If blocks in your current layer are linked to libraries, the Web view shows to which libraries they belong.

Note If your current layer has no subsystems or subcharts, it is not displayed in the treeview browser.

As you move the mouse over an exported subsystem or subchart, it is highlighted or the cursor changes, depending on which Web browser you are using. You also see the following visual cues that enable you to quickly assess your location in the model:

- The subsystem or subchart's corresponding icon in the treeview browser is also highlighted, showing to where in the hierarchy you have navigated.
- The names of the subsystem or subchart's parent systems are grayed out and italicized in the treeview browser.

You can use the navigation buttons at the top left to view other layers.

Action	Navigation Button
View the previous layer in the view history	**View previous**
View the top-level layer	**View home**
View all layers in one window	**View all layers**

If you have clicked **View all layers** to view all layers in one window, clicking that button again returns to the previous view. This action is also available as the **Show all layers** command on the navigation and shortcut menus and as the **X** keyboard accelerator. When you view all layers in one window, you can click a layer to view it. You can also use the **Find text** command on the shortcut menu to search for a text string in all layers.

7

Working with Components

The tutorials in this chapter, which applies to both MATLAB and Simulink users, cover the following topics.

Working with Looping Components (p. 7-2)	Describes the use of looping components
Working with Property Table Components (p. 7-14)	Describes the use of property table components
Working with Summary Table Components (p. 7-28)	Describes the use of summary table components

Working with Looping Components

The following topics are covered in this section:

About Looping Components

A *looping component* runs its child components several times. Report Generator offers numerous looping components, including logical loops, Simulink loops, Stateflow loops, and Handle Graphics® loops. Most loops share common characteristics. For that reason, this tutorial focuses on the Figure Loop component as representative of most looping components.

The Figure Loop component runs its child components several times, each time applying its child components to Handle Graphics figures. The `figloop-tutorial` report template generates a report that documents multiple Handle Graphics figures.

The `figloop-tutorial` report template contains a Figure Loop component as well as other components, including the following:

- Title Page
- Chapter/Subsection
- Evaluate MATLAB Expression
- Import File
- Summary Table
- Handle Graphics Name
- Handle Graphics Linking Anchor
- Figure Snapshot
- Handle Graphics Property Table

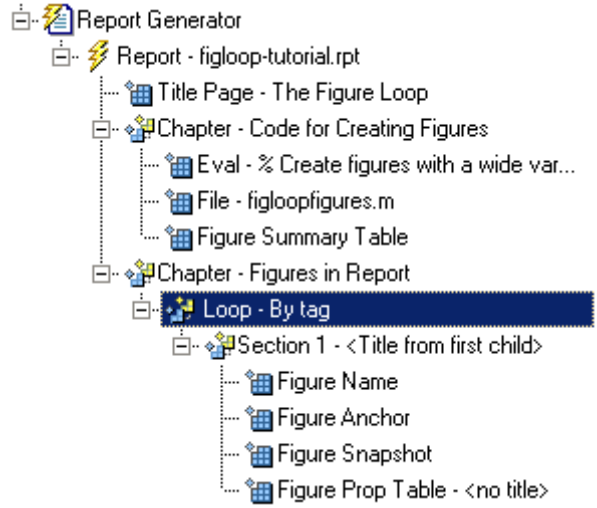
Note For details about property table components, see “Working with Property Table Components” on page 7-14.

Editing the Report Template

To use the figure loop tutorial and edit the report template, type the following at the command line:

```
setedit figloop-tutorial
```

The report template for the figure loop tutorial appears in the Report Explorer.



Creating Handle Graphics Figures

The following topics are covered in this section:

There are three important Handle Graphics concepts you need to be familiar with before proceeding with the rest of this tutorial:

- Invisible figures
- `HandleVisibility` parameter
- Tags

For details on Handle Graphics concepts, see the MATLAB Handle Graphics documentation.

Invisible Figures

When you type `figloopfigures`, Figures 1, 3, and 5 appear, but Figures 2 and 4 do not. Figures 2 and 4 are invisible (`visible='off'`), so you can only see three figures. An invisible figure does not appear on screen; it exists, but it is hidden.

HandleVisibility Parameter

The figures An Application on page 7-7 and Invisible Application on page 7-10 have `HandleVisibility='off'`. `HandleVisibility` is a parameter of a Handle Graphics figure, and it determines whether the handles in your figure are visible. The figure can be found because it is a child of root. For details on `HandleVisibility`, see the MATLAB Graphics documentation.

To list all figures in your workspace that have `HandleVisibility='on'`, type `get(0,'children')`at the command line. The results show that the figures Peaks Data on page 7-8, Membrane Data on page 7-6, and An Application on page 7-7 have `HandleVisibility='on'`.

```
ans =
     5
     1
     2
```

Note that most figures have `HandleVisibility='on'`. To turn `HandleVisibility` off, you have to specify `HandleVisibility='off'` with the following:

```
set(handle,'HandleVisibility','off')
```

Tags

A Handle Graphics figure can have a tag. A tag is a Handle Graphics figure parameter. A tag does not appear in a figure; it is an identifier that you specify.

If you want to see what tag, if any, a figure has, type `get(handle, 'tag')`. The handle of a figure is the figure number; enter the figure number for *handle*. For example, to see whether Membrane Data on page 7-6 has a tag, type `get(1, 'tag')`, and MATLAB outputs the following response:

```
ans =
     membrane
```

This means that the figure Membrane Data on page 7-6 has a tag called `membrane`. Every figure in this tutorial has a tag. However, by default, figures have empty tags.

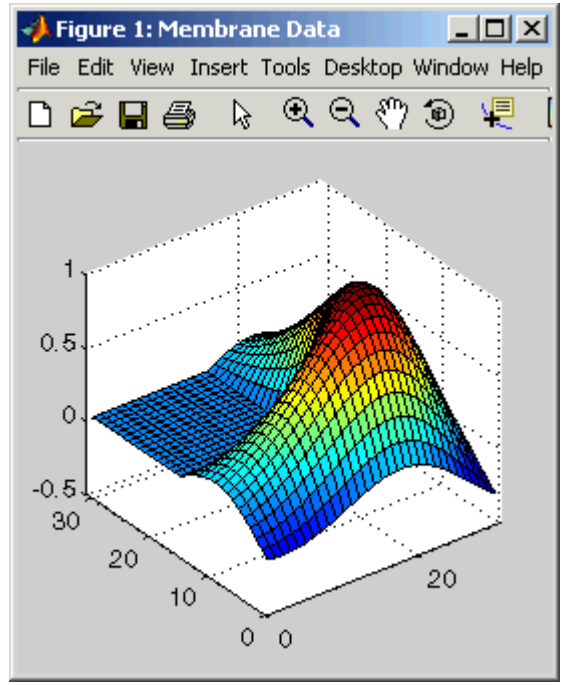
Figure Summary

The following table shows visibility (figure is visible if `visible='on'` and invisible if `visible='off'`), `HandleVisibility`, and tags for the figures in the tutorial.

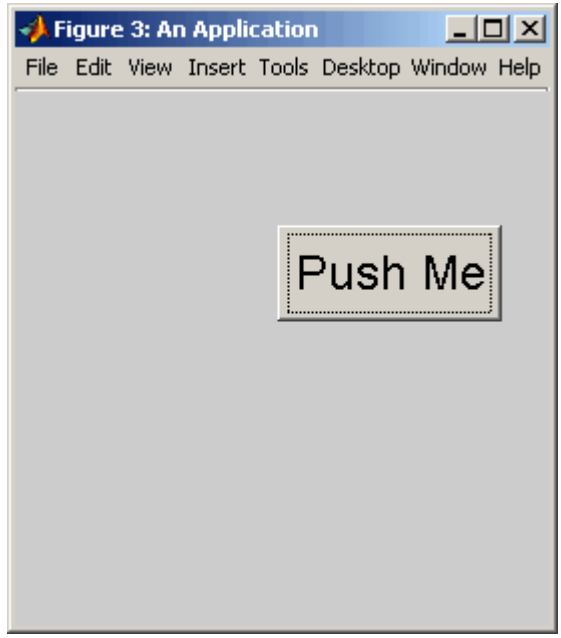
Name	Visible	HandleVisibility	Tag
Membrane Data	`on`	`on`	membrane
Invisible Membrane Data	`off`	`on`	membrane
An Application	`on`	`off`	app
An Invisible Application	`off`	`off`	app
Peaks Data	`on`	`on`	peaks

To create the Handle Graphics figures, type `figloopfigures` at the MATLAB command line. The figures Membrane Data on page 7-6, An Application on page 7-7, and Peaks Data on page 7-8 appear on the screen, but Invisible Membrane Data on page 7-9 and Invisible Application on page 7-10 do not because they are invisible.

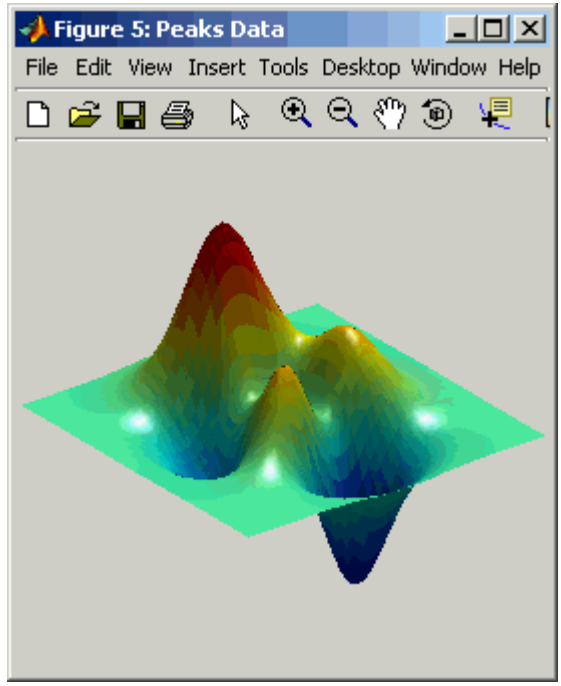
Following are the figures Membrane Data on page 7-6 (`membrane`), An Application on page 7-7 (`app`), and Peaks Data on page 7-8 (`peaks`).



Membrane Data

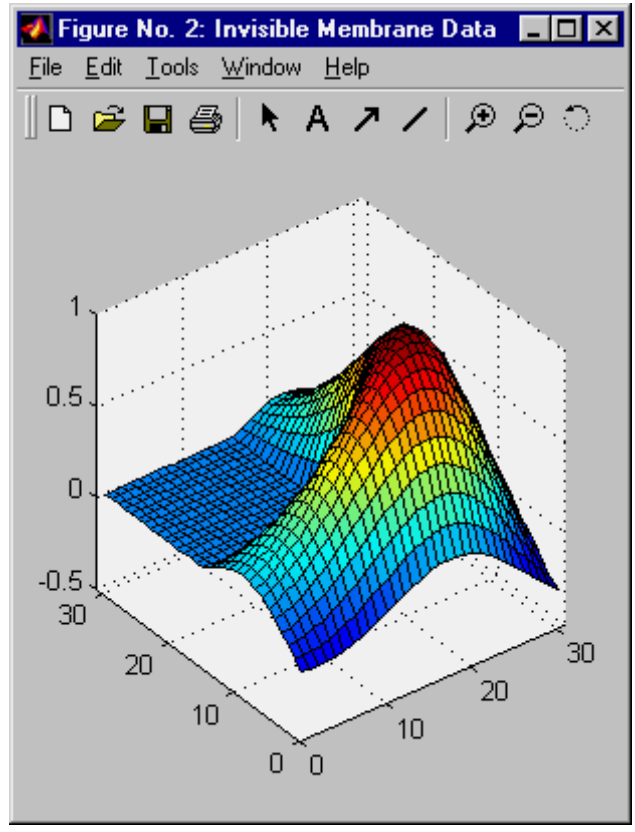


An Application

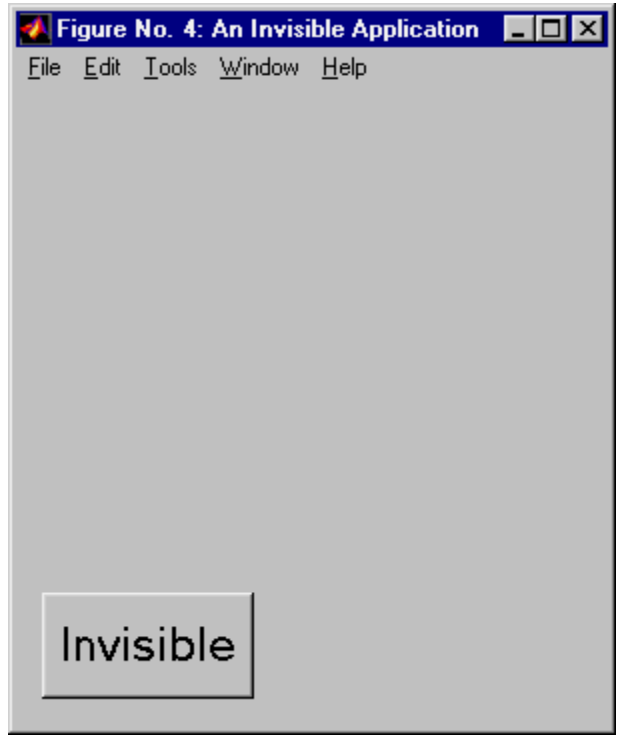


Peaks Data

The figures An Application on page 7-7 (`membrane`) and Invisible Application on page 7-10 (`app`), which do not appear on the screen, are shown below.



Invisible Membrane Data

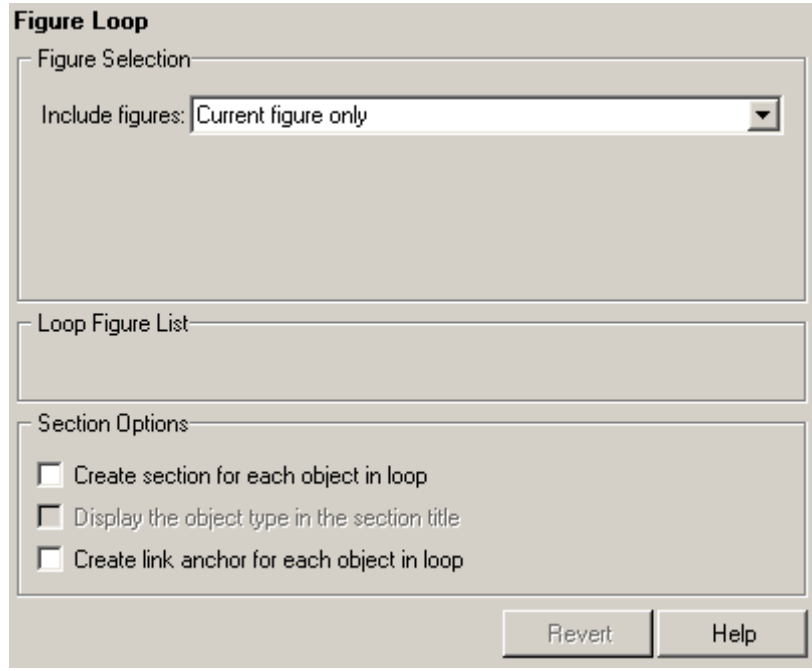


Invisible Application

Before you continue with the rest of the tutorial, it is important that you understand some important Handle Graphics concepts, such as invisible figures, `HandleVisibility` parameters, and tags.

Editing the Figure Loop Component

Select the `Figure Loop` component from the Outline pane on the left. The Properties pane for the `Figure Loop` component appears.



Looping on the Current Figure

To include a snapshot of the a figure in the report, you must select the figure. For example, select thePeaks Data on page 7-8 window to include a snapshot of it in your report.

To include only the current figure in the report, select `Current figure only` from the `Include figures` list. Click **Report** to generate a report.

Note When you choose `Current figure only`, "current figure" refers to the figure that is current when the report is generated. This might not be the same figure you selected as the current figure in the Report Explorer before report generation. For example, if the report generation process creates figures in your report, the last figure created with `HandleVisibility = 'on'` is the current figure.

Looping on All Visible Figures

To include snapshots of all visible figures in your report, select `Visible figures` in the `Include figures` list. This option inserts a snapshot and property table for all figures that are currently open and visible.

With HandleVisibility='on'. The `Data figures only (Exclude applications)` option is selected by default and excludes figures with `HandleVisibility = 'off'` from the loop. Click the **Report** button to generate a report. Notice that the report contains the figures Membrane Data and Peaks Data.

With HandleVisibility='on' or 'off'. Now clear the `Data figures only` option. The figures with `HandleVisibility = 'on'`(Membrane Data on page 7-6 and Peaks Data on page 7-8) and `HandleVisibility = 'off'` (An Application on page 7-7) are now included in the loop.

Click **Report** to generate a report. Notice that the report now contains Membrane Data on page 7-6, An Application on page 7-7, and Peaks Data on page 7-8.

Looping on Figures with Certain Tags

To include figures with certain tags, select the `All figures with tags` option. When you select a tag, all figures with that tag are included, regardless of whether the figure is visible or whether `HandleVisibility` is `'on'` or `'off'`.

For a brief discussion of Handle Graphics tags, see "Tags" on page 7-5.

Modifying Section Options

The Figure Loop component provides controls for section options. In a Report Generator loop, a section refers to a space in the generated report in which information appears, including text, images, tables, and so on. Using the options available in the **Section Options** pane, you can alter how the sections generated by the loop are displayed in the report.

Create a Section for Each Object in the Loop. To enable the loop to create a section for each object in the loop, select this option. This option is useful when a loop does not contain a child Chapter/Subsection component that organizes the loop results. When selected, the loop automatically creates an individual section for each object found in the loop and uses the object title as the section title.

In the `figloop-tutorial` report template, this option is not needed because the Figure Loop component has a child Chapter/Subsection component that already organizes the loop results.

Display the Object Type in the Section Title. To include the object type in the section title, select this option. When selected, the section titles in the generated report are prefaced by object types. For example, in a report generated from the `figloop-tutorial` report template without this option selected, the list of figures produced by the loop would look like this:

```
Membrane Data
Invisible Membrane Data
An Application
An Invisible Application
```

With the option enabled, the list of figures would look like this:

```
Figure - Membrane Data
Figure - Invisible Membrane Data
Figure - An Application
Figure - An Invisible Application
```

Create a Link Anchor for Each Object in the Loop. Select this option to create a hyperlink to the object in the generated report.

Working with Property Table Components

The following topics are covered in this section:

Property table components display property name/property value pairs in a table. Report Generator offers numerous property table components, including the MATLAB Property Table component, the Simulink Property Table component, the Stateflow Property Table component, and so on.

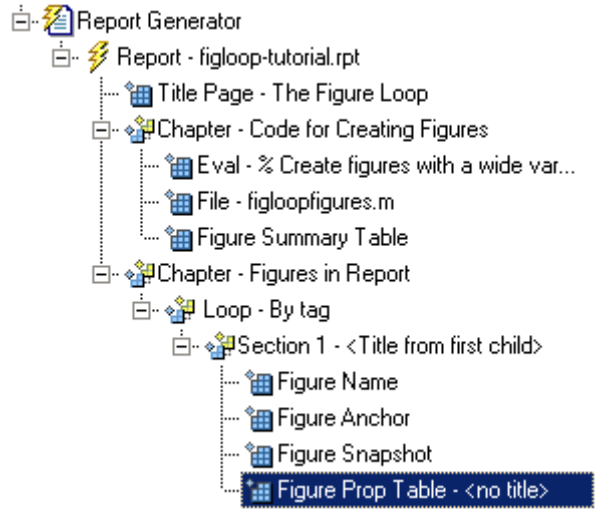
Report Generator property table components share common characteristics. For that reason, this tutorial uses the Handle Graphics Property Table component as a representative of all property table components.

Opening the Report Template

To use the figure loop tutorial and edit the report template, type the following at the MATLAB command line:

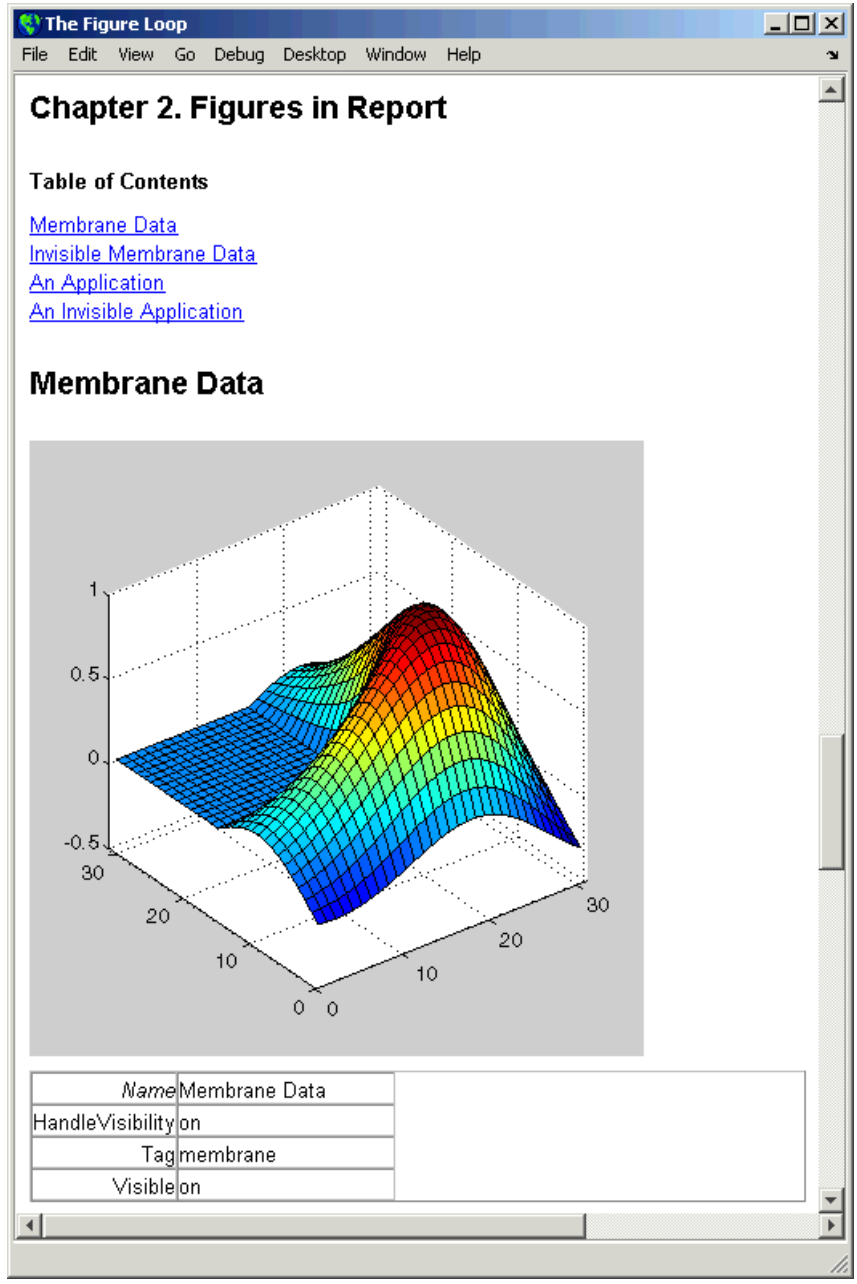
```
setedit figloop-tutorial
```

The figure loop tutorial report template appears in the Report Explorer.

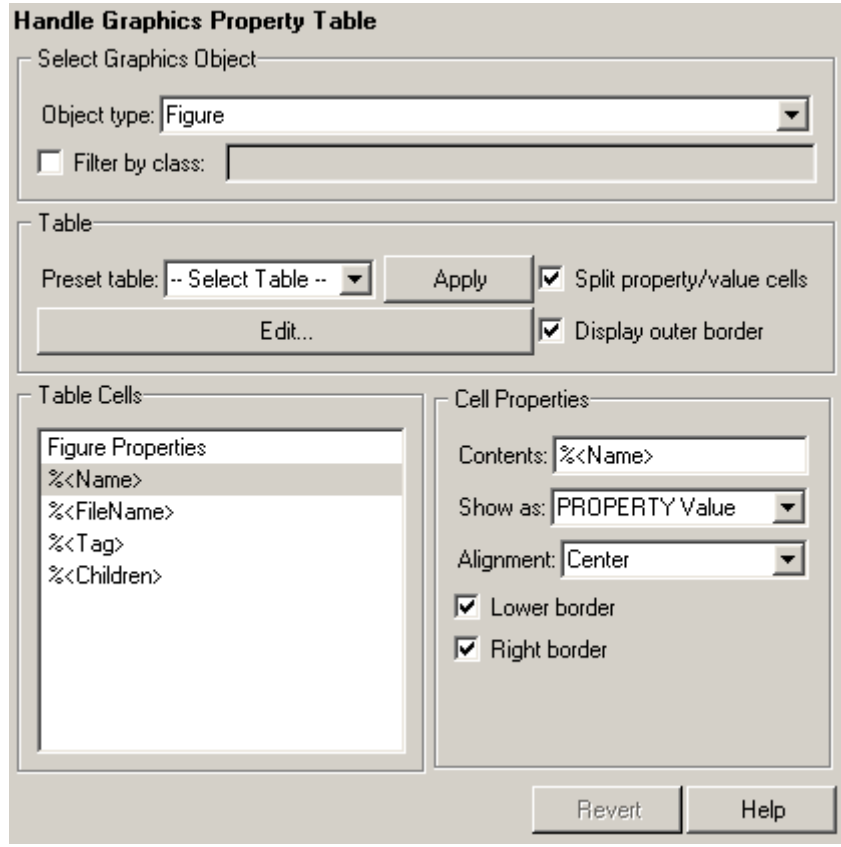


Examining the Property Table Output

The Handle Graphics Property Table component displays property name/property value pairs in a table for Handle Graphics figures, axes, or plots. The property table appears below the graphic display.


The Figure Loop
File Edit View Go Debug Desktop Window Help
Chapter 2. Figures in Report
Table of Contents
Membrane Data
Invisible Membrane Data
An Application
An Invisible Application
Membrane Data
1
0.5
0
-0.5
30
20
10
0
0
10
20
30
Name Membrane Data
HandleVisibility on
Tag membrane
Visible on

Property pages for all property table components are similar in form. The following figure shows the Properties pane for the Handle Graphics Property Table component.



Click **Edit** to modify most table settings.

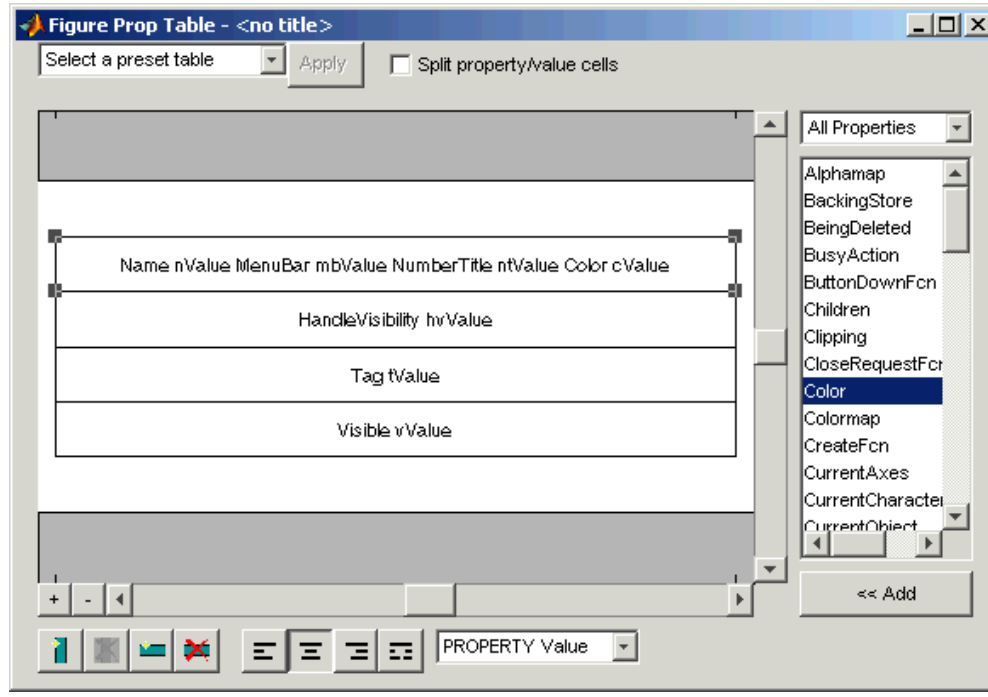
Displaying Property Name/Property Value Pairs

The following topics are covered in this section:

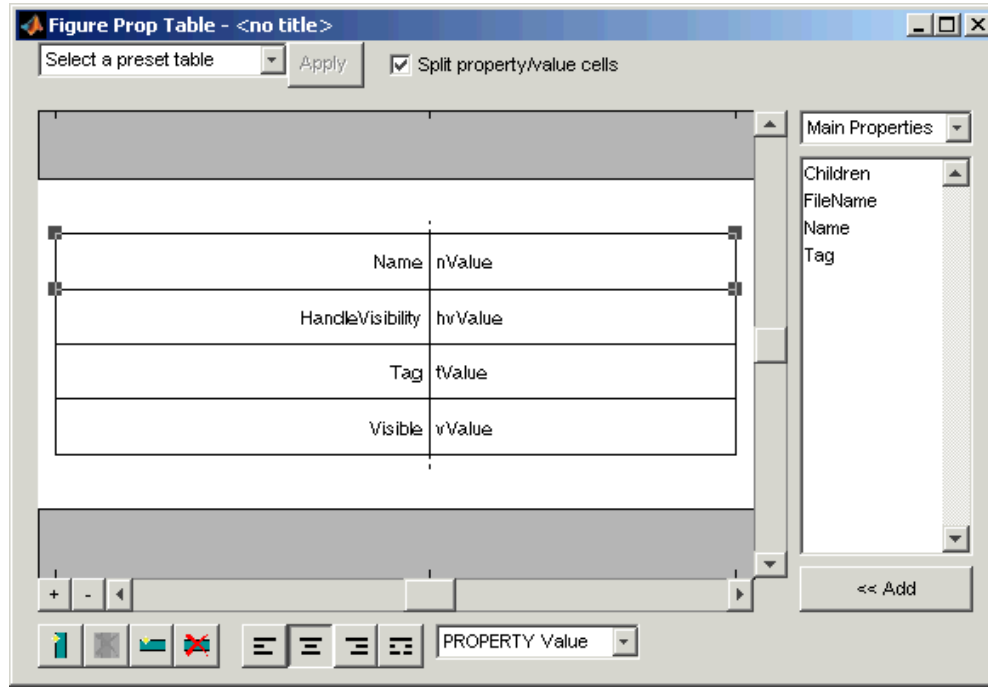
- “Options” on page 7-20

Split Property/Value Cells

In the Handle Graphics Property Table component’s Properties pane, click **Edit**, and a dialog box appears. To have the property name and property value appear together in one cell, clear the **Split property/value cells** check box. This is nonsplit mode. In nonsplit mode, you can have more than one property name/property value pair per cell as well as additional text.



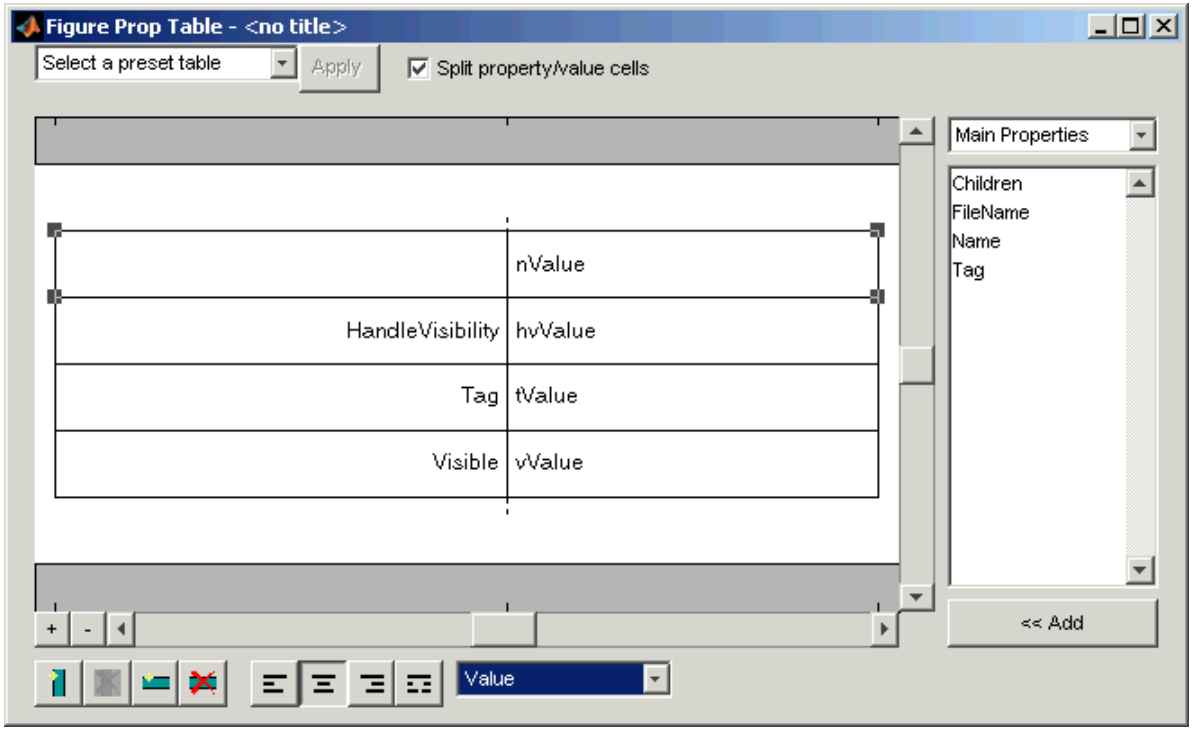
To have the property name and property value appear in adjacent horizontal cells in the table, select the **Split property/value cells** check box. In this case, the table is in split mode and there can be only one property name/property value pair in a cell.



If you have more than one property in a cell, only the first pair appears in the report; all subsequent pairs are ignored.

Display Options

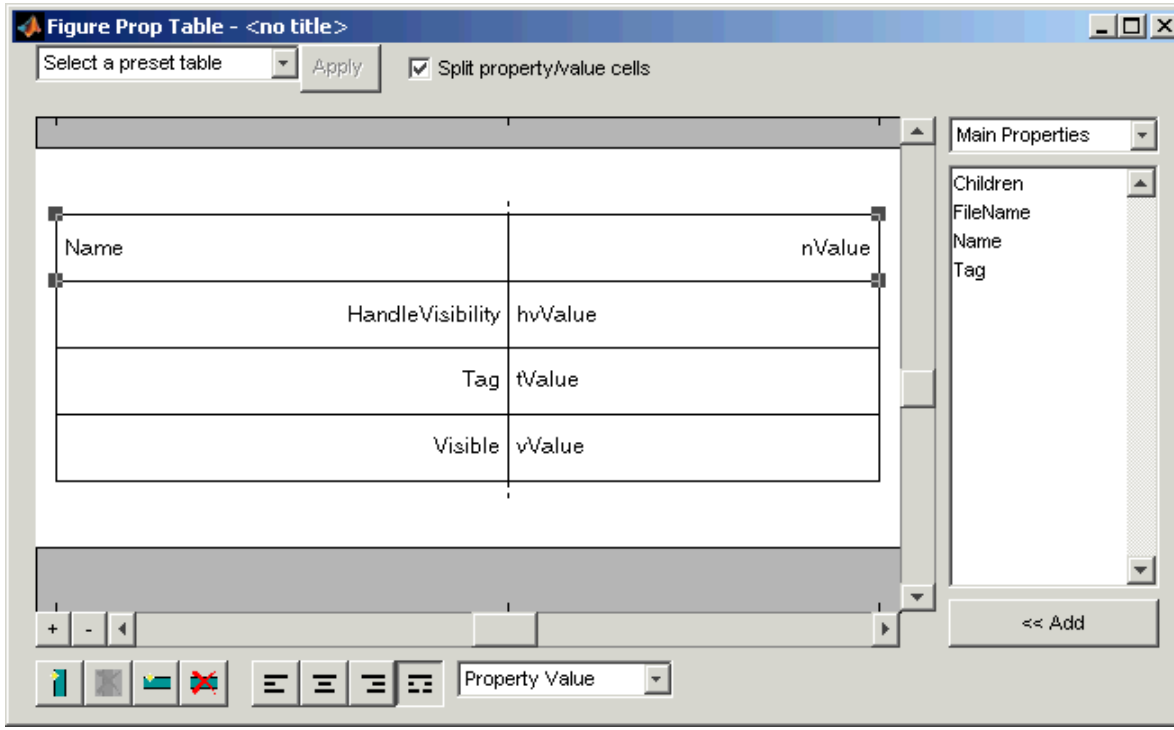
Each table cell in the report can display a property name/property value pair in several ways. To select how a property name/property value pair appears in a table cell in the report, select that field in the table and choose `Value` in the **Display As** text box.



Note that in the selected table row, only the value appears.

Options

Use the four justification buttons in the toolbar below the figure to specify the way that text in a cell appears in the report.

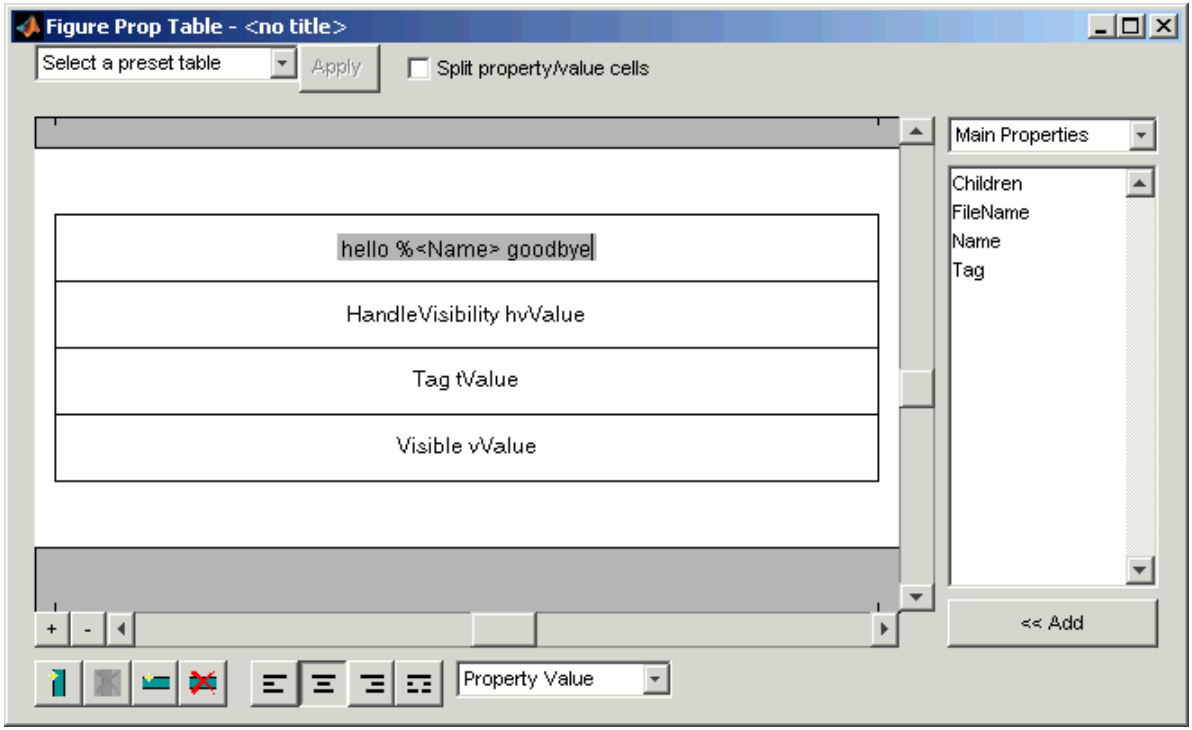


Note that in this example, the selected table row is double justified.

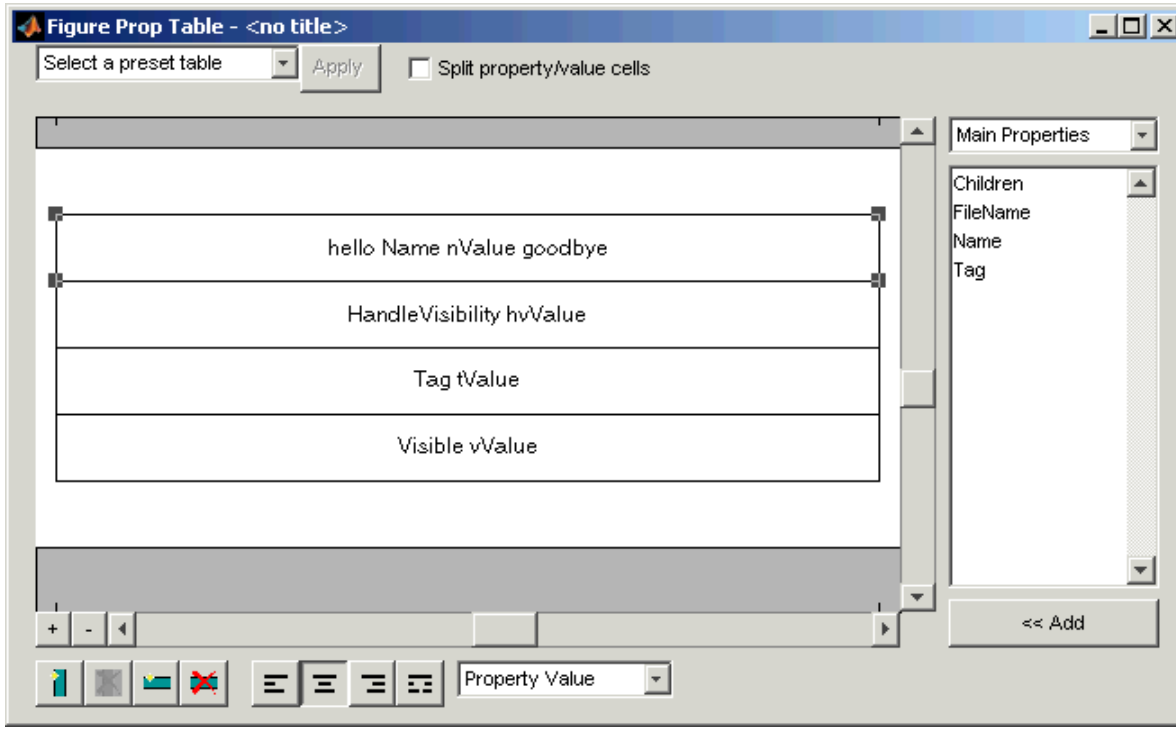
You can also right-click a table row and change justification options in the context menu that appears.

Putting Text into a Table Cell

You can type text in a table cell in the table dialog box. For the text to be visible, the table must be in nonsplit mode. To enter text in a cell, double-click the table cell. A gray box appears with the label for the cell property.



If you type text outside the angle brackets, the text appears as text in the report. If you type text inside the table brackets, the text must be a valid property name. If you type an invalid property name, that property name appears in the report without a property value.



Editing the Table Title

The table has a title that you can edit. The title of a table is the same as the value of the `%<Name>` property. The title of the table can contain properties and text. The title is always in nonsplit mode. To modify the title, modify the `%<Name>` entry.

If the table has a title, it appears in the list of tables in the generated report. Some stylesheets offer a list of tables. For a list of stylesheets that offer this option, see “Setting the Report Format” on page 5-3.

To enter a property, follow the steps described in “Adding, Replacing, and Deleting Properties in a Table” on page 7-24.

Adding, Replacing, and Deleting Properties in a Table

To add, replace, or delete properties in a table, use the table editor dialog box. Click **Edit** in the Property Table Component's Properties pane.

Adding a Property

To add a Handle Graphics property to a table cell in the table editor:

1. Start the table editor.
2. Add a new, blank table row to the current table.
   a. Select a table row directly below the location where you want to add a new property.
   b. Click **Add Row Above Current Cell**. A blank table row appears.
3. Add the property to the new table row.
   a. Select the new table row.
   b. In the Properties Type menu, select the property type that you want. In the Properties menu, select the property that you want to add.
   c. Click **Add** or double-click the property name. The property appears in the table row.

If you know the name of the property, you can also type the property name in the cell. For details, see "Putting Text into a Table Cell" on page 7-21. For information about adding new table rows, see "Adding and Deleting Columns and Rows" on page 7-25.

Replacing or Deleting Properties

In nonsplit mode, you can replace a property in a table cell in the report template by following the instructions in "Putting Text into a Table Cell" on page 7-21. You can delete a property by backspacing over it.

To replace a Handle Graphics property in a cell when the table is in split mode, follow the instructions in "Adding a Property" on page 7-24. You cannot use these steps to delete a Handle Graphics property in a cell when the table is in nonsplit mode, because such cells can have more than one property name/property value pair.

Displaying or Hiding Table and Cell Borders

You can toggle the table and cell borders on and off in the Property Table by using the context menu.

To toggle a cell border on and off, place your cursor in the desired cell and right-click to invoke the context menu. Choose **Cell borders** and then select **Top**, **Bottom**, **Right**, or **Left** to toggle the desired border on or off.

Adding and Deleting Columns and Rows

To add or delete a column or row, select a cell, and then click one of the buttons described in the following table.

Button	Action
	Add column (added to the left of the selected column)
	Delete selected column
	Add row (added above the selected row)
	Delete selected row

You cannot delete a row or column when it is the only row or column in the table.

Resizing Columns

To resize the column width, click and drag the vertical border between cells.

Zooming and Scrolling

While you are editing the table in the report template, you can zoom in and out of the table with the zoom buttons, located to the left of the horizontal scroll bar.

Button	Action
+	Zoom in
-	Zoom out

You can scroll vertically and horizontally with the corresponding scroll bars around the table.

Selecting a Table

The following topics are covered in this section:

You can select a preset table to display property name/property value pairs, or you can create a custom table.

Preset Tables

A preset table is a table that is already formatted and set up. You can select a preset table in the preset table drop box in the upper-left corner of the Properties pane. To apply a preset table, select the table and click **Apply**.

Custom Tables

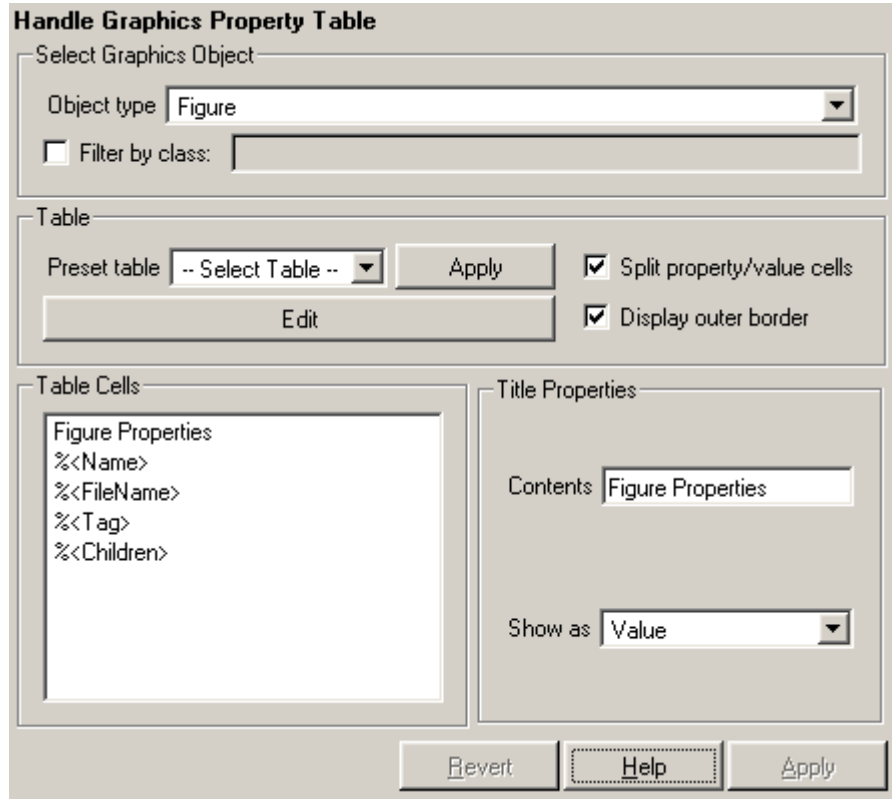
To create a custom table, you can select a preset table and modify that table to fit your needs. You might want to start with the **Blank 4x4** preset table. You can add and delete rows and you can add properties.

Note You cannot save a custom table as a preset table. If you create a table and then select a preset table, all your changes are lost.

Selecting Object Types

Property and summary table components offer multiple object types on which to report. For example, the Handle Graphics Property table lets you report on a figure, an axes object, or a Handle Graphics object.

To select a different object type, use the **Object type** list in the component's Properties pane.



The Handle Graphics Summary Table component offers the same object type options. Deciding which object type to select depends on the object that you want to include in the generated report. The Simulink Property Table and Simulink Summary Table components include block, model, signal, and system object types.

Working with Summary Table Components

The following topics are covered in this section:

About Summary Table Components

Summary table components insert tables containing the details on the selected object into generated reports. The generated tables contain one object per table row, and each object property is a table column. Report Generator offers numerous summary table components, including the Handle Graphics Summary Table component, Simulink Summary Table component, and the Stateflow Summary Table component.

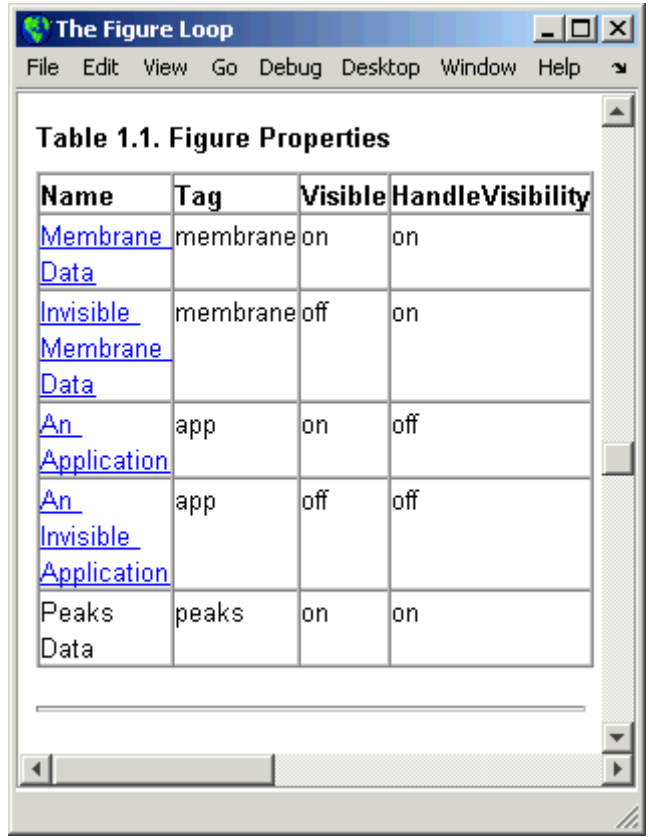
Report Generator summary table components share common characteristics. For that reason, this tutorial uses the Handle Graphics Summary Table component as a representative of all summary table components.

A summary table can contain many properties. For example, the `simulink-summary` report template contains multiple Simulink Summary Table components that report on Simulink blocks in the `f14` model and their properties.

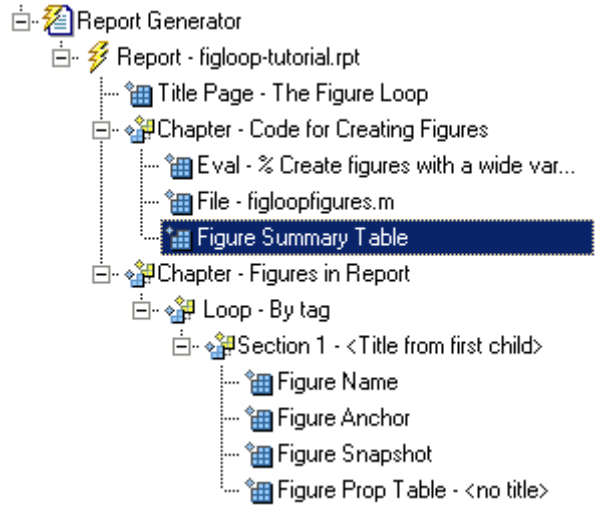
Like the other sections in the chapter, this tutorial uses the `figloop-tutorial` report template.

Examining the Report Output and the Report Template

In the generated report, summary table components generate tables that contain the specified properties for the selected object. The following figure shows the summary table produced by the `figloop-tutorial` report template.



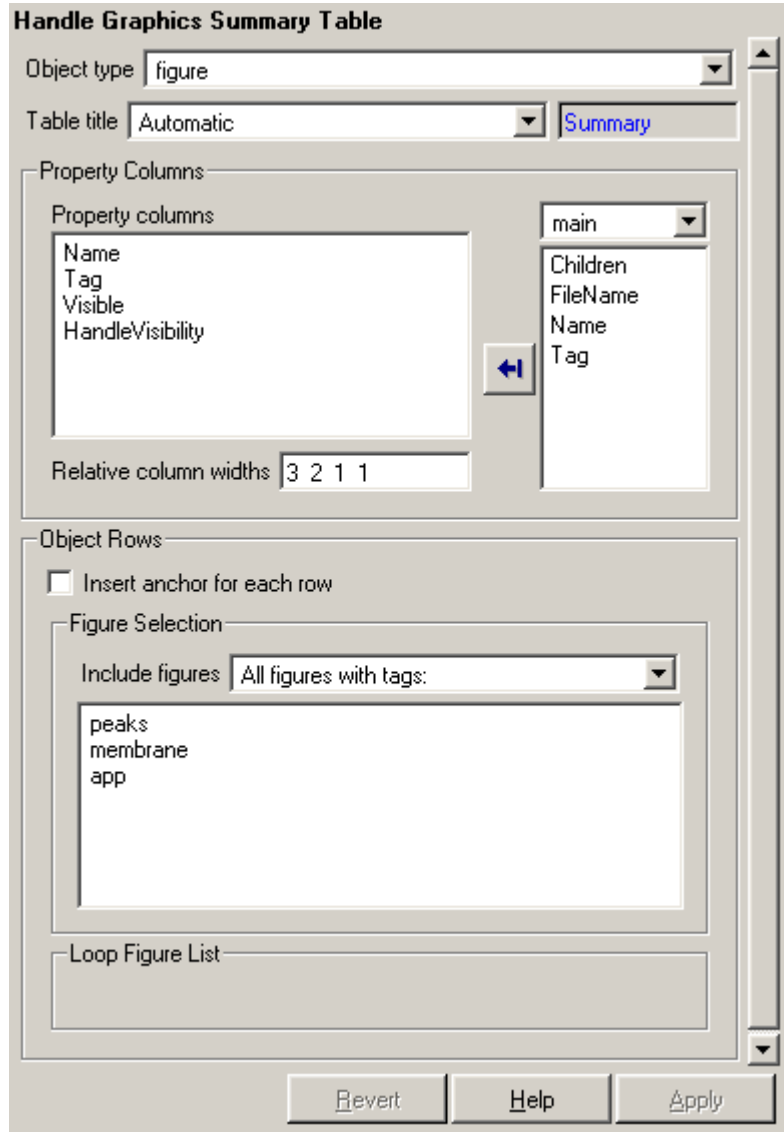
In the `figloop-tutorial` report template, the Handle Graphics Summary Table component is the child of a Chapter/Subsection component.



Selecting an Object Type

You can select different Handle Graphics object types for the summary table, including blocks, signals, systems, and models, with the **Object type** list in the component's Properties pane. The `figloop-tutorial` report template contains one Handle Graphics Summary Table component, and it reports on figure objects.

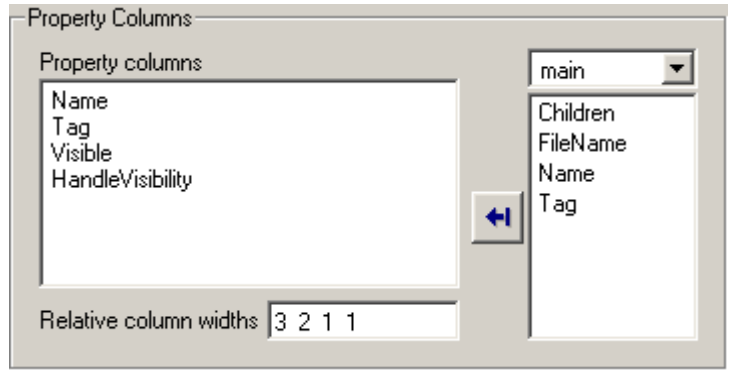
The properties available to add to the table depend on the selection in the **Object type** list.


Handle Graphics Summary Table
Object type
figure
Table title
Automatic
Summary
Property Columns
Property columns
Name
Tag
Visible
HandleVisibility
main
Children
FileName
Name
Tag
Relative column widths
3 2 1 1
Object Rows
Insert anchor for each row
Figure Selection
Include figures
All figures with tags:
peaks
membrane
app
Loop Figure List
Revert
Help
Apply

Adding and Removing Properties

You select the object properties to be displayed in the generated report in the **Property Columns** pane. Depending on the object type, you can select any object property to add to the summary table.

To add a property to the summary table, select the property category from the property category box to the right of the **Property columns** field. As you select different property categories, notice that the field below the box changes to display the category's properties. The following figure shows `main` as the selected category.



To add a property to the table, select the category from the property category box, select a property in the properties list, and click **Add**. The property should appear in the **Property columns** field.

To remove a property from the table, select the property in the **Property columns** field and press **Delete**. The property name should disappear from the **Property columns** field.

Note After making a change in the Report Explorer interface, you must click **Apply** for the changes to take effect.

You can also specify your own properties by entering the name manually in the **Property columns** field. You must use proper Report Generator variable

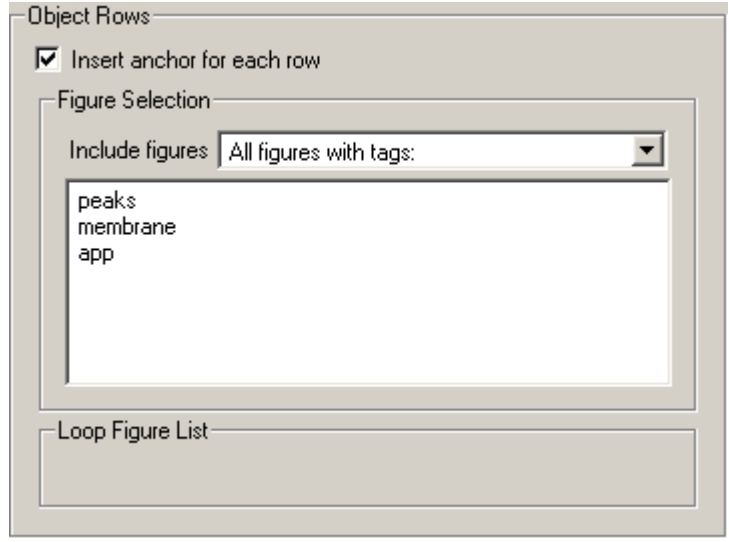
notation. For details, see "%<VariableName> Notation" on page 11-269 in the Text component reference page.

Setting Relative Column Widths

Using the **Relative column widths** field, you apply a relative column width to the summary table columns in the generated report. Column width is entered as a relative ratio. If omitted, column widths are determined automatically.

Setting Object Row Options

In the **Object Rows** pane, you configure options for the table rows, including anchor, filtering, and sorting options. Selecting the **Insert anchor for each row** option places an anchor in each table row in the generated report.



Using the **Include figures** list, you specify what objects are reported on in the summary table. For example, in the `figloop-tutorial` report template, the summary table components report on figure objects. If you select `All Figures with tags`, Report Generator reports on all figure objects that match the tags specified. If you select `Current figure only`, Report

Generator only reports on the selected figure. If you select `Visible figures`, Report Generator only reports on visible figure objects.

8

Building Custom Components

This chapter covers the following topics, which apply to both MATLAB and Simulink users.

Creating Components

The following topics are covered in this section:

Components are self-contained, modular elements that control the report generation process and insert elements into a report. A component is a MATLAB object (in the object-oriented programming sense).

Why Create Your Own?

In most cases, the components provided with Report Generator should be more than adequate to meet your needs. However, you may want to create your own components if you want to generate a report that cannot be easily created by components included with Report Generator. For example, you can create a component that inserts a corporate logo into your report, or you can create a component that plots data.

What's Involved in Creating Components

You can create custom components with the Component Creator, a tool for creating new components. The Component Creator creates a "skeleton" object, i.e., it sets up the raw structure of the component. It does not create all of the executable M-code.

There are two important steps to follow when creating and running a component:

1. Use the Component Creator to create the M-files that form the framework of the component. See "Component Description" on page 8-13.

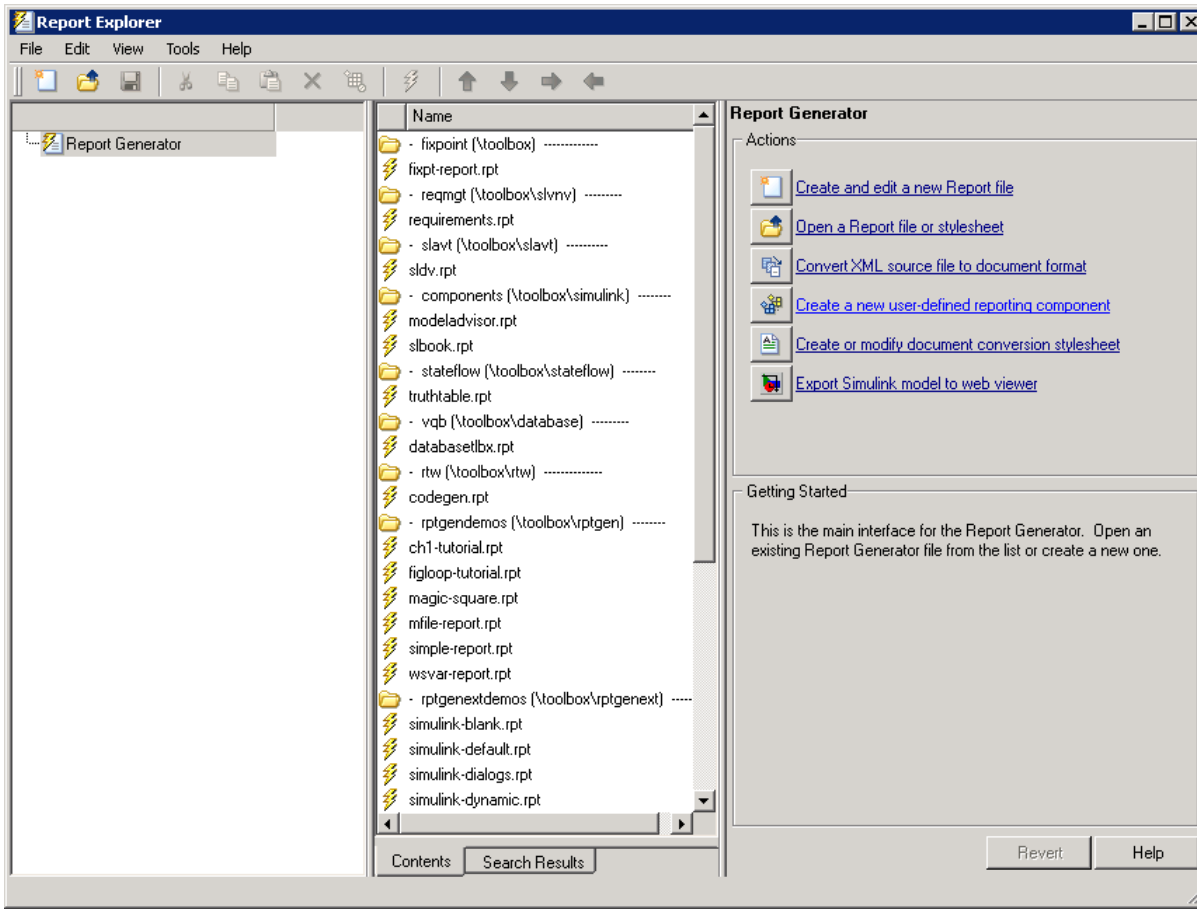
2. Edit `execute.m` to make the component executable. (This is an M-file that is generated when you create a component.) See "Editing Generated Files" on page 8-19.

Note You must restart MATLAB before using a newly created or rebuilt component.

Starting the Component Creator

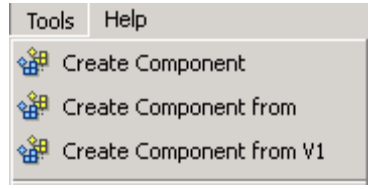
The following topics are covered in this section:

To begin the component creation process, click the **Create a new user-defined reporting component** link in the Report Explorer Properties pane on the right.



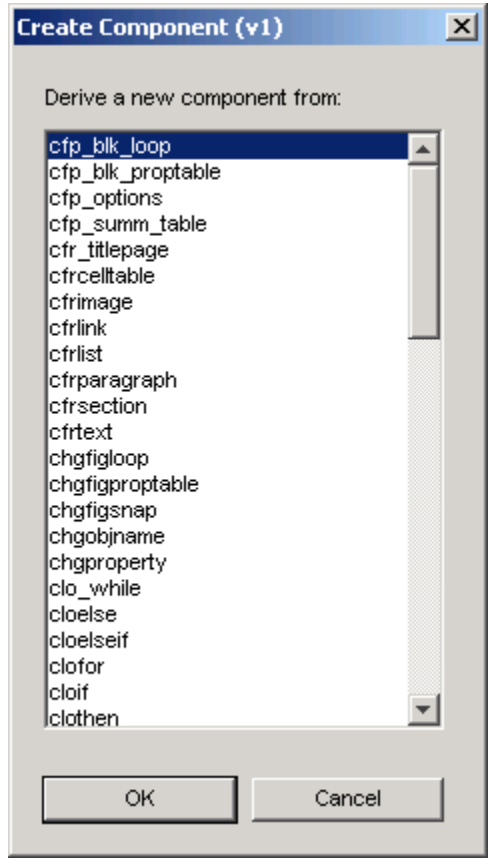
Two alternative methods to begin component creation are:

- Type `compwiz` at the command line.
- Select one of the component creation choices from the **Tools** menu in the Report Explorer. The **Tools** menu provides options for creating a new component, deriving a component by modifying an existing Version 2 component, or deriving a component by modifying an existing Version 1 component.



Modifying an Existing Component

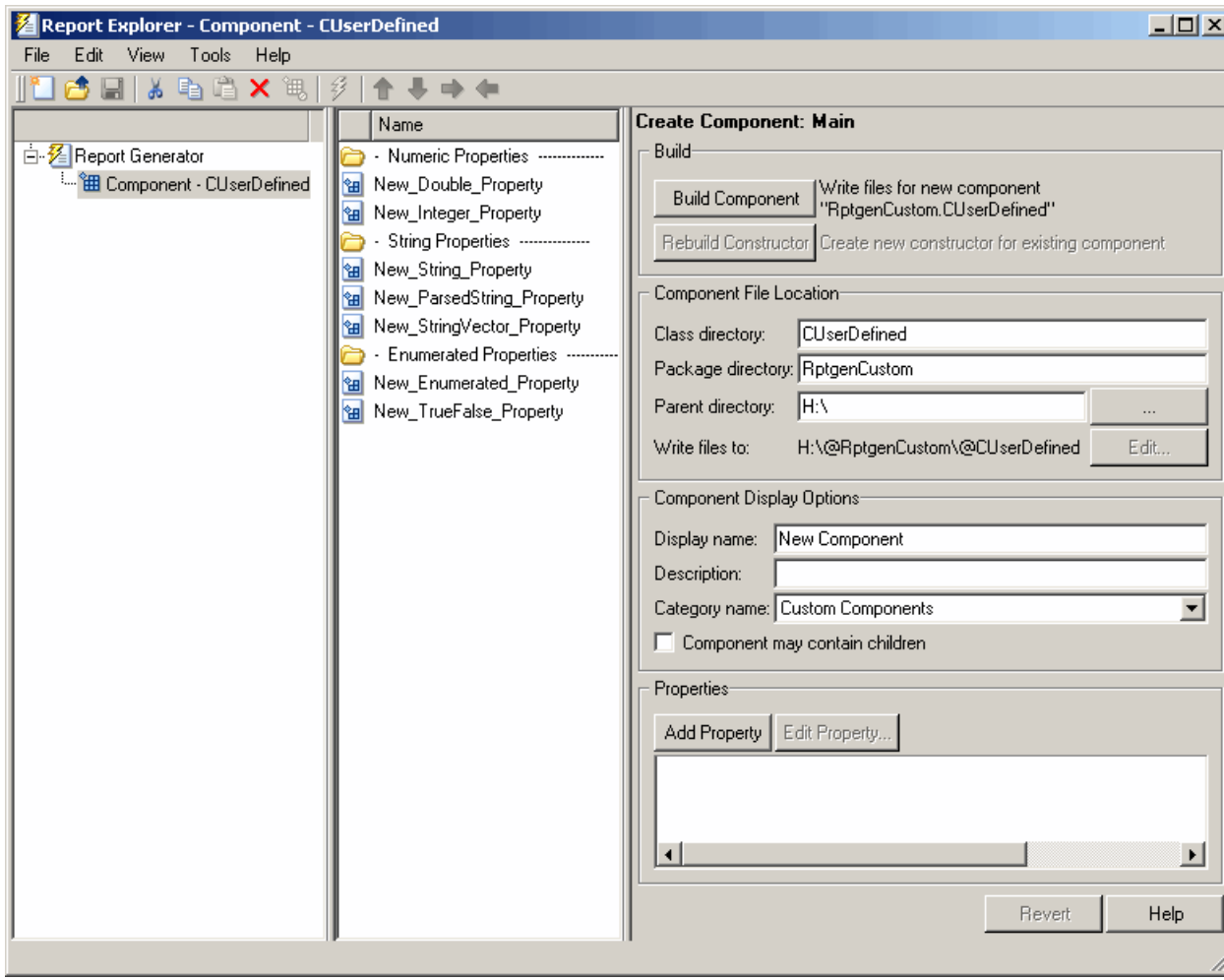
If you want to derive a component from an existing Version 1 or Version 2 component, the Report Explorer lists components from which you can choose a component to modify. The lists of components are similar to the following:



Double-click the name of any component appearing in either of these lists. The Component Creator GUI appears, showing the values and properties previously determined for this component. (See “Creating a New Component” on page 8-6 for details about the layout of the Component Creator GUI.)

Creating a New Component

If you choose to create a new component, the Component Creator GUI appears. The Component Creator is a three-pane GUI displayed within the framework of the Report Explorer.



Outline pane	Displays the structure of the component you are creating.
Options pane	Lists example properties that you can add to a component.
Properties pane	Specifies the behavior of component properties.

When you create a component, you also create several M-files, including `execute.m`. The default `execute.m` file for a new component inserts an

empty space into the report. Therefore, you must edit `execute.m` to run the component properly.

Defining a Component

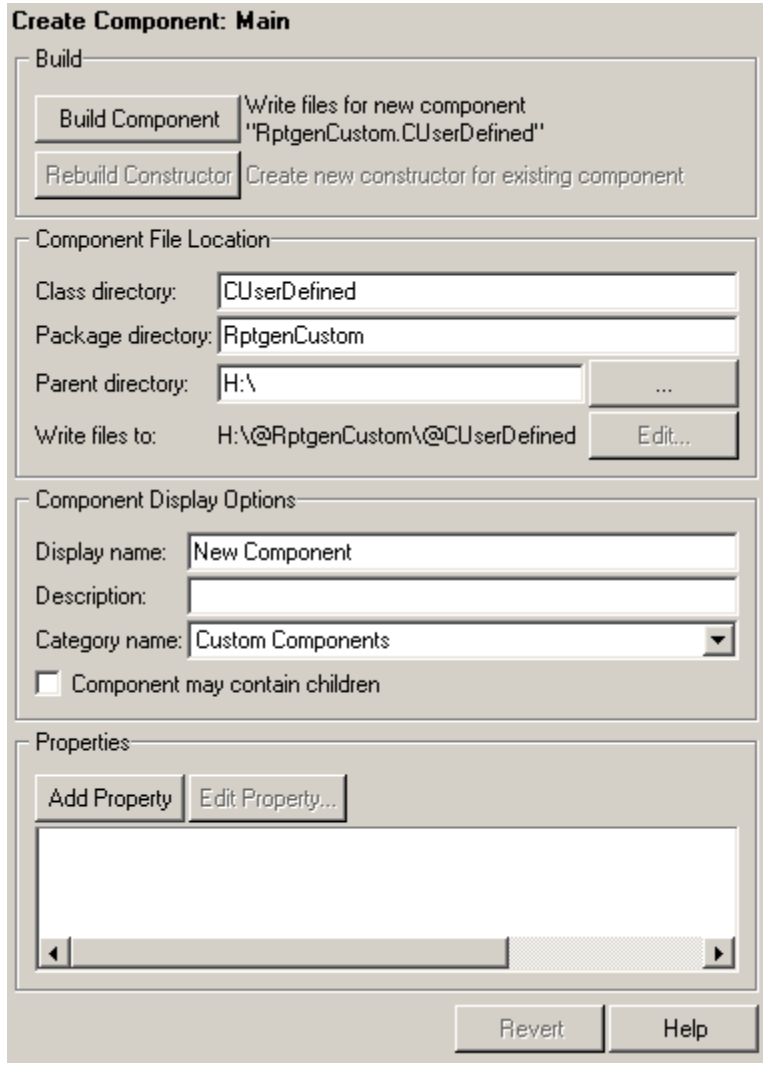
The following topics are covered in this section:

When you create a component, you have to decide the following:

- Where to put the directory that contains all the files for the component
- What class name to give the component
- What name will identify the component when it appears in the Report Explorer
- What description to give the component in the Report Explorer
- Whether the component can have child components

You enter this information in the Properties pane of the Component Creator.

The **Component File Location** and **Component Display Options** subpanes of the Properties pane allow you to indicate the location where you want to store a new component and how it is to be displayed.



Component File Location Data

You indicate the location where you want to store your component in the **Component File Location** subpane of the Properties pane. When the component is built, the information in this pane creates the directory structure `<parent>/@package_name/@class_name` and stores the component files here.

Class Directory Field

Assign a class name to your component. This name is used to create a subdirectory under the package directory field. (See "Package Directory Field" on page 8-11.) The build process places the component's files in this class directory. The class directory must be unique for each component in the package. By convention, component class names begin with an uppercase or lowercase letter c.

Package Directory Field

The set of files created during the build process (see "Building or Rebuilding a Component" on page 8-17) are stored in a directory under the name you enter in this field. The directory is located as a subdirectory of the specified package root directory.

Parent Directory Field

When you create a component, a set of files is created for the component. These files are located in the class directory you designated in the **Class directory** field. The class directory is in turn a subdirectory of the package directory, which itself is a subdirectory of the directory that you specify in the **Parent directory** field.

You specify the parent directory when you create a package for the first time.

You should choose a parent directory that is on the MATLAB path. Otherwise, it may be difficult for MATLAB to find your component.

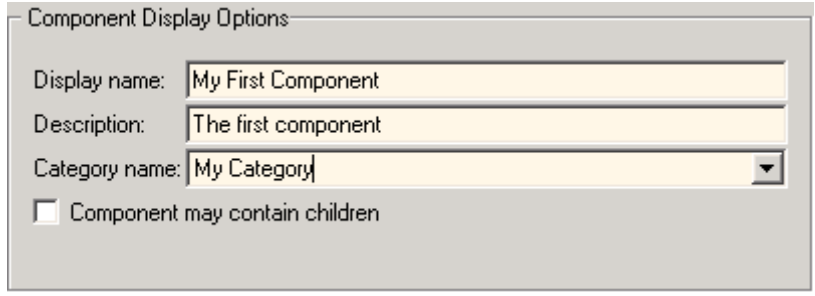
Component Display Options

You indicate how you want your component to appear in the Report Explorer by entering data in the **Component Display Options** pane of the Properties pane.

Display Name

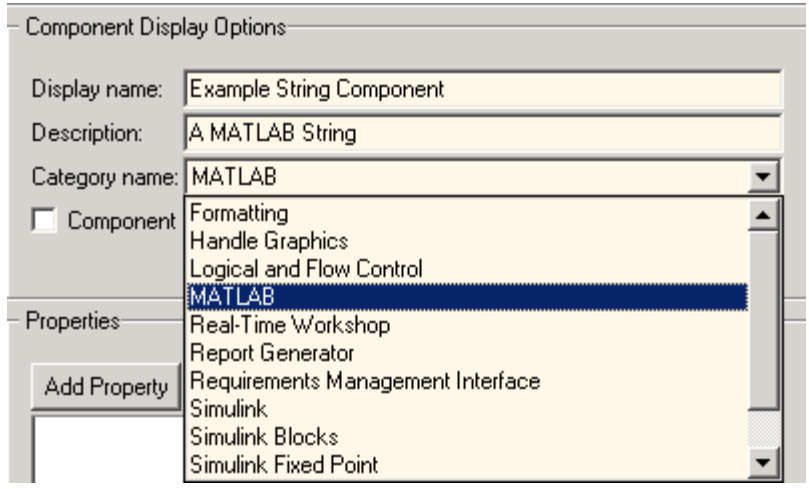
You need to specify a display name for your component. This component name is added to the list of components for its associated category. (See "Category Name" on page 8-12.) Component categories and component display names appear in the Options pane in the middle of the Report Explorer. Here is

an example of creating a component called `My First Component` under a category called `My Category`.



Category Name

When you create a component, you have to decide the category of components to which it will belong. This is the category under which the new component appears in the Options pane in the middle of the Report Explorer. The available choices are shown in the **Category name** list.



The `Category name` selection menu shows all available component categories such as Formatting, Handle Graphics, etc. To select an existing component category, click that component category on the menu.

To create a new component category, type the name for the category directly into the **Category name** field.

This category name is added to the list of category names shown in the Options pane in the middle of the Report Explorer.

Component Description

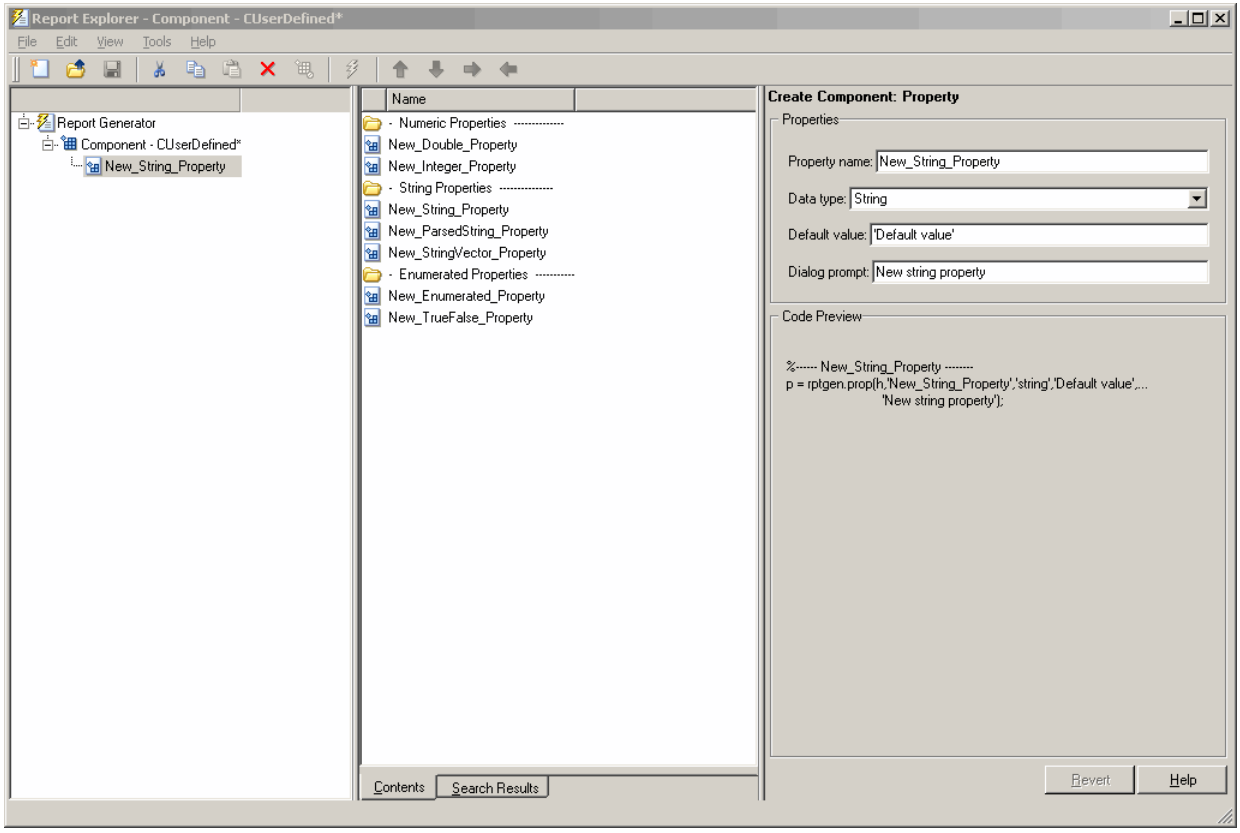
The description of the component can be anything you want. You should keep the description to one or two sentences. The description appears when you click the component name or category name in the **Options** pane in the middle of the Report Explorer.

Child Components

If you select the **Component may contain children** check box, your component can have child components. Child components appear indented under the component in the Report Explorer hierarchy. During execution, the component will usually run all child components and include their output in the report. (The actual behavior of child components is controlled by your entries in the `execute.m` file.)

Specifying Component Properties

Component properties determine how a component behaves and what information it inserts into a report. If you double-click a property name for a component shown in the Outline pane on the left in the Report Explorer, the Properties pane on the right displays the current values for the property. For example, the next figure displays the property values for `New_String_Property`.



By editing the values in the text fields, you can modify the information shown in this GUI to create new component properties that control the behavior of your component in ways that you specify.

Choose a Property Name

Edit the default **Property name** appropriately to create a name for your new property. A property name must be a valid MATLAB variable name (e.g., no spaces, etc.). Property names must be unique within a component.

Specify the Data Type

The data type is the type of data that is used by the attribute. For details, see "Data Types" in the MATLAB Programming documentation.

The `Data type` list contains the following options:

- `Double`
- `Enumeration`
- `Integer`
- `String`
- `String Vector`
- `%<Parsed String>` This data type allows you to include the value of a variable in the MATLAB workspace in a component. (See "%<VariableName> Notation" on page 11-269 for a discussion and example of the specification and use of this data type.)
- `True/False`

Specify the Default Value

Edit the **Default value** field to set a default value for the property. The default value needs to be compatible with the data type. For instance, if the data type is a MATLAB string, the default value must be enclosed in single quotation marks (`''`). If incompatibilities exist between the default value and the data type, the component may fail to build.

Specify the Dialog Prompt

This text appears next to the widget on the component's dialog box. It provides you with an indication of what the property does and what kind of effect it has on execution.

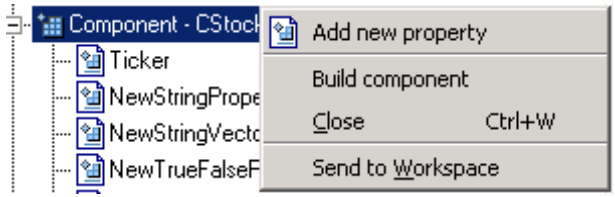
Note When the component is built, a colon is appended to your entry in the **Dialog prompt** field. Your entry appears in the Properties pane with the colon appended.

Adding Properties to a Component

The Component Creator provides several additional ways to add properties to a component, using properties from the properties library. Each library property provides an appropriate default value that you can modify to suit your requirements.

- **Add new property** context menu (Outline pane is on the left).

  Right-click the component name.

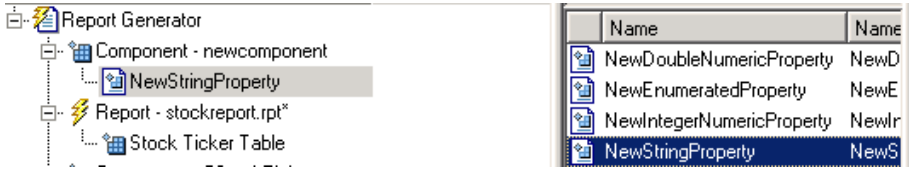


- **Add property** context menu (Options pane in the middle).

  Right-click the property name.

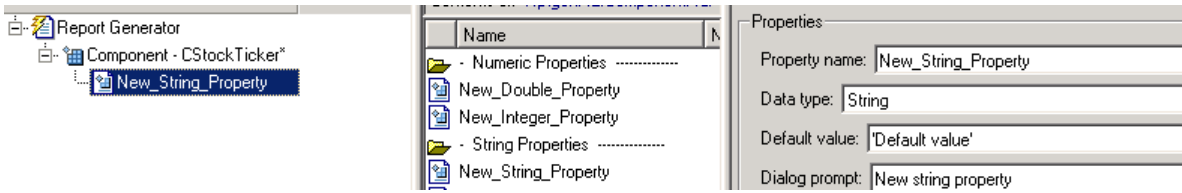
- Drag and drop from the Options pane in the middle.

  Left-click the name of a property in the Options pane and drag it on top of the name of a component in the Outline pane on the left.



- Double-click the property name in the Options pane in the middle.

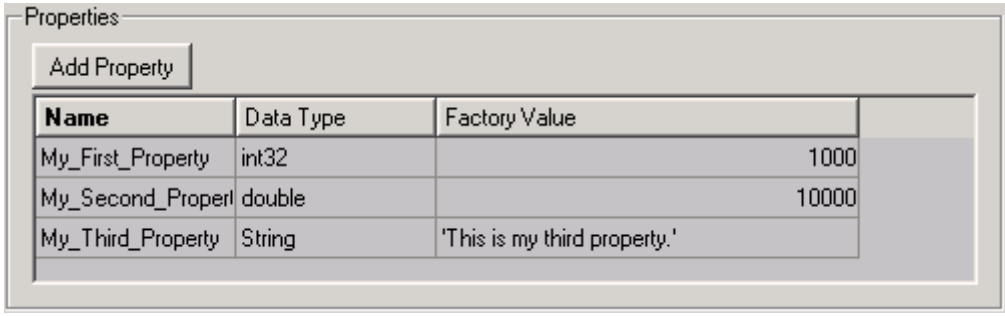
  The property is added to the component and property values are displayed in the Properties pane on the right.



- **Add Property** button (Property pane)

  The **Properties** pane within the Property pane allows you to add new properties to a component. Click **Add Property** to add an additional property.

  This pane additionally provides a display of the properties and values assigned to the currently selected component.



Building or Rebuilding a Component

The **Build** subpane of the Properties pane allows you to build a new component or rebuild an existing one.

Build Component Button

After you have entered all data required for defining your category and component, click **Build Component** to complete the process. The Build process creates all the files needed to define the component and stores them where indicated by the entries in the **Location** pane. Any existing files in this location will be overwritten.

Rebuild Constructor Button

Use this option when adding, removing, or changing properties of an existing component. This button becomes active only after you have previously created a component with the **Build Component** button. To activate **Rebuild Constructor**, set the **Package name** and **Class name** fields in the **Location** pane to the existing values for a defined component. If you select **Create component from** on the **Tools** menu and select one of the component choices presented to you, the fields are filled in automatically and the button becomes active.

In the other fields enter your changes to the component and **Rebuild Constructor** to make your changes effective. User-editable files in this location will not be overwritten.

Removing a Component

To remove a built component, find the class directory (`<root>/@package_name/@class_name`). If this is the only component in the package, you can delete the entire package. If there are other components you want to preserve, delete the class directory. Also, edit `<root>/@package_name/rptcomps2.xml` to remove the XML element that registers the component. It may also be necessary to remove the component's category.

Editing Generated Files

The following topics are covered in this section:

Editing M-files

Building a component creates several M-files in your MATLAB workspace. The MATLAB editor initially displays the three most significant files. You *must* edit `execute.m` to make the component perform as you expect. Editing of the other M-files is optional. The table below describes the significant M-files that you can edit for specific purposes.

Edit this M-file ...	To complete this task ...
`execute.m`	Specify the report output format and content.
`getdialog` `boxeschema.m`	Modify the characteristics of the properties dialog box for a created component.
`getOutlineString.m`	Change the string that describes the component in the hierarchy. This string is initially entered in the **Display name** field of the **Properties** pane on the **Create Component: Main** pane.

Editing the execute.m File

The `execute.m` M-file is called during report generation to produce the component's contribution to the report. After you have built the file during component creation, you must edit it to specify the explicit format and content of your output.

The generic form of the `execute` command is

```
out = execute(thisComp, parentDoc)
```

where:

- `thisComp` is a handle to the component that is being run.
- `parentDoc` is a handle to the current document being generated.
- `out` is a Document Object Model (DOM) node or string to be added to the report. (For more help with manipulating DOM nodes, see `xmlwrite`.)

Within the file one or more default lines of code shows each component property. Property values control how the component behaves during execution. Property values should not be changed or set during execution. Here is an example of a component property line within an `execute.m` file.

```
pstring = thisComp.NewStringProperty;  % New string property;
```

You can create these basic documentation elements using additional blocks of commented code:

- Tables
- Lists
- Text
- Figures

You can edit this code and add any needed additional code to the file to specify your output content. After you have processed your data, this code returns the component's formatted output.

See the following sections for more details.

Creating a Table. To create a table, replace the `Source` property value below with the name of your cell array or structure. (Type `help(rptgen.cfr_table)` at the MATLAB command line for additional information.)

```
out = execute(rptgen.cfr_table(...
'Source', tableSrc,...
'numHeaderRows',1,...
'TableTitle','Example Title'),...
 parentDoc);
```

Note For an example of an `execute.m` file that has been edited to produce tabular output, see "Editing execute.m" on page 8-29.

Creating a List. To create a list, replace the `Source` property value below with the name of your cell vector.

For more information about inserting lists into reports, type `help(rptgen.cfr_list)` at the MATLAB command line.

```
out = execute(rptgen.cfr_list(...
'Source', listSrc,...
'ListStyle','orderedlist',...
'ListTitle','Example List'),...
 parentDoc);
```

Creating Text. To create text, replace the `ParaText` property value below with your text string.

For more information about inserting paragraphs into reports, type `help(rptgen.cfr_paragraph)` at the MATLAB command line.

```
out = execute(rptgen.cfr_paragraph(...
'ParaText', paraSrc,...
 parentDoc);
```

Creating Figures. To create figures, set the `FigureHandle` property value with your figure.

For more information about inserting figures into reports, type `help(rptgen_hg.chg_fig_snap)` at the MATLAB command line.

```
figSrc = gcf;
out = execute(rptgen_hg.chg_fig_snap(...
'FigureHandle', figSrc,...
'Title', '',...
'isResizeFigure', 'manual',...
'PrintSize', [6 4],...
'PrintUnits', 'inches'),...
 parentDoc);
```

Child Components. An additional block of code shows how to run child components. The first line calls the `execute.m` method for all child components. The second line appends the result of the child component executions to the document.

```
childOut = thisComp.runChildren(parentDoc);
out = parentDoc.createDocumentFragment(out, childOut);
```

Changing the Outline String

The `getOutlineString` M-file returns the string used in the Report Explorer hierarchy. The generic form of the `getOutlineString` command

```
olstring = getOutlineString(thisComp)
```

returns a single-line string that displays important information about the component, where:

- `thisComp` is the component being described.
- `olstring` is the descriptive string.

This string appears in the Report Explorer hierarchy.

By default `getoutlinestring` returns the display name of the component. The string should be fewer than 32 characters long.

Customizing the String

You can customize the string to show additional information about the component, such as the state of its properties. The `truncatestring` function converts any data type into a single line string. The second argument is the return value if the data is empty. The third argument is the maximum allowed size of the resulting string.

```
cInfo = '';
pstring = rptgen.truncateString(thisComp.string,'<empty>',16);
```

The string typically uses a dash (`-`) as a separator between the name and additional component information.

```
if ~isempty(cInfo)
   olstring = [olstring, '-', cInfo];
end
```

Editing the Dialog Schema

The `getdialog boxeschema` M-file returns all information necessary to create a properties dialog box display. The format of the `getdialog boxeschema` command is as follows:

```
dlgstruct = getdialog boxeschema(thisComp, name)
```

where:

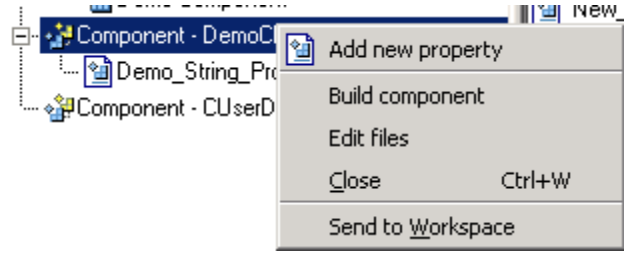
- `thisComp` is an instance of the component being edited.
- `name` is a string that can be passed to getdialog boxeschema to build a specific type of pane. Normally, `name` is empty in the Report Explorer.

The `getdialog boxeschema` file provides a comprehensive, extensively documented template with which you can modify most aspects of dialog box layout, including creation and placement of widgets, organization of widgets into panes, and creation of the top-level display within which the panes reside.

Caution Advanced MATLAB users are cautioned against modifying fields not explicitly mentioned in this file. These fields are undocumented and subject to change in future releases.

Editing Additional Files

There is an additional set of editable files that does not initially appear in the MATLAB editor. To access these files, right-click the component in the Outline pane on the left in the Report Explorer and select **Edit files** from the menu.



The contents of the MATLAB editor now expand to include the additional files listed in the following table.

M-File	Purpose
`getParentable.m`	Indicate whether or not a component can contain children.
`getDescription.m`	Edit to change a component description.
`getName.m`	Edit to change the display name.
`getType.m`	Edit to change the category name.
`rptcomps2.xml`	Change entries in the registry. Must remain consistent with entries in `getName.m` and `getType.m`.
`viewHelp.m`	Open the file in the Help viewer.
`_help.html`	Help file viewer.

getParentable.m

This file indicates whether or not a component can contain children. It returns the value `true` or `false`. For example, if the requirement for child components has changed, you can modify the value within the code as follows:

```
p = false;
```

getDescription.m

The description within this file corresponds to the entry you made in the **Description** field in the Component Creator. The following example shows

how to edit the `compDesc` string to change a component's description to `A demonstration component`:

```
compDesc = 'A demonstration component';
```

getName.m

The name within this file corresponds to the entry you made in the **Display name** field in the Component Creator. The following example shows how to edit the `compName` string to change a component's display name to `Demo Component`:

```
compName = 'Demo Component';
```

getType.m

The name within this file corresponds to the entry you made in the **Category name** field in the Component Creator. The following example shows how to edit the `compCategory` string to change a component's category name to `Custom Components`:

```
compCategory = 'Custom Components';
```

rptcomps2.xml

This file registers components with the Report Explorer and helps to build the list of available components. Information in this file must be consistent with the values entered in the `getName.m` and `getType.m` files. If you have changed values in either of these files, you must change the respective values in the `rptcomps2.xml` file accordingly. The Report Explorer displays the modified information after you have restarted MATLAB.

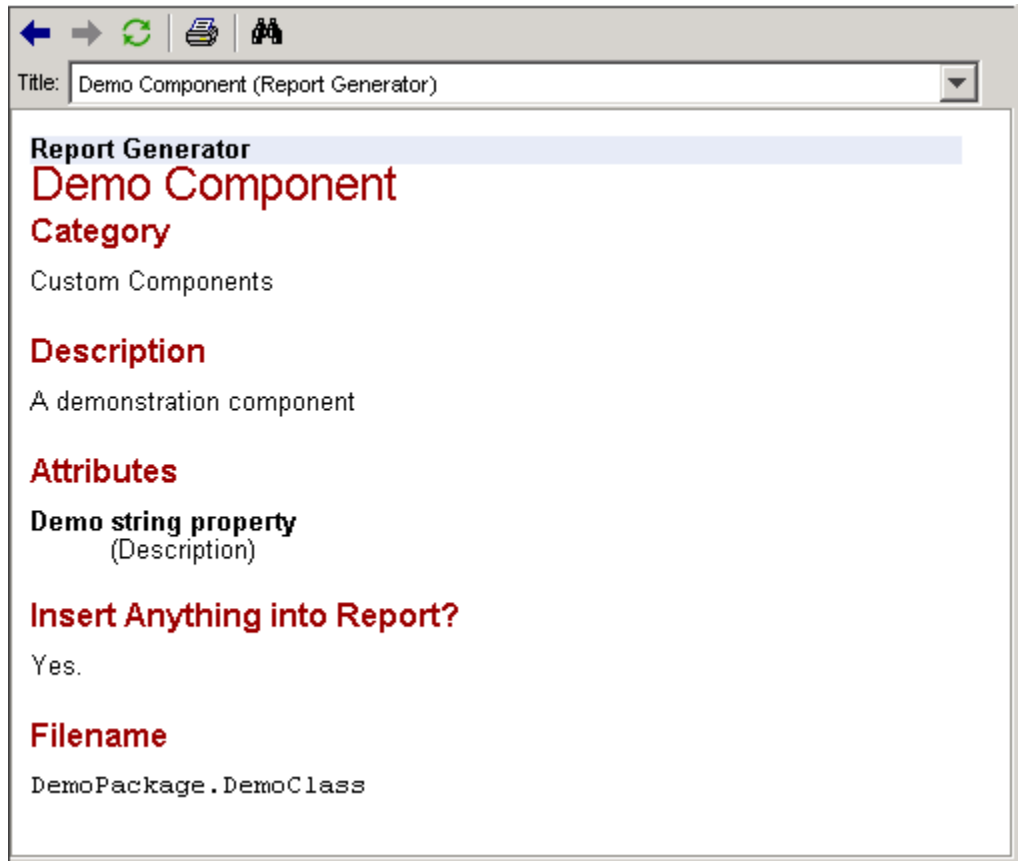
viewHelp.m

This file displays a help file for the component within the MATLAB Help browser. To display the help file, highlight the name of the component in the Report Explorer and click **Help**.

help.html

This file works in conjunction with `viewHelp.m` to display help information for a designated component.

Here is an example of a help file display for a user-created component.



Component Creation Examples

This section provides two examples showing the creation and use of component properties:

- The first example, “Creating and Using Component Properties” on page 8-27, provides an example of developing a component that fetches stock market data and displays the result in tabular format.
- The second example, “Report Displaying Two Tickers” on page 8-33, expands upon the first example, showing the use of multiple properties within a component.

Creating and Using Component Properties

To illustrate the creation and use of component properties, you will create a component named `Equity Values` that displays a table of stock market information in a report. To achieve the desired result, you will do the following:

- Give this component a string property named `Ticker`.
- Build the component.
- Edit the generated `execute.m` file.
- Display the final output.

Note This example requires Datafeed Toolbox.

Step 1: Create a Component

Start by creating a component named `Equity Values` in the class directory named `CStockTicker`. The remainder of the component information is arbitrary.

Step 2: Create a Property

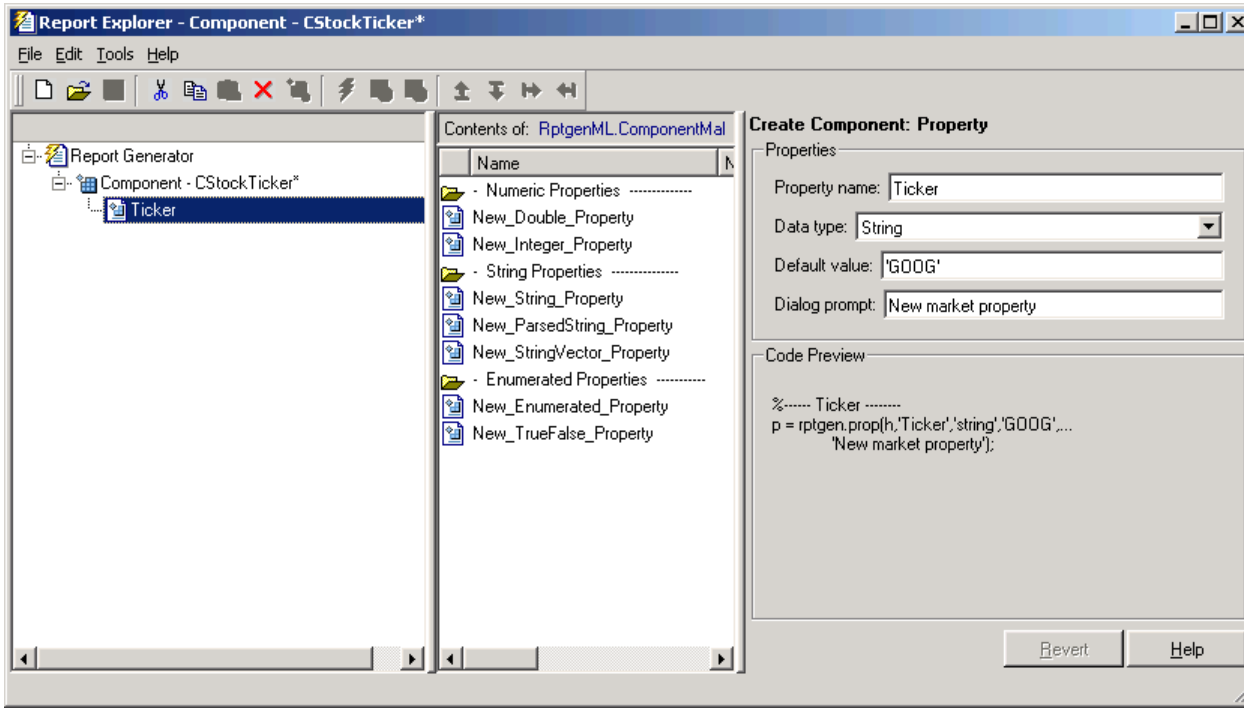
Now give the component one string property named `Ticker`. Start by double-clicking **New_String_Property** in the Options pane of the Component Creator. This adds a property to the component you are constructing. Then

edit **Property name** appropriately. A property name must be a valid MATLAB variable name (e.g., no spaces, etc.).

Specify the Data Type. The data type is the type of data that is used by the attribute. Your `Ticker` property is a string value, so there is no need to change the default.

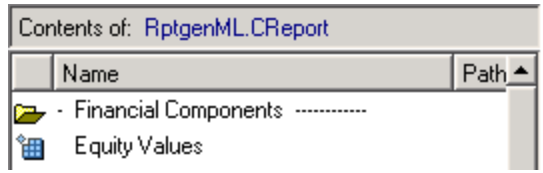
Specify the Default Value. Edit the **Default value** field to set a default value for the property. Since for this example you will obtain ticker values for the security Google, set the **Default value** to `'GOOG'`. (Note that since this is a MATLAB expression, the single quotation marks are required to make this a string value.)

The GUI with these property values entered appears below. Note that in the **Code Preview** pane the code has been modified to reflect the settings you entered above.



Step 3: Build the Component

Click **Build** on the Component Creator GUI to create the new component. The Equity Values component is now displayed among the choices in the Options pane in the middle of the Report Explorer.



Step 4: Initialize the New Component

Building a component creates a number of M-files. Among the created files is `execute.m`, which runs the component when the report is generated. The default `execute.m` file created by the Component Creator inserts nothing into the report. You must edit `execute.m` to make the component perform as you expect. You can optionally edit other generated files, such as `getOutlineString.m` and `getdialog boxeschema.m` to provide further customization of your output.

Editing execute.m. When you create a component, Report Generator creates an `@<compfilename>` directory; in this example the created directory is `@CStockTicker`. In this directory, open the `execute.m` file, which runs the component when the report is generated. This method inserts into the report the information that the component creates when the report is generated.

In this example you want the component to obtain information about Google and produce output for your report in tabular format. An edited version of an `execute.m` file that accomplishes this appears below. The MATLAB `datestr` function converts the date and time information obtained from Yahoo into a more user-friendly format.

```
function out=execute(thisComp,parentDoc,varargin)

stockQuote = fetch(yahoo, thisComp.Ticker);
stockQuote.Date = datestr(stockQuote.Date,1);
stockQuote.Time = datestr(stockQuote.Time,13);
out = execute(rptgen.cfr_table(...
     'Source', stockQuote,...
```

```
 'numHeaderRows', 0,...
 'TableTitle', 'Stock Market Pricing Data'),...
parentDoc);
```

Here's an example of the report output produced using this version of `execute.m`.

Table 1. Stock Market Pricing Data

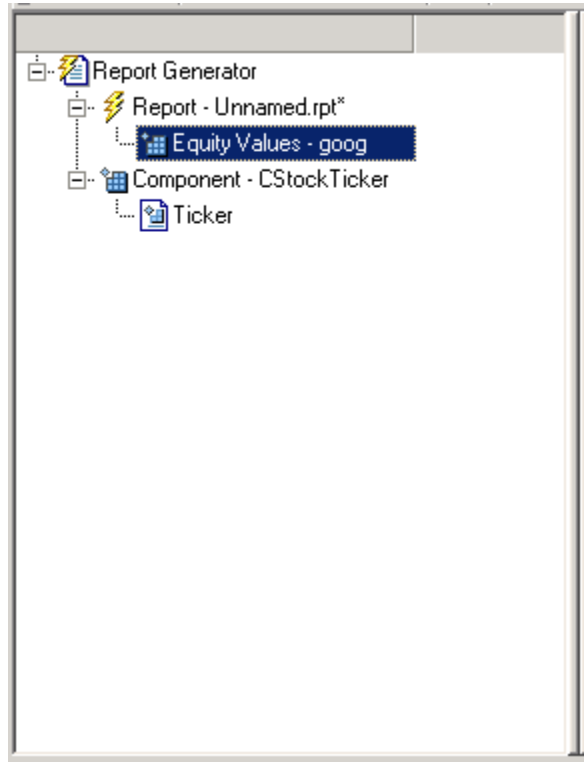
Symbol	GOOG
Last	185.25
Date	15-Nov-2004
Time	15:20:00
Change	3.25
Open	180.45
High	188.32
Low	178.75
Volume	10651060

Editing getOutlineString.m. The following version of `getOutlineString.m` has been edited to append the symbol for the security to the display name:

```
function olstring=getOutlineString(thisComp)

olstring = [getName(thisComp),' - ',thisComp.Ticker];
```

The following figure shows the result of this file modification.



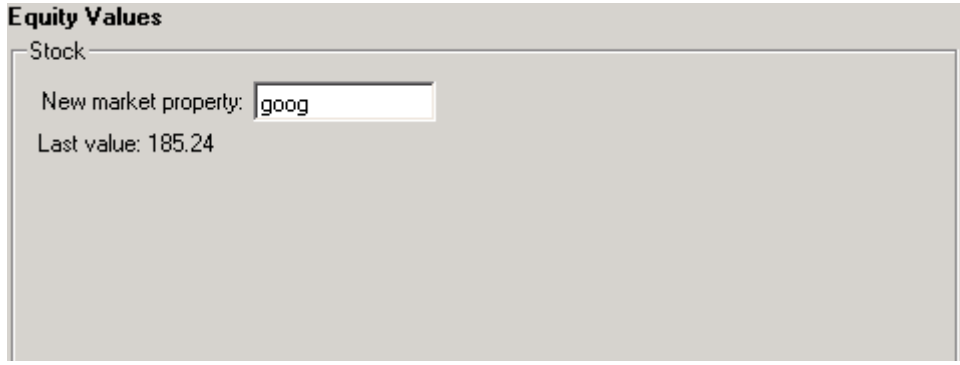
Editing getdialog boxeschema.m. You can modify the `getdialog boxeschema.m` file to change the appearance of the properties display in numerous ways. The edited version that follows displays the last quoted price for the security as part of the properties display. Some additional edits to improve the appearance of the graphical output have also been made.

```
function dlgStruct = getdialog boxeschema(thisComp, name)

try
    currQuote = fetch(yahoo, thisComp.Ticker);
    quoteStr = sprintf('Last value: %g', currQuote.Last);
catch
    quoteStr = sprintf('Warning: "%s" is not a valid symbol.',...
    thisComp.Ticker);
end
```

```
dlgStruct = thisComp.dlgMain(name,...
    thisComp.dlgContainer({
      thisComp.dlgWidget('Ticker',...
        'DialogRefresh',true,...
        'RowSpan',[1 1],'ColSpan',[1 1]);
      thisComp.dlgText(quoteStr,...
        'RowSpan',[2 2],'ColSpan',[1 1]);
    },'Stock',...
      'LayoutGrid',[3 2],...
      'RowStretch',[0 0 1],...
      'ColStretch',[0 1]));
```

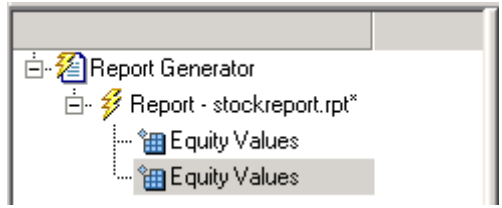
Here is the resulting display from the changes made above.



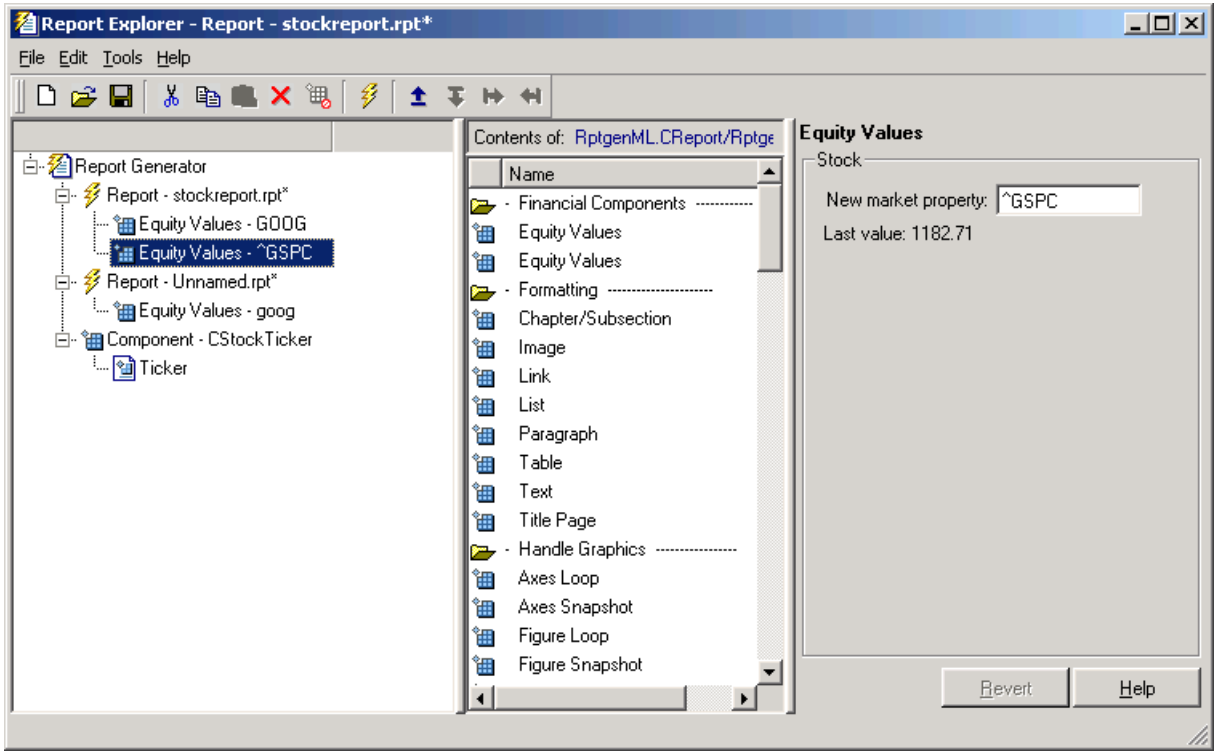
If you enter an unrecognized value into the **New market property** field, a warning message appears.

Report Displaying Two Tickers

Create a new report and save it as `stockreport.rpt`. Add two of your custom components to the report.



Change the ticker property of the second component to `'^GSPC'` (S&P 500 index) by editing the entry in the **New marker property** field.



Run the report. The report provides two tables of data, one for Google and another for the S&P 500 index. The report showing the resulting tables appears as follows.

Table 1. Stock Market Pricing Data

Symbol	GOOG
Last	185.25
Date	15-Nov-2004
Time	15:20:00
Change	3.25
Open	180.45
High	188.32
Low	178.75
Volume	10651060

Table 2. Stock Market Pricing Data

Symbol	^GSPC
Last	1183.13
Date	15-Nov-2004
Time	15:35:00
Change	-1.04
Open	1183.81
High	1184.48
Low	1179.85
Volume	1503245440

9

Creating and Using Customized Styles

This chapter covers the following topics, which apply to both MATLAB and Simulink users.

Stylesheets: An Overview

The following topics are covered in this section:

A stylesheet specifies the formatting and display of a document. Report Generator uses stylesheets to control settings for documents of the following kinds:

HTML	Uses stylesheets for either single- or multiple-page documents
PDF	Uses Formatting Object (FO) stylesheet
RTF, Word	Uses Document Style Semantics and Specification Language (DSSSL) stylesheet

How Do Stylesheets Fit into the Report Generator Process?

Whenever Report Generator creates a report, it first generates a DocBook XML source file, which it then converts to the formatted document. This conversion process uses a stylesheet. If you want to generate the same report in a different format, you can convert the XML document using a different or modified stylesheet.

When Do You Need to Edit a Stylesheet?

In many cases, the stylesheets provided with Report Generator are adequate to meet your needs. Nonetheless, you may want to modify the built-in styles to meet special requirements.

For example, suppose your company requires a certain size font for the title of any reports that are part of the company database, and this specification is not part of the built-in stylesheet that otherwise matches your requirements. In this case, you can edit the stylesheet you are using to match this special requirement. You can also create a new stylesheet.

Working with Stylesheets

The following sections describe how to use and edit a built-in Report Generator stylesheet. Topics covered are:

Opening the Stylesheet Editor

To begin the stylesheet editing process, open the Stylesheet Editor in any of the following ways:

- From the menu bar in the Report Explorer, select **Tools > Edit stylesheet**.
- From the Outline pane on the left in the Report Explorer, right-click **Report Generator** and select **Edit stylesheet**.
- From the Outline pane on the left in the Report Explorer, right-click a report (if one is displayed) and select **Edit stylesheet**.
- From the Properties pane on the right in the Report Explorer, click **Create or modify document conversion stylesheet**.
- At the command line in MATLAB, type `rpteditstyle`.

Editing Stylesheet Properties

To edit properties of an existing stylesheet:

1. Open the stylesheet for editing in any of the following ways:

- Drag a stylesheet from the Options pane in the middle to the Stylesheet Editor.
- Select the stylesheet from the list displayed in the Options pane in the middle, and click **Edit stylesheet** in the Properties pane on the right.
- Double-click the name of the stylesheet.

The Properties pane on the right now contains editable fields and a list of the customizations in the selected stylesheet.

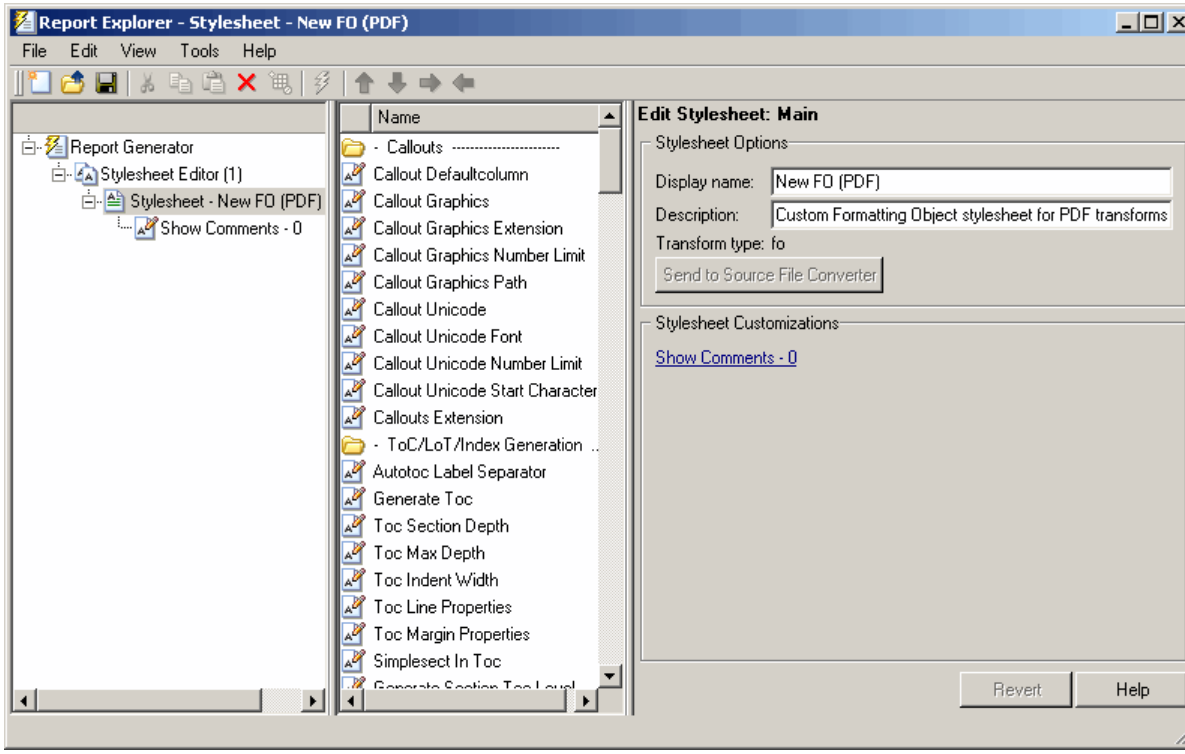
2 Add to or change the text in the property fields.

After you edit the properties for the stylesheet, you may want to see "Editing Style Data Items" on page 9-19 for details on the styles you want to customize and for details about editing data items.

Sample Stylesheet Properties

In the following illustration:

- The Stylesheet Editor is open and appears in the Outline pane on the left.
- The default print stylesheet is of type `fo` and is open in the Stylesheet Editor and contains customizations.
- The Options pane in the middle lists all styles that are available for the current stylesheet.
- The Properties pane on the right contains:
  - Editable fields describing the selected stylesheet
  - A list of any customizations in the selected stylesheet
  - Transform type
  - A **Send to Source File Converter** button



Saving a Stylesheet

To save a stylesheet:

1. In the Outline pane on the left, select the stylesheet that you want to save.
2. Do one of the following:
   - Select **File > Save** from the menu bar.
   - Select **File > Save As** from the menu bar.
   - Right-click and select **Save** from the shortcut menu.
   - Right-click and select **Save As** from the shortcut menu.

If you choose **Save**, the Stylesheet Editor saves the file with the current file name in its current location. You cannot overwrite the built-in stylesheets; instead use **Save As** and give the stylesheet a new name.

If the file has not yet been saved, the Stylesheet Editor prompts you for a file name and a location for the file.

Note By convention, files containing Report Generator stylesheets have `.rgs` as their file name extension.

Typically you should choose a location that is in your MATLAB path, and the file name must be unique in the MATLAB path. If the file is not saved to your MATLAB path, the stylesheet does not appear in the Report Explorer.

If you choose **Save As**, the Stylesheet Editor prompts you for a file name and location.

You must save a stylesheet before you can use it for converting a source file or associating with a report.

Using Unique File Names. Report Generator identifies stylesheets internally by file name without any path information. Thus, stylesheet file names must be unique across all paths. Repeated file names may cause certain stylesheets to be ignored upon generation.

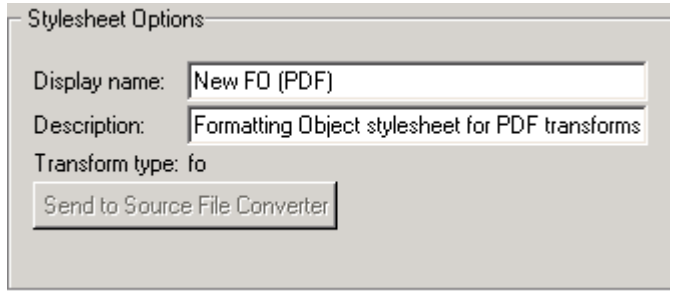
For example, if you have a stylesheet `~/project1/largetype.rgs`, you cannot also have a stylesheet `~/project2/largetype.rgs`. If you try to save a file with a duplicate file name, Report Generator prompts you to change the file name.

Converting a Report Using a New Stylesheet

Use a modified or newly created stylesheet just as you would use a built-in stylesheet. Keep in mind that you do not need to generate a new report to make the changes in styles; you can skip the generation step and go directly to the conversion step.

For example, click **Edit** in the **Convert Source File** area of the Report Explorer if you want to edit a stylesheet before converting the report.

You can also use a customized stylesheet for conversion by clicking **Send to File Converter** in the Stylesheet Editor, as shown.



See “Using the Report Explorer to Convert XML Documents” on page 5-10 for information about converting XML file source files that are generated by Report Generator.

Closing the Stylesheet Editor

Close the Stylesheet Editor in either of the following ways:

- Right-click the **Stylesheet Editor** in the Outline pane on the left and select **Close**.
- Select **Stylesheet Editor** from the Outline pane on the left and press **Ctrl+W**.

Deleting a Stylesheet

To delete a stylesheet that you created:

1. Select the Stylesheet Editor in the Outline pane on the left.
2. Select the stylesheet to delete from the Stylesheet Editor Options pane in the middle.
3. Click **Delete stylesheet** in the stylesheet’s Properties pane on the right.

Note You cannot delete the built-in stylesheets.

Deletions or renaming of stylesheets from the file system may not appear in the Options pane until you restart MATLAB.

Deleting a Customized Style

To delete a customized data item:

1. Right-click the customized data item in the Outline pane on the left.
2. Select **Delete**.

Creating a New Stylesheet

To create a new stylesheet:

1. Open the Stylesheet Editor.
2. In the Properties pane on the right, do one of the following:
   - Click **New HTML** to create a new stylesheet for HTML reports.
   - Click **New multi-page HTML** to create a new stylesheet for HTML reports with more than one page.
   - Click **New FO (PDF)** to create a new stylesheet for PDF reports.
   - Click **New DSSSL (RTF)** to create a new stylesheet for RTF reports.

   The new stylesheet appears in the Outline pane on the left.
3. In the Properties pane on the right, modify the properties for the stylesheet as needed.
4. Add styles to the new stylesheet:
   a. Drag the style you want to add from the Options pane in the middle to the stylesheet in the Outline pane on the left.
   b. In the Properties pane on the right, edit the data items for the selected style.

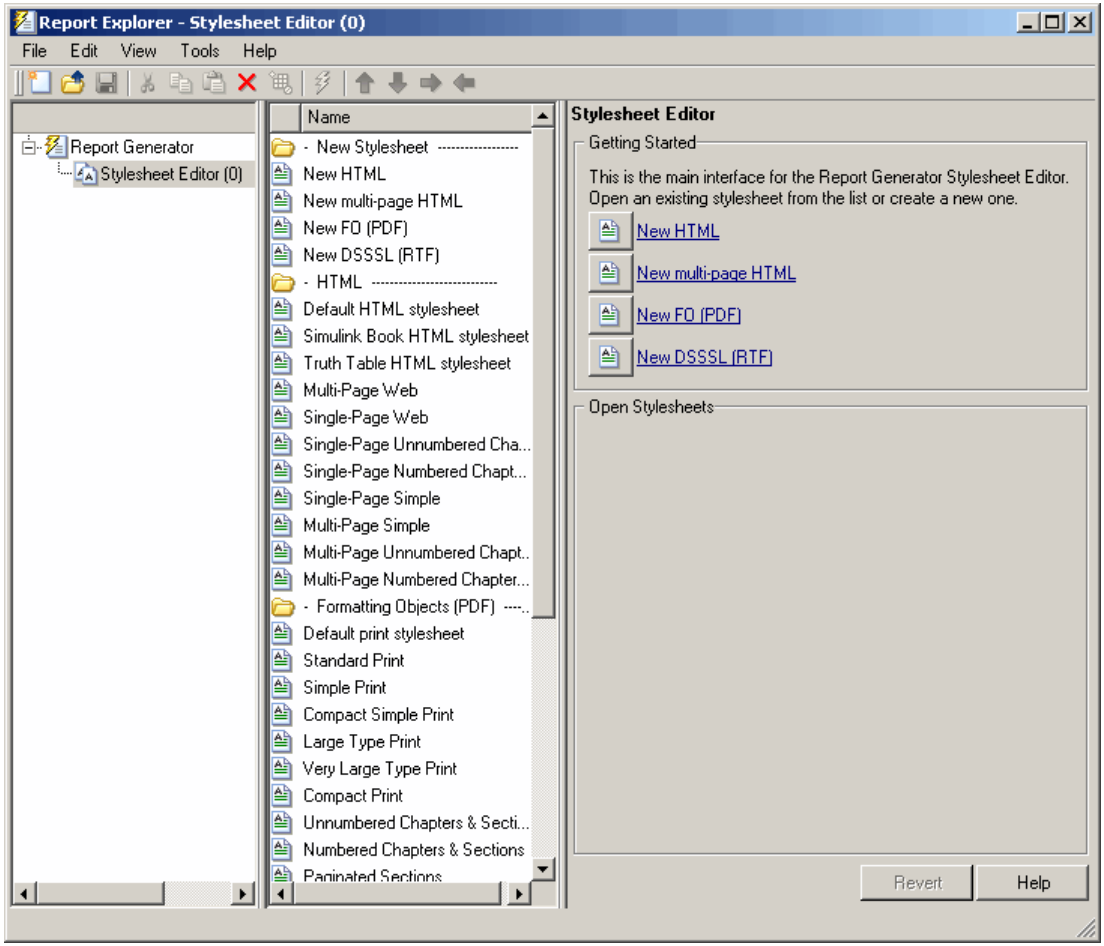
5 Save the stylesheet. For information about how to save a stylesheet, see “Saving a Stylesheet” on page 9-6.

Understanding the Stylesheet Graphical User Interface

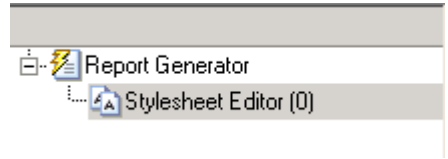
The following topics are covered in this section:

Stylesheet Editor

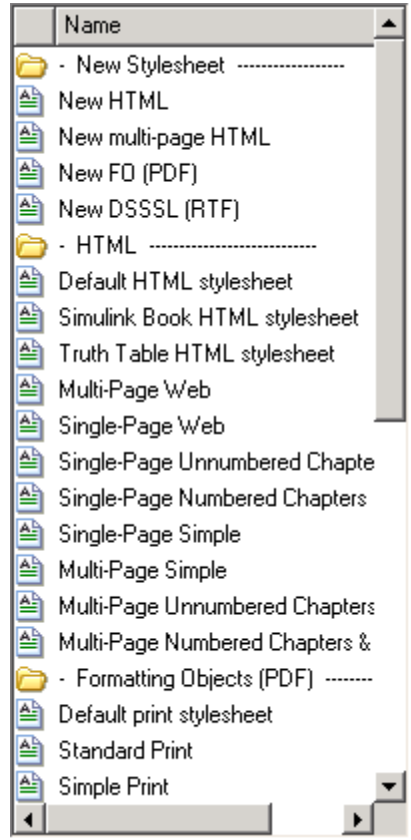
When you start the Stylesheet Editor, the Report Explorer opens as shown.


Report Explorer - Stylesheet Editor (0)
File Edit View Tools Help
Report Generator
Stylesheet Editor (0)
Name
- New Stylesheet ----------
New HTML
New multi-page HTML
New FO (PDF)
New DSSSL (RTF)
- HTML ----------
Default HTML stylesheet
Simulink Book HTML stylesheet
Truth Table HTML stylesheet
Multi-Page Web
Single-Page Web
Single-Page Unnumbered Cha...
Single-Page Numbered Chapt...
Single-Page Simple
Multi-Page Simple
Multi-Page Unnumbered Chapt...
Multi-Page Numbered Chapter...
- Formatting Objects (PDF) ----...
Default print stylesheet
Standard Print
Simple Print
Compact Simple Print
Large Type Print
Very Large Type Print
Compact Print
Unnumbered Chapters & Secti...
Numbered Chapters & Sections
Paginated Sections
Stylesheet Editor
Getting Started
This is the main interface for the Report Generator Stylesheet Editor.
Open an existing stylesheet from the list or create a new one.
New HTML
New multi-page HTML
New FO (PDF)
New DSSSL (RTF)
Open Stylesheets
Revert
Help

- The Outline pane on the left contains the Stylesheet Editor.



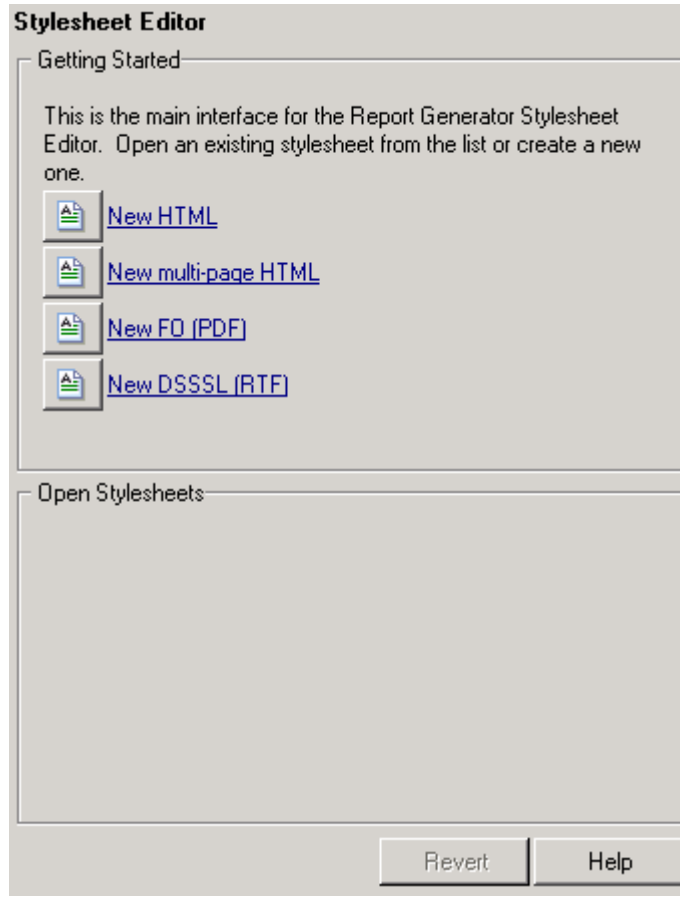
- The Options pane in the middle lists the available stylesheets.



Note By default, the folders in the Options pane in the middle are open. You might want to close them to see all categories at once. Whenever you are in the Options pane, double-click a folder to close it.

Alternatively, you can right-click each folder and select **Hide/show category**.

The Properties pane on the right shows the available options for the Stylesheet Editor (which is currently selected in the Outline pane on the left).



Tasks that you can accomplish using the Stylesheet Editor are listed in the following table.

Task	Pane to Use	Action
Create a new stylesheet.	Properties	Click the link that corresponds to the kind of stylesheet you want to create.
Open a custom stylesheet that has already been created.	Properties	Click the name of the stylesheet, which should appear in the **Open Stylesheets** area.
Select a stylesheet to use when converting an XML source file.	Option	Select a stylesheet.
View a list of customized styles in a stylesheet.	Outline	Expand any open stylesheet.
View a list of styles in a stylesheet.	Outline or Options	Double-click the stylesheet.
View a list of stylesheets available for editing in each category.	Options	Double-click the folder that corresponds to the kind of output you want (that is, HTML, PDF, RTF, or Word).
View stylesheets that are currently open.	Outline	Expand the Stylesheet Editor.

Main Stylesheet Properties

When you open a stylesheet, the Report Explorer displays the following information:

- The Outline pane on the left shows the name of the current stylesheet and any customized styles that it contains.
- The Options pane in the middle shows folders containing all the style data items available for the current stylesheet.

- The Properties pane on the right displays **Stylesheet Editor: Main** and shows the properties of the stylesheet, such as name and description, and customizations, if any, that belong to the stylesheet.

From the **Stylesheet Editor: Main** you can perform the following tasks.

Task	Pane to Use	Action
Change the name or description of the current stylesheet.	Properties	Edit the text in the **Display Name** or **Description** field.
Convert an XML source file using the current stylesheet.	Properties	Click **Send to Source File Converter** in the Properties pane.
Edit style data that is already customized.	Properties	Click the style data item, which is displayed in the **Stylesheet Customizations** area.
Open a style data item for editing or viewing.	Options	Double-click the data item that you want to edit.
View a list of style data that is already customized.	Outline	Expand the stylesheet.

Data Properties Pane

When you open a style data item, the Report Explorer appears as follows:

- The Outline pane on the left shows the name of the current style data item inside its stylesheet.
- The Options pane in the middle shows a list of all available stylesheet data items.
- The Properties pane on the right displays **Stylesheet Editor: Data** and shows the following.

- The value of the data item is in a pane labeled **Value** or **Value (XML)**, depending on the current edit mode for the data item.
- To the right of the value is the **Edit as** toggle button. You can edit the value using simple or advanced edit mode.
- The **Preview** pane contains a fragment of the stylesheet that specifies the data item. You cannot edit this pane.
- The **Help** pane contains information about the data item. You cannot edit this information.

Simple and Advanced Edit Modes

You edit values for most data items in PDF and HTML stylesheets in either *simple edit mode* or *advanced edit mode*. You edit values for RTF stylesheets only in simple edit mode.

- Simple edit mode means that you are editing a simple string that corresponds to the data in the stylesheet. This string appears in the field to the right of the **Value** label. For some values you can use a drop-down list to change the value instead of typing in text.
- Advanced edit mode means that you are editing the XML code directly. To switch from simple to advanced edit mode, click **Edit as XML**.

The user interface is set to simple edit mode when the data item is displayed in a pane labeled **Value** and to advanced edit mode when the data item is displayed in a pane labeled **Value (XML)**.

Note To modify content for headers and footers you edit stylesheet cells, which do not appear in either simple or advanced mode. See “Stylesheet Cells” on page 9-30 for details.

Working with Data Items in the Stylesheet Editor

Use the Properties pane on the right to modify the values for each data item that you want to customize. You use different methods of editing the stylesheet depending on the kind of data item you are editing.

Stylesheet Editing Methods

Type of Data	How to Edit	For More Details
Attribute sets	Type text in the **Inherit Attribute sets** field.	See "Attribute Sets" on page 9-27.
Boolean values	Use simple or advanced edit mode according to whether you want to edit a string (simple) or edit the XML directly (advanced).	See "Boolean and Enumerated Values" on page 9-20.
Enumerated values		
Strings		See "Strings" on page 9-22.
Title Placement Properties	Use drop-down menus to indicate whether title should go before or after the figure or table.	See "Title Placement Properties" on page 9-26.
TOC Generation Properties	Use check boxes to indicate which components of the document should have which level of TOC.	See "TOC Generation Properties" on page 9-26.
XML Expressions	Use simple or advanced edit mode according to whether you want to edit a string (simple) or edit the XML directly (advanced).	See "XML Expressions" on page 9-23.
Stylesheet cells	Use templates to specify XML code that defines the content for a header or footer.	See "What Is a Stylesheet Cell?" on page 9-30 and "Using Templates to Add Content to Headers and Footers" on page 9-35.

Editing Style Data Items

The styles that the file converter uses for a report are specified in a stylesheet as values for data items such as parameters and attributes. The data items are of several types. Each type requires a different way of editing.

See “Working with Data Items in the Stylesheet Editor” on page 9-18 for an overview of editing data items.

See the following sections for details about working with each particular type of data item:

- “Boolean and Enumerated Values” on page 9-20
- “Strings” on page 9-22
- “XML Expressions” on page 9-23
- “TOC Generation Properties” on page 9-26
- “Title Placement Properties” on page 9-26
- “Attribute Sets” on page 9-27
- “Attributes” on page 9-28
- “Varpair Values” on page 9-29
- “Cell Groups” on page 9-29
- “Stylesheet Cells” on page 9-30

Note See the **Help** area at the bottom of the Properties pane on the right for a description of a specific data item that you are editing.

Boolean and Enumerated Values

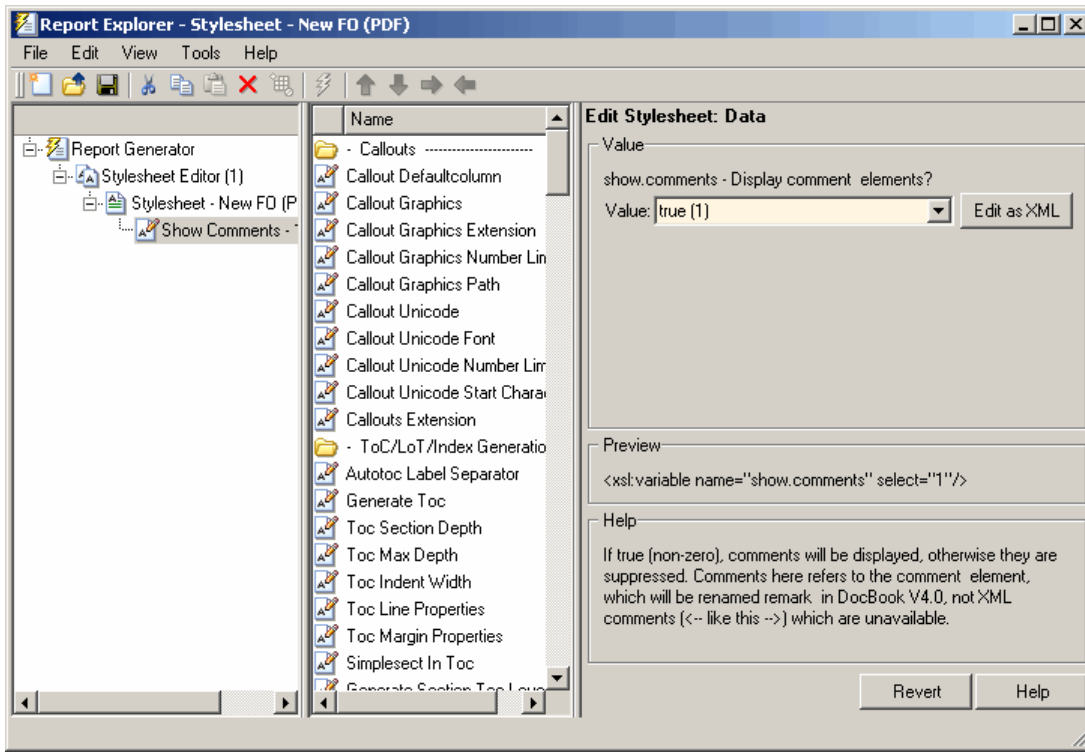
For most stylesheets, you can use either simple or advanced edit mode to edit data items of Boolean type and to edit data items that have finite enumerated values.

To use simple edit mode, set the Properties pane on the right to display the string value in a pane labeled **Value**.

Note If the pane is labeled **Value (XML)** you are in advanced mode. Click **Edit as String** to switch to simple edit mode.

To use advanced edit mode, click **Edit as XML**. In advanced mode, the Properties pane on the right displays the data item as **Value (XML)**.

Here is an example of the Properties pane for a data item in the **Default print stylesheet**, namely the **Show Comments** data item.



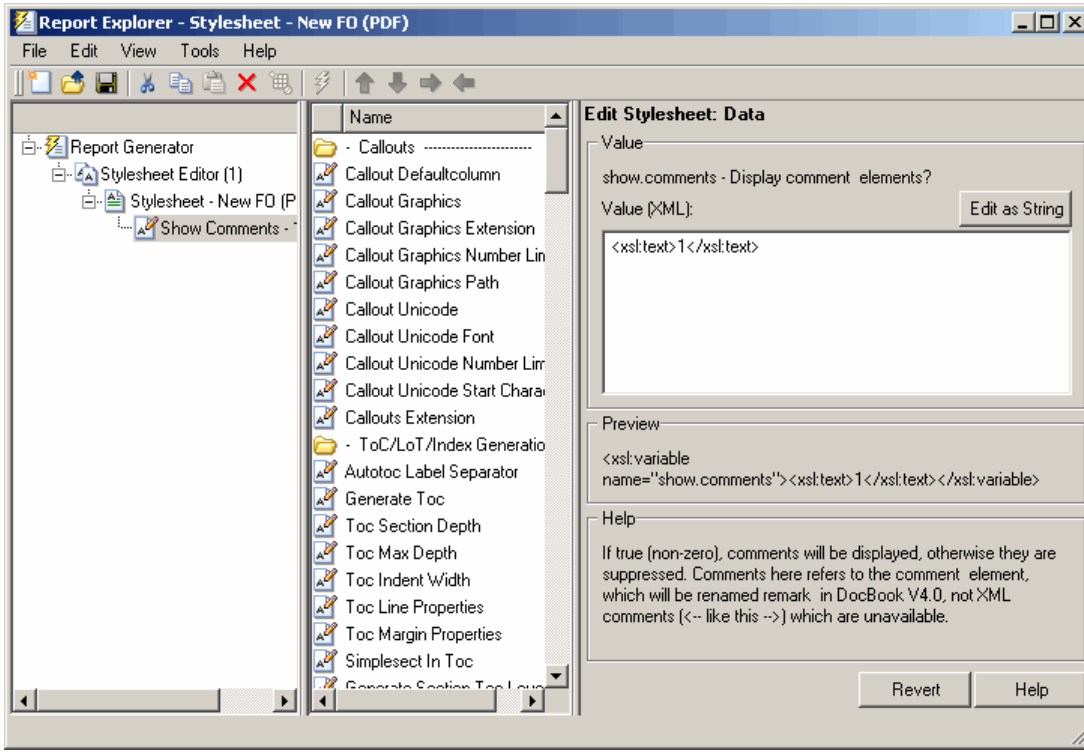
As shown in this example, the value of the data item is of type Boolean and the current value is specified as **true(1)**. The Properties pane on the right is set to simple edit mode.

To change this value, use the drop-down menu for the value field and click a different value. In this case, the only other possible value is **false(2)**.

For more complex editing, click **Edit as XML**.

Note Advanced edit mode is not supported for data items in RTF stylesheets.

In advanced edit mode, the Properties pane for this data item appears as shown.



Strings

For the values of some data items, the Stylesheet Editor displays text in the **Value** field. This text is not an XML expression, and you can edit it freely in the **Value** field. The text does not have to be specified as an XML expression, but you have the option of typing XML characters.

If you prefer to edit the XML directly, use advanced edit mode as follows:

1 Click **Edit as XML**.

2 Edit the XML in the **Value (XML)** pane.

Note You cannot edit data items for RTF stylesheets as XML.

XML Expressions

If you want to make complex changes to a stylesheet, consider using advanced edit mode, that is, edit XML expressions directly in the **Value (XML)** pane. If this pane is not displayed, click **Edit as XML** to switch to advanced edit mode.

When you edit XML expressions, you must make sure that you type valid XML in the **Value (XML)** pane. Values that are not legal generate an error. The error appears at the top of the Properties pane.

If you get an error, you can fix the problem by editing the expression in the **Value (XML)** pane. The Stylesheet Editor does not allow you to use illegal XML code in the report generation process.

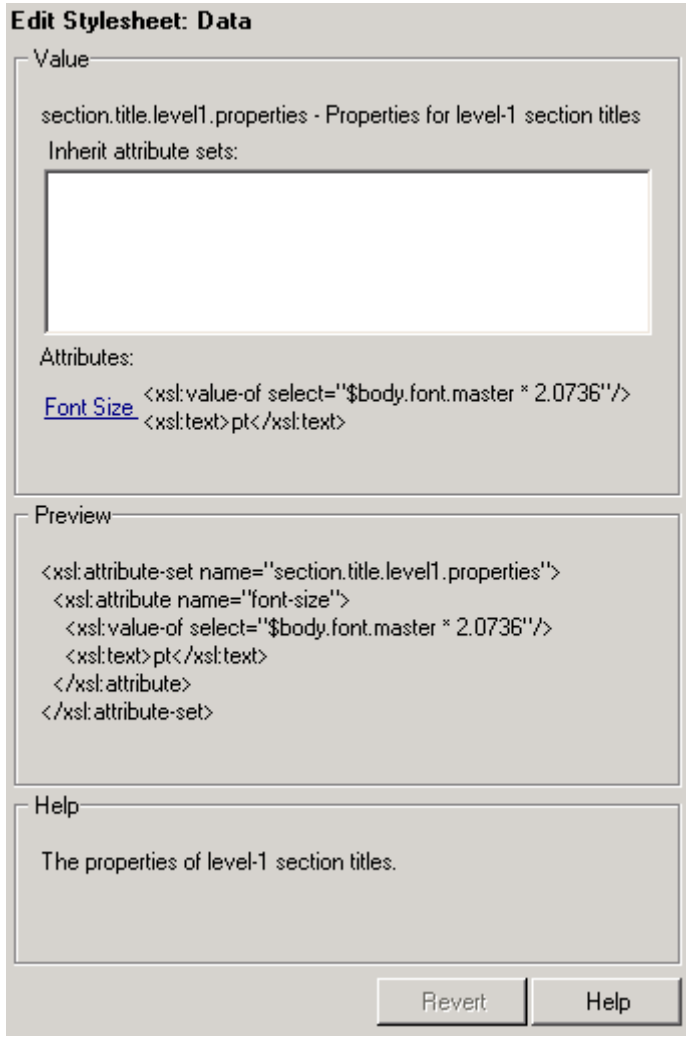
Editing Font Size as a Derived Value in XML

You might want to edit in XML directly if you need to change conditional values or values derived from others.

To see an example of values derived from others:

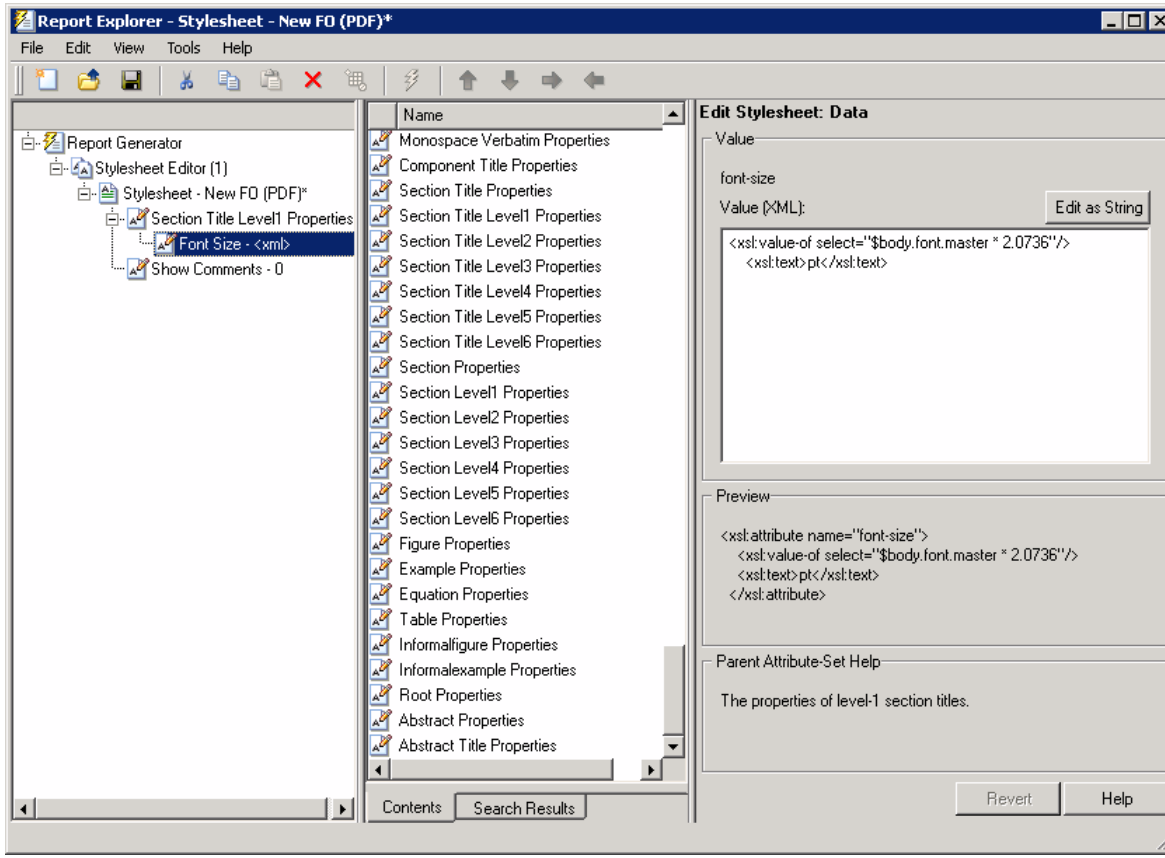
1. Open the default print stylesheet for PDF documents.
2. In the Options pane in the middle, select and expand the **Property Sets** folder.
3. In the Options pane, double-click the **Section Title Level1 Properties** data item.

   The Properties pane on the right appears as shown.



4 In the **Attributes** area of the Properties pane on the right, click **Font Size - <xml>**.

The Report Explorer appears as shown.



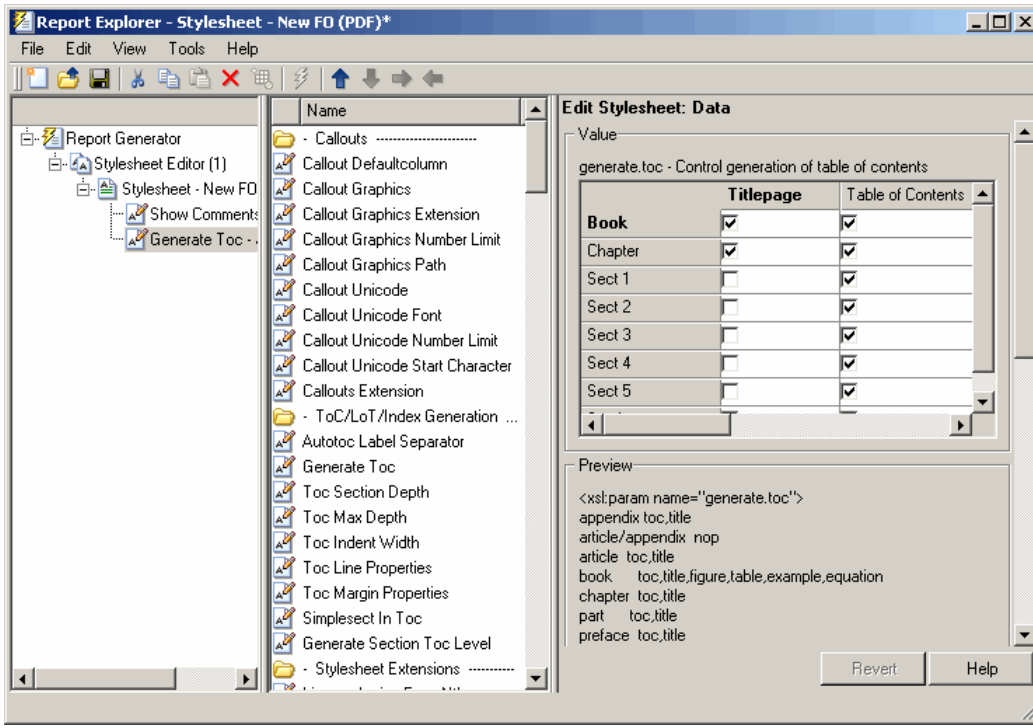
Notice that the value is specified as a product of **$body.font.master** and **2.0736**. To change the font size to a larger size, you could change the multiplication factor to **3.0736**.

Note The value for the **$body.font.master** data item is specified in the **Body Font Master** property, which is in the **Pagination and General Styles** category in the Options pane in the middle. The default value is 10. Changing this value causes the derived values to change accordingly.

TOC Generation Properties

To change values for generation of the table of contents (TOC) you use simple edit mode and select the values you want from a matrix of check boxes.

Here is an illustration of the Report Explorer ready for editing the values for the **Generate Toc** data item, which belongs to the **PDF** stylesheet.



Title Placement Properties

The **Title Placement** data items are in the **Miscellaneous** category. They control the position of titles for figures and tables. You can edit the values for these items in simple edit mode.

When you select one of these data items for editing, the Properties pane on the right shows the possible values in a drop-down menu. Use the menu to indicate whether you want the title to go before or after the figure or the table.

Attribute Sets

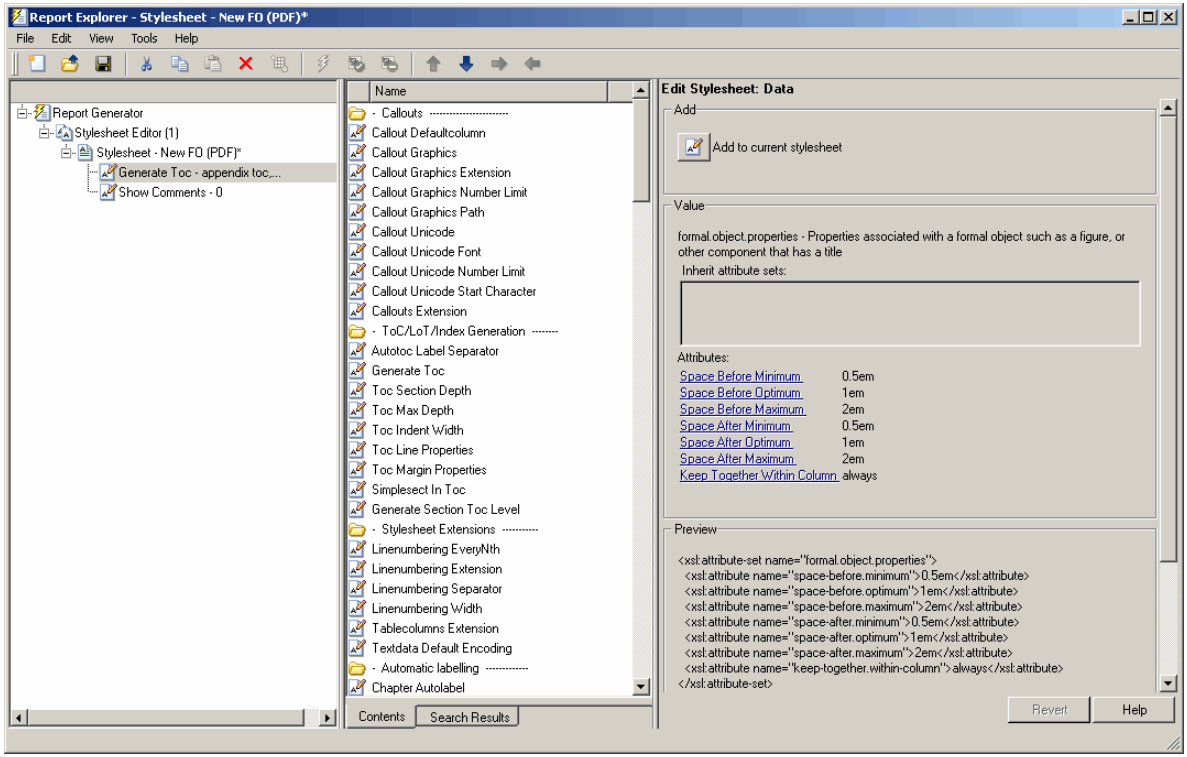
An attribute set is a group of attributes. When you select an attribute set in the Outline pane on the left, the Properties pane lists the attributes that belong to the set.

To edit an attribute, expand the attribute set in the Outline pane and select the attribute.

To edit the attribute set, type text in the **Inherit attribute sets** area of the Properties pane.

An example of an attribute set is **Formal Object Properties**, a data item in the **Property Sets** category of the default print stylesheet for PDF documents.

Here is an example of the Report Explorer showing the **Formal Object Properties** attribute set in the Properties pane.



Attributes

An *attribute* is a data item that specifies information for an XML element. An attribute must be a child of an Attribute Set.

You can edit the values for attributes in either simple or advanced edit mode.

Note The information in the Help area of the Properties pane for an attribute describes the set to which the attribute belongs.

An example of an attribute is the **Space Before Minimum** attribute in the attribute set **Formal Object Properties**.

Varpair Values

All the data items in RTF stylesheets are represented as varpair data items, which are name/value pairs of information. RTF stylesheets are the only type of Report Generator stylesheets that have varpair data.

You can edit varpair data items only as strings or as Boolean values. Boolean values are represented as `true (#t)` and `false (#f)`.

You cannot edit RTF stylesheet data items as XML.

Note Because the varpair data is sometimes represented in the stylesheet as DSSSL rather than XML, the code that appears in the **Preview** pane of the Properties pane on the right looks different from the code associated with other kinds of stylesheets supported by Report Generator.

Cell Groups

There are two cell groups: **Header Content** and **Footer Content**, which are available for PDF reports. A *cell group* contains one or more stylesheet cells. A *stylesheet cell* is a set of values that determine the content of a particular portion of the header or footer for a page.

The Properties pane for each of cell group lists the current stylesheet cell definitions for the group in a two-column list called **Conditional cell values**. The first column displays the name of a condition and the second column displays the content and default value to be used if a specified condition is met.

For instance, `Page sequence - Blank` is the name of a stylesheet cell. It specifies the content for a blank page; by default the content is set to empty. Similarly, `Cell - Right Side` specifies the content for the right side of the header on every page.

There are many combinations of conditions and values that you can use to customize the content of headers and footers. Report Generator provides several predefined conditions that are frequently used. These predefined cells are listed in the Properties pane for Header Content and for Footer Content cell groups.

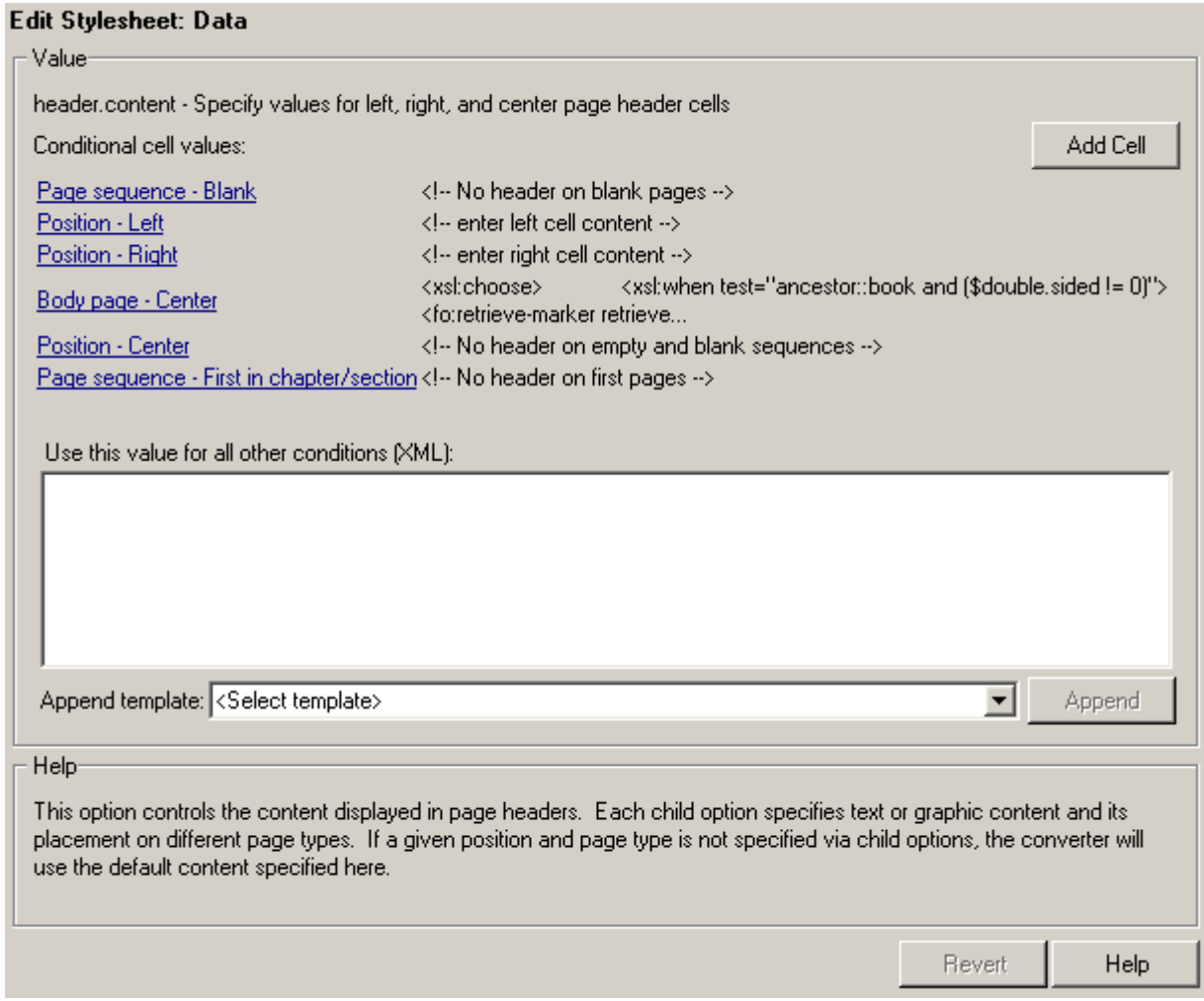
See “Stylesheet Cells” on page 9-30 for details about stylesheet cells and how the File Converter works with them.

Stylesheet Cells

This section covers the following topics:

What Is a Stylesheet Cell?

For purposes of specifying the content for headers and footers, Report Generator defines a page to include six possible locations, or cells. These positions correspond to the left, right, and center of the header and the left, right, and center of the footer.



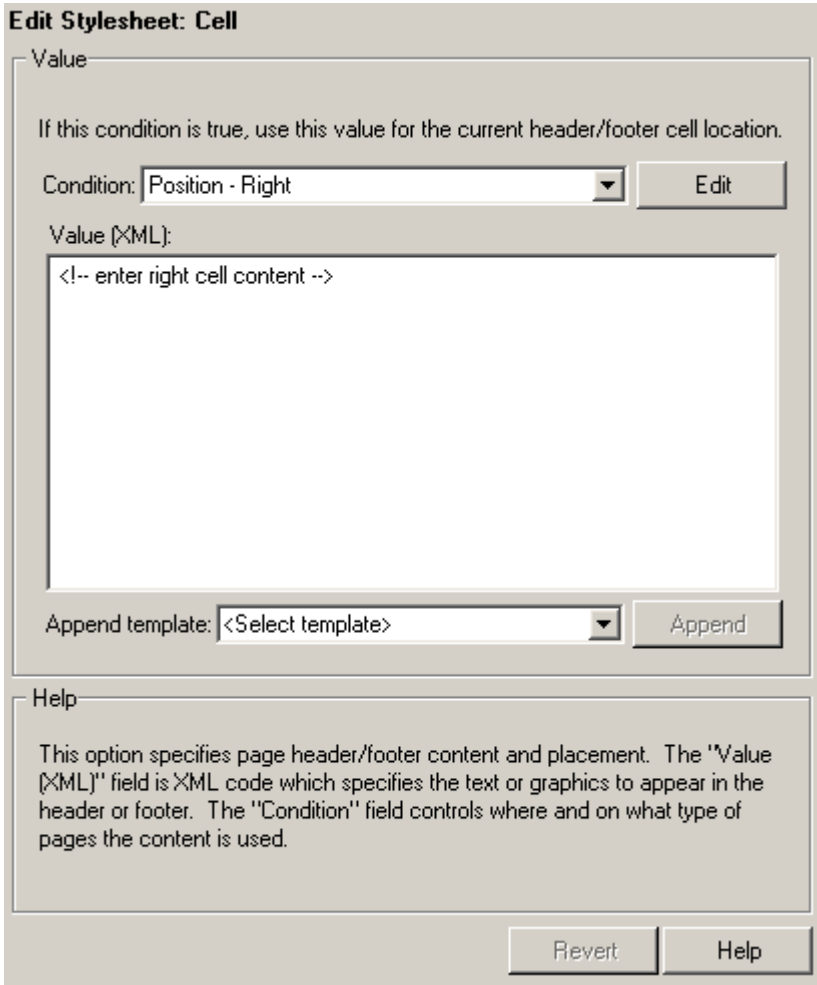
Conditions and Their Values. A stylesheet cell contains a condition and content. For example, say you want the right side of the top of all pages to have a page number. In this case you could define a basic stylesheet cell in the Header Content cell group to have a condition set to `right` and content set to be the current page number. To view this example:

1 Open the stylesheet editor for a PDF stylesheet.

2 Double-click **Header Content** in the Options pane in the middle.

3 Click **Position - Right** in the Properties pane on the right.

For `Position - Right`, the XML code for the condition and content appear as shown:



As you can see, the XML code for the condition is as follows:

```
$position='right'
```

In this example the code for content is empty. Select a template and click **Append** to add the content that you need for your particular report.

See “Using Templates to Add Content to Headers and Footers” on page 9-35 for details about templates.

Complex Conditions. You can specify more complex conditions. Complex conditions typically specify not only the position on the page but other settings, such as whether the content should appear on a title page (The condition of being on a title page belongs to the `$pageclass` condition.)

You can use standard logical operators (such as `=` , `!=` , `and`, `or`) to specify a complex condition. You can also use nested expressions (characters between parentheses are an expression within an expression).

For instance, you can specify that the center position of the header should be empty if a title page is being generated, but that the center should contain the company logo if a body page is being generated. This specification corresponds to a default stylesheet cell named `Body page Center`, which is defined as follows:

```
($sequence='odd' or $sequence='even') and
$position='center' and
$pageclass != 'titlepage'
```

This expression indicates a condition where the page is either odd or even, the position must be the center (of the header or footer), and the page is not a title page.

See “Adding a Company Logo to Headers in a PDF Report” on page 9-41 for an example of how apply this condition to a report.

When the File Converter processes a page it evaluates the settings that are relevant to each of the six cells on the page and adds content accordingly. If there are no conditions in effect for a cell, the File Converter uses the default values as defined in the cell group.

The possible conditions and their values as coded in XML are shown in the following table.

Name of Condition	Possible Values for the Condition	Sample XML Code

`$position`	`right`
	`center`
	`left`
		`$position='right'`
		`$position='center'`
		`$position='left'`

`$sequence`	`odd`
	`even`
	`first`
	`blank`
		`$sequence=odd`
		`$sequence=even`
		`$sequence=first`
		`$sequence=blank`

`$double-sided`	`0` `1`	`$double-sided=0` `$double-sided=1`

`$pageclass`	`$titlepage`
	`$lot`
	`$body`
		`$pageclass=$titlepage`
		`$pageclass=$lot`
		`$pageclass=$body`

Specifying Content for Headers and Footers

Use the Properties pane for a stylesheet cell to specify content according to conditions.

The Properties pane for a stylesheet cell displays the following areas.

Label	Definition	Description
Condition	Condition that must be satisfied for the content to be used.	This is a drop-down list of frequently used and predefined conditions. They are listed by their display names. Click **Edit** to view or change the XML code that goes with the name. See "What Is a Stylesheet Cell?" on page 9-30 for details about the concept of a stylesheet cell.

Label	Definition	Description
Value (XML)	Content to be used if the condition is met.	This is an editable field where you modify or create the XML code that specifies the content.
Append Template	Name of the template you want to use to add content.	To help you specify the values for a stylesheet cell, the Stylesheet Editor provides templates that contain some or all of the XML code needed to add some kinds of content. See "Using Templates to Add Content to Headers and Footers" on page 9-35 for details about this feature.

Using Templates to Add Content to Headers and Footers

To specify content without using a template, type XML code directly into the **Value (XML)** text box.

To use a template to specify content:

1. In the **Append template** drop-down list, select the kind of content you want to add.
2. Click **Append**.

   The Properties pane on the right displays the XML code for the default content of that kind. You can edit the XML to change the default.

There are templates available for adding text, author names, page numbers, titles for chapters and sections, chapter numbering, draft information, comments, and graphics.

For example, to specify text as the content:

1. Select **Text** from the **Append template** drop-down list.
2. Click **Append**.
3. Edit the value for `xsl:text`.

   The default content is `Confidential`.

Note Templates used by the File Converter are Extensible Style Language Transformations (XSLT), which is a language for transforming XML documents into other XML documents. For details about XSLT, see the website for the World Wide Web Consortium (W3C).

Note on the Relationship Between Properties and Cells for Headers and Footers

This section covers the following topics:

- "Modifying Fonts and Other Properties" on page 9-36
- "Inserting Graphics Files" on page 9-36

Modifying Fonts and Other Properties. You cannot use stylesheet cells to modify the font family or other such properties of headers and footers. To specify the style of the content in headers and footers, use **Header Content Properties** and **Footer Content Properties**, which are attribute sets.

Each of these attribute sets is a pagination style data item for PDF stylesheets. You modify an attribute in the set, such as the Font Family attribute in Header Content Properties, by selecting the attribute in the Outline pane on the left.

Inserting Graphics Files. When you add content in the form of a graphics file, you must specify not only the name of the file in the stylesheet cell that belongs to **Header Content** or **Footer Content**; you must also edit other pagination properties to allow room for the graphic to appear where you specify. Typically you can do this by selecting and editing the values of the `Region Before Extent` and `Region After Extent` data items, which are in the **Pagination and General Styles** folder of the **Options** pane for PDF formatting.

See "Adding a Company Logo to Headers in a PDF Report" on page 9-41 for an example of adding a graphic.

Note Only bitmap images (`.bmp` files) are supported by the Stylesheet Editor as image content in headers and footers.

Making Specific Stylesheet Changes

This section provides the following extended examples:

See also "Editing Font Size as a Derived Value in XML" on page 9-23 for another short example.

Changing Font Size, Page Orientation, and Paper Type

This example shows how to do the following:

- Generate an XML source file without converting it to a report.
- Make the section headers larger in a report.
- Change the page orientation to landscape.
- Change the paper type to A4 (a popular paper size in Europe).

For this example you create a new stylesheet by editing an existing stylesheet. The purpose of creating the new stylesheet is to change the appearance of **wsvar-report**, a demo that is provided with Report Generator.

1 Generate a source file for the report.

For this example, you convert the same report several times. If you start by generating the report to XML without any formatting, you can use the source file more than once without regenerating the report.

a. From the command line, issue the following command:

```
setedit wsvar-report
```

b. In the **Report Format and Stylesheet** pane, change the format to **DocBook (no transform)**.

c. Select **File > Report** to generate the report.

Report Generator creates an XML source file without converting it to a standard documentation format. The XML source file appears in the MATLAB editor when the report is created.

2 Convert the report to see how it looks.

a. Select **Tools > Convert Source File** to open the File Converter.

b. From the **Source file** drop-down menu, select **wsvar-report.xml**.

c. From the **Format** drop-down menu, select **PDF**.

d. From the **Stylesheet** drop-down menu, select **Unnumbered Chapters and Sections**.

e. Click **Convert File**.

Report Generator converts the XML source file for `wsvar-report` to a PDF and opens the PDF in your viewer for PDF files.

3 Change the specification for the section headers to make the headers more prominent.

a. In the File Converter, click **Edit**.

Report Generator opens the Stylesheet Editor displaying the settings associated with the **Unnumbered Chapters and Sections** stylesheet.

b. In the Properties pane on the right, change the name of the stylesheet to `Custom Large Section Headers`.

c. Type in the description as `No chapter and section numbering, larger section titles`.

d. In the Outline pane on the left, select the **Custom Large Section Headers** stylesheet (or any of its children), if not already selected.

e. In the Options pane in the middle, select **Section Title Level 1 Properties**.

f. In the Properties pane on the right, click **Add to current stylesheet**.

The **Section Title Level 1 Properties** data item appears in the Outline pane on the left as a child of the custom stylesheet. Because

this data item is an attribute set, it has child members (that is, attributes) displayed as well.

g. In the Outline pane on the left, select **Font Size**, a data item that is one of the attributes in the set.

   The Properties pane on the right displays an XML expression specifying a font size as a multiple of the Body Font Size attribute.

h. Click **Edit as string**.

   The Stylesheet Editor converts the XML expression to a simple string, which appears in a pane labeled **Value**.

i. Type in the value as `18pt`.

   This change in the stylesheet causes the size of the font to be absolutely 18 points, rather than a multiple of the body font size attribute.

j. Select **File > Save** to save the stylesheet.

   Report Generator prompts you for a file name because it does not allow you to overwrite built-in stylesheets.

k. Save the stylesheet to a directory on your MATLAB path with the name `customheader.rgs`.

The `customheader.rgs` stylesheet appears as an available stylesheet in the Options pane in the middle of the Stylesheet Editor. It also appears as an option in the File Converter.

4 Use the new stylesheet to convert the current XML source file.

a. In the **Stylesheet Editor: Main** Properties pane on the right, click **Send to File Converter**

   The File Converter appears with `customheader.rgs` stylesheet selected.

b. Click **Convert file**.

5 Change additional specifications (page orientation and paper type).

a. On the File Converter Properties pane, click **Edit**.

b. In the Options pane on the left, double-click the **Page Orientation** data item.

c. In the Properties pane on the right, use the drop-down menu to change the value of the data item to **landscape**.

d. In the Options pane in the middle, double-click **Paper Type**, which is also in the **Pagination and General Styles** folder.

e. In the Properties pane on the right, use the drop-down menu to select **A4**.

f. Save the stylesheet.

6 Convert `wsvar-report.xml` using the `Custom Large Section Headers` stylesheet.

The PDF shows a horizontally oriented page with slightly different dimensions.

Adding a Company Logo to Headers in a PDF Report

This example shows how to edit the default header content for a PDF stylesheet so that body pages will have an image in the center. The report that is used in the example is `mfile-report.rpt`, a sample report that documents the `peaks.m` file.

The image to be used for the content can be any bitmap or jpeg image you choose. You must know the size of the image so that you can allow enough room for it in the header. The image used here is named `sample_logo.bmp`. It is `50x44` pixels, which is about an inch on each side when printed on paper. It looks like this:

Note Only bitmap (`.bmp`) and jpeg (`.jpg`) images are supported by the Stylesheet Editor as image content in PDF reports.

Here are the steps to include `sample_logo.bmp` in the center of the header for each body page in a PDF version of `mfile-report.rpt`; that is, exclude the image only from the title page and from the first page in each chapter.

1. Open `mfile-report.rpt` by typing the following at the MATLAB command prompt:

   ```
   setedit mfile-report
   ```

2. Create a new stylesheet:

   a. Open the Stylesheet Editor and double-click it in the Outline pane on the left.

   b. Click **New FO (PDF)** in the Properties pane on the right.

   c. Type `Logo stylesheet for PDF` as the **Display name** and `Company logo in center of header` as **Description**.

   d. Select **File > Save** and save the file as `logo_stylesheet.rgs` in your MATLAB path.

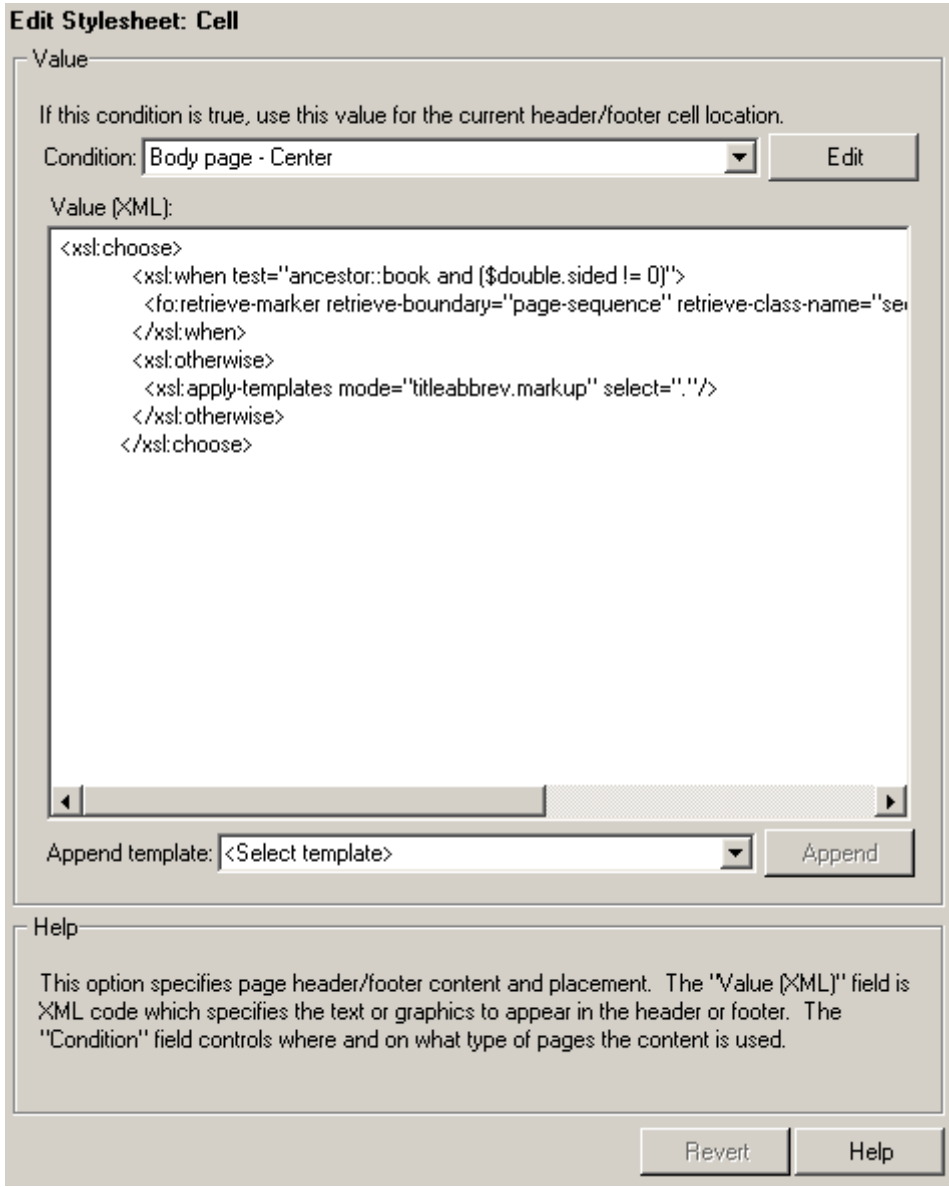
3. Open the cell group for editing:

   a. Scroll through the Options pane on the left to the **Pagination and General Styles** folder.

   b. Double-click **Header Content** in the Options pane.

   c. Click **Body – page Center** from the list of cells in the Properties pane on the right.

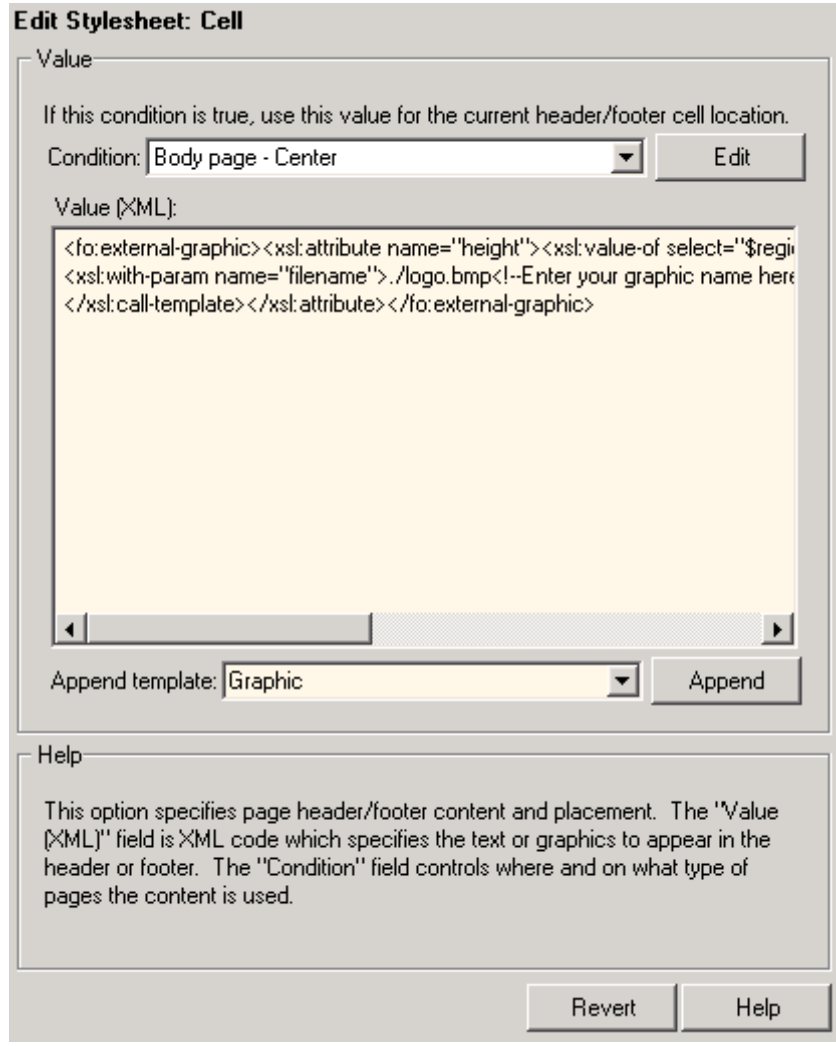
      The Properties pane appears as shown.



d. Select and delete all of the text currently in the **Value (XML)** area.

e. Select **Graphic** from the **Append template** drop-down menu, and click **Append**.

   The Properties pane on the right shows the XML code that tells the File Converter to include the graphic. You must edit this code to specify the name of the file and the space needed for it in the header. By default, the name of the graphic is `logo.bmp`, as shown:



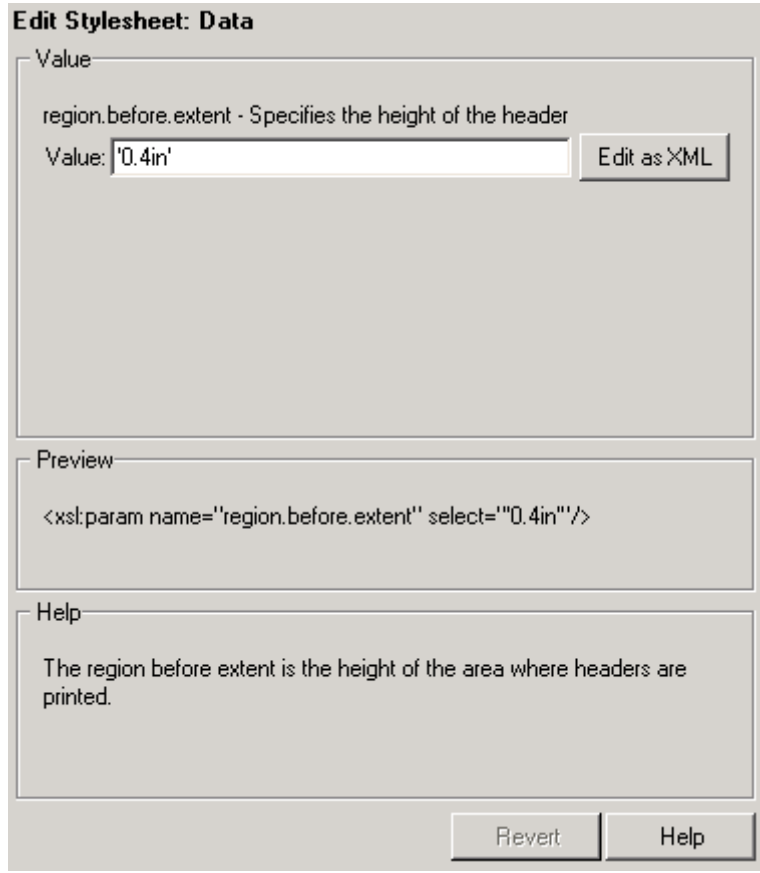
4 Add the name of your bitmap file to replace `logo.bmp` where it appears in the **Value (XML)** code.

5 Select **File > Save**.

6 Make sure that the amount of room available in the header is at least as large as the icon image file. You can set this with the parameter `region.before.extent` as follows:

   a. In the Options pane in the middle, double-click **Region Before Extent**, which is in the **Pagination and General Styles** folder.

      By default the value for the height of the header is `0.4` inches, as shown:



   b. Type the value as `1.0in` to replace `0.4in`.

   c. Select **File > Save**.

7 Generate the report with the new styles:

a. Select **mfile-report.rpt** in the Outline pane on the left.

b. Set the drop-down menus in the **Report Format and Stylesheet** text box of the Properties pane on the right to specify `Logo stylesheet for PDF` and `Acrobat (PDF)`

c. Click **Report** on the toolbar.

Stylesheet Reference Information

See the following table for a list of the properties for a Report Generator stylesheet.

Stylesheet Properties

Name	Description
Description	An explanation of the features that distinguish the stylesheet. This is an editable field.
Display name	The name that you want to display in the Options pane. This is not the same as the name of the `.rgs` file that you specify to store the stylesheet. This is an editable field.
Transform type	The process that Report Generator uses to generate reports that use this particular stylesheet. The types are • FO (Formatting Object) for PDF reports • HTML • DSSSL (Document Style Semantics and Specification Language) for RTF reports This is not an editable field.

Categories of Stylesheets Within Each Type

The categories for the styles that you can edit for each kind of stylesheet are listed in the following tables:

- Categories of Editable Styles in PDF (FO) Stylesheets on page 9-49
- Categories of Editable Styles in HTML Stylesheets on page 9-50
- Categories of Editable Styles in RTF (DSSSL) Stylesheets on page 9-51

Categories of Editable Styles in PDF (FO) Stylesheets

Category	Description of Data Items in Category
Automatic labeling	Options for enumeration of parts of the report, such as chapters and sections.
Callouts	Options and specifications related to callouts, such as defaults, use of graphics, size, path, fonts, characters, and extensions.
Font Families	Specification of defaults for body text, copyright, quotes, symbols, dingbats, monospace, sans serif, and titles.
Graphics	Specification of default width and options related to scaling attributes.
Lists	Specification of spacing related to lists and list items.
Meta/*Info	Options related to year ranges.
Miscellaneous	Options and specifications for placement of titles, comments, variable lists, block quotations, ulinks, hyphenations of URLs, verbatim environment display, use of SVG, table footnote numbers, superscript, and subscript.
Pagination and General Styles	Specifications of page orientation, margins, double-sided, paper type, hyphenation, line height, columns, master font, draft mode, watermark, blank pages, rules for headers and footers, and content of headers and footers.

Categories of Editable Styles in PDF (FO) Stylesheets (Continued)

Category	Description of Data Items in Category
Properties	Specification and options related to figure titles, monospace properties, verbatim text, section titles, and levels of sections.
Stylesheet Extensions	Line numbering and table columns extensions.
Table of Contents (TOC)/List of Tables (LOT)/Index Generation	Specifications for layout of TOC, depth of sections, indentation, and margins.
Tables	Specifications for size of tables and their borders.

Categories of Editable Styles in HTML Stylesheets

Category of Style	Description of Data Items in Category
Automatic labeling	Options for enumeration of parts of the report, such as chapters and sections.
Callouts	Options and specifications related to callouts, such as defaults, use of graphics, size, path, fonts, characters, and extensions.
Chunking	Options related to using an explicit TOC for chunking, depth of section chunks, navigational graphics, and display of titles in headers and footers.
Extensions	Line numbering, graphic size, and table columns extensions.
Graphics	Specification of default width and depth, use of HTML embed for SVG, viewports, and options related to scaling attributes.

Categories of Editable Styles in HTML Stylesheets (Continued)

Category of Style	Description of Data Items in Category
HTML	Specifications related to dynamically served HTML, base and head elements, type of stylesheet, css, propagation of styles, longdesc, validation, cleanup, draft mode, watermark, and generation of abstract.
Linking	Specification of Mailto URL and target for ulinks.
Meta/*Info	Options related to year ranges.
Miscellaneous	Options and specifications for comments, verbatim environment pixels, em space, use of SVG, and table footnote numbers.
Table of Contents (TOC)/List of Tables (LOT)/Index Generation	Specifications for layout of TOC, depth of sections, indentation, and margins.
Tables	Specifications for size of tables, table cell spacing and padding, and borders.
XSLT Processing	Options related to header and footer navigation and rules.

Categories of Editable Styles in RTF (DSSSL) Stylesheets

Category of Style	Description of Data Items in Category
Admonitions	Options and path for admonition graphics.
Backends	Options for Tex, MIF, and RTF back end usage.
Bibliographies	Options related to checking citations; suppressing, enumerating, and using titles of entries.
Fonts	Specifications for font family and size to use for some elements.

Categories of Editable Styles in RTF (DSSSL) Stylesheets (Continued)

Category of Style	Description of Data Items in Category
Footnotes	Options for ulinks as footnotes and page location.
Graphics	Specifications for file extensions, file names, and loading library database.
Indents	Specifications for hanging indents, first paragraphs, and start of blocks.
Labeling	Enumeration of sections and other elements.
Miscellaneous	Options for floating formal objects, punctuation for run-in heads and honorifics, bold for first use of term, minimum leading between lines, and automatic hyphenation.
OLinks	Using an extension for finding outline information.
Object Rules	Specifications for placement and width of rules.
Paper/Page Characteristics	Specifications for paper type, page numbers, width of pages, margins, and columns; heading-levels, sides; and writing mode (such as left-to-right).
Quadding	Specifications for justifying paragraphs.
RefEntries and Functions	Options related to generation and display of reference entries and synopses for functions.
Running heads	Options for generating and displaying running heads of chapters.
Table of Contents (TOC)/List of Tables (LOT)	Options to produce or display TOC for sets, books, parts, references, articles. Options to display TOC on title page.
Tables	Specification of width in simple list.
Titlepages	Options to produce and display title pages for sets, books, parts, references, articles. Options for author's name and ordering elements.

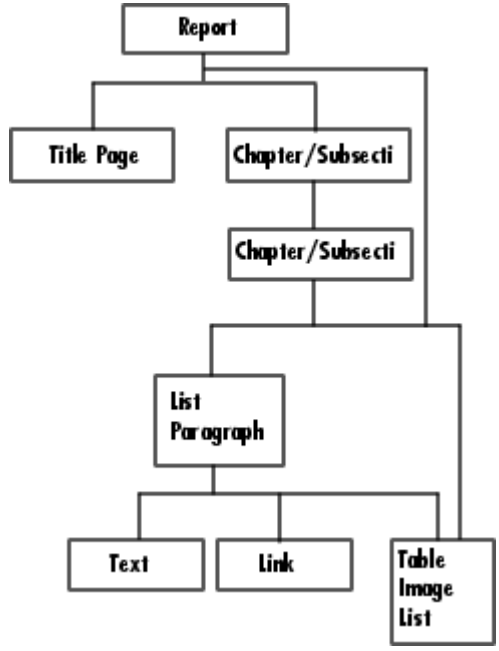
Categories of Editable Styles in RTF (DSSSL) Stylesheets (Continued)

Category of Style	Description of Data Items in Category
VariableLists	Options and specifications for term length and formatting.
Verbatim Environment	Specifications for width, enumeration, size, indentation, line frequency, and callouts.
Vertical Spacing	Specifications for space between lines and paragraphs.

Components — By Category

Formatting Components

You can use Formatting components to insert documentation elements in your report. Formatting components must have the following parent/child relationships.



The following table describes the Formatting components.

Component	Description
`Chapter/Subsection`	Groups portions of the report into sections. The following rules apply to the Chapter/Subsection component: • Children of the Chapter/Subsection component are placed inside the section created by this component. • The Chapter/Subsection component can nest inside another Chapter/Subsection component.
`Image`	Inserts an image into the report. The Image component can have the Chapter/Subsection or Paragraph component as its parent.
`Link`	Inserts a link or anchor into the report. The Link component should have the Paragraph component as its parent. If you select **Anchor** in the `Link Type` list, the Link component can also have the Chapter/Subsection component as its parent.
`List`	Inserts a bulleted or numbered list into the report. The List component must have the Chapter/Subsection or Paragraph component as its parent.
`Paragraph`	Inserts a paragraph into the report. The Paragraph component must have the Chapter/Subsection component as its parent. The Paragraph component can have child components, or it can have text. Text must be entered in the **Paragraph Text** parameter field.

Component	Description
`Table`	Converts an array into a table in the report. The Table component can have either the Chapter/Subsection or Paragraph component as its parent and cannot have children.
`Text`	Inserts text into the report. In general, the Text component should have the Paragraph component as its parent.
`Title page`	Inserts a title page at the beginning of the report. The Title Page component should have the Report component as its parent and should be the first component that inserts information into the report. This component cannot have any children. If your report contains a title page, it should also contain Chapter/Subsection components.

Report Generator Components

Report Generator components are “general utility” components. The following table describes these components.

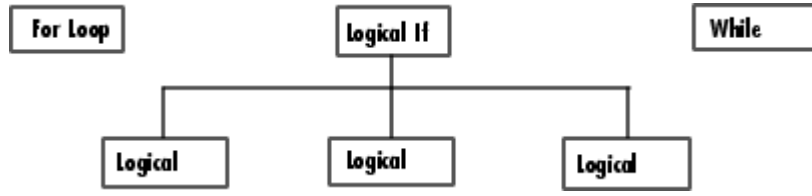
Component	Description
`Comment`	Inserts a comment into the DocBook source file created by the report generation process. The comment is not visible in the generated report, but you can edit the DocBook source file to make the component appear in the report.
`Empty Component`	Does not insert anything into the report and can have any component as a child. It can be used to group components so that the components can be moved or activated/deactivated together, or it can create a blank space in a list.
`Import File`	Imports plain text, DocBook XML, HTML, or RTF documents into the report.
`Nest Setup File`	Inserts a report template at the point where the Nest Setup File component is listed in the current report template. The Nest Setup File component allows one report template (`.rpt`) to run inside another. The components of the inserted report template enter the current report template at the level of the Nest Setup File component. Thus, the inserted components have the same parent component as the Nest Setup File component. You must make sure that the inserted components can have this parent component.

Component	Description
`Stop Report Generation`	Halts report generation. The Stop Report Generation component acts like the **Stop** button during report generation, and can be used to halt report generation under specified conditions.
`Time/Date Stamp`	Inserts the time and date of report generation. The Time/Date Stamp component behaves like a Text component; it must be inside a paragraph or title.

Logic and Flow Control Components

Logical and Flow Control components execute conditionally, enabling you to decide when a child component executes or how many times a child component executes.

These components have the following parent/child relationships.



The following table describes the Logical and Flow Control components.

Component	Description
`For Loop`	This component runs its child components iteratively. The For Loop component must have at least one child. The purpose of this component is to run its children several times. If it does not have any children, this component does not add anything to the report.
`Logical Else`	When the parent `if` condition is false, only the children of the `else` component run.
`Logical Elseif`	When the parent `if` condition is false, the process checks all the `elseif` statements in order, and the children of the first true `elseif` statement run.
`Logical If`	`if` statement for report generation. When the `if` statement results in true, the component runs its child components.
`Logical Then`	When the parent `if` condition is true, only the children of the `then` component run.
`While Loop`	This component loops while a workspace expression is true. This component runs its child components until its conditional string is true. The number of repetitions can be limited to prevent infinite loops. Like the For Loop component, the While Loop component must have at least one child. If it does not have any children, this component does not add anything to the report.

You can use the Logical components, including Logical If, Logical Then, Logical Else, and Logical Elseif, in the following ways.

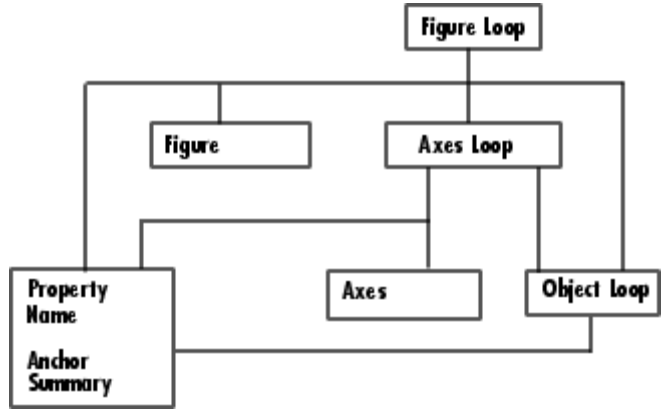
Component Used	Appearance in the Outline	How the Components Execute
`Logical If`	`if`	When the if condition is true, this component runs its children.

`Logical If`
`Logical Then`
`Logical Else`
	`if`
	
	`then`
	`else`
		• When the `if` condition is true, only the children of the `then` component run, and the process exits the `if` family.
		• When the `if` condition is false, only the children of the `else` component run.

`Logical If`
`Logical Then`
`Logical Elseif`
`Logical Else`
	`if`
	
	`then`
	`elseif`
	`elseif`
	
	.
	.
	.
	
	`else`
		• When the `if` condition is true, only the children of the `then` component run, and the process exits the `if` family.
		• When the `if` condition is false, the process checks all the `elseif` statements in order, and the children of the first true `elseif` statement run. The process then exits the `if` family.
		• If none of the `elseif` statements is true, the process runs the children of the `else` statement.
		Note that only one of the conditional statements (`then`, `else`, `elseif`) actually executes.

Handle Graphics Components

Handle Graphics components report on Handle Graphics figures. The following figure shows the Handle Graphics component hierarchy.



The following table describes the Handle Graphics components.

Component	Description
`Axes Loop`	Runs its child components for all axes objects for the current figure.
`Axes Snapshot`	Inserts an image of the current axes into the generated report.
`Figure Loop`	Runs its children for each Handle Graphics figure that you select. If a Handle Graphics component does not have the Figure Loop component as its parent, it operates on the current figure.
`Figure Snapshot`	Inserts an image of a figure window into your report. The Figure Snapshot component behaves like the Image component and expects the Figure Loop component as its parent.
`Handle Graphics Linking Anchor`	Designates a location to which other links point. This component should have a looping component as its parent.
`Graphics Object Loop`	Runs its child components for each Graphics object in the current figure or axes.
`Handle Graphics Name`	Inserts the name of a Handle Graphics object into your report. The Handle Graphics Name component behaves like the Text component and expects the Figure Loop component as its parent.
`Handle Graphics Property Table`	Creates and inserts a property name/property value table for Handle Graphics figures, axes, or other objects. The Handle Graphics Property Table component behaves like the Table component and expects the Figure Loop component, Axes Loop component, or Graphics Object Loop component as its parent.

Component	Description
`Handle Graphics Parameter`	Inserts a single property name/property value pair from a Handle Graphics figure, axes, or any other Handle Graphics object into your report.
`Handle Graphics Summary Table`	Inserts a table that summarizes many Handle Graphics objects' properties. You can specify the object properties to be included in the report.

The following example shows how Handle Graphics components can be used in a report template.

```
Report
 Figure Loop
  Chapter/Subsection
  Name
  Anchor
  Property Table
  Summary Table
   Axes Loop
    Chapter/Subsection
    Name
    Anchor
    Axes Snapshot
    Property Table
    Summary Table
     Graphics Object Loop
      Chapter/Subsection
      Name
      Anchor
      Property Table
      Summary Table
```

MATLAB Components

You can use MATLAB components to interact with the workspace. The following table describes the MATLAB components.

Component	Description
`Evaluate MATLAB Expression`	Evaluates a specified MATLAB expression. The result of the evaluation and/or MATLAB expression (M-code) can appear in the report.
`Insert Variable`	Inserts a specified variable into the report.
`MATLAB/Toolbox Version number`	Creates a table that shows the version number, release number, and/or release date of any MathWorks software you are currently using.
`MATLAB Property Table`	Creates a table that includes variables from the MATLAB workspace. The MATLAB Property Table component behaves like the `Table` component.
`Variable Table`	Creates a table that includes all variables in the MATLAB workspace. The Variable Table component behaves like the `Table` component.

Requirements Management Interface Components

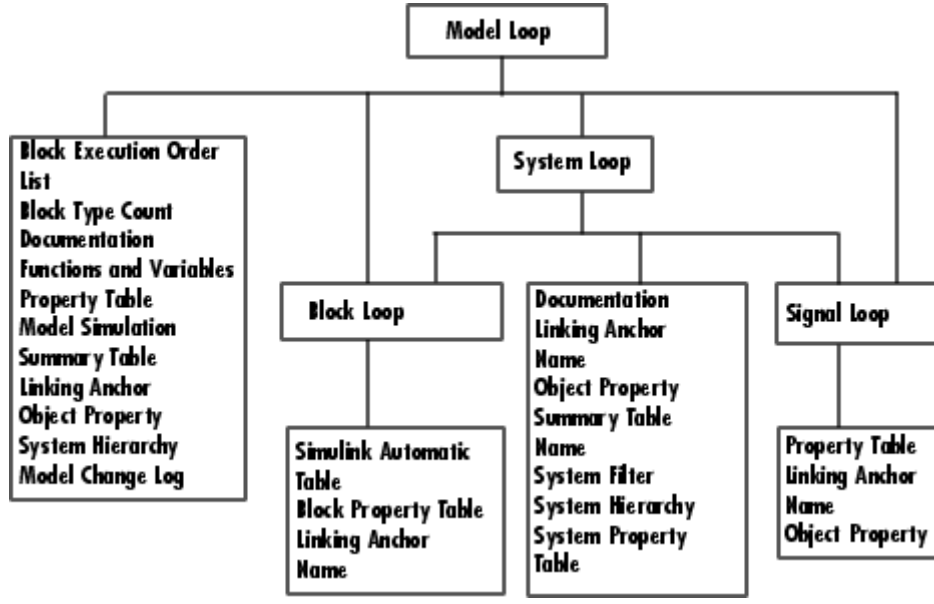
You can use Requirements Management Interface components to include information about requirements in your report.

Component	Description
`Requirements Table`	Creates a table that contains information from Simulink Verification and Validation.

Simulink Components

You can use Simulink components to include information on Simulink models in your report. Simulink components can report on four levels of a Simulink model: model, system, block, and signal.

Simulink components have the following parent/child relationships.



The following table describes the Simulink components.

Component	Description
`Block Execution Order List`	Creates and inserts a list or table of all blocks in the model (displayed in order of their execution sequence) into the report. The Block Execution Order List component should have the Model Loop component as its parent. If you place this component inside the System Loop component, you will get the same list over and over again.
`Block Loop`	Specifies blocks to be included in the report. The Block Loop component can have the Model Loop, System Loop, or Signal Loop component as its parent.
`Block Type Count`	Counts the number of each block type in a model or system. The Simulink Block Type Count component can have either the Model Loop or the System Loop as its parent.
`Model Advisor`	Inserts a Model Advisor report.
`Model Change Log`	Constructs a model history table that displays information about each logged revision to a model. This component must be executed before the Model Simulation component.
`Model Loop`	Runs its child components for all selected Simulink models and systems. The Model Loop component can run its child components for a specific model, all models, or the current model.
`Model Simulation`	Simulates the current Simulink model. The Model Simulation component expects the Model Loop component as its parent.
`Signal Loop`	Runs its child components for each signal in its parent loop. The Signal Loop component can have the Model Loop, System Loop, or Block Loop component as its parent.

Component	Description
`Simulink Automatic Table`	Creates a two-column property name/property value table.
`Simulink Dialog Snapshot`	Takes snapshots of Simulink editor dialog boxes
`Simulink Functions and Variables`	Inserts a table listing all workspace variables and functions used by blocks that are reported on in the current model. The Simulink Functions and Variables component expects the Model Loop component, System Loop component, or Block Loop component as its parent.
`Simulink Library Information`	Inserts a table listing all blocks, libraries, reference blocks, and link status for the current model.
`Simulink Linking Anchor`	Acts like a link in a paragraph or title. This component can have any Simulink looping component as its parent.
`Simulink Name`	Inserts the name of the current system, model, block, or signal, according to the option you specify. The Simulink Name component can have any Simulink looping component as its parent. This component behaves like the Text component.
`Simulink Property`	Inserts a single property name/property value pair for a Simulink object as text. The Simulink Property component can have any Simulink looping component as its parent.
`Simulink Property Table`	Inserts a property name/property value table for the model, system, block, or signal specified by the Model Loop component. The Simulink Property Table component has the Model Loop component as its parent.

Component	Description
`Simulink Summary Table`	Inserts a table that lists several Simulink objects and their properties. You can specify the Model Loop as the parent of the Simulink Summary Table component to summarize all systems, blocks, or signals in your model. You can specify the System Loop component as the parent of the Simulink Summary Table to summarize all blocks or signals in a given system.
`System Filter`	Runs its children only if certain conditions are met by the current system. The System Filter component expects the System Loop component as its parent.
`System Hierarchy`	Inserts a list of the names of parent and child systems in the current system; child systems are shown indented under the parent systems. The System Hierarchy component can have either the Model Loop or System Loop component as its parent.
`System Loop`	Runs its child components for each system defined in the Model Loop component. The System Loop component must have the Model Loop component as its parent.
`System Snapshot`	Inserts a picture of the current system into the report. The current system is defined by the System Loop component. The System Snapshot component should have the System Loop component as its parent.

Simulink Blocks Components

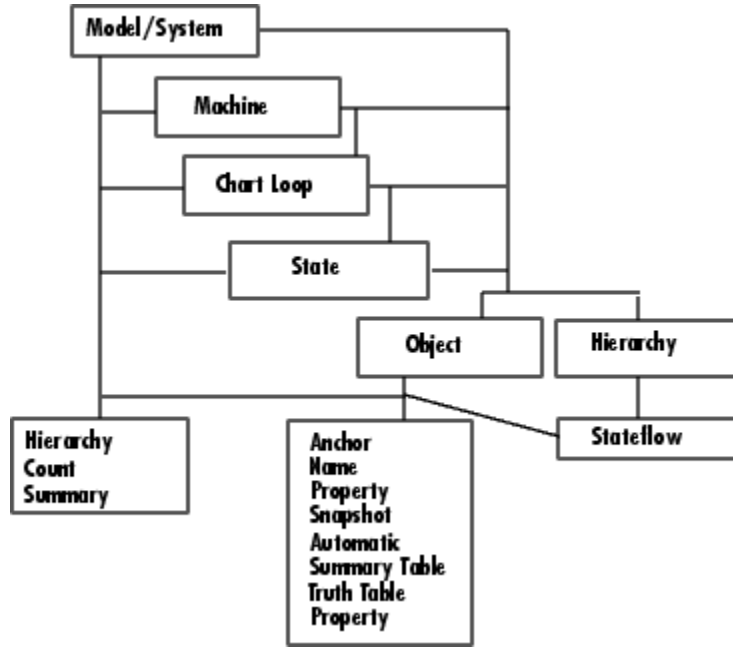
You can use the Simulink Blocks components to include information in the generated report about specific Simulink block types. Simulink Blocks components can be used in model, system, or block loops to report on block types in context.

The following table describes the Simulink Blocks components.

Component	Description
`Bus`	Creates a list of all signals exiting from a Bus Selector block. The list can contain only those signals leaving from the reported block or it can be hierarchical and display downstream buses and signals.
`Documentation`	Inserts text extracted from doc blocks in Simulink models. The Documentation component can have the Model Loop, the System Loop, or the Block Loop as its parent.
`Look-Up Table`	Reports on lookup table blocks; it inserts a figure and/or table into the report. The table contains input and output numeric values, and the figure is a plot of the values.
`Scope Snapshot`	Inserts a snapshot of all scope blocks and XY plots in your report. The Scope Snapshot component can have any Simulink looping component as its parent.
`To Workspace Plot`	Captures a plot created in the MATLAB workspace. This component inserts a figure and/or table into the report. The table contains input and output numeric values and the figure is a plot of the values.

Stateflow Components

You can use Stateflow components to include information on Stateflow models in your report. The following figure shows the parent/child relationships of Stateflow components.



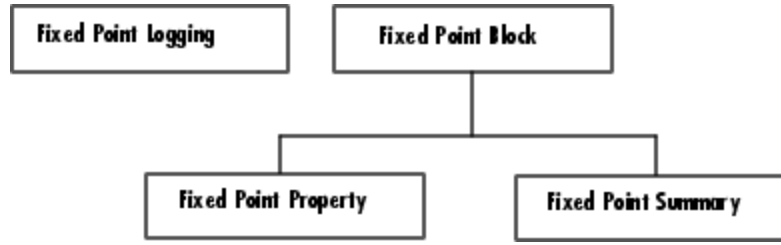
The following table describes the Stateflow components.

Component	Description
`Chart Loop`	Runs its children for each Stateflow chart that you choose. The Chart Loop component can have the Simulink Model Loop, System Loop, Signal Loop, or Block Loop component as its parent.
`Machine Loop`	Runs its child components for all selected Stateflow machines. The Machine Loop component can run its child components for a specific machine, all machines, or the current machine.
`Stateflow Count`	Counts the number of Stateflow objects in the current system. See the Stateflow documentation for a description of Stateflow objects.
`Stateflow Dialog Snapshot`	Takes snapshots of Stateflow editor dialog boxes.
`Stateflow Filter`	Runs its children only if the current object type (set by the parent State Loop) matches the component's object type. The Stateflow Filter component must have the Stateflow Loop component as its parent.
`State Loop`	Runs its children for each Stateflow object defined by its parent component. The State Loop component must have the Model Loop, System Loop, Signal Loop, Block Loop, Chart Loop. or Machine Loop component as its parent.
`Stateflow Automatic Table`	Inserts a table that contains the properties of a Stateflow object.
`Stateflow Hierarchy Loop`	Runs its child components on the entire Stateflow object hierarchy from the current point down.

Component	Description
`Stateflow Linking Anchor`	Acts like a link. This component must have the State Loop, Chart Loop, or Stateflow Filter component as its parent.
`Stateflow Name`	Inserts the name of the Stateflow object specified by its parent component into the report. This component must have the State Loop, Chart Loop, or Stateflow Filter component as its parent.
`Stateflow Hierarchy`	Inserts a tree representation into a generated report that describes the Stateflow object hierarchy.
`Object Loop`	Runs its child components for each terminal Stateflow object. The component inserts a list of object types.
`Stateflow Property`	Inserts a table, text, or paragraph that contains details of the selected Stateflow object property.
`Stateflow Property Table`	Inserts a property name/property value table for a Stateflow object into the report. The Stateflow Property Table component must have the Stateflow Filter component as its parent.
`Stateflow Snapshot`	Inserts an image of a Stateflow object into the report. The Stateflow Snapshot component is only valid for certain object types.
`Stateflow Summary Table`	Shows properties or parameters of the selected Stateflow objects in a table. The Stateflow Summary Table component can have the Chart Loop or any Simulink looping component (Model Loop, System Loop, Block Loop, State Loop, or Signal Loop) as its parent.
`Truth Table`	Inserts a table. The component must have the State Loop component as its parent.

Simulink Fixed Point Components

You can use Simulink Fixed Point components to include information on Simulink Fixed Point blocks in your report. The following figure shows the parent/child relationships of the Simulink Fixed Point components.



The following table describes the Simulink Fixed Point components.

Component	Description
`Fixed Point Block Loop`	Runs its children for the Simulink model, system, or signal defined by its parent component. For example, if the Fixed Point Block Loop is parented by the Simulink Model Loop, it runs its children for all fixed-point blocks in the Simulink model. The Fixed Point Block Loop component can have either the Simulink Model Loop, System Loop, or Signal Loop component as its parent.
`Fixed Point Logging Options`	Sets fixed-point options similar to those set in the Fixed Point Settings interface. You can use this component before a Model Simulation component in order to control how the model behaves during simulation.
`Fixed Point Property Table`	Inserts a property name/property value table for a fixed-point block. The Fixed Point Property Table component must have the Fixed Point Block Loop component as its parent.
`Fixed Point Summary Table`	Shows properties or parameters of the selected fixed-point blocks in a table. The Fixed Point Summary Table component must have the Fixed Point Block Loop as its parent.

Real-Time Workshop Components

You can use Real-Time Workshop components to include information on code generated by Real-Time Workshop in your report.

The following table describes the Real-Time Workshop components.

Component	Description
`Code Generation Summary`	Reports version information, a list of the generated files, code generation configuration information, and, if the model includes subsystems, a subsystem map.
`Import Generated Code`	Imports source and header (include) files generated by Real-Time Workshop and any custom files you specify as part of your model.

11

Components — Alphabetical List

Annotation Loop

Category

Simulink

Description

A Simulink annotation is a text area that you can edit directly in the block diagram. You typically use an annotation to display quick information about a subsystem.

The Annotation Loop component runs its child components multiple times for each Simulink annotation in the current context.

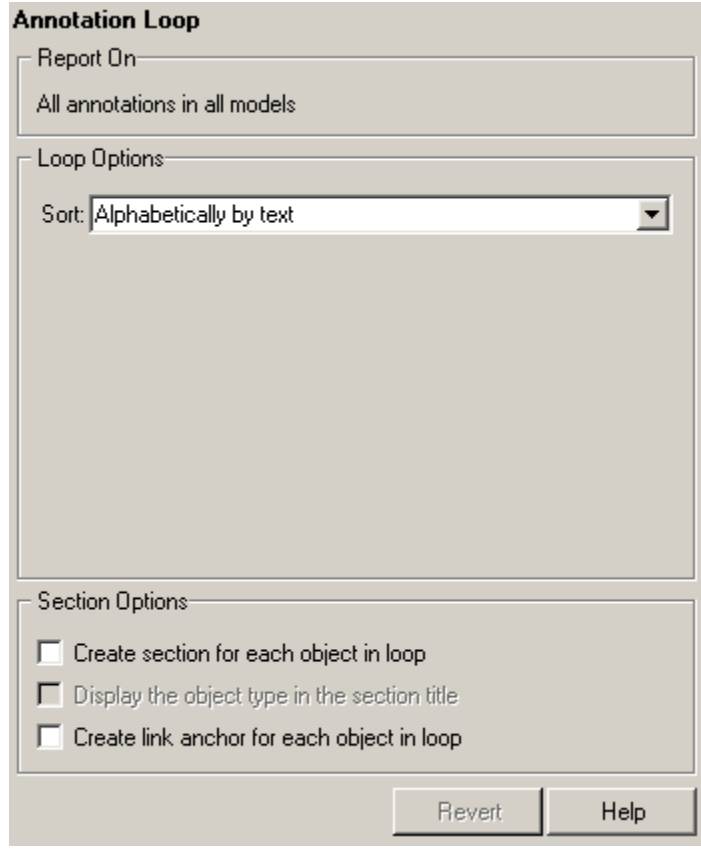
If the Annotation Loop is nested inside a Model Loop, it reports on all annotations inside the reported portion of the reported model.

If the Annotation Loop is nested inside a System Loop, it reports on all annotations inside the current system.

Annotation Loop components nested inside a Block or Signal Loop component do nothing.

Attributes

The following figure shows the **Annotations Loop** Properties pane.



Loop Options

The Loop Options pane displays information about the current context. You can sort `Alphabetically by text` or `In traversal order`.

All child components of the Annotation Loop consider their context to be annotations when the report is running.

For example, the following components report on the looped annotation:

- Simulink Automatic Table
- Simulink Linking Anchor

- Simulink Name
- Simulink Property
- Simulink Property Table

You can also show annotation objects in reports by using a summary table component. Each summary table component creates a single table with each reported annotation on a single row of the table.

Section Options

Create section for each object in loop
Select this option to insert a section in the generated report for each object found in the loop.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the object in the generated report.

Category

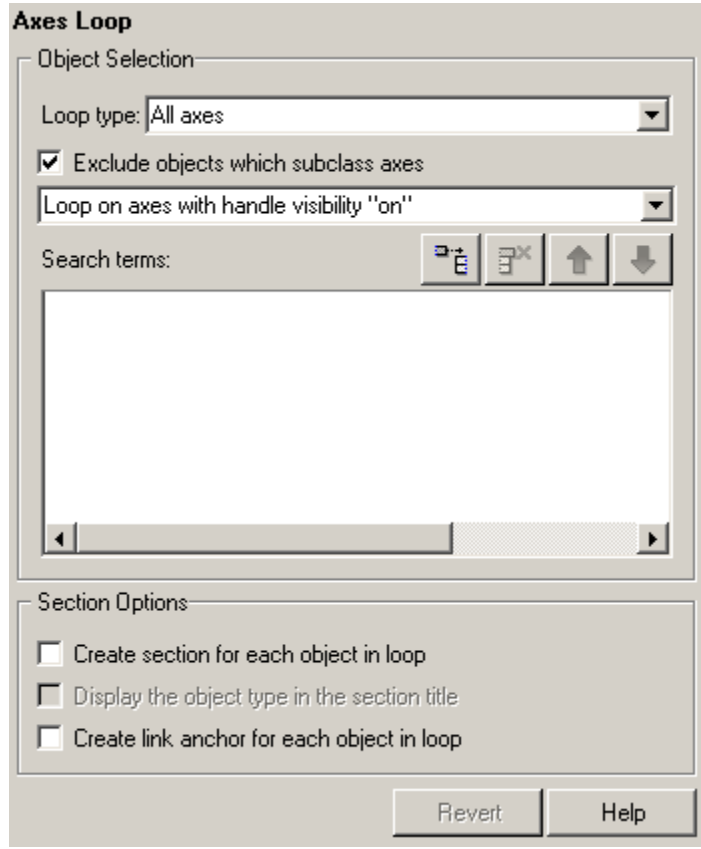
Handle Graphics

Description

The Axes Loop component runs its child components for all axes objects in the MATLAB workspace. For details about working with looping components, see “Working with Looping Components” on page 7-2.

Attributes

The following figure shows the **Axes Loop** Properties pane.



Axes Loop

Object Selection

Loop type
To loop on all axes objects, select `All axes`. To loop on the selected axes object, select `Current axes`.

Exclude objects which subclass axes
Select this option to exclude objects such as legends and color bars.

Loop Menu
To loop only on visible axes objects, select `Loop on axes with handle visibility "on"`. To loop on all axes objects, select `Loop on all axes`.

Search terms
Enter search terms for the loop. For example, to search for `Tag` and `My Data`, enter `"Tag", "My Data"`.

Section Options

Create section for each object in loop
Select this option to insert a section in the generated report for each object found in the loop.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the object in the generated report.

Insert Anything into Report?

Yes, inserts a section if the **Create section for each object in loop** option is selected.

File Name

`rptgen_hg.chg_ax_loop`

Category

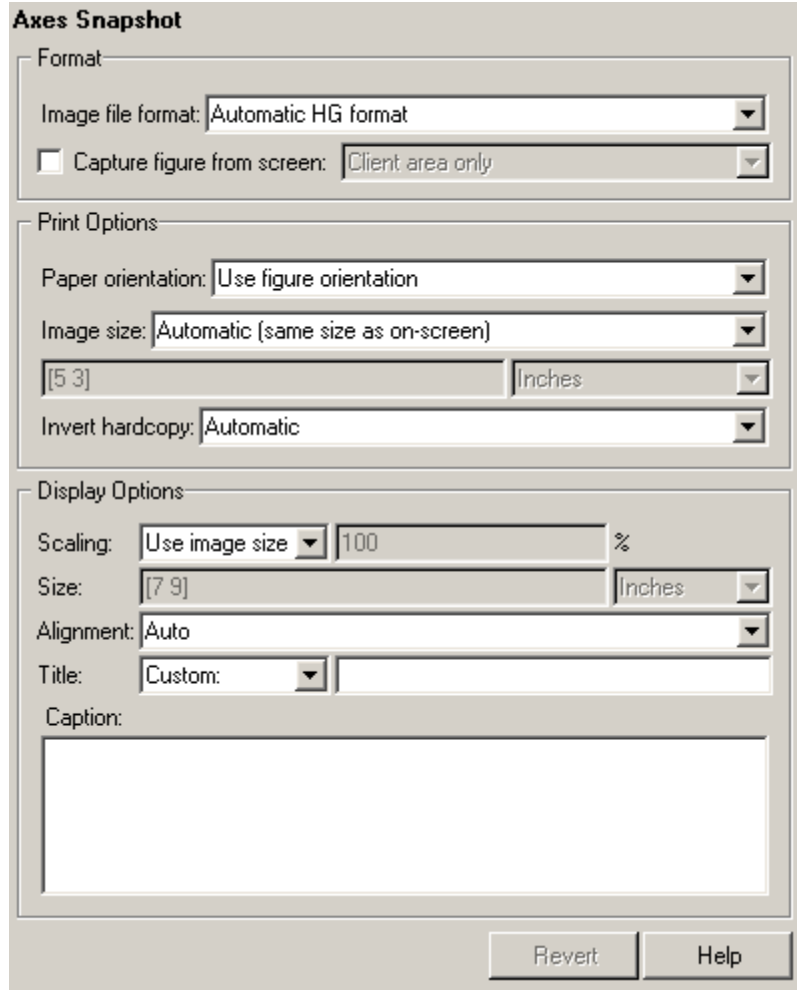
Handle Graphics

Description

The Axes Snapshot component inserts an image of the selected axes objects in MATLAB into the generated report.

Attributes

The following figure shows the **Axes Snapshot** Properties pane.



Format

Image file format

Choose the image file format (e.g., JPEG, TIFF, etc.) from this list. Select `Automatic HG Format` to automatically choose the format best suited for the output format that you chose in the Report component. Otherwise, choose an image format that your output

viewer can read. `Automatic HG Format` is the default option. The options include

- `Automatic HG Format` (uses the Handle Graphics file format selected in the Preferences dialog box)
- `Adobe Illustrator`
- `Bitmap (16m-color)`
- `Bitmap (256-color)`
- `Black and white encapsulated PostScript`
- `Black and white encapsulated PostScript (TIFF)`
- `Black and white encapsulated PostScript2`
- `Black and white encapsulated PostScript2 (TIFF)`
- `Black and white PostScript`
- `Black and white PostScript2`
- `Color encapsulated PostScript`
- `Color encapsulated PostScript (TIFF)`
- `Color encapsulated PostScript2`
- `Color encapsulated PostScript2 (TIFF)`
- `Color PostScript`
- `Color PostScript2`
- `JPEG high quality image`
- `JPEG medium quality image`
- `JPEG low quality image`
- `PNG 24-bit image`
- `TIFF - compressed`
- `TIFF - uncompressed`

- `Windows metafile`

Capture figure from screen

Select this option to capture the figure for the generated report directly from the screen. You can capture part of the figure by choosing the `Client area only` option, or capture the entire figure window by choosing `Entire figure window`. This option executes faster, but is not applicable for all formats. Also, the image to be captured might be obscured by other windows on the screen.

Print Options

Paper orientation

You can choose `Landscape`, `Portrait`, or `Rotated` as your paper orientation, or you can choose the `Use figure orientation` option. For information about paper orientation, see `orient` command in the MATLAB Function Reference documentation.

Image size

Select `Use figure PaperPositionMode setting` to use the `PaperPositionMode` property of the Handle Graphics figure to set the image size in the report. See `orient` in the MATLAB Function Reference documentation for information about paper position mode.

Select the `Automatic (same size as onscreen)` option if you want the image in your report to be the same size as the image on the screen.

Select the `Custom` option to choose a custom image size. If you choose this option, specify the image size in the **Size** field and **Units** list.

- **Size**: Enter the size of the Handle Graphics figure snapshot in the form `[w h]` (width height). This field is active only if you choose `Custom` in the **Image size** list box.
- **Units**: Enter the units for the size of the Handle Graphics figure snapshot. This field is active only if you choose `Set image size` in the **Custom** list box.

Invert hardcopy

Handle Graphics figures have an `InvertHardcopy` property. This parameter inverts colors for printing; i.e., it changes dark colors to light colors and vice versa.

Select one of the following options:

- `Automatic` — Select this option to automatically change dark axes colors to a light axes color. If the axes color is a light color, it is not inverted.
- `Invert` — Select this option to change dark axes colors to light axes colors and vice versa.
- `Don't invert` — Select this option if you do not want to change the colors in the image displayed on the screen for printing.
- `Use figure's InvertHardcopy setting` — Select this option to use the `InvertHardcopy` property set in the Handle Graphics image.
- `Make figure background transparent` — Select this option to make the image background transparent.

Display Options

Scaling

Select `Fixed size` to specify the number and type of units.

Select `Zoom` to specify the percentage, the maximum size, and the units of measure.

Select `Use image size` if you want the image in your report to be the same size as the image on the screen.

Size

Enter the exact size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Fixed size` in the **Scaling** drop-down list.

Max size
Enter the maximum size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Zoom` from the **Scaling** drop-down list.

Units
Enter the units for the size of the snapshot. This field is active only if you choose `Zoom` or `Fixed size` in the **Image size** list box.

Alignment
Options are `Auto`, `Right`, `Left`, and `Center`.

Note This option is supported only in `PDF` and `RTF` reports.

Title
Type the text you want to appear above the snapshot.

Caption
Type the text you want to appear below the snapshot.

Insert Anything into Report?

Yes. Image.

File Name

`rptgen_hg.chg_ax_snap`

Category

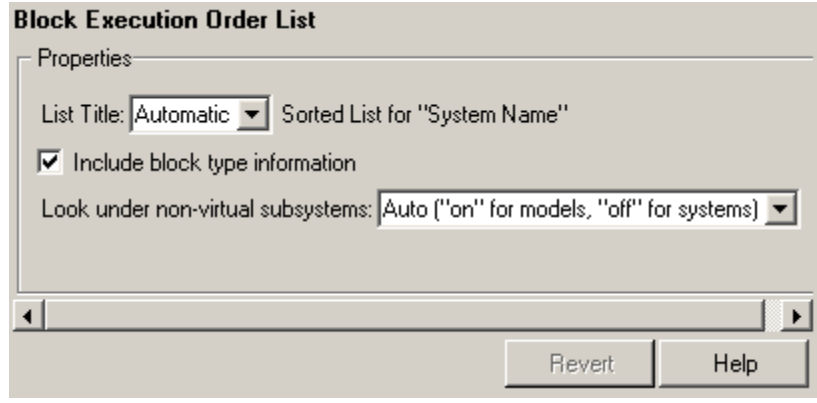
Simulink

Description

The Block Execution Order List component creates a list or table of all nonvirtual blocks in the model, showing the order in which they are executed. For more information about virtual and nonvirtual blocks, see "About Blocks" in the Simulink documentation.

Attributes

The following figure shows the **Block Execution Order List** Properties pane.



List Title

To generate a title automatically, select `Automatic`. To enter a custom title, select `Custom`. Enter a title in this field if you want to have a title for the table or list. If you have a title for a table, it is included in the List of Tables if the stylesheet you chose in the Report Options attribute page supports it.

Include block type information

Select this option if you want to include each block's `BlockType` property in the list or table.

Look under nonvirtual subsystems

The default is `Automatic (On for models, Off for systems)`. You can set it to `On` or `Off`.

Block Execution Order List

Insert Anything into Report?	Yes. List.
File Name	`rptgen_sl.csl_blk_sort_list`

Category

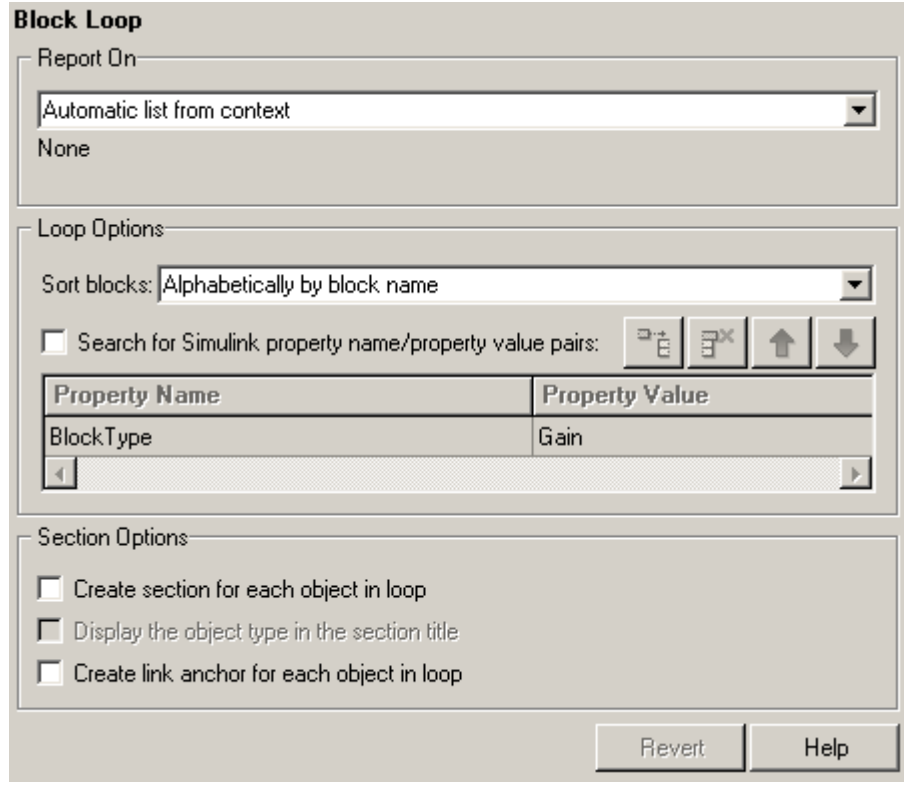
Simulink

Description

The Block Loop component runs its child components for each block contained in the current system, model, or signal. For details about working with looping components, see “Working with Looping Components” on page 7-2.

Attributes

The following figure shows the **Block Loop** Properties pane.



Report On

This pane describes what the Block Loop component is operating on.

Automatic list from context

Select this option to report on all blocks in the current context. The context is set by the Block Loop component's parent component:

- If the parent component is the Model Loop, selecting this option causes the Block Loop component to report on all blocks in the current model.
- If the parent component is a System Loop, selecting this option causes the Block Loop component to report on all blocks in the current system.
- If the parent component is the Signal Loop, selecting this option causes the Block Loop component to report on all blocks connected to the current signal.
- If the Block Loop does not have the Model Loop, System Loop, Signal Loop, or Block Loop component as its parent, selecting this option causes the Block Loop component to report on all blocks in all models.

Custom - use block list

Select this option to report on a list of blocks that you specify and enter the block names in the corresponding field. You must use the full path name when specifying blocks. If you want to sort the blocks in this list, select the `Sort blocks in list` option and then select a sorting option from the `Sort blocks` option (in the **Loop Options** section). Blocks are reported on in the order that they are sorted. If you do not select this option, blocks are reported on in the order that they are listed in the block list.

You can enter `%<VariableName>` if you want to insert the value of a variable from the MATLAB workspace. The %<> notation can be a string or cell array. For more details about this notation, see "%<VariableName> Notation" on page 11-269 on the Text component reference page.

For example, if you want to report on the `theta dot` integrator block and the `theta` integrator block in the demo model `simppend`, you could enter the following in the block list.

```
simppend/theta dot

%<Z>
```

If you define `Z` in the workspace as the following,

```
Z={'simppend/theta'}
```

then Report Generator reports on the following blocks.

```
simppend/theta dot

simppend/theta
```

Loop Options

You can choose block sorting options and reporting options in this pane.

Sort blocks

Use this option to select how blocks are sorted:

- Select `Alphabetically by block name` to sort blocks alphabetically by their names.
- Select `Alphabetically by system name` to sort systems alphabetically.

  In this option, the systems are sorted alphabetically, and blocks in each system are listed, but in no particular order.
- Select `Alphabetically by full Simulink path` to sort blocks alphabetically by Simulink path.
- Select `By block type` to sort blocks alphabetically by block type.
- Select `By block depth` to sort blocks by their depth in the model.

- Select `By traversal order` to sort blocks by traversal order.
- Select `By simulation order` to sort blocks by execution order.

Search for Simulink property name/property value pairs
Select this option to report only on Simulink blocks with certain property name/property value pairs. Enter the property name and the property value in the appropriate fields.

Section Options

Create section for each object in loop
Select this option to insert a section in the generated report for each block found in the loop.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the block in the generated report.

Insert Anything into Report?

Yes, inserts a section if the **Create section for each object in loop** option is selected.

File Name

`rptgen_sl.csl_blk_loop`

Category

Simulink

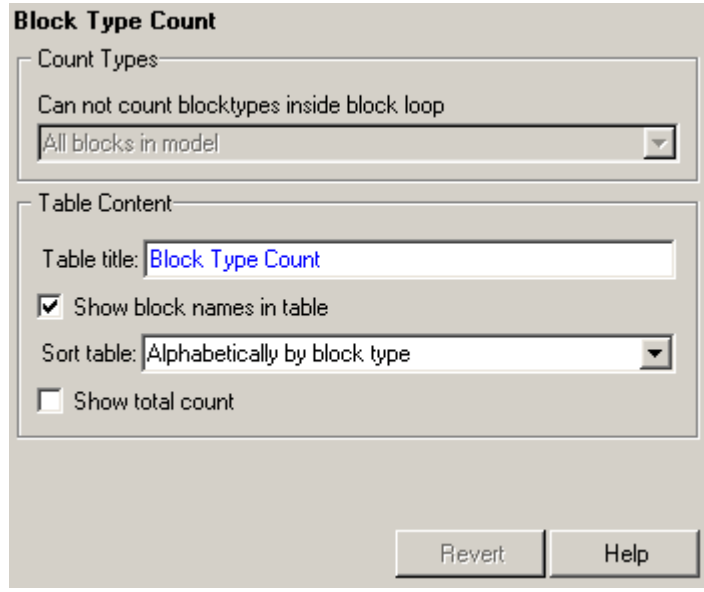
Description

The Block Type Count component counts the number of each block type in the current model or system. Within a model, the Block Type Count component counts blocks underneath masks and inside library links.

For details about block types, see the Simulink documentation.

Attributes

The following figure shows the **Block Type Count** Properties pane.



Count Types

This pane describes where block types are being counted.

- If the Block Type Count component has the Model Loop as its parent, it reports all block types in the current model.
  - Use the `All blocks in model` option to count block types in the entire model.

  - Use the `All blocks in reported systems` option to count block types only in systems that are included in the report.
- If the Block Type Count component has the System Loop as its parent, it reports all block types in the current system.
- The Block Loop and the Signal Loop components are not valid parents for the Block Type Count component.

Table Content

Table title
Enter the title of the table in this field.

Show block names in table
Select this option to include a column that displays all block names in the table.

Sort table
To sort blocks alphabetically by block type, select `Alphabetically by block type`.

To sort by decreasing number of occurrences, select `By number of blocks`.

Show total count
Displays total number of block types for your search.

Insert Anything into Report?

Yes. Table.

File Name

`rptgen_sl.csl_blk_count`

Category

Simulink Blocks

Description

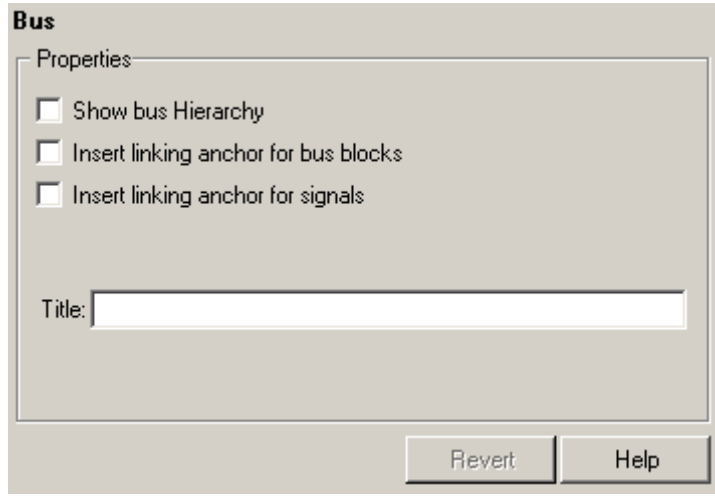
The Bus component creates a list of all signals exiting from a Bus Selector block. The list can contain only those signals leaving from the reported block or it can be hierarchical and display downstream buses and signals.

When working with the Bus component, keep the following points in mind:

- If the Bus component has the Model Loop as its parent, it includes all buses in the current model.
- If the Bus component has the System Loop as its parent, it includes all buses in the current system.
- If the Bus component has the Block Loop as its parent, it includes the current block if it is a bus.
- If the Bus component has the Signal Loop as its parent, it includes all buses connected to the current signal.
- If the Bus component does not have any of the Simulink looping components as its parent, it includes all buses in all open models.

Attributes

The following figure shows the **Bus** Properties pane.



Show bus Hierarchy

Use this option to specify whether or not the list should display downstream buses in a hierarchical form.

Hierarchical list:

```
List Title
   Bus Selector
      signal3
      signal2
      signal4
         Bus Selector1
            signal3
               Bus Selector2
                  signal2
                  signal1
               Bus Selector3
                  signal2
                  signal1
```

Nonhierarchical list:

```
List Title
     Bus Selector
      signal3
      signal2
      signal4
List Title
     Bus Selector1
        signal3
List Title
      Bus Selector2
        signal2
        signal1
List Title
     Bus Selector3
        signal2
        signal1
```

Insert linking anchor for bus blocks

When this option is selected, Report Generator inserts a linking anchor for each bus block. This designates the list item as the location to which other links for that block will point. (See the `Simulink Linking Anchor` or `Link` components for more help.) You should not use this option if you have already specified an anchor location for the bus block with an Object Linking Anchor component.

Insert linking anchor for signals

When this option is selected, Report Generator inserts a linking anchor for each signal. This designates the list item as the location to which other links for that signal will point. (See the `Simulink Linking Anchor` or `Link` components for more help.) You should not use this option if you have already specified an anchor location for the signal with an Object Linking Anchor component.

Title

Use this option to insert a title before each list. Note that this attribute supports the `%<varname>` notation.

Insert Anything into Report?	Yes. List.
File Name	`rptgen_sl.csl_blk_bus`

Category

Formatting

Description

The Chapter/Subsection component groups portions of the report into sections. Each section has a title and some content. The following rules apply to the Chapter/Subsection component:

- Children of the Chapter/Subsection component are placed inside the section created by this component.
- The Chapter/Subsection component takes paragraph-level children unless you select the **Get title from first child component** check box. In this case, a Chapter/Subsection component's first child must be a Text component.
- The Chapter/Subsection component can be the child of another Chapter/Subsection component.

Sections can be nested, and titles automatically become smaller inside subsections. There are seven levels of nesting possible.

```
Chapter
 Section 1
  Section 2
   Section 3
    Section 4
     Section 5
      Section 6
        No Section
```

Any Chapter/Subsection components beyond the sixth section do not create a titled section, although the child components of these sections do insert information into the report.

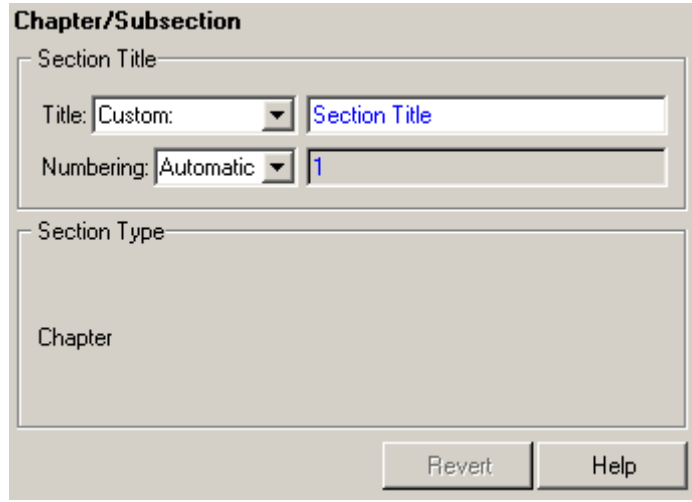
Chapter Numbering

By default, chapters are numbered and sections are not numbered. Chapter and section numbering are specified by the stylesheet you choose in the Report Options attribute page. See "Setting the Report

Format" on page 5-3 for chapter and section numbering options in Web and print stylesheets.

Attributes

The following figure shows the **Chapter/Subsection** Properties pane.



Section Title

Title

Choose a title for the display in the generated report. To generate a title automatically, select `Automatic`. To enter a title, select `Custom`.

Numbering

Choose a numbering style for the report. To number by context, select `Automatic`. To create your own numbering style, select `Custom`.

Section Type

The item in the **Section Type** pane shows you what level the selected section is residing in.

Insert Anything into Report? Yes. Chapter or section.

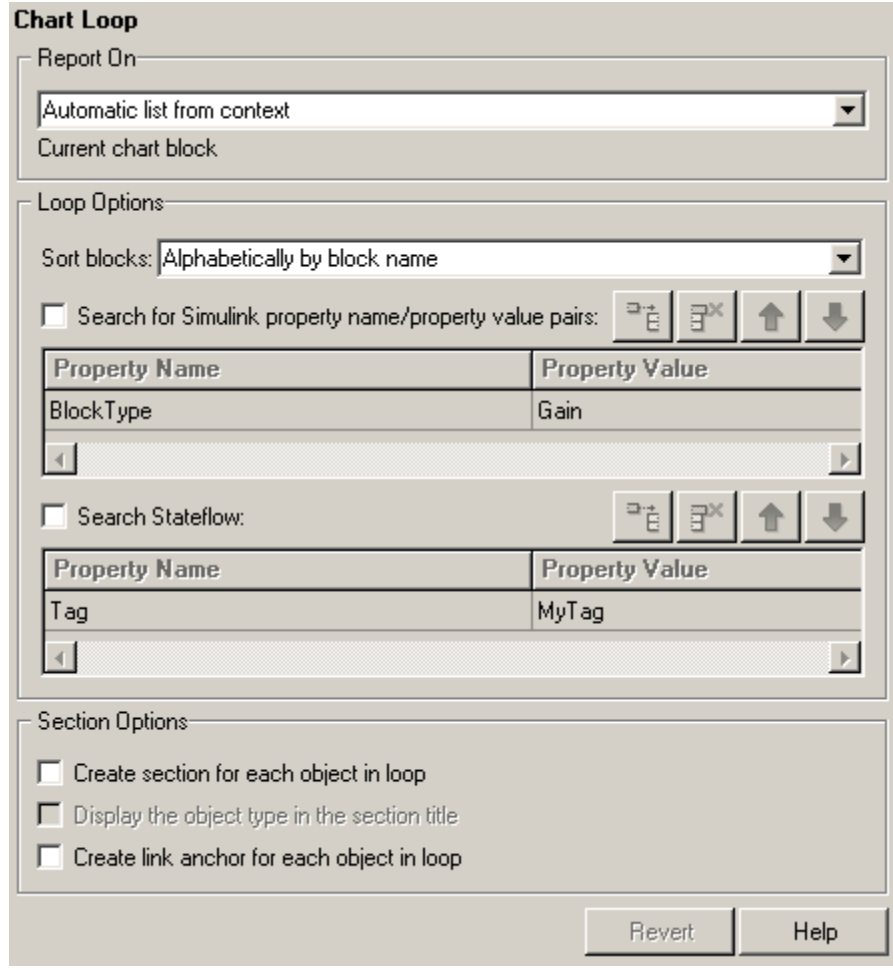
File Name `rptgen.cfr_section`

Chart Loop

Category Stateflow

Description The Chart Loop component runs its children for each Stateflow chart that you choose. The Chart Loop component can have the Model Loop, System Loop, Signal Loop, or Block Loop component as its parent.

Attributes The following figure shows the **Chart Loop** properties pane.



Report On

Automatic list from context

Select this option to report on all chart blocks in the current context. The context is set by the Chart Loop component's parent component:

- If the parent component is the Model Loop, selecting this option causes the Chart Loop component to report on all reported Stateflow chart blocks in the current model.
- If the parent component is the System Loop, selecting this option causes the Chart Loop component to report on all Stateflow chart blocks in the current system.
- If the parent component is the Signal Loop, selecting this option causes the Chart Loop component to report on all Stateflow chart blocks connected to the current signal.
- If the parent component is the Machine Loop, selecting this option causes the Chart Loop component to report on the current Simulink block if it is a Stateflow chart.
- If the Chart Loop does not have the Model Loop, System Loop, Signal Loop, or Block Loop component as its parent, selecting this option causes the Chart Loop component to report on all Stateflow chart blocks.

Custom - use block list

Select this option to report on a list of Stateflow chart blocks that you specify, and enter the chart block names in the corresponding field. If you want to sort the chart blocks in this list, select the **Sort blocks in list** option and then select a sorting option from the **Sort blocks** option (in the **Loop Options** section). Chart blocks are reported on in the order that they are sorted. If you do not select this option, chart blocks are reported on in the order that they are listed in the chart list.

You can enter `%<VariableName>` if you want to insert the value of a variable from the MATLAB workspace. The %<> notation can be a string or cell array. For more details about this notation, see "%<VariableName> Notation" on page 11-269 on the Text component reference page.

Loop Options

You can choose chart block sorting options and reporting options in this pane.

Sort blocks

Use this option to select how blocks are sorted (this option is available if you select the `Automatic list from context` option in the **Report On** section or if you select `Custom - use block list` and the `Sort blocks` option).

- Select `Alphabetically by block name` to sort blocks alphabetically by name.
- Select `Alphabetically by system name` to sort systems alphabetically by name.
- In this option, the systems are sorted alphabetically, and blocks in each system are listed, but in no particular order.
- Select `Alphabetically by full Simulink path` to sort models alphabetically by their full path name.
- Select `By block type` to sort blocks alphabetically by block type.
- Select `By depth` to sort blocks by their depth in the model.
- Select `By simulation order` to sort blocks by execution order.

Search for Simulink property name/property value pairs

Select this option to report only on Simulink blocks with certain property name/property value pairs. Enter the property name and the property value in the appropriate fields.

Search Stateflow

Select this option to report only on Stateflow charts with certain property name/property value pairs. Enter the property name and the property value in the appropriate fields.

Chart Loop

Section Options

Create section for each object in loop
Select this option to insert a section for each object found in the loop in the generated report.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the object in the generated report.

Insert Anything into Report?

Yes, inserts a section if the **Create section for each object in loop** option is selected.

File Name

`rptgen_sf.csf_chart_loop`

Category

Real-Time Workshop

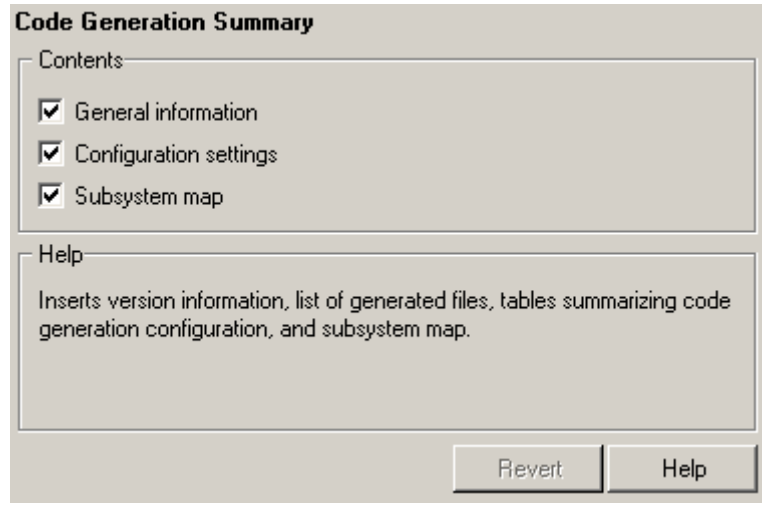
Description

The Code Generation Summary component reports

- Version information
- List of generated files
- Code generation configuration information
- Subsystem map

Attributes

The following figure shows the **Code Generation Summary** Properties pane.



General information

Select this option to include the following information in the report:

- `Model name and version`
- `Real-Time Workshop version`

- `List of full paths for generated files`

Configuration settings
Select this option to include in the report tables that list optimization and Real-Time Workshop target selection and build process Configuration Parameter settings.

Subsystem map
If your model includes at least one subsystem, select this option to include in the report a mapping between subsystem numbers, uniquely assigned by Simulink, and subsystem labels in the model.

Insert Anything into Report?

Yes. Tables and list.

File Name

`RptgenRTW.CCodeGenSummary`

Category

Report Generator

Description

The Comment component inserts a comment into the XML source file created by the report generation process. The comment is not visible in the generated report.

This component can have children. Child components insert their output into the XML source file, but this output appears inside comment tags and does not appear in the final report.

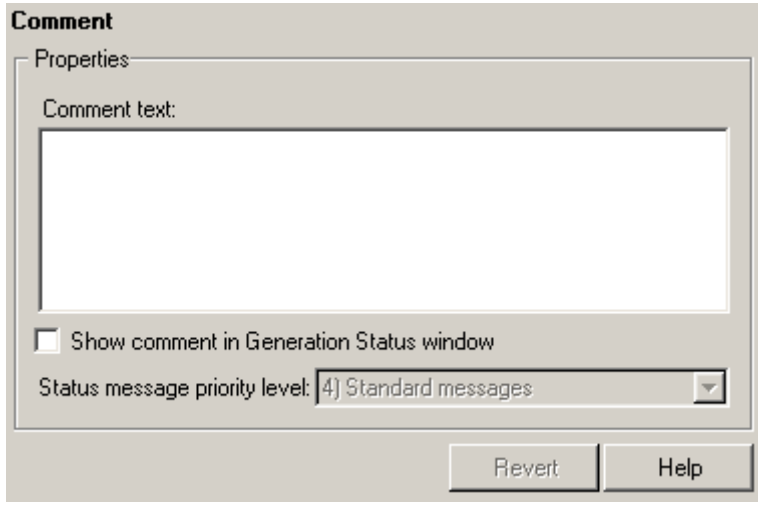
If you want the comment text to appear in the report, follow these steps:

1. Edit the XML source file (the XML source file has the same name as your report file, but has a `.xml` extension). Note that you must generate a report to create the XML source file.
2. Find the comment area in the XML source file by locating the comment tags `<--` and `-->`. Remove both of these tags.
3. Convert the XML source file using the `rptconvert` command (for more details on the `rptconvert` command, see "Converting XML Documents" on page 5-10).

Attributes

The following figure shows the **Comment** Properties pane.

Comment



Comment text
Enter the comments that you want to include in the report.

Show comment in Generation Status window
Select this option if you want to display the comment in the **Generation Status** tab when the report is generating.

Status message priority level
This option is only available if you select the **Show comment in Generation Status window** option. Select the status message priority level that appears during report generation. The priority options range from `1) Error messages` only to `6) All messages`. The default is `4) Standard messages`.

Insert Anything into Report?

No, but this component inserts a comment into the XML source file, which can appear in the report.

File Name

`rptgen.crg_comment`

Category

Simulink Blocks

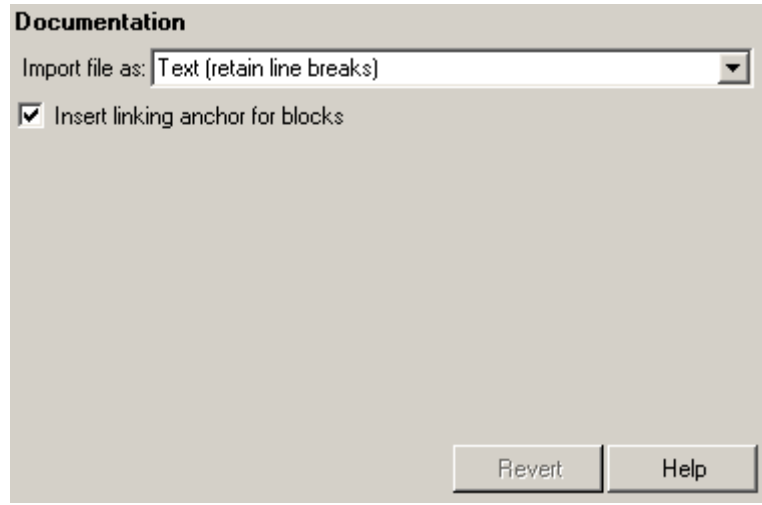
Description

The Documentation component inserts text extracted from doc blocks in Simulink models. The Documentation component can have the Model Loop, the System Loop, or the Block Loop component as its parent. For example, you can insert doc blocks into your report in succession by specifying a System loop as the parent of a documentation component.

If the report format is set to HTML, you can import HTML documentation. If the report format is set to RTF, you can import RTF documentation.

Attributes

The following figure shows the **Documentation** Properties pane.



Import file as

Indicate what kind of format to use for the imported file. There are five formats to choose from. To illustrate how they each work, the following text is used as the input text:

First row.
  Second row.

Third row follows blank line.

- `Plain text (ignore line breaks)`: Select this option to import plain text without any line breaks (no paragraphs). The following output is produced:

  First row. Second row. Third row follows blank line.

- `Paragraphs defined by line breaks`: Select this option to import the text contained in paragraphs defined by line breaks (hard returns or carriage returns). The following output is produced:

  First row.

  Second row.

  Third row follows blank line.

- `Paragraphs defined by empty rows`: Select this option to import text contained in paragraphs defined by empty rows (rows that do not contain text). The following output is produced:

  First row. Second row.

  Third row follows blank line.

- `Text (retain line breaks)`: Select this option to import plain text, including line breaks. The following output is produced:

  First row.
    Second row.

  Third row follows blank line.

- `Fixed-width text (retain line breaks)`: Select this option to import fixed-width text (all letters have the same width

or size) including line breaks. This option is best suited for importing M-files. The following output is produced:

```
First row.
  Second row.

Third row follows blank line.
```

Insert linking anchor for blocks

Select this option to insert a linking anchor for each doc block. This designates the location to which other links for that block will point. (See the `Simulink Linking Anchor` or `Link` components for more help.) You should not use this option if you have already specified an anchor location for the doc block with an Object Linking Anchor component.

Insert Anything into Report?

Yes. Text, paragraph, or external RTF/HTML.

File Name

`rptgen_sl.csl_blk_doc`

Empty Component

Category

Report Generator

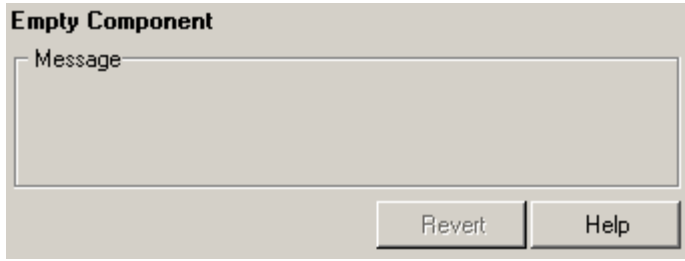
Description

The Empty Component does not insert anything into the report and can have any component as a child. You can use it to group components together so that the components can be moved or activated/deactivated easily, or to create a blank space in a list.

If Report Generator does not recognize a component when loading a report template, it replaces the unrecognized component with the Empty Component.

Attributes

The following figure shows the **Empty Component** Properties pane:



Insert Anything into Report?

No.

File Name

`rptgen.crg_empty`

Category

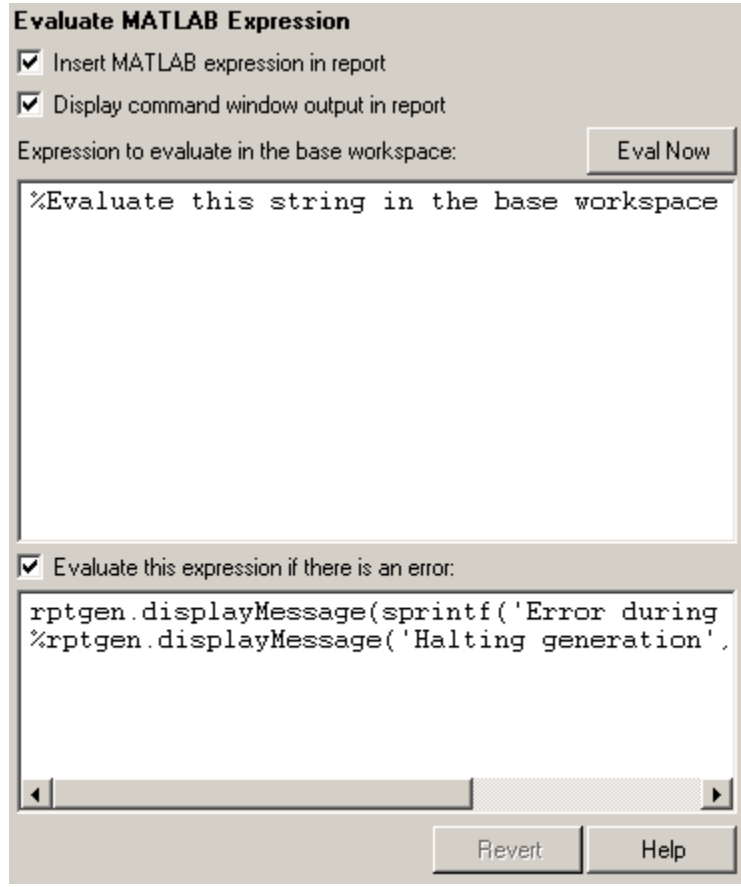
MATLAB

Description

The Evaluate MATLAB Expression component evaluates a specified MATLAB expression in the workspace. The code and/or command-line output can be inserted into the report.

Attributes

The following figure shows the **Evaluate MATLAB Expression** Properties pane.



Insert MATLAB expression in report
Select this option if you want the MATLAB expression that is evaluated by this component to appear in the report.

Display command window output in report
Select this option if you want to include the command window output that results when the MATLAB expression is evaluated.

Expression to evaluate in the base workspace
Enter the expression that you want to evaluate in the MATLAB workspace.

You can include this attribute in a loop component, such as a Block Loop, a Model Loop, a Signal Loop, or a System Loop to obtain a handle to the object in whose context the loop is executing:

- Enter the following if you want to obtain a handle to the block in whose context a Block Loop is executing:

```
%Access to block
blockname = RptgenSL.getReportedBlock;
blockHandle = get_param(blockname,'handle');
```

- Enter the following if you want to obtain a handle to the model in whose context a Model Loop is executing:

```
%Access to model
modelname = RptgenSL.getReportedModel;
modelHandle = get_param(Modelname,'handle');
```

- Enter the following if you want to obtain a handle to the signal in whose context a Signal Loop is executing:

```
%Access to signal
signalname = RptgenSL.getReportedSignal;
signalHandle = get_param(signalname,'handle');
```

- Enter the following if you want to obtain a handle to the system in whose context a System Loop is executing:

```
%Access to system
systemname = RptgenSL.getReportedSystem;
systemHandle = get_param(systemname,'handle');
```

Similarly, you can use `set` calls in place of the `get` calls in the previous examples to set your current block, model, signal, or system within the context of a loop component.

Evaluate this expression if there is an error

Select this option if you want to evaluate another MATLAB expression if the evaluated expression produces an error. Enter the expression that should be evaluated in case of an error in this field.

Insert Anything into Report?

Inserts text if you select one of the following options:

- `Insert MATLAB expression string in report`
- `Display command window output in report`

Otherwise, this component does not insert anything into the report.

File Name

`rptgen.cml_eval`

Figure Loop

Category

Handle Graphics

Description

The Figure Loop component applies each child component to graphics figures specified in the attribute page.

See “Working with Looping Components” on page 7-2 for a discussion of this component.

Attributes

The following figure shows the **Figure Loop** Properties pane.

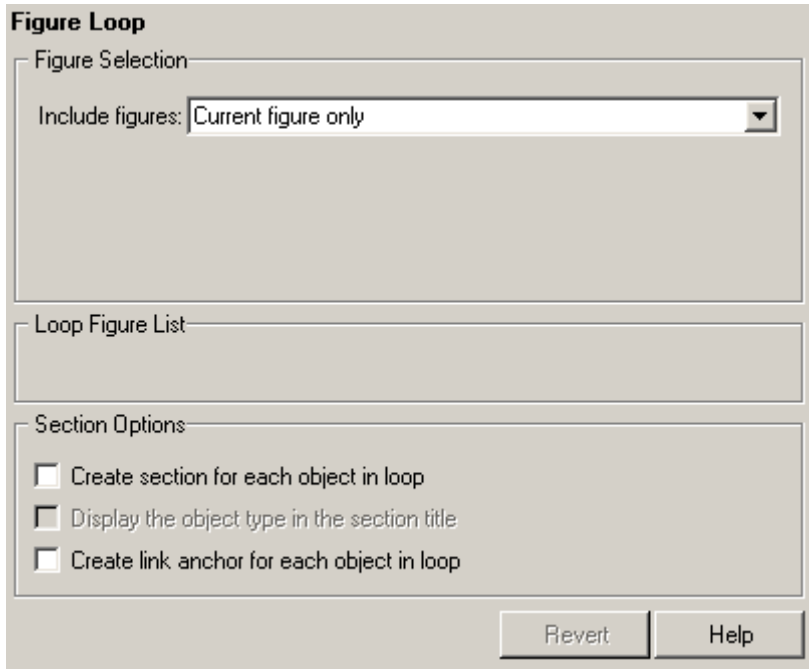


Figure Selection

You can choose to loop on the following:

- The current figure
- All visible figures

- Figures with certain tags

Include figures

To include only the current figure in the report, select the `Current figure only` option.

To loop on all visible figures, select the `Visible figures` option. This option executes child components for all figures that are currently open and visible. The `Data figures only` option is checked by default and excludes figures with `HandleVisibility = 'off'` from the loop.

To loop on figures with certain tags, select the `All figures with tags` option. When you are selecting a tag, all figures with that tag are included, regardless of whether the figure is visible or whether `HandleVisibility` is `'on'` or `'off'`. The tag list box (located under the `All figures with tags` option) shows the selected tags. To add a tag to the tag list box, select the tag from the list on the right and click **Add**. The tag list shows a list of tags for figures that are open. If you want to add a tag that does not appear in the tag list, you can type in the tag.

Loop Figure List

The **Loop Figure List** shows you the figures that would be looped on given the current set of figures.

Note If your report template generates new figures or changes existing figures, the figures in the **Loop Figure List** will not be the figures that are actually reported on.

Section Options

Create section for each object in loop

Select this option to insert a section in the generated report for each object found in the loop.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the object in the generated report.

Insert Anything into Report?

Yes, inserts a section if the **Create section for each object in loop** option is selected.

File Name

`rptgen_hg.chg_fig_loop`

Category

Handle Graphics

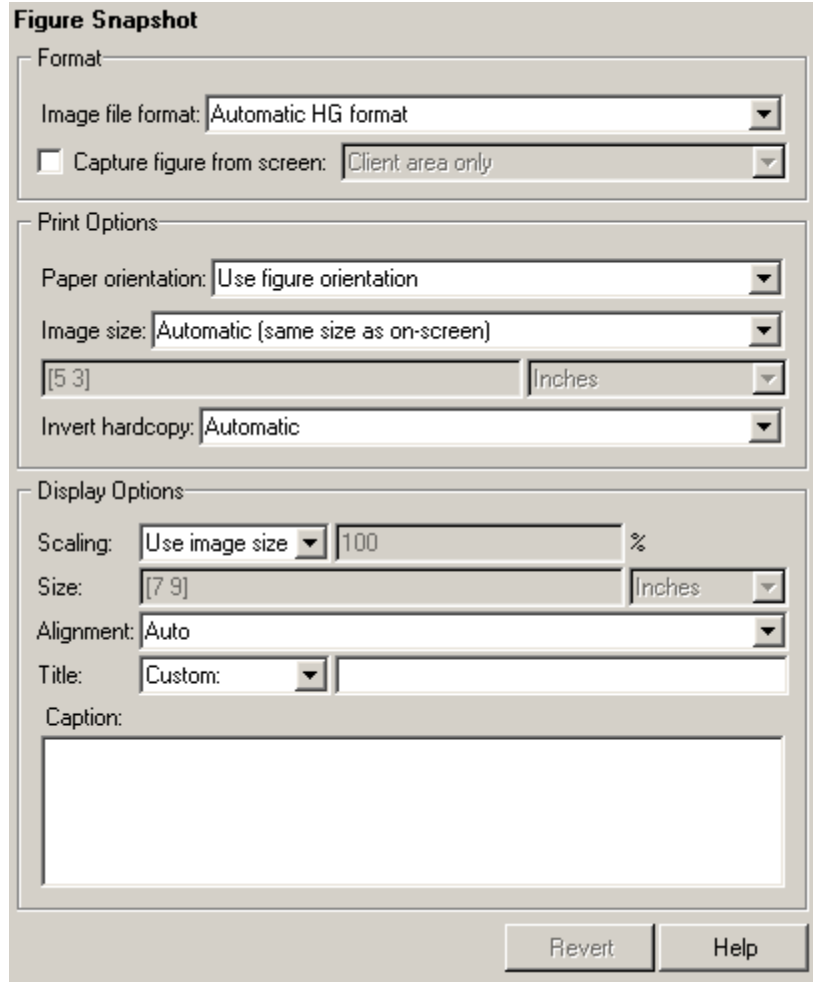
Description

The Figure Snapshot component inserts a snapshot of a Handle Graphics figure.

Attributes

The following figure shows the **Figure Snapshot** Properties pane.

Figure Snapshot



Format

Image file format

Choose the image file format (e.g., JPEG, TIFF, etc.) from this list. Select `Automatic HG Format` to automatically choose the format best suited for the output format that you chose in the Report component. Otherwise, choose an image format that your output

viewer can read. `Automatic HG Format` is the default option. The options include

- `Automatic HG Format` (uses the Handle Graphics file format selected in the Preferences dialog box)
- `Adobe Illustrator`
- `Bitmap (16m-color)`
- `Bitmap (256-color)`
- `Black and white encapsulated PostScript`
- `Black and white encapsulated PostScript (TIFF)`
- `Black and white encapsulated PostScript2`
- `Black and white encapsulated PostScript2 (TIFF)`
- `Black and white PostScript`
- `Black and white PostScript2`
- `Color encapsulated PostScript`
- `Color encapsulated PostScript (TIFF)`
- `Color encapsulated PostScript2`
- `Color encapsulated PostScript2 (TIFF)`
- `Color PostScript`
- `Color PostScript2`
- `JPEG high quality image`
- `JPEG medium quality image`
- `JPEG low quality image`
- `PNG 24-bit image`
- `TIFF - compressed`
- `TIFF - uncompressed`

- `Windows metafile`

Capture picture from screen
Use the option to capture a figure from the screen. Choose `Client area only` to capture a portion of the figure window, or `Entire figure window` to capture the whole window.

Print Options

Use the attributes in this section to set printing parameters.

Paper orientation
You can choose `Landscape`, `Portrait`, `Rotated`, or `Use figure orientation`. See the `orient` command in the MATLAB Function Reference documentation for information about paper orientation.

Image size
Select `Use figure PaperPositionMode setting` to use the `PaperPositionMode` property of the Handle Graphics figure to set the image size in the report. See the `orient` command in the MATLAB Reference Manual for information about paper position mode.

Select the `Automatic (same size as on screen)` option if you want the image in your report to be the same size as the image on the screen.

Select the `Custom` option to choose a custom image size. If you choose this option, specify the image size in the **Size** field and **Units** list.

- **Size**: Enter the size of the Handle Graphics figure snapshot in the form `wxh` (width times height). This field is active only if you choose `Custom` in the **Image size** list box.
- **Units**: Enter the units for the size of the Handle Graphics figure snapshot. This field is active only if you choose `Custom` in the **Image size** list box.

Invert hardcopy

Handle Graphics figures have an `InvertHardcopy` property. This parameter inverts colors for printing; i.e., it changes dark colors to light colors and vice versa.

Select one of the following options:

- `Automatic` — Select this option to automatically change dark axes colors to a light axes color. If the axes color is a light color, it will not be inverted.
- `Invert` — Select this option to change dark axes colors to light axes colors and vice versa.
- `Don't invert` — Select this option if you do not want to change the colors in the image displayed on the screen for printing.
- `Use figure's InvertHardcopy setting` — Select this option to use the `InvertHardcopy` property set in the Handle Graphics image.
- `Make figure background transparent` — Select this option to make the image background transparent.

Display Options

Scaling

Select `Fixed size` to specify the number and type of units.

Select `Zoom` to specify the percentage, the maximum size, and the units of measure.

Select `Use image size` if you want the image in your report to be the same size as the image on the screen.

Size

Enter the exact size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Fixed size` in the **Scaling** drop-down list.

Max size
Enter the maximum size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Zoom` from the **Scaling** drop-down list.

Units
Enter the units for the size of the snapshot. This field is active only if you choose `Zoom` or `Fixed size` in the **Image size** list box.

Alignment
Options are `Auto`, `Right`, `Left`, and `Center`.

Note This option is supported only in `PDF` and `RTF` reports.

Title
Enter a title for the snapshot figure. Select Custom to enter a custom title. Select Name to specify the figure name as the title.

Caption
Type the text you want to appear below the snapshot.

Insert Anything into Report?

Yes. Image.

File Name

`rptgen_hg.chg_fig_snap`

Category

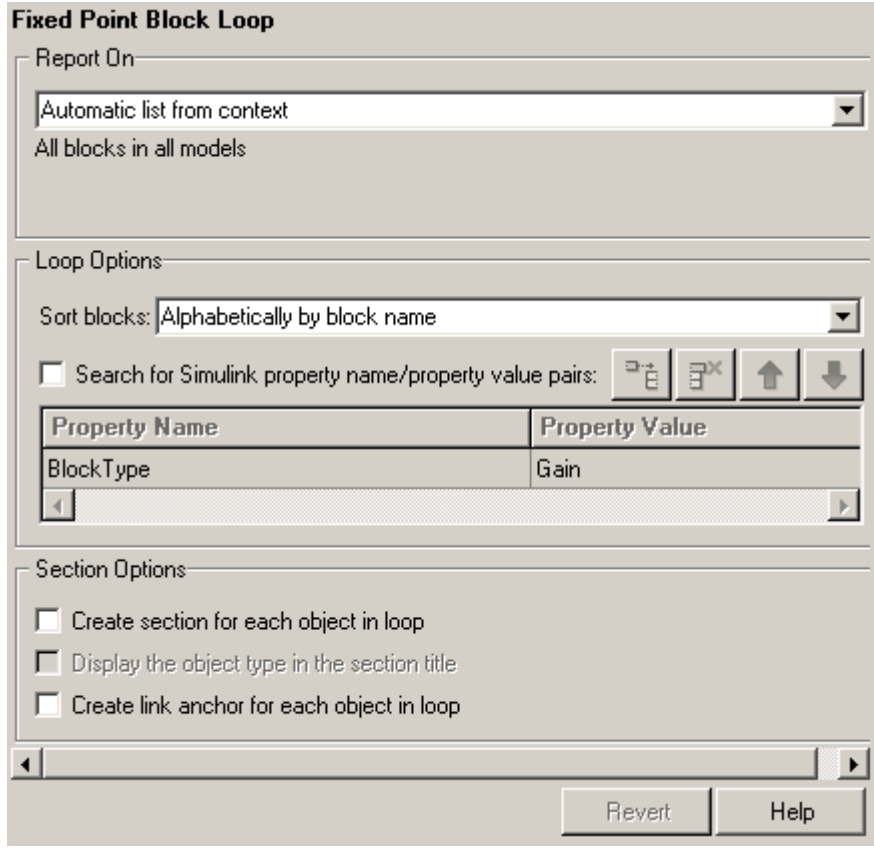
Simulink Fixed Point

Description

The Fixed Point Block Loop component runs its children for the Simulink model, system, or signal defined by its parent component. For example, if the Fixed Point Block Loop is parented by the Simulink Model Loop, it runs its children for all fixed-point blocks in the Simulink model. The Fixed Point Block Loop component can have the Simulink Model Loop, System Loop, or Signal Loop component as its parent.

Attributes

The following figure shows the **Fixed Point Block Loop** Properties pane.



Report On

Automatic list from context

Select this option to report on all fixed-point blocks in the current context. The context is set by the Fixed Point Block Loop component's parent component. For example, if the parent component is the Simulink System Loop, this component reports on all fixed-point blocks in the current system.

If the Fixed Point Block Loop component does not have a Simulink Model Loop, System Loop, Block Loop, or Signal Loop component

as its parent, selecting this option causes the component to report on all fixed-point blocks in all models.

Custom - use block list:

Select this option to report on a list of blocks that you specify, and enter the block names in the corresponding field. If you want to sort the blocks in this list, select the `Sort blocks in list` option and then select a sorting option from the **Sort blocks** list (in the **Loop Options** section). Blocks are reported on in the order that they are sorted. If you do not select this option, blocks are reported on in the order that they are listed in the block list.

You can enter `%<VariableName>` if you want to insert the value of a variable from the MATLAB workspace. The %<> notation can be a string or cell array. For more details about this notation, see "%<VariableName> Notation" on page 11-269 on the Text component reference page.

For example, if you want to report on the `A2D` block and the `D2A` block in the demo model `fxpdemo_feedback`, you could enter the following in the block list:

```
A2D

Z
```

If you define `Z` in the workspace as

```
Z={'fxpdemo_feedback/D2A'}
```

then Report Generator reports on the following blocks:

```
A2D

D2A
```

Loop Options

You can choose block sorting options and reporting options in this pane.

Sort blocks

Use this option to select how blocks are sorted (this option is available if you select the `Automatic list from context` option in the `Report On` section or if you select `Custom - use block list` and the `Sort blocks` option).

- Select `Alphabetically by block name` to sort blocks alphabetically by their names.
- Select `Alphabetically by system name` to sort systems alphabetically.

  In this option, the systems are sorted alphabetically, and blocks in each system are listed, but in no particular order.
- Select `By fixed point block type` to sort blocks alphabetically by block type.
- Select `By fixed point block depth` to sort blocks by their depth in the model.
- Select `By traversal order` to sort blocks by traversal order.
- Select `By simulation order` to sort blocks by execution order.

Search for Simulink property name/property value pairs

Select this option to report only on blocks with certain property name/property value pairs. Enter the property name and the property value in the appropriate fields.

Section Options

Create section for each object in loop

Select this option to insert a section in the generated report for each object found in the loop.

Display the object type in the section title

Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the object in the generated report.

Insert Anything into Report?

Yes, inserts a section if the **Create section for each object in loop** option is selected.

File Name

`rptgen_fp.cfp_blk_loop`

Fixed Point Logging Options

Category

Simulink Fixed Point

Description

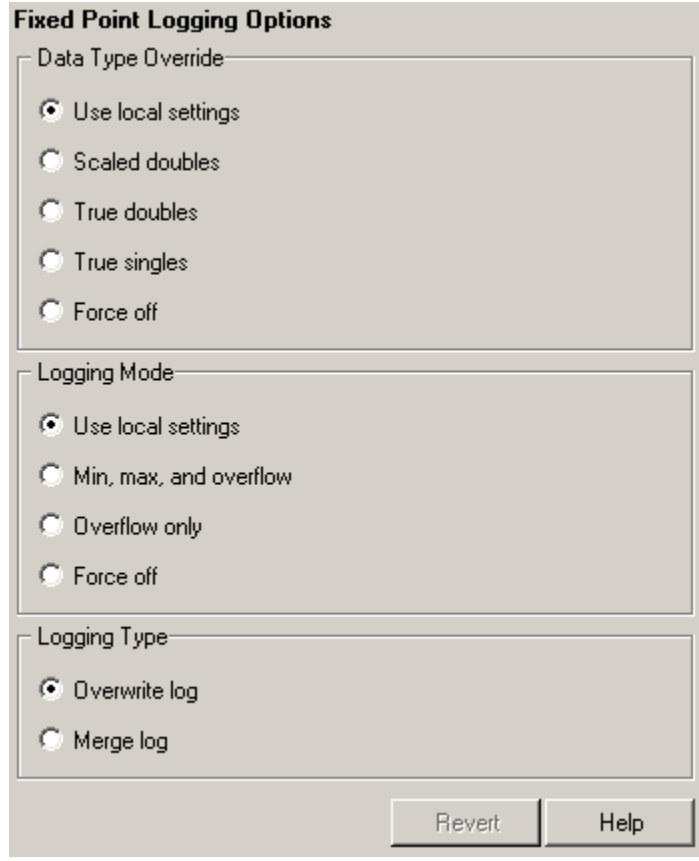
The Fixed Point Logging Options component sets fixed-point options similar to those set in the Fixed Point Settings interface. You can set the following options with this component:

- Logging options
- Doubles override
- Log mode

You can use this component before a Model Simulation component in order to control how the model behaves during simulation.

Attributes

The following figure shows the **Fixed Point Logging Options** Properties pane.



Data Type Override

Use local settings

Data types are overridden according to the value of this parameter set for each subsystem. Otherwise, settings for parent systems override those of child systems.

Scaled doubles

The output data type of all blocks in the current system or subsystem is overridden with doubles; however, the scaling and bias specified in the mask of each block is maintained.

True doubles
The output data type of all blocks in the current system or subsystem is overridden with true doubles. The overridden values have no scaling or bias.

True singles
The output data type of all blocks in the current system or subsystem is overridden with true singles. The overridden values have no scaling or bias.

Force off
No data type override is performed on any block in the current system or subsystem.

Logging Mode

You can select logging options in this section. If a block is logged, minimum and maximum simulation values are logged to the workspace. See the Simulink Fixed Point documentation for details about logging simulation results.

Use local settings
Data is logged according to the value of this parameter set for each subsystem. Otherwise, settings for parent systems always override those of child systems.

Min, max, and overflow
Minimum value, maximum value, and overflow data is logged for all blocks in the current system or subsystem.

Overflow only
Only overflow data is logged for all blocks in the current system or subsystem.

Force off
No data is logged for any block in the current system or subsystem. Use this selection to work with models containing fixed point-enabled blocks if you do not have a Simulink Fixed Point license.

Logging Type

Select how you want logs to be recorded in this section.

Overwrite log
Information in the logs is cleared before new logging data is entered.

Merge log
New logging data is merged with previously logged information.

Insert Anything into Report?

No.

File Name

`rptgen_fp.cfp_options`

Fixed Point Property Table

Category

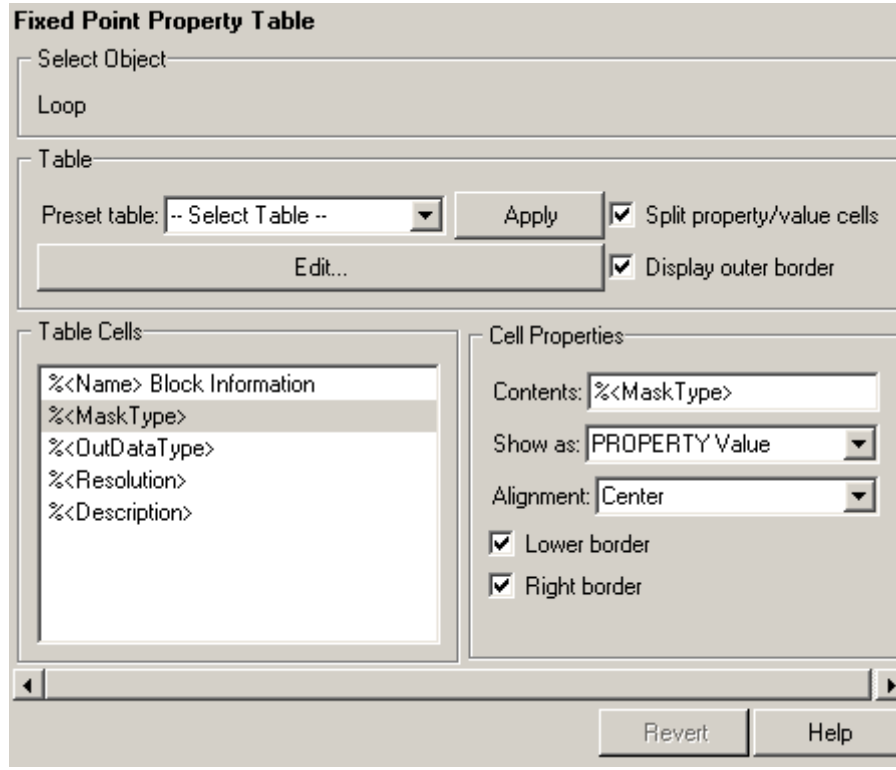
Simulink Fixed Point

Description

The Fixed Point Property Table inserts a table that reports on fixed-point block property name/property value pairs.

Attributes

The following figure shows the **Fixed Point Property Table** Properties pane.



Table

A preset table is a table that is already formatted and set up. You can select a preset table in the preset table list in the upper-left corner of

the attributes page. To apply a preset table, select the table and click **Apply**.

Preset table

Choose a type of table to display the object property table:

- `Default`
- `Mask properties`
- `Block limits`
- `Out-of-range errors`
- `All fixed-point properties`
- `Blank 4x4`

Split property/value cells

Select this option to split property name/property value pairs into separate cells. To have the property name and property value appear in adjacent horizontal cells in the table, select the **Split property/value cells** check box. In this case, the table is in split mode and there can be only one property name/property value pair in a cell. If you have more than one name/property pair in a cell, only the first pair appears in the report. All subsequent pairs are ignored.

To have the property name and property value appear together in one cell, clear the **Split property/value cells** check box. This is nonsplit mode. In nonsplit mode, you can have more than one property name/property value pair and you can also have text.

If you want to switch from nonsplit mode to split mode, make sure that you have only one property name/property value pair per table cell before you switch modes. If you have more than one property name/property value pair or any text, only the first property name/property value pair appears in the report; subsequent pairs and text are omitted.

Display outer border
Select this option to display the outer border of the table in the generated report.

Table Cells

Use this field to select table properties to modify. The selection in this pane affects the available fields in the **Cell Properties** pane.

Cell Properties

Contents
Use this field to modify the contents of the table cell selected in the **Table Cells** pane.

Show as
Choose the format for the contents of the table cell:

- `PROPERTY Value`
- `Value`
- `Property Value`
- `Property:  Value`
- `PROPERTY: Value`
- `Property - Value`
- `PROPERTY - Value`

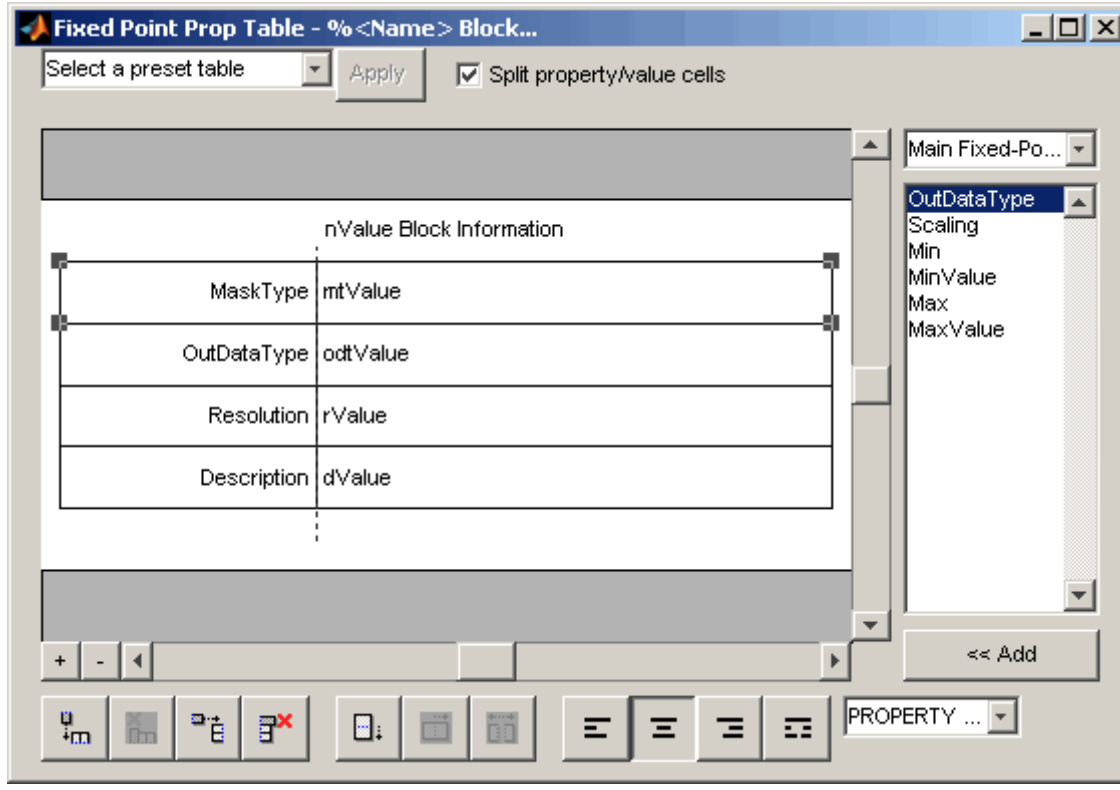
Alignment
Use this option to align the contents of the table cell. Options are `Center`, `Left`, `Right`, and `Double justified`.

Lower border
Select this option to display the lower border of the table in the generated report.

Right border
Select this option to display the right border of the table in the generated report.

Creating Custom Tables

To create a custom table, you can first select a preset table and then edit it. You might want to start with the `Blank 4x4` preset table. You can add and delete rows and you can add properties. To open the Edit Table dialog box, click **Edit**.



For details about using this dialog box to create custom property tables, see “Working with Property Table Components” on page 7-14.

Fixed Point Property Table

Insert Anything into Report?	Yes. Table.
File Name	`rptgen_fp.cfp_prop_table`

Category

Simulink Fixed Point

Description

The Fixed Point Summary Table component shows properties or parameters of the selected fixed-point blocks in a table.

Attributes

The following figure shows the **Fixed Point Summary Table** Properties pane.

Fixed Point Summary Table

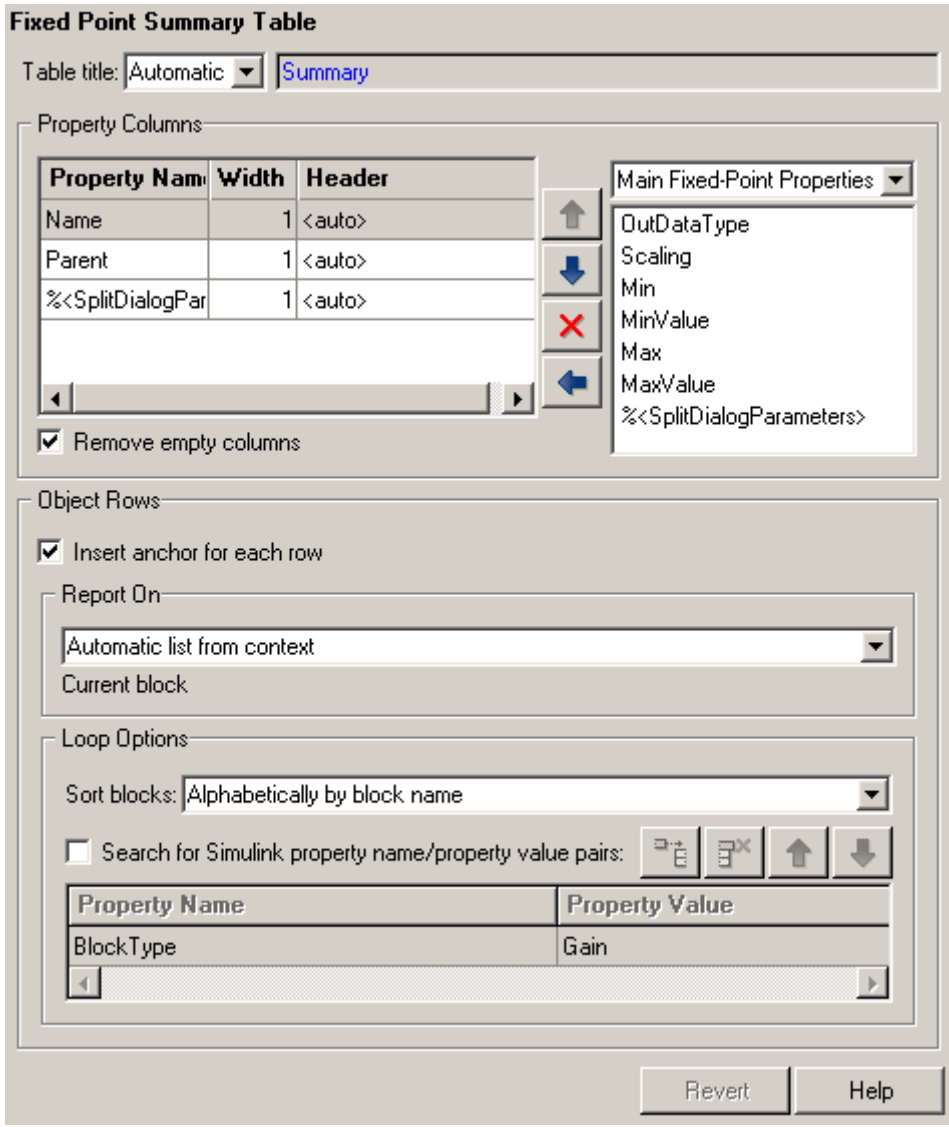


Table title

Choose a title for the summary table in the generated report:

- `Automatic` — Select this option to generate a title automatically from the parameter.
- `Custom` — Select this option to enter a title manually.

Property Columns

Property name

This field displays the object properties to be included in the summary table in the generated report. To add a property, select the appropriate property level in the menu. Next, in the contextual list below the menu, select the property that you want to add and click **Add** . To delete a property, select the property name and click the **Delete** button. To move properties up and down in the list, click the **Up** and **Down** buttons.

Note Some entries in the list of available properties (such as `Depth`) are "virtual" properties which cannot be accessed using the `get_param` command. The properties used for property/value filtering in the block and system loop components must be retrievable by the `get_param`. Therefore, you cannot configure your summary table to report on all blocks of `Depth == 2`.

Object Rows

Insert anchor for each row

Select this option to insert an anchor for each row in the summary table.

Report On

Use this option to select which blocks are reported on.

- `Automatic list from context` — Select this option to report on all blocks in the current context.
- `Custom - use block list` — Select this option to report on a list of blocks that you specify, and enter the block names in

the corresponding field. You must use the full path name when specifying blocks.

Loop Options

Sort blocks

Use this option to select how blocks are sorted:

- Select `Alphabetically by block name` to sort blocks alphabetically by their names.
- Select `Alphabetically by system name` to sort systems alphabetically.

  In this option, the systems are sorted alphabetically, and blocks in each system are listed, but in no particular order.
- Select `Alphabetically by full Simulink path` to sort blocks alphabetically by Simulink path.
- Select `By block type` to sort blocks alphabetically by block type.
- Select `By block depth` to sort blocks by their depth in the model.
- Select `By traversal order` to sort blocks by traversal order.
- Select `By simulation order` to sort blocks by execution order.

Search for Simulink property name/property value pairs

Select this option to report only on Simulink blocks with certain property name/property value pairs. Enter the property name and the property value in the appropriate fields.

Insert Anything into Report?

Yes. Table.

File Name

`rptgen_fp.cfp_summ_table`

Category

Logical and Flow

Description

The For Loop component iteratively executes its child components and must have at least one child component to execute properly. This component functions like the `for` loop in MATLAB, except that instead of executing a statement, it executes its child components while one of the following loops is active:

```
for varname=x:y:z
for varname=[a b c ...]
```

where `varname` must be a string that is valid as a variable name. `x`, `y`, `z` can be scalars or workspace expressions that are evaluated in the workspace as scalar numbers. If any one of these is a string that cannot be evaluated as a scalar, the For Loop does not execute.

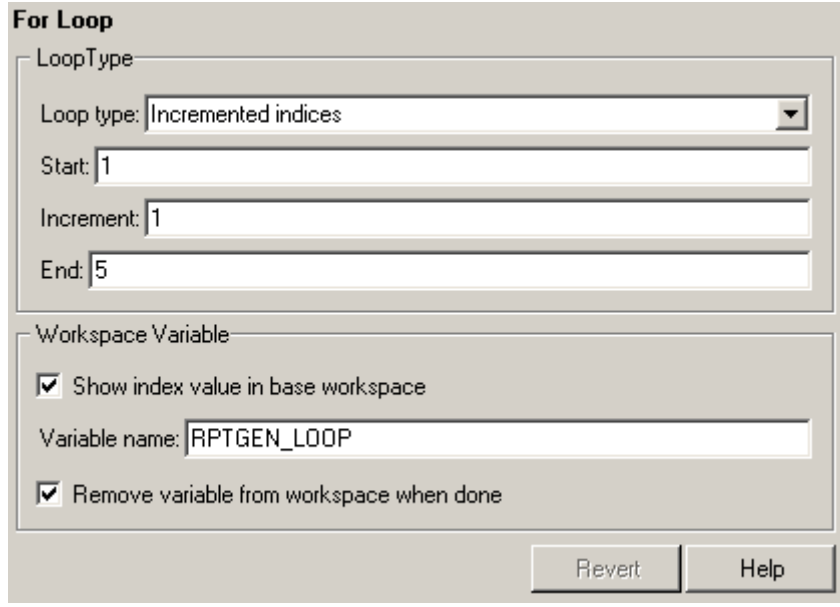
For example, if `red` is a workspace variable, the following is a valid `for` statement.

```
for i=1:1:length(red)
```

`length(red)` is 3, which is a scalar.

Attributes

The following figure shows the **For Loop** Properties pane.



Loop Type

In this section you can choose the loop type to be incremented indices or a vector of indices. See the MATLAB documentation for more details on the for loop and indices.

Incremented indices

Choose this option to execute a `for` loop of the following form:

```
for varname=x:y:z
```

Enter the appropriate values in the following fields:

- **Start** corresponds to `x` in the `for` loop shown above.
- **Increment** corresponds to `y` in the `for` loop shown above.
- **End** corresponds to `z` in the `for` loop shown above.

Vector of Indices

Choose this option to execute a `for` loop of the following form:

```
for varname=[a b c ...]
```

Enter the appropriate values in the **Vector** field in the form `a b c ...`.

Workspace Variable

Show index value in base workspace
Use this option to place the loop index (the value of the `for` loop variable) in the workspace while other components are executing.

Variable name
Enter the variable name in this field. The default is `RPTGEN_LOOP`.

Remove variable from workspace when done
Use this option to remove the loop index from the workspace. This option is only available if you selected the **Show index value in base workspace** option.

Insert Anything into Report?

No.

File Name

`rptgen_lo.clo_for`

Handle Graphics Linking Anchor

Category

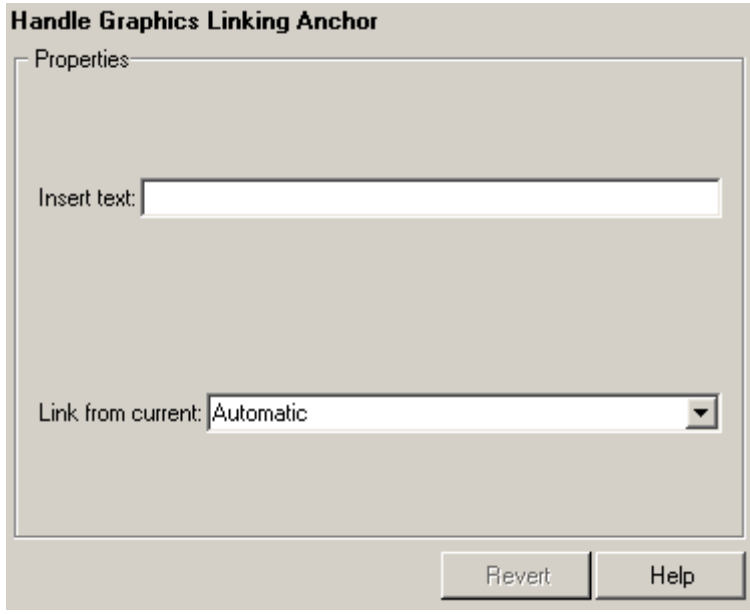
Handle Graphics

Description

The Handle Graphics Linking Anchor component designates a location to which other links point. This component should have a looping component as its parent.

Attributes

The following figure shows the **Handle Graphics Linking Anchor** Properties pane.



Insert text

If you want text to appear after the linking anchor, enter the text in this field.

Link from current

This option sets the current model, system, block, or signal as the linking anchor:

- Select `Automatic` to automatically select the appropriate figure, axes, or object as a linking anchor. If the Figure Loop component is the parent component, the linking anchor is set on the current figure. Similarly, if the Graphics Object Loop is the parent component, the linking anchor is inserted for the current object.
- Select `Figure` to set the linking anchor to the current figure.
- Select `Axes` to set the linking anchor to the current axes.
- Select `Object` to set the linking anchor to the current object.

Insert Anything into Report?

Yes. Anchor.

File Name

`rptgen_hg.chg_obj_anchor`

Handle Graphics Name

Category

Handle Graphics

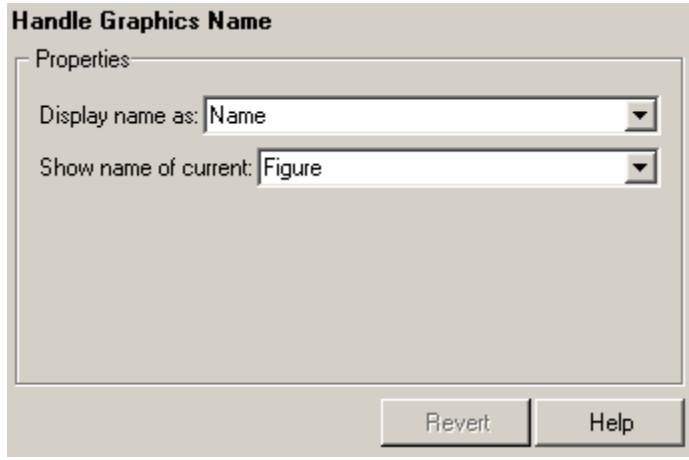
Description

The Handle Graphics Name component inserts the name of a Handle Graphics object into the report as text.

You can use this component to create a section title based on the current figure. First, make sure that this component is the first child component of a `Chapter/Subsection` component, and select the **Get title from first child component** option in the Chapter/Subsection component.

Attributes

The following figure shows the **Handle Graphics Name** Properties pane.



Display name as

Select `Figure` to show the name of the current Handle Graphics figure. The name of the current figure comes from the first nonempty figure parameter in the following order:

- `Name`
- `Tag`
- `file name`

- `Handle`

Show name of current

Select `Axes` to show the name of the current axes. The name of the current axes comes from the first nonempty axes parameter in the following order:

- `Title`
- `Xlabel`
- `Ylabel`
- `Zlabel`
- `Tag`
- `Handle`

Select the `Other Object` option to take the name of the current object from the figure's `CurrentObject` parameter and its first nonempty figure parameter in the following order:

- `uicontrol`
  - `String`
  - `Style`
  - `Handle`
- `uimenu`
  - `Label`
  - `Accelerator`
  - `Tag`
  - `Handle`
- `uicontextmenu`, `image`, `light`, `line`, `patch`, or `surface`
  - `Tag`
  - `Handle`

- `text`
  - `String`
  - `Tag`
  - `Handle`

See the MATLAB graphics documentation for details.

Show name of current
Select one of the three options listed below **Name** (example: `Name`)

- `Figure`
- `Axes`
- `Other Object`

Insert Anything into Report?

Yes. Text.

File Name

`rptgen_hg.chg_obj_name`

Category

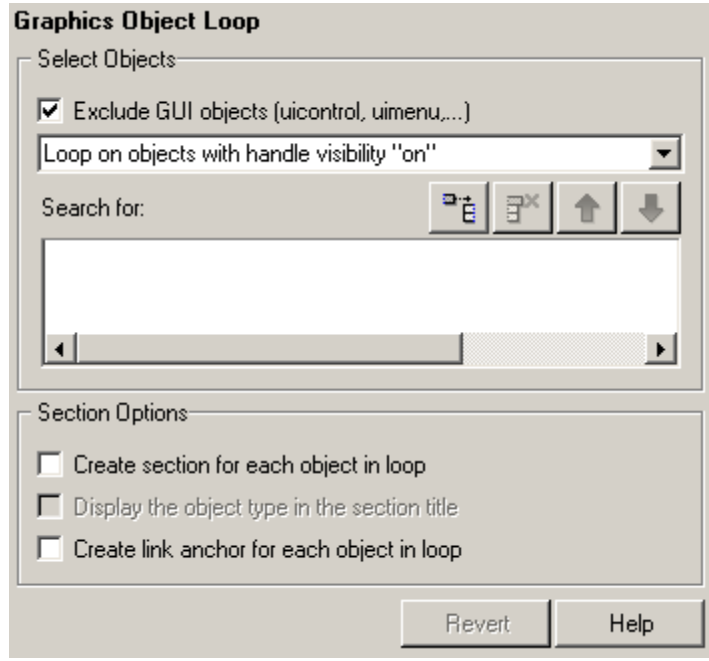
Handle Graphics

Description

The Graphics Object Loop component runs its child components for each Handle Graphics object currently open in the MATLAB workspace. The component inserts a table into the generated report.

Attributes

The following figure shows the **Graphics Object Loop** Properties pane.



Select Objects

Exclude GUI objects (uicontrol, uimenu, ...)

Select this option to exclude GUI objects, such as `uicontrol` and `uimenu`, from the loop.

Loop list

Use this list to select the loop level for Handle Graphics objects:

- `Loop on objects with handle visibility "on"`

- `Loop on all objects`

Search for
Enter search terms in this field. Separate search terms with a space.

Section Options

Create section for each object in loop
Select this option to insert a section in the generated report for each object found in the loop.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the object in the generated report.

Insert Anything into Report?

Yes, inserts a section if the **Create section for each object in loop** option is selected.

File Name

`rptgen_hg.chg_obj_loop`

Category

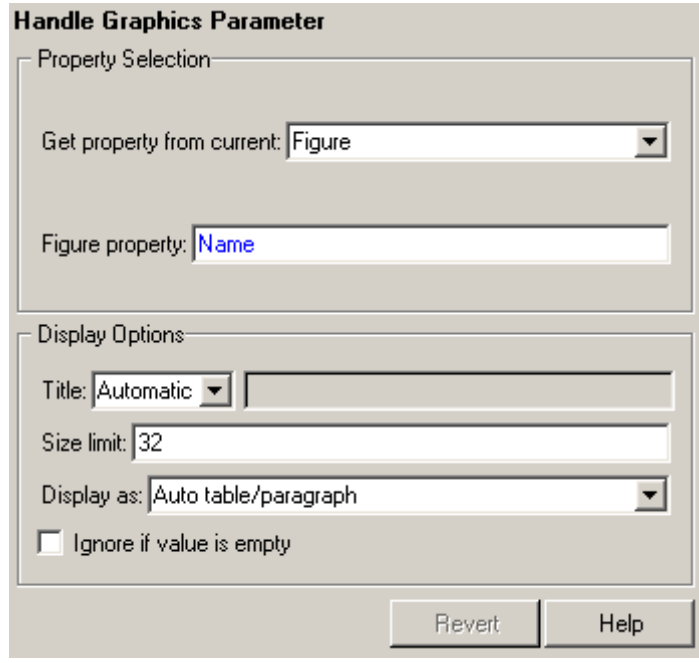
Handle Graphics

Description

The Handle Graphics Parameter component inserts a single property name/property value pair from a Handle Graphics figure, axes, or other object.

Attributes

The following figure shows the **Handle Graphics Parameter** Properties pane.



Property Selection

Get property from current

Select a Handle Graphics object to report on:

- Select `Figure` to insert a figure's property name/property value pair.

- Select `Axes` to insert an axes' property name/property value pair.
- Select `Object` to insert an object's property name/property value pair.

Figure property

Choose the type of property you want to include from the list in this section. `All` shows every parameter for the current object. Select the property you want to include in the report from the list box in this section.

Display Options

Title

Choose a title for the display in the generated report. The default is `None`. To generate a title automatically from the parameter, select `Automatic`. To enter a title, select `Custom`.

Size limit

Use this size to limit the width of the display in the generated report. Units are in pixels. When creating a table, the size limit is the hypotenuse of the table width and height [`sqrt(w^2+h^2)`]. When creating text, the size limit is the number of characters squared. If the size limit is exceeded, the variable is rendered in condensed form, such as `[64x64 double]`. Setting a size limit of zero will ignore the size limit and always display the variable, no matter how large.

Display as

Choose a display style from the menu:

- `Auto table/paragraph` — Displays as a table or paragraph based on the information.
- `Inline text` — Displays as inline, which fits in line with the surrounding text.
- `Paragraph` — Displays as a text paragraph.
- `Table` — Displays as a table.

Ignore if value is empty
Select this option to exclude empty parameters in the generated report.

Insert Anything into Report?

Yes. Text.

File Name

`rptgen_hg.chg_property`

Handle Graphics Property Table

Category

Handle Graphics

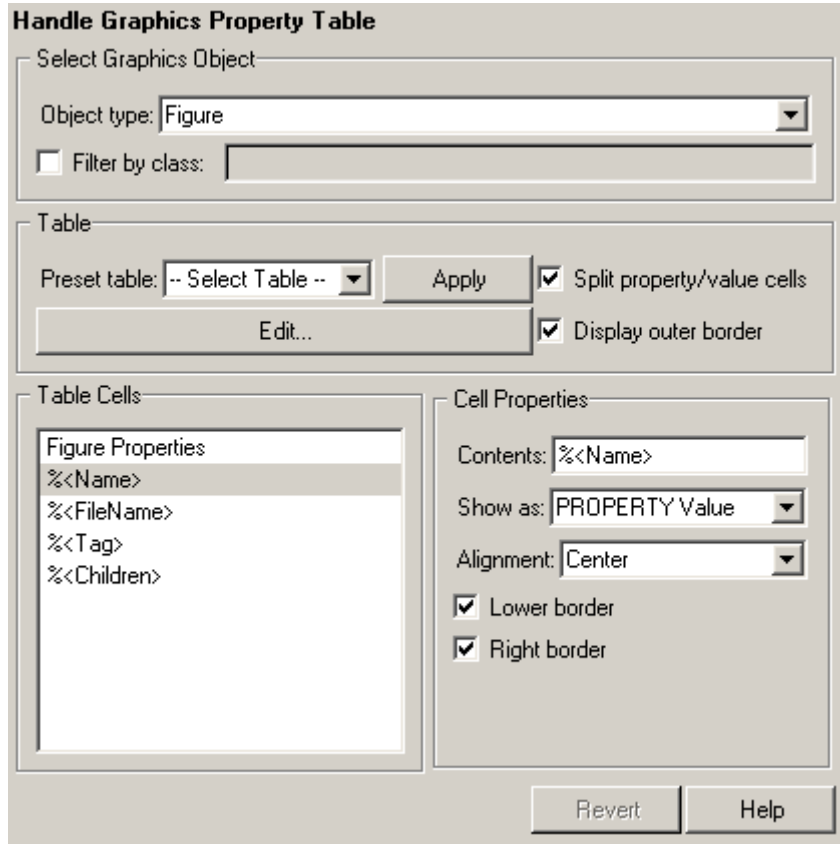
Description

The Handle Graphics Property Table component inserts a table that reports on property name/property value pairs.

See “Working with Property Table Components” on page 7-14 for a tutorial on the Handle Graphics Property Table component.

Attributes

The following figure shows the **Handle Graphics Property Table** Properties pane.



Select Graphics Object

Object type

Choose the object type for the generated report:

- `Figure`
- `Axes`
- `Object`

Filter by class

Select this option to filter the object by the specified class or classes. This option enables you to select a specific class for the table. For example, when selecting type `Object`, you can filter by class `line`. If the object is type `surface`, it does not generate a property table.

Table

A preset table is a table that is already formatted and set up. You can select a preset table in the preset table list in the upper-left corner of the attributes page. To apply a preset table, select the table and click **Apply**.

Preset table

Choose a type of table to display the object property table:

- `Defaults`
- `Callbacks`
- `Graphics`
- `Printing`
- `Blank 4x4`

Split property/value cells

Select this option to split property name/property value pairs into separate cells. To have the property name and property value appear in adjacent horizontal cells in the table, select the **Split property/value cells** check box. In this case, the table is in split mode and there can be only one property name/property value pair in a cell. If you have more than one name/property pair in

a cell, only the first pair appears in the report. All subsequent pairs are ignored.

To have the property name and property value appear together in one cell, clear the **Split property/value cells** check box. This is nonsplit mode. In nonsplit mode, you can have more than one property name/property value pair and you can also have text.

If you want to switch from nonsplit mode to split mode, make sure that you have only one property name/property value pair per table cell before you switch modes. If you have more than one property name/property value pair or any text, only the first property name/property value pair appears in the report; subsequent pairs and text are omitted.

Display outer border

Select this option to display the outer border of the table in the generated report.

Table Cells

Use this pane to select table properties to modify. The selection in this pane affects the available fields in the **Title Properties** pane.

Cell Properties

The visible options in the **Title Properties** pane depend on the object selected in the **Table Cells** pane. If `Figure Properties` is selected, only the **Contents** and **Show** options are displayed. If any other object is selected in the **Table Cells** pane, the **Lower border** and **Right border** options are displayed.

Contents

Use this field to modify the contents of the table cell selected in the **Table Cells** pane.

Choose the of the contents of the selected table cell in the **Table Cells** pane.

- `Left`
- `Center`

- `Right`
- `Double justified`

Show as
Choose the format for the contents of the table cell:

- `Value`
- `Property Value`
- `PROPERTY Value`
- `Property:  Value`
- `PROPERTY: Value`
- `Property - Value`
- `PROPERTY - Value`

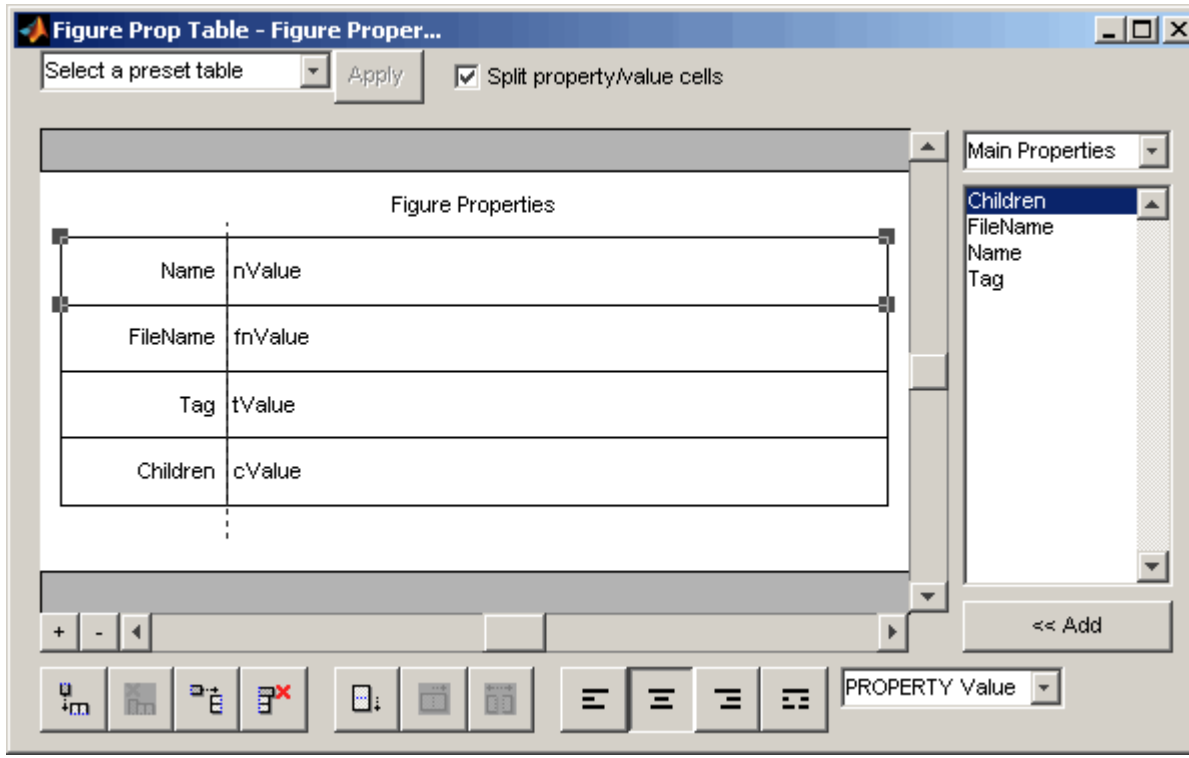
Alignment
Select this option to align text in your cells. Choices are `Left`, `Center`, `Right`, and `Double-justified`.

Lower border
Select this option to display the lower border of the table in the generated report.

Right border
Select this option to display the right border of the table in the generated report.

Creating Custom Tables

To create a custom table, you can first select a preset table and then edit it. You might want to start with the `Blank 4x4` preset table. You can add and delete rows and you can add properties. To open the Edit Table dialog box, click **Edit**. The dialog box appears, as the following figure shows:



For details about using this dialog box to create custom property tables, see “Working with Property Table Components” on page 7-14.

Insert Anything into Report?

Yes. Table.

File Name

`rptgen_hg.chg_prop_table`

Category

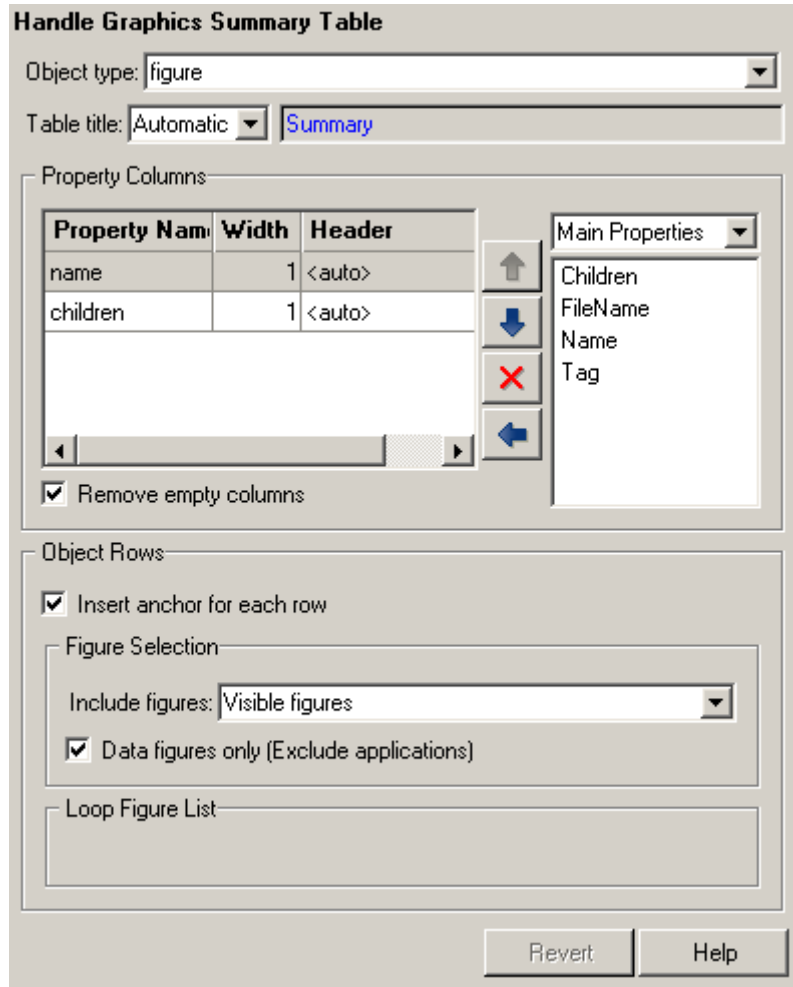
Handle Graphics

Description

The Handle Graphics Summary Table component inserts a table that summarizes Handle Graphics object properties. Each row in the table represents an object, and each column in the table represents a property. You can specify the object properties to be included in the report.

Attributes

The following figure shows the **Handle Graphics Summary Table** Properties pane.



Object type

Choose the object type to display in the generated report:

- `figure`
- `axes`

- `object`

The available options in the **Select Objects** pane depend on your selection in the **Object type** menu.

Table title

Choose a title for the summary table in the generated report:

- `Automatic` — Select this option to generate a title automatically from the parameter.
- `Custom` — Select this option to enter a title manually.

Property Columns

Property columns

This figure displays the object properties to be included in the summary table in the generated report. To add a property, select the appropriate property level in the menu. Next, in the contextual list below the menu, select the property that you want to add and click **Add**. To delete a property, select the property name and click **Delete**.

Note Some entries in the list of available properties (such as `Depth`) are "virtual" properties which cannot be accessed using the `get_param` command. The properties used for property/value filtering in the block and system loop components must be retrievable by the `get_param`. Therefore, you cannot configure your summary table to report on all blocks of `Depth == 2`.

Remove empty columns

Use this option to remove empty columns from the summary table in the generated report.

Object Rows

Insert anchor for each row

Select this option to insert an anchor for each row in the summary table.

Figure Selection

The options displayed in the **Figure Selection** pane depend on the object type selected in the **Object type** list.

- If `figure` is selected in **Object type**, the following options are displayed:

  Include figures

  To include only the current figure in the report, select the `Current figure only` option.

  To loop on all visible figures, select the `Visible figures` option. This option executes child components for all figures that are currently open and visible. The `Data figures only` option is checked by default and excludes figures with `HandleVisibility = 'off'` from the loop.

  To loop on figures with certain tags, select the `All figures with tags` option. When you are selecting a tag, all figures with that tag are included, regardless of whether the figure is visible or whether `HandleVisibility` is `'on'` or `'off'`. The tag list box (located under the `All figures with tags` option) shows the selected tags. To add a tag to the tag list box, type in the tag.

  Data figure only (Exclude applications)

  select this check box to show only data figures.

  Loop Figure List

  The **Loop Figure List** shows you the figures that would be looped on given the current set of figures.

- If `axes` is selected in **Object type**, the following options are displayed:

  Loop type

  To loop on all axes objects, select `All axes`. To loop on the selected axes object, select `Current axes`.

Exclude objects which subclass axes

Select this option to exclude objects such as legends and color bars.

Loop Menu

To loop only on visible axes objects, select `Loop on axes with handle visibility "on"`. To loop on all axes objects, select `Loop on all axes`.

Search terms

Enter search terms for the loop. For example, to search for `Tag` and `My Data`, enter `"Tag", "My Data"`.

- If `object` is selected in **Object type**, the following options are displayed:

Exclude GUI objects (uicontrol, uimenu, ...)

Select this option to exclude GUI objects, such as `uicontrol` and `uimenu`, from the loop.

Loop menu

Use this menu to select the loop level for Handle Graphics objects:

- `Loop on objects with handle visibility "on"`
- `Loop on all objects`

Search for

Enter search terms in this field. Separate search terms with a space.

Insert Anything into Report?

Yes. Table.

Handle Graphics Summary Table

File Name `rptgen_hg.chg_summ_table`

Category

Formatting

Description

The Image component inserts an image from an external file into the report. The Image component can have the Chapter/Subsection or Paragraph component as its parent. If the Paragraph component is its parent, the **Insert as inline image** check box must be selected.

Attributes

The following figure shows the **Image** Properties pane.

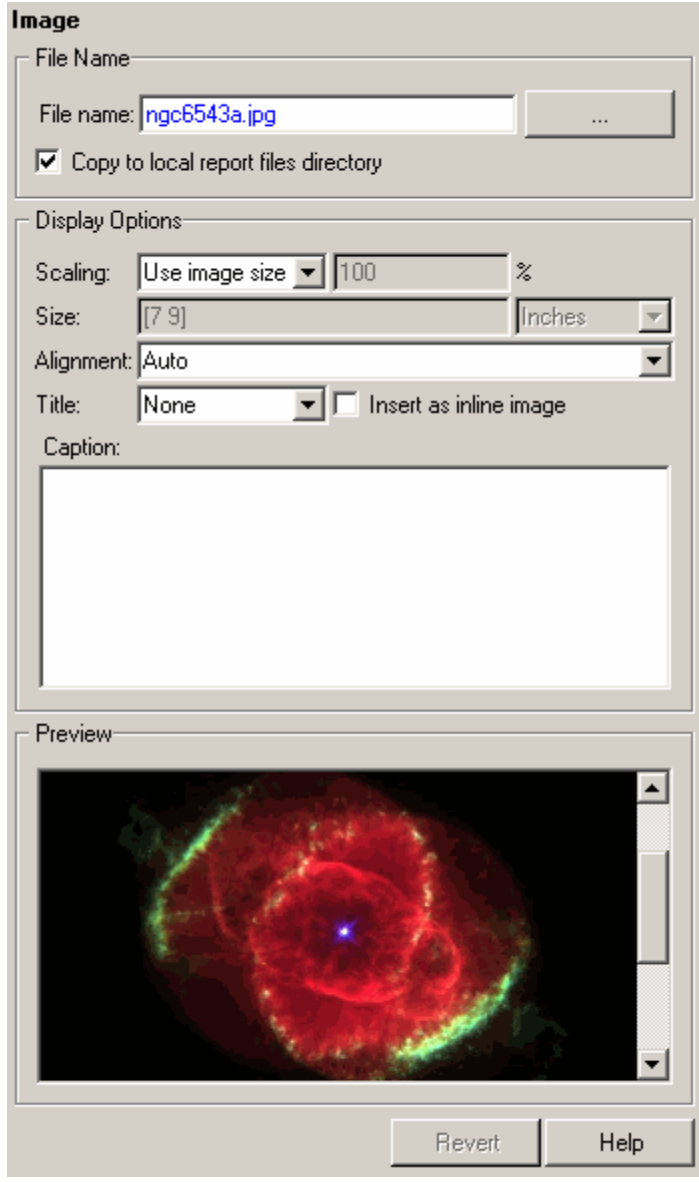


Image
File Name
File name: ngc6543a.jpg
...
Copy to local report files directory
Display Options
Scaling: Use image size 100 %
Size: [7 9] Inches
Alignment: Auto
Title: None
Insert as inline image
Caption:
Preview
Revert
Help

File Name

File name

Enter the image file name in this field or use the **Browse** button (...) to find the file that contains the image.

Make sure that the image is in a format that your viewer can read. Your viewer depends on the report format you selected in the Report Options page in the Report Explorer. For a discussion on report format options, see "Setting Report Output Options" on page 5-2.

You can use the `%<VariableName>` notation. For more details about this notation, see "%<VariableName> Notation" on page 11-269 on the Text component reference page.

Copy to local report files directory

Select this option to save a copy of the image to a local report files directory.

Display Options

Scaling

Select `Fixed size` to specify the number and type of units.

Select `Zoom` to specify the percentage, the maximum size, and the units of measure.

Select `Use image size` if you want the image in your report to be the same size as the image on the screen.

Size

Enter the exact size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Fixed size` from the **Scaling** drop-down list.

Max size

Enter the maximum size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Zoom` from the **Scaling** drop-down list.

Units
Enter the units for the size of the snapshot. This field is active only if you choose `Zoom` or `Fixed size` in the **Image size** list box.

Alignment
Options are `Auto`, `Right`, `Left`, and `Center`.

Note This option is supported only in `PDF` and `RTF` reports.

Title
Type the text you want to appear above the snapshot.

Caption
Type the text you want to appear below the snapshot.

Preview

The **Preview** pane shows the image that you specify in the **Image file name** field. You cannot preview PostScript images or images with formats that are not supported by the `imread` function, such as `.gif`.

If you click the image in the **Preview** pane, it appears in full size on the screen.

Insert Anything into Report?

Yes. Image.

File Name

`rptgen.cfr_image`

Category

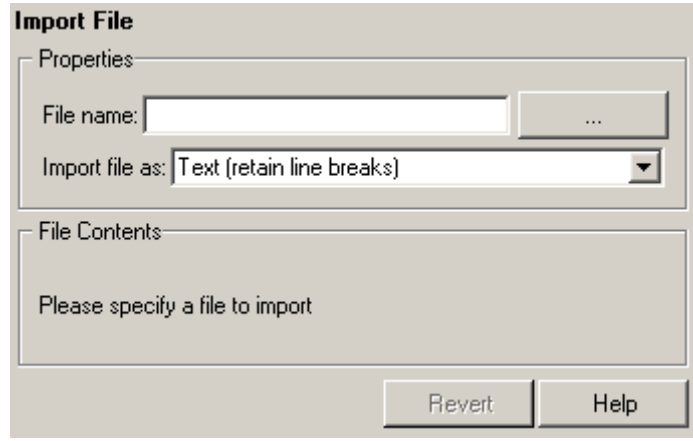
Report Generator

Description

The Import File component imports an ASCII text file.

Attributes

The following figure shows the **Import File** Properties pane.



File name

Enter the name of the file to import in the text field in this section or use the **Browse** button (...) to find the file. This component executes only if you enter a valid file name in this field.

Import file as

Indicates what kind of format to use for the imported file:

- `Plain text (ignore line breaks)`: Select this option to import the file as plain text without any line breaks (no paragraphs). If you select this option, the Import File component acts like the `Text` component and should have the `Paragraph` component as its parent.

  In this example and subsequent examples, the following text is used as the input file.

  ```
  This is the first row of text from the imported file.
  ```

```
  The second row follows a line break in the first row.

There is a blank line above the third row.
```

With the `Plain text (ignore line breaks)` option, the following is produced.

```
This is the first row of text from the imported file.
The second row follows a line break in the first row.

There is a blank line above the third row.
```

- `Paragraphs defined by line breaks`: Select this option to import the file as text contained in paragraphs defined by line breaks (hard returns or carriage returns).

  With the `Paragraphs defined by line breaks` option, the following is produced.

  ```
  This is the first row of text from the imported file.
  The second row follows a line break in the first row.

  There is a blank line above the third row.
  ```

- `Paragraphs defined by empty rows`: Select this option to import the file as text contained in paragraphs defined by empty rows (rows that do not contain text).

  With the `Paragraphs defined by empty rows` option, the following is produced.

  ```
  This is the first row of text from the imported file.
  The second row follows a line break in the first row.

  There is a blank line above the third row.
  ```

- `Text (retain line breaks)`: This is the default. Select this option to import the file as plain text, including line breaks.

With the `Text (retain line breaks)` option, the following is produced.

```
This is the first row of text from the imported file.
      The second row follows a line break in the first row.

There is a blank line above the third row.
```

- `Fixed-width text (retain line breaks)`: Select this option to import the file as fixed-width text (all letters have the same width or size) including line breaks. This option is best suited for importing M-files.

  With the `Fixed-width text (retain line breaks)` option, the following is produced.

  ```
  This is the first row of text from the imported file.
        The second row follows a line break in the first row.

  There is a blank line above the third row.
  ```

- `DocBook XML`: Select this option if the inserted text file is in an XML source file. The format of the inserted XML file is not changed.
- `Formatted Text (RTF/HTML)`: Select this option if the inserted text file is in an RTF or HTML source file. The format of the inserted RTF or HTML source file is not changed.
- `Syntax highlighted M-code`: Select this option if the inserted text file is an M-file containing highlighted M-code.

The **File Contents** field displays the first few lines of the file to be imported.

Insert Anything into Report?

Yes. Inserts text if you select one of the following options:

- `Plain text (ignore line breaks)`
- `Text (retain line breaks)`
- `Fixed-width text (retain line breaks)`

Inserts paragraphs if you select one of the following options:

- `Paragraphs defined by line breaks`
- `Paragraphs defined by empty rows`

Inserts the contents of the XML file if you select the `DocBook XML` option. This can include text, paragraphs, tables, etc.

Inserts the contents of the RTF or HTML file if you select the `Formatted text (RTF/HTML)` option. This can include text, paragraphs, tables, etc.

Inserts a link to a file if you import the file into an HTML report.

File Name

`rptgen.crg_import_file`

Category

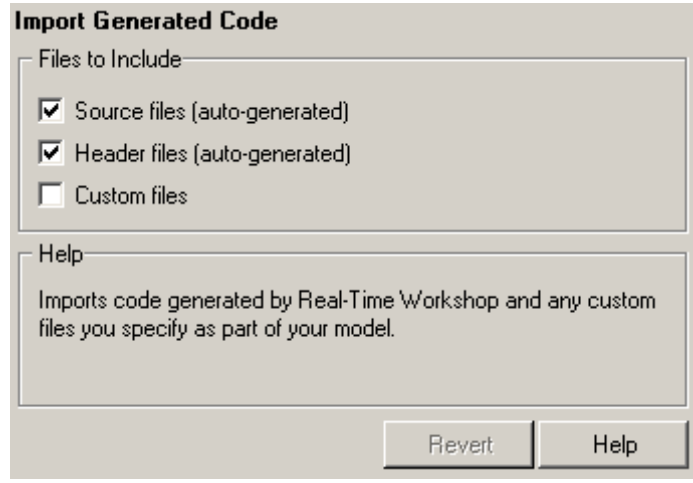
Real-Time Workshop

Description

The Import Generated Code component imports source and header files generated by Real-Time Workshop and any custom files you specify as part of your model.

Attributes

The following figure shows the **Import Generated Code** Properties pane.



Source files (auto-generated)

Select this option to include the following files in the report:

- `.c` and `.cpp` source files that Real-Time Workshop generates based on your model
- Real-Time Workshop source files, such as the template main file and other support files in the build directory

Clear the check box to omit source files. This option is selected by default.

Header files (auto-generated)
Select this option to include the following files in the report:

- `.h` and `.hpp` header files that Real-Time Workshop generates based on your model
- Real-Time Workshop header files in the build directory

Clear the check box to omit header files. This option is selected by default.

Custom files
Select this option to include custom source files that you specify in the **Real-Time Workshop > Custom Code** pane of the Configuration Parameters dialog box. This option is cleared by default.

Insert Anything into Report?

Yes. Generated code listings.

File Name

`RptgenRTW.CImportCode`

Category

MATLAB

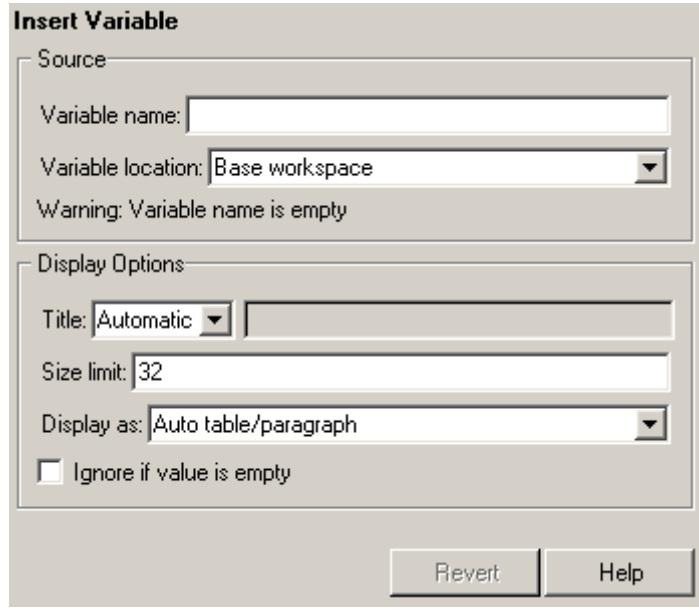
Description

The Insert Variable component takes one of the following variables and inserts its value (and name, if you choose to do so) into the report:

- A variable from the MATLAB workspace
- A variable from a MAT-file
- A global variable

Attributes

The following figure shows the **Insert Variable** Properties pane.



Source

Variable name

Enter the name of the variable in this field.

You can use `%<VariableName>` to insert the value of a variable from the MATLAB workspace. For example, if you enter the following expression in this field:

```
%<my_variable>
```

and you have the following in your MATLAB workspace:

```
my_variable='velocity'
velocity=50
```

then your report will contain the following:

```
50
```

If you enter `%<velocity>` in this field, and your MATLAB workspace contains the same information, your report will also contain `50`.

Variable location

Select one of the following options:

- `Base Workspace` — Gets a variable from the MATLAB workspace.
- `MAT File` — Gets a variable from a binary file with a `.mat` extension.
- `Global variable` — Gets a global variable.

Display Options

Title

Choose a title for the display in the generated report. To generate a title automatically from the parameter, select `Automatic`. To enter a title, select `Custom`. To specify no title, select `None`.

Size limit

Use this size to limit the width of the display in the generated report. Units are in pixels. When creating a table, the size limit is the hypotenuse of the table width and height [`sqrt(w^2+h^2)`]. When creating text, the size limit is the number of characters

squared. If the size limit is exceeded, the variable renders in condensed form, such as `[64x64 double]`. Setting a size limit of zero will ignore the size limit and always display the variable, no matter how large.

Display as

Choose a display style from the menu:

- `Table` — Displays as a table.
- `Paragraph` — Displays as a text paragraph.
- `Inline text` — Displays as inline, which fits in line with the surrounding text.
- `Auto table/paragraph` — Displays as a table or paragraph based on the information.

Ignore if value is empty

Select this option to exclude empty parameters in the generated report.

Insert Anything into Report?

Yes. Text.

File Name

`rptgen.cml_variable`

Link

Category

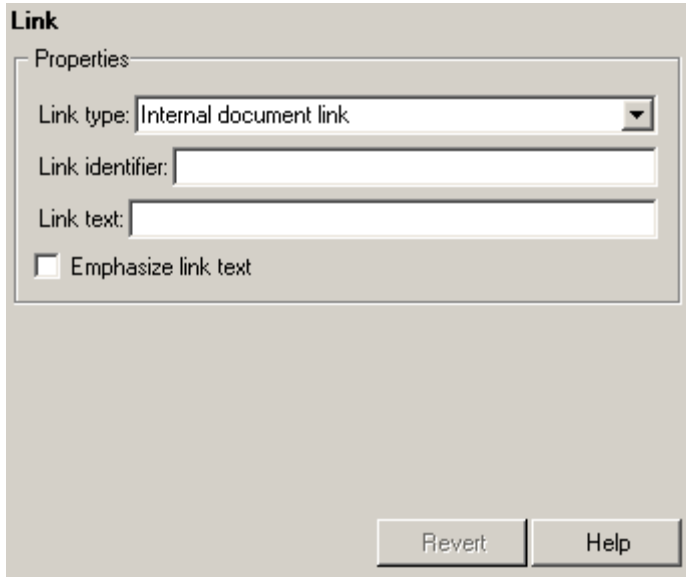
Formatting

Description

The Link component inserts linking anchors or pointers into the report.

Attributes

The following figure shows the **Link** Properties pane.



d

Link type

Select the type of link to insert into the report:

- `Linking anchor`: An anchor is the object to which something links; i.e., links "link" to an anchor.
- `Internal document link`: A link takes you to the location in your report (specified by an anchor) to which it points when you click it.

- `URL (external) link`: A URL link is a link to a Web site. It takes you to the location on the Web to which your link points when you click it.

Link identifier
This field marks the location to which the link points.

The link identifier options are context sensitive and change depending on the link type selected above. For example, say you want to link to an external file `foo.txt`. On UNIX , the link would look something like this:

```
file:///home/janedoe/foo.txt
```

While on Windows, it would look something like this:

```
H:\foo.txt
```

The link is not visible in the report.

Link text
Enter text to use in the link.

Emphasize link text
Select this option to italicize the link text.

Insert Anything into Report?

Yes. Text or anchor.

File Name

`rptgen.cfr_link`

List

Category

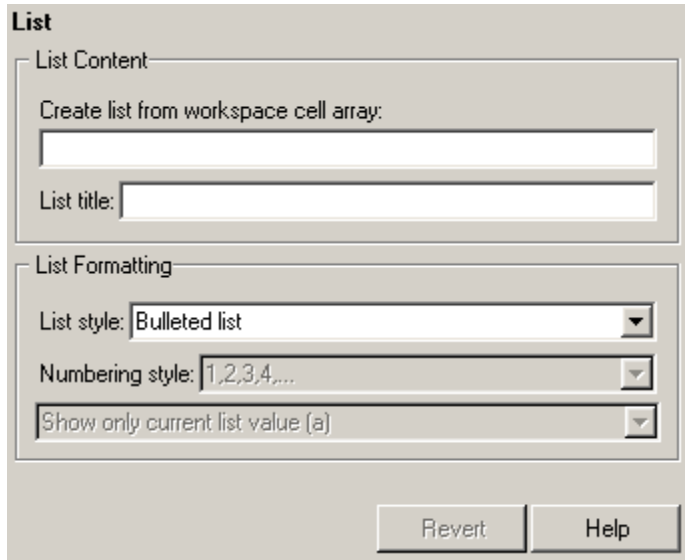
Formatting

Description

The List component creates a bulleted or numbered list from a cell array or from child components.

Attributes

The following figure shows the **List** Properties pane.



List Content

Create list from workspace cell array

Enter the name of the 1-by-n or n-by-1 cell array from which the list will be created in this field. This option is not available when this component has child components, because the list is then automatically generated from the child components.

List title

Enter the title of the list in this field.

List Formatting

List style

Select either `Bulleted list` or `Numbered list`. If you choose `Numbered list`, you can select numbering options in the **Numbering style** field.

Numbering style

Select a numbering style in this pull-down menu if you selected `Numbered list` for **List style**. The options are

- `1,2,3,4,...`
- `a,b,c,d,...`
- `A,B,C,D,...`
- `i,ii,iii,iv,...`
- `I,II,III,IV,...`

Show parent number in nested list (1.1.a)

Select this option to show all level numbers in a nested list. You can create a nested list by putting one cell array inside another or by nesting one List component inside another. If you choose this option, your list will have this type of appearance (the actual appearance depends on the option chosen in the **Numbering style** field).

```
1. Example
2. Example
 2.1. Example
 2.2. Example
  2.2.a. Example
  2.2.b. Example
3. Example
```

Note that this option is not available if you select the `Show only current list value (a)` option.

Show only current list value (a)

Select this option to show only the current list value. If you choose this option, the list shown above will have this type of

appearance (the actual appearance depends on the option chosen in the **Numbering style** field).

```
1. Example
2. Example
 1. Example
 2. Example
  1. Example
  2. Example
3. Example
```

Note that this option is not available if you select the **Show parent number in nested list (1.1.a)** option.

Example 1: Creating a Nested List

You can create a nested list by putting one cell array inside another or by nesting one List component inside another.

For example, if you have the following report template, with one List component nested inside another List component,

```
[-] Report - Unnamed.rpt
   [-] Bulleted list from child components
      [ ] Text - sky
      [ ] Table - varname
      [ ] Image - test.jpg
      [ ] Text - grass
      [-] Bulleted list from child components
         [ ] Text - clouds
         [ ] Text - sun
      [-] Paragraph - information
```

then the report would contain the following bulleted lists:

- `sky`
- Table from variable — `varname`
- Snapshot of image — `test.jpg`

- grass
  - clouds
  - sun
- information

Example 2: Creating the Same List with Child Components and a Cell Array

Set up file with child components:

```
[-] Report - Unnamed.rpt
   [-] Bulleted list from child components
      [ ] Text - red
      [ ] Text - green
      [ ] Text - blue
```

Set up file to call a cell array:

```
[-] Report - Unnamed.rpt
   [-] Bulleted list from cell array called colors
```

Cell array:

```
colors={'red','green','blue'}
```

Using either of these two report templates, the report contains the following bulleted list:

- red
- green
- blue

Insert Anything into Report?	Yes. List.
File Name	`rptgen.cfr_list`

Category

Logical and Flow

Description

The Logical Else component acts as an `else` when it is the child of the `Logical If` component. There are two ways to use this component:

```
if
 then
 else
```

or

```
if
 then
 elseif
 elseif
    .
    .
    .
 else
```

Attributes

The following figure shows the **Logical Else** Properties pane.

Logical Else

If component has no children, insert text

When the Logical Else component has no child components, the string in this field is inserted into your report. In this case, this component acts like the Text component.

Insert Anything into Report?

Depends on whether `if` or `elseif` statement is true.

File Name

`rptgen_lo.clo_else`

Category

Logical and Flow

Description

The Logical Elseif component acts as an `elseif` when it is the child of the `Logical If` component. You must use this component in this format:

```
if
 then
 elseif
 elseif
    .
    .
    .
 else
```

Attributes

The following figure shows the **Logical Elseif** Properties pane.

Test expression
Enter a MATLAB expression to evaluate.

If component has no children, insert text
When the Logical Elseif component has no child components, the string in this field is inserted into the report. In this case, this component acts like the Text component.

Insert Anything into Report?

Yes, only when parent `if` statement is false.

File Name

`rptgen_lo.clo_else_if`

Category

Logical and Flow

Description

The Logical If component acts as a logical `if`; it can have the Logical Then, Logical Elseif, and Logical Else components as its child components. This component executes its child components when the given workspace expression is true. It displays a specified string when it has no child components. There are three ways to use this component:

```
if
 then
```

or

```
if
 then
 else
```

or

```
if
 then
 elseif
 elseif
    .
    .
    .
 else
```

Attributes

The following figure shows the **Logical If** Properties pane.

Logical If

Test expression
Enter a MATLAB expression to evaluate.

If component has no children, insert text
When the Logical If component has no child components, the string in this field is inserted into your report.

Insert Anything into Report?

Depends on attributes.

File Name

`rptgen_lo.clo_if`

Category

Logical and Flow

Description

The Logical Then component acts as a `then` when it is the child of the `Logical If` component. There are three ways to use this component:

```
if
 then
```

or

```
if
 then
 else
```

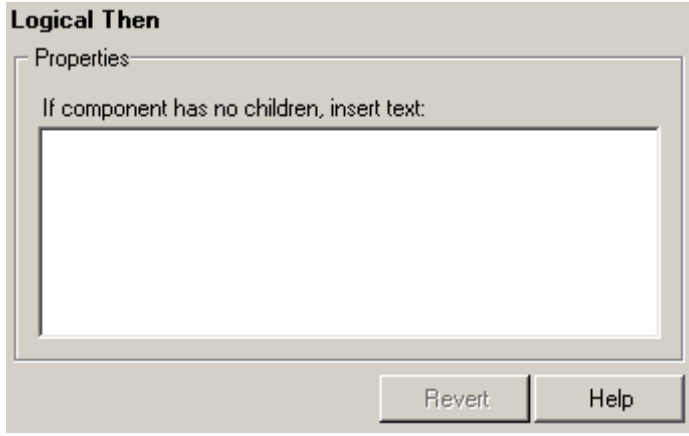
or

```
if
 then
 elseif
 elseif
    .
    .
    .
 else
```

Attributes

The following figure shows the **Logical Then** Properties pane.

Logical Then



If component has no children, insert text

When the Logical Then component has no child components, the string in this field is inserted into your report. In this case, this component acts like the Text component.

Insert Anything into Report?

Yes, only when parent `if` statement is true.

File Name

`rptgen_lo.clo_then`

Category

Simulink Blocks

Description

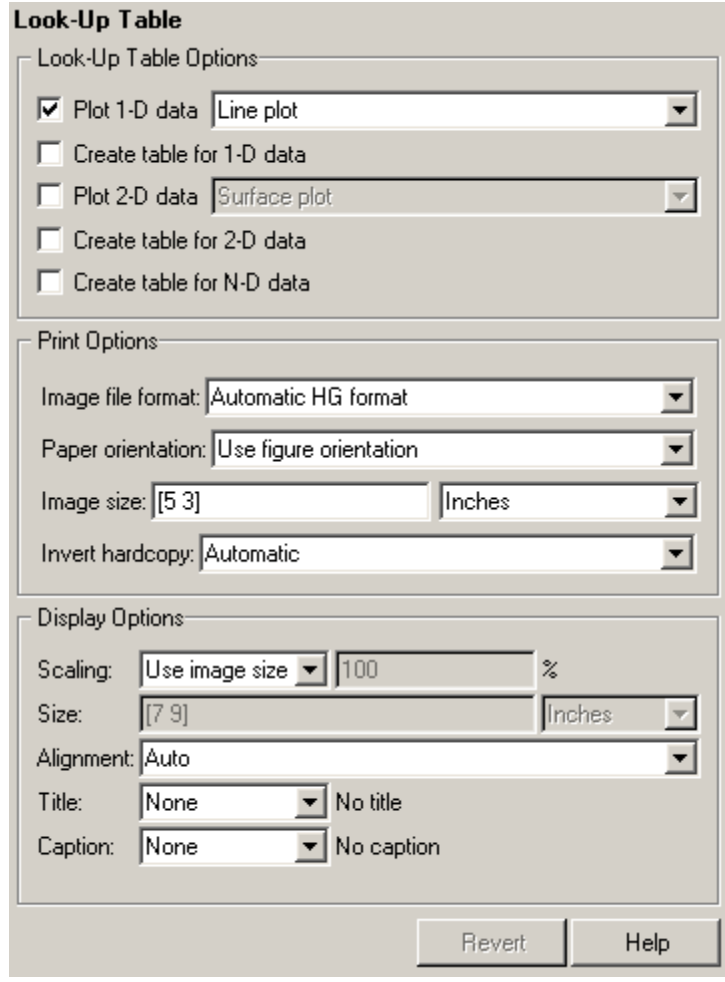
The Look-Up Table component reports on the following lookup table blocks:

- Lookup Table
- Lookup Table (2-D)
- Lookup Table (n-D)

This component inserts a figure and/or table into the report; the table contains input and output numeric values, and the figure is a plot of the values.

Attributes

The following figure shows the **Look-Up Table** Properties pane.


Look-Up Table
Look-Up Table Options
Plot 1-D data
Line plot
Create table for 1-D data
Plot 2-D data
Surface plot
Create table for 2-D data
Create table for N-D data
Print Options
Image file format:
Automatic HG format
Paper orientation:
Use figure orientation
Image size:
[5 3]
Inches
Invert hardcopy:
Automatic
Display Options
Scaling:
Use image size
100
%
Size:
[7 9]
Inches
Alignment:
Auto
Title:
None
No title
Caption:
None
No caption
Revert
Help

Look-Up Table Options

This pane allows you to specify the types of Look-Up Table blocks that are included in the report and how they are displayed. If none of the check boxes in this pane are selected, the component does not insert anything into the report.

These options display the lookup tables, depending on what the component is operating on:

- If the Look-Up Table has the Model Loop as its parent, it includes all lookup tables in the current model.
- If the Look-Up Table has the System Loop as its parent, it includes all lookup tables in the current system.
- If the Look-Up Table has the Block Loop as its parent, it includes the current block if it is a lookup table.
- If the Look-Up Table has the Signal Loop as its parent, it includes all lookup tables connected to the current signal.
- If the Look-Up Table does not have any of the Simulink looping components as its parent, it includes all lookup tables in all open models.

Plot 1-D data

Select this option to plot data from a Look-Up Table block and choose the plot type (`Line plot` or `Bar plot`) from the corresponding list. See the MATLAB Graphics documentation for details on line plots and bar plots. The input data is shown on the horizontal or *x*-axis and the output data is shown on the vertical or *y*-axis.

Create table for 1-D data

Select this option to create a table that contains numeric data values from the Look-Up Table block.

Plot 2-D data

Select this check box to include a plot of 2-D Look-Up Table blocks. You can specify whether the data appears as a surface plot or a line plot. The line plot is best for small data sets and the surface plot is most appropriate for larger tables. See the MATLAB Graphics documentation for details on surface plots and line plots.

Create table for 2-D data
Select this option to create a table that contains numeric data values from the Look-Up Table (2-D) block.

Create table for N-D data
Select this option to create a table that contains numeric data values from the Look-Up Table (n-D) block. Note that this is a 2-D slice through the n-D data.

Print Options

Image file format
Choose the image file format (e.g., JPEG, TIFF, etc.) from this list. Select `Automatic HG Format` to automatically choose the format best suited for the output format that you chose in the Report component. Otherwise, choose an image format that your output viewer can read. `Automatic HG Format` is the default option. The options include:

- `Automatic SL Format` (Uses the Simulink file format selected in the Preferences dialog box)
- `Adobe Illustrator`
- `Bitmap (16m-color)`
- `Bitmap (256-color)`
- `Black and white encapsulated PostScript`
- `Black and white encapsulated PostScript (TIFF)`
- `Black and white encapsulated PostScript2`
- `Black and white encapsulated PostScript2 (TIFF)`
- `Black and white PostScript`
- `Black and white PostScript2`
- `Color encapsulated PostScript`
- `Color encapsulated PostScript (TIFF)`
- `Color encapsulated PostScript2`

- Color encapsulated PostScript2 (TIFF)
- Color PostScript
- Color PostScript2
- JPEG high quality image
- JPEG medium quality image
- JPEG low quality image
- PNG 24-bit image
- TIFF - compressed
- TIFF - uncompressed
- Windows metafile

Paper orientation

You can choose Landscape, Portrait, or Rotated as your paper orientation, or you can choose the Use figure orientation option. See the orient command in the MATLAB Reference Manual for information about paper orientation.

Image size

Select Use figure PaperPositionMode setting to use the PaperPositionMode property of the Handle Graphics figure to set the image size in the report. See the orient command for information about paper position mode.

Select the Automatic (same size as on screen) option if you want the image in your report to be the same size as the image on the screen.

Select the Custom option to choose a custom image size. If you choose this option, specify the image size in the **Size** field and **Units** list.

- **Size**: Enter the size of the Handle Graphics figure snapshot in the form wxh (width times height). This field is active only if you choose Custom in the **Image size** list box.

- **Units**: Enter the units for the size of the Handle Graphics figure snapshot. This field is active only if you choose `Custom` in the **Image size** list box.

Invert hardcopy

Handle Graphics figures have an `InvertHardcopy` property. This parameter inverts colors for printing; i.e., it changes dark colors to light colors and vice versa.

Select one of the following options:

- `Automatic` — Select this option to automatically change dark axes colors to a light axes color. If the axes color is a light color, it is not inverted.
- `Invert` – Select this option to change dark axes colors to light axes colors and vice versa.
- `Don't invert`— Select this option if you do not want to change the colors in the image displayed on the screen for printing.
- `Use figure's InvertHardcopy setting` — Select this option to use the `InvertHardcopy` property set in the Handle Graphics image.
- `Make figure background transparent` — Select this option to make the image background transparent.

Display Options

Scaling

Select `Fixed size` to specify the number and type of units.

Select `Zoom` to specify the percentage, the maximum size, and the units of measure.

Select `Use image size` if you want the image in your report to be the same size as the image on the screen.

Size
Enter the exact size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Fixed size` in the **Scaling** drop-down list.

Max size
Enter the maximum size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Zoom` from the **Scaling** drop-down list.

Units
Enter the units for the size of the snapshot. This field is active only if you choose `Zoom` or `Fixed size` in the **Image size** list box.

Alignment
Options are `Auto`, `Right`, `Left`, and `Center`.

Note This option is supported only in `PDF` and `RTF` reports.

Title
Type the text you want to appear above the snapshot.

Caption
Type the text you want to appear below the snapshot.

Insert Anything into Report?

Yes. Figure and/or table.

File Name

`rptgen_sl.csl_blk_lookup`

Machine Loop

Category

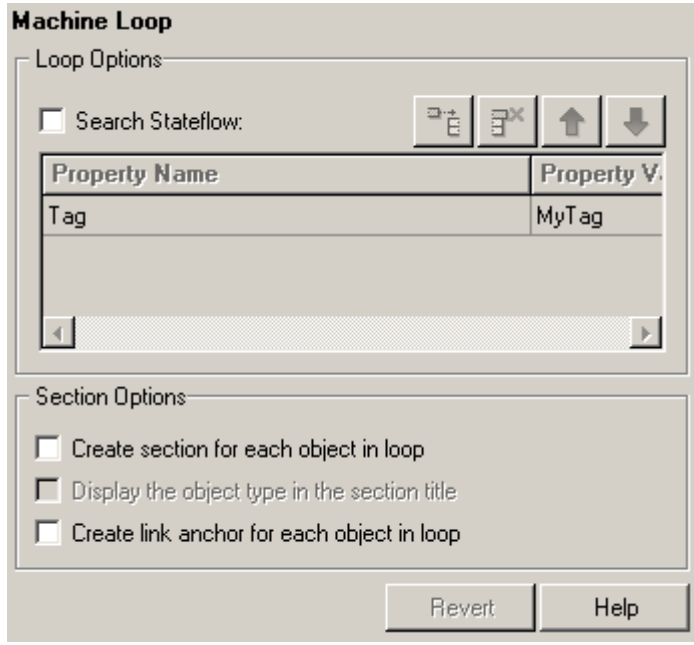
Stateflow

Description

The Machine Loop component runs its child components for all selected Stateflow machines. The Machine Loop component is context sensitive of its parent component. If no parent, the Machine Loop runs its child components for all machines. If a model is its parent, it runs its child components for all machines in the model.

Attributes

The following figure shows the **Machine Loop** Properties pane.



Loop Options

Search Stateflow

Select this option to search states that you specify in the field that appears below the check box.

Section Options

Create section for each object in loop
Select this option to insert a section in the generated report for each object found in the loop.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the object in the generated report.

Insert Anything into Report?

Yes, inserts a section if the **Create section for each object in loop** option is selected.

File Name

`rptgen_sf.csf_machine_loop`

MATLAB Property Table

Category

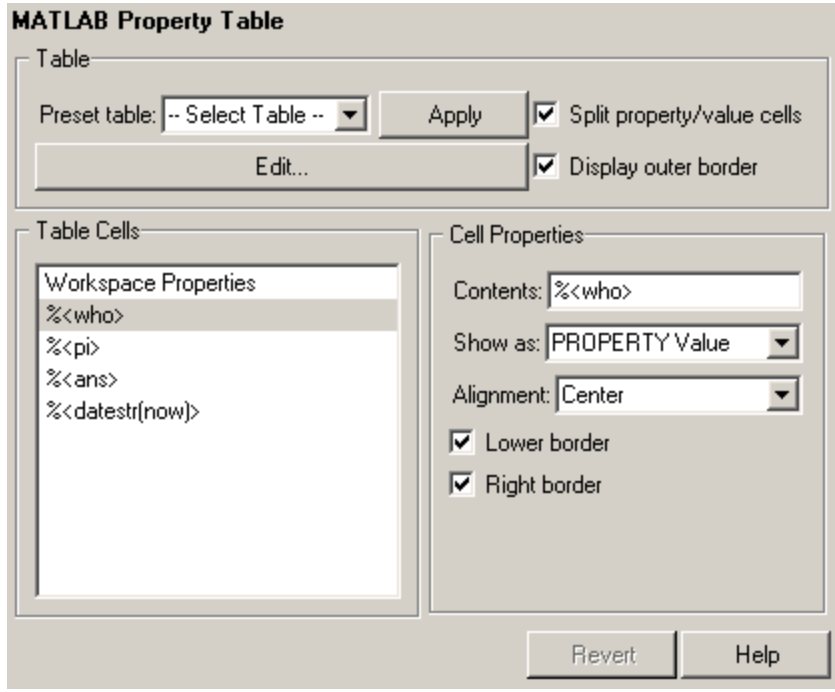
MATLAB

Description

The MATLAB Property Table inserts a table that reports on MATLAB object property name/property value pairs.

Attributes

The following figure shows the **MATLAB Property Table** Properties pane.



Table

A preset table is a table that is already formatted and set up. You can select a preset table in the preset table list in the upper-left corner of the attributes page. To apply a preset table, select the table and click **Apply**.

Preset table

Choose a type of table to display the object property table:

- `Default`
- `Blank 4x4`

Split property/value cells

Select this option to split property name/property value pairs into separate cells. To have the property name and property value appear in adjacent cells, select the **Split property/value cells** check box. In this case, the table is in split mode and there can be only one property name/property value pair in a cell. If you have more than one name/property pair in a cell, only the first pair appears in the report. All subsequent pairs are ignored.

To have the property name and property value appear together in one cell, clear the **Split property/value cells** check box. This is nonsplit mode. In nonsplit mode, you can have more than one property name/property value pair and you can also have text.

If you want to switch from nonsplit mode to split mode, make sure that you have only one property name/property value pair per table cell before you switch modes.

Display outer border

Select this option to display the outer border of the table in the generated report.

Table Cells

Use this field to select table properties to modify. The selection in this pane affects the available fields in the **Cell Properties** pane.

Cell Properties

The visible options in the **Cell Properties** pane depend on the object selected in the **Table Cells** pane. If `Workspace Properties` is selected, only the **Contents** and **Show** options are displayed. If any other object is selected in the **Table Cells** pane, the **Lower border** and **Right border** options are displayed.

Contents
Use this field to modify the contents of the table cell selected in the **Table Cells** pane.

Show as
Choose the format for the contents of the table cell:

- `Value`
- `Property Value`
- `PROPERTY Value`
- `Property:  Value`
- `PROPERTY: Value`
- `Property - Value`
- `PROPERTY - Value`

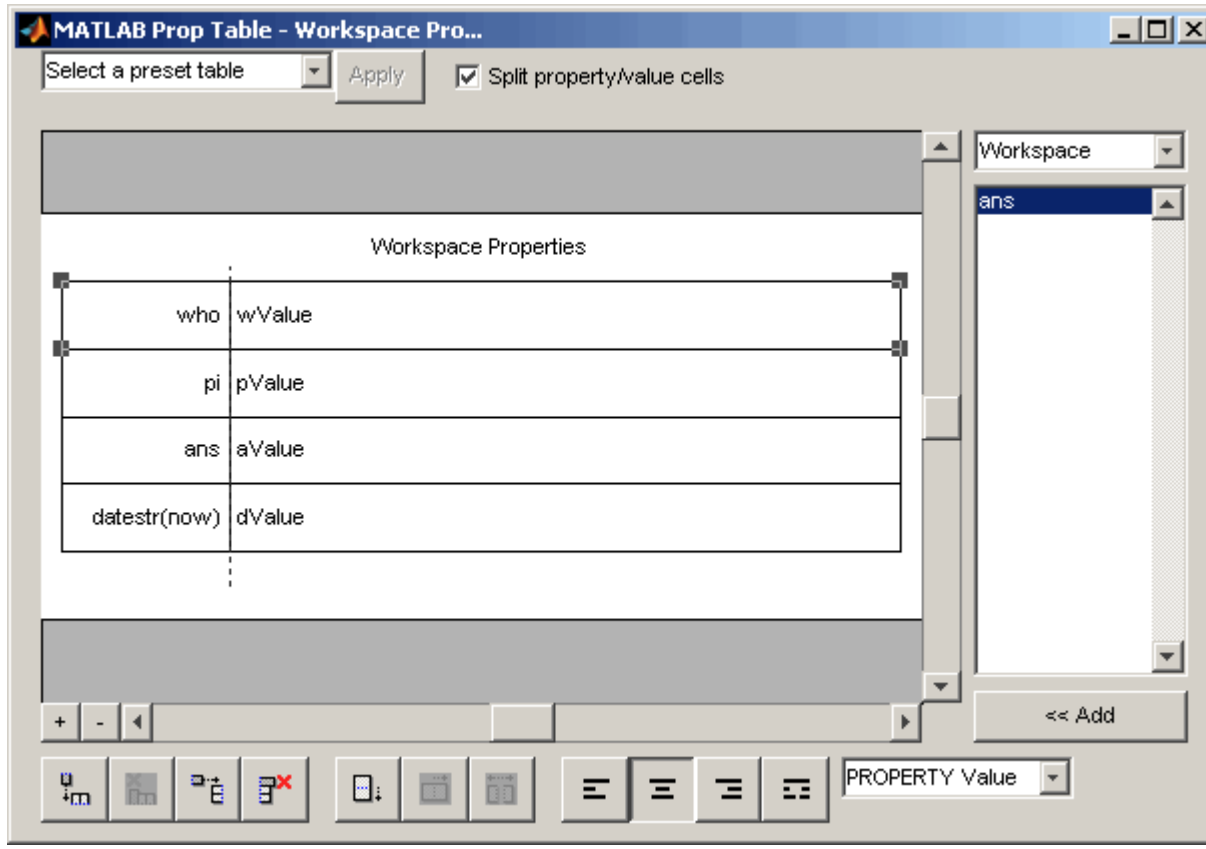
Alignment
Choose the alignment of the contents of the selected table cell in the **Table Cells** field. Options are `Left`, `Center`, `Right`, `Double justified`.

Lower border
Select this option to display the lower border of the table in the generated report.

Right border
Select this option to display the right border of the table in the generated report.

Creating Custom Tables

To create a custom table, you can first select a preset table and then edit it. You might want to start with the `Blank 4x4` preset table. You can add and delete rows and you can add properties. To open the Edit Table dialog box, click **Edit**. The dialog box appears.



For details about using this dialog box to create custom property tables, see “Working with Property Table Components” on page 7-14.

Insert Anything into Report?

Yes. Table.

File Name

`rptgen.cml_prop_table`

MATLAB/Toolbox Version Number

Category

MATLAB

Description

The MATLAB/Toolbox Version Number component creates and inserts a table that shows the version number, release number, and/or release date of any MathWorks software you are currently using. You can get a list of all your toolboxes by typing `ver` at the command line.

Attributes

The following figure shows the **MATLAB/Toolbox Version Number** Properties pane.

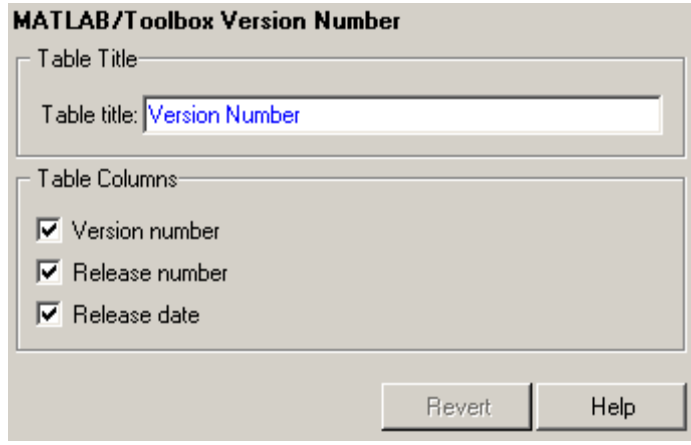


Table Title

Table title

Enter the title of the table in this field. The default is `version number`.

Table Columns

Version number

Select this option to include the version number of any MathWorks software you are currently using.

Release number

Select this option to include the release number of any MathWorks software you are currently using.

Release date
Select this option to include the release date of any MathWorks software you are currently using.

Insert Anything into Report?

Yes. Table.

File Name

`rptgen.cml_ver`

Model Advisor Report

Category

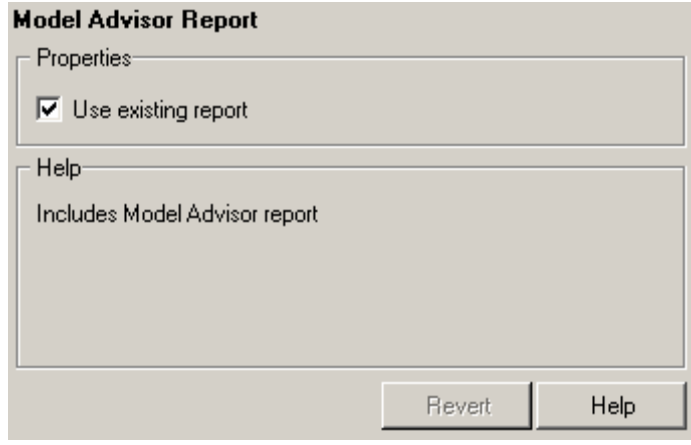
Simulink

Description

The Model Advisor Report component inserts a Model Advisor report for the current model into the report if the report is in HTML format. For other report formats, it inserts a link to a Model Advisor report for the current model into the report.

Attributes

The following figure shows the **Model Advisor Report** Attribute pane.



Use existing report

Select the **Use existing report** check box to include an existing Model Advisor report in the report. If this check box is cleared, a new Model Advisor report is generated. **Use existing report** is selected by default.

Insert Anything into Report?

Yes, a Model Advisor report.

File Name

`rptgen_sl.CModelAdvisor`

Category

Simulink

Description

The Model Change Log component uses a reported model's `ModifiedHistory` parameter to construct a model history table that displays information about each logged revision to the model. This model history table can contain information about any of the following:

- Author of each change
- Model version of the change
- Time and date of the change
- Description of the change

To see the model revision history parameters, select **Model Properties** from the model's **File** menu, which displays the Block Diagram Properties dialog box. The **Model Properties** tab in this dialog box contains the model revision history parameters.

The **Options** tab in the Block Diagram Properties dialog box controls how the version number is incremented. The **History** tab controls history updates and lets you edit the `ModifiedHistory` string. You must select the `Prompt For Comments When Save` option from the **Modified History Update** list in order to save your revisions. This component does not run if the model's `ModifiedHistory` parameter is empty or if you edit the parameter in such a way that Report Generator cannot parse it; in these cases, the Comment component cannot produce the table.

For details on the Block Diagram Properties dialog box and model history, see the Simulink documentation.

Note that if your model has a long revision history, you might want to limit the number of revisions reported.

The Model Change Log component should run before the Model Simulation component.

Attributes

The following figure shows the **Model Change Log** Properties pane.

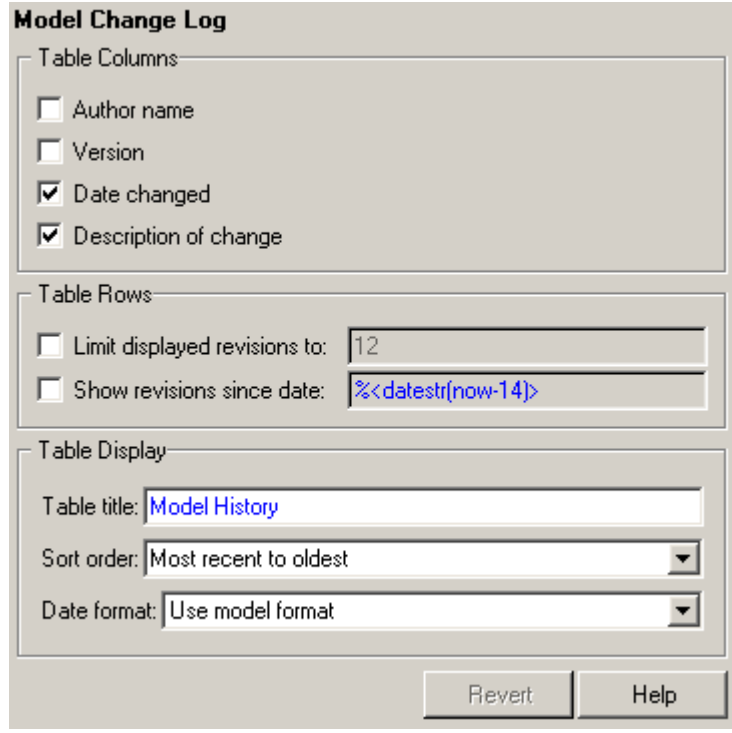


Table Columns

You can choose the information displayed in the model revision table in this section.

Author name
Select this option if you want to include the name of the person responsible for the revision of the model.

Version
Select this option if you want to include the version number of the model.

Date changed
Select this option if you want to include the date on which the model was revised.

Description of change
Select this option if you want to include a description of the revision to the model.

Table Rows

Limit displayed revisions to
Select this option to limit the number of revisions that appears in the report. Enter the number in the corresponding text field.

Show revisions since date
Select this option to limit the number of revisions that appears in the report by date. Enter the date in the corresponding text field. The `%<varname>` notation is legal. For example, `%<datestr(now-14)>` returns the revisions for the last two weeks.

Table Display

You can choose how the model revision history table appears in this section.

Table title
Enter the title of the table in this field.

Sort order
Enables you to sort your table entries from most recent to oldest or vice-versa.

Date format
Enables you to choose a preferred date format for the date/time stamps in your table.

Example

This is an example of a report generated from a report template with the Model Change Log component.

Model Change Log

Table 0. Model History

Ver	Name	Date	Description
1.8	Engineer7	Fri May 07 09:13:50 1999	Changed Sample Time to 8.
1.7	Engineer3	Thu May 06 16:35:31 1999	Changed the sine wave amplitude to 5.
1.6	Engineer6	Thu May 06 16:34:31 1999	Changed the Gain to 10.
1.5	Engineer2	Thu May 06 16:33:08 1999	Changed Sample Time to 5.
1.4	Engineer8	Thu May 06 16:31:59 1999	Changed the Gain to 3.

Insert Anything into Report? Yes. Table.

File Name `rptgen_sl.csl_mdl_changelog`

Category

Simulink

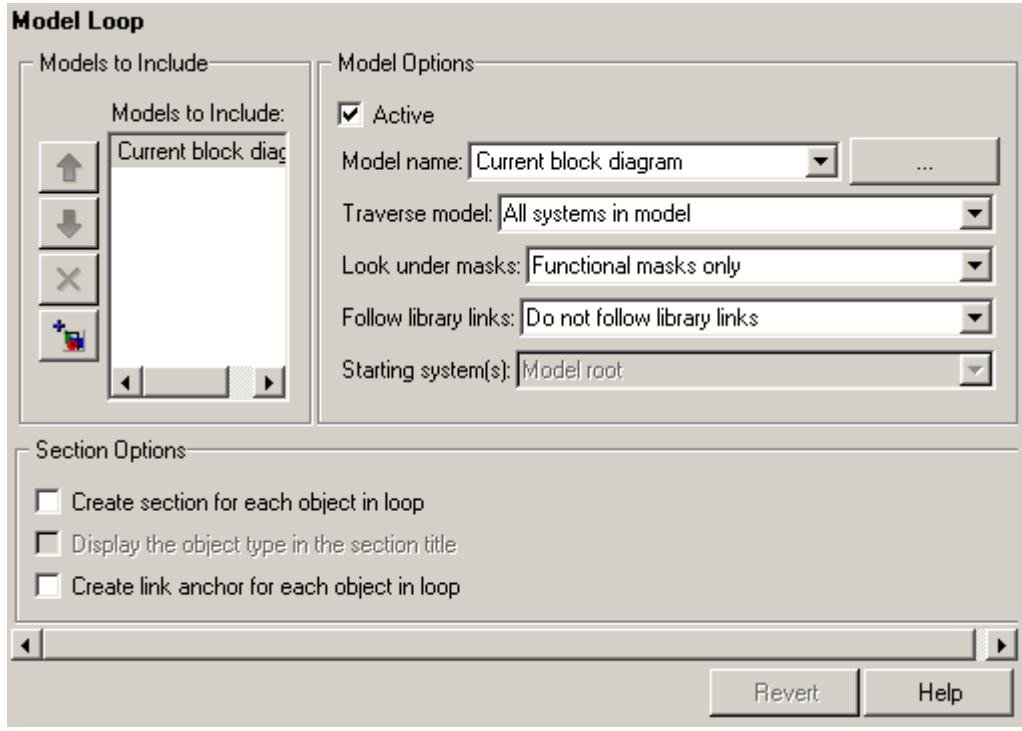
Description

The Model Loop component runs its child components for all selected Simulink models and systems. The Model Loop component can run its child components for a specific model, such as the demo model `F14`, or it can run its child components for all models or the current model.

Because Simulink components are context sensitive, they should be children of the Model Loop (although the Model Loop does not have to be the immediate parent of some Simulink components).

Attributes

The following figure shows the **Model Loop** Attribute pane.



Models to Include

The **Models to Include** field contains a list of reported models. To add another model to the list, click **Add New Model to List**. The following table shows the button you should use to move a model up or down in the list, or to add or delete a model.

Button	Action
	Move a model up in the list.
	Move a model down in the list.
	Remove a model from the list.
	Add a model to the list.

Model Options

Active

Select this option to include the active model. If not selected, the model will be omitted. This option provides a way of temporarily deactivating a model.

Model name

Choose a model name:

- Enter the model name. The `%<varname>` notation is allowed.
- `Current block diagram`
- `All open models`
- `All open libraries`
- `Block diagrams in current directory`
- `Custom block diagram`

This selection affects the available options for the **Traverse model** option.

Traverse model

Choose the system(s) to traverse:

- `All systems in model`
- `Selected system(s) only`
- `Selected system(s) and ancestors`
- `Selected system(s) and children`

The options available in this list are affected by the selection in the **Model name** list.

Look under masks

Simulink systems can have masks. The masks can be functional or graphical. Choose how to handle masks:

- `Functional masks only`
- `No masks`
- `All masks`
- `Graphical masks only`

See the Simulink documentation for details about masks.

Follow library links

Simulink blocks and systems can be linked to a library. Choose the library links to include:

- `Do not follow library links`
- `Include library links`
- `Include unique library links`

See the Simulink documentation for details about library links.

Starting system(s)
Select the system in which to start the loop. The available options in the **Starting system(s)** list are affected by the selection in the **Traverse model** list. For example, if `All systems in model` is selected in the **Traverse model** list, the **Starting system(s)** list is not relevant.

Section Options

Create section for each object in loop
Select this option to insert a section in the generated report for each object found in the loop.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the object in the generated report.

Insert Anything into Report?

Yes, inserts a section if the **Create section for each object in loop** option is selected.

File Name

`rptgen_sl.csl_mdl_loop`

Category

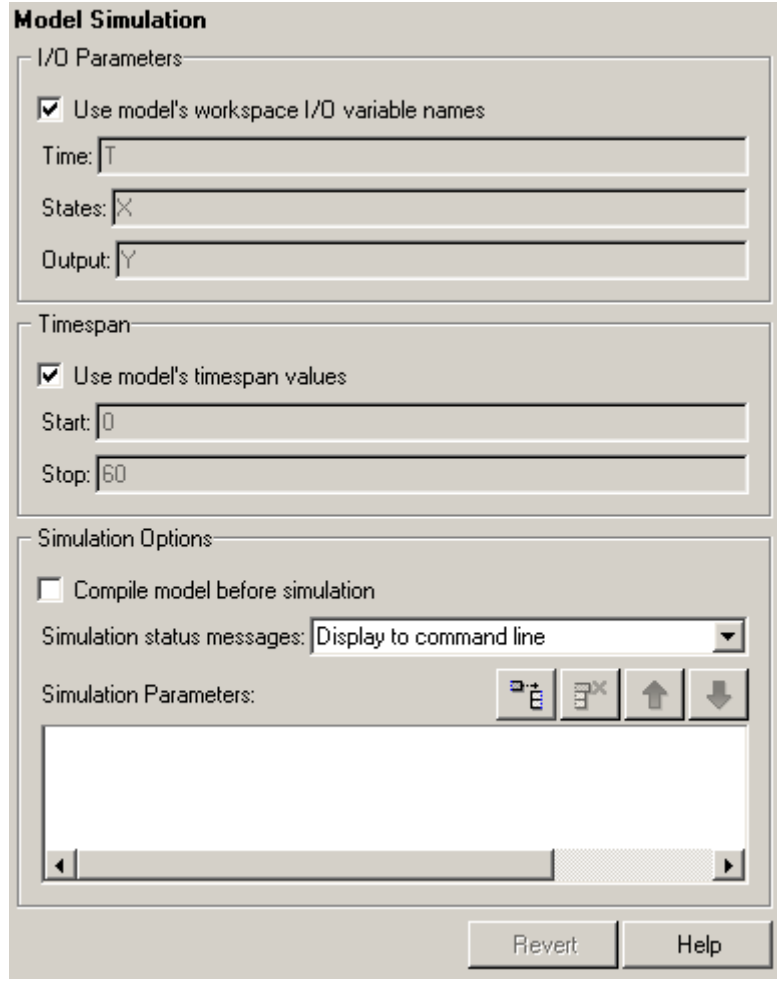
Simulink

Description

The Model Simulation component runs the current model using the simulation parameters specified in the attributes page. This component should have the Model Loop component as its parent.

Attributes

The following figure shows the **Model Simulation** Properties pane.



I/O Parameters

Use model's workspace I/O variable names

Select this option to use the names of the variables specified in the Simulation Parameters dialog box. For a description of these variables, see the Simulink documentation.

The following options are available if you do not select the `Use model's workspace I/O variable names` option:

- **Time** — Enter a new variable name for the `Time` parameter in this field.
- **States** — Enter a new variable name for the `States` parameter in this field.
- **Output** — Enter a new variable name for the `Output` parameter in this field.

Timespan

Use model's timespan values

Select this option to use the model's `Start time` and `Stop time` values specified in the **Solver** tab in the Simulation Parameters dialog box. For a description of these variables, see the Simulink documentation.

The following options are available if you do not select the **Use model's timespan values** option:

- **Start** — Enter a new simulation starting time in this field.
- **Stop** — Enter a new simulation ending time in this field.

Note If your model's stop time is set to `inf` (infinity) in Simulink or on this component's attribute page, Report Generator terminates the model simulation after 60 seconds to prevent the report from generating for an infinite amount of time.

Simulation Options

Compile model before simulation

Select this option if you are using Real-Time Workshop Summary properties, if you sort systems or blocks by simulation order, or if you use scope snapshots. Compiling the model before simulating preserves scope content.

Simulation status messages

Use this option to display your simulation status messages or to insert them into your report. Choose `Display to command line` to send messages to a command-line window, `Display to Report Generator Message List` to send messages to the Report Generator message window, or `Insert into report` to include them in your report.

Simulation parameters

Use this options to specify simulation parameters. For details on simulation parameters, see the `simset` command in the Simulink Reference Manual.

Insert Anything into Report?

No.

File Name

`rptgen_sl.csl_mdl_sim`

Category

Report Generator

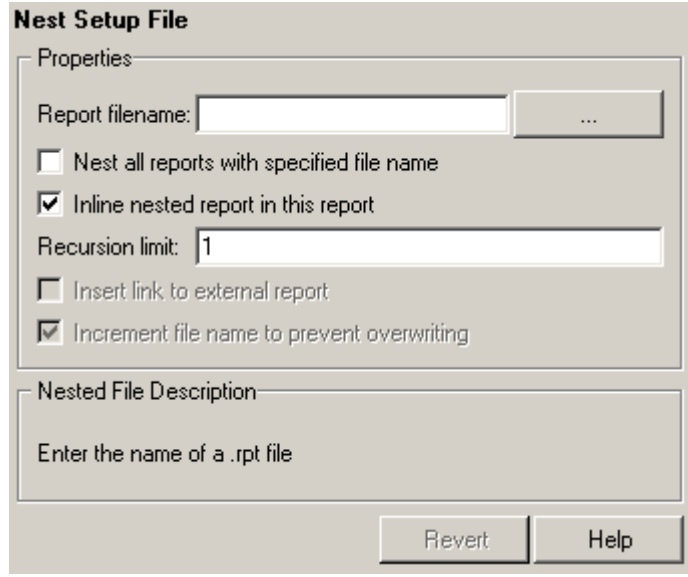
Description

The Nest Setup File component runs another report template at the point where the Nest Setup File component is listed in the current report template. The Nest Setup File component allows one report template (`.rpt`) to run inside another.

The components of the inserted report template enter the current report template at the level of the Nest Setup File component. Thus, the inserted components have the same parent component as the Nest Setup File component. Make sure that the inserted components can have this parent component.

Attributes

The following figure shows the **Nest Setup File** Properties pane.



Setup file to run
Enter the name of the report template to import and run in this field or use the **browse** button (...) to find the file. This component executes only if you enter a valid file name in this field.

Inline nested report in this report
Select this option to insert the nested report at the place in the original report where this component is located.

Insert link to external report
Select this option to create two separate reports, one with the original report template and one with the nested report template.

Recursion limit
If you want to nest a report template inside itself, you can set a recursion limit in this field. The recursion limit sets a limit on the number of times the report template can run itself.

Nest all reports with specified file name
Select this option to nest all reports with the same name as specified in the **Setup file to run** option.

Example

In the example shown below, the report template `R2.rpt` is nested in `R1.rpt`:

```
[-] Report - R1.rpt                [-] Report - R2.rpt
 [ ] Chapter                              [ ] 1
      [-] B                               [ ] 2
         [ ] Nest Setfile - R2.rpt       [-] Chapter
         [ ] C                              [ ] 4
      [ ] D                                 [ ] 5
```

The report executes as if the original report template file looked as follows:

```
[-] Report - R1.rpt
  [ ] Chapter
  [-] B
```

```
    [ ] 1
    [ ] 2
    [-] Section 1
      [ ] 4
      [ ] 5
    [ ] C
[ ] D
```

Components that determine their behavior from their parents, such as Chapter/Subsection, are affected by components in the parent report template (as shown above). See the Chapter/Subsection reference page for more details on the Chapter/Subsection component.

Insert Anything into Report?

Yes, if the nested report template produces visible output.

File Name

`rptgen.crg_nest_set`

Object Loop

Category

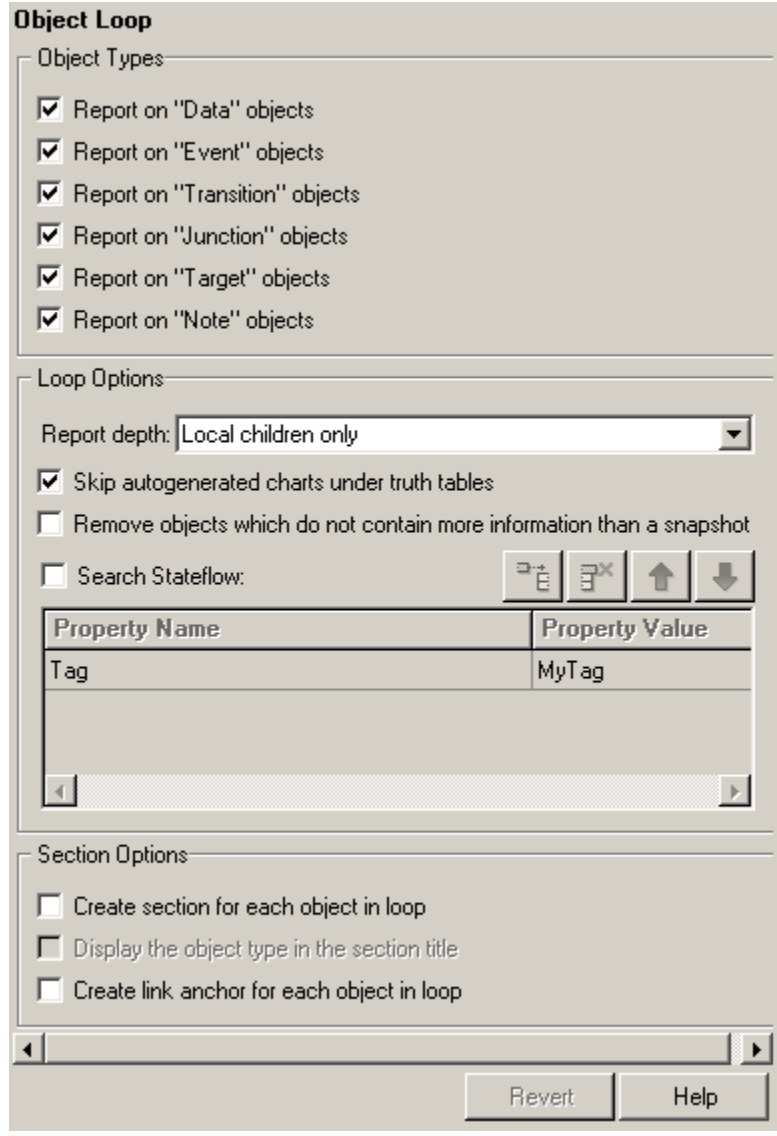
Stateflow

Description

The Object Loop component runs its child components for each Stateflow object. The component inserts a table into the generated report.

Attributes

The following figure shows the **Object Loop** Properties pane.


Object Loop
Object Types
Report on "Data" objects
Report on "Event" objects
Report on "Transition" objects
Report on "Junction" objects
Report on "Target" objects
Report on "Note" objects
Loop Options
Report depth:
Local children only
Skip autogenerated charts under truth tables
Remove objects which do not contain more information than a snapshot
Search Stateflow:
Property Name
Property Value
Tag
MyTag
Section Options
Create section for each object in loop
Display the object type in the section title
Create link anchor for each object in loop
Revert
Help

Object Loop

Object Types

Report on “Data” objects
Select this option to include Stateflow data objects in the loop.

Report on “Event” objects
Select this option to include Stateflow event objects in the loop.

Report on “Transition” objects
Select this option to include Stateflow transition objects in the loop.

Report on “Junction” objects
Select this option to include Stateflow junction objects in the loop.

Report on “Target” objects
Select this option to include Stateflow target objects in the loop.

Report on “Note” objects
Select this option to include Stateflow note objects in the loop.

Loop Options

Report depth
Select the level on which to loop:

- `Local children only` (Default) — Reports only on children one level down.
- `All objects` — Reports on all Stateflow objects.

Skip autogenerated charts under truth tables
Select this option to exclude autogenerated charts under truth tables from the report.

Remove objects which do not contain details than a snapshot
Select this option to exclude objects that only contain a snapshot. Other information can include a document link or a description.

Search Stateflow
Select this option to report only on Stateflow charts with certain property name/property value pairs. Enter the property name and the property value in the appropriate fields.

Section Options

Create section for each object in loop
Select this option to insert a section in the generated report for each object found in the loop.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the Stateflow object in the generated report.

Insert Anything into Report?

Yes, inserts a section if the **Create section for each object in loop** option is selected.

File Name

`rptgen_sf.csf_obj_loop`

Paragraph

Category

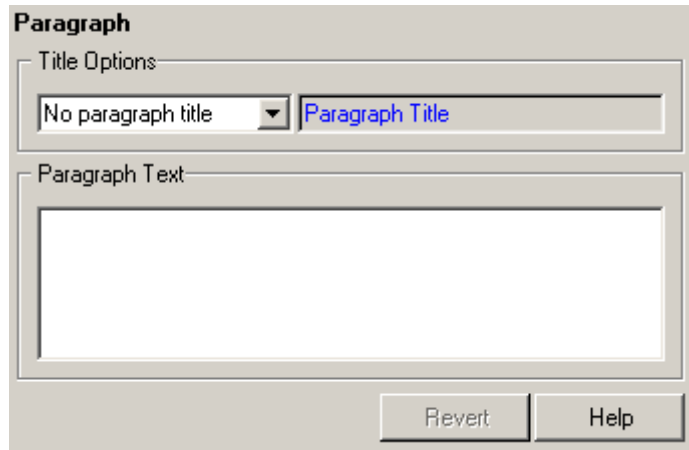
Formatting

Description

The Paragraph component inserts a paragraph into the report. The paragraph text is taken from a child text component or from text that you enter in the **Paragraph Text** field.

Attributes

The following figure shows the **Paragraph** Properties pane.



Title Options

No paragraph title

Select this option (the default) if you do not want the paragraph to have a title.

Get title from first child

Select this option if you want the title of the paragraph to be taken from the first child component. The first child component should be a Text component.

Custom title

Select this option and enter the title of the paragraph in this field if you do not want the title to come from the first child component.

Paragraph Text

Enter paragraph text in this field. If the Paragraph component has child components, the paragraph content is taken from the child components. Otherwise, the Paragraph component inserts text from this field.

If the Paragraph component does not have any child components and you do not enter any text in this field, the Paragraph component does not insert anything into the report.

You can use the `%<VariableName>` notation in this field if you want to insert the value of a variable from the MATLAB workspace. For more details about this notation, see "%<VariableName> Notation" on page 11-269 on the Text component reference page.

Insert Anything into Report?

Yes. Can be text, an image, etc.; depends on child components.

File Name

`rptgen.cfr_paragraph`

Requirements Table

Category

Requirements Management Interface

Description

The Requirements Table component creates a table that contains information from Simulink Verification and Validation. Objects can have multiple requirements. Each requirement is a row in the table.

Attributes

The following figure shows the **Requirements Table** Properties pane.

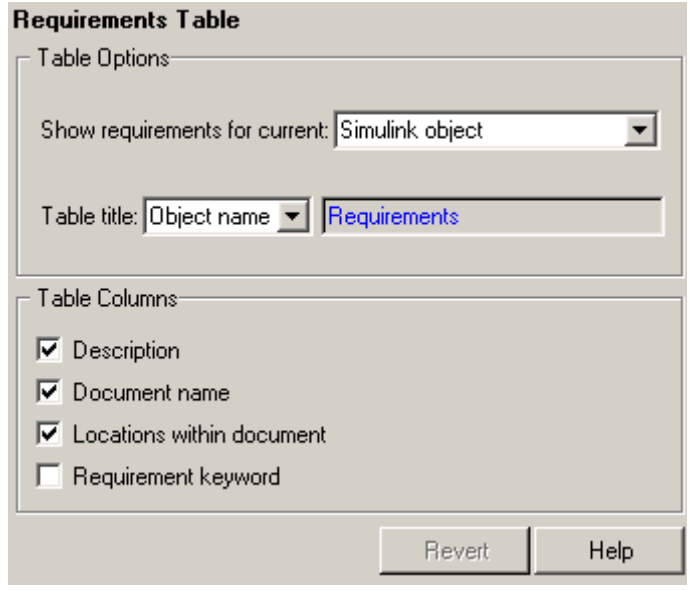


Table Options

Show requirements for current

Select the object type to display:

- `Simulink object`
- `Stateflow object`

Table title

Select a title for the table:

- `No title` (Default)

- `Object name`
- `Custom`

Table Columns

Description
Select this option to include the requirement's description in the table.

Document name
Select this option to include the requirement's document name in the table.

Locations within document
Select this option to include the requirement's locations within the document in the table.

Requirement keyword
Select this option to include the requirement's requirement keyword in the table.

Insert Anything into Report?

Yes. Table.

File Name

`RptgenRMI.CReqTable`

Scope Snapshot

Category

Simulink Blocks

Description

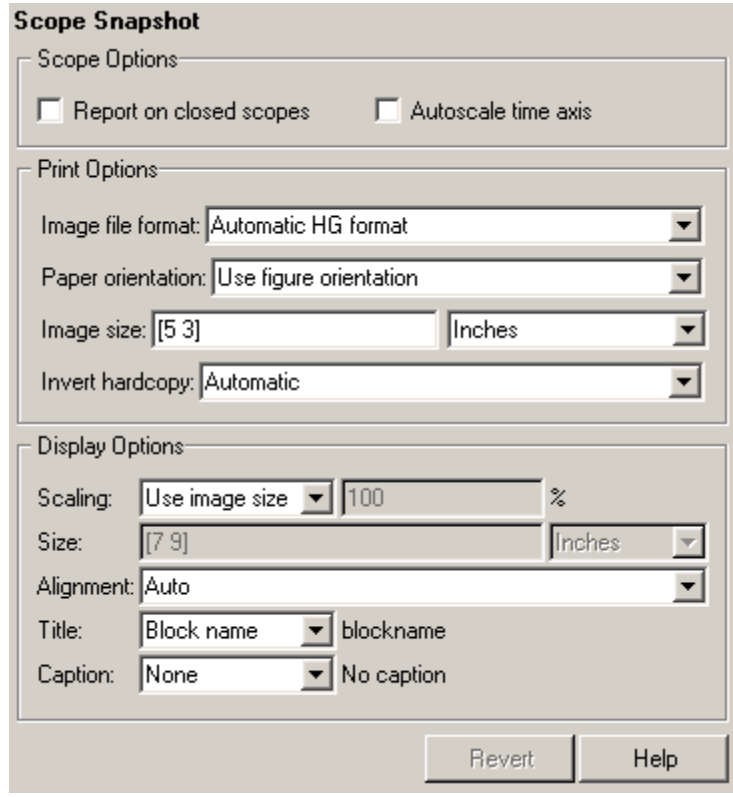
The Scope Snapshot component inserts images of Simulink scopes and XY graphs. If the model has not been simulated, scopes are empty. For details, see the `Model Simulation` reference page.

Like other components, the Scope Snapshot component must conform to the Report Generator component hierarchy. When building report templates with the Scope Snapshot component, keep the following points in mind:

- If the Scope Snapshot has the Model Loop as its parent or if it does not have any Simulink looping component as its parent, it includes all XY graphs and scopes in the current model.
- If the Scope Snapshot has the System Loop as its parent, it includes all XY graphs and scopes in the current system.
- If the Scope Snapshot has the Block Loop as its parent, it includes the current block if it is an XY graph or scope.
- If the Scope Snapshot has the Signal Loop as its parent, it includes all XY graphs and scopes connected to the current signal.
- If the Scope Snapshot does not have any of the Simulink looping components as its parent, it includes all XY graphs and scopes in all open models.

Attributes

The following figure shows the **Scope Snapshot** Properties pane.



Scope Options

Report on closed scopes

Select this option to take a snapshot of all scopes in context. This option forces closed scopes to open when the report is generating.

Autoscale time axis

Select this option to scale the Simulink scope's time axis to include the entire log.

Print Options

Image file format

Choose the image file format (e.g., JPEG, TIFF, etc.) from this list. Select `Automatic HG Format` to automatically choose the format

best suited for the output format that you chose in the Report component. Otherwise, choose an image format that your output viewer can read. `Automatic HG Format` is the default option. The options include

- `Automatic HG Format` (uses the Simulink file format selected in the Preferences dialog box)
- `Adobe Illustrator`
- `Bitmap (16m-color)`
- `Bitmap (256-color)`
- `Black and white encapsulated PostScript`
- `Black and white encapsulated PostScript (TIFF)`
- `Black and white encapsulated PostScript2`
- `Black and white encapsulated PostScript2 (TIFF)`
- `Black and white PostScript`
- `Black and white PostScript2`
- `Color encapsulated PostScript`
- `Color encapsulated PostScript (TIFF)`
- `Color encapsulated PostScript2`
- `Color encapsulated PostScript2 (TIFF)`
- `Color PostScript`
- `Color PostScript2`
- `JPEG high quality image`
- `JPEG medium quality image`
- `JPEG low quality image`
- `PNG 24-bit image`
- `TIFF - compressed`

- `TIFF - uncompressed`
- `Windows metafile`

Paper orientation

You can choose `Landscape`, `Portrait`, or `Rotated` as your paper orientation, or you can choose the `Use figure orientation` option. See the `orient` command in the MATLAB Reference Manual for information about paper orientation.

Image size

Enter the size of the Handle Graphics figure snapshot in the form `[w h]` (width/height). In the units text box, select `Inches`, `Centimeters`, `Points`, or `Normalized`.

Invert hardcopy

Handle Graphics figures have an `InvertHardcopy` property. This parameter inverts colors for printing; i.e., it changes dark colors to light colors and vice versa.

Select one of the following options:

- `Automatic` — Select this option to automatically change dark axes colors to a light axes color. If the axes color is a light color, it is not inverted.
- `Invert` — Select this option to change dark axes colors to light axes colors and vice versa.
- `Don't invert` — Select this option if you do not want to change the colors in the image displayed on the screen for printing.
- `Use figure's InvertHardcopy setting` — Select this option to use the `InvertHardcopy` property set in the Handle Graphics image.
- `Make figure background transparent` — Select this option to make the image background transparent.

Display Options

Scaling

Select `Use image size` if you want the image in your report to be the same size as the image on the screen.

Select `Fixed size` to specify the number and type of units.

Select `Zoom` to specify the percentage, the maximum size, and the units of measure.

Size

Enter the exact size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Fixed size` from the **Scaling** drop-down list.

Max size

Enter the maximum size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Zoom` from the **Scaling** drop-down list.

Units

Enter the units for the size of the snapshot. This field is active only if you choose `Zoom` or `Fixed size` in the **Image size** list box.

Alignment

Options are `Auto`, `Right`, `Left`, and `Center`.

Note This option is supported only in `PDF` and `RTF` reports.

Title

Select or enter a title for the snapshot figure.

- `Block name` — Use the block name as the title.
- `Full Simulink path name` — Use the Simulink path name as the title.
- `Custom` — Enter a custom title.

Caption

Select or enter a short text description for the snapshot figure.

- `No caption`
- `Automatic (use block description)` — Use the Simulink block's description as the caption.
- `Custom` — Enter a short text description for the snapshot figure.

Insert Anything into Report?

Yes. Image.

File Name

`rptgen_sl.csl_blk_scope`

Signal Loop

Category

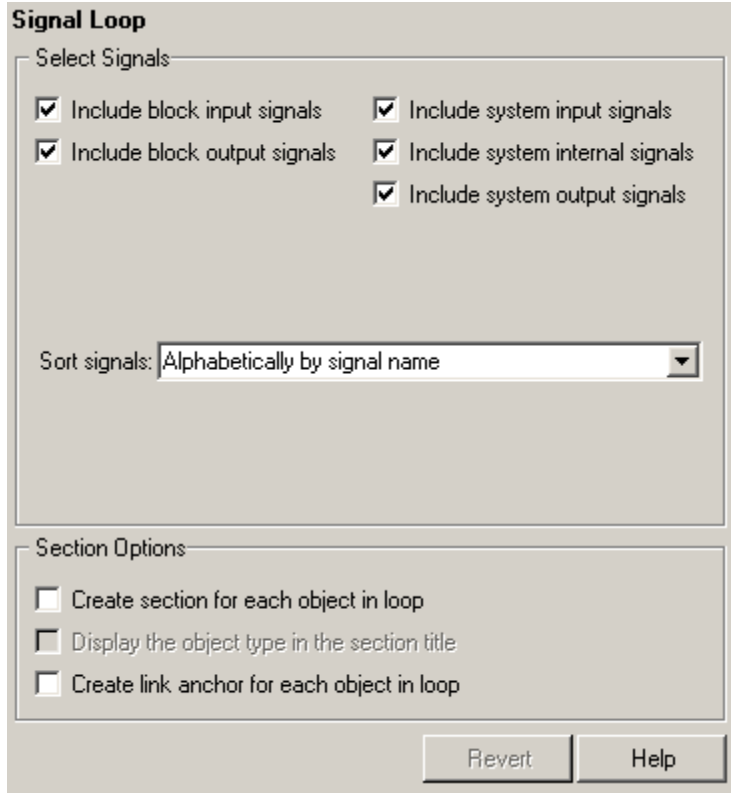
Simulink

Description

The Signal Loop component runs its child components for each signal contained in the current system, model, or block.

Attributes

The following figure shows the **Signal Loop** Properties pane.



Select Signals

Include block input signals

Select this option to loop on signals that feed into blocks. This option is valid when the Signal Loop component has a Block Loop component as its parent.

Include block output signals
Select this option to loop on signals that leave the block. This option is valid when the Signal Loop component has a Block Loop component as its parent.

Include system input signals
Select this option to loop on signals coming from inports. This option is valid when the Signal Loop component has a System Loop component as its parent.

Include system internal signals
Select this option to loop on system internal signals. This option is valid when the Signal Loop component has a System Loop component as its parent.

Include system output signals
Select this option to loop on signals going to outports. This option is valid when the Signal Loop component has a System Loop component as its parent.

Sort signals
Use this section to select how the signals are sorted:

- Select `Alphabetically by signal name` to sort signals alphabetically by name.
- Select `Alphabetically by signal name (exclude empty)` to sort signals alphabetically by name as long as the signals have names.
- Select `Alphabetically by system name` to sort by parent systems. In this option, the systems are sorted alphabetically, and signals in each system are listed, but in no particular order.
- Select `By signal depth` to sort signals by their depth in the model.

Section Options

Create section for each object in loop
Select this option to insert a section in the generated report for each object found in the loop.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the object in the generated report.

When using the Signal Loop component, keep the following points in mind:

- If the Signal Loop has the Model Loop as its parent, it loops on all signals in the current model.
- If the Signal loop has the System Loop as its parent, it loops on all signals in the current system. You can choose not to report on the following types of signals by clearing the corresponding option:
  - `System input signals`
  - `System output signals`
  - `System internal signals`
- If the Signal Loop has another Signal Loop as a parent, it loops on the current signal.
- If the Signal Loop has the Block Loop as a parent, it loops on all signals connected to the current block. You can choose not to report on the following types of signals by clearing the corresponding option:
  - `Block input signals`
  - `Block output signals`
- If the Signal Loop does not have the Model Loop, System Loop, Signal Loop, or Block Loop as its parent, it loops on all signals in all models. You can choose not to report on the following types of signals by clearing the corresponding option:
  - `Block input signals`
  - `Block output signals`

- `System input signals`
- `System output signals`
- `System internal signals`

Insert Anything into Report?

Yes, inserts a section if the **Create section for each object in loop** option is selected.

File Name

`rptgen_sl.csl_sig_loop`

Simulink Automatic Table

Category

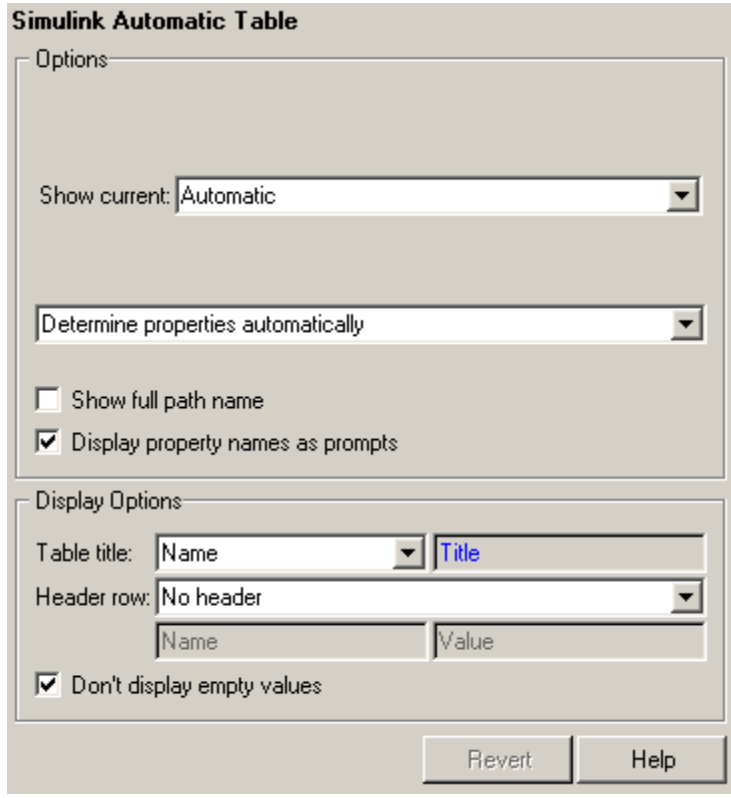
Simulink

Description

The Simulink Automatic Table component inserts a two-column table that contains details for the selected model, system, signal, or block into a generated report.

Attributes

The following figure shows the **Simulink Automatic Table** Properties pane.



Options

Show current

Choose the Simulink feature to include in the generated report:

- `Automatic` — Uses the context of the parent loop.
- `Model`
- `System`
- `Block`
- `Annotation`
- `Signal`

Properties list

Choose whether to select the properties manually or automatically:

- `Determine properties automatically` — Uses blocks and dialog box properties.
- `Show properties` — Enter properties manually.

Show full path name

Select this option to show the full file path name for the selected Simulink model.

Display property names as prompts

Select this option to show property names as prompts in the generated report. The report includes the dialog box string instead of the underlying code property.

Display Options

Table title

Choose a title for the display in the generated report. To generate a title automatically from the parameter, select `Name`, which uses the object name as the title. To enter a title, select `Custom`. To specify no title, select `No title`.

Header row

Use this option to select a header row for the table in the generated report. To include no header row, select `No header`. To include a header row with columns for name and object type, select `Type and Name`. When selected, this option creates a header row for the table with object name and type. To include a custom header, select `Custom`.

Don't display empty values
Select this option to exclude empty parameters in the generated report.

Insert Anything into Report?

Yes. Table.

File Name

`rptgen_sl.csl_auto_table`

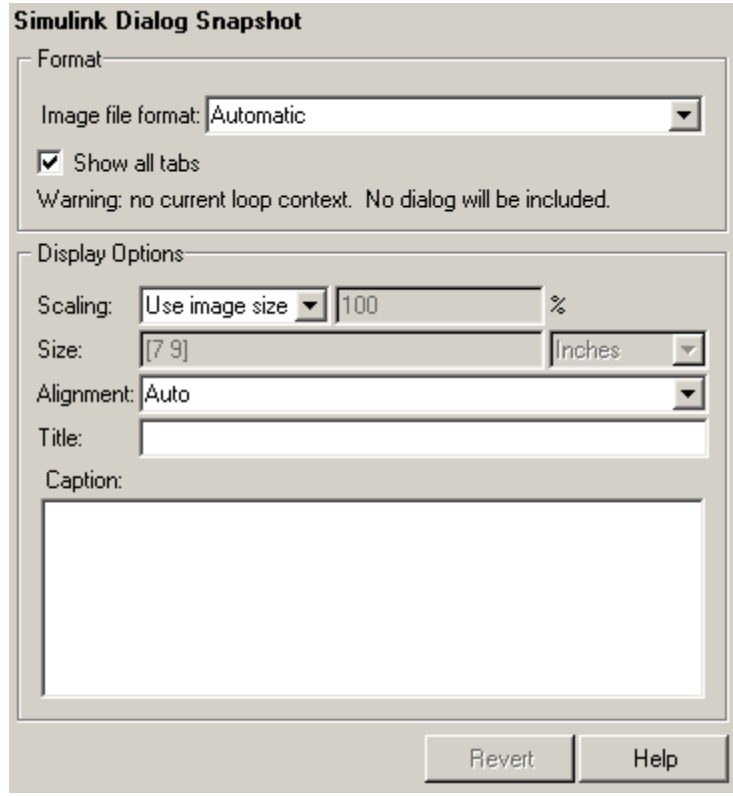
Category

The Simulink Dialog Snapshot component takes snapshots of Simulink editor dialog boxes. You can display the current settings associated with an object or document the appearance of your custom mask dialog boxes.

The Dialog Snapshot component reports on the current reported Simulink object by context. If it is the child of a Block Loop, the current reported Block's dialog box is documented. If it is the child of a System Loop, the current reported System's dialog box is documented.

Attributes

The following figure shows the **Simulink Dialog Snapshot** Properties pane.



Simulink Dialog Snapshot

Format

Image file format

Choose the format for the snapshot image file. The **automatic** format chooses BMP format for PDF files and PNG for other formats.

Show all tabs

Select this check box if you want to automatically generate images for all the tabs for the dialog box. If the check box is not selected, Report Generator creates an image of only the first tab.

Display Options

Scaling

Select `Fixed size` to specify the number and type of units.

Select `Zoom` to specify the percentage, the maximum size, and the units of measure.

Select `Use image size` if you want the image in your report to be the same size as the image on the screen.

Size

Enter the exact size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Fixed size` from the **Scaling** drop-down list.

Max size

Enter the maximum size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Zoom` from the **Scaling** drop-down list.

Units

Enter the units for the size of the snapshot. This field is active only if you choose `Zoom` or `Fixed size` in the **Image size** list box.

Alignment

Options are `Auto`, `Right`, `Left`, and `Center`.

Note This option is supported only in `PDF` and `RTF` reports.

Title
Type the text you want to appear above the snapshot.

Caption
Type the text you want to appear below the snapshot.

Insert Anything into Report?

Yes. Snapshot.

File Name

`rptgen_sl.Cdialog boxesnapshot`

Simulink Functions and Variables

Category

Simulink

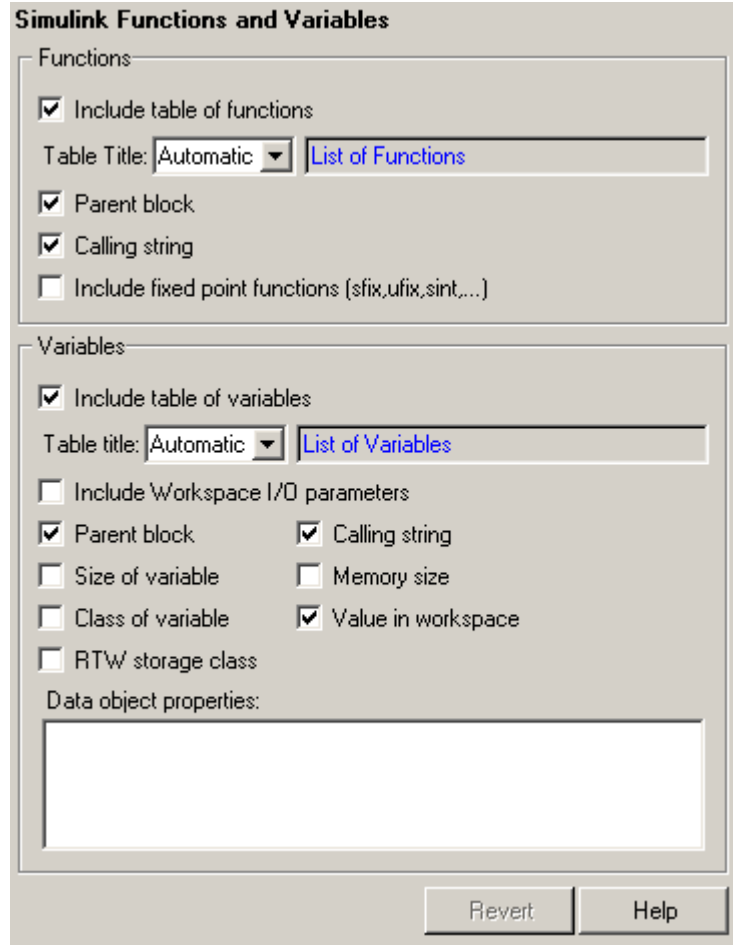
Description

The Simulink Functions and Variables component creates a table that shows all workspace variables and MATLAB functions used by reported blocks in a Simulink model. The current model and systems in which the blocks appear are specified in the Model Loop component.

For example, a Simulink Gain block might have a string `cos(x)` instead of a number. Simulink looks for a variable `x` in the workspace and uses the `cos` function.

Attributes

The following figure shows the **Simulink Functions and Variables** Properties pane.



Functions

Include table of functions

Select this option to include a table of Simulink functions in the generated report.

Table Title

Choose a title for the table in the generated report:

- `Automatic` — Select this option to generate a title automatically from the parameter.
- `Custom` — Select this option to enter a title manually.

Parent block
Select this option to include a column in the table that includes the name of the block, which contains the reported variable.

Calling string
Select this option to include the M-code that calls the reported variable.

`Include fixed-point functions` (`sfix`, `ufix`, `sint`, ...)
Select this option to include fixed-point functions. If you use any fixed-point functions in your model and want those functions to appear in the generated report, select this option.

Variables

Include table of variables
Select this option to include a table of Simulink variables in the generated report.

Table title
Choose a title for the table in the generated report:

- `Automatic` — Select this option to generate a title automatically from the parameter.
- `Custom` — Select this option to enter a title manually.

Include Workspace I/O parameters
Select this option to report on variables that contain parameters with time vectors and state matrices. These parameters are set in the **Workspace I/O** pane in the Simulation Parameters dialog box in a Simulink model.

In the following table, if any of the entries in the first column are on, the component looks for the variable listed in the second column in the workspace. If the component finds the variable, it includes it in the report.

LoadExternalInput	ExternalInput
SaveTime	TimeSaveName
SaveState	StateSaveName
SaveOutput	OutputSaveName
LoadInitialState	InitialState
SaveFinalState	FinalStateName

Parent block
Select this option to include the name of the block that contains the reported variable.

Calling string
Select this option to include the M-code that calls the reported variable.

Size of variable
Select this option to include the size of the reported variable.

Class of variable
Select this option to include the variable class to which the reported variable belongs.

Memory size
Select this option to include the amount of memory in bytes that the reported variable needs.

Value in workspace
Select this option to include the actual value of the reported variable. Note that very large arrays might be displayed as `[MxN CLASS]`. For example, if you have a 300-by-200 double array, it appears in the report as `[300x200 DOUBLE]`.

RTW storage class
Select this option to include the storage class of the reported variable.

The title of this column is **Storage Class**. This option looks at the model's `TunableVars` property to see if any of the model variables

specify their storage class. If the storage class is specified, the model variables table displays `TunableVarsStorageClass` and `TunableVarsTypeQualifier` in a table column.

The column entries are `TunableVarsStorageClass (TunableVarsTypeQualifier)` if `TunableVarsTypeQualifier` is not empty. If `TunableVarsTypeQualifier` is empty, the column entry is `TunableVarsStorageClass`.

The values for `TunableVarsStorageClass` are:

- `Exported Global`
- `Auto`
- `ImportedExtern`
- `ImportedExtern Pointer`

Data object properties
Enter data object properties to report on.

Example

This is an example of a table created by the Model Variables component. This property table reports on the variables in the `Controller` in the `F14` demo model.

Variable Name	Parent Blocks	Calling String	Value
Ka	`f14/Controller/Gain3`	Ka	`0.677`
Kf	`f14/Controller/Gain`	Kf	`-1.746`
Ki	`f14/Controller/Proportional plus integral compensator`	[Ki]	`-3.864`
Kq	`f14/Controller/Gain2`	Kq	`0.8156`

Insert Anything into Report? Yes. Table.

File Name `rptgen_sl.csl_obj_fun_var`

Simulink Library Information

Category

Simulink

Description

The Simulink Library Information component inserts a table that lists any library links in the current model, system, or block.

Attributes

The following figure shows the **Simulink Library Information** Properties pane.

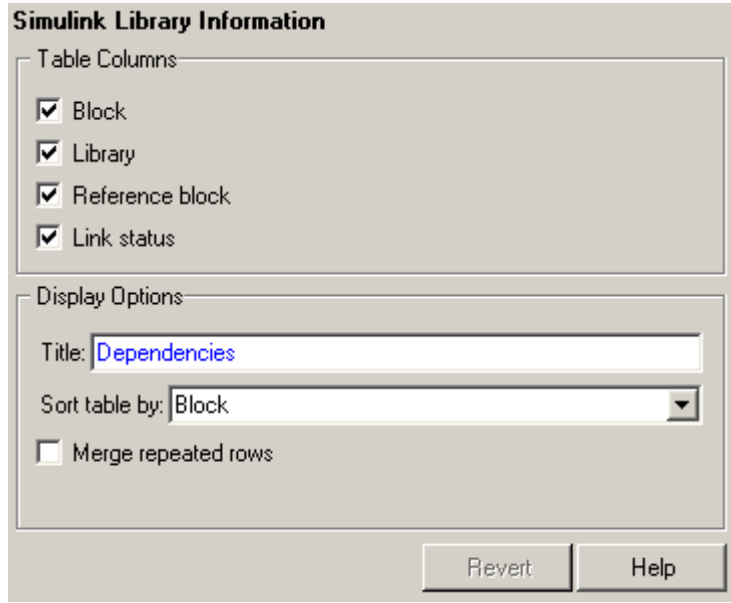


Table Columns

Block

Select this option to include the Simulink block name in the generated table.

Library

Select this option to include the Simulink library root name in the generated table.

Reference block
Select this option to include the Simulink reference block name in the generated table.

Link status
Select this option to include the link status in the generated table.

Display Options

Title
Enter a title for the display in the generated report.

Sort table by
To sort the table by block name, select `Block`. To sort the table by library name, select `Library`. To sort the table by reference block name, select `Reference Block`. To sort the table by link status, select `Link Status`.

Merge repeated rows
Select this option to merge sorted rows in the generated table.

Example

If you select the **Merge repeated rows** option, the generated table will collapse identical rows into one. For example, the following table, which is sorted on the Reference Block column, is generated from the `aero_guidance` model without the option selected:

Block	Library	Reference Block	Status
Equations of Motion (Body Axes)	Aerospace	Equations of Motion (Body Axes)	resolved
Incidence & Airspeed	Aerospace	Incidence & Airspeed	resolved
Fin Actuator	Aerospace	2nd Order Nonlinear Actuator	resolved
3DoF Animation	Aerospace	3DoF Animation	resolved

Block	Library	Reference Block	Status
Atmosphere	Aerospace	Atmosphere model	resolved
Cm	Simulink	Interpolation (n-D) using PreLookup	resolved
Cx	Simulink	Interpolation (n-D) using PreLookup	resolved
Cz	Simulink	Interpolation (n-D) using PreLookup	resolved
Kg	Simulink	Interpolation (n-D) using PreLookup	resolved
Ki	Simulink	Interpolation (n-D) using PreLookup	resolved
Alpha Index	Simulink	PreLookup Index Search	resolved
Mach Index	Simulink	PreLookup Index Search	resolved
Mach Index	Simulink	PreLookup Index Search	resolved
|Alpha| Index	Simulink	PreLookup Index Search	resolved

With the **Merge repeated rows** option selected, the generated table collapses rows in the Block column so that each row in the Reference Block column is unique:

Block	Library	Reference Block	Status
Equations of Motion (Body Axes)	Aerospace	Equations of Motion (Body Axes)	resolved
Incidence & Airspeed	Aerospace	Incidence & Airspeed	resolved

Block	Library	Reference Block	Status
Fin Actuator	Aerospace	2nd Order Nonlinear Actuator	resolved
3DoF Animation	Aerospace	3DoF Animation	resolved
Atmosphere	Aerospace	Atmosphere model	resolved
Cm Cx Cz Kg Ki	Simulink	Interpolation (n-D) using PreLookup	resolved
Alpha Index Mach Index Mach Index |Alpha| Index	Simulink	PreLookup Index Search	resolved

Insert Anything Into Report?

Yes. Table.

File Name

`rptgen_sl.CLibinfo`

Simulink Linking Anchor

Category

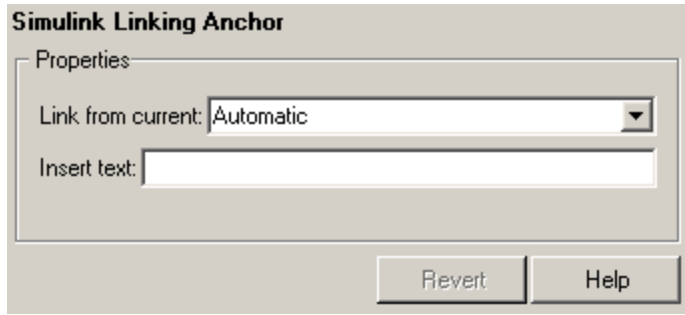
Simulink

Description

The Simulink Linking Anchor component designates a location to which other links point. This component should have the Model Loop, System Loop, Block Loop, or Signal Loop component as its parent.

Attributes

The following figure shows the **Simulink Linking Anchor** Properties pane.



Insert text

If you want text to appear after the linking anchor, enter the text in this field.

Link from current

This option sets the current model, system, block, or signal as the linking anchor:

- Select `Automatic` to select the appropriate model, system, block, or signal as a linking anchor automatically. If the Model Loop component is the parent component, the linking anchor is set on the current model. Similarly, if the Block Loop or Signal Loop is the parent component, the linking anchor is inserted for the current system, block, or signal, respectively.
- Select `Model` to set the linking anchor to the current model.
- Select `System` to set the linking anchor to the current system.

- Select `Block` to set the linking anchor to the current block.
- Select `Signal` to set the linking anchor to the current signal.

> **Note** You should not use the Simulink Linking Anchor component to create an anchor for an object that has anchor created elsewhere, such as a loop component or summary table component. Each object should have only one anchor per report. See the Simulink Summary Table component reference page.

- Select `Annotation` to set the linking anchor to the current annotation.

Insert Anything into Report?

Yes. A link, and possibly text, depending on attribute choices.

File Name

`rptgen_sl.csl_obj_anchor`

Simulink Name

Category

Simulink

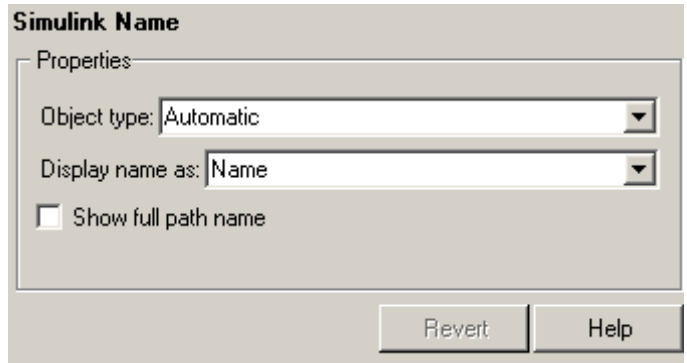
Description

The Simulink Name component inserts the name of a Simulink model, system, block, or signal into the report.

You can use this component as the first child component of a Chapter/Subsection component, which allows the current Simulink model, system block, or signal name to be the chapter or section title.

Attributes

The following figure shows the **Simulink Name** Properties pane.



Object type

- Select `Automatic` to automatically select the appropriate model, system, block, or signal name as the Simulink object name to include in the report. If the Model Loop component is the immediate parent component, the object name is the current model name. Similarly, if the System Loop, Block Loop, or Signal Loop is the parent component, then the object name is the name of the current system, block, or signal, respectively.
- Select `Model` to include the current model name in the report.
- Select `System` to include the current system name in the report.
- Select `Block` to include the current block name in the report.

- Select `Signal` to include the current signal name in the report. If the signal name is empty, the signal `<handle>` is included in the report. The signal handle is a unique numerical identifier to that signal.
- Select `Annotation` to include the current annotation name in the report.

Display name as

Select one of the options listed below to display the Simulink object name in the report:

- `Name` — For example, `f14`
- `Type Name` — For example, `Model f14`
- `Type - Name` — For example, `Model - f14`
- `Type:  Name` — For example, `Model:  f14`

Show full path name

Select this option to show the full path name for a system or block. For example, if you choose this option for a block, the Simulink block name appears as `<Model Name>/<System Name>/<Block Name>`. Note that this option is not available for models and signals.

Insert Anything into Report?

Yes. Text.

File Name

`rptgen_sl.csl_obj_name`

Simulink Property

Category

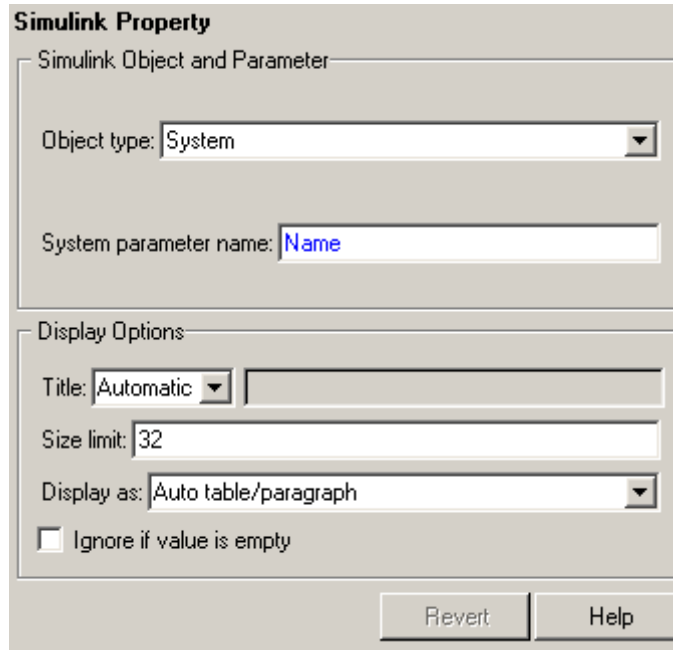
Simulink

Description

The Simulink Property component inserts a single property name/property value pair for the current Simulink model, system, block, or signal.

Attributes

The following figure shows the **Simulink Property** Properties pane.



Simulink Object and Parameter

Object type

Choose the Simulink object type to include in the report:

- `System`
- `Model`
- `Block`

- `Signal`
- `Annotation`

System parameter name
Enter a Simulink parameter name to include in the generated report. If `Model` is selected in **Object type**, this option appears as **Model parameter name**. If `Block` is selected in **Object type**, this option appears as **Block parameter name**. If `Signal` is selected in **Object type**, this option appears as **Signal parameter name**.

Display Options

Title
Choose a title for the display in the generated report. To use the parameter name as the title, select `Automatic`. To enter a title, select `Custom`. To specify no title, select `None` (the default).

Size limit
Use this size to limit the width of the display in the generated report. Units are in pixels. When creating a table, the size limit is the hypotenuse of the table width and height [`sqrt(w^2+h^2)`]. When creating text, the size limit is the number of characters squared. If the size limit is exceeded, the variable is rendered in condensed form, such as `[64x64 double]`. Setting a size limit of zero will ignore the size limit and always display the variable, no matter how large.

Display as
Choose a display style from the menu:

- `Auto table/paragraph` — Displays as a table or paragraph based on the information.
- `Table` — Displays as a table.
- `Paragraph` — Displays as a text paragraph.
- `Inline text` — Displays as inline, which fits in line with the surrounding text.

Ignore if value is empty
Select this option to exclude the parameter from the generated report if the parameter is empty.

Insert Anything into Report?

Yes. Text.

File Name

`rptgen_sl.csl_property`

Category

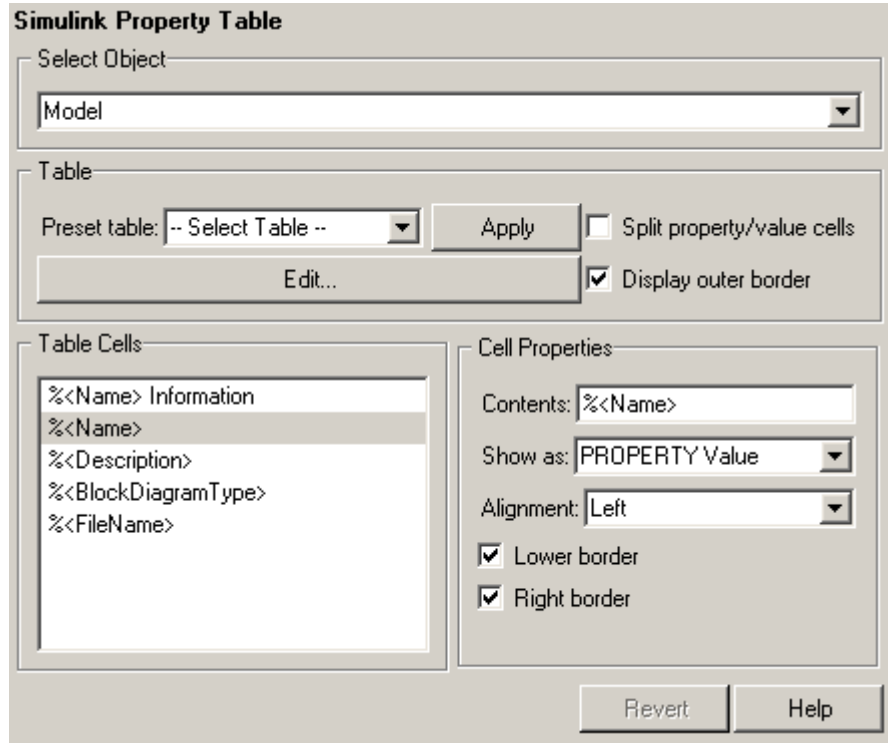
Simulink

Description

The Simulink Property Table component inserts a table that reports on model-level property name/property value pairs.

Attributes

The following figure shows the **Simulink Property Table** Properties pane.



Select Object

Choose the object for the property table in the generated report:

- `Model`

- System
- Block
- Signal
- Annotation

For details about selecting object types in property table components, see “Selecting Object Types” on page 7-27.

Table

A preset table is a table that is already formatted and set up. You can select a preset table in the preset table list in the upper-left corner of the attributes page. To apply a preset table, select the table and click **Apply**.

Preset table

Choose a type of table to display the object property table:

- Default
- Simulation parameters
- Version information
- RTW information
- Summary (req. RTW)
- Blank 4x4

Split property/value cells

Select this option to split property name/property value pairs into separate cells. To have the property name and property value appear in adjacent horizontal cells in the table, select the **Split property/value cells** check box. In this case, the table is in split mode and there can be only one property name/property value pair in a cell. If you have more than one name/property pair in a cell, only the first pair appears in the report. All subsequent pairs are ignored.

To have the property name and property value appear together in one cell, clear the **Split property/value cells** check box. This is nonsplit mode. In nonsplit mode, you can have more than one property name/property value pair and you can also have text.

If you want to switch from nonsplit mode to split mode, make sure that you only have one property name/property value pair per table cell before you switch modes. If you have more than one property name/property value pair or any text, only the first property name/property value pair appears in the report; subsequent pairs and text are omitted.

Display outer border

Select this option to display the outer border of the table in the generated report.

Table Cells

Use this field to select table properties to modify. The selection in this pane affects the available fields in the **Cell Properties** pane.

Cell Properties

The visible options in the **Cell Properties** pane depend on the object selected in the **Table Cells** pane. If `%<Name> Information` is selected, only the **Contents** and **Show** options are displayed. If any other object is selected in the **Table Cells** pane, the **Lower border** and **Right border** options display.

Contents

Use this field to modify the contents of the table cell selected in the **Table Cells** pane.

Show as

Choose the format for the contents of the table cell:

- `Value`
- `Property Value`
- `PROPERTY Value`
- `Property:  Value`

- `PROPERTY: Value`
- `Property - Value`
- `PROPERTY - Value`

Alignment

Choose the alignment of the contents of the selected table cell in the **Table Cells** pane.

- `Left`
- `Center`
- `Right`
- `Double justified`

Lower border

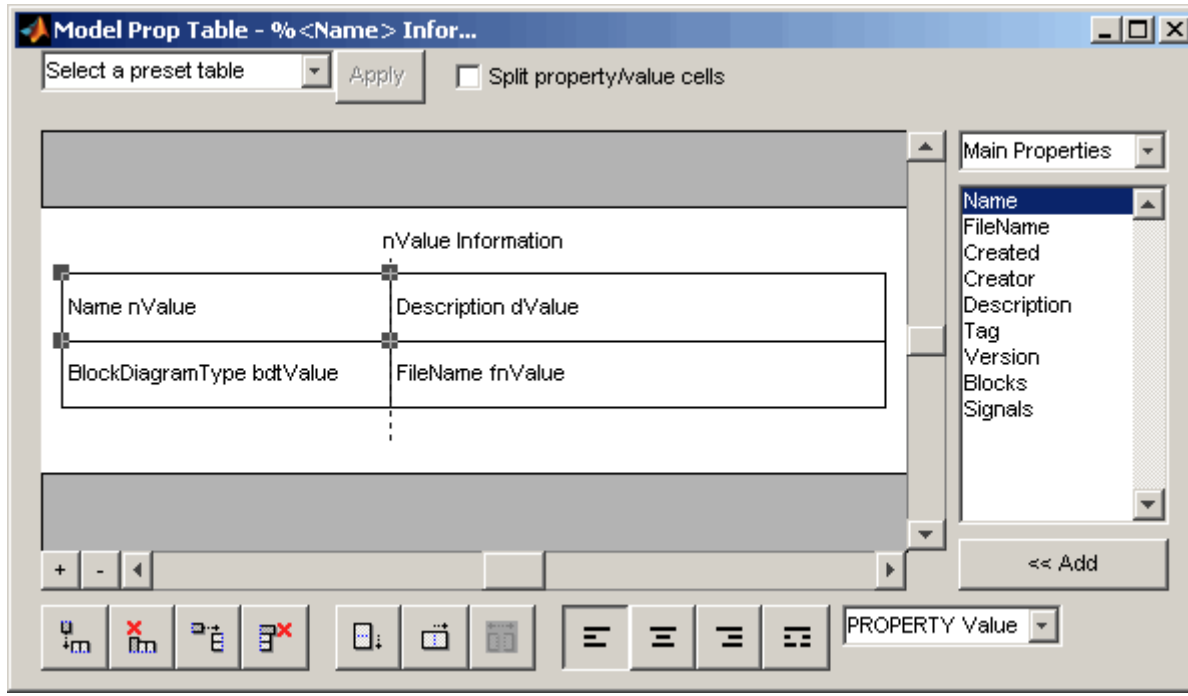
Select this option to display the lower border of the table in the generated report.

Right border

Select this option to display the right border of the table in the generated report.

Creating Custom Tables

To create a custom table, you can first select a preset table and then edit it. You might want to start with the `Blank 4x4` preset table. You can add and delete rows and you can add properties. To open the Edit Table dialog box, click **Edit**. The dialog box appears.



For details about using this dialog box to create custom property tables, see “Working with Property Table Components” on page 7-14.

Note that if you do not have Real-Time Workshop, `Summary (req RTW)` is not listed in this list, because summary properties require Real-Time Workshop. If you are using a report template that contains a summary property, the property name appears in the report, but the property value does not.

Example

The table below is an example of a model property table. It is the **Simulation Parameters** preset table. This property table is reporting the F14 demo model.

Solver ode45	*ZeroCross* on	*StartTime* 0.0 *StopTime* 60
RelTol 1e-4	*AbsTol* 1e-6	*Refine* 1
InitialStep auto	*FixedStep* auto	*MaxStep* auto
LimitMaxRows off	*MaxRows* 1000	*Decimation* 1

Insert Anything into Report?

Yes. Table.

File Name

`rptgen_sl.csl_prop_table`

Category

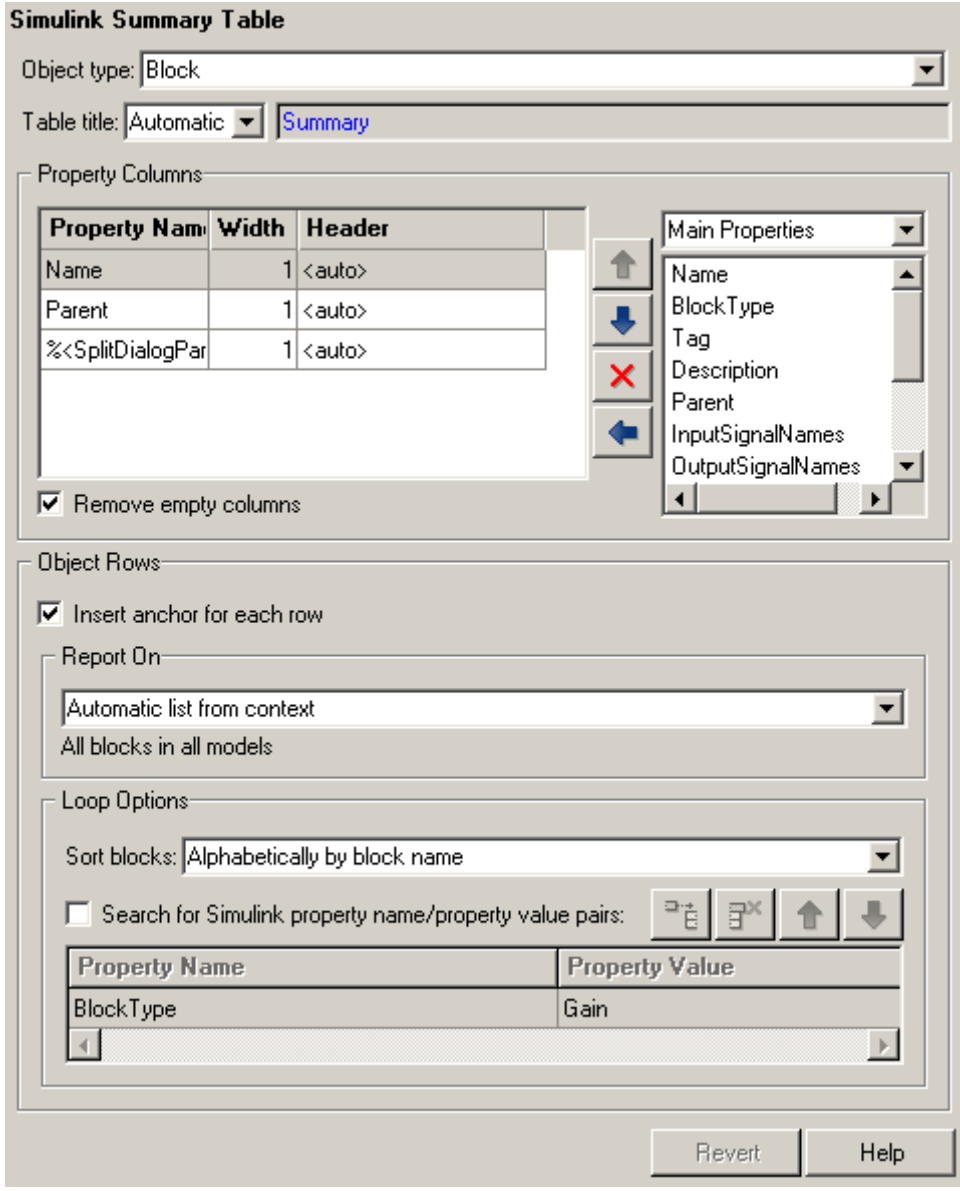
Simulink

Description

The Simulink Summary Table component shows properties or parameters of the selected Simulink models, systems, blocks, or signals in a table.

Attributes

The following figure shows the **Simulink Summary Table** Properties pane for a **Block** object:


Simulink Summary Table
Object type: Block
Table title: Automatic
Summary
Property Columns
Property Nam
Width
Header
Name
1
<auto>
Parent
1
<auto>
%<SplitDialogPar
1
<auto>
Main Properties
Name
BlockType
Tag
Description
Parent
InputSignalNames
OutputSignalNames
Remove empty columns
Object Rows
Insert anchor for each row
Report On
Automatic list from context
All blocks in all models
Loop Options
Sort blocks: Alphabetically by block name
Search for Simulink property name/property value pairs:
Property Name
Property Value
BlockType
Gain
Revert
Help

Object type

Choose the object type to display in the generated report:

- `Block` (Default)
- `Model`
- `System`
- `Signal`
- `Annotation`

The menu item selected in this menu affects the options available in the **Property Columns** pane.

Table title

Choose a title for the summary table in the generated report:

- `Automatic` — Select this option to generate a title automatically from the parameter.
- `Custom` — Select this option to enter a title manually.

Property Columns

The list in the left of the Property Columns pane displays the object properties to be included in the summary table in the generated report. To add a property, select the appropriate property level in the text box on the left. Next, in the text box on the right, select the property that you want to add and click **Add**.

To delete a property, select the property name and click **Delete**.

Note The `%<SplitDialogParameters>` property is a unique property that you can specify for Simulink summary tables where the object type is `Block`. This property of the Simulink Summary Table component generates multiple summary tables, grouped by block type. Each summary table group contains the dialog box parameters for that block. Dialog box parameters are the specific properties used by that block type and are usually visible on the block's editing dialog box.

Note Some entries in the list of available properties (such as `Depth`) are "virtual" properties which cannot be accessed using the `get_param` command. The properties used for property/value filtering in the block and System Loop components must be retrievable by the `get_param`. Therefore, you cannot configure your summary table to report on all blocks of `Depth == 2`.

Remove empty columns
Select this option to remove empty columns from the summary table.

Object Rows

Insert anchor for each row
Select this option to insert an anchor for each row in the summary table.

Report On

- `Automatic list from context`

  Select this option to report on all blocks in the current context. The context is set by the parent of the Simulink Summary Table component.

- `Custom - use block list`

  Select this option to report on a list of blocks that you specify, and enter the block names in the corresponding field. You must use the full path name when specifying blocks. If you want to sort the blocks in this list, select the `Sort blocks in list` option and then select a sorting option from the **Sort blocks** list (in the **Loop Options** section). Blocks are reported on in the order that they are sorted. If you do not select this option, blocks are reported on in the order that they are listed in the block list.

  You can enter `%<VariableName>` if you want to insert the value of a variable from the MATLAB workspace. The %<> notation can be a string or cell array. For more details about

this notation, see “%<VariableName> Notation” on page 11-269 on the Text component reference page.

For example, if you want to report on the `theta dot` integrator block and the `theta` integrator block in the demo model `simppend`, you could enter the following in the block list:

```
simppend/theta dot

%<Z>
```

If you define `Z` in the workspace as the following,

```
Z={'simppend/theta'}
```

then Report Generator reports on the following blocks:

```
simppend/theta dot

simppend/theta
```

Loop Options

You can choose block sorting options and reporting options in this pane.

Sort blocks

- Select `Alphabetically by block name` to sort blocks alphabetically by their names.
- Select `Alphabetically by system name` to sort systems alphabetically.

  In this option, the systems are sorted alphabetically, and blocks in each system are listed, but in no particular order.
- Select `Alphabetically by full Simulink path` to sort blocks alphabetically by Simulink path.
- Select `By block type` to sort blocks alphabetically by block type.
- Select `By block depth` to sort blocks by their depth in the model.
- Select `By traversal order` to sort blocks by traversal order.

- Select `By simulation order` to sort blocks by execution order.

Search for Simulink property name/property value pairs

Select this option to report only on Simulink blocks with certain property name/property value pairs. Enter the property name and the property value in the appropriate fields.

Example

If you are reporting on the demo model F14, and you choose the following options, the summary table shown below is generated:

- Sort on systems by system depth.
- Name and blocks are the system parameters to be included in the table.

Name	Blocks
f14	u, Actuator Model, Aircraft Dynamics Model, Angle of Attack, Controller, Dryden Wind Gust Models, Gain, Gain1, Gain2, Gain5, More Info, More Info1, Nz pilot calculation, Pilot, Pilot G force Scope, Stick Input, Sum, Sum1, alpha (rad), Nz Pilot (g)
Aircraft Dynamics Model	Elevator Deflection d (deg), Vertical Gust wGust (ft/sec), Rotary Gust qGust (rad/sec), Gain3, Gain4, Gain5, Gain6, Sum1, Sum2, Transfer Fcn.1, Transfer Fcn.2, Vertical Velocity w (ft/sec), Pitch Rate q (rad/sec)
Controller	Stick Input (in), alpha (rad), q (rad/sec), Alpha-sensor Low-pass Filter, Gain, Gain2, Gain3, Pitch Rate Lead Filter, Proportional plus integral compensator, Stick Prefilter, Sum, Sum1, Sum2, Elevator Command (deg)
Dryden Wind Gust Models	Band-Limited White Noise, Q-gust model, W-gust model, Wg, Qg

Name	Blocks
More Info	None
More Info1	None
Nz pilot calculation	w, q, Constant, Derivative, Derivative1, Gain1, Gain2, Product, Sum1, Pilot g force (g)

Insert Anything into Report?

Yes. Table.

File Name

`rptgen_sl.csl_summ_table`

State Loop

Category

Stateflow

Description

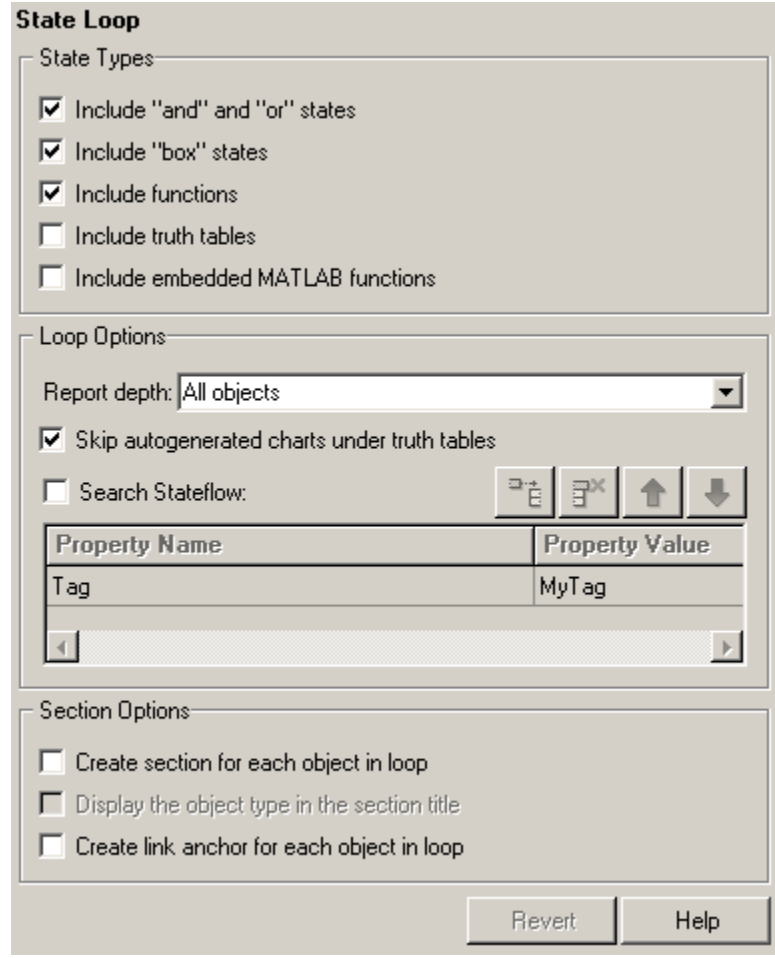
The State Loop component runs its children for all states in its context. For example, if a State Loop component is parented by a Model Loop component, the State Loop component's context is the Model Loop. If the State Loop has no context, it loops on all states in all open models.

When using the State Loop component, keep the following points in mind:

- If the State Loop has the Model Loop as its parent, it includes all states in the models.
- If the State Loop has the System Loop as its parent, it includes all states in the systems.
- If the State Loop has the Machine Loop as its parent, it includes all states in the machines.
- If the State Loop has the Chart Loop as its parent, it includes all states in the charts.
- If the State Loop has another State Loop as its parent, it includes all states in the current state. Use this context with a shallow loop.

Attributes

The following figure shows the **State Loop** Properties pane.



State Types

Include “and” and “or” states

Select this option to include AND and OR states in the loop.

Include “box” states

Select this option to include “box” states in the loop.

Include functions
Select this option to include “function” states in the loop.

Include truth tables
Select this option to include truth tables in the loop.

Include embedded MATLAB functions
Select this option to include Embedded MATLAB functions in the loop.

Loop Options

Report depth
Select the level on which to loop:

- `Local children only`
- `All objects`

Skip autogenerated charts under truth table
Select this option to keep autogenerated state objects under truth tables from being displayed in the report.

Search Stateflow
Select this option to indicate specific states to include in the loop.

Section Options

Create section for each object in loop
Select this option to insert a section in the generated report for each object found in the loop.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the object in the generated report.

Insert Anything into Report?

Yes, section, if the **Create section for each object in loop** option is selected.

File Name `rptgen_sf.csf_state_loop`

Stateflow Automatic Table

Category

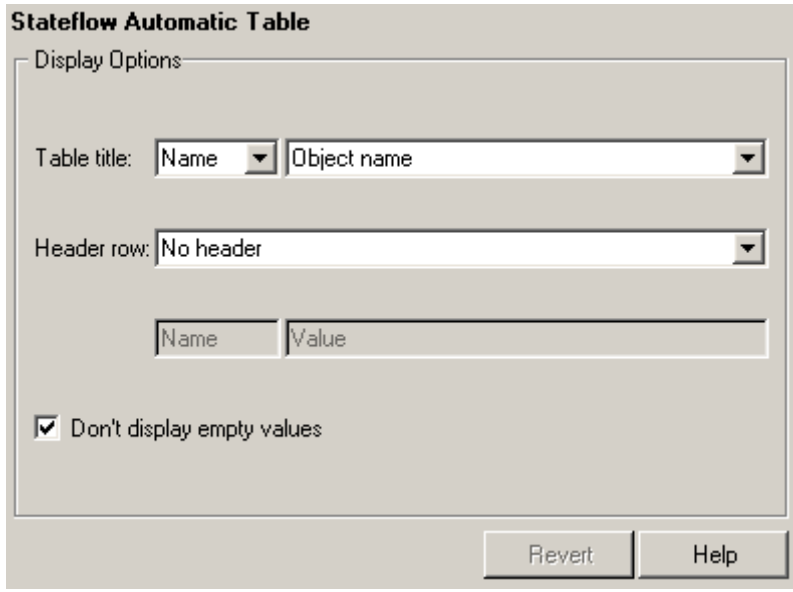
Stateflow

Description

The Stateflow Automatic Table component inserts a table that contains the properties of the current Stateflow object as set in a Machine Loop component, State Loop component, Chart Loop component, or Graphics Object Loop component.

Attributes

The following figure shows the **Stateflow Automatic Table** Properties pane.



Display Options

Table title

Use this option to select a title for the table in the generated report. To include no title, select `No title`. To use an object name as the title, select `Name` (the default). To include a custom title, select `Custom`.

If `Name` is selected, select one of the following options:

- `Object name`
- `Object name with Stateflow path`
- `Object name with Simulink and Stateflow path`

Header row

Use this option to select a header row for the table in the generated report. To include no header row, select `No header`. To include a header row with columns for name and object type, select `Type and Name`. When selected, this option creates a header row for the table with object name and type. To include a custom header, select `Custom`.

Don't display empty values

Select this option to exclude empty values from the generated report.

Insert Anything into Report?

Yes. Table.

File Name

`rptgen_sf.csf_auto_table`

Stateflow Count

Category

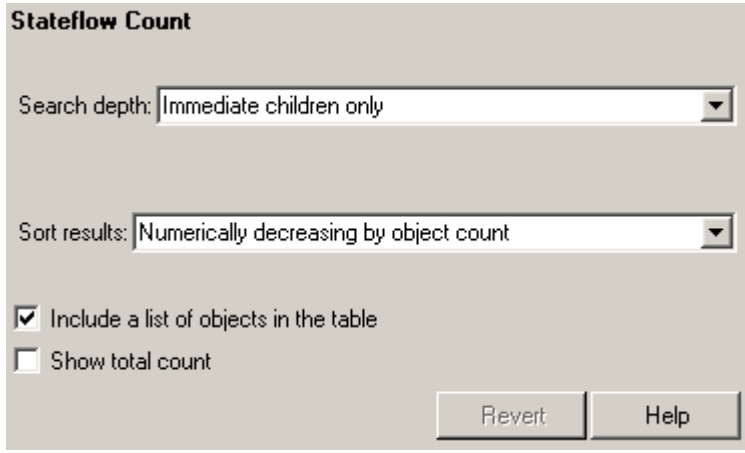
Stateflow

Description

The Stateflow Count component counts the number of Stateflow objects in the current context, such as system, model, signal, and so on. See the Stateflow documentation for a description of Stateflow objects.

Attributes

The following figure shows the **Stateflow Count** Properties pane.



Search depth

Choose the search depth for the count:

- `Immediate children only` — (Default) Searches only children one level below the Stateflow object.
- `All descendants` — Searches all children of the Stateflow object.

Sort results

Choose the sort method for the count results:

- `Numerically decreasing by object count` (Default)
- `Alphabetically increasing by object type`

Include a list of objects in table
Select this option to insert a column containing the counted objects.

Show total count
Select this option to show a total of all counted objects.

Insert Anything into Report?

Yes. Table.

File Name

`rptgen_sf.csf_count`

Stateflow Dialog Snapshot

Category

Stateflow

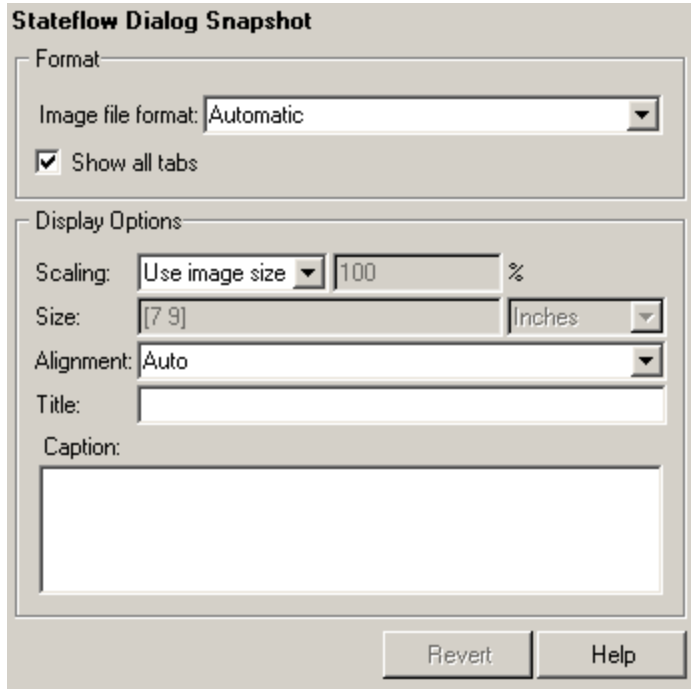
Description

The Stateflow Dialog Snapshot component takes snapshots of Stateflow editor dialog boxes. You can display the current settings associated with an object or document the appearance of your custom mask dialog boxes.

The Stateflow Dialog Snapshot component reports on the current reported Stateflow object. If the component is the child of a State Loop, for example, the current reported State's dialog box is documented.

Attributes

The following figure shows the **Stateflow Dialog Snapshot** Properties pane.



Format

Image file format

Choose the format for the snapshot image file. The **Automatic** format chooses BMP format for PDF files and PNG for other formats.

Show all tabs

Select this check box if you want to automatically generate images for all the tabs for the dialog box. If the check box is not selected, Report Generator creates an image of only the first tab.

Display Options

Scaling

Select `Fixed size` to specify the number and type of units.

Select `Zoom` to specify the percentage, the maximum size, and the units of measure.

Select `Use image size` if you want the image in your report to be the same size as the image on the screen.

Size

Enter the exact size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Fixed size` in the **Scaling** drop-down list.

Max size

Enter the maximum size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Zoom` in the **Scaling** drop-down list.

Units

Enter the units for the size of the snapshot. This field is active only if you choose `Zoom` or `Fixed size` in the **Image size** list box.

Alignment

Use this option to align your snapshot. Values are `Auto`, `Right`, `Center`, and `Left`.

Note This option is supported only in `PDF` and `RTF` reports.

Title
Type the text you want to appear above the snapshot.

Caption
Type the text you want to appear below the snapshot.

Insert Anything into Report?

Yes. Snapshot.

File Name

`rptgen_sl.Cdialog boxesnapshot`

Category

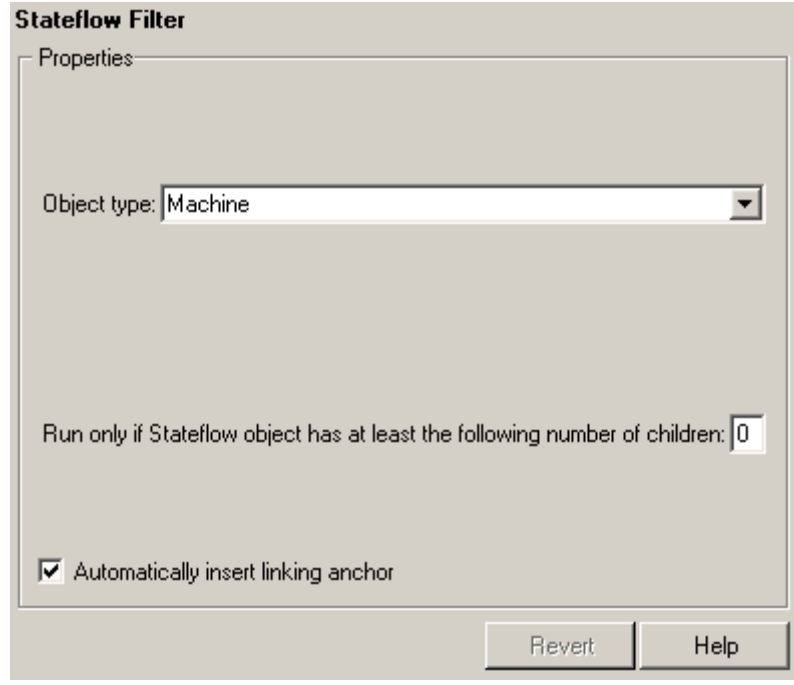
Stateflow

Description

The Stateflow Filter component is a filter that runs its children only if the current object type (set by the parent Stateflow Hierarchy Loop component) matches the selected object type. The Stateflow Filter component must have the Stateflow Loop component as its parent.

Attributes

The following figure shows the **Stateflow Filter** Properties pane.



Object type

Choose the Stateflow object type to include in the report. The current object type (set by the parent `State Loop`) must match this selected object type in order for the current object to be included in the report.

Run only if Stateflow object has at least the following number of Stateflow children

Use this option to limit the number of Stateflow objects that is reported on. If you enter a number in this field, a Stateflow object is not included in the report unless it has at least that number of children.

Automatically insert linking anchor

Select this option to insert a linking anchor before the reported object. If an anchor for this object already exists (inserted by the `Stateflow Linking Anchor` component), a second anchor is not created.

Insert Anything into Report?

No.

File Name

`rptgen_sf.csf_obj_filter`

Category

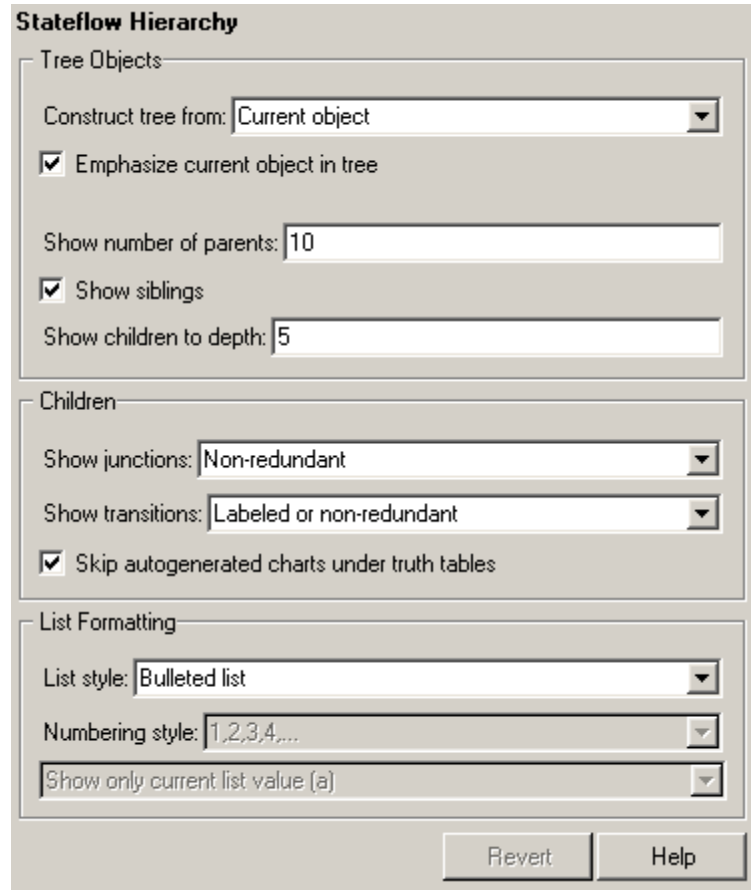
Stateflow

Description

The Stateflow Hierarchy component inserts a tree representation that describes the Stateflow object's hierarchy.

Attributes

The following figure shows the **Stateflow Hierarchy** Properties pane.



Tree Options

Construct tree from
Select the object to use for the tree representation:

- `Current object`
- `Root of current object` — Starts from the top of the hierarchy.

Emphasize current object in tree
Select this option to highlight the current object in the tree representation.

Show number of parents
Enter the number of parents to allow in the tree representation.

Show siblings
Select this option to display siblings in the tree representation.

Show children to depth
Enter the depth of children to display for each object in the tree representation.

Children

Show junctions
Select the level of junction detail to display in the generated report:

- `All`
- `Non-redundant`
- `None`

Show transitions
Select the level of transition detail to display in the generated report:

- `All`
- `Labeled or non-redundant`
- `Non-redundant`
- `Labeled`

- None

Skip autogenerated charts under truth tables
Select this option to not include autogenerated charts beneath truth tables.

List Formatting

List style
Select either Bulleted list or Numbered list. If you choose Numbered list, you can select numbering options in the **Numbering style** section.

Numbering style
Select a numbering style from the menu:

- 1,2,3,4...
- a,b,c,d...
- A,B,C,D...
- i,ii,iii,iv...
- I,II,III,IV...

To show the parent number in each list entry, select Show parent number in nested list (1.1.a). To show only the current number or letter, select Show only current list value (a).

Insert Anything into Report?

Yes. Tree.

File Name

rptgen_sf.csf_hier

Stateflow Hierarchy Loop

Category

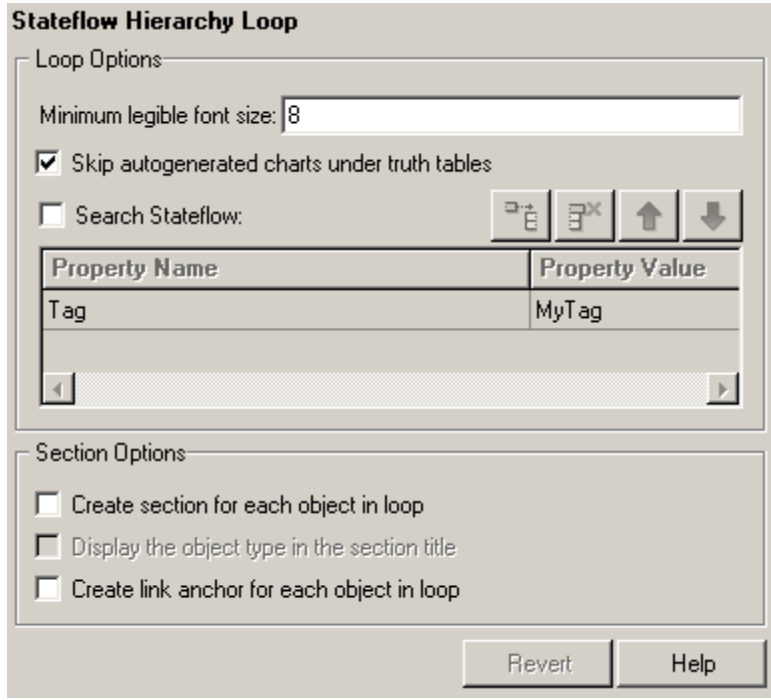
Stateflow

Description

The Stateflow Hierarchy Loop component runs its child components on the Stateflow object hierarchy.

Attributes

The following figure shows the **Stateflow Hierarchy Loop** Properties pane.



Loop Options

Minimum legible font size

Enter the minimum font size to use in the report. The default font size, `8`, is the smallest recommended font size.

Skip autogenerated charts under truth tables
Select this option to exclude autogenerated charts under truth tables in the report.

Search Stateflow
Select this option to report only on Stateflow charts with certain property name/property value pairs. Enter the property name and the property value in the appropriate fields.

Section Options

Create section for each object in loop
Select this option to insert a section in the generated report for each object found in the loop.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Create link anchor for each object in loop
Select this option to create a hyperlink to the object in the generated report.

Insert Anything into Report?

No.

File Name

`rptgen_sf.csf_hier_loop`

Stateflow Linking Anchor

Category

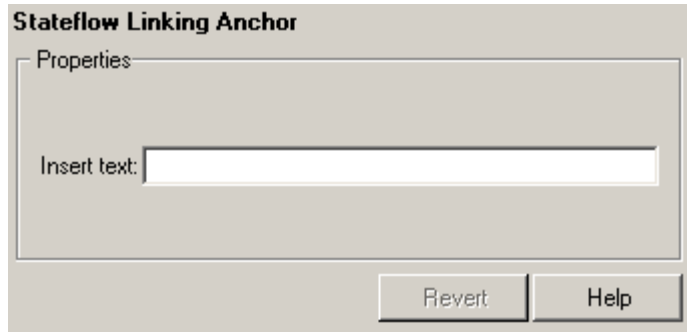
Stateflow

Description

The Stateflow Linking Anchor component designates a location to which other links point. The linking anchor is set to the current object, which is defined by the parent component. This component must have the Chart Loop, State Loop, Machine Loop, Terminal Loop, or Stateflow Filter component as its parent.

Attributes

The following figure shows the **Stateflow Linking Anchor** Properties pane.



Insert text

If you want text to appear after the linking anchor, enter the text in this field.

Note You should not use the Stateflow Linking Anchor component to create an anchor for an object that is anchored in the object summary table. See the Simulink Summary Table component reference page for information on how to do this.

Insert Anything into Report? Yes. A link, and possibly text, depending on attribute choices.

File Name `rptgen_sf.csf_obj_anchor`

Stateflow Name

Category

Stateflow

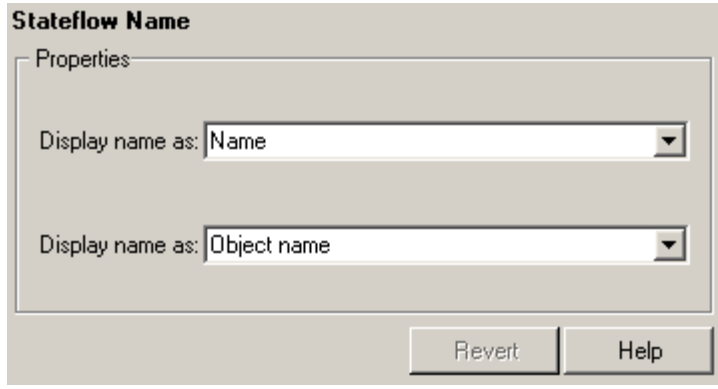
Description

The Stateflow Name component inserts the name of the Stateflow object specified by its parent component into the report. This component must have the State Loop, Chart Loop, or Stateflow Filter component as its parent.

You can use this component as the first child component of a Chapter/Subsection component, which allows the current Stateflow object name to be the chapter or section title.

Attributes

The following figure shows the **Stateflow Name** Properties pane.



Display name as

Select one of the options listed below to display the Stateflow object name in the report:

- `Name` — For example, `Object`
- `Type Name` — For example, `Object <ObjectName>`
- `Type - Name` — For example, `Object - <ObjectName>`
- `Type: Name` — For example, `Object: <ObjectName>`

Display name as
Select one of the options listed below to display the Stateflow object name in the report:

- `Object name`
- `Object name with Stateflow path`
- `Object name with Simulink and Stateflow path`

Insert Anything into Report?

Yes. Text.

File Name

`rptgen_sf.csf_obj_name`

Stateflow Property

Category

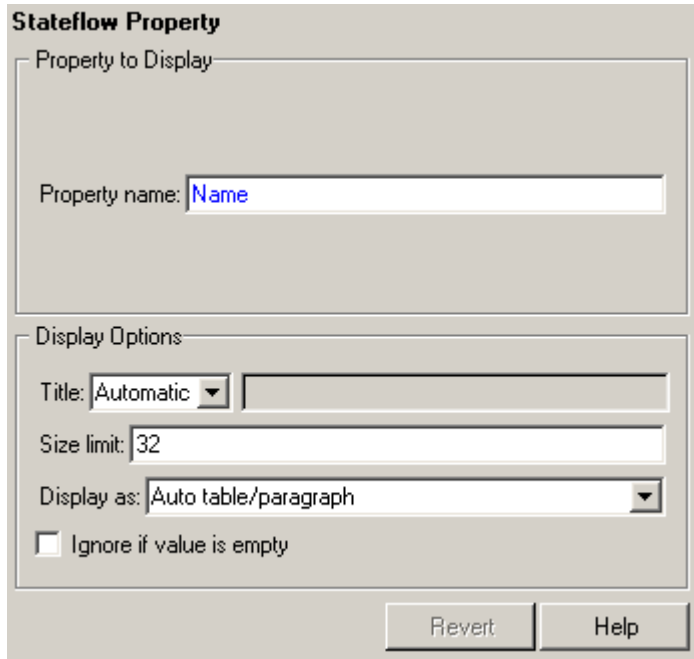
Stateflow

Description

The Stateflow Property component inserts a table, text, or paragraph that contains details of the selected Stateflow object property.

Attributes

The following figure shows the **Stateflow Property** Properties pane.



Property to Display

Property name

Enter the Stateflow property name to display.

Display Options

Title

Choose a title for the display in the generated report. To use the parameter name as the title, select `Automatic`. To enter a title, select `Custom`. To specify no title, select `None`.

Size limit
Use this size to limit the width of the display in the generated report. Units are in pixels. When creating a table, the size limit is the hypotenuse of the table width and height `sqrt(w^2+h^2)`. When creating text, the size limit is the number of characters squared. If the size limit is exceeded, the variable is rendered in condensed form, such as `[64x64 double]`. Setting a size limit of 0 will ignore the size limit and always display the variable, no matter how large.

Display as
Choose a display style from the menu:

- `Auto table/paragraph` (Default): Displays as a table or paragraph based on the information.
- `Table`: Displays as a table.
- `Paragraph`: Displays as a text paragraph.
- `Inline text`: Displays inline, which fits in line with the surrounding text.

Ignore if value is empty
Select this option to exclude empty parameters in the generated report.

Insert Anything into Report?

Yes. Text, paragraph, or table.

File Name

`rptgen_sf.csf_property`

Stateflow Property Table

Category

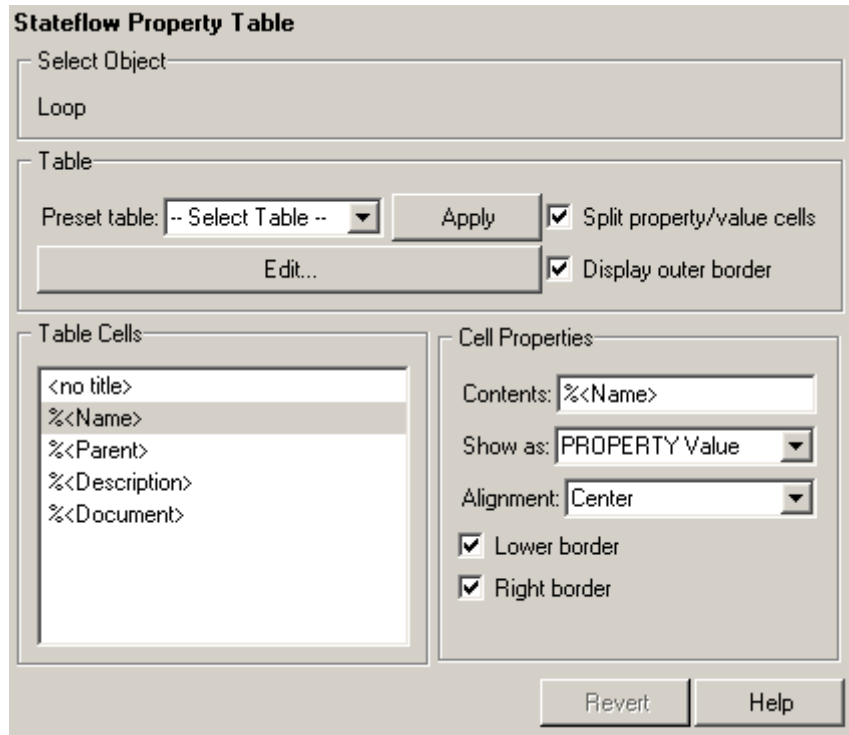
Stateflow

Description

The Stateflow Property Table component inserts a property-value table for a Stateflow object into the report. This object is defined by the `Stateflow Filter` component, which must be the parent of the Stateflow Property Table. For details about working with property table components, see “Working with Property Table Components” on page 7-14.

Attributes

The following figure shows the **Stateflow Property Table** Properties pane.



Table

A preset table is a table that is already formatted and set up. You can select a preset table in the preset table list in the upper-left corner of the attributes page. To apply a preset table, select the table and click **Apply**.

Preset table

Choose a type of table to display the object property table:

- `Default`
- `Machine`
- `Chart`
- `State`
- `Truth table`
- `EM function`
- `Data`
- `Event`
- `Junction`

Split property/value cells

Select this option to split property name/property value pairs into separate cells. To have the property name and property value appear in adjacent horizontal cells in the table, select the **Split property/value cells** check box. In this case, the table is in split mode and there can be only one property name/property value pair in a cell. If you have more than one name/property pair in a cell, only the first pair appears in the report. All subsequent pairs are ignored.

To have the property name and property value appear together in one cell, clear the **Split property/value cells** check box. This is nonsplit mode. In nonsplit mode, you can have more than one property name/property value pair and you can also have text.

If you want to switch from nonsplit mode to split mode, make sure that you have only one property name/property value pair per table cell before you switch modes. If you have more than one property name/property value pair or any text, only the first property name/property value pair appears in the report; subsequent pairs and text are omitted.

Display outer border
Select this option to display the outer border of the table in the generated report.

Table Cells
Use this field to select table properties to modify. The selection in this pane affects the available fields in the **Cell Properties** pane.

Cell Properties

The visible options in the **Title Properties** pane depend on the object selected in the **Table Cells** pane. If `%<Name>` is selected, only the **Contents** and **Show** options appears. If any other object is selected in the **Table Cells** pane, the **Lower border** and **Right border** options appears.

Contents
Use this field to modify the contents of the table cell selected in the **Table Cells** pane.

Choose the of the contents of the selected table cell in the **Table Cells** pane.

- `Left`
- `Center`
- `Right`
- `Double justified`

Show as
Choose the format for the contents of the table cell:

- `Value`
- `Property Value`

- `PROPERTY Value`
- `Property:  Value`
- `PROPERTY: Value`
- `Property - Value`
- `PROPERTY - Value`

Alignment
The default is `Center`. Other options are `Left`, `Right`, and `Double justified`.

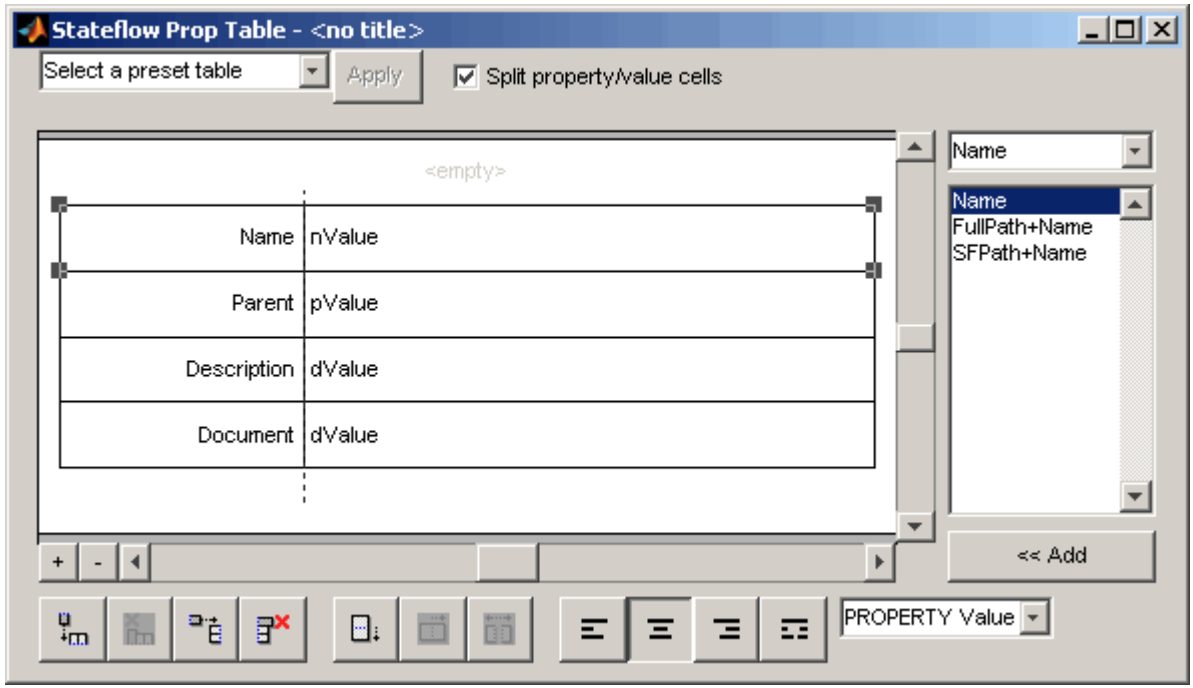
Lower border
Select this option to display the lower border of the table in the generated report.

Right border
Select this option to display the right border of the table in the generated report.

Creating Custom Tables

To create a custom table, you can first select a preset table and then edit it. You might want to start with the `Blank 4x4` preset table. You can add and delete rows and you can add properties. To open the Edit Table dialog box, click **Edit**. The dialog box appears.

Stateflow Property Table



For details about using this dialog box to create custom property tables, see “Working with Property Table Components” on page 7-14.

Note that available properties and preset tables depend on the Stateflow Filter parent component.

Insert Anything into Report?

Yes. Table.

File Name

`rptgen_sf.csf_prop_table`

Category

Stateflow

Description

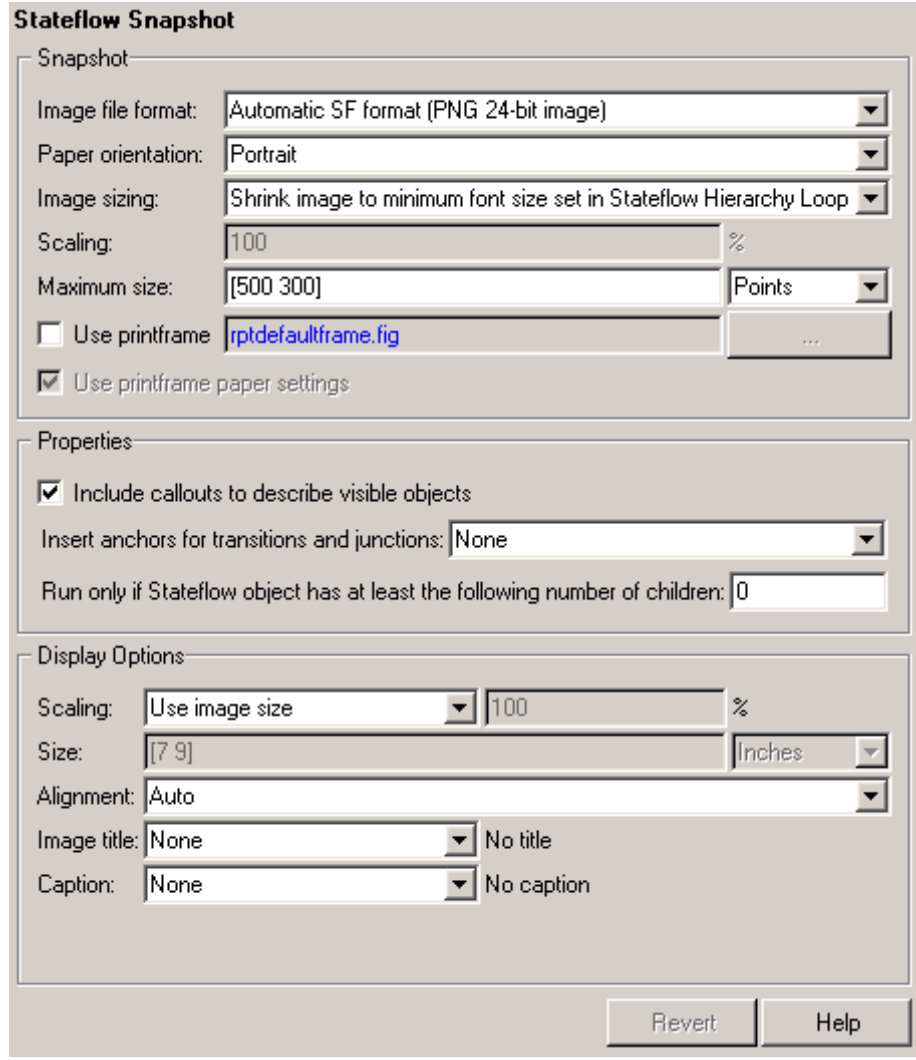
The Stateflow Snapshot component inserts a snapshot of a Stateflow object (defined by the Stateflow Filter component) into the report. The Stateflow Snapshot component must have the Stateflow Filter component as its parent.

This component only executes if the selected object in the Stateflow Filter component is a graphical object. The following are graphical Stateflow objects:

- Chart
- State
- Transition
- Frame

Attributes

The following figure shows the **Stateflow Snapshot** Properties pane.



Snapshot

Image file format

Choose the image file format (e.g., JPEG, TIFF, etc.) from this list. Select `Automatic SF Format` to automatically choose the format

best suited for the output format that you chose in the Report component. Otherwise, choose an image format that your output viewer can read. `Automatic SF Format` is the default option. The options include:

- `Automatic SF Format` (uses the Handle Graphics file format selected in the Preferences dialog box)
- `Adobe Illustrator`
- `Bitmap (16m-color)`
- `Bitmap (256-color)`
- `Black and white encapsulated PostScript`
- `Black and white encapsulated PostScript (TIFF)`
- `Black and white encapsulated PostScript2`
- `Black and white encapsulated PostScript2 (TIFF)`
- `Black and white PostScript`
- `Black and white PostScript2`
- `Color encapsulated PostScript`
- `Color encapsulated PostScript (TIFF)`
- `Color encapsulated PostScript2`
- `Color encapsulated PostScript2 (TIFF)`
- `Color PostScript`
- `Color PostScript2`
- `JPEG high quality image`
- `JPEG medium quality image`
- `JPEG low quality image`
- `PNG 24-bit image`
- `Scalable vector graphics (SVG)`

- `TIFF - compressed`
- `TIFF - uncompressed`
- `Windows metafile`

Paper orientation
Options include `Portrait`, `Landscape`, `Rotated`. Choose `Largest dimension vertical` if you want to position the image so that its largest dimension is vertical. Choose `Use Chart PaperOrientation setting` to use the paper orientation setting you have specified for your chart.

Image sizing
`Shrink image to minimum font size specified in Stateflow Hierarchy Loop` resizes the image so that the text label font size is the minimum font size. `Fixed` and `Zoom` enable you to specify the size of your image.

Scaling
Specify a percentage of the image's size to which to scale it.

Maximum size
Enter the maximum size for the snapshot in the generated report in the selected units. Use `[width height]` format. In the units text box, select `Inches`, `Centimeters`, `Points`, or `Normalized`.

Use printframe
Select this check box to insert a frame around your image. You can use the default frame or create a custom one.

Use printframe paper settings
Check this box to use the dimensions and parameters specified by your printframe to size your image. If you choose this option, all other options (with the exception of Image file format) are grayed out.

Properties

Include callouts to describe visible objects
Select this option to display descriptive callouts for visible objects.

Insert anchors for transitions and junctions

Select this option to insert anchors for transitions and junctions into the report:

- `None`
- `Redundant children only`
- `All`

Run only if Stateflow object has at least the following number of children

To run this component only when the current Stateflow object has at least a certain number of children, enter that number in this field. The current Stateflow object is specified by the Stateflow Filter component. This option is not active unless the selected object in the Stateflow Filter (the parent component) is a graphical object.

This option lets you decrease the size of the report by excluding certain images. You might want to use this option when you have a large model.

Display Options

Scaling

`Use image size` uses the image size that you specify in the snapshot option. `Zoom` and `Fixed size` allow you to specify the size of the image.

Size

Specify a size in inches for your image. The default is 7 inches by 9 inches.

Max size

Enter the maximum size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Zoom` from the **Scaling** drop-down list.

Units

Enter the units for the size of the snapshot. This field is active only if you choose `Zoom` or `Fixed size` in the **Image size** list box.

Alignment
The default is `Auto`, which automatically aligns your image. You can also choose `Right`, `Center`, or `Left` to align your image accordingly.

> **Note** This option is supported only in `PDF` and `RTF` reports.

Image title
The default is `None`. Choose `Object name` to use the object name as the title, `Full Stateflow name` to specify the stateflow path and the name of the object, `Full Simulink + Stateflow name` to specify the Simulink path and name of the object, or `Custom` to enter a different title.

Caption
Use this option to specify a caption for your image. The default is `None`. Set it to `Custom` to enter a caption, or to `Description` to set the caption to the value of the object Description property.

Insert Anything into Report?

Yes. Image.

File Name

`rptgen_sf.csf_obj_snap`

Category

Stateflow

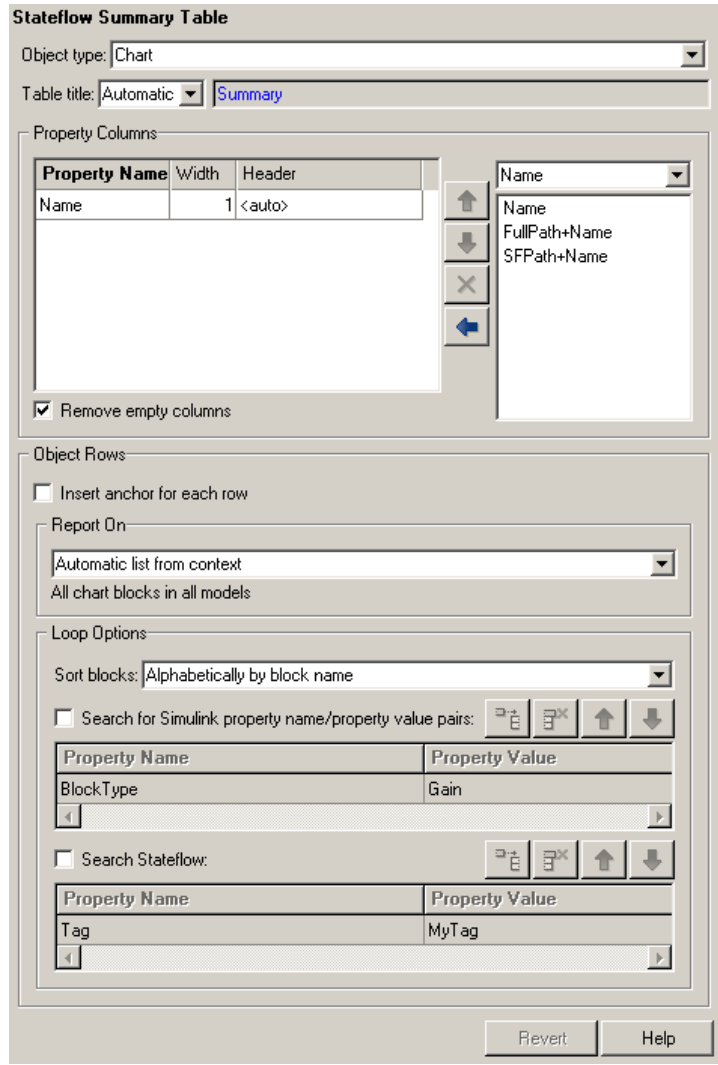
Description

The Stateflow Summary Table component shows properties or parameters of the selected Stateflow objects in a table. The Stateflow Summary Table component can have the following parents:

- Any Stateflow looping component, except Terminal Loop
- Any Simulink looping component (Model Loop, System Loop, Block Loop, or Signal Loop)

Attributes

The following figure shows the **Stateflow Summary Table** Properties pane.


Stateflow Summary Table
Object type: Chart
Table title: Automatic
Summary
Property Columns
Property Name
Width
Header
Name
1
<auto>
Name
Name
FullPath+Name
SFPath+Name
Remove empty columns
Object Rows
Insert anchor for each row
Report On
Automatic list from context
All chart blocks in all models
Loop Options
Sort blocks: Alphabetically by block name
Search for Simulink property name/property value pairs:
Property Name
Property Value
BlockType
Gain
Search Stateflow:
Property Name
Property Value
Tag
MyTag
Revert
Help

Object type

Choose the object type to display in the generated report. The menu item selected in this menu affects the options available in the **Property Columns** pane.

Table title

Choose a title for the summary table in the generated report:

- `Automatic` — Select this option to generate a title automatically from the parameter.
- `Custom` — Select this option to enter a title manually.

Property Columns

Property columns

This figure displays the object properties to be included in the summary table in the generated report. To add a property, select the appropriate property level in the text box. Next, in the contextual list below the text box, select the property that you want to add and click **Add**. To delete a property, select the property name and press the **Delete** key.

Note Some entries in the list of available properties (such as `Depth`) are "virtual" properties which cannot be accessed using the `get_param` command. The properties used for property/value filtering in the block and System Loop components must be retrievable by the `get_param`. Therefore, you cannot configure your summary table to report on all blocks of `Depth == 2`.

Remove empty columns

Use this option to remove empty columns from the summary table in the generated report.

Object Rows

Insert anchor for each row

Select this option to insert an anchor for each row in the summary table.

Report On

Automatic list from context

Select this option to report on all blocks in the current context. The context is set by the Stateflow Summary Table component's parent component.

Custom - use block list

Select this option to report on a list of blocks that you specify and enter the block names in the corresponding field. You must use the full path name when specifying blocks. If you want to sort the blocks in this list, select the `Sort blocks in list` option and then select a sorting option from the **Sort blocks** list (in the **Loop Options** section). Blocks are reported on in the order that they are sorted. If you do not select this option, blocks are reported on in the order that they are listed in the block list.

You can enter `%<VariableName>` if you want to insert the value of a variable from the MATLAB workspace. The %<> notation can be a string or cell array. For more details about this notation, see "%<VariableName> Notation" on page 11-269 on the Text component reference page.

For example, if you want to report on the `theta dot` integrator block and the `theta` integrator block in the demo model `simppend`, you could enter the following in the block list:

```
simppend/theta dot

%<Z>
```

If you define `Z` in the workspace as the following,

```
Z={'simppend/theta'}
```

Report Generator reports on the following blocks:

```
simppend/theta dot
```

```
simppend/theta
```

Loop Options

You can choose block sorting options and reporting options in this pane.

Sort blocks

Use this option to select how blocks are sorted:

- Select `Alphabetically by block name` to sort blocks alphabetically by their names.
- Select `Alphabetically by system name` to sort systems alphabetically.

  In this option, the systems are sorted alphabetically, and blocks in each system are listed, but in no particular order.
- Select `Alphabetically by full Simulink path` to sort blocks alphabetically by Simulink path.
- Select `By block type` to sort blocks alphabetically by block type.
- Select `By block depth` to sort blocks by their depth in the model.
- Select `By traversal order` to sort blocks by traversal order.
- Select `By simulation order` to sort blocks by execution order.

Search for Simulink property name/property value pairs

Select this option to report only on Simulink blocks with certain property name/property value pairs. Enter the property name and the property value in the appropriate fields.

Search Stateflow

Select this option to report only on Stateflow charts with certain property name/property value pairs. Enter the property name and the property value in the appropriate fields.

Stateflow Summary Table

Insert Anything into Report?	Yes. Table.
File Name	`rptgen_sf.csf_summ_table`

Category

Report Generator

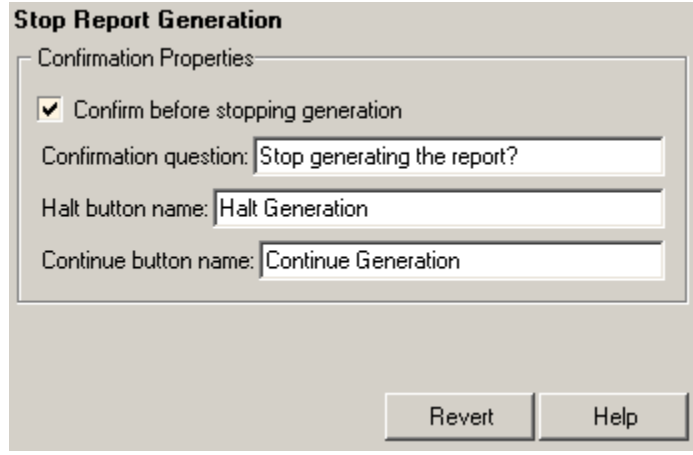
Description

The Stop Report Generation component halts report generation. This component acts like **Stop** during report generation.

You can use this component inside an `if/then` statement by using Logical and Flow Control components to halt the report generation process under a certain condition. Note that when report generation is halted, the XML source file is produced, but not converted.

Attributes

The following figure shows the **Stop Report Generation** Properties pane.



Confirmation Properties

Confirm before stopping generation

Select this option to generate a confirmation dialog box before stopping report generation.

Confirmation question

Enter a confirmation question for the prompt. The default is “Stop generating the report?”.

Halt button name
Enter a name for the button that stops report generation. The default is “Halt Generation”.

Continue button name
Enter a name for the button that continues report generation. The default is “Continue Generation”.

Example

The report template in this example creates a simple report that takes a snapshot of the current figure. If there is no current figure, the report generation automatically halts.

```
[-] Report - figure-report.rpt
 [-] if (isempty(get(0,'CurrentFigure')))
  [ ] Stop Generation
 [-] Figure Loop - current
  [-] Chapter - <Title from SubComponent1>
   [ ] Figure Name
   [ ] Graphics Figure Snapshot
   [ ] Figure Prop Table - Figure Properties
```

Insert Anything into Report?

No.

File Name

`rptgen.crg_halt_gen`

Category

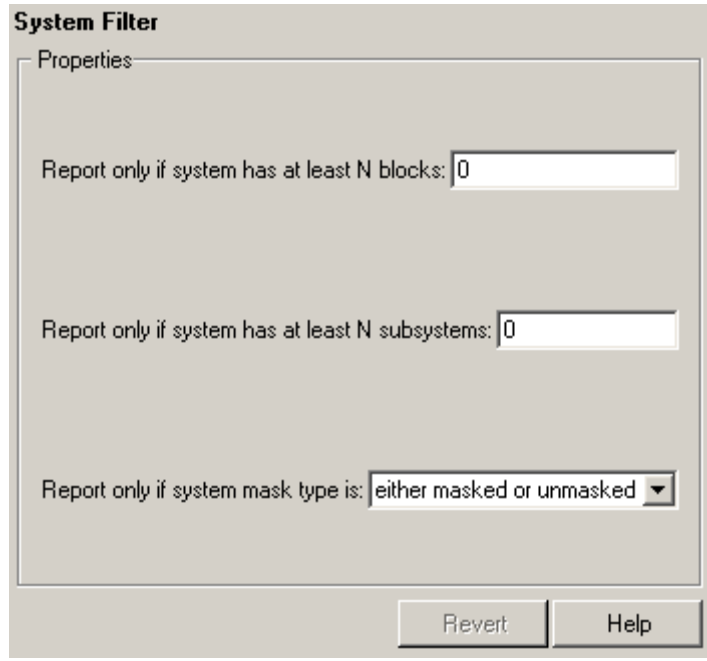
Simulink

Description

The System Filter component runs child components only if the current systems meets the conditions specified in the attribute page.

Attributes

The following figure shows the **System Filter** Properties pane.



Report only if system has at least N blocks

Enter a number to execute child components only if there are at least the specified number of blocks in the system. Note that if you enter 0, there is no filtering; child components are executed regardless of the number of blocks in the system.

Report only if system has at least N subsystems

Enter a number to execute child components only if there are at least the specified number of subsystems in the system. Note that

if you enter 0, there is no filtering; child components are executed regardless of the number of subsystems in the system.

Report only if system mask type is
Choose what masks to include in the generated report:

- `Either masked or unmasked`
- `Masked`
- `Unmasked`

Insert Anything into Report?

No.

File Name

`rptgen_sf.csf_obj_filter`

Category

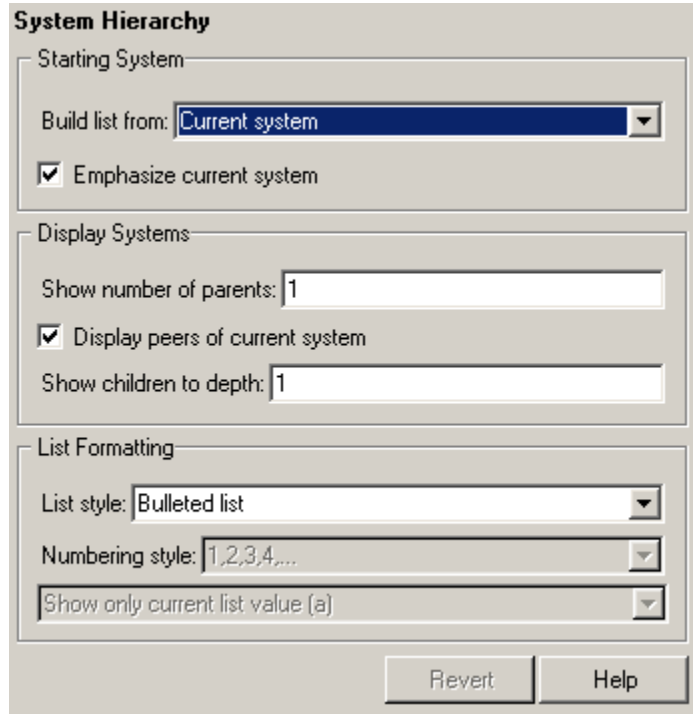
Simulink

Description

The System Hierarchy component creates a nested list that shows the hierarchy of the system specified in the attribute page. The list can display all systems in a model or it can show the parents and children of the current system.

Attributes

The following figure shows the **System Hierarchy** Properties pane.



Starting System

Build list from

Select the system or model from which the list will be built:

- `Current system`

- `Current model`

Emphasize current system
Select this option to highlight the current system or model in the generated report.

Display Systems

Show number of parents
Enter the number of parents to list.

Display peers of current system
Select this option to show the current system's peers in the generated report.

Show children to depth
Enter the depth of children to list.

List Formatting

List style
Select either `Bulleted list` or `Numbered list`. If you choose `Numbered list`, you can select numbering options in the **Numbering style** section.

Numbering style
Select a numbering style in this drop-down menu if you selected `Numbered List` for the **List style**. The options are:

- `1,2,3,4,...`
- `a,b,c,d,...`
- `A,B,C,D,...`
- `i,ii,iii,iv,...`
- `I,II,III,IV,...`

Insert Anything into Report?

Yes. List.

File Name `rptgen_sl.csl_sys_list`

System Loop

Category

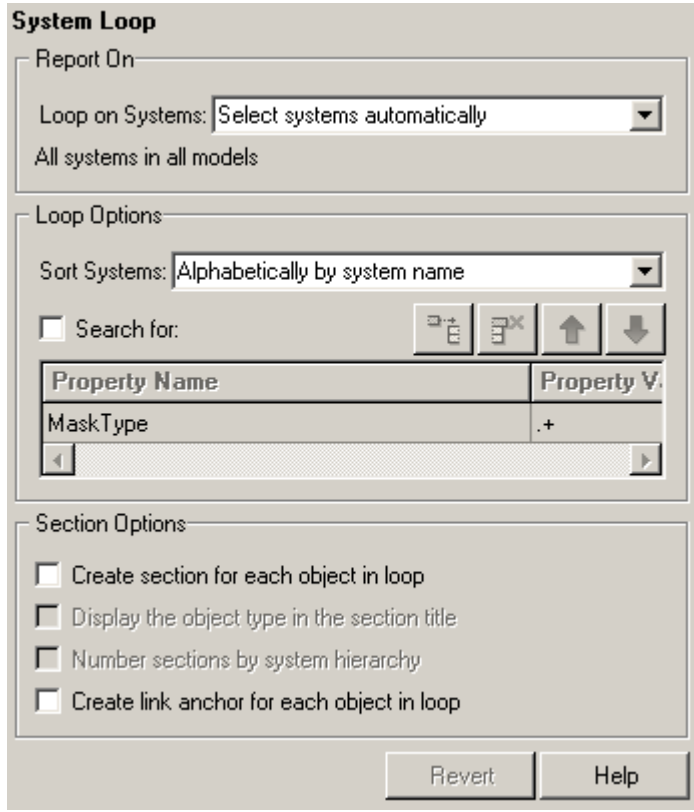
Simulink

Description

The System Loop component runs its child components for each system defined by the Model Loop. The System Loop component must have the Model Loop component as its parent.

Attributes

The following figure shows the **System Loop** Properties pane.



Report On

Loop on Systems

- **Select systems automatically**

Select the `Select systems automatically` option to report on all systems in the current context. The context is set by the System Loop component's parent component:

- If the parent component is the Model Loop, selecting this option causes the System Loop component to report on systems in the current model.
- If the parent component is another System Loop, selecting this option causes the System Loop component to report on the current system.
- If the parent component is the Signal Loop, selecting this option causes the System Loop component to report on the current signal's parent system.
- If the parent component is the Block Loop, selecting this option causes the System Loop component to report on the current block's parent system.
- If the System Loop does not have the Model Loop, System Loop, Signal Loop, or Block Loop component as its parent, selecting this option causes the System Loop component to report on all systems in all models.

Custom - use system list

Select this option to report on a list of systems that you specify, and enter the system names in the corresponding field. You must use the full path name when specifying systems. If you want to sort the systems in this list, select the **Sort systems in list** option and then select a sorting option from the **Sort Systems** option (in the **Loop Options** section). Systems are reported on in the order that they are sorted. If you do not select this option, systems are reported on in the order that they are listed in the block list.

You can enter `%<VariableName>` if you want to insert the value of a variable from the MATLAB workspace. The `%<>` notation can be a string or cell array. For more details about this notation, see "%<VariableName> Notation" on page 11-269 on the Text component reference page.

For example, if you want to report on the `Controller` system and the `Nz pilot calculation` system in the demo model `f14`, you could enter the following in the block list:

```
f14/Nz pilot calculation

%<Z>
```

If you define `Z` in the workspace as the following:

```
Z={'f14/Controller'}
```

Report Generator reports on the following systems:

```
f14/Nz pilot calculation

f14/Controller
```

Loop Options

Sort Systems

Use this section to select how the systems are sorted:

- Select `Alphabetically by system name` (default) to sort systems alphabetically by name.
- Select `By number of blocks in system` to sort systems by the number of blocks in the system. The list shows systems by decreasing number of blocks; i.e., the system with the largest number of blocks is listed first.
- Select `By system depth` to sort systems by their depth in the model.
- Select `By traversal order` to sort systems in the order they are traversed.

Search for

Select this option to report only on Simulink blocks with certain property name/property value pairs. Enter the property name and the property value in the appropriate fields.

Section Options

Create section for each object in loop
Select this option to insert a section in the generated report for each object found in the loop.

Display the object type in the section title
Select this option to insert the object type automatically into the section title in the generated report.

Number sections by system hierarchy
Select this option to number sections in the generated report hierarchically.

Create link anchor for each object in loop
Select this option to create a hyperlink to the object in the generated report.

Insert Anything into Report?

Yes, inserts a section if the **Create section for each object in loop** option is selected.

File Name

`rptgen_sl.csl_sys_loop`

System Snapshot

Category

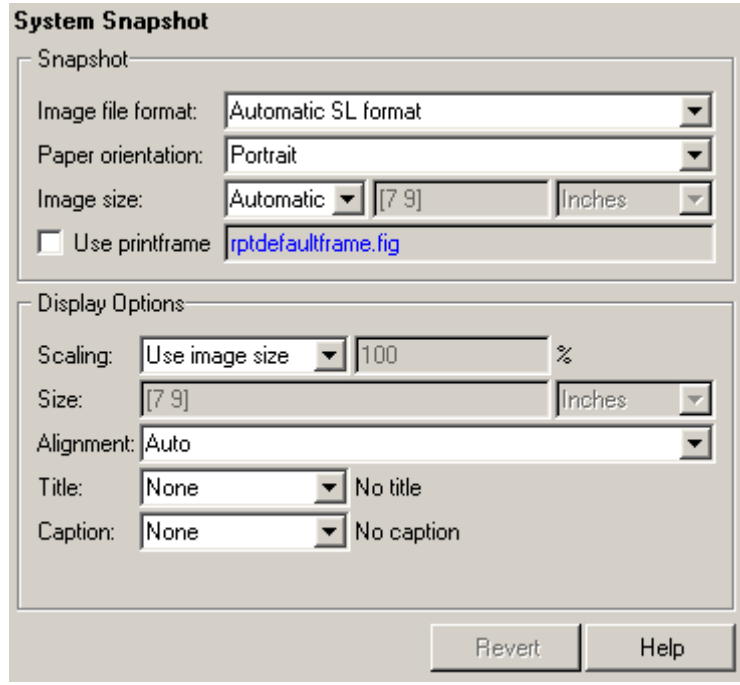
Simulink

Description

The System Snapshot component inserts a snapshot of the current system in your report.

Attributes

The following figure shows the **System Snapshot** Properties pane.



Snapshot

Image file format

Choose the image file format (e.g., JPEG, TIFF, etc.) from this list. Select `Automatic SL Format` to automatically choose the format best suited for the output format that you chose in the Report component. Otherwise, choose an image format that your output viewer can read. `Automatic SL Format` is the default option. The options include:

- `Automatic SL Format` (uses the Handle Graphics file format selected in the Preferences dialog box)
- `Adobe Illustrator`
- `Bitmap (16m-color)`
- `Bitmap (256-color)`
- `Black and white encapsulated PostScript`
- `Black and white encapsulated PostScript (TIFF)`
- `Black and white encapsulated PostScript2`
- `Black and white encapsulated PostScript2 (TIFF)`
- `Black and white PostScript`
- `Black and white PostScript2`
- `Color encapsulated PostScript`
- `Color encapsulated PostScript (TIFF)`
- `Color encapsulated PostScript2`
- `Color encapsulated PostScript2 (TIFF)`
- `Color PostScript`
- `Color PostScript2`
- `JPEG high quality image`
- `JPEG medium quality image`
- `JPEG low quality image`
- `PNG 24-bit image`
- `Scalable vector graphics (SVG)`
- `TIFF - compressed`
- `TIFF - uncompressed`
- `Windows metafile`

Paper orientation

You can choose `Largest dimension vertical`, `Landscape`, `Portrait`, or `Use system orientation` as the paper orientation.

The `Use system orientation` option uses the `PaperOrientation` parameter in Simulink, which you can set as `Landscape`, `Portrait`, or `Rotated`.

Image Size

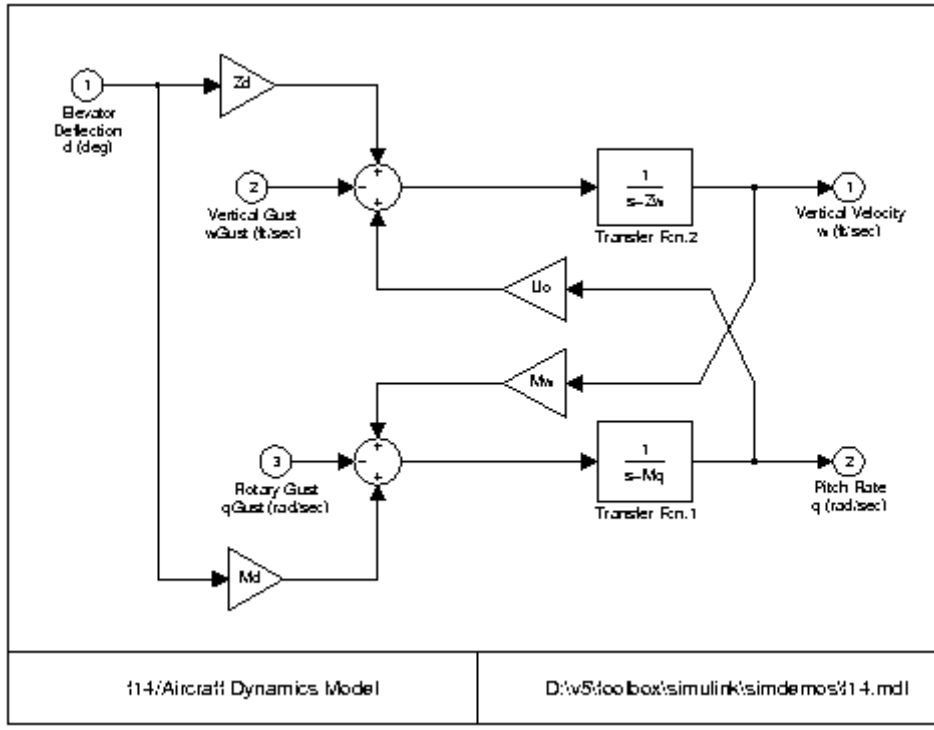
Select `Automatic` to automatically scale the image to output dimensions. Select `Custom` to specify image size. The default for this field is `Automatic`. Note that if `Use printframe` is selected, the `Custom` option is deactivated and the image is automatically scaled to the print frame size.

Use printframe

Choose this option to print a frame around the image. You can use the default Report Generator frame, `rptdefaultframe.fig` , or you can use the Frame Editor to build a custom frame.

To build a custom frame, type `frameedit` at the command line to invoke the Print Frame Editor. For help with the Print Frame Editor, select `Print Frame Editor Help` from the **Help** menu. See `frameedit` in the MATLAB reference pages for details.

The default Report Generator frame is five inches wide and four inches high. It includes the name of the system and the model directory. This frame is optimized for use with a `portrait` paper orientation. The Aircraft Dynamics Model in the `f14` Simulink demo model is shown below with the default Report Generator frame option.



Display Options

Scaling

Select `Fixed size` to specify the number and type of units.

Select `Zoom` to specify the percentage, the maximum size, and the units of measure.

Select `Use image size` if you want the image in your report to be the same size as the image on the screen.

Size

Enter the exact size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Fixed size` in the **Scaling** drop-down list.

Max size

Enter the maximum size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Zoom` in the **Scaling** drop-down list.

Units

Enter the units for the size of the snapshot. This field is active only if you choose `Zoom` or `Fixed size` in the **Image size** list box.

Values are `Auto`, `Right`, `Left`, and `Center`.

Note This option is supported only in `PDF` and `RTF` reports.

Image title

Choose `No title`, `System name`, `System name (full path)`, or `Custom` to specify a title for the image. The default is no title. If you choose `Custom`, you need to type in the title you want to appear.

Caption

Choose `None`, `Description (use system description)`, or `Custom` to specify a caption for the image. The default is `None`. If you choose `Custom`, you need to type in the caption you want to appear.

Insert Anything into Report?

Yes. Image.

File Name

`rptgen_sl.csl_sys_snap`

Category

Formatting

Description

The Table component converts a rectangular cell array into a table and inserts the table into the report.

Attributes

The following figure shows the **Table** Properties pane.

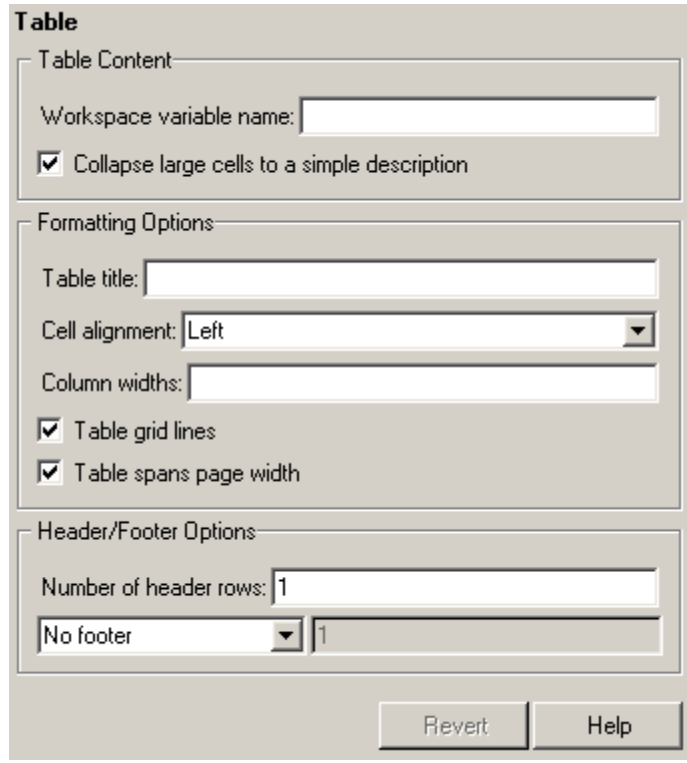


Table Content

Workspace variable name

Enter the workspace variable name with which to construct the table.

Collapse large cells to a single description
Select this option to consolidate large cells into one description.

Formatting Options

Table title
Enter the title of your table. If you have a title for a table, it is included in the list of tables if the stylesheet you chose in the Report Options attribute page supports it. To find stylesheets that have a list of tables, see "Setting the Report Format" on page 5-3.

Cell alignment
Select a cell alignment option (`left`, `center`, `right`, or `double justified`).

Column widths
Input a vector with *m* elements, where *m* equals the number of columns in the table. Column sizing is relative and normalized to page width. For example, say you have a 2-by-3 cell array and input the following into the **Column widths** field:

```
[1 2 3]
```

The report output format for the cell array would be such that the second column is twice the width of the first column and the third column is three times the width of the first column. Note that if the vector is greater than the number of columns in the table, the vector is truncated so that the number of elements equals the number of columns. If *m* is less than the number of columns in the table, the vector is padded with 1s so that the number of elements equals the number of columns.

Table grid lines
Select this option if you want to have grid lines in your table. Grid lines create borders between the fields.

Table spans page width (HTML only)
Select this option if you want the table to have the same width as the page on which it appears.

Header/Footer Options

Designating a row as a header or footer row causes the contents of the row to be displayed in boldface.

Number of header rows

Enter the number of header rows (from the top): 0 = none, 1 = 1 row from the top, 2 = two rows from the top, etc.

Footer list

If you do not want a footer, select `No footer`. If you want to select a footer that is different from your header, select `Last N rows are footer` and enter the number of footer rows (from the bottom) in the corresponding field: 0 = none, 1 = one row from the bottom, 2 = two rows from the bottom, etc.

Example

For example, if you have the following cell array in the MATLAB workspace,

```
{'foo','bar';[3],[5]}
```

the resulting cell table has the following appearance.

foo	bar
3	5

Note that the table has no headers or footers and no title. The table does have grid lines.

Insert Anything into Report?

Yes. Table.

File Name

`rptgen.cfr_table`

Text

Category

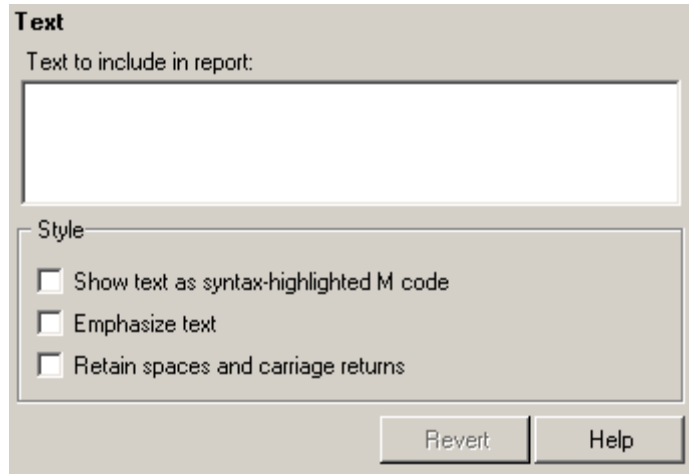
Formatting

Description

The Text component formats text and inserts it into your report. The text component should have the Paragraph component as its parent.

Attributes

The following figure shows the **Text** Properties pane.



Text to include in report
Enter the text you want to include in your report in this field.

Show text as syntax-highlighted M code
Select this option to display text as highlighted M-code.

Emphasize text
If you want to have the text appear italicized in the report, select this option.

Retain spaces and carriage returns
If you want to keep spaces and carriage returns in your text, select this option. If you do not select this option, the text still inserts one space any time you have one or more spaces; it does not keep any carriage returns.

%<VariableName> Notation

You can enter `%<VariableName>` in this field (and in any field where the text appears blue) to include the value of a variable from the base MATLAB workspace. You cannot enter more than one variable in the `%<>`. If you enter an invalid variable name, the report will contain `%<VariableName>` instead of the value of the variable.

Example 1

If you enter the following:

```
I have a %<ObjName> and it has %<NumLeaves> leaves. The word
'%<ObjName>' has %<size(ObjName)> letters.
```

and if `ObjName = "plant"` and `NumLeaves = 3` in the MATLAB workspace, the report contains the following:

```
I have a plant and it has 3 leaves.
The word 'plant' has 5 letters.
```

Example 2

If you enter the following:

```
%<help('peaks')>
```

the report contains the text that you get when you type `help peaks` at the command line.

Insert Anything into Report?

Yes. Text.

File Name

`rptgen.cfr_text`

Time/Date Stamp

Category

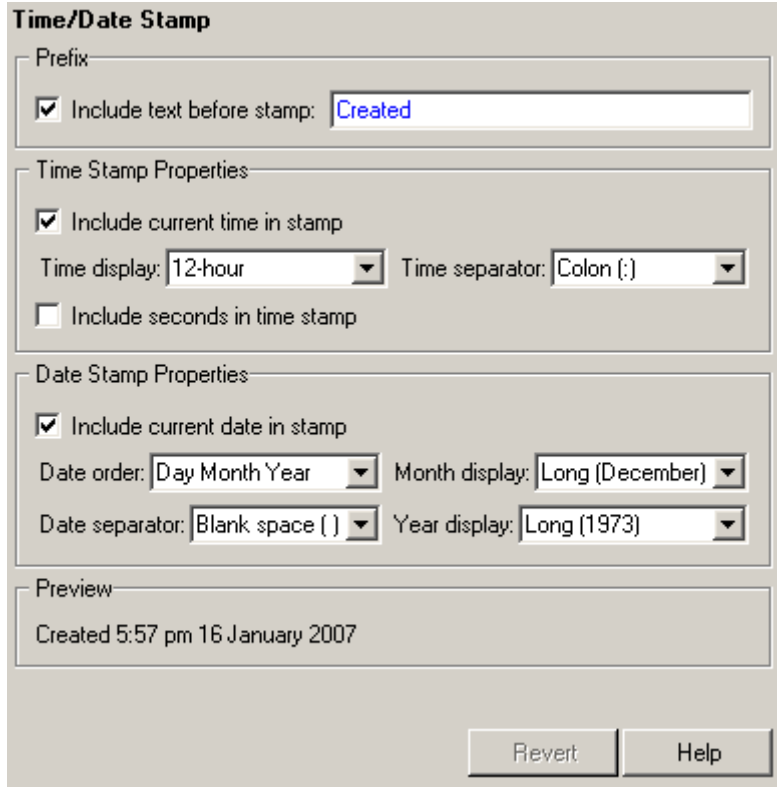
Report Generator

Description

The Time/Date Stamp component inserts the time and date of the report generation into your report as text. It must have the Paragraph or Chapter/Subsection component as its parent.

Attributes

The following figure shows the **Time/Date Stamp** Properties pane.



Prefix

Include text before stamp

Select this option to include text before the time/date stamp, and enter the text in the corresponding field.

Time Stamp Properties

Include current time in stamp
Select this option to insert the current time in the time/date stamp.

Time display
Use this option to select the appearance of the time display. The following options are available:

- `12-hour`
- `24-hour`

Time Separator
Use this option to select a separation marker between hours, minutes, and seconds. The following options are available for the separator:

- `Blank space ( )` - For example, `Hour Minute Second`
- `Colon (:)` - For example, `Hour:Minute:Second`
- `Period (.)` - For example, `Hour.Minute.Second`
- `None ()` - For example, `HourMinuteSecond`

Include seconds in time stamp
Select this option to display seconds in the time/date stamp.

Date Stamp Properties

Include current date in stamp
Select this option to insert the current date in the time/date stamp.

Date order
Use this option to select the order in which the day, month, and year should appear. The following options are available for this order:

- `Day Month Year`
- `Month Day Year`
- `Year Month Day`

Date separator
Use this option to select a separation marker between day, month, and year. The following options are available for the separator:

- Blank space ( ) — For example, `Day Month Year`
- Colon (:) — For example, `Day:Month:Year`
- Slash (/) — For example, `Day/Month/Year`
- Period (.) — For example, `Day.Month.Year`
- None () — For example, `DayMonthYear`

Month display
Use this option to select the appearance of the month display. The following options are available for the display:

- `Long (December)`
- `Short (Dec)`
- `Numeric (12)`

Year display
Use this option to select the appearance of the year display. The following options are available for the display:

- `Long (2007)`
- `Short (07)`

Preview

This pane displays the time/date stamp as it will appear in the report.

Insert Anything into Report?

Yes. Text.

File Name

`rptgen.crg_tds`

Category

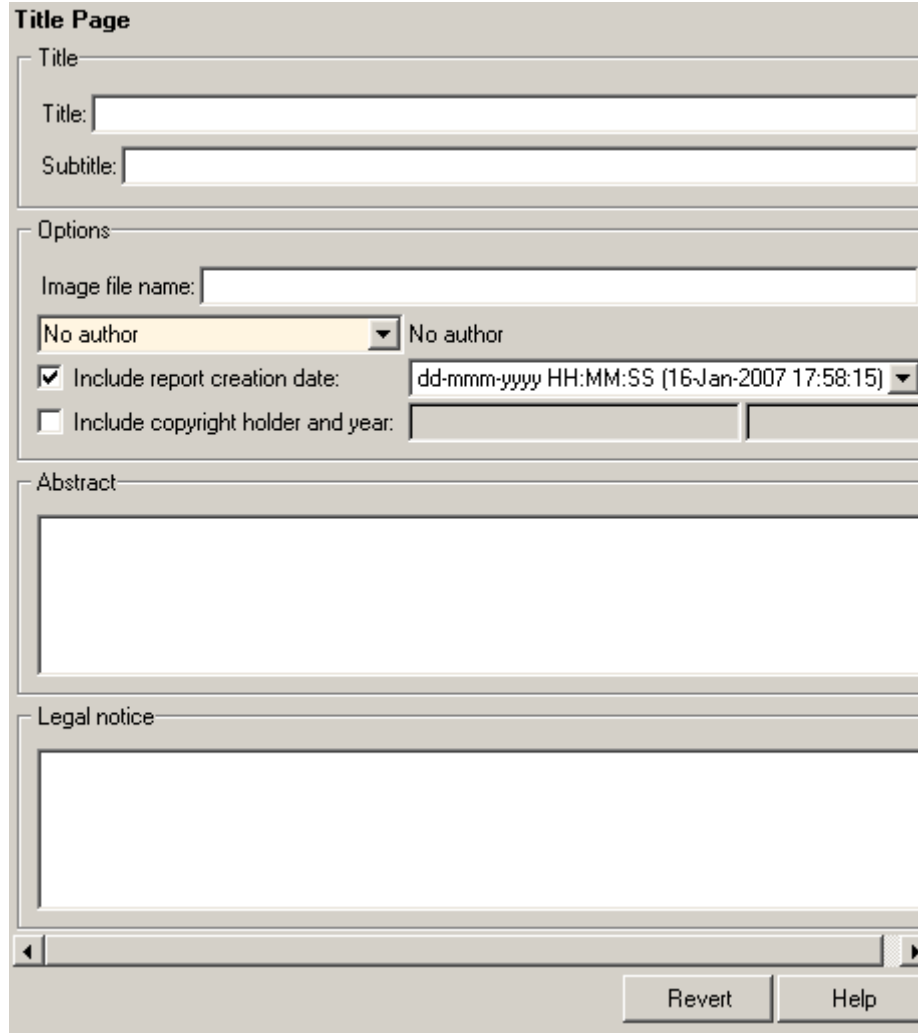
Formatting

Description

The Title Page component creates and inserts a title page at the beginning of the report. The Title Page component can be used as a child component to a Chapter/Subsection component, or it can be used alone in a report template.

Attributes

The following figure shows the **Title Page** Properties pane.



All the text fields in this attribute page can contain the `%<VariableName>` notation. For more details about this notation, see "%<VariableName> Notation" on page 11-269 on the Text component reference page.

Title

Title

Enter the title of the report in this field. The title will appear in a large font.

Subtitle

Enter the subtitle in this field. The subtitle will appear under the title in a smaller font than the title.

Options

Image file name

Enter the file name for an image you want to appear on the title page. The image will appear under the subtitle.

Author

Choose `Custom` (the default) to enter a name for your report's author. Choose `No author` if you do not want an author's name to appear on the title page. Choose `Automatic author` to automatically include your user name as the author name.

The author name will appear under the subtitle, in a smaller font than the subtitle.

Include report creation date

Select this option if you want to include the report creation date, and choose the date format in the corresponding drop-down list.

Include copyright holder and year

Select this option to include copyright holder and year information.

Abstract

Enter an optional abstract for the report.

Legal notice

Enter an optional legal notice for the report.

Insert Anything into Report?

Yes. Title page.

Title Page

File Name `rptgen.cfr_titlepage`

Category

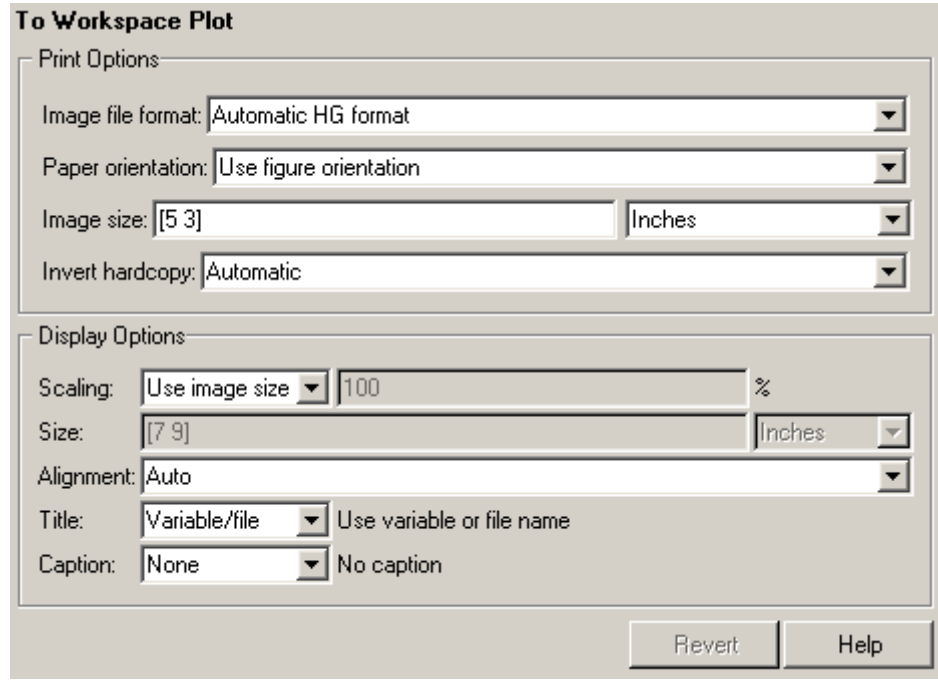
Simulink Blocks

Description

The To Workspace Plot component captures a plot figure created in the MATLAB workspace. This component inserts a figure and/or table into the report; the table contains input and output numeric values, and the figure is a plot of the values.

Attributes

The following figure shows the **To Workspace Plot** Properties pane.



Print Options

Image file format

Choose the image file format (e.g., JPEG, TIFF, etc.) from this list. Select `Automatic HG Format` to automatically choose the format best suited for the output format that you chose in the Report component. Otherwise, choose an image format that your output

viewer can read. `Automatic HG Format` is the default option. The options include:

- `Automatic HG Format` (Uses the Simulink file format selected in the Preferences dialog box)
- `Adobe Illustrator`
- `Bitmap (16m-color)`
- `Bitmap (256-color)`
- `Black and white encapsulated PostScript`
- `Black and white encapsulated PostScript (TIFF)`
- `Black and white encapsulated PostScript2`
- `Black and white encapsulated PostScript2 (TIFF)`
- `Black and white PostScript`
- `Black and white PostScript2`
- `Color encapsulated PostScript`
- `Color encapsulated PostScript (TIFF)`
- `Color encapsulated PostScript2`
- `Color encapsulated PostScript2 (TIFF)`
- `Color PostScript`
- `Color PostScript2`
- `JPEG high quality image`
- `JPEG medium quality image`
- `JPEG low quality image`
- `PNG 24-bit image`
- `TIFF - compressed`
- `TIFF - uncompressed`

- `Windows metafile`

Paper orientation

You can choose `Landscape`, `Portrait`, or `Rotated` as your paper orientation, or you can choose the `Use figure orientation` option. See the `orient` command for information about paper orientation.

Image size

Select `Use figure PaperPositionMode setting` to use the `PaperPositionMode` property of the Handle Graphics figure to set the image size in the report. See the `orient` command for information about paper position mode.

Select the `Automatic (same size as on screen)` option if you want the image in your report to be the same size as the image on the screen.

Select the `Custom` option to choose a custom image size. If you choose this option, specify the image size in the `size` and `units` fields.

- Size: Enter the size of the Handle Graphics figure snapshot in the form `wxh` (width times height). This field is active only if you choose `Custom` in the **Image size** list box.
- Units: Enter the units for the size of the Handle Graphics figure snapshot. This field is active only if you choose `Set image size` in the **Custom** list box.

Invert hardcopy

Handle Graphics figures have an `InvertHardcopy` property. This parameter inverts colors for printing; i.e., it changes dark colors to light colors and vice versa.

Select one of the following options:

- `Automatic` — Select this option to automatically change dark axes colors to a light axes color. If the axes color is a light color, it is not inverted.

- `Invert` — Select this option to change dark axes colors to light axes colors and vice versa.
- `Don't invert` — Select this option if you do not want to change the colors in the image displayed on the screen for printing.
- `Use figure's InvertHardcopy setting` — Select this option to use the `InvertHardcopy` property set in the Handle Graphics image.
- `Make figure background transparent` — Select this option to make the image background transparent.

Display Options

Scaling

Select `Fixed size` to specify the number and type of units.

Select `Zoom` to specify the percentage, the maximum size, and the units of measure.

Select `Use image size` if you want the image in your report to be the same size as the image on the screen.

Size

Enter the exact size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Fixed size` in the **Scaling** drop-down list.

Max size

Enter the maximum size of the snapshot in the form `w h` (width height) and press **Enter**. This field is active only if you choose `Zoom` from the **Scaling** drop-down list.

Units

Enter the units for the size of the snapshot. This field is active only if you choose `Zoom` or `Fixed size` in the **Image size** list box.

Alignment

Options are `Auto`, `Right`, `Left`, and `Center`.

Note This option is supported only in PDF and RTF reports.

Title
Type the text you want to appear above the snapshot.

Caption
Type the text you want to appear below the snapshot.

Insert Anything into Report?

Yes. Figure.

File Name

`rptgen_sl.csl_blk_toworkspace`

Truth Table

Category

Stateflow

Description

The Truth Table component reports on the truth tables in Simulink and Stateflow models. The component displays both the condition table and the action table. This component is context sensitive.

If the Truth Table component has the Simulink Model Loop as its parent, it reports on all truth tables in the current model.

If it has the Simulink System Loop as its parent, it reports on all truth tables in the current system.

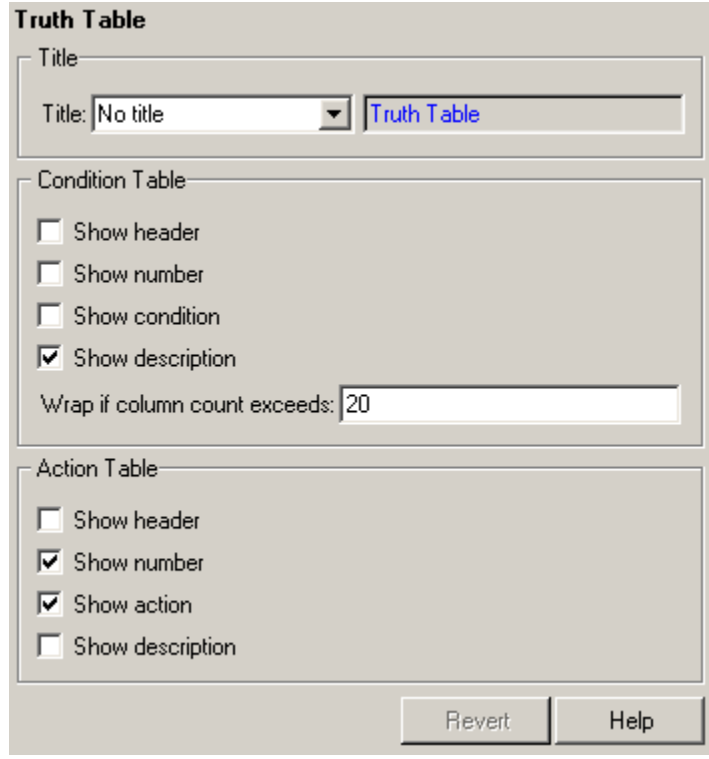
If it has the Simulink Block Loop as its parent, it reports on all truth tables in the current block.

If it has the Simulink Signal Loop as its parent, it reports on all truth tables in the current signal.

If it has a Stateflow component, such as Stateflow Filter, Machine Loop, or State Loop, as its parent, it reports on all truth tables in the current object type specified by the Stateflow component.

Attributes

The following figure shows the **Truth Table** Properties pane.



Title

Title

Choose the title for the truth table:

- `No title`
- `Use Stateflow name`
- `Custom`

Condition Table

Specify the display of the condition table:

Show header

Select this option to display the column headers in the table.

Show number
Select this option to display the condition number column in the table.

Show condition
Select this option to display the condition column in the table.

Show description
Select this option to display the description column in the table.

Wrap if column count
Select this option to specify how many table columns to display before creating a table continuation. Note that if the number specified is higher than the number of columns that can be displayed on the page, some columns are not displayed in the report.

Action Table

Show header
Select this option to display the column headers in the table.

Show number
Select this option to display the condition number column in the table.

Show condition
Select this option to display the condition column in the table.

Show description
Select this option to display the description column in the table.

Note If **Show description** is not selected, no action table appears in the report.

Insert Anything into Report?

Yes. Table.

File Name `rptgen_sf.csf_truthtable`

Variable Table

Category

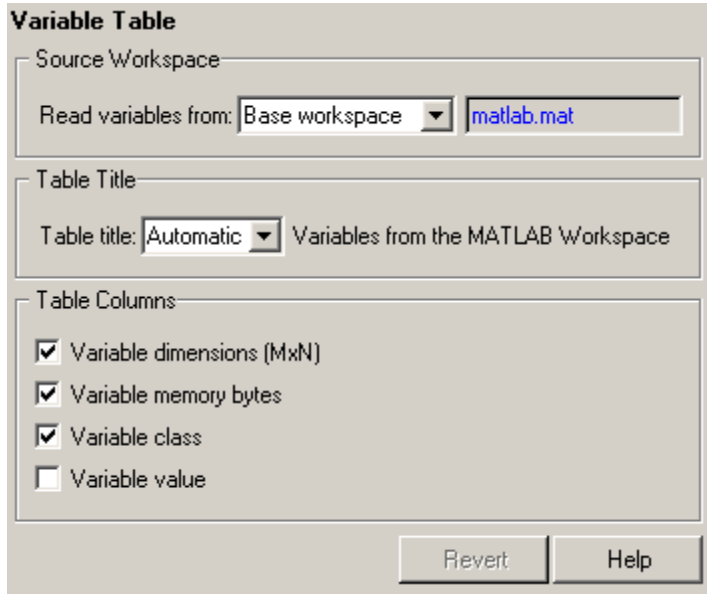
MATLAB

Description

The Variable Table component creates and inserts a table that includes all variables in the MATLAB workspace. You can find all the variables in the MATLAB workspace by typing `whos` at the command line.

Attributes

The following figure shows the **Variable Table** Properties pane.



Source Workspace

Read variables from

Select one of the following options:

- `Base workspace` — Select this option to read variables from the MATLAB workspace.
- `MAT-file` — Select this option to read variables from a binary file with a `.mat` extension. You can use the `%<VariableName>` notation. For more details about this notation, see

"%<VariableName> Notation" on page 11-269 on the Text component reference page.

Table Title

Table title

Select one of the following options for the table title:

- `Automatic (Variables from MATLAB workspace)` — Select this option if you want the table title to be a variable name from the MATLAB workspace.
- `Custom` — Select this option and enter the title in this field if you want a custom title for the variable table.

If you have a title for a table, it is included in the list of tables if the stylesheet you chose in the Report Options attribute page supports it. See "Setting the Report Format" on page 5-3 to find stylesheets that have a list of tables.

Table Columns

Select information to be contained in table columns:

- `Variable dimensions (MxN)` — Select this option to include the size of the variable.
- `Variable memory bytes` — Select this option to include the number of bytes of memory occupied by the variable.
- `Variable class` — Select this option to include the variable class.
- `Variable value` — Select this option to include the value of the variable. Note that large variable arrays are collapsed to `[MxN CLASS]`. For example, if you have a 300-by-200 double array, it appears in the report as [`300x200 DOUBLE`].

Example

This is an example of a variable table that includes size, memory bytes, and value information in the table columns.

Variable Table

Name	Size	Bytes	Value
aCell	1x2	238	{ [1 2 3 4] Speed (kph) }
aNumber	1x1	8	1
aString	1x11	22	Speed (kph)
aStructure	1x1	302	[struct w/ fields. Inputs, Outputs]
aVector	1x4	32	[1 2 3 4]

Insert Anything into Report?

Yes. Table.

File Name

`rptgen.cml_whos`

Category

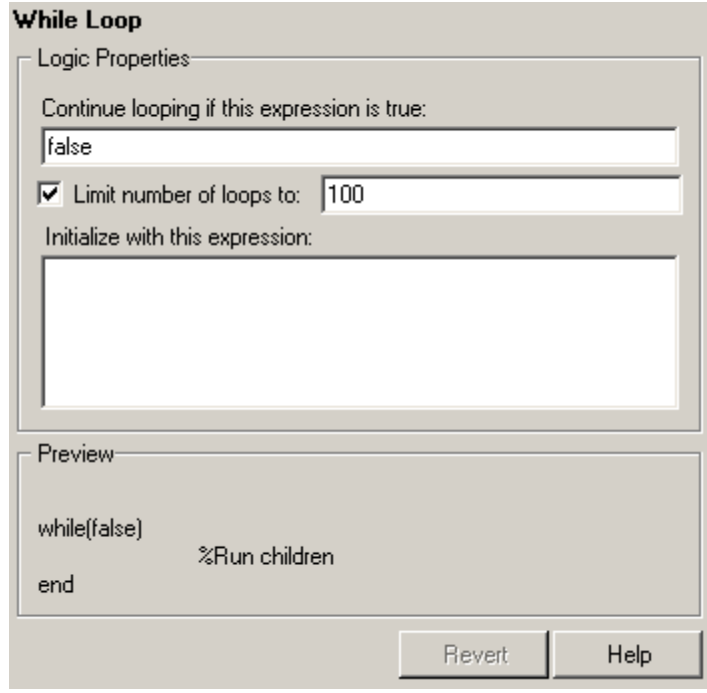
Logical and flow

Description

The While Loop component iteratively executes its child components while the string specified in the attributes page is true. You can limit the number of repetitions to prevent infinite loops. The While Loop component must have at least one child component; the purpose of this component is to run its children several times. If it does not have any children, this component does not add anything to the report.

Attributes

The following figure shows the **While Loop** Properties pane.



Logic Properties

Continue looping if this expression is true

Enter the string to be evaluated in this field. This string should be a valid MATLAB expression that evaluates to 1 or 0 (true or

false) in the workspace. For example, if a = 1, b = 2, and c = 3 in the MATLAB workspace, and you enter the following at the command line,

```
d=(a>b/c)
```

then MATLAB returns the following:

```
d = 1
```

Because 1 is greater than `b/c` (2/3), this expression is true, and evaluates to 1.

Limit number of loops to
Use this option to prevent infinite loops. Use the left and right arrows to increase or decrease the number of loops.

Initialize with this expression
Enter a valid MATLAB expression to initialize the loop.

Insert Anything into Report?

Yes, as long as it has a child component.

File Name

`rptgen_lo.clo_while`

A

Examples

Use this list to find examples in the documentation.

Looping Components Example

"Working with Looping Components" on page 7-2

Property Table Components Example

"Working with Property Table Components" on page 7-14

Summary Table Components Example

"Working with Summary Table Components" on page 7-28

Using Components

"Creating and Using Component Properties" on page 8-27
"Report Displaying Two Tickers" on page 8-33

Editing Font Size As a Derived Value in XML

"Editing Font Size as a Derived Value in XML" on page 9-23

Procedures for Some Specific Stylesheet Changes

"Making Specific Stylesheet Changes" on page 9-38

Making Changes in Font Size, Page Orientation, and Paper Type

"Changing Font Size, Page Orientation, and Paper Type" on page 9-38

Adding a Company Logo to a Report

"Adding a Company Logo to Headers in a PDF Report" on page 9-41

Index

F

G

H

I

L

M

N

O

P

R

S

T

V

W

X