

# Report Generator

For Use with MATLAB<sup>®</sup> and Simulink<sup>®</sup>

Computation  
└─

Visualization  
└─

Programming  
└─



User's Guide

*Version 1*

## How to Contact The MathWorks:



508-647-7000

Phone



508-647-7001

Fax



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

Mail



<http://www.mathworks.com>  
<ftp.mathworks.com>  
<comp.soft-sys.matlab>

Web  
Anonymous FTP server  
Newsgroup



[support@mathworks.com](mailto:support@mathworks.com)  
[suggest@mathworks.com](mailto:suggest@mathworks.com)  
[bugs@mathworks.com](mailto:bugs@mathworks.com)  
[doc@mathworks.com](mailto:doc@mathworks.com)  
[subscribe@mathworks.com](mailto:subscribe@mathworks.com)  
[service@mathworks.com](mailto:service@mathworks.com)  
[info@mathworks.com](mailto:info@mathworks.com)

Technical support  
Product enhancement suggestions  
Bug reports  
Documentation error reports  
Subscribing user registration  
Order status, license renewals, passcodes  
Sales, pricing, and general information

### *Report Generator User's Guide*

© COPYRIGHT 1984 - 2000 by The MathWorks, Inc.

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

**FEDERAL ACQUISITION:** This provision applies to all acquisitions of the Program and Documentation by or for the federal government of the United States. By accepting delivery of the Program, the government hereby agrees that this software qualifies as "commercial" computer software within the meaning of FAR Part 12.212, DFARS Part 227.7202-1, DFARS Part 227.7202-3, DFARS Part 252.227-7013, and DFARS Part 252.227-7014. The terms and conditions of The MathWorks, Inc. Software License Agreement shall pertain to the government's use and disclosure of the Program and Documentation, and shall supersede any conflicting contractual terms or conditions. If this license fails to meet the government's minimum needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to MathWorks.

MATLAB, Simulink, Stateflow, Handle Graphics, and Real-Time Workshop are registered trademarks, and Target Language Compiler is a trademark of The MathWorks, Inc.

Other product or brand names are trademarks or registered trademarks of their respective holders.

Printing History: January 1999 First printing for MATLAB 5.3 and Simulink 2 (Release 11)  
September 2000 Revised for Release 12 (online only)

## Getting Started

1

<b>What Is the Report Generator?</b> .....	<b>1-2</b>
Two Report Generator Packages .....	1-2
Multiple Report Formats .....	1-2
<b>Creating and Generating a Report</b> .....	<b>1-3</b>
Setup File .....	1-3
Using a Standard Setup File .....	1-3
Generating and Viewing the Report .....	1-5
Comparing Components in the Setup File to the Report .....	1-7
<b>Report Generator Overview</b> .....	<b>1-9</b>
<b>What's in This Manual</b> .....	<b>1-11</b>
Typographical Conventions .....	1-12

## Generating a Report

2

<b>Overview</b> .....	<b>2-2</b>
Methods of Generating a Report .....	2-2
When to Use Each Method .....	2-2
<b>Using the Command Line To Generate a Report</b> .....	<b>2-4</b>
<b>Using the Setup File List To Generate a Report</b> .....	<b>2-5</b>
<b>Using the Setup File Editor To Generate a Report</b> .....	<b>2-7</b>
Getting Started With the Setup File Editor .....	2-7
Elements of the Setup File Editor .....	2-8
Report Generation .....	2-10

<b>Controlling Report Generation</b> .....	<b>2-12</b>
Output Options .....	<b>2-12</b>
Other Options .....	<b>2-12</b>
<b>Specifying the Report Output</b> .....	<b>2-14</b>
Specifying the Directory for the Report File .....	<b>2-14</b>
Naming the Report File .....	<b>2-14</b>
Report Format and Stylesheets .....	<b>2-15</b>
Viewing and Printing Your Report .....	<b>2-18</b>
Autosaving the Setup File .....	<b>2-19</b>
Regenerating Images .....	<b>2-19</b>
Report Description .....	<b>2-19</b>
<b>Setting the Generation Status Update Priority Level</b> ....	<b>2-20</b>
<b>Converting a Generated Report Into Another Format</b> ...	<b>2-21</b>
Working with the Report Conversion UI .....	<b>2-21</b>
Working from the Command Line .....	<b>2-23</b>
<b>Creating Log Files</b> .....	<b>2-24</b>

## Editing Component Attributes

# 3

<b>Introduction</b> .....	<b>3-2</b>
Displaying and Editing Component Attributes .....	<b>3-2</b>
Undoing an Attribute Edit .....	<b>3-2</b>
<b>The Figure “For” Loop Component</b> .....	<b>3-3</b>
Using the Handle Graphics Figure “For” Loop: A Tutorial ....	<b>3-3</b>
Starting the Tutorial .....	<b>3-4</b>
Creating Handle Graphics Figures .....	<b>3-4</b>
Understanding Important Handle Graphics Concepts .....	<b>3-6</b>
Tutorial Figure Summary .....	<b>3-8</b>
Editing the Figure “For” Loop Component .....	<b>3-9</b>
Report Generation .....	<b>3-10</b>
Choosing Figures to Loop On .....	<b>3-10</b>

Looping on the Current Figure .....	3-11
Looping on All Visible Figures .....	3-11
Looping on Figures With Certain Tags .....	3-12
Tutorial Summary .....	3-13
<b>Property Table Components .....</b>	<b>3-14</b>
Selecting a Table .....	3-15
Choosing How a Property Name/Property Value Pair	
Is Displayed .....	3-16
Putting Text in a Table Cell .....	3-20
Title .....	3-20
Adding, Replacing, and Deleting Properties to a Table Cell ..	3-21
Displaying or Hiding Table and Cell Borders .....	3-22
Adding and Deleting Columns and Rows .....	3-23
Resizing Columns .....	3-23
Zooming and Scrolling .....	3-24
Property Table Summary .....	3-24

## Building and Editing the Setup File

# 4

<b>Overview .....</b>	<b>4-2</b>
Components .....	4-2
<b>Building a Setup File .....</b>	<b>4-3</b>
Setup File Outline .....	4-3
<b>Adding a Component to the Setup File .....</b>	<b>4-4</b>
Getting Help on a Component .....	4-5
<b>Deleting a Component .....</b>	<b>4-6</b>
<b>Deactivating/Activating a Component .....</b>	<b>4-7</b>
Deactivating a Component .....	4-7
Activating a Component .....	4-7
<b>Moving Components .....</b>	<b>4-8</b>

Up, Up and Out .....	4-9
Down, Down and Out .....	4-11
In .....	4-13
Cut, Copy, and Paste from Edit Menu .....	4-16

## How to Create Your Own Components

# 5

<b>Creating Components</b> .....	5-2
Why Create Your Own? .....	5-2
What's Involved in Creating Components .....	5-2
<b>Starting the Component Creation Wizard</b> .....	5-3
Tutorial Overview .....	5-4
<b>Entering Component Category Information</b> .....	5-5
Choosing the Component Category .....	5-5
Specifying the Component Category ID .....	5-6
Specifying the Component Category Name .....	5-6
Looking at the New Component Category Page .....	5-7
<b>Creating the Component Name</b> .....	5-8
Naming the Component .....	5-9
Specifying the Component Function Name .....	5-9
Specifying the Parent Directory .....	5-9
Specifying the Component Description .....	5-10
Subcomponents .....	5-10
Looking at the New Component Name Page .....	5-11
<b>Creating Component Attributes</b> .....	5-12
Component Attribute Tutorial Task 1 .....	5-12
Specifying the Field Name .....	5-13
Specifying the Data Type .....	5-13
Setting the Default Value .....	5-14
Setting the Control Type .....	5-14
Specifying the Attribute Name .....	5-15
Looking at the New Component Attributes Page .....	5-16

Creating Another Attribute .....	5-16
Component Attribute Tutorial Task 2 .....	5-16
<b>Creating Component Methods .....</b>	<b>5-19</b>
Creating a Custom Attributes Method .....	5-19
Creating a Custom Outlinestring Method .....	5-21
<b>Reviewing All Component Information .....</b>	<b>5-23</b>
Creating and Verifying the Component .....	5-23
<b>Tutorial: Editing and Running the New Component .....</b>	<b>5-25</b>
Tutorial Activities .....	5-25
Editing a New Component .....	5-26
Editing the Execute Method .....	5-27
Editing execute.m to Insert Text Output into the Report .....	5-28
Using the New Component .....	5-29
Creating a New Attribute .....	5-29
Changing the Outline String .....	5-34
Running the New Component .....	5-35
Editing execute.m to Insert a Figure into the Report .....	5-36
The Report Created by the Figure Random Plot Component .....	5-39
<b>Summary .....</b>	<b>5-40</b>

## Component Reference

# 6

<b>Introduction .....</b>	<b>6-2</b>
What's In This Chapter .....	6-2
Report Options .....	6-3
<b>Formatting Components .....</b>	<b>6-5</b>
Cell Table .....	6-8
.....	6-9
Chapter/Subsection .....	6-12
Image .....	6-15
Link .....	6-18

List .....	6-21
Paragraph .....	6-26
Text .....	6-28
Title Page .....	6-31
<b>Handle Graphics Components .....</b>	<b>6-34</b>
Figure “For” Loop .....	6-36
Figure Property Table .....	6-39
Graphics Figure Snapshot .....	6-48
Graphics Object Name .....	6-51
Handle Graphics Parameter .....	6-54
<b>Logical and Flow Components .....</b>	<b>6-56</b>
<if> Else .....	6-58
<if> Elseif .....	6-61
For Loop .....	6-64
Logical If .....	6-67
<if> Then .....	6-70
While Loop .....	6-73
<b>MATLAB Components .....</b>	<b>6-75</b>
Evaluate MATLAB Expression .....	6-76
Insert Variable .....	6-78
MATLAB/Toolbox Version Number .....	6-81
Variable Table .....	6-84
<b>Report Generator Components .....</b>	<b>6-87</b>
Comment .....	6-88
Empty Component .....	6-90
Import File .....	6-92
Nest Setup File .....	6-96
Stop Report Generation .....	6-99
Time/Date Stamp .....	6-101
<b>Simulink Components .....</b>	<b>6-104</b>
Block Automatic Property Table .....	6-108
Block Execution Order .....	6-111
Block Loop .....	6-115
Block Property Table .....	6-119



Block Type: Bus .....	<b>6-121</b>
Block Type Count .....	<b>6-125</b>
Block Type: Look-Up Table .....	<b>6-127</b>
Block Type: Scope Snapshot .....	<b>6-131</b>
Model Change Log .....	<b>6-134</b>
Model Functions .....	<b>6-138</b>
Model Loop .....	<b>6-141</b>
Model Property Table .....	<b>6-149</b>
Model Simulation .....	<b>6-151</b>
Model Variables .....	<b>6-154</b>
Object Linking Anchor .....	<b>6-158</b>
Object Name .....	<b>6-160</b>
Object Property .....	<b>6-163</b>
Object Summary Table .....	<b>6-166</b>
Signal Loop .....	<b>6-173</b>
Signal Property Table .....	<b>6-176</b>
System Filter .....	<b>6-178</b>
System Hierarchy .....	<b>6-180</b>
System Loop .....	<b>6-183</b>
System Property Table .....	<b>6-186</b>
System Snapshot .....	<b>6-188</b>
 <b>Stateflow Components</b> .....	 <b>6-192</b>
Chart Loop .....	<b>6-194</b>
Stateflow Linking Anchor .....	<b>6-198</b>
Stateflow Name (Chart) .....	<b>6-200</b>
Object Report .....	<b>6-202</b>
Object Summary Table .....	<b>6-204</b>
Property Table .....	<b>6-210</b>
Stateflow Snapshot (chart) .....	<b>6-212</b>
Stateflow Loop .....	<b>6-215</b>
 <b>Fixed-Point Components</b> .....	 <b>6-218</b>
Fixed-Point Block Loop .....	<b>6-219</b>
Fixed-Point Logging Options .....	<b>6-223</b>
Fixed-Point Property Table .....	<b>6-226</b>
Fixed-Point Summary Table .....	<b>6-228</b>



# Getting Started

---

<b>What Is the Report Generator?</b> . . . . .	1-2
Two Report Generator Packages . . . . .	1-2
Multiple Report Formats . . . . .	1-2
 <b>Creating and Generating a Report</b> . . . . .	1-3
Setup File . . . . .	1-3
Using a Standard Setup File . . . . .	1-3
Generating and Viewing the Report . . . . .	1-5
Comparing Components in the Setup File to the Report . . . .	1-7
 <b>Report Generator Overview</b> . . . . .	1-9
 <b>What's in This Manual</b> . . . . .	1-11
Typographical Conventions . . . . .	1-12

## What Is the Report Generator?

The Report Generator is a software package that can take any information from your MATLAB® workspace and export it to a document in the form of a report. The reports that you create with the Report Generator can include figures, data, variables, and functions from your models or M-files, as well as snapshots of system graphics and figures.

### Two Report Generator Packages

There are two Report Generator packages: the MATLAB Report Generator and the Simulink® Report Generator. If you want to create reports for MATLAB M-files, you need the MATLAB Report Generator. If you want to create reports for Simulink or Stateflow® models, you need both the MATLAB Report Generator and the Simulink Report Generator, which is built atop the MATLAB Report Generator. This manual documents both the MATLAB Report Generator and the Simulink Report Generator. This manual also documents fixed-point components, which are included with the Fixed-Point blockset.

### Multiple Report Formats

One of the key features of the Report Generator is that you can create reports in multiple documentation formats, such as:

- RTF
- XML
- SGML
- HTML

This chapter discusses how to generate a report and shows an example of a report.

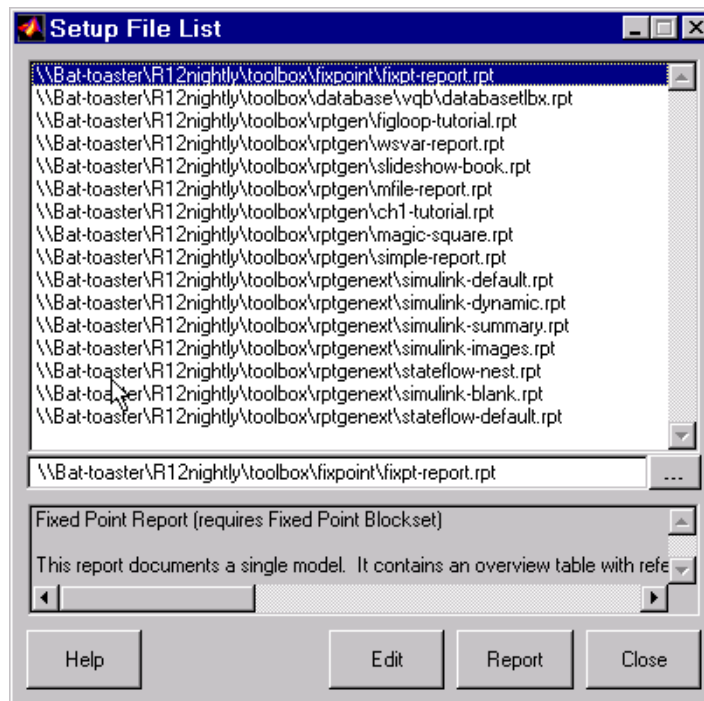
## Creating and Generating a Report

### Setup File

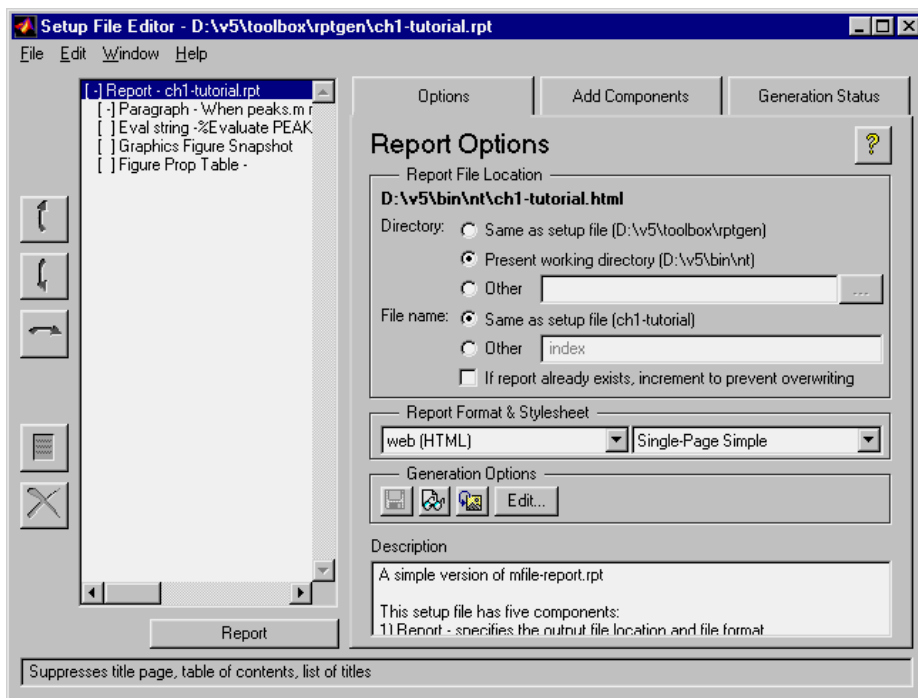
The contents of a report are specified in its setup file. You can use the standard setup files provided with the Report Generator to generate a default report or you can create a customized report by using the Setup File Editor. The Setup File Editor is the primary graphical user interface (GUI) for the Report Generator.

### Using a Standard Setup File

You can generate a report with any of the setup files that are on your path. To list all of the setup files on your path, type `rptlist` in the MATLAB command window. The Setup File List will appear.



Select `ch1-tutorial` and click on **Edit** to invoke the Setup File Editor.



## Generating and Viewing the Report

To generate a report from the Setup File Editor, click on the **Report** button, located at the bottom of the setup file outline on the left side of the Setup File Editor. This report is generated in HTML.

While the report is generating, the following events occur:

- All options in the Setup File Editor become inactive, except for the **Stop** button (the **Report** button changes to the **Stop** button while the report is generating).
- The outline highlights each component as it is executing.
- The Setup File Editor shows the **Generation Status** tab, which displays generation messages.

After report generation is complete, your Web browser will open, displaying your report.

If your report does not display on the screen, click on **Edit...** in the **Generation Options** section. Make sure that the **View report after generating** option is selected (this is selected by default).

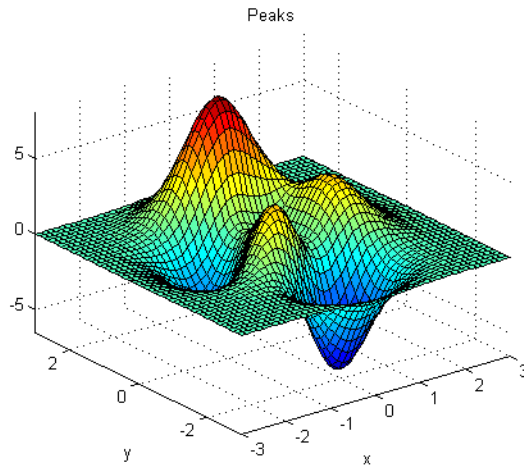


### Report Contents

The report is shown below.

When `peaks.m` runs in the workspace, the following text is displayed to the command window.

```
z = 3*(1-x).^2.*exp(-(x.^2) - (y+1).^2) ...  
    - 10*(x/5 - x.^3 - y.^5).*exp(-x.^2-y.^2) ...  
    - 1/3*exp(-(x+1).^2 - y.^2)
```



PaperPosition: (1.00, 1.00) 5.00 x 4.00 inches
--

PaperOrientation: portrait
----------------------------



## Comparing Components in the Setup File to the Report

The components in the setup file outline determine the information contained in the report. The following sections show the information each component in the outline inserts into the report.

### Report Component

The first component in the setup file outline is always the Report component. The Report component specifies output options, including output file format. The default output format is HTML. The Report component does not insert any visible output into the report.

### Paragraph Component

The next component in the setup file outline is a Paragraph component. The Paragraph component inserts the following paragraph in the report.

When peaks.m runs in the workspace, the following text is displayed to the command window:

### Evaluate MATLAB Expression Component

The third component in the setup file outline is an Evaluate MATLAB Expression component. This is represented in the outline by

Eval String- %Evaluate PEAKS in the base workspace.

This component evaluates the following M-code

```
%Evaluate PEAKS in the base workspace.
```

```
%This clears other open figures.
close all
```

```
%This creates the figure.
peaks
```

```
%This sets the figure print size to 5x4 inches
set(gcf,...
    'PaperPosition',[1 1 5 4],...
    'PaperUnits','inches');
```

and inserts the command-line output in the report.

```
z = 3*(1-x).^2.*exp(-(x.^2) - (y+1).^2) ...  
    - 10*(x/5 - x.^3 - y.^5).*exp(-x.^2-y.^2) ...  
    - 1/3*exp(-(x+1).^2 - y.^2)
```

Graphics Figure Snapshot Component

The fourth component in the setup file outline is the Graphics Figure Snapshot component. The Graphics Figure Snapshot component inserts a snapshot of the current figure into the report. Since the Evaluate MATLAB Expression component created the Peaks image, it is the current image and is inserted into the report.

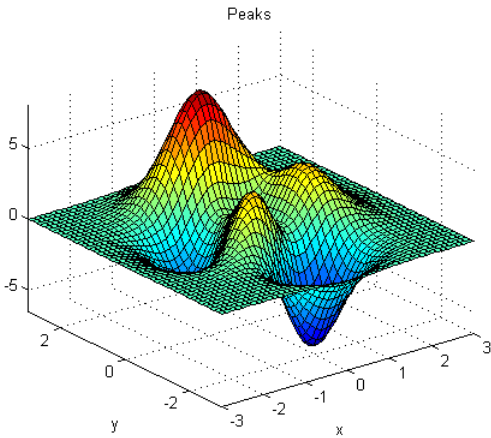


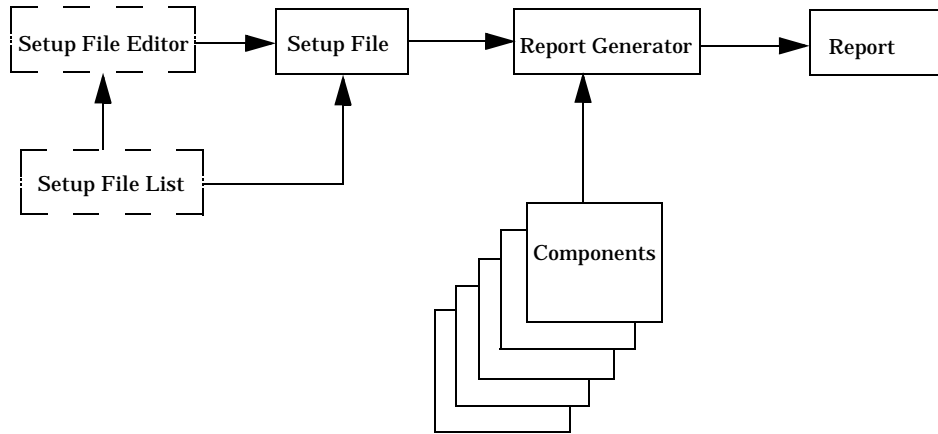
Figure Property Table Component

The last component in the setup file outline is the Figure Property Table component. The Figure Property Table component inserts a table of selected Handle Graphics® figure property name/property value pairs for the Peaks image.

PaperPosition: (1.00, 1.00) 5.00 x 4.00 inches
PaperOrientation: portrait

## Report Generator Overview

The Report Generator has several parts, which are shown in the diagram below.



Report Generator Part	Description
Setup File Editor	Primary graphical user interface (GUI) for the Report Generator; you can view and modify setup files with it. The Setup File Editor is shown in a dashed box because it is not necessary to use the Setup File Editor to generate a report. You can generate a report with any of the setup files that are on your MATLAB path.
Setup File List	Shows all the setup files in your path.
Components	Self-contained, modular elements that control the report generation process and insert information into a report.

Report Generator Part	Description
Setup File	Specifies which components will be in the report, as well as component attributes and component relationships.
Report Generator	Generating engine that uses the setup file to create a report.
Report	Formatted document that contains the information specified by the setup file.

## What's in This Manual

Chapter	Description
Chapter 1, "Getting Started"	How to generate a report using the Setup File List.
Chapter 2, "Generating a Report"	How to generate a report from the command line and from the Setup File Editor. Discusses output and generation options.
Chapter 3, "Editing Component Attributes"	How to edit attributes, using the Figure "For" Loop and the Figure Property Table components as examples.
Chapter 4, "Building and Editing the Setup File"	How to build and edit a setup file outline (move, add, delete, activate/deactivate, cut, copy, paste). Also discusses component hierarchy.
Chapter 5, "How to Create Your Own Components"	How to build custom components.
Chapter 6, "Component Reference"	Components attributes and functions; it is a reference chapter.

## Typographical Conventions

Item	Convention Used	Example
Function names and syntax, variables, example code, user-entered text	Monospace font	The set edit function opens the Setup File Editor.
Keys and menu names, menu items, and controls	<b>Boldface</b> with an initial capital letter	Click on the <b>Report</b> button to generate a report.
New terms	<i>Italics</i>	An <i>array</i> is an ordered collection of information.

# Generating a Report

---

<b>Overview</b>	2-2
Methods of Generating a Report	2-2
When to Use Each Method	2-3
<b>Using the Command Line</b>	2-4
<b>Using the Setup File List</b>	2-5
<b>Using the Setup File Editor</b>	2-7
Getting Started With the Setup File Editor	2-7
Elements of the Setup File Editor	2-8
Generating a Report	2-9
<b>Controlling Report Generation</b>	2-11
Output Options	2-11
Other Options	2-11
<b>Specifying the Report Output</b>	2-13
Specifying the Directory for the Report File	2-13
Naming the Report File	2-13
Report Format and Stylesheets	2-14
Viewing and Printing Your Report	2-17
Autosaving the Setup File	2-18
Regenerating Images	2-18
Report Description	2-18
<b>Setting the Generation Status Update Priority Level</b>	2-19
<b>Converting a Generated Report Into Another Format</b>	2-20
Working with the Report Conversion UI	2-20
Working from the Command Line	2-22
<b>Creating Log Files</b>	2-23

# Overview

## Methods of Generating a Report

There are three ways you can generate a report, as described in these sections:

- “Using the Command Line To Generate a Report” on page 2-4 (using the report command)
- “Using the Setup File List To Generate a Report” on page 2-5
- “Using the Setup File Editor To Generate a Report” on page 2-7

All three methods use setup files. A setup file specifies the following:

- Which components will be in the report
- Component attributes
- Component relationships

The Setup File Editor is the primary graphical user interface (GUI) for the Report Generator; you can view and modify setup files with it.

The Setup File List contains a list of setup files on your path and contains a brief, user-specified description of the setup file.

The report is generated for the current model or M-file, unless you specify otherwise (e.g., see the Model Loop component reference page).

## When to Use Each Method

To decide which report generation method to use, see the table below.

If you...	Use the...	Enter this command
Know the setup file you want to use and you don't need to edit it	Command line	report <SetupFileName>



If you...	Use the...	Enter this command
Want to generate a report from an existing setup file (or edit the setup file) but are not sure of the name of the file or if it is on your path	SetupFile List	rptl i st
Want to create a new report, or edit an existing setup file (for example, if you want to customize a standard setup file or edit one of your own setup files)	SetupFile Editor	setedi t or setedi t <SetupFil eName>

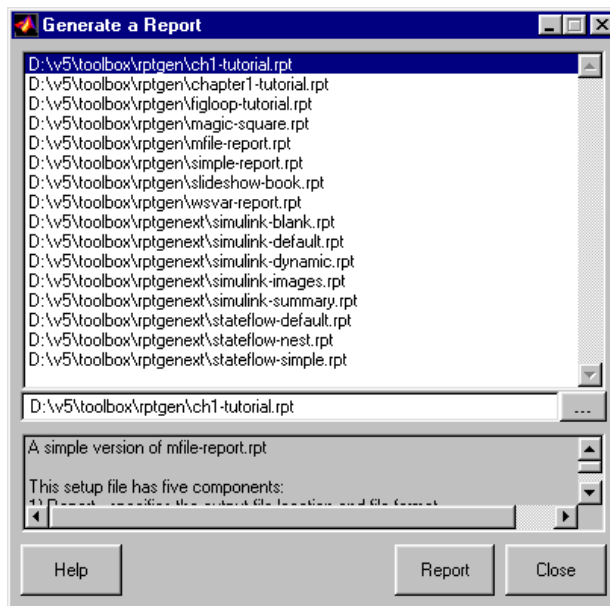
## Using the Command Line To Generate a Report

You can generate a report from the command line by typing `report <SetupFileName>` in the MATLAB command window.

For example, if you type `report ch1-tutorial` you will get the report shown in “Using a Standard Setup File” on page 1-3.

If you type `report ('<SetupFileName1>', '<SetupFileName2>', ...)`, the corresponding reports are generated and the report names are displayed in the MATLAB window.

If you type `report` without any input argument, the Generate a Report user interface (UI) appears. The Generate a Report UI lists all the setup files you have on your path.

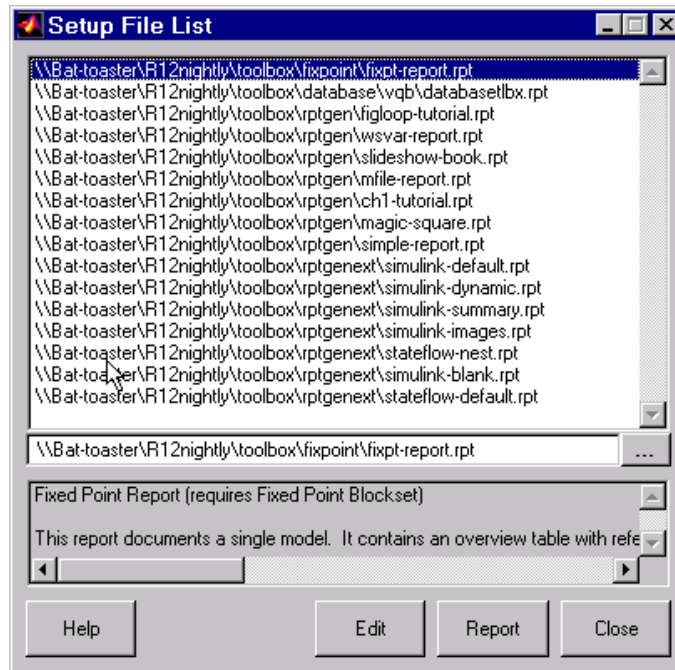


Note that a user-supplied description of the selected setup file appears in a text box at the bottom of this window.

To generate a report, select a setup file and click on the **Report** button.

## Using the Setup File List To Generate a Report

You can use the Setup File List to generate or edit a report. To invoke the Setup File List, type `rptlist` at the command line. The following will appear.

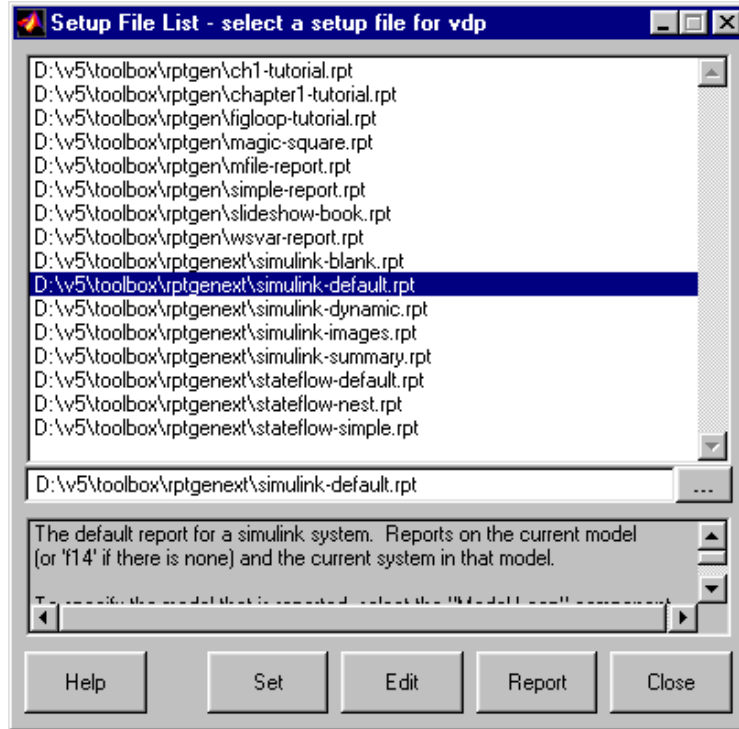


To generate a report, select a setup file and click on the **Report** button.

To edit a report, select a setup file and click on the **Edit** button. This opens the Setup File Editor. The Setup File Editor is discussed in detail in the following section, “Getting Started With the Setup File Editor”.

All Simulink systems have a property called `ReportName`. This property is used to specify which report to use for a particular system. Type `rptlist <system_name>` to start the Setup File List with Simulink’s `ReportName` property selected. (You must have Simulink to use the `rptlist` command in this manner, but you do not need the Simulink Report Generator.)

If you type `rptlist vdp`, the following UI will appear.



This Setup File List UI has an additional button, **Set**, which sets the selected setup file name as the system's new `ReportName` property. Using the **Edit** or **Report** button opens the Setup File Editor or generates a report, respectively.

You can also call this UI from a Simulink model by selecting **Report Generator...** from the **Tools** menu.

Note that you can display a list of all the setup files on your path in the command window instead of in the Setup File List UI; to do so, type

```
list = rptlist
```

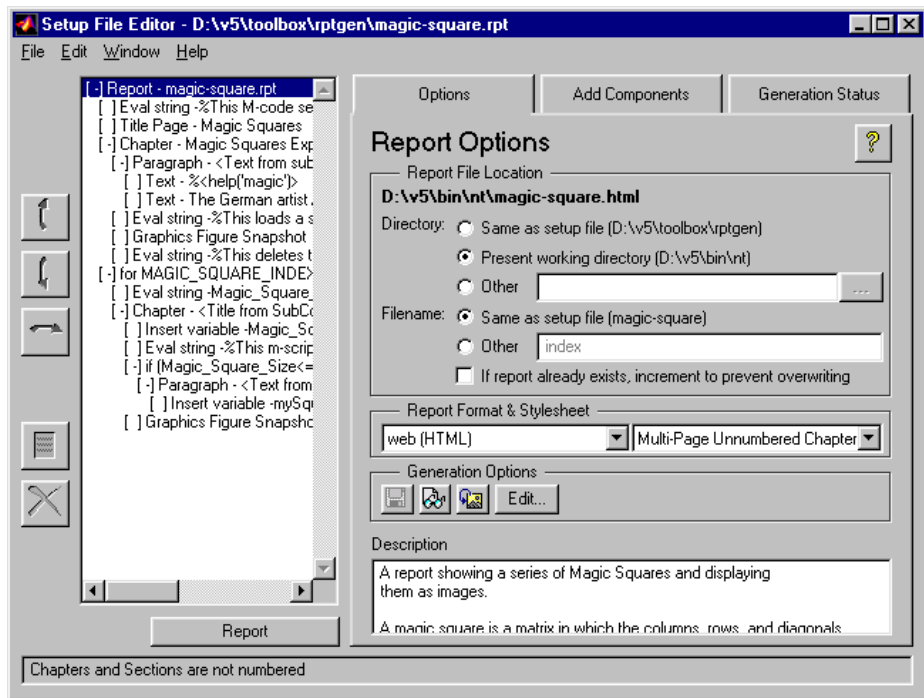
## Using the Setup File Editor To Generate a Report

You can use the Setup File Editor to view and modify your setup files and generate reports from these files. The setup file establishes the framework for a report and contains information about:

- Components included in the report and their relationships
- Report output format and style sheet used
- Name and location of the report file

### Getting Started With the Setup File Editor

To open the Setup File Editor, type `setedit` in the MATLAB command window; this opens a new (empty) setup file. If you type `setedit <setupfilename>`, the Setup File Editor opens, displaying the setup file for that report. For example, if you type `setedit magic-square`, the following window will appear.



## Elements of the Setup File Editor

### Setup File Outline

The outline of the current setup file is in the left pane of the window. The outline shows all of the components in the current setup file. Components are self-contained, modular elements that control the report generation process and insert elements into a report.

Some components can contain other components, which are then called subcomponents or child components. Subcomponents or child components are shown indented under their parent components. Components and the relationships between child and parent components are discussed in Chapter 6, “Component Reference.”

## Tab Dialog

The right pane of the setup file window contains three tabs:

- The **Options** tab lets you set report output options and change component attributes.
- The **Add Components** tab lets you add components to the outline.
- The **Generation Status** tab shows you the status of the current generation process.

The sections that follow describe these tabs in more detail.

### Menu Bar

The Setup File Editor also contains a menu bar with the following menus:

- The **File** menu lets you:
  - start a new setup file
  - open an existing setup file
  - save a setup file
  - generate a report
  - create a log file
  - close a setup file
  - exit the Setup File Editor
- The **Edit** menu lets you:
  - undo (5 levels)
  - cut
  - copy
  - paste
  - activate/deactivate
  - delete components
  - view report options

Chapter 4, “Building and Editing the Setup File”, discusses **Edit** actions:

- The **Window** menu lets you:
  - switch between multiple open setup files.
  - open the Setup File List
  - convert an SGML file to other output formats
- The **Help** menu lets you go to the Report Generator Help (this manual) or go to the Setup File Editor Help (see “Using the Setup File Editor To Generate a Report” on page 2-7).

### Report Generation

To generate a report using the Setup File Editor, click on the **Report** button, which is located under the setup file outline in the Setup File Editor. You can also select **Report** from the **File** menu. A report containing the components



listed in setup file outline will be generated on any M-file, Simulink model, or Stateflow model.

While the report is in the process of generating, the **Report** button changes to the **Stop** button and all other controls (buttons and menus) are inactive. In addition, as each component executes, it is highlighted in the setup file outline.

### Report Generation Cleanup

After the report is finished generating, the Report Generator closes all images, models, Stateflow windows, and Target Language Compiler™ (TLC) handles that were opened in the report generation process.

### Halting Report Generation

During report generation, the **Report** button changes to the **Stop** button. To stop report generation, click on the **Stop** button, which is located under the setup file outline in the Setup File Editor.

# Controlling Report Generation

When you generate a report using the Setup File Editor, you have a set of options that control aspects of report generation, such as how the report is generated, what it looks like, and whether it is displayed after generation is complete. You can set these options using the Setup File Editor.

## Output Options

These are several options you can control. Each of the options listed below is described in more detail following this section.

You can use the **Options** tab for the Report Options component to specify:

- The name and location of the report file
- The report format and style
- Whether the setup file is saved automatically upon report generation
- Whether the report will be displayed after it is generated
- Whether images will be regenerated
- A description of the report

---

**Note** To view the **Options** tab for the Report Options component, select the first component in the outline of the setup file, which is the Report Options component, and select the **Options** tab.

---

You can specify any of the above report characteristics before you generate the report.

## Other Options

You can use the **Generation Status** tab to specify what kinds of generation messages are displayed.

You can use the **Create Log File** option of the **File** menu to generate a log file. A log file is a text file that lets you view setup file information and report generation information.

For more details, see:

- “Setting the Generation Status Update Priority Level” on page 2-20
- “Creating Log Files” on page 2-24

# Specifying the Report Output

The **Options** tab for the Report Options component contains report output preferences.

## Specifying the Directory for the Report File

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

You can use the %<VariableName> notation. See “%<VariableName> Notation” in the Text component reference.

To save your report in...	Select this option...
The same directory as the setup file	<b>Same as setup file</b>
The present working directory	<b>Present working directory</b>
Another directory	Specify a directory in the <b>Other</b> text box (you can use the browse button (...) to select from a list of directories)

## Naming the Report File

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

Note that 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\_html\_files.

You can use the %<VariableName> notation. See “%<VariableName> Notation” in the Text component reference.

If you want...	Select this option...
Your report have the same filename as the setup file	<b>Same as setup file</b> (this is the default report name)
Your report to have a filename that is different than the setup file name	<b>Other</b> ; then enter the name of the report in the text box
To prevent writing over a report with the same filename (e.g., report1. html , report2. html ,...)	<b>If report already exists, increment to prevent overwriting</b>

### Report Format and Stylesheets

You can choose the report output format in the drop box on the left in this section. You should choose the output format according to the software or viewer you want to use to view the report. For example, if you want to use Microsoft Word 95, you should choose **RTF 95**.

Specify the stylesheet in the drop box on the right in this section. A stylesheet formats the report in terms of fonts, headings, etc. Stylesheets depend on the output format, that is, certain stylesheets are available for certain output formats. Each output format has a default stylesheet.

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.

If you want to use...	Select this output format...	Stylesheet Options
An HTML viewer	<b>web (HTML)</b> (default)	Web (see “Web Stylesheets” later in this chapter)
SGML	<b>Doc Book (SGML)</b>	None (no transform)
Microsoft Word 95	<b>Rich Text Format (RTF) 95</b>	Print (see “Print Stylesheets” later in this chapter)

If you want to use...	Select this output format...	Stylesheet Options
Microsoft Word 97	<b>Rich Text Format (RTF) 97</b>	Print
XML parser	<b>Flow Object Tree (XML)</b>	Print

If you want to create a report in multiple formats without regenerating your report each time, see “Converting a Generated Report Into Another Format” on page 2-21.

## Report Output Format Descriptions

- **Web (HTML)** is the default output format. It is used for publishing on the World Wide Web.
- **Doc Book (SGML)** produces source SGML. The SGML source code is always generated in addition to the output format you choose.
- **Rich Text Format (RTF) 95** is used with Microsoft Word 95 or any Word 95 viewer.
- **Rich Text Format (RTF) 97** is used with Microsoft Word 97 or any Word 97 viewer.
- **Flow Object Tree (XML)** is the XML representation of low-level DSSSL formatting constructs. It can be parsed by an XML parser.

## Web Stylesheets

Web Stylesheet	Description
<b>Single-Page Web</b>	HTML on a single page
<b>Multi-Page Web</b>	HTML with each chapter on a separate page
<b>Single-Page Simple</b>	Suppresses title page, table of contents, list of titles
<b>Single-Page Unnumbered Chapters &amp; Sections</b>	Chapters and sections are not numbered.
<b>Single-Page Numbered Chapters &amp; Sections</b>	Chapters and sections are both numbered.
<b>Multi-Page Simple</b>	Suppresses title page, table of contents, list of titles
<b>Multi-Page Unnumbered Chapters &amp; Sections</b>	Chapters and sections are not numbered.
<b>Multi-Page Numbered Chapters &amp; Sections</b>	Chapters and sections are both numbered.

Note that all Web and Print stylesheets include a list of titles, unless otherwise shown in table. The list of titles includes a list of (titled) tables and a list of (titled) figures.

Print Stylesheets

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

Viewing and Printing Your Report

You can use the **View report after generating** option to view your report. Click on **Edit...** in the **Generation Options** section and then select this option, or click on the corresponding button in the Setup File Editor.





Note that when this option is not selected, the button appears grayed out. When report generation is complete, the viewer associated with the report format output you chose is launched, displaying the report.

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

## Autosaving the Setup File

To automatically save the setup file before you generate a report, select **Save setup file on generation**. Click on **Edit...** in the **Generation Options** section and then select this option, or click on the corresponding button in the Setup File Editor.

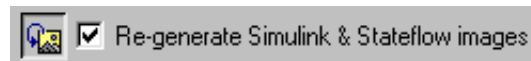


Note that when this option is deselected, the button appears grayed out.

## Regenerating Images

By default, the **Regenerate Images** button is selected. If you deselect this option, previously generated images (Simulink, Stateflow, etc.) will not be regenerated each time you generate a report; images from the previous generation will be used. This results in faster report generation. Use this option only if you have not made any changes that could affect these images.

Click on **Edit...** in the **Generation Options** section and then select this option, or click on the corresponding button in the Setup File Editor.



Note that when this option is deselected, the button appears grayed out.

## Report Description

Use the **Report Description** field to make notes and comments on your setup file. This description appears at the bottom of the Setup File List.

# Setting the Generation Status Update Priority Level

The **Generation Status** tab in the Setup File Editor displays messages about the report generation and is automatically selected when you start report generation.

You can select the types of messages you want to see during report generation from the **Show status updates** drop box, which shows the level of importance (priority level) for each type of message, with 1 being most important and 6 being least important.

The following is a list of message priority levels and the messages types associated with each level (message priority levels ranked as follows: 1= most important, 6 = least important):

- 1) **Error messages only**
- 2) **Warning messages**
- 3) **Important messages (running a loop)**
- 4) **Standard messages (running a component)**
- 5) **Low-level messages (running a nested component)**
- 6) **All messages**

If you do not want to see any messages, choose **0) No message**.

As the report is generated, messages of the chosen priority level and above will appear in the **Generation Status** tab. For example, the default message priority level is 3, which means that messages of priority level 3 (**Important messages**) and above will appear.

If you want to view messages of a different priority level after the report generation is complete, you can select the new priority level, and messages corresponding to that priority level and above will be displayed.

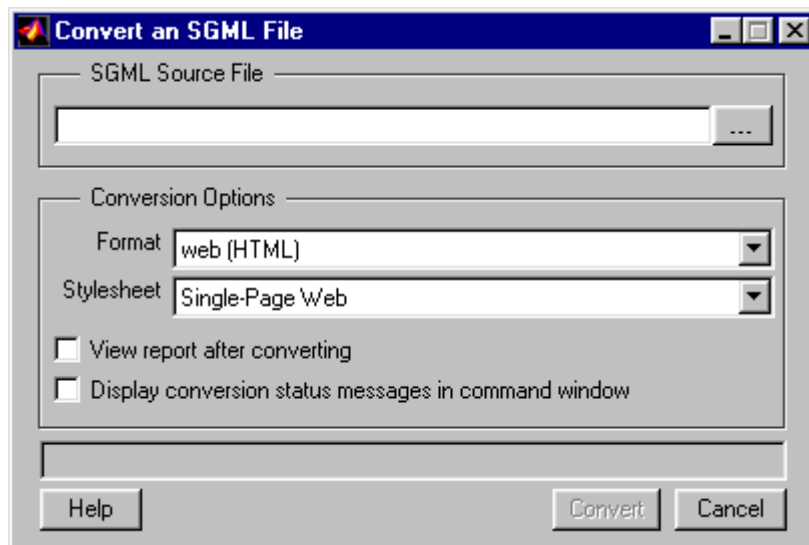
## Converting a Generated Report Into Another Format

The report generation process always produces an SGML source file and then converts the file to the output format specified in the “Report Format & Stylesheet” section. The SGML source file is `<reportname>.sgml`.

If you want to convert a report in an SGML source file to one of the other file output formats without regenerating the report, you can use the `rptconvert` command. The `rptconvert` command transforms a DocBook SGML source file created by the Report Generator to a formatted document.

### Working with the Report Conversion UI

If you type `rptconvert` without any input arguments, the report conversion UI will appear.



#### SGML Source File

Enter the name of the SGML source file you want to convert in this text box or use the browse button (...) to find it.

### Conversion Options

Specify output options in this section.

**Format.** Choose the desired output format.

**Stylesheet.** Choose the desired stylesheet.

See “Report Format and Stylesheets” earlier in this chapter for more information about format and stylesheet.

**View report after converting.** Select this option if you want to view the report after the conversion is complete.

**Display conversion status messages in command window.** Select this option if you want to display conversion status messages in the MATLAB command window after report conversion is complete.

### Converting the Report

Click on the **Convert** button to convert the report.

### How Unreadable Image Files Are Handled

If the original document contains an image that your viewer cannot read, (e.g., if you are using an HTML viewer that cannot read EPS files), then a placeholder for the image will appear in your report instead of the actual image.

Make sure that the report you are converting has images saved in a format that your viewer can read.

## Working from the Command Line

You can use `rptconvert` from the command line with varying levels of definition:

- `<ReportName>=rptconvert(' <DocBookFileName>');` where `<ReportName>` is the name of your report, `DocBookFileName` is the name of the DocBook file (with or without the `.sgml` extension). The default format (HTML) and stylesheet (Single-Page Web) are used.
- `<ReportName>=rptconvert(' <DocBookFileName>', format);` where the valid format arguments are:
  - HTML
  - RTF95
  - RTF97
  - FOT (for XML)

The format name should be entered in single quotes, for example, `'HTML'`. The default stylesheet for the selected option is used.

- `<ReportName>=rptconvert(' <DocBookFileName>', format, stylesheet);` for a discussion of valid stylesheet options for each output format, see “Report Format and Stylesheets”. Type `rptconvert #STYLESHEETLIST` for a list of valid stylesheet arguments. Note that the stylesheet name should be entered in single quotes.

When using `rptconvert` from the command line, you can create a variable that is a cell array of strings that contain conversion status messages. For example, to create a cell array of conversion status messages called `messages`, type the following.

```
[<ReportName>, messages]=rptconvert(' <ReportName>.sgml', 'html')
```

Note that this will convert the report to HTML format.

## Creating Log Files

The main function of a log file is to document a setup file. The log file is a text file that allows you to view setup file information and report generation information. A log file can be used for many purposes, including:

- As a debugging tool
- As a reference on a setup file
- To share information about a setup file through e-mail

A log file includes the following:

- Setup file outline
- Components and their attributes
- Generation status messages currently displayed in the **Generation Status** tab.
- All MATLAB licenses

To generate a log file, select **Create Log File** from the **File** menu. This will launch the MATLAB Editor/Debugger, displaying the log file.

The log file is saved in the same directory as the setup file and is named <setupfilename>.log.

The following two pages show an example of a log file (this is the log file for ch1-tutorial.rpt).

Report Generator Log File

Created: 12-Oct-1998  
Setup File: D:\v5\toolbox\rptgen\ch1-tutorial.rpt

=====

Outline Treeview:

```
[ - ] Report - ch1-tutorial.rpt
  [ - ] Paragraph - When peaks.m runs in the workspace, the
        following text is displayed to the command window:
    [ ] Eval string - %Evaluate PEAKS in the base workspace
    [ ] Graphics Figure Snapshot
```

[ ] Figure Property Table -

=====  
Generation Status Messages:

Running component Paragraph.  
Running component Evaluate MATLAB Expression.  
Running component Graphics Figure Snapshot.  
Running component Figure Property Table.  
Beginning html output transform (this may take a moment)  
Output transform complete.  
Output file is "D:\v5\bin\nt\ch1-tutorial.html".  
File viewer for "D:\v5\bin\nt\ch1-tutorial.html" launched.  
Report complete

=====

(1) coutline

DirectoryType: 'pwd'  
DirectoryName: ''  
FilenameType: 'setfile'  
FilenameName: 'index'  
isIncrementFilename: 0  
Format: 'HTML'  
Stylesheet: 'html - !SingleClearTitleToCLot'  
isView: 1  
isRegenerateImages: 1  
isAutoSaveOnGenerate: 0  
isDebug: 0  
Description: [11x78 char]

(2) cfrparagraph

TitleType: 'none'

```
ParaTitle: ''  
ParaText: {[1x90 char]}
```

### (3) cml eval

```
EvalString: {8x1 cell}  
isCatch: 0  
CatchString: {'disp(['Error during eval: ', lasterr])'}  
isDiary: 1  
isInsertString: 0
```

### (4) chgfigsnap

```
FigureHandle: []  
isCapture: 0  
PaperOrientation: ''  
isResizeFigure: ''  
PrintSize: [5 4]  
PrintUnits: 'inches'  
InvertHardcopy: ''  
ImageFormat: 'AUTOHG'  
ImageTitle: ''
```

### (5) chgfigproptable

```
TableTitle: ''  
isBorder: 1  
TitleRender: 'v'  
ColWidths: [0.3000 1.0500]  
TableContent: [2x1 struct]  
SingleValueMode: 0  
isRemoveEmptyColumns: 0  
isRemoveEmptyRows: 1
```



=====

-----  
MATLAB Version 5.3.0.30976a (R11) Beta 4 on PCWIN  
MATLAB License Number: mathworks  
-----

MATLAB Tool box	Version 5.3 (R11) beta 4
4-Sep-1998	
Simulink	Version 3.0 (R11) beta 3
29-May-1998	
Stateflow	Version 2.0 (R11) beta 4
06-Sep-1998	
Real-Time Workshop	Version 2.2.0
01-Jan-1998	
Report Generator tools	Version 1.0 (R11)
11-May-1998	
Simulink Report Generator	Version 1.0 (R11)
11-May-1998	



# Editing Component Attributes

---

<b>Introduction</b> . . . . .	3-2
Displaying and Editing Component Attributes . . . . .	3-2
Undoing an Attribute Edit . . . . .	3-2
 <b>The Figure “For” Loop Component</b> . . . . .	3-3
Using the Handle Graphics Figure “For” Loop: A Tutorial . . . . .	3-3
Starting the Tutorial . . . . .	3-4
Creating Handle Graphics Figures . . . . .	3-4
Understanding Important Handle Graphics Concepts . . . . .	3-6
Tutorial Figure Summary . . . . .	3-8
Editing the Figure “For” Loop Component . . . . .	3-9
Generating a Report . . . . .	3-10
Choosing Figures to Loop On . . . . .	3-10
Looping on the Current Figure . . . . .	3-11
Looping on All Visible Figures . . . . .	3-11
Looping on Figures With Certain Tags . . . . .	3-12
Tutorial Summary . . . . .	3-13
 <b>Property Table Components</b> . . . . .	3-14
Selecting a Table . . . . .	3-15
Choosing How a Property Name/Property Value Pair Is Displayed . . . . .	3-16
Putting Text in a Table Cell . . . . .	3-20
Title . . . . .	3-20
Adding, Replacing, and Deleting Properties to a Table Cell . . . . .	3-21
Displaying or Hiding Table and Cell Borders . . . . .	3-22
Adding and Deleting Columns and Rows . . . . .	3-23
Resizing Columns . . . . .	3-23
Zooming and Scrolling . . . . .	3-24
Property Table Summary . . . . .	3-24

# Introduction

Component attributes (parameters) determine what information a component inserts in the report and how the component will execute.

This chapter discusses changing attributes for two important components: the Handle Graphics Figure “For” Loop component and the Handle Graphics Figure Property Table component. These components are used as examples because they represent two large classes of components: looping components and property table components.

Attributes of individual components are discussed in detail in Chapter 6, “Component Reference.”

## Displaying and Editing Component Attributes

When you select a component in the setup file outline, the **Options** tab displays component attributes. You can edit the attributes that are displayed.

You can edit only one component’s attributes at a time. If you select multiple components, the **Options tab** displays the attributes of the first component in the list, but this attribute page will be inactive and appear gray.

## Undoing an Attribute Edit

If you want to undo an edit action to a component attribute, select **Undo** from the **Edit** menu. There are five levels of “undo” (i.e., you can undo as many as your last five actions).

If you cannot undo a particular action, the **Undo** option appears grayed out or inactive.

## The Figure “For” Loop Component

A looping component executes its subcomponents several times. There are two types of looping components: *flow control* (e.g., For Loop, While Loop) and *object* (e.g., Handle Graphics Figure “For” Loop, Simulink Model Loop).

One type of object looping component is the Handle Graphics Figure “For” Loop. The Handle Graphics Figure “For” Loop executes its subcomponents (or loops over them) several times, each time applying its subcomponents to a different figure. You use the attributes page to specify the figures on which looping is to occur. The tutorial in the next section shows how the Handle Graphics Figure “For” Loop works.

### Using the Handle Graphics Figure “For” Loop: A Tutorial

This section presents a figure loop tutorial to demonstrate how to use the attributes page for the Handle Graphics Figure “For” Loop. This tutorial shows how editing a component’s attributes changes the information that the component inserts in the report.

In this figure loop tutorial, you will:

- Create Handle Graphics figures for this tutorial.
- Edit Handle Graphics Figure “For” Loop attributes to loop on certain figures that you created. You will loop on:
  - The current figure
  - Figures with certain tags
  - All visible figures

The figures included in the loop will appear in this report.

---

**Note** This tutorial creates .png images. Make sure you have a viewer that can read .png images (for example, Netscape 4.05 or later), and choose the appropriate output format in the Setup File Editor. See “Specifying the Report Output” on page 2-14 for more information on output formats and viewers.

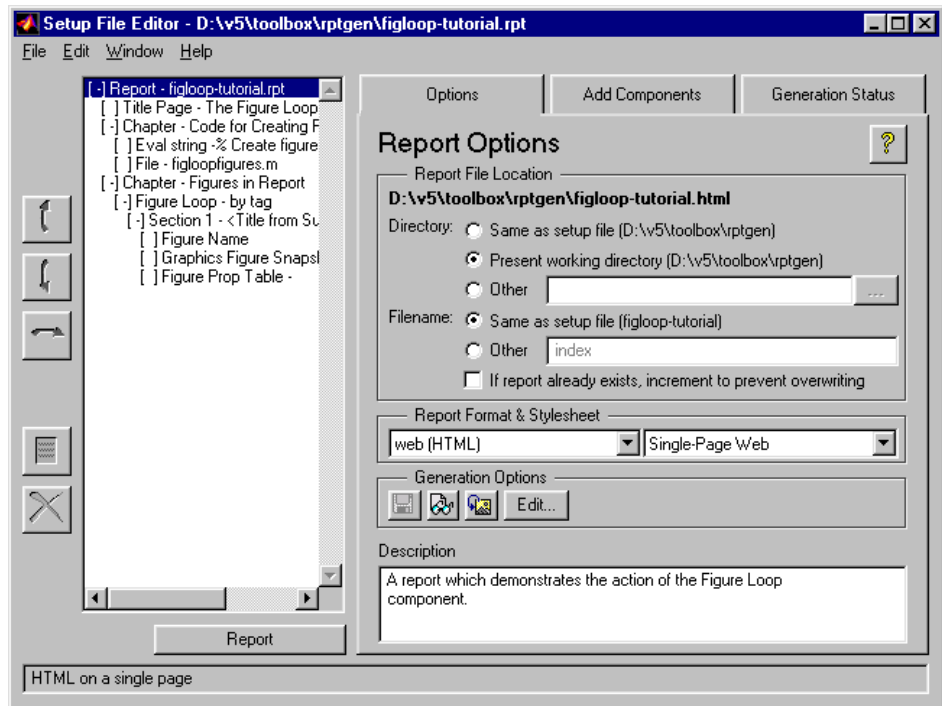
---

## Starting the Tutorial

To use the figure loop tutorial and edit the setup file, type the following at the command line.

```
setedit figloop-tutorial
```

The figure loop tutorial setup file will appear in the Setup File Editor, described in “Using the Setup File Editor To Generate a Report” on page 2-7.



## Creating Handle Graphics Figures

To create the Handle Graphics figures that you will be using in this tutorial, you should first close all of your figures. To close all figures that may still be open, type `close all` at the command line.

Type `fig loopfigures` to create Figures 1 - 5. Figures 1,3, and 5 appear on the screen, but Figures 2 and 4 do not.

Figures 1, 3 and 5 are shown below.

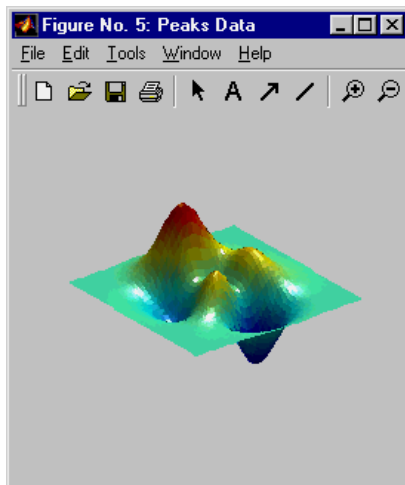
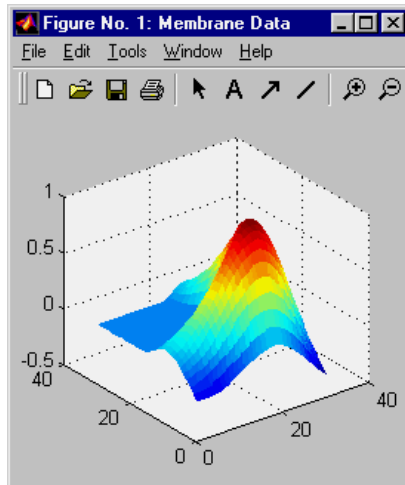
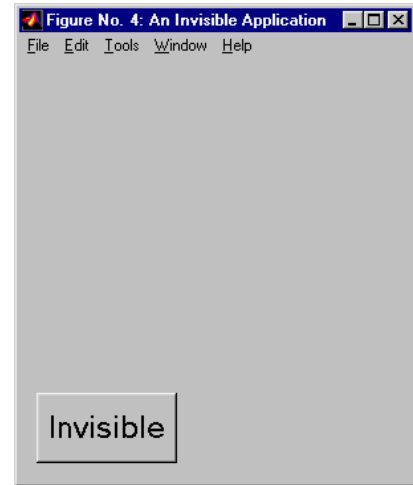
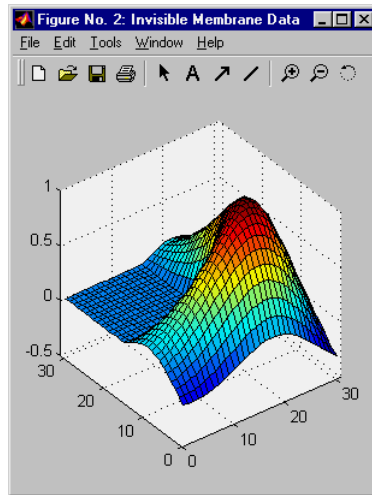


Figure 2 and 4, which do not appear on the screen because they are invisible, are shown below.



Before you continue with the rest of the tutorial, it is important that you understand some important Handle Graphics concepts, such invisible figures, `HandleVisibility`, and tags. These are described in the next section.

## Understanding Important Handle Graphics Concepts

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

- Invisible Figures
- `HandleVisibility`
- Tags

### Invisible Figures

When you type `figure`, Figures 1, 3, and 5 appear (see previous page), 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

Figures 3 and 4 have `HandleVisibility='off'`. `HandleVisibility` is a parameter of a Handle Graphics figure, and it determines whether the handles in your figure are visible. See *Using MATLAB Graphics* for more information on `HandleVisibility`.

To list all figures in your workspace that have `HandleVisibility='on'`, type `get(0, 'children')` at the command line. The results show that Figures 5, 1, and 2 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, if any tag, 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 Figure 1 has a tag, type `get(1, 'tag')`. The following will appear.

```
ans =
    membrane
```

This means that Figure 1 has a tag called `membrane`.

Every figure in this tutorial has a tag. However, by default, figures have empty tags.

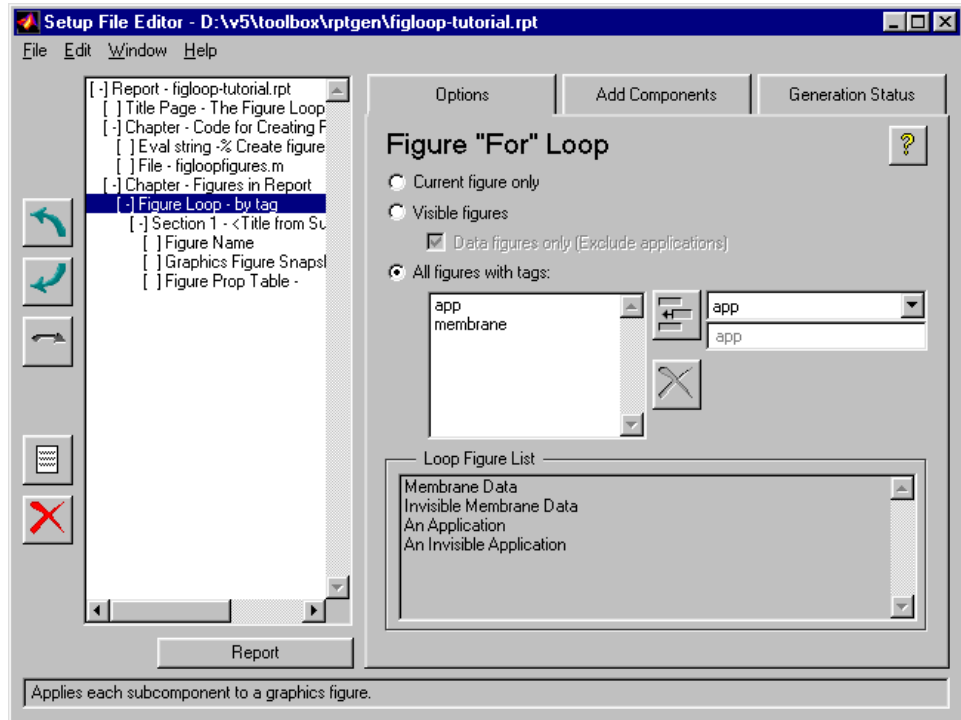
### Tutorial Figure Summary

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.

Fig. No.	Name	Visible	HandleVisibility	Tag
1	Membrane Data	on	on	membrane
2	Invisible Membrane Data	off	on	membrane
3	An Application	on	off	app
4	An Invisible Application	off	off	app
5	Peaks Data	on	on	peaks

## Editing the Figure “For” Loop Component

Select the Figure Loop component and click on the **Options** tab. This will display the attributes page for the Figure “For” loop component.



### Finding Out What Figures Are Looped On

The **Loop Figure List** shows you what figures are looped on when the report generates.

**Note** If your setup file generates new figures or changes existing figures, then the figures in the **Loop Figure List** will not be the figures that are actually reported on.

To see a figure in the **Loop Figure List**, click on the name of the figure. That figure becomes current and comes to the foreground on your screen, independent of the Setup File Editor window.

If you click on the name of an invisible figure, the figure becomes visible for a moment. For example, click on **Invisible Membrane Data** in the **Loop Figures List**; the figure briefly appears on the screen. Then type `gcf` (get current figure) in the MATLAB window. The following appears.

```
ans =  
    2
```

This means that Figure 2, “Invisible Membrane Data,” is the current figure.

---

**Note** An invisible figure won’t appear on the screen when it is created; it still exists, but it is hidden. For more information on invisible figures, see *Using MATLAB Graphics*.

---

## Report Generation

To generate a report, click on the **Report** button.

The generated report has two chapters. Chapter 1 contains the code for creating the figures. The second chapter contains the images that were looped on: “Membrane Data,” “Invisible Membrane Data,” “An Application,” and “An Invisible Application.” These are the four figures shown in the **Loop Figure List**.

## Choosing Figures to Loop On

The following sections describe what you can choose to loop on:

- “Looping on the Current Figure” on page 3-11
- “Looping on All Visible Figures” on page 3-11
- “Looping on Figures With Certain Tags” on page 3-12

## Looping on the Current Figure

Select the Figure 5 window (“Peaks Data”) to make Figure 5 current.

To include only the current figure in the report, select **Current figure only**. Notice that the **Loop Figure List** now shows **Peaks Data** only. Click on the **Report** button 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 may not be the same figure you selected as the current figure in the Setup File Editor before report generation. For example, if the report generation process creates figures in your report, then the last figure created with `HandleVisibility = 'on'` is the current figure.

---

## Looping on All Visible Figures

To loop on all visible figures, select **Visible figures**. This option runs subcomponents of the Figure “For” Loop component for all figures that are currently open and visible.

With `HandleVisibility='on'`

The **Data figures only** option is checked by default and excludes figures with `HandleVisibility = 'off'` from the loop. The **Loop Figure List** contains:

- **Membrane Data** (Figure 1)
- **Peaks Data** (Figure 5)

These are the figures that will be looped on.

Click on the **Report** button to generate a report. Notice that the report contains “Membrane Data” and “Peaks Data.”

### With HandleVisibility='on' or 'off'

Now deselect the **Data figures only** option. Figures with `HandleVisibility = 'on'` (“Membrane Data” and “Peaks Data”) and `HandleVisibility = 'off'` (“An Application”) are now included in the loop.

The **Loop Figure List** now contains:

- **Membrane Data** (Figure 1)
- **An Application** (Figure 3 - `HandleVisibility 'off'`)
- **Peaks Data** (Figure 5).

Click on the **Report** button to generate a report. Notice that the report now contains “Membrane Data,” “An Application,” and “Peaks Data.”

### Looping on Figures With Certain Tags

To loop on 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'.

See the “Tags” section earlier in this tutorial for a brief discussion of Handle Graphics tags.

### Adding a Tag to the Tag List Box

To add a tag to the tag list box, select the tag from the drop box on the right and click on the Add button, shown below.



The tutorial tag list box on the left starts with **app** and **membrane**. For example, to add peaks to the list, click on the down arrow in the tag drop box on the right and select **peaks**. Then click on the Add button.

### Adding a Tag Manually

The tag drop box on the right shows a list of tags for figures that are open. If you want to add a tag that does not appear in the tag drop box, you can manually type in the tag.

To type in a tag, select **Enter tag in edit field** and type in the name of the tag in the edit field below the tag drop box. Then click on the Add button.

### Removing a Tag From the Tag List Box

To remove a tag from the tag list box, select the tag and click on the delete button. Note that you can select multiple tags in the tag list box.

The tag list box should now contain **app**, **membrane**, and **peaks**. To remove **membrane** and **peaks**, hold the **Ctrl** key down and click on **membrane** and **peaks**. Both **membrane** and **peaks** are now selected. To delete them from the list box, click on the Delete button, shown below.



The only tag remaining in the tag list box is **app**. The **Loop Figure List** now contains **An Application** and **An Invisible Application**. Click on the **Report** button. Notice that the report only contains “An Application” and “An Invisible Application.”

## Tutorial Summary

This completes the figure loop tutorial. This tutorial showed you how to edit the Handle Graphics Figure “For” Loop attribute page to loop on different figures and include them in your report.

This tutorial showed you how to loop on:

- The current figure
- Figures with certain tags
- All visible figures

### Property Table Components

This section discusses attributes of property table components. Property table components display property name/property value pairs in a table. For example, the Handle Graphics Figure Property Table component displays property name/property value pairs in tabular form for Handle Graphics figures.

See *Using MATLAB Graphics* for more information on Handle Graphics property name/property value pairs.

Attribute pages for all property table components are similar in form. The attribute page for the Handle Graphics Figure Property Table is shown on the next page.

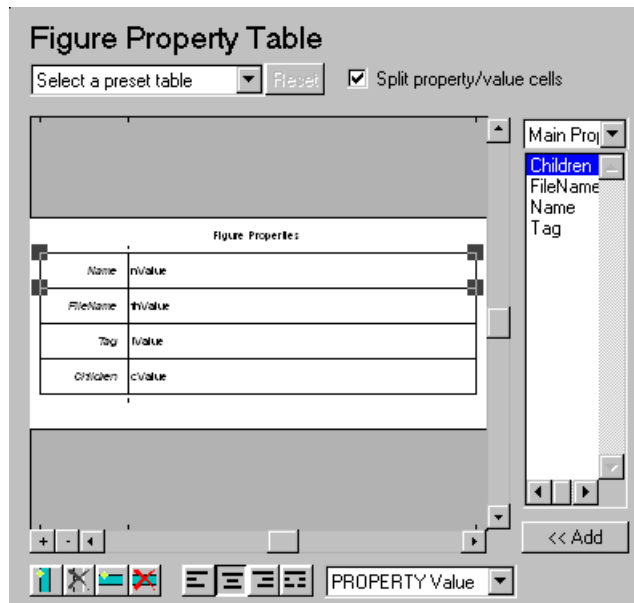
To view the component attribute page in the Setup File Editor, select the Figure Property Table component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first.

To add the Figure Property Table component to the setup file, first select the component in the Add Component tab in the Setup File Editor. Then click on the Add button, shown below.





The Handle Graphics Figure Property Table attribute page is shown below.



This section discusses how to use the Handle Graphics Figure Property Table attribute page to:

- Select a table
- Choose how a property name/property value is displayed
- Add, replace, and delete properties to a table cell
- Display or hide table and cell borders
- Add and delete columns and rows

## Selecting a Table

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-hand corner of the

attributes page. To apply a preset table, select the table and click on the **Reset** button. Note that the **Reset** button is active only when you select a preset table.

### Custom Tables

To create a custom table, you can first select a preset table and then edit it. You may want to start with the **Blank 4x4** preset table. You can add and delete rows and you can add properties.

If you want to use a custom table in more than one setup file, use **Copy Component and Paste Component** from the **Edit** menu to copy the custom table from one setup file to another.

You cannot save a custom table as a preset table. If you create a table and then select a preset table, all your changes will be lost once you click on the **Reset** button.

## Choosing How a Property Name/Property Value Pair Is Displayed

The following sections discuss property name/property value pair display options.

### Split Property/Value Cells

To have the property name and property value appear together in one cell, deselect the **Split Property/Value Cells** check box. This is the non-split mode. In the non-split mode, you can have more than one property name/property value pair per cell and you can also have 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 only be one property name/property value pair in a cell. If you have more than one pair in the cell in the Setup File Editor, then only the first pair is displayed 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 one of several ways, shown in the table below.

Display Option	"Split" Mode		"Non-Split" Mode
Value		Value	Value
		Value	Value
Property Value	Property	Value	Property Value
	Property	Value	Property Value
<i>Property Value</i> (note: this is shown as PROPERTY value on the attributes page)	<i>Property</i>	Value	<i>Property Value</i>
	<i>Property</i>	Value	<i>Property Value</i>
Property: Value	Property:	Value	Property: Value
	Property:	Value	Property: Value





Display Option	"Split" Mode		"Non-Split" Mode
Property: Value	Property:	Value	Property:Value
	Property:	Value	Property:Value
Property - Value	Property -	Value	Property - Value
	Property -	Value	Property - Value
Property - Value	Property -	Value	Property - Value
	Property -	Value	Property - Value

To select how a property name/property value pair is displayed in a table cell in the report, select that cell and then choose one of the above display formats from the drop box at the bottom of the attributes page. You can also right-click on the table to invoke the context menu, and then select one of the above formats from the menu. Note that in the above table, the cells are center-justified. The next section discusses justification options.

Justification Options

The text in each cell can appear in the report in one of four ways, using the corresponding justification button. You can also right click on the table to invoke the context menu, select **Cell justification**, and then select one of the

options shown below from the menu. These options are shown in the table below (note that the *Property Value* display option is shown here).

Option	Button	"Split" Mode		"Non-Split" Mode
Left-justified		Property	Value	Property Value
		Property	Value	Property Value
Centered		Property	Value	Property Value
		Property	Value	Property Value
Right-justified		Property	Value	Property Value
		Property	Value	Property Value
Double-justified		Property	Value	Property Value
		Property	Value	Property Value

Putting Text in a Table Cell

You can type text in a table cell in the Setup File Editor. The text appears in the report as long as you are in the non-split mode.

To enter text in a cell, double-click on the cell. When a gray box appears in the cell, you are in the edit mode. To exit the edit mode, press the **Esc** key.

In the edit mode, %<PropertyName> appears in the gray box. If you type text outside the %<>, then it will appear as text in the report. If you type text inside the %<>, it should be a valid property name. If you type an invalid property name, then that property name will appear in the report without a property value.

For example, if you choose to display the property name/property value pair as value, and enter the following in a table cell in the attribute page

My title: %<Name> is visible=%<visible>

and the value of Name is 'Membrane' and the value of visible is 'on', then the following table cell appears in the report if the cell is in the non-split mode.

My title: Membrane is visible=on
----------------------------------

If the table is in the split mode, then the following table cells appear in the report.

	Membrane
--	----------

Title

The table has a title that you can edit. The title of the table can contain properties and text. The title is always in the non-split mode.

If the table has a title, it will appear in the List of Tables. Some stylesheets offer a list of tables. For a list of stylesheets that offer this option, see “Report Format and Stylesheets” on page 2-15.

To enter title text, double click on the title, and enter text in the gray box.

To enter a property, follow the steps described in the next section, “Adding a Property”.

## Adding, Replacing, and Deleting Properties to a Table Cell

The following sections discuss how to add, replace, and delete properties to a table cell.

### Adding a Property

To add a Handle Graphics property to a table cell in the Setup File Editor, follow these steps:

- 1 Select the desired table cell.
- 2 Choose the type of property from the type of property drop box on the right side of the attributes page. The type of property drop box acts like a filter by listing properties by category.

If you want to choose from all properties associated with a Handle Graphics figure, choose **All properties** in the type of property drop box.

- 3 Choose the desired property name from the property list box.
- 4 Double-click on the property name or use the **Add** button.

If you know the name of the property, you can also type the property name in the cell. See “Putting Text in a Table Cell” in the previous section.

---

**Note** The attribute page in the Setup File Editor does not display text wrapping inside a cell, but when the report is created, the text inside a cell will wrap.

---

### Replacing or Deleting Properties

In the non-split mode, you can replace a property in a table cell in the Setup File Editor by following the instructions in “Putting Text in a Table Cell”. 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”. You cannot use these steps to delete a Handle Graphics property in a cell when the table is in non-split mode, since a cell 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 click the right mouse button to invoke the context menu. Choose **Cell borders** and then select **Top**, **Bottom**, **Right**, or **Left** to toggle the desired border on or off.

To toggle the border of the table on and off, use the context menu. Place your cursor over the table and click on the right mouse button to invoke the context menu. Then choose **Table frame** to toggle the table border on or off.



## Adding and Deleting Columns and Rows

To add or delete a column or row, select a cell, and then click on one of the following four buttons.

Press this button:	To do this:
	Add column (added to the right of the selected column)
	Delete selected column
	Add row (added below 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 setup file, you can zoom in to and out of the table with the zoom buttons, located to the left of the horizontal scroll bar.

Press this button:	To do this:
	Zoom in
	Zoom out

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

### Property Table Summary

This completes the discussion of the Property Table component. This section showed you how to edit the Handle Graphics Figure Property Table attribute page to insert a property name/property value table in your report.

This section discussed how to:

- Select a preset table or build a custom table
- Edit the appearance of the table and of individual table cells
- Edit the contents of the table by choosing property name/property value pairs to display

# Building and Editing the Setup File

---

<b>Overview</b>	4-2
Components	4-2
<b>Building a Setup File</b>	4-3
Setup File Outline	4-3
<b>Adding a Component to the Setup File</b>	4-4
Getting Help on a Component	4-5
<b>Deleting a Component</b>	4-6
<b>Deactivating/Activating a Component</b>	4-7
Deactivating a Component	4-7
Activating a Component	4-7
<b>Moving Components</b>	4-8
Up, Up and Out	4-9
Down, Down and Out	4-12
In	4-14
Cut, Copy, and Paste from Edit Menu	4-17

## Overview

### Components

A report consists of information that is specified by components in its setup file. If you have the MATLAB Report Generator package, you have the following types of components:

- Formatting
- Handle Graphics
- Logical and Flow
- MATLAB
- Report Generator

If you also have the Simulink Report Generator package, you have the Simulink and Stateflow components as well. Chapter 6, “Component Reference” is a components reference chapter and contains detailed information about each component.

You can use the Setup File Editor to select and edit components. This chapter explains how to use the Setup File Editor to edit a setup file.

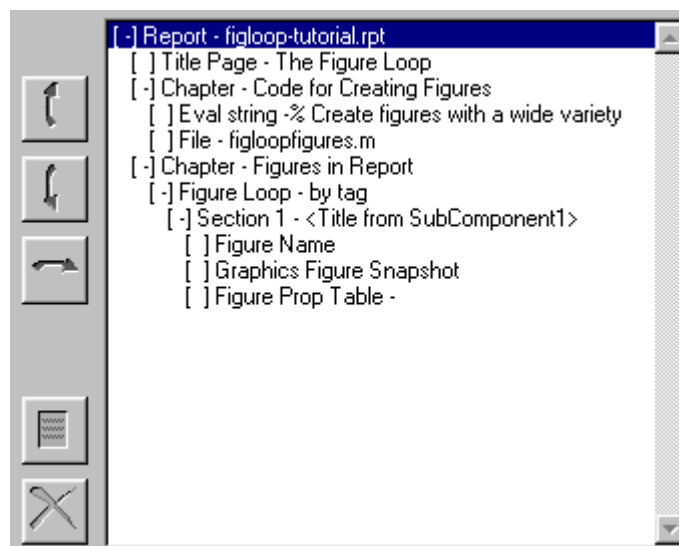
## Building a Setup File

You can build your setup file by choosing components that you want in your report. You can add, delete, move, deactivate, and activate components using the Setup File Editor. You can always undo your last change by selecting **Undo** from the **Edit** menu. There are five levels of undo (i.e., you can undo as many as your last five actions).

In order to create a valid setup file, you must nest child components inside an appropriate parent component. A parent component usually determines how or whether its child components will execute. Chapter 6, “Component Reference” discusses parent/child relationships between components in the same class.

### Setup File Outline

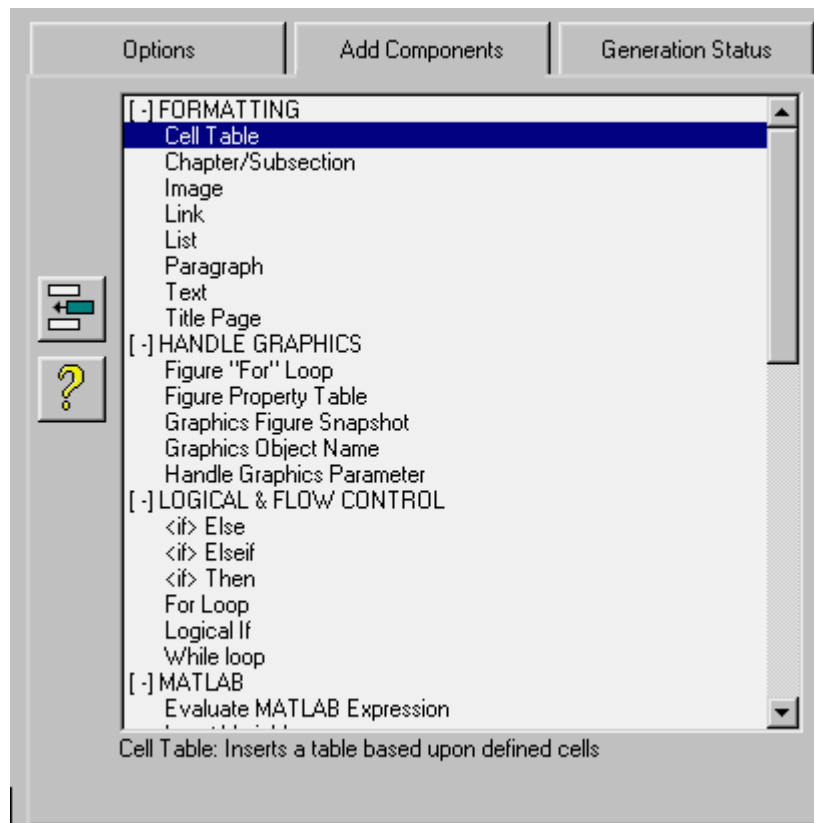
The outline of the current setup file is displayed in the left pane of the Setup File Editor. The outline contains all the components which will be included in the report. Child components are always shown indented under their parent component. A sample setup file outline is shown below.



## Adding a Component to the Setup File

To add a component to your setup file outline, click on the **Add Components** tab. It will take a moment to load all of the components on the MATLAB path, and the following message will appear: Searching MATLAB path for Report Generator components.

The **Add Components** tab (shown below) contains a list of all components. The general classes of components are listed in capital letters (e.g., FORMATTING, HANDLE GRAPHICS). Components for each class are listed below the class to which they belong. For example, Figure Property Table and Graphics Figure Snapshot are listed below HANDLE GRAPHICS because they are Handle Graphics components.



Select the component you want to add to the setup file and then click on the Add button, shown below.



Alternatively, you can double-click on the desired component to add it to the setup file. The added component appears under the currently selected component in the setup file outline.

Immediately after you add a component to the setup file, a message appears at the bottom of the **Add Components** tab, indicating that the component has been added to the setup file. Otherwise, a brief description of the highlighted component appears at the bottom of the tab.

## Getting Help on a Component

If you need help with a component, select the component in the **Add Components** tab and click on the Help button. If you need a quick summary of all components in a certain component category, select the component category and click on the Help button, shown below.



### Deleting a Component

To delete a component from your setup file, select the component you want to delete from the setup file outline and click on the Delete button, shown below.



Alternatively, you can choose **Delete Component** from the **Edit** menu.

If you are deleting a component that has subcomponents in your setup file, all of its subcomponents will be deleted as well.

---

**Note** You cannot delete the Report component, which is always the first component in the outline.

---



## Deactivating/Activating a Component

### Deactivating a Component

If you do not want a component to execute when the report is generated, you can deactivate the component. To deactivate a component, select the component and click on the Activate/Deactivate button.



Alternatively, you can also choose **Deactivate Component** from the **Edit** menu.

When a component is deactivated, an exclamation point (!) will appear to the left of the component in the setup file outline. If you select the **Options** tab for a component that you have deactivated, the title of the tab appears grayed-out.

If you are deactivating a component that has subcomponents in your setup file, all of its subcomponents will be deactivated as well.

---

**Note** You cannot deactivate the Report component, which is always the first component in the outline.

---

### Activating a Component

To activate a component that has been deactivated, select the component and click on the Activate/Deactivate button or choose **Activate Component** from the **Edit** menu.

Note that when you select a deactivated component, the Activate/Deactivate button has the following appearance.



## Moving Components

Three move buttons allow you to move components in the setup file outline. If the selected component cannot be moved to a particular position, the corresponding move button is inactive.

You can select multiple components and move them at the same time.

You must be careful when you move components, especially when there are parent-child relationships involved. Some components cannot have children and some components should not be children of other components. See Chapter 6, “Component Reference” for details.

---

**Note** The Report component cannot be moved.

---

## Up, Up and Out

You can move a component “up” or “up and out” with the Up button. The Up button is disabled for the first component after the Report component.

### Up

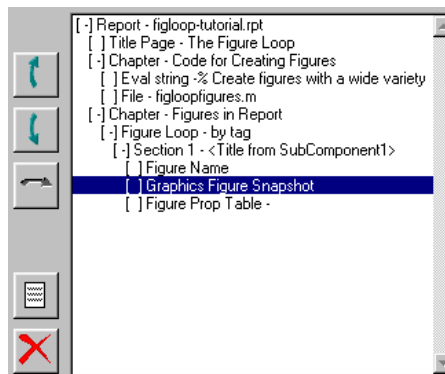
“Up” moves the selected component above the component currently listed above it. A component can move up if it is not first in its list of peers (components at the same level). No component can move above the Report component.

When the Up button is enabled for an “up” operation, it has the appearance shown below.

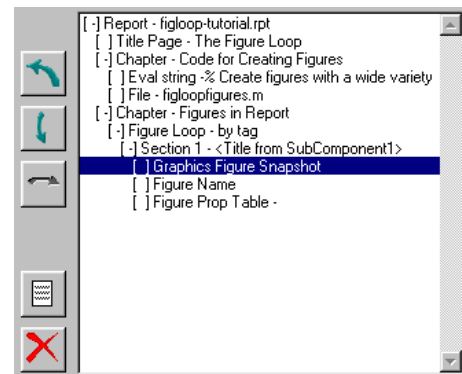


**Example.** Select the Graphics Figure Snapshot component in Outline A below. Then click on the Up button. The Graphics Figure Snapshot component moves above the Figure Name component, as shown in Outline B.

Outline A



Outline B



Up and Out

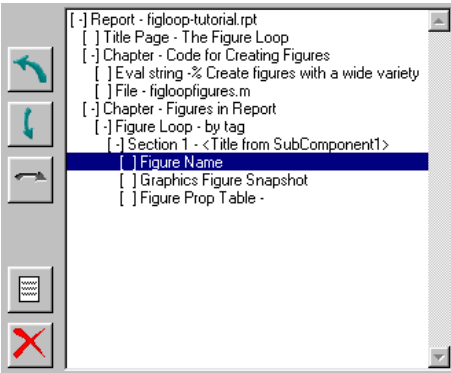
“Up and out” moves the selected component directly above its parent and up to the level of its parent. A component can move up and out if it is first in its list of peers.

When the Up button is enabled for an “up and out” operation, it has the appearance shown below.

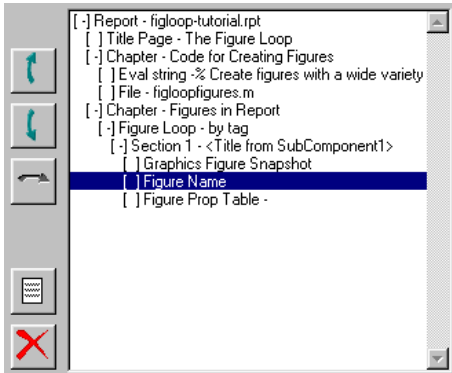


**Example.** Select the Figure Name component in Outline A below. Click on the Up button (which is now enabled for “up and out”). The Figure Name moves above the Section component, and is at the same level as the Section component.

Outline A



Outline B



## Down, Down and Out

The Down button can move a component “down” or “down and out.” The Down button is disabled if a component is the last component in the outline.

### Down

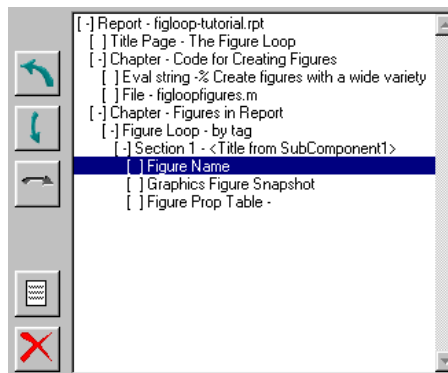
“Down” moves the selected component below the component currently listed below it. A component can move down if it is not last in its list of peers.

When the Down button is enabled for a “down” operation, it has the appearance shown below.

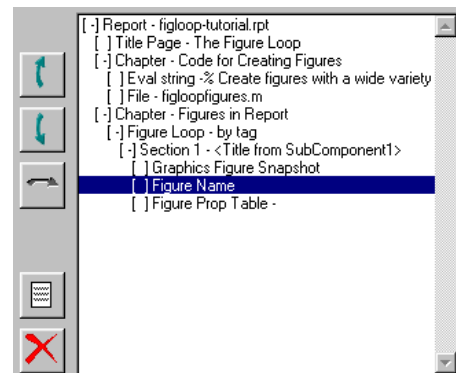


**Example.** Select the Figure Name component in Outline A below. Click on the Down button. The Figure Name moves below the Graphics Figure Snapshot component.

Outline A



Outline B



Down and Out

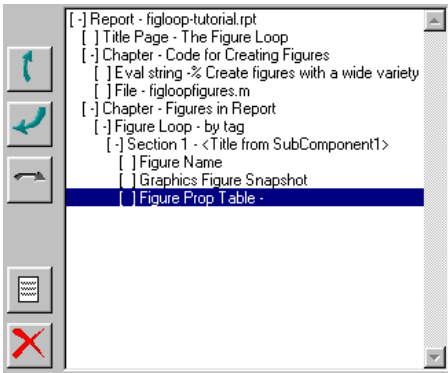
“Down and out” moves the selected component directly below its parent and to the level of its parent. A component can move down and out if it is last in its list of peers.

When the Down button is enabled for a “down and out” operation, it has the appearance shown below.

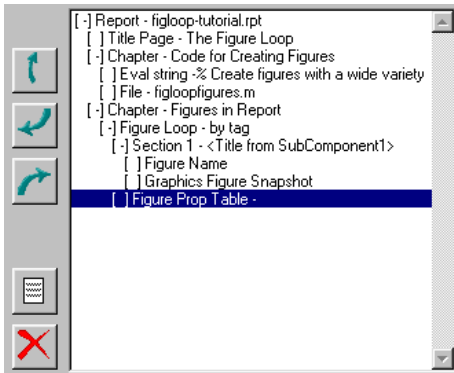


**Example.** Select the Figure Property Table component in Outline A below. Click on the Down button (which is now enabled for “down and out”). The Figure Property Table component moves to the level of its parent (the Section component) and it is the next component at that level.

Outline A



Outline B



## In

The In button can move a component “up and in” or “down and in.” The In button is disabled when neither the component above nor below the selected component can have children.

### Up and In

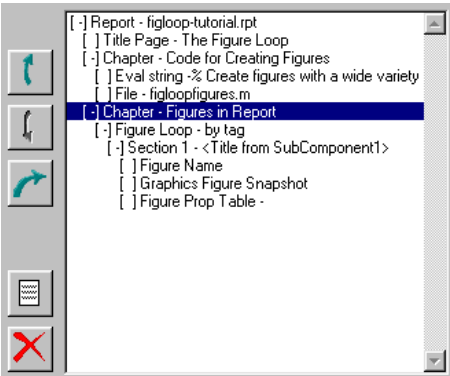
If the component listed above the selected component can have children, then “up and in” causes the selected component to become the child of the component listed above it. A component can move “up and in” if the peer component above it can have children.

When the In button is enabled for an “up and in” operation, it has the appearance shown below.

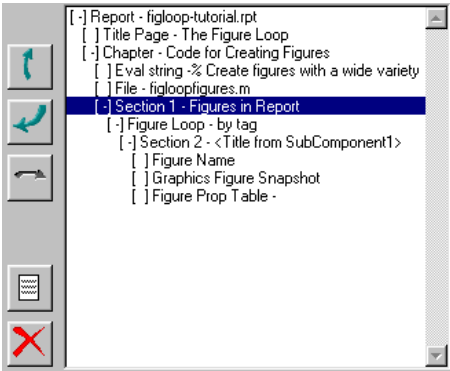


**Example.** Select the second Chapter component (Figures in Report) in Outline A below. Click on the In button (which is now enabled for “up and in”). The Chapter component becomes the child of the first Chapter component (Code for Creating Figures), and it becomes a section instead of a chapter.

Outline A



Outline B





## Down and In

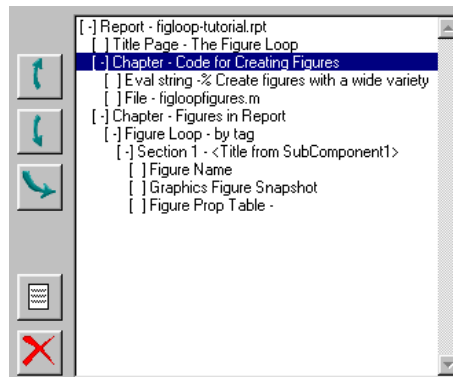
If the component listed below the selected component can have children, then the selected component becomes its child and is shown indented under the parent component below it. A component can move “down and in” if the peer component below it can have children.

When the In button is enabled for a “down and in” operation, it has the appearance shown below.

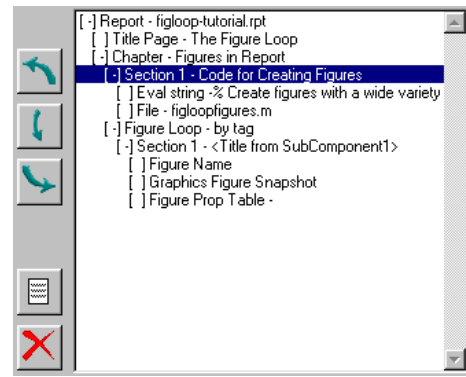


**Example.** Select the first Chapter component (Code for Creating Figures) in Outline A below. Click on the Down button (which is now enabled for “down and in”). The Chapter component will become the child of the first component below it that can take children, which is the second Chapter component (Figures in Report), and it becomes a section instead of a chapter.

Outline A



Outline B



### Cut, Copy, and Paste from Edit Menu

You can cut, copy, and paste components using the **Edit** menu.

#### Removing (Cutting) a Component

To remove a component from the outline, select the component and then choose **Cut** from the **Edit** menu.

#### Copying a Component

To copy a component from the outline:

- Select the component to copy.
- Choose **Copy** from the **Edit** menu.
- Select the component under which you want to paste the copied component (for example, if you want to copy the Text component under the Paragraph component, select the Paragraph component in this step).
- Choose **Paste** from the **Edit** menu.

Note that the **Paste** option always places the pasted component below the component that is currently selected.

# How to Create Your Own Components

---

<b>Creating Components . . . . .</b>	<b>5-2</b>
<b>Starting the Component Creation Wizard . . . . .</b>	<b>5-3</b>
<b>Entering Component Category Information . . . . .</b>	<b>5-5</b>
<b>Creating the Component Name . . . . .</b>	<b>5-8</b>
<b>Creating Component Attributes . . . . .</b>	<b>5-12</b>
<b>Creating Component Methods . . . . .</b>	<b>5-19</b>
<b>Reviewing All Component Information . . . . .</b>	<b>5-23</b>
<b>Tutorial: Editing and Running the New Component . .</b>	<b>5-25</b>
<b>Summary . . . . .</b>	<b>5-40</b>

## Creating Components

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). See the MATLAB documentation for a discussion of MATLAB object-oriented programming.

### Why Create Your Own?

In most cases, the components provided with the 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 the 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 Creation Wizard. There are two main steps to creating and running a component:

- Use the Component Creation Wizard to create the component.
- Edit `execute.m` to make the component executable (this is an M-file that is generated when you create a component).

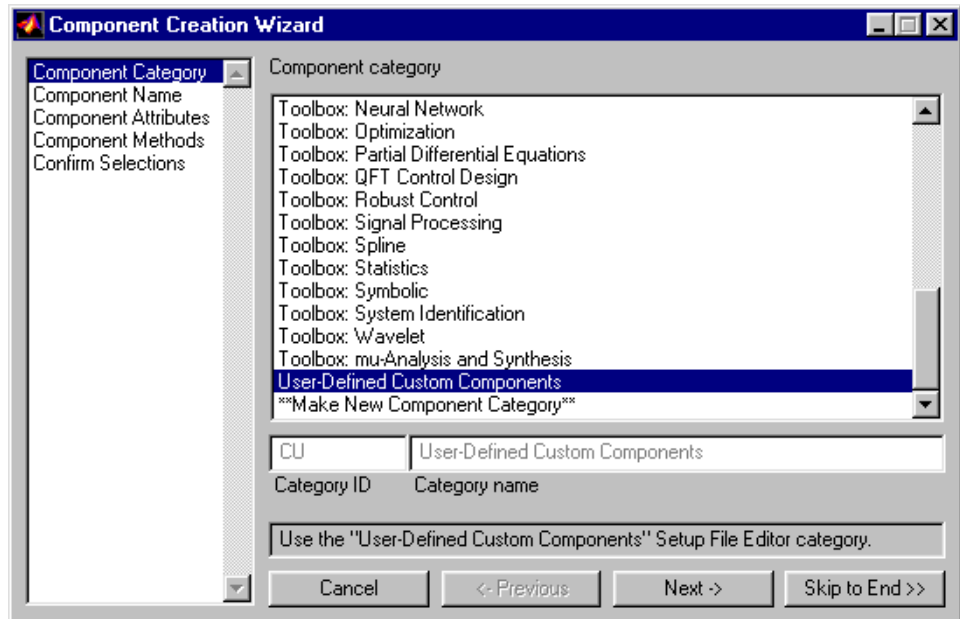
The Component Creation Wizard is a tool for creating components. It takes you through five steps; there is a screen for each step.

The Component Creation Wizard 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.

After you create the component, several M-files are created, including `execute.m`. The default `execute.m` file for a new component inserts an empty space into the report. Therefore, you should edit `execute.m` to run the component properly.

## Starting the Component Creation Wizard

To open the Component Creation Wizard, type `compwi z` at the command line.



The list box on the left side lists five pages that you should complete when you are creating a component. Although you should complete all five pages, you can skip ahead at any time if you prefer.

The five pages are:

- Component Category
- Component Name
- Component Attributes
- Component Methods
- Confirm Selections

### **Tutorial Overview**

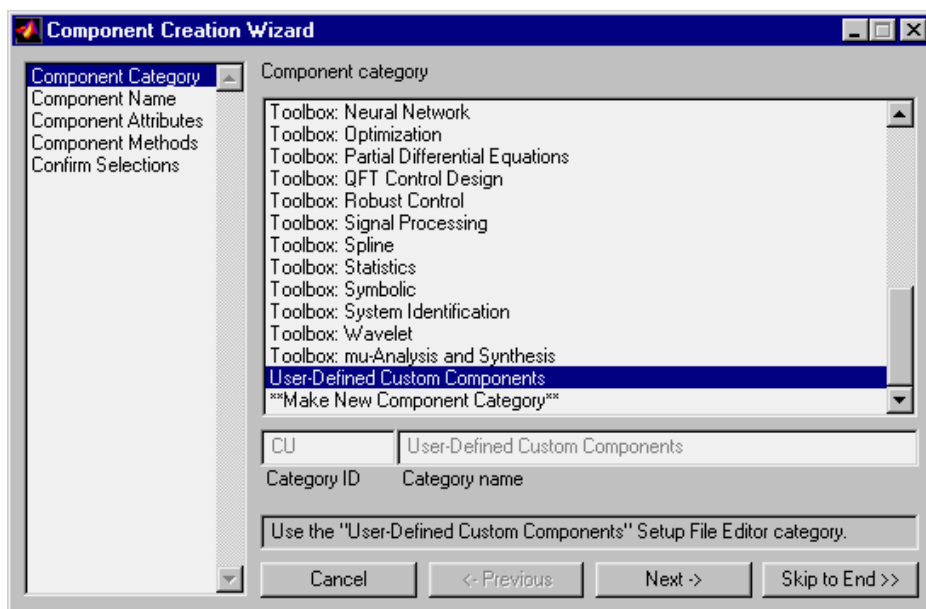
This chapter discusses these options in detail, and uses a tutorial to guide you through the process of creating a new component. In this tutorial, you will create a component that will produce a plot and insert the plot into a report.

You will use the Component Wizard to create the skeleton component, which consists of a directory and M-files or methods. Then you will edit the component's M-files to complete the component.

## Entering Component Category Information

When you create a component, you have to decide the category of components to which it will belong. This is the category in which the new component is displayed in the **Add Component** tab of the Setup File Editor. You can select from existing categories, such as Formatting, Handle Graphics, etc., or you can create a new category of components.

Click on **Component Category** in the list box on the left to get to the Component Category page. You will enter the component category information on this page.



### Choosing the Component Category

The **Component category** list box on the right shows all the available component categories. To select an existing component category, select that component category from the list. The **Category ID** and **Category name** fields display the two-letter code and the name of the category of component you selected, respectively.

If you choose **\*\*Make New Component Category\*\***, you can choose a new two-letter code and description (**Category ID** and **Category name**).

### Tutorial Task

This step creates a new component category.

From the bottom of the **Component Category** list box, select

**\*\*Make New Component Category\*\***

### Specifying the Component Category ID

The **Category ID** is a unique, two-letter code. Any component in this group will have this code. If the code already exists, a warning is displayed.

### Tutorial Task

In the **Category ID** field, enter

RP

(for Random Plotting).

### Specifying the Component Category Name

The **Category name** is the name of the new component category. This is the component category that will appear in the Setup File Editor.

### Tutorial Task

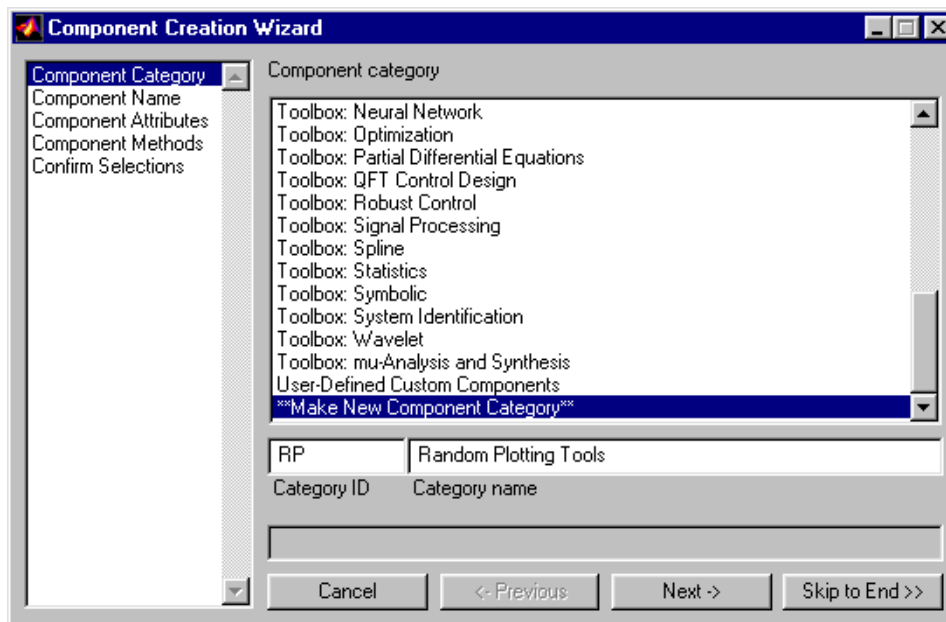
In the **Category name** field, enter

Random Plotting Tools



## Looking at the New Component Category Page

After you have completed the tutorial tasks, the Component Category page will have the following appearance.



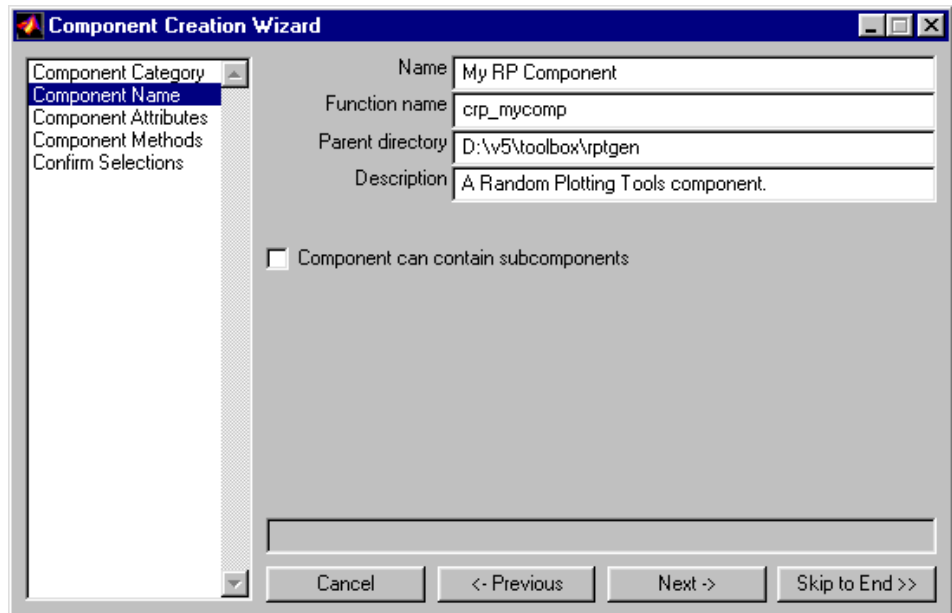
## Creating the Component Name

When you create a component, you have to decide:

- What to name the component
- What function name to give the component
- Where to put the directory that contains all the M-files for the component
- What description to give the component
- Whether the component can have subcomponents

You enter this information on the Component Name page of the Component Creation Wizard.

Click on **Component Name** in the list box on the left to get to the Component Name page. Or, if you have just completed the Component Category page, click on the **Next ->** button.



The screenshot shows the 'Component Creation Wizard' dialog box. On the left is a list box with five items: 'Component Category', 'Component Name' (which is selected and highlighted in blue), 'Component Attributes', 'Component Methods', and 'Confirm Selections'. To the right of the list box are four text input fields: 'Name' with the value 'My RP Component', 'Function name' with 'crp\_mycomp', 'Parent directory' with 'D:\v5\toolbox\rptgen', and 'Description' with 'A Random Plotting Tools component.'. Below these fields is a checkbox labeled 'Component can contain subcomponents' which is currently unchecked. At the bottom of the dialog are four buttons: 'Cancel', '<- Previous', 'Next ->', and 'Skip to End >>'.

## Naming the Component

You need to specify a name for the component you are creating. The name you choose will appear in the **Add Component** tab of the Setup File Editor.

### Tutorial Task

In the **Name** field, enter

Figure Random Plot

## Specifying the Component Function Name

By convention, the component function name should have the format `cxx_<function name>`. The first letter of the component function name should always be `c`. The second and third letters are from the two-letter code that you entered in the **New category ID** field in the Component category page. In this case, the letters are `rp`.

The function name should be simple, but descriptive. The function name should be made up of valid characters (for a MATLAB function); for example, the function name should not contain an ampersand (&). Any invalid characters are changed to an underscore (`_`).

After you create the new component, you can get an instance of the object or component by typing the component function name at the command line.

### Tutorial Task

In the **Component Function Name** field, enter

`crp_randplot`

## Specifying the Parent Directory

When you create the component, methods or M-files are created for the component. These M-files are located in a directory called `@cxx_<FunctionName>`, which is located in the directory that you specify in the **Parent directory** field.

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

### Tutorial Task

Enter the appropriate directory for the **Parent Directory** field.

What you enter for the parent directory depends on where you have MATLAB installed. For example, if your copy of MATLAB is in `c:\matlab`, you can select `c:\matlab` or `c:\matlab\<directory name>` as the parent directory. All the M-files created for the new component will then be placed in `c:\matlab\@cxx_filename` or `c:\matlab\<directory name>\@cxx_filename`, respectively.

### Specifying the 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 at the bottom of the **Add Components** tab in the Setup File Editor.

### Tutorial Task

In the **Description** field, enter

This component creates a random figure and inserts it into the report.

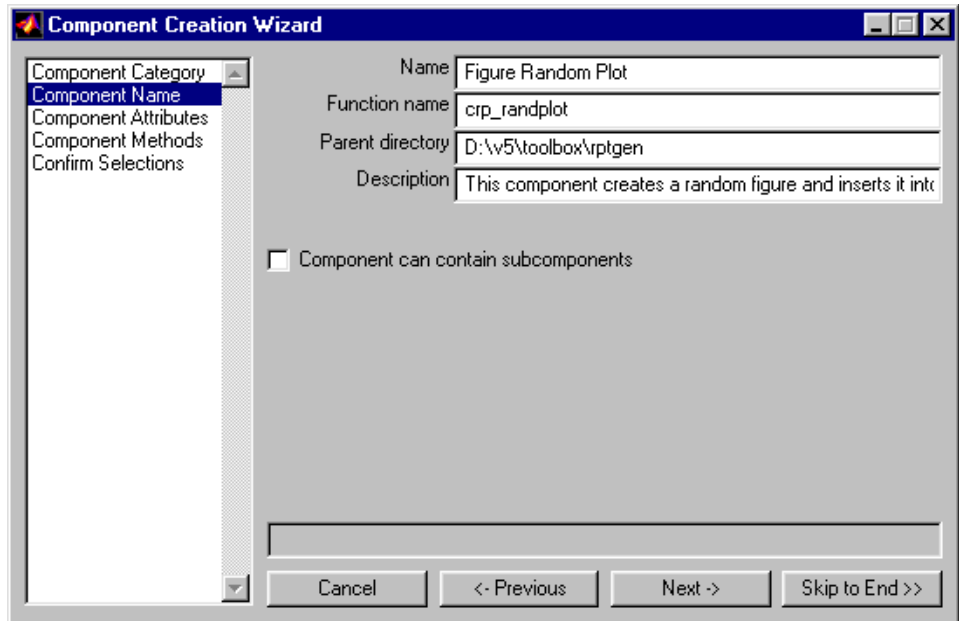
### Subcomponents

If you select the **Component can contain subcomponents** option, then the component you are creating can have subcomponents or children.

For the tutorial, do not select the **Component can contain subcomponents** option.

## Looking at the New Component Name Page

After you have completed the tutorial tasks, the Component Name page will have the following appearance.



The screenshot shows the 'Component Creation Wizard' dialog box. On the left is a vertical list of steps: 'Component Category', 'Component Name' (which is highlighted in blue), 'Component Attributes', 'Component Methods', and 'Confirm Selections'. The main area of the dialog contains the following fields and options:

- Name:** Figure Random Plot
- Function name:** crp\_randplot
- Parent directory:** D:\v5\toolbox\lrptgen
- Description:** This component creates a random figure and inserts it into
- ☐ Component can contain subcomponents

At the bottom of the dialog are four buttons: 'Cancel', '< Previous', 'Next >', and 'Skip to End >>'.

## Creating Component Attributes

Component attributes determine how a component behaves and what information it inserts into the report. You can see component attributes in the **Options** tab of the Setup File Editor.

You create an attribute page for the new component on the Component Attributes page of the Component Creation Wizard.

Click on **Component Attributes** in the list box on the left to get to the Component Attribute page. Or, if you have just completed the Component Name page, click on **Next** ->.



### Component Attribute Tutorial Task 1

For the first component attribute tutorial task, you will create an attribute that lets you put a title on your plot. This attribute will have an editable text field. The default title will be **My Random Plot**. This attribute should appear on the attribute page with the name: **Title of created plot**.

## Specifying the Field Name

The field name is the name of the attribute. It must be a valid variable name (e.g., no spaces, etc.).

### Tutorial Task

In the **Fieldname** field, enter

`Plot_Ti t l e`

Note that the field name must be a valid variable name, so if you enter `Pl ot Ti t l e`, it changes to `Pl ot_Ti t l e` as soon as you press the **Enter** key.

## Specifying the Data Type

The data type is the type of data that is used by the attribute. For more information, see “Programming and Data Types” in the MATLAB documentation.

The **Data type** drop box contains the following options:

- **Logical T/F**
- **Enumerated List**
- **Number**
- **Character String**
- **Cell Array**
- **Other (struct, object)**

### Tutorial Task

In the **Data type** drop box, select

**Character String**

## Setting the Default Value

You set the value that will be displayed on the attribute page as a default value for this attribute.

### Tutorial Task

In the **Default value** field, enter

`' My Random Plot '`

## Setting the Control Type

The control type is the type of control you want to use to represent and change the attribute.

To see what each type of control looks like, type `gui de` at the command line to get the Guide Control Panel. The icons for each control has a tooltip, and you can see what a control looks like by selecting it and bringing it into the palette area. Or, you can see the help for the `ui control` function for information about each kind of control.

The available options for **Control type** depend on the **Data type** that you chose earlier. The table below shows the available data types for each control type.

Control Type	Logial T/F	Enum. List	Number	Char. String	Cell Array	Other
<b>checkbox</b>	yes	yes				
<b>togglebutton</b>	yes	yes				
<b>radiobutton</b>	yes	yes				
<b>popupmenu</b>	yes	yes				
<b>listbox</b>	yes					
<b>edit</b>			yes	yes	yes	yes
<b>multiedit</b>				yes	yes	
<b>slider</b>			yes			



**Tutorial Task**

In the **Control type** drop box, select

**edit**

**Specifying the Attribute Name**

The attribute name appears in the **Options** tab of the Setup File Editor (in the component attribute page). The attribute name should be short (two to four words).

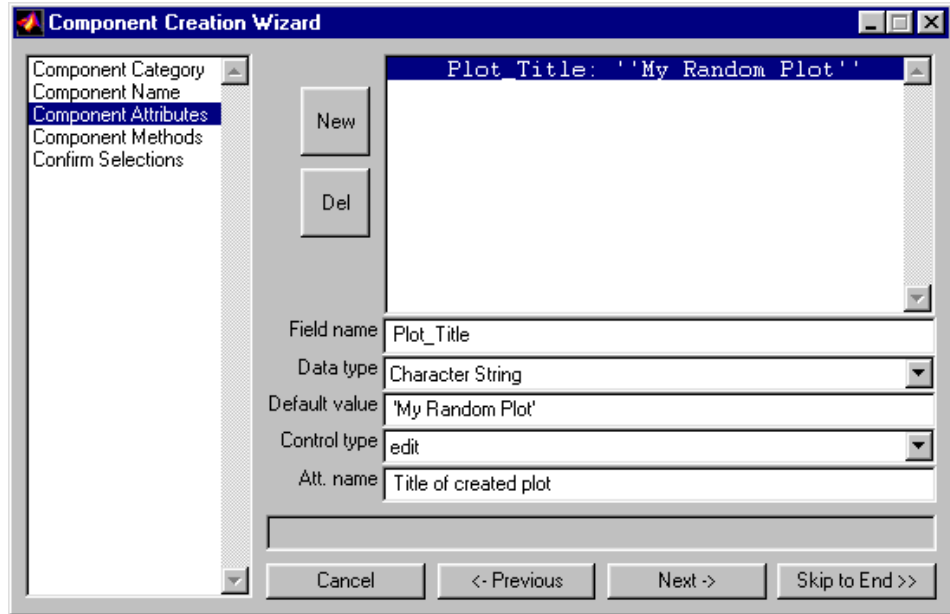
**Tutorial Task**

In the **Att. name** field, enter

Title of created plot

## Looking at the New Component Attributes Page

After you have completed the tutorial tasks, the Component Attributes page will have the following appearance.



## Creating Another Attribute

To create another attribute, click on the **New** button. To delete an attribute, select it and click on the **Del** button.

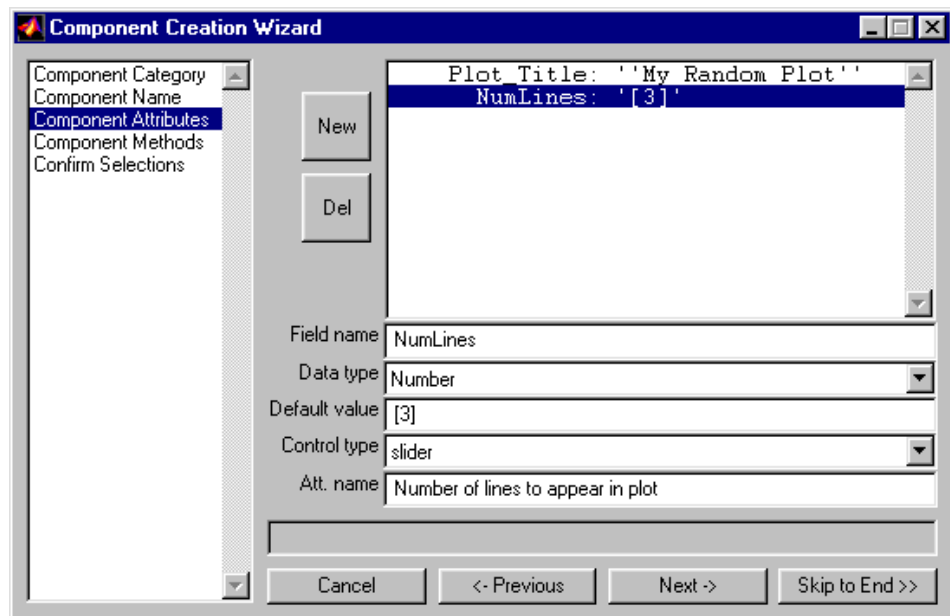
## Component Attribute Tutorial Task 2

This task creates another attribute. This attribute lets you control the number of lines that appear in the plot. The default number will be 3. The UI control for this attribute will be a numerical slider. This attribute will appear on the attribute page with the name: **Number of lines to appear in plot.**

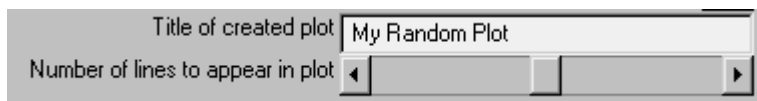
First click on the **New** button and then enter the following parameters:

- 1 **Fieldname:** NumLi nes
- 2 **Data type:** Number
- 3 **Default value:** [3]
- 4 **Control type:** slider
- 5 **Att. name:** Number of lines to appear in plot

After you have completed the tutorial tasks, the Component Attributes page will have the following appearance.



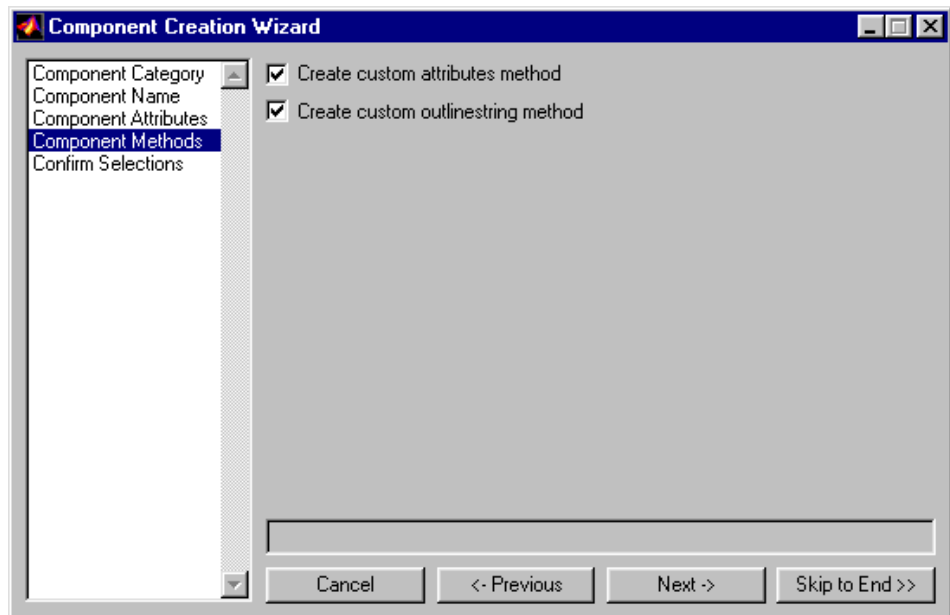
These are the attributes you just created.



The image shows a form with two attributes. The first attribute is labeled "Title of created plot" and has a text input field containing the text "My Random Plot". The second attribute is labeled "Number of lines to appear in plot" and has a numeric input field with a value of 1, indicated by a small square in the center of the field. The form has a light gray background and a thin border.

## Creating Component Methods

Click on **Component Methods** in the list box on the left to get to the Component Methods page. Or, if you have just completed the Component Attributes page, click on **Next** ->.



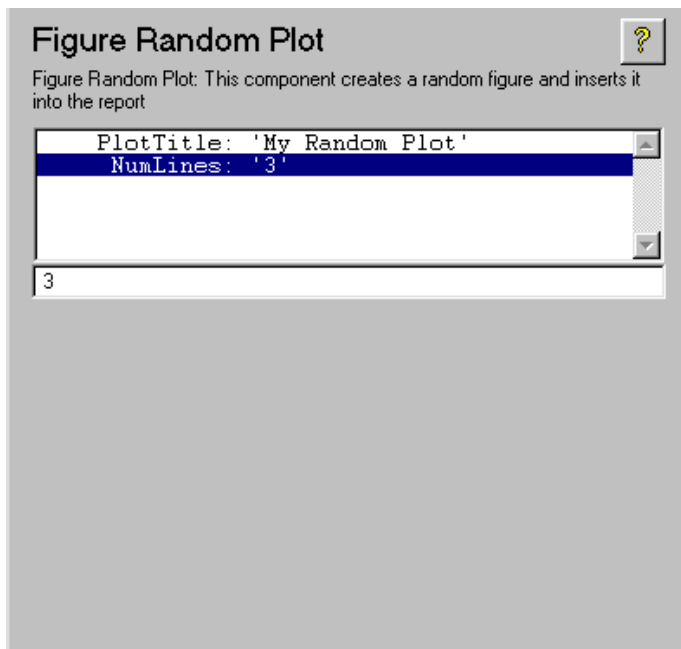
### Creating a Custom Attributes Method

The **Create custom attributes method** option creates a method called `attribute.m` that controls the creation of an attribute page.

If you do not select this option, the new component inherits an attribute page from `rptcomponent`, which is the component parent class. The `attribute.m` method is not created for the component, so you cannot customize the attributes page.

See the MATLAB documentation for a discussion of class, object, and method.

If you *do not* select the **Create custom attributes method** option, then this is the inherited attribute page.



The screenshot shows a configuration window titled "Figure Random Plot" with a help icon in the top right corner. Below the title is a description: "Figure Random Plot: This component creates a random figure and inserts it into the report". The main area contains a list box with two items: "PlotTitle: 'My Random Plot'" and "NumLines: '3'", with the second item selected. Below the list box is a text field containing the value "3".

---

**Note** If you are modifying an existing component, you should *not* select this option unless you want to overwrite the existing attributes method.

---

If you *do* select the **Create custom attributes method** option, then this is the attribute page.

### Tutorial Task

Select the **Create custom attributes method** option for the tutorial.

## Creating a Custom Outlinestring Method

The **Create custom outlinestring method** option creates a method called `outlinestring.m`. This method inserts a single-line string representation of the component in the setup file outline when the component is added to it.

The default `outlinestring` method gets the component string from the `getinfo` method. The string is taken from the **Name** field in the Component Name page of the Component Creation Wizard.

If you select this option, the `outlinestring.m` method is created for the component, and you can edit the outline representation of the component in the

`outlinestring.m` method. This is discussed later, in “Changing the Outline String.”

If you do not select this option, the new component inherits the `outlinestring` method from `rptcomponent`; the `outlinestring.m` method is not created for the component. The string that is displayed in the setup file outline is still taken from the **Name** field in the Component Name page of the Component Creation Wizard, but you cannot change the way the component is shown in the outline.

---

**Note** If you are modifying an existing component that has an existing `outlinestring` method, you should *not* select this option. Otherwise, you will overwrite the existing `outlinestring` method.

---

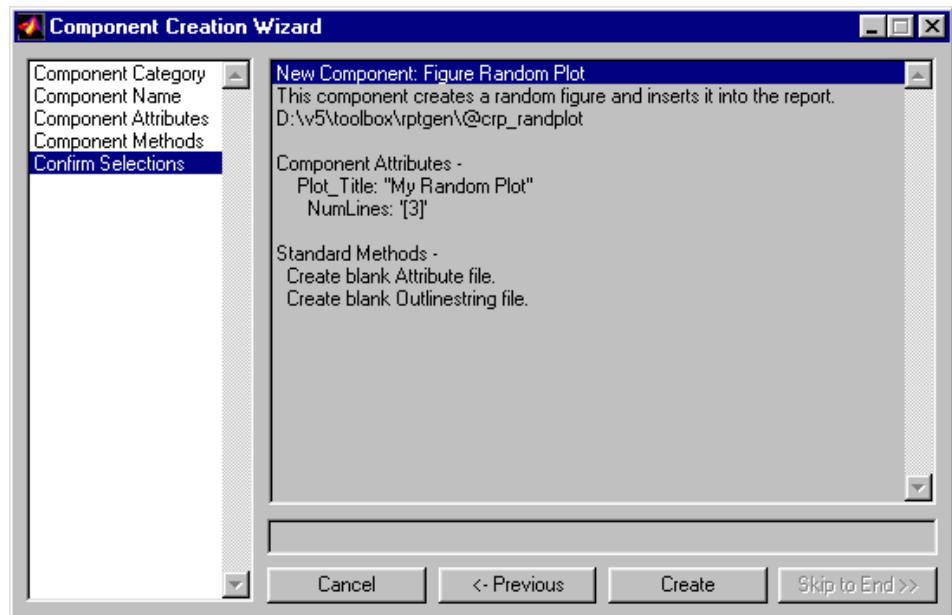
### Tutorial Task

Select the **Create custom outlinestring method** option for the tutorial (you will edit the `outlinestring` method in a tutorial task later).



## Reviewing All Component Information

To review all component information, click on **Confirm Selections** in the list box on the left to get to the Confirm Selections page. Or, if you have just completed the Component Methods, click on **Next** ->.



To create the component, click on **Create**.

## Creating and Verifying the Component

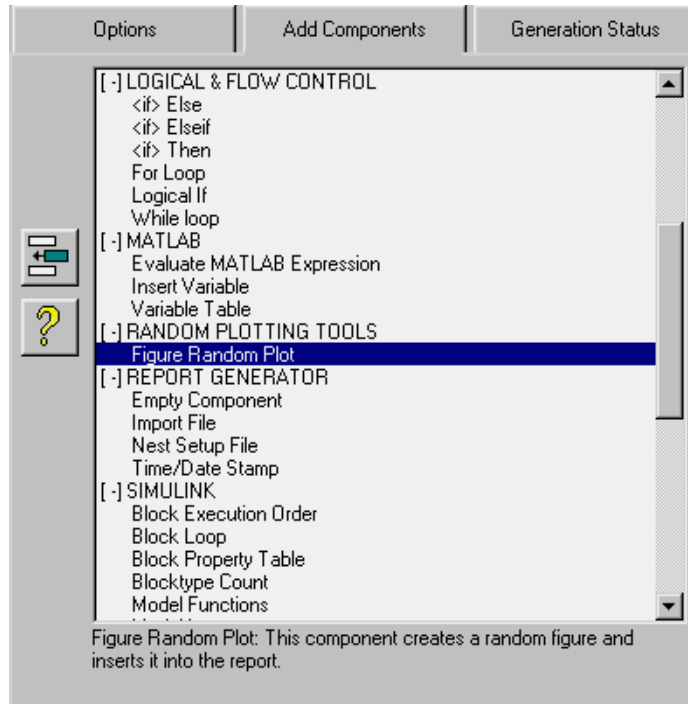
In this section, you will create the component and then verify that it was created.

### Creating the Component

Click on **Create** to create the new component.

## Verifying Component Creation

To verify that your new component has been created, open the Setup File Editor if it is not already open and click on the **Add Components** tab. You will see the a new component category, Random Plotting Tools, and a new component in that category, Figure Random Plot.



## Tutorial: Editing and Running the New Component

You have now completed the part of the tutorial that uses the Component Creation Wizard.

For the rest of this tutorial, you will be editing M-files that are produced when you create the component. The table below shows the M-files you will be editing.

Edit This M-File	To Complete This Task
<code>execute.m</code>	Incorporate new attributes and insert a snapshot of the figure into the report
<code>getinfo.m</code>	<ul style="list-style-type: none"><li>• Add new attributes</li><li>• Change an existing attribute</li></ul>
<code>outlinestring.m</code>	Change the string that appears in the setup file outline in the Setup File Editor

---

**Note** The only M-file that you *must* edit is `execute.m`, which runs the component when the report is generated. In general, editing `getinfo.m` and `outlinestring.m` is optional.

---

In general, to edit these M-files, you must have some knowledge of MATLAB. To edit these M-files for the tutorial, you should have some knowledge of Handle Graphics as well.

### Tutorial Activities

In this tutorial, you will work through examples relating to:

- “Editing a New Component” on page 5-26
- “Editing the Execute Method” on page 5-27
- “Editing `execute.m` to Insert Text Output into the Report” on page 5-28
- “Using the New Component” on page 5-29

- “Creating a New Attribute” on page 5-29
- “Changing the Outline String” on page 5-34
- “Running the New Component” on page 5-35
- “Editing execute.m to Insert a Figure into the Report” on page 5-36

You can see the results in the section called “The Report Created by the Figure Random Plot Component” on page 5-39.

Editing a New Component

When you create a component, the Report Generator creates an @<compfilename> directory; in this case the created directory is @crprandplot. The M-files in this directory are methods; they are intended to be used by this object only. The following are M-files in @crp\_randplot.

M-File	What It Does
crp_randplot.m	This M-file is the constructor; it creates the object. You can change the help in this file.
attribute.m	This M-file displays the attribute page.
outlinestring.m	This M-file returns the string that represents the component in the setup file outline when a component is added to it.
subasgn.m	This M-file accesses fields of object outside its method. Do not edit this file.
subsref.m	This M-file accesses fields of object outside its method. Do not edit this file.

M-File	What It Does
<code>getinfo.m</code>	This M-file is used to store information about components - it tells the Report Generator how to create a component. You can edit this file to create a new attribute.
<code>execute.m</code>	This M-file is the method that is called during generation time. It inserts into the report the information that is created by the component when the report is generated.

Method, object, and constructor are object-oriented programming concepts that are discussed in the MATLAB documentation.

## Editing the Execute Method

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. The default `execute.m` file created by the Component Wizard inserts nothing into the report.

The `execute.m` file for the new component contains the following lines of code.

```
function out=execute(c)
%EXECUTE returns a report element during generation
out=sgml tag;
```

In order for your `execute.m` method to insert meaningful information into the report, you must modify this M-file.

The single output argument of the `execute.m` method is inserted into the report. If the output is a string or number, the component acts like a text component and inserts the output directly into the report. It is possible to create a more sophisticated output by running another component within your `execute` method.

The following tutorial tasks show you how to use both output forms:

- “Editing `execute.m` to Insert Text Output into the Report” shows you how to edit `execute.m` to use the component to insert a string or number into the report

- “Editing `execute.m` to Insert a Figure into the Report” shows you how to edit `execute.m` to run the Graphics Figure Snapshot component within the Figure Random Plot component to insert the snapshot of the figure into the report.

## Editing `execute.m` to Insert Text Output into the Report

Edit the `execute.m` file to create a figure, plot random lines in the figure's axes, and insert the handle to the figure (text output) into the report.

To complete this task, replace the default lines of code with the following lines of code (comments are optional).

```
%EXECUTE returns a report element during generation
function out=execute(c)

%Create the figure.
figHandle=figure('Name', c.att.Plot_Title);

%Create the axes and make them a child of the figure.
%Set the axes "Color" property to be the value of
%the AxesColor attribute.
axHandle=axes('Parent', figHandle);

%This creates the data to be plotted
%and displays it in the axes. The
%number of lines to be plotted is
%defined by the NumLines property.
plotData=rand(10,floor(c.att.NumLines));
plot(plotData,'Parent',axHandle);

%By setting 'out' to the figHandle variable,
%we will insert the figure handle into the report.
out=figHandle

%Clean up by deleting the figure
delete(figHandle)
```

## Using the New Component

After editing the `execute.m` method, you are ready to use your component in a report. Add the component to a setup file and click on the **Report** button. If there is an error during generation, the Report Generator will skip over the component.

Create a setup file that looks like this.

```
[ - ] Report - rplot-test.rpt
[ - ] Paragraph - <Text from subcomponents>
[ ] Text - Random Plotted Figure has handle=
[ ] Figure Random Plot
```

Your report will look like this (it will contain the following line).

```
Random Plotted Figure has handle=1
```

Note that the handle number may be different depending on how many figure windows you had open at the time of generation.

You now have a functioning component. The next section discusses how to add new attributes to the component and change the `execute.m` method so that it inserts a snapshot of the random plot figure into the report.

## Creating a New Attribute

You can create a new attribute by editing `getinfo.m`. This methods file contains information on component attributes and how they are displayed in the component attribute page (shown in the **Options** tab of the Setup File Editor).

All attribute information for a component is taken from `out.att.XXX` and `out.attx.XXX` lines in `getinfo.m`. The `out.att.XXX` lines are attributes and the `out.attx.XXX` lines are UI options. The `out.attx.XXX` lines are optional; they do not need to be included in `getinfo.m`. If you do not include them, default UI controls are used for the attribute.

## Finding Attributes and Their Default Values

Find the following section in `getinfo.m`.

```
%----- ATTRIBUTES -----
%The out.att.XXX section sets attribute defaults.
out.att.Plot_Title = 'My Random Plot';
out.att.NumLines = [3];
```

This section shows that there are two attributes.

- `Plot_Title`, which has a default value of 'My Random Plot'
- `NumLines`, which has a default value of 3

---

**Note** `Plot_Title` and `NumLines` were specified in the **Fieldname** field in the Component Attributes page of the Component Creation Wizard. The default values were specified in the **Default value** field in the Component Attributes page.

---

## Adding a New Attribute and Default Value

Create another attribute called `iVisibleAxes`. This attribute determines whether the figure axes are visible. The attribute will have a default value of logical 1 (axes on).

To complete this task, add the following line to this section.

```
out.att.iVisibleAxes = logical(1);
```

## Specifying Attribute Name, Data Type and UI Control

To specify attribute name, data type, and UI Control, (corresponds to **Att. name**, **Data type**, and **Control type** fields in the Component Attributes page), find the following section in `getinfo.m`.

```
%----- ATTRIBUTE DISPLAY -----
%The out.attx.XXX section sets attribute GUI information.
%Each .attx structure has the following fields:
% .String - Appears as a text field next to the UIcontrol
% .Type - data type of the corresponding attribute.
%         STRING - character string
%         NUMBER - scalar or vector number
```



```

%      LOGICAL - boolean logical (1) /logical (0)
%      ENUM - enumerated list of STRING, NUMBER, or LOGICAL
%      CELL - cell array
%      OTHER - structure or object
%           note: "OTHER" has no automated uicontrol updating
% .enumValues - options for an enumerated list (.Type='ENUM')
% .enumNames - display representation of .enumValues
%           note: must be same length as .enumValues
%           note: empty enumNames implies display from enumValues
% .UIcontrol - type of control to use in GUI
% .numberRange - min and max values (.Type-'NUMBER')

out.attx.Plot_Title.String='Title of created plot';
out.attx.Plot_Title.Type='STRING';
out.attx.Plot_Title.UIcontrol='edit';

out.attx.NumLines.String='Number of lines to appear in plot';
out.attx.NumLines.Type='NUMBER';
out.attx.NumLines.UIcontrol='slider';
out.attx.NumLines.numberRange=[inf inf];

```

**Attribute Name.** out.attx.i sVi si bl eAxes.String is the name of the attribute as it appears in the Setup File Editor (corresponding to the **Att. name** field of the Component Attributes page in the Component Creation Wizard).

To create the attribute name for `i sV i s i b l e A x e s` called 'Make figure axes visible', add the following line to this section.

```
out. attx. i sV i s i b l e A x e s. S t r i n g = ' M a k e   f i g u r e   a x e s   v i s i b l e ' ;
```

**Attribute Data Type.** `out. attx. i sV i s i b l e A x e s. T y p e` is the data type of the attribute (corresponds to the **Data type** field of the Component Attributes page in the Component Creation Wizard).

To specify the data type for `i sV i s i b l e A x e s` to be logical, add the following line to this section.

```
out. attx. i sV i s i b l e A x e s. T y p e = ' L O G I C A L ' ;
```

LOGICAL sets the data type to be a logical or Boolean number.

---

**Note** If you do not set the type, the Report Generator will infer the type from the default value you supplied earlier.

---

**Attribute UI Control.** `out. attx. i sV i s i b l e A x e s. U I c o n t r o l` is the type of UI control for the attribute (corresponding to the **Control type** option of the Component Attributes page in the Component Creation Wizard).

To specify the UI control for `i sV i s i b l e A x e s` to be a check box, add the following line to this section.

```
out. attx. i sV i s i b l e A x e s. U I c o n t r o l = ' c h e c k b o x ' ;
```

### Creating Another Attribute

Create another attribute called `A x e s C o l o r`, which lets the user choose one of four colors for the axes: white, green, red and blue. The default color will be white. The name of the attribute, as it appears in the Setup File Editor, will be `C o l o r o f A x e s`.

To do this task, add the following lines of code to `getinfo.m` (comments are preceded by `%` and are optional).

```
% This creates an attribute called AxesColor, which has a default
% color of white (1 1 1).
out.att.AxesColor=[1 1 1];

% This creates a name for the attribute, which appears in the
% attribute page in the Options tab.
out.attx.AxesColor.String='Color of axes';

% This creates an enumerated list with color choices of
% white (1 1 1), green (0 1 0), red (1 0 0), and blue (0 0 1).
out.attx.AxesColor.enumValues = {[1 1 1] [0 1 0] [1 0 0] [0 0 1]};

% This creates the names for the entries in the enumerated
% list: white, green, red and blue.)
out.attx.AxesColor.enumNames = {'white' 'green' 'red' 'blue'};
```

Note that `out.attx.AxesColor.UIControl` and `out.attx.AxesColor.Type` are not specified here. A default UI control and data type are used, according to the default value.

### Changing a Previously Created Attribute

You can change a field for any attribute that you create by editing the `getinfo.m`.

The slider created for `NumLines` has a range of `-inf` to `inf`.

```
out.attx.NumLines.numberRange=[-inf inf];
```

A slider by default has a range of: `[-inf +inf]`; it guesses about the range. Change the slider range to `[1 10]` (minimum and maximum number of lines).

## Changing the Outline String

The `outlinestring.m` method creates a string that is a one-line description of the component in the setup file outline in the left-hand side of the Setup File Editor.

`outlinestring.m` contains the following lines of code.

```
function strout=outlinestring(c)
% OUTLINESTRING display short component description
% STR=OUTLINESTRING(C) Returns a terse description of the
% component in the setup file editor report outline. The
% default outlinestring method returns the component's name.
info=getinfo(c);
```

To change the outline string, add the following line of code in `outlinestring.m`.

```
strout=[info.Name ' - ' c.att.Plot_Title];
```

This takes the first part of the outline string from `out.Name` in `getinfo.m`, adds a dash, and the second part from `out.att.Plot_Title`, which is `My Random Plot`. The string representing the new component in the setup file outline now has following appearance.

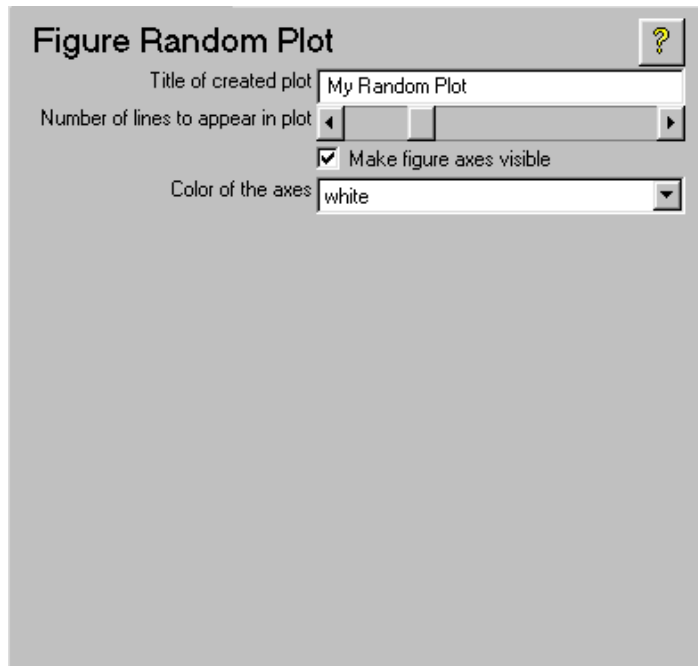
```
Figure Random Plot - My Random Plot
```

## Running the New Component

To run the new component, you must edit `execute.m`.

First, save all your changes to `getinfo.m` and `outlinestring.m`. You can use the Setup File Editor to view and check the attributes of your new component.

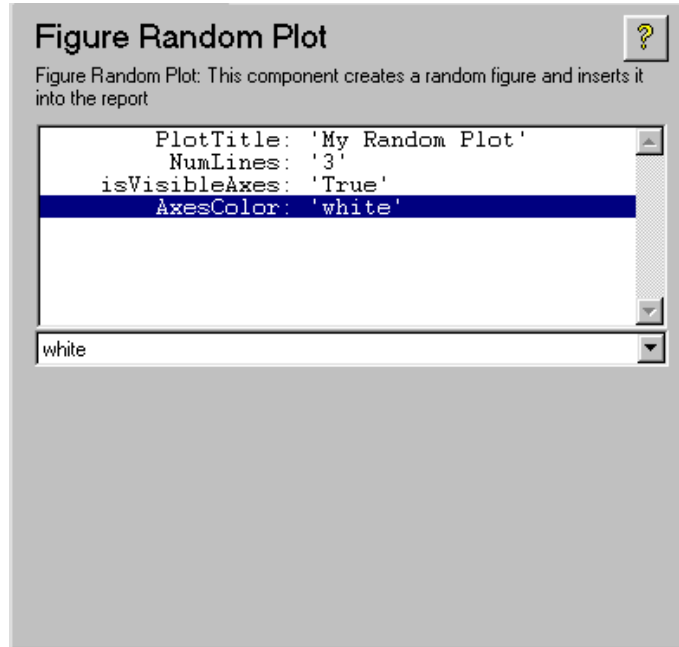
This is the attribute page.



The image shows a dialog box titled "Figure Random Plot" with a help icon (question mark) in the top right corner. The dialog contains the following controls:

- "Title of created plot": A text field containing "My Random Plot".
- "Number of lines to appear in plot": A numeric spinner box with arrows on either side.
- "Make figure axes visible": A checked checkbox.
- "Color of the axes": A dropdown menu currently showing "white".

If you did not select the **Create custom attributes method** option on the Component Methods page, then you get the inherited attribute page, shown below.



### Editing execute.m to Insert a Figure into the Report

Open the `execute.m` file, which runs the component when the report is generated. Edit the file so that the component uses the new `isVisibleAxes` and `AxesColor` attributes that you just created. The new code will also insert a snapshot of the random plot figure into the report instead of returning the figure handle as text.

To complete this task, replace the old lines of code with the new lines of code shown below.

```
function out=execute(c)
%EXECUTE returns a report element during generation

%Create the figure.
figHandle=figure('Name', c.att.PlotTitle);

%Create the axes and make them a child of the figure
axHandle=axes('Parent', figHandle);

%Turn axes visibility on or off depending
%on the value of the isVisibleAxes attribute
if c.att.isVisibleAxes
    axis on;
else
    axis off;
end

%This creates the data to be plotted
%and displays it in the axes. The
%number of lines to be plotted is
%defined by the NumLines property.
plotData=rand(10,floor(c.att.NumLines));
plot(plotData,'Parent', axHandle);

%Set the axes "Color" property to be the value of
%the AxesColor attribute.
set(axHandle,'Color', c.att.AxesColor);

%Now what we have created the figure, we want
%to display it in the report.

%Create an HG Figure Snapshot component
snapComp=c.rptcomponent.comps.chgfigsnap;

%Set the title of the resulting snapshot
%to be the Plot_Title attribute.
snapComp.att.ImageTitle=c.att.Plot_Title;
```

```
%Since the random plot figure is the current
%figure, it will be the one captured by the
%snapshot component. This command runs the
%snapshot component and returns the DocBook
%<figure> tag. The second argument to
%runcomponent shows the priority of the
%"Running Component" message in the Generation
%Status tab.
out=runcomponent (snapComp, 6);
```

```
%Clean up by deleting the figure
delete(figHandle)
```

Note the second argument to runcomponent in the line

```
out=runcomponent (snapComp, 6);
```

sets the message priority level to 6. If you set the Generation Status message priority level in the Setup File Editor to be less than 6, then no messages will be generated for this component as it is executing. See “Setting the Generation Status Update Priority Level” on page 2-20 for a discussion on the message priority level.



## The Report Created by the Figure Random Plot Component

To create a simple report with the Figure Random Plot Component, create a setup file that looks like this. This setup file uses the modified component to create a report. This report will include a paragraph of description text and a picture of a randomly plotted figure.

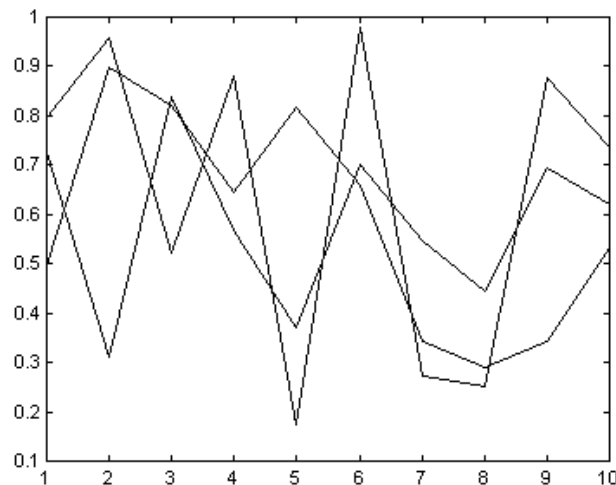
To do this task, create the following setup file.

```
[ - ] Report - new-rp-test.rpt
  [ - ] Paragraph - This is a Random Plotted Figure
    [ ] Figure Random Plot - Random Lines
```

Click on the **Report** button. The following report is generated.

This is a Random Plotted Figure

**Figure 0. Random Lines**



### Summary

In this chapter, you completed a tutorial that demonstrated how to create a new component and its attributes with the Component Creation Wizard. You then learned how to edit certain method files (`getinfo.m`, `outlinestring.m`, and `execute.m`) to:

- Create new attributes
- Edit existing attributes
- Change the string that appears in the setup file outline in the Setup File Editor
- Determine how the component behaves during execution and what it inserts into the report

# Component Reference

---

<b>Introduction</b> . . . . .	6-2
What's In This Chapter . . . . .	6-2
<b>Formatting Components</b> . . . . .	6-5
<b>Handle Graphics Components</b> . . . . .	6-33
<b>Logical and Flow Components</b> . . . . .	6-55
<b>MATLAB Components</b> . . . . .	6-74
<b>Report Generator Components</b> . . . . .	6-86
<b>Simulink Components</b> . . . . .	6-103
<b>Stateflow Components</b> . . . . .	6-190
<b>Fixed-Point Components</b> . . . . .	6-216

## Introduction

The MATLAB Report Generator package contains the following types of components:

- “Formatting Components” on page 6-5
- “Handle Graphics Components” on page 6-34
- “Logical and Flow Components” on page 6-56
- “MATLAB Components” on page 6-75
- “Report Generator Components” on page 6-87
- “Simulink Components” on page 6-104
- “Stateflow Components” on page 6-192
- “Fixed-Point Components” on page 6-218

The Simulink Report Generator package contains Simulink and Stateflow components as well. The Simulink Report Generator is built atop the MATLAB Generator.

If you have the Fixed-Point Blockset, you also have Fixed-Point components.

## What’s In This Chapter

This is a reference chapter that discusses:

- Different categories of components
- Parent/child organization for each component category
- Each component and its attributes (shown on the component attribute page)

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File”, for a discussion on how to add a component to the outline.

---

Component	Report Options
Category	none
Description	<p>The Report Options component is the top-level component in the setup file. This component lets you specify the following output options:</p> <ul style="list-style-type: none"><li>• The name and location of the report file</li><li>• The report format and style</li><li>• Whether the setup file is saved automatically upon report generation</li><li>• Whether the report will be displayed</li><li>• Whether images will be regenerated</li><li>• A description of the report</li></ul> <p>The Report Options component is <i>always</i> the first component in the setup file and cannot be moved, deactivated, added, or deleted.</p>

# Report Options

---

## Attributes

This is the Report Options attribute page.

Report Options

Report File Location

D:\Work\r12\toolbox\rptgenext\simulink-default.html

Directory: 

☒ Same as setup file (D:\Work\r12\toolbox\rptgenext)

☐ Present working directory (h:\documents\rptgen\ScreenCaptu

☐ Other

Filename: 

☒ Same as setup file (simulink-default)

☐ Other




☐ If report already exists, increment to prevent overwriting

Report Format & Stylesheet

web (HTML)

Single-Page Web

Generation Options



Edit...

Description

A report

---

**Note** To view the Report Options attribute page in the Setup File Editor, select the Report Options component in the setup file outline on the left and click on the **Options** tab.

---

This component is discussed in detail in Chapter 2, “Generating a Report.”

Insert anything  
into report?

No.

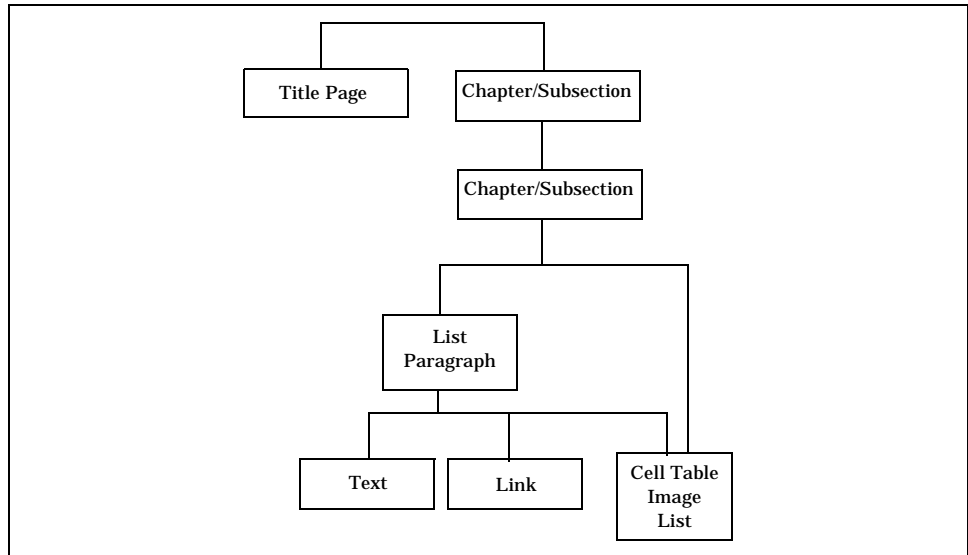
Filename

coutline

## Formatting Components

You can use Formatting components to insert basic documentation elements in your report.

Formatting components have the following parent/child relationships.



**Cell Table.** This component converts a cell array into a table in the report. The Cell Table component can have either the Chapter/Subsection or Paragraph component as its parent and cannot have children.

**Chapter/Subsection.** This component 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 takes paragraph-level children unless you check the **Take section title from the first subcomponent?** check box. In this case, a Chapter/Subsection component's first child must be a Text component.

# Formatting Components

---

- 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.** This component 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** drop box, then the Link component can also have the Chapter/Subsection component as its parent.

**List.** This component inserts a bulleted or numbered list into the report. The List component must have the Chapter/Subsection or Paragraph component as its parent.

**Paragraph.** This component inserts a paragraph into the report. The Paragraph component must have the Chapter/Subsection component as its parent. The Paragraph component can have subcomponents, or it can have text. Text must be entered in the **Paragraph Text** parameter field.

**Text.** This component inserts text into the report. The Text component should have the Paragraph component as its parent unless you check the **Take section title from the first subcomponent?** check box in the Chapter/Subsection component; in this case, the Text component can have the Chapter/Subsection component as its parent.

**Title Page.** This component 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.



---

**Note** The components discussed above are general classes of components; there are other components that act like the List, Paragraph, Text, Link, Image and Cell Table components. For example, the Block Execution Order component behaves like the List component, the Graphics Figure Snapshot component behaves like the Image component, the Simulink property table components behave like the Cell Table component, etc.

---

# Cell Table

---

<b>Component</b>	Cell Table
<b>Category</b>	Formatting
<b>Description</b>	The Cell Table component converts a rectangular cell array into a table and inserts the table into the report.
<b>Attributes</b>	This is the Cell Table attribute page.

---

### Cell Table

Create table from MxN cell array with workspace variable name

Formatting Options

Table title

Cell alignment

Left

☒ Table grid lines

☒ Table spans page width

☐ Rotate table 90 degrees

Header/Footer Options

Number of header rows

1

☒ No footer

☐ Footer is the same as header

☐ Last N rows are footer

1

---

**Note** To view the component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

---

**Create table from MxN cell array with workspace variable name**

Enter the name of the cell array that will be used to create the cell table.

**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. See “Report Format and Stylesheets” on page 2-15 to find stylesheets that have a List of Tables.

**Cell alignment**

Select a cell alignment option (left, center, right or double justified).

**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**

Select this option if you want the table to have the same width as the page on which it appears.

**Rotate table 90 degrees**

Select this option to rotate the table 90 degrees.

**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 top, 2 = two rows from the top, etc.

**No footer**

If you do not want a footer, select **No footer**.

**Footer is the same as header**

Select this option to create a footer that is identical to the header.

**Last N rows of cell array are footer**

If you want to select a footer that is different from your header, then select **Last N rows of cell array are footer** and enter the number of footer rows (from the bottom) in the corresponding field: 0 = none, 1 = one row from 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] }
```

then the resulting cell table has the following appearance.

foo	bar
3	5

Note that in this example, the table has no headers or footers and no title. The table does have grid lines.

**Insert anything into report?** Yes. Table.

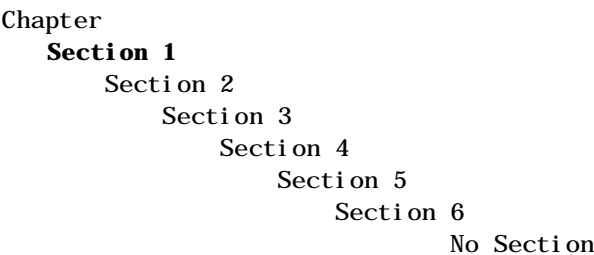
**Filename** cfrcelltable

# Chapter/Subsection

---

Component	Chapter/Subsection
Category	Formatting
Description	<p>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:</p> <ul style="list-style-type: none"><li>• Children of the Chapter/Subsection component are placed inside the section created by this component.</li><li>• The Chapter/Subsection component takes paragraph-level children unless you check the <b>Take section title from the first subcomponent?</b> check box. In this case, a Chapter/Subsection component's first child must be a Text component.</li><li>• The Chapter/Subsection component can be the child of another Chapter/Subsection component.</li></ul>

Sections can be nested and titles will automatically become smaller inside sub-sections. There are seven levels of nesting possible.



Any Chapter/Subsection components beyond the sixth section do not create a titled section, although the subcomponents 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 “Report Format and Stylesheets” on page 2-15 for chapter and section numbering options in Web and Print stylesheets.

## Attributes

This is the Chapter/Subsection attribute page.

The screenshot shows a dialog box titled "Chapter/Subsection". It has two main sections. The first section is labeled "Section title" and contains two radio buttons: "Get title from first subcomponent" (which is unselected) and "Specify title:" (which is selected). Next to the "Specify title:" radio button is a text input field containing the text "Section Title". The second section is labeled "Section type" and contains a list of options: "Chapter", "Section 1", "Section 2", "Section 3", "Section 4", "Section 5", "Section 6", and "No section". The "Chapter" option is selected.

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

### Section Title

#### **Get title from first subcomponent**

Select this option if you want to take the title of the section of the first subcomponent. If you choose this option, the title is taken from the first subcomponent and section content is drawn from the second subcomponent until the end of the subcomponent list. The first subcomponent should be a Text component.

#### **Specify title**

## Chapter/Subsection

---

If you do not want to get the title from the first subcomponent, then enter the title in this field. You can enter %<VariableName> if you want to insert the value of a variable from the MATLAB workspace. Please see “%<VariableName> Notation” in the Text component reference page.

### Section Type

The item in boldface shows you what level the selected section is residing in.

For example, if you have the following in the setup file outline

```
Chapter/Subsection
  Paragraph
  Text
  Chapter/Subsection
    Paragraph
    Text
```

And you select the second Chapter/Subsection component, the Section Type hierarchy field will look like this.

```
Chapter
  Section 1
    Section 2
      Section 3
        Section 4
          Section 5
            Simple Section
              No Section
```

Section 1 appears in boldface to indicate the nesting level of the selected Chapter/Subsection component.

**Insert anything into report?** Yes. Chapter or section.

**Filename** cfrsection



<b>Component</b>	Image
<b>Category</b>	Formatting
<b>Description</b>	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, then the <b>Insert as inline image</b> check box must be checked.
<b>Attributes</b>	This is the Image attribute page.

**Image**

Image Filename:  ...

☐ Copy to local report files directory

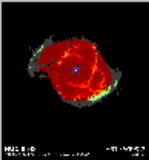
Image Title:

☒ No title ☐ Insert as inline image

☐ Title:

☐ Title from filename

Image Preview:



**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

## Image Filename

Enter the image filename in this field or use the browse button (...) to find the file that contains the image. You can choose to copy the image to the local report files directory by selecting that option.

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 Setup File Editor. See “Report Format and Stylesheets” on page 2-15 for a discussion on report format options.

You can use the %<VariableName> notation. See “%<VariableName> Notation” in the Text component reference.

## Image Title

Choose an title option in this section:

- **No title:** Select this option if you do not want an image title.
- **Title:** Select this option and type in the image title in the corresponding field if you want an image title that is not the same as the image file name.
- **Title from filename:** Select this option if you want the image title to be the same as the image filename.

## Insert as inline image

Select this option if you want to insert an inline image in a paragraph. This option is available only when the image does not have a title, i.e., if you select the **No title** option in the **Image Title** section. To make this a valid option, the Image component must have the Paragraph component as its parent and Text components as its peers.

Example: If the following components are included in a setup file,

```
paragraph
  text - "This is an example of an inline image."
  text - "This is an example."
  image - *image.jpg*
  text - "This is an example of an inline image."
```

and you select the **Insert as inline image** option, the report would contain the following paragraph.

This is an example of an inline image. This is an example. \*image.jpg\* This is an example of an inline image.

If you do not select the **Insert as inline image** option, then the report would contain the following paragraph (notice that the image is not inline).

This is an example of an inline image. This is an example.  
\*image.jpg\*  
This is an example of an inline image.

## Copy to local report files directory

Select this option to copy the image to the local report files directory. This option creates a relative link from the report to the copied image file. You should use this option:

- If you generate your report in SGML format and then want to produce the report in multiple formats
- If you intend to move your report from one directory to another

The image file is in the same directory as the report. You specify this in the **Directory:** option in the Report Options page in the Setup File Editor. See “Specifying the Directory for the Report File” on page 2-14 for a discussion of the **Directory:** option.

## Image Preview

The **Image Preview** frame 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 on the image in the **Image Preview** frame, it is displayed in full size on the screen.

**Insert anything into report?** Yes. Image.

**Filename** cfrimage

# Link

---

Component	Link
Category	Formatting
Description	The Link component inserts linking anchors or pointers into the report.
Attributes	This is the Link attribute page.

Link

Link type

☐

Anchor

☒

Link

☐

External link (Web URL)

Anchor ID to reference

Link text (required for link type "Link")

☐

Emphasize link text

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Link type

Select the type of link to insert into the report:

- **Anchor:** An anchor is the object to which something links; i.e., links “link” to an anchor.
- **Link:** A link takes you to the location in your report (specified by an anchor) to which it points when you click on it.
- **External link (Web URL):** 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 on it.

## Anchor ID

The anchor ID options are context-sensitive and change depending on the link type selected above. The anchor ID marks the location to which the link points. The anchor ID is not visible in the report.

### Anchor ID to create

This option is available when you select **Anchor** as the Link type. Enter the anchor ID. The anchor ID should be an alphanumeric string, e.g., `my_document_mfile3`.

The anchor ID should be unique; if you choose an anchor ID that is already defined in your report, any links that point to this anchor ID may not link to the correct location.

You can use the `%<VariableName>` notation. See “`%<VariableName>` Notation” in the Text component reference.

### Anchor ID to reference

This option is available when you select **Link** as the Link type. Enter the anchor ID to which the link points. Make sure you have another link component that is an anchor with the same ID as you are specifying in this field; otherwise, the link will not point to anything.

### Web URL (`http://`) to link

This option is available when you select **External Link (Web URL)** as the Link type. Enter the URL to which the link points.

For example, to create a link to the Web site for The MathWorks, Inc., enter `http://www.mathworks.com`.

# Link

---

## Link Text

The Link Text options change depending on the link type selected above.

### Link text (optional for link type Anchor)

This option is available when you select **Anchor** as the Link type. The text that you enter in this field appears next to the location to which the link points. i.e., the anchor is located next to this text.

### Link text (required for link type Link)

This option is available when you select **Link** as the Link type. Enter the text that serves as the link. When you click on this text, you will go to the location in your report to which the link points.

### Link text (required for link type Web URL)

This option is available when you select **External Link (Web URL)** as the Link type. Enter the text that serves as the link. When you click on this text, you will go to the Web location to which the link points.

### Emphasize link text

Select this option to *italicize* the link text.

Insert anything  
into report?

Yes. Text.

Filename

cf`link`

Component	List
Category	Formatting
Description	The List component creates a bulleted or numbered list from a cell array or from subcomponents.
Attributes	This is the List attribute page.

List

Create list from cell array with workspace variable name:

Formatting Options

List title

Spacing

Normal

List style

Bulleted list

Numeration Options

Numbering style

1,2,3,4,...

Show parent number in nested list (1.1.a)

Show only current list value (a)

Continue numbering from previous list

Always start at 1

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

6-21

## Create list from cell array with workspace variable name

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 subcomponents, since the list is then automatically generated from the subcomponents.

## Formatting Options

### List title

Enter the title of the list in this field.

### Spacing

Select either Compact spacing or Normal spacing.

### List style

Select either Bulleted list or Numbered list. If you choose Numbered list, then you can select numbering options in the **Numeration Options** section.

## Numeration Options

### Numbering Style

Select a numbering style in this pull-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, . . .



**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 numbering style chosen in **Numeration Options**).

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 numbering style chosen in **Numeration Options**).

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.

**Continue numbering from previous list**

Select this option to continue list numbering from a previous list. Note that this option is not available if you select the **Always start at 1** option.

## Always start at 1

Select this option to always start list numbering at 1. Note that this option is not available if you select the **Continue numbering from previous list** option.

## Example1: 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 setup file, with one List component nested inside another List component.

```
[ - ] Report - Unnamed.rpt
  [ - ] Bulleted list from subcomponents
    [ ] Text - sky
    [ ] Table - varname
    [ ] Image - test.jpg
    [ ] Text - grass
  [ - ] Bulleted list from subcomponents
    [ ] 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 Subcomponents and a Cell Array**

Setup file with subcomponents:

```
[ - ] Report - Unnamed.rpt
    [ - ] Bulleted list from subcomponents
        [ ] Text - red
        [ ] Text - green
        [ ] Text - blue
```

Setup file calling a cell array:

```
[ - ] Report - Unnamed.rpt
    [ - ] Bulleted list from cell array called colors
```

Cell array:

```
colors={' red' , ' green' , ' blue' }
```

Using either of these two setup files, the report contains the following bulleted list:

- red
- green
- blue

**Insert anything into report?** Yes. List.

**Filename** cfrlist

# Paragraph

---

Component	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 <b>Paragraph Text</b> field.
Attributes	This is the Paragraph attribute page.

Paragraph

Title Options

☒ No paragraph title

☐ Get title from first subcomponent

☐ Specify title:

Paragraph Text

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Title Options

### No paragraph title

Select this option if you do not want the paragraph to have a title.

### Get title from first subcomponent

Select this option if you want the title of the paragraph to be taken from the first subcomponent. You must select the **Paragraph has a title** check box to use this check box. The first subcomponent should be a Text component.

### Specify 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 subcomponent.

## Paragraph Text

Enter paragraph text in this field. If the Paragraph component has subcomponents, then the paragraph content is taken from the subcomponents. Otherwise, the Paragraph component inserts text from this field.

If the Paragraph component does not have any subcomponents and you do not enter any text in this field, then the Paragraph component will not insert anything into the report.

Note that you can use the %<VariableName> notation in this field if you want to insert the value of a variable from the MATLAB workspace. Please see “%<VariableName> Notation” in the Text component reference page

## Insert anything into report?

Yes. Can be text, an image, etc.; depends on subcomponents.

## Filename

cfrparagraph

# Text

---

Component	Text
Category	Formatting
Description	The Text component inserts and formats text into your report. The text component should have the Paragraph component as its parent.
Attributes	This is the Text attribute page.

Text

Text to include in report

☐ Emphasize text

☐ Retain spaces and carriage returns

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

**Text to include in report**

Enter the text you want to include in your report in this field.

**%<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 MATLAB workspace. You cannot enter more than one variable in the %<>. If you enter an invalid variable name, then the report will contain %<VariableName> instead of the value of the variable.

**Example 1**

If you enter the following

```
I have a %<Obj Name> and it has %<NumLeaves> leaves. The word  
' %<Obj Name>' has %<size(Obj Name)> letters.
```

and if Obj Name=' plant ' and NumLeaves=3 in the MATLAB workspace, then the report will contain 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' )>
```

then the report will contain the text that you get when you type help peaks at the command line.

**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 will still insert one space any time you have one or more spaces; it will not keep any carriage returns.

# Text

---

**Insert anything  
into report?**      Yes. Text.

**Filename**              cfrtext



Component	Title Page
Category	Formatting
Description	The Title Page component creates and inserts a title page at the beginning of the report.
Attributes	This is the Title Page attribute page.

Title Page

Title

Subtitle

Image

...

Author

☒ Include date report was created

19-May-2000 14:05:09

☒ Include copyright information

Name of copyright holder

copyright year2000

Abstract

Legal notice

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

# Title Page

---

All of the text fields in this attribute page can contain the %<VariableName> notation. See “%<VariableName> Notation” in the Text component reference page.

## Title

Enter the title of the report in this field. The title will appear in large font.

## Subtitle

Enter the subtitle in this field. The subtitle will appear under the title, in a smaller font than the title.

## Image

Enter the filename for an image you want to appear on the title page. The image will appear under the subtitle.

## Author

Enter the author in this field. The author will appear under the subtitle, in a smaller font than the subtitle.

## Include date report was created

Select this option if you want to include the report creation date and choose the date format in the corresponding drop box.

## Include copyright information

Select this option to include copyright holder and year information.

## Name of copyright holder

If you check the **Include copyright information** check box, enter the name of the copyright holder in the **Name of copyright holder** field.

## Copyright year

If you check the **Include copyright information** check box, enter the copyright year in the **copyright Year** field.

For example, if you select **Include copyright information** and enter The MathWorks, Inc. in the **Name of copyright holder** field and enter 2000 in the **copyright year** field, the following will appear in your report.

Copyright © 2000 by The MathWorks, Inc.

If you check the **Include copyright information** check box, and leave the **Name of copyright holder** field and the **copyright year** field empty, then the following will appear in the report.

Copyri ght © by

**Abstract**

Enter the abstract for the report in this field. This is optional.

**Legal notice**

Enter the legal notice for the report in this field. This is optional.

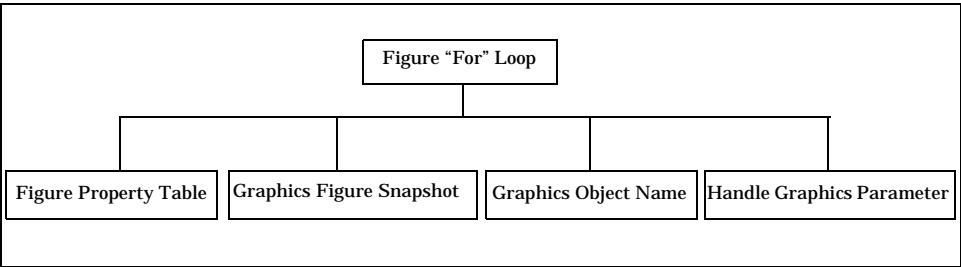
**Insert anything  
into report?** Yes. Title page.

**Filename** cfr\_titlepage

# Handle Graphics Components

## Handle Graphics Components

Handle Graphics components report on Handle Graphics figures. Handle Graphics components have the following parent/child hierarchy.



**Figure “For” Loop.** The Figure “For” Loop component runs its children for each Handle Graphics figure that you select. If a Handle Graphics component does not have the Figure “For” Loop as its parent, then it operates on the current figure.

**Figure Property Table.** This component creates and inserts a property name/property value table for Handle Graphics figures. The Figure Property Table component behaves like the Cell Table component and expects the Graphics Figure “For” Loop component as its parent.

**Graphics Figure Snapshot.** This component inserts an image of a figure window into your report. The Graphics Figure Snapshot component behaves like the Image component and expects the Figure “For” Loop component as its parent.

**Graphics Object Name.** This component inserts the name of a Handle Graphics object into your report. The Graphics Object Name component behaves like the Text component and expects the Figure “For” Loop component as its parent.

**Handle Graphics Parameter.** This component inserts a single property/value pair from a Handle Graphics figure, axis, or any other Handle Graphics object into your report. The Handle Graphics Parameter component behaves like the Text component and expects the Figure “For” Loop component as its parent.

This is an example of how Handle Graphics components can be used in a setup file.

Figure “For” Loop

Section (take title from first subcomponent)

Graphics Object Name

Figure Property Table

Graphics Figure Snapshot

Paragraph

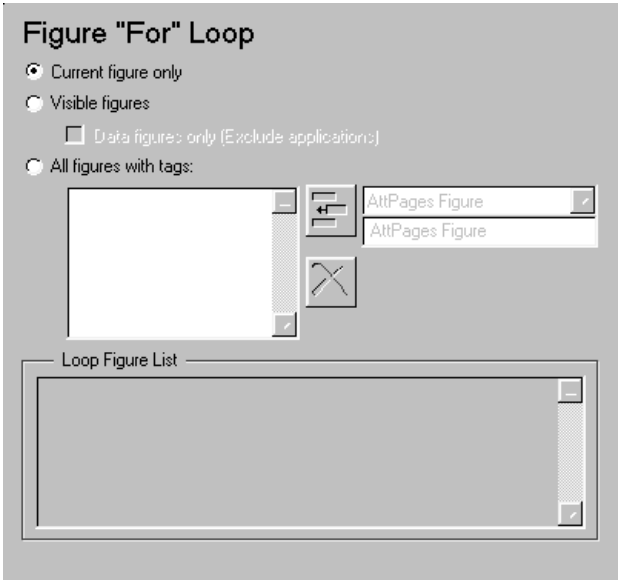
Text ("My Parameter")

Handle Graphics Parameter

# Figure “For” Loop

---

Component	Figure “For” Loop
Category	Handle Graphics
Description	<p>The Figure “For” Loop component applies each subcomponent to graphics figures specified in the attribute page.</p> <p>See “Using the Handle Graphics Figure “For” Loop: A Tutorial” on page 3-3 for a thorough, tutorial-based discussion of this component.</p>
Attributes	This is the Figure “For” Loop attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

### Choosing Figure to Loop On

You can choose to loop on:

- The current figure
- All visible figures
- Figures with certain tags

#### Current figure only

To include only the current figure in the report, select the **Current figure only** option.

#### Visible figures

To loop on all visible figures, select the **Visible figures** option. This option executes subcomponents 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.

#### All figures with tags:

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 drop box on the right and click on the Add button, shown below.



The tag drop box shows a list of tags for figures that are open. If you want to add a tag that does not appear in the tag drop box, you can type in the tag. To type in a tag, select **Enter tag in edit field** and type in the name of the tag in the edit field below the tag drop box.

### Loop Figure List

The **Loop Figure List** shows you what figures are looped on.

# Figure “For” Loop

---

---

**Note** If your setup file generates new figures or changes existing figures, then the figures in the **Loop Figure List** will not be the figures that are actually reported on.

---

To see a figure in the **Loop Figure List**, click on the name of the figure. That figure becomes current and comes to the foreground.

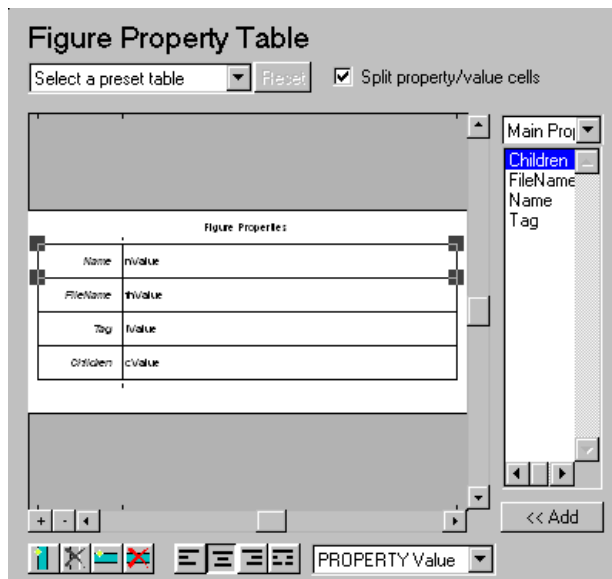
If you click on the name of an invisible figure, the figure becomes visible for a moment.

**Insert anything into report?** No.

**Filename** chgfi gl oop



<b>Component</b>	Figure Property Table
<b>Category</b>	Handle Graphics
<b>Description</b>	<p>The Figure Property Table component inserts a table that reports on property name/property value pairs.</p> <p>See Chapter 3, “Editing Component Attributes” for a tutorial on the Figure Property Table.</p>
<b>Attributes</b>	This is the Figure Property Table attribute page.



**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

# Figure Property Table

---

## Selecting a Preset Table

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-hand corner of the attributes page. To apply a preset table, select the table and click on the **Reset** button. Note that the **Reset** button is only active when you select a preset table.

## Custom Tables

To create a custom table, you can first select a preset table and then edit it. You may want to start with the Blank 4x4 preset table. You can add and delete rows and you can add properties.

If you want to use a custom table in more than one setup file, use Copy Component and Paste Component from the **Edit** menu to copy the custom table from one setup file to another. You cannot save a custom table as a preset table. If you create a table and then select a preset table, all your changes will be lost once you click on the **Reset** button.

## Split Property/Value 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 only be one property name/property value pair in a cell. If you have more than one pair in the cell in the Setup File Editor, then only the first pair is displayed in the report; all subsequent pairs are ignored.

To have the property name and property value appear together in one cell, deselect the **Split Property/Value Cells**. This is the “non-split” mode. In the non-split mode, you can have more than one property name/property value pair and you can also have text.

If you want to switch from the non-split mode to the 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 property name/property value pair or any text, then only the first property name/property value pair appears in the report; subsequent pairs and text are omitted.

Display Options

Each table cell in the report can display a property name/property value pair in one of several ways. To select how a property name/property value pair is displayed in a table cell in the report, select that cell and then choose the one of the following display formats shown in the table below from the drop box at the bottom of the attributes page.

Display Option	"Split" Mode		"Non-Split" Mode
Val ue		Value	Value
		Value	Value
Property Val ue	Property	Value	Property Value
	Property	Value	Property Value
<i>Property</i> Val ue (note. this is shown as PROPERTY val ue on the attributes page)	<i>Property</i>	Value	<i>Property</i> Value
	<i>Property</i>	Value	<i>Property</i> Value
Property: Val ue	Property:	Value	Property: Value
	Property:	Value	Property: Value

# Figure Property Table

Display Option	"Split" Mode		"Non-Split" Mode
Property: Value	Property	Value	Property:Value
	Property	Value	Property:Value
Property - Value	Property	Value	Property - Value
	Property	Value	Property - Value
Property - Value	Property	Value	Property - Value
	Property	Value	Property - Value





You can also right-click on the table to invoke the context menu, select Cell rendering, and then choose a display format from the menu. Note that in the above table, the cells are center-justified. The next section discusses justification options.

## Justification Options

The text in each cell can appear in the report in one of four ways, using the corresponding justification button. These options are shown in the table below (note that the *Property value* display option is shown here).

You can also right click on the table to invoke the context menu, select Cell justification, and select a display format (**Left**, **Center**, **Right**, or **Double justify**) from the menu.

:

Option	Button	"Split" Mode		"Non-Split" Mode
Left-justified		Property	Value	PropertyValue
		Property	Value	PropertyValue
Centered		Property	Value	PropertyValue
		Property	Value	PropertyValue
Right-justified		Property	Value	PropertyValue
		Property	Value	PropertyValue
Double-justified		Property	Value	PropertyValue
		Property	Value	PropertyValue

Putting Text in a Table Cell

You can type text in a table cell in the Setup File Editor. The text appears in the report as long as you are in the non-split mode.

# Figure Property Table

---

To enter text in a cell, double-click on the cell. When a gray box appears in the cell, you are in the edit mode; enter text in the box. To exit the edit mode, press the **Esc** key.

In the edit mode, %<PropertyName> appears in the gray box. If you type text outside the %<>, then it will appear as text in the report. If you type text inside the %<>, it should be a valid property name. If you type an invalid property name, then that property name will appear in the report without a property value.

For example, if you choose to display the property name/property value pair as value, and enter the following in a table cell in the attribute page

My title: %<Name> is visible=%<visible>

and the value of Name is 'Membrane' and the value of visible is 'on', then the following table cell appears in the report if the cell is in the non-split mode.

My title: Membrane is visible=on
----------------------------------

If the table is in the split mode, then the following table cells appear in the report.

	Membrane
--	----------

## Title

The table has a title that you can edit. The title of the table can contain properties and text. To enter text, double click on the title, and enter text in the gray box.

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 “Report Format and Stylesheets” on page 2-15 to find stylesheets that have a List of Tables.

## Adding Properties

To add a property name to a cell in the Setup File Editor, follow these steps:

- 1 Select the desired cell
- 2 Choose the type of property name from the type of property drop box on the right side of the attributes page. The property drop box acts like a filter by listing properties by category.

If you want to choose from all properties associated with a Handle Graphics figure, choose **All properties** in the type of property list box.

- 3 Choose the desired property name from the property list box
- 4 Double-click on the property name or use **Add** button

In the non-split mode, the steps shown above will add a new property to a table cell, in the split mode, they will replace the existing property with the new property.

If you know the name of the property, you can also type the property name in the cell as %<PropertyName>. If you enter a property name that is not valid, then nothing will appear for the corresponding value in the table.

---

**Note** The attributes page in the Setup File Editor does not display text wrapping inside a cell, but when the report is created, the text inside a cell will wrap.

---

## Replacing or Deleting Properties

In the non-split mode, you can replace a property in a table cell in the Setup File Editor by following the instructions in “Putting Text in a Table Cell” on page 6-43. You can delete a property by selecting it and either backspacing over it or pressing the **Delete** key.

## Changing Table and Cell Borders

You can toggle the table and cell borders in the Property Table by using the context menu. To invoke the context menu, place your cursor over the table and click on the right mouse button.





# Figure Property Table

To toggle border of the table on or off, invoke the context menu and select **Table frame**.

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

## Adding and Deleting Columns and Rows

To add or delete a column or row, select a cell, and then click on one of the four buttons shown in the table below.

Action	Button
Add column (added to the right of the selected column)	
Delete column	
Add row (added below the selected row)	
Delete row	

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



## Resizing Columns

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



Zooming and Scrolling

You can zoom in to and out of the table with the zoom buttons, located to the left of the horizontal scroll bar. These are shown in the table below.

Zoom Option	Button
Zoom in	
Zoom out	

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

Example

The table below is an example of a Handle Graphics figure property table. It is the **Graphics** preset table.

<i>BackingStore</i>	on	<i>DoubleBuffer</i>	off
<i>RendererMode</i>	auto	<i>Renderer</i>	zbuffer
<i>MinColormap</i>	64	<i>ShareColors</i>	on
<i>Visible</i>	on	<i>Clipping</i>	on

Insert anything into report?

Yes. Table.

Filename

chgfigproptable

# Graphics Figure Snapshot

---

Component	Graphics Figure Snapshot
Category	Handle Graphics
Description	The Graphics Figure Snapshot component inserts a snapshot of a Handle Graphics figure.
Attributes	This is the Graphics Figure Snapshot attribute page.

### Graphics Figure Snapshot

Image file format Automatic (PNG - Portable Network Graphics 24-bit imag ▾

Print Options

☐ Capture figure from screen

Paper orientation Use figure orientation ▾

Image size

☐ Use figure PaperPositionMode setting

☒ Automatic (same size as on-screen)

☐ Manual: 3.75x3 inches ▾

Invert hardcopy Automatic ▾

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Image file format

Choose the image file format (e.g., JPEG, TIFF, etc.) from this drop box. To automatically choose the format best suited for the output format that you chose in the Report component, select **Automatic**, which is the default option. Otherwise, make sure that you choose an image format that your output viewer can read.

## Print Options

Use the attributes in this section to set printing parameters.

## Capture picture from screen

Use the check box to capture a figure from the screen.

## 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 the **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 **Manual** option to choose a custom image size. If you choose this option, specify the image size in the **Size** field and **Units** drop box.

- **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 **Manual** 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 **Manual** list box.

# Graphics Figure Snapshot

---

## **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.

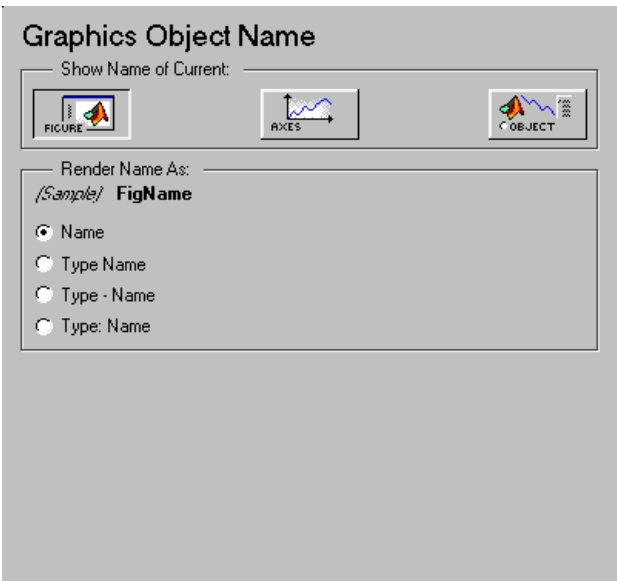
Select one of the following options:

- `Automatic` - Select this option to automatically change dark axes colors to a light axes color. If the axes color is a light color, then 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.

**Insert anything into report?**      Yes. Image.

**Filename**                      `chgfigsnap`

<b>Component</b>	Graphics Object Name
<b>Category</b>	Handle Graphics
<b>Description</b>	<p>The Graphics Object Name component inserts the name of a Handle Graphics object as text into the report.</p> <p>You can use this component to create a section title based on the current figure. First, make sure that this component is the first subcomponent of a Chapter/Subsection component. Then select the <b>Get title from first subcomponent</b> option in the Chapter/Subsection component.</p>
<b>Attributes</b>	This is the Graphics Object Name attribute page.



**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline

first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Show Name of Current:

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
- FileName
- Handle

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:

- ui control
  - String
  - Style
  - Handle
- ui menu
  - Label
  - Accelerator
  - Tag
  - Handle
- ui contextmenu, image, light, line, patch, or surface

- Tag
- Handle
- text
  - String
  - Tag
  - Handle

See the MATLAB graphics documentation for a discussion of the `CurrentObject` property.

**Render Name As:**

Select one of the four options listed below to display the Handle Graphics object name in the report:

- **Name** (example: Name)
- **Type Name** (example: Axes Name)
- **Type - Name** (example: Axes - Name)
- **Type: Name** (example: Axes: Name)

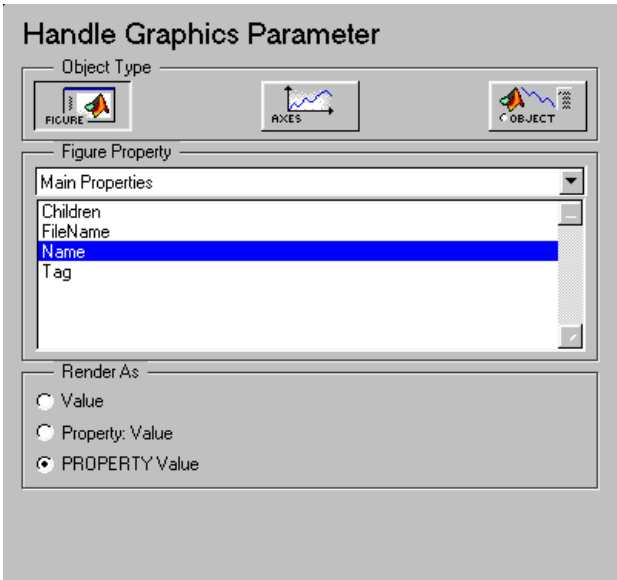
**Insert anything into report?** Yes. Text.

**Filename** chgobj name

# Handle Graphics Parameter

---

Component	Handle Graphics Parameter
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	This is the Handle Graphics Parameter attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---



## Object Type

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 drop box 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.

## Render As

Select one of the three options listed below to control how the Handle Graphics object name displays in the report:

- **Value** - displays the value of the property only (e.g., if the name of the property is Tag and the value of the property is red, only red is displayed)
- **Property: Value** - e.g., Tag: Red
- **PROPERTY Value** - displays the name of the property in italics and the value of the property in regular text (e.g., *Tag* Red)

**Insert anything  
into report?** Yes. Text.

**Filename** chgproperty

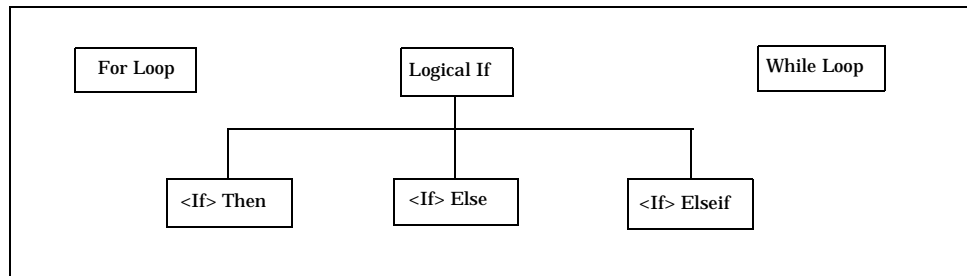
# Logical and Flow Components

---

## Logical and Flow Components

Logical and Flow components execute conditionally, enabling you to decide when a subcomponent will execute or how many times a subcomponent will execute.

Logical and Flow components have the following parent/child relationships.



**For Loop.** This component iteratively runs its child components. 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, then this component does not add anything to the report.

**While Loop.** This component loops while a workspace expression is true. This component runs its subcomponents until its conditional string is true. The number of repetitions can be limited to prevent infinite loops. Similar to the For Loop component, the While Loop component must have at least one child. If it does not have any children, then this component does not add anything to the report.

The “If” Family. There are three possible ways in which you can use the “If” family components.

Components Used	Appearance in the Outline	How the Components Execute
Logical If	i f	When the i f condition is true, this component will run its children.
Logical If, <if> then, <if> else	i f then else	<ul style="list-style-type: none"><li>- When the i f condition is true, only the children of the then component will run and the process will exit the i f family.</li><li>- When the i f condition is false, only the children of the el se component will run.</li></ul>
Logical If, <if> then, <if> elseif, <if> else	i f then el sei f el sei f . . . el se	<ul style="list-style-type: none"><li>- When the i f condition is true, only the children of the then component will run and the process will exit the i f family.</li><li>- When the i f condition is false, the process checks all the el sei f statements in order, and the children of the first true el sei f statement will run. The process will then exit the i f family.</li><li>- If none of the el sei f statements is true, the process runs the children of the el se statement.</li></ul> <p>Note that only one of the conditional statements (then, el se, el sei f) will actually execute.</p>

# <if> Else

---

Component	<if> Else
Category	Logical and Flow
Description	<p>The &lt;if&gt; Else component acts as an el se when it is the child of the Logical If component. There are two ways to use this component</p> <pre>    i f       then     el se</pre> <p>or</p> <pre>    i f       then     el sei f     el sei f       .       .       .     el se</pre>

For an explanation of how the above cases execute, see the introduction to Logical and Flow Components at the beginning of this section.

### Attributes

This is the <if> Else attribute page.



The screenshot shows a gray rectangular area representing the attribute page for the <if> Else component. At the top left, the text "<if> Else" is displayed. Below it, the instruction "If 'else' has no subcomponents, display this string:" is shown. Underneath this instruction is a large, empty white rectangular box with a thin black border, intended for the user to enter a string to be displayed when the component has no subcomponents.

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

**If “else” has no subcomponents, display this string:**

When the <if> Else component has no subcomponents, the string in this field is printed in your report. In this case, this component acts like the Text component.

## <if> Else

---

**Insert anything into report?** Depends on whether `if` or `elseif` statements are true.

**Filename** `elseif`

<b>Component</b>	<if> Elseif
<b>Category</b>	Logical and Flow
<b>Description</b>	<p>The &lt;if&gt; Elseif component acts as an el sei f when it is the child of the Logical If component. You must use this component in this format.</p> <pre>if   then     el sei f     el sei f     .     .     .   el se</pre> <p>For an explanation of how the above case executes, see the introduction to “Logical and Flow Components” on page 6-56.</p>

# <if> Elseif

---

## Attributes

This is the <if> Elseif attribute page.



The screenshot shows a gray rectangular window titled "<if> Elseif". Inside the window, there are two text labels and two input fields. The first label reads "Run subcomponents if this string is true in the base workspace:" and is followed by a single-line text input field. The second label reads "If 'elseif' has no subcomponents, display this string:" and is followed by a larger, multi-line text input field.

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

### **Run subcomponents if this string is true in the base workspace:**

When the statement in this field is true, the <if> Elseif's subcomponents execute.



**If “elseif” has no subcomponents, display this string:**

When the <if> Elseif component has no subcomponents, the string in this field is printed in the report. In this case, this component acts like the Text component.

**Insert anything into report?** Yes, only when parent i f statement is false.

**Filename** cl oel sei f

# For Loop

---

<b>Component</b>	For Loop
<b>Category</b>	Logical and Flow
<b>Description</b>	<p>The For Loop component iteratively executes its subcomponents and must have at least one subcomponent to execute properly. This component functions like the for loop in MATLAB, except that instead of executing a statement, it executes its subcomponents while one of the following loops is active</p> <pre>for varname=x: y: z</pre> <pre>for varname=[a b c ...]</pre> <p>where varname must be a string that is valid as a variable name. x, y, z can be scalar numbers 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 number, then the For Loop will not execute.</p> <p>For example, if red is a workspace variable, the following is a valid for statement.</p> <pre>for i=1: 1: length(red)</pre> <p>length(red) is 3, which is a scalar number.</p>

## Attributes

This is the For Loop attribute page.

The screenshot shows the 'For Loop' attribute page. It has three main sections: 'Loop Type', 'Workspace Variable', and 'Preview'.  
1. 'Loop Type' section: Contains two radio buttons. The first, 'Incremented indices', is selected. Below it are three input fields: 'Start' with value '1', 'Increment' with value '1', and 'End' with value '5'. The second radio button, 'Vector of indices', is unselected. Below it is a 'Vector' input field with the value '1 2 3 4 5'.  
2. 'Workspace Variable' section: Contains a checked checkbox 'Show index value in base workspace'. Below it is a 'Variable name' input field with the value 'RPTGEN\_LOOP'. Below that is another checked checkbox 'Remove variable from workspace when done'.  
3. 'Preview' section: Contains a text area with the following MATLAB code:

```
for RPTGEN_LOOP = 1:1:5
    %run child components
end
```

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Loop Type

In this section you can choose the loop type 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
```

# For Loop

---

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 variable name is varname in the for loop forms described above.

### 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.

## Preview

This section shows a schematic representation of how this component will execute when the report is generated.

**Insert anything into report?**

No.

**Filename**

cl ofor

<b>Component</b>	Logical If
<b>Category</b>	Logical and Flow
<b>Description</b>	The Logical If component acts as a logical if; it can have the <if> Then, <if> Elseif, and <if> Else components as its subcomponents. This component executes its subcomponents when the given workspace expression is true. It displays a specified string when it has no subcomponents. There are three ways to use this component.

```
i f
  then
```

or

```
i f
  then
  el se
```

or

```
i f
  then
  el sei f
  el sei f
  .
  .
  .
  el se
```

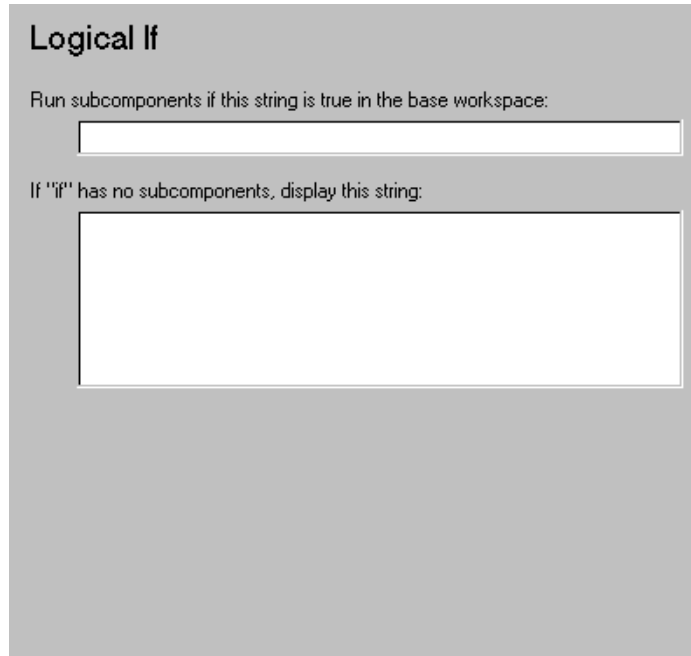
For an explanation of how the above cases execute, see the introduction to “Logical and Flow Components” on page 6-56.

# Logical If

---

## Attributes

This is the Logical If attribute page.



The screenshot shows a gray rectangular window titled "Logical If". Inside the window, there are two text input fields. The first field is preceded by the text "Run subcomponents if this string is true in the base workspace:". The second field is preceded by the text "If 'If' has no subcomponents, display this string:". Both fields are empty and have a thin black border.

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

### **Run subcomponents if this string is true in the base workspace:**

When the statement in this field is true, the Logical If's subcomponents execute.

**If "if" has no subcomponents, display this string:**

When the Logical If component has no subcomponents, the string in this field is printed in your report.

**Insert anything into report?** Depends on attributes.

**Filename** cl oi f

# <if> Then

---

Component	<if> Then
Category	Logical and Flow
Description	The <if> 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

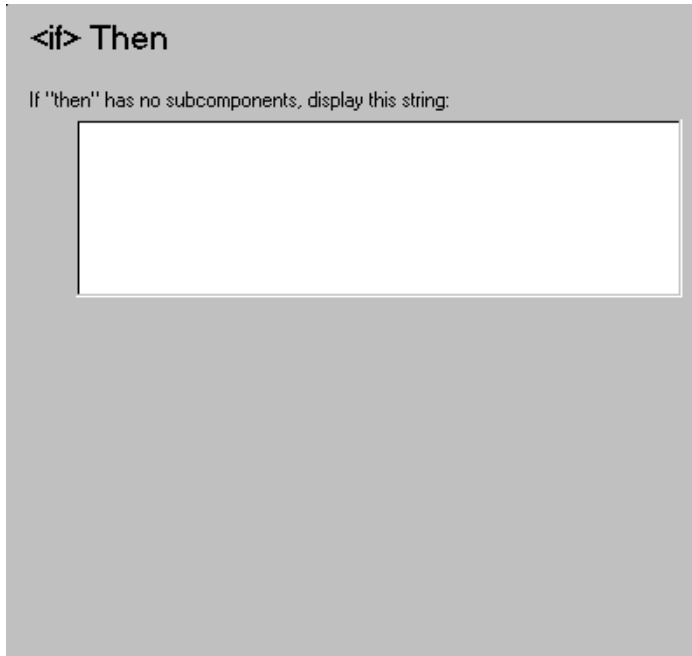
or  
if  
then  
elseif  
elseif  
.  
.  
.  
else

For an explanation of how the above cases execute, see the introduction to “Logical and Flow Components” on page 6-56.



### Attributes

This is the <if> Then attribute page.



The screenshot shows a gray rectangular window titled "<if> Then". Below the title, there is a text label that reads "If 'then' has no subcomponents, display this string:". Underneath this label is a large, empty white rectangular text box for input.

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

#### **If “then” has no subcomponents, display this string:**

When the <if> Then component has no subcomponents, the string in this field is printed in your report. In this case, this component acts like the Text component.

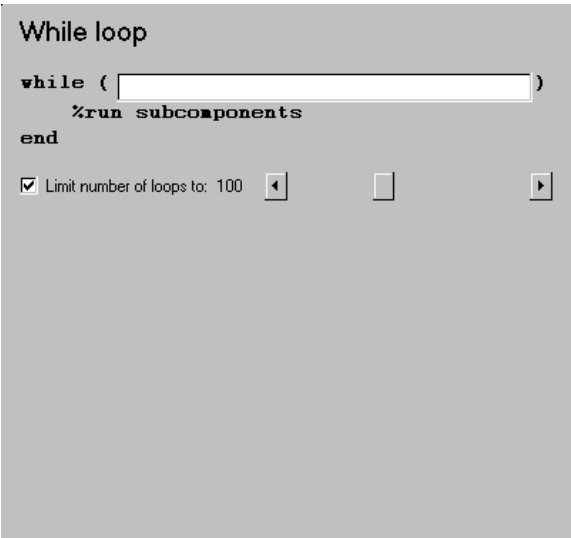
# < if > Then

---

**Insert anything into report?**      Yes, only when parent i f statement is true.

**Filename**              cl othen

Component	While Loop
Category	Logical and Flow
Description	The While Loop component iteratively executes its subcomponents 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 subcomponent; the purpose of this component is to run its subcomponents several times. If it does not have any subcomponents, this component does not add anything to the report.
Attributes	This is the While Loop attribute page.



**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

# While Loop

---

## **while**

Enter the string to be evaluated in this field. This string should be a valid MATLAB expression that evaluates to be a 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
```

Since a (1) is greater than b/c (2/3), this expression is true, and evaluates to be a 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.

**Insert anything into report?** Yes, as long as it has subcomponents.

**Filename** `cl o_w h i l e`

## MATLAB Components

You can use MATLAB components to interact with the workspace. The following are MATLAB components.

**Evaluate MATLAB Expression.** This component evaluates a specified MATLAB expression. The result of the evaluation and/or MATLAB expression (M-code) can appear in the report.

**Insert Variable.** This component inserts a specified variable into the report. If you check the **Represent variable as text only** check box, then this component will act like a Text component, and should be inside a paragraph. If you do not check the **Represent variable as text only** check box, then this component will act like a Paragraph component, Text component, or Cell Table component, depending on the variable.

**MATLAB/Toolbox Version Number.** This component creates a table that shows the version number, release number, and/or release date of any MathWorks software you are currently using.

**Variable Table.** This component creates a table that includes all variables in the MATLAB workspace. The Variable Table component behaves like the Cell Table component.

# Evaluate MATLAB Expression

---

Component	Evaluate MATLAB Expression
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	This is the Evaluate MATLAB Expression attribute page.

Evaluate MATLAB Expression

☐ Insert MATLAB expression string in report

☐ Display command window output in report

Expression to evaluate in the base workspace:

Eval Now

%Evaluate this string in the base workspace

☒ Evaluate this expression if there is an error:

disp(sprintf('Error during eval: %s',  
lasterr))

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

**Insert MATLAB expression string 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.

**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.

**Filename**

cml eval

# Insert Variable

---

**Component** Insert Variable

**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** This is the Insert Variable attribute page.

The screenshot shows the 'Insert Variable' configuration window. It has a title bar 'Insert Variable'. Below the title bar, there are two main sections. The first section is titled 'Variable to Include' and contains three radio buttons: 'Workspace' (which is selected), 'MAT-File' (with an adjacent text input field and a browse button), and 'Global variable'. Below these is a text input field for 'Variable name'. The second section is titled 'Render Options' and contains three radio buttons: 'Value' (selected), 'Name: Value', and 'NAME Value'. Below these are two dropdown menus: 'Render 1xN structures and non-character arrays as tables' and 'Show large arrays as "[MxN CLASS]"'. The window has a standard Mac OS X-style title bar with red, yellow, and green buttons.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

### Variable to Include

#### Variable location

Select one of the following options:

- **Workspace** - this options gets a variable from the MATLAB workspace.
- **MAT-file** - this option gets a variable from a binary file with a .MAT extension.
- **Global variable** - this option gets a global variable.

#### 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, then your report will also contain 50.

# Insert Variable

---

## Render Options

### Name display

Select one of the options listed below to display the variable in the report:

- **Value** - displays the value of the variable only (e.g., if the variable is `velocity`, and its value is 25, only 25 is displayed).
- **Name: Value** - displays the variable name, followed by the value of the variable (e.g., `Velocity: 25`)
- **NAME Value** - displays the variable name in italics, followed by the value of the variable (e.g., *Velocity* 25)

You can choose how the variable appears in the report in the first drop box in this section:

- Choose **Render variable as single-line text only** to display the variable as text. If you select this option, this component will act like the Text component, and should have the Paragraph component as its parent.
- Choose **Render 1xN structures and non-character arrays as tables** to display the variables that are in one of these forms as tables.

You can choose how a variable array appears in the report in the second drop box in this section:

- Choose **Show entire array** to display the entire array in the report.
- Choose **Show large arrays as “[MxN CLASS]”** to display large variable arrays as “[MxN CLASS]”. For example, if you have a 300-by-200 double array, it will display in the report as [300x200 DOUBLE].

Insert anything into report? Yes. Text.

Filename `cml variable`

<b>Component</b>	MATLAB/Toolbox Version Number
<b>Category</b>	MATLAB
<b>Description</b>	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 <code>ver</code> at the command line.
<b>Attributes</b>	This is the MATLAB/Toolbox Version Number attribute page.

The screenshot shows the 'MATLAB/Toolbox Version Number' attribute page. It has two main sections: 'Table Columns' and 'Table Display'. In the 'Table Columns' section, there are three checkboxes: 'Version number' (checked), 'Release number' (unchecked), and 'Release date' (unchecked). In the 'Table Display' section, there is a 'Table title:' text box containing 'Version Number', and two checkboxes: 'Display header row' (checked) and 'Display table border' (checked).

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

# MATLAB/Toolbox Version Number

## Table Columns

This section specifies the information contained in the 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.

## Table Display

This section contains information about the appearance of the table.

### Table title:

Enter the title of the table in this field.

### Display header row

Select this option to display a title for each column in the table. The following table shows the title for each option available in the **Table Columns** section.

Table Column Option	Header Row
Version number	Version
Release number	Release
Release date	Date

### Display table border

Select this option to display a border around the table.

## Example

This is an example of a table generated from a setup file containing the MATLAB/Toolbox Version Number component.

**Table 0. Version Number**

Name	Version	Release	Date
MATLAB Toolbox	5.3	(R11)	31-Dec-1998
Simulink	3.0	(R11)	01-Sep-1998
Stateflow	2.0.1	(R11.1)	05-May-1999
Real-Time Workshop	3.0.0	(R11)	13-Oct-1998
Real Time Windows Target	1.0	(R11)	10-Jul-1998
MATLAB Report Generator	1.1	(R12) Beta1	01-Apr-1999
Simulink Report Generator	1.1	(R12) Beta1	01-Apr-1999

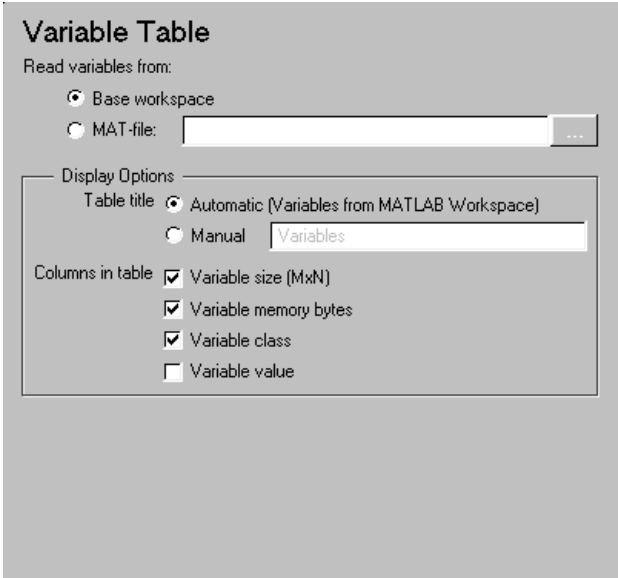
**Insert anything  
into report?** Yes. Table.

**Filename** cml\_ver

# Variable Table

---

Component	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	This is the Variable Table attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## 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. See “%<VariableName> Notation” in the Text component reference

## Display Options

### 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
- **Manual** - 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 “Report Format and Stylesheets” on page 2-15 to find stylesheets that have a List of Tables.

### Columns in table

Select information to be contained in table columns:

- **Variable size (MxN)** - select this option to include the size of the variable.
- **Variable memory bytes** - select this option to include the number of memory bytes 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 will display in the report as [300x200 DOUBLE].

# Variable Table

## Example

This is an example of a variable table that includes variable size, variable memory bytes, and variable value information in the table columns.

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.

Filename

cml whos



## Report Generator Components

Report Generator components are general “utility” components.

**Comment.** This component inserts a comment into the SGML source file created by the report generation process. The comment is not visible in the generated report, but you can edit the SGML source file to make the component appear in the report.

**Empty Component.** This component does not insert anything into the report and can have any component as a child. It can be used to group components together so that the components can be moved or activated/deactivated easily, or it can create a blank space in a list.

**Import File.** This component can do one of the following:

- Import plain text from a file and insert the text into the report.
- Import a Doc Book SGML document from a file and insert it into the report.

**Nest Setup File.** This component inserts a setup file at the point where the Nest Setup File component is listed in the current setup file. The Nest Setup File component allows one setup file (. rpt) to run inside another.

The components of the inserted setup file enter the current setup file 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.

**Time/Date Stamp.** This component inserts the time and date of report generation. The Time/Date Stamp component behaves like the Text component; it must be inside a paragraph or title.

**Stop Report Generation.** This component 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.

# Comment

---

Component	Comment
Category	Report Generator
Description	<p>The Comment component inserts a comment into the SGML source file created by the report generation process. The comment is not visible in the generated report.</p> <p>This component can have children. Child components insert their output into the SGML source file, but this output appears inside comment tags and does not appear in the final report.</p> <p>If you want the comment text to appear in the report, follow these steps:</p> <ol style="list-style-type: none"><li>1 Edit the SGML source file (the SGML source file has the same name as your report file, but has a . sgml extension). Note that you must generate a report to create the SGML source file.</li><li>2 Find the comment area in the SGML source file by locating the comment tags &lt;- - and - - &gt;. Remove both of these tags.</li><li>3 Convert the SGML source file using the rptconvert command (see “Converting a Generated Report Into Another Format” on page 2-21 for more details on the rptconvert command).</li></ol>
Attributes	This is the Comment component attribute page.

Comment

Comment Text

Status Message

☐ Show comment in Generation Status window

Status message priority level 3 Important messages

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Comment Text

Enter the comments that you want to include in the report.

## Status Message

### 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 is displayed during report generation. See “Setting the Generation Status Update Priority Level” on page 2–20 for more information on the message priority level.

## Insert anything into report?

No, but this component inserts a comment into the SGML source file, which can appear in the report.

## Filename

crg\_comment

# Empty Component

---

Component	Empty Component
Category	Report Generator
Description	<p>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.</p> <p>If the Report Generator does not recognize a component when loading a setup file, it replaces the unrecognized component with the Empty Component.</p>
Attributes	This is the Empty Component attribute page.

## Empty Component

Empty Component  
This component does not insert anything into the report and can contain children. It can be used to group components together for easy moving and deactivation. It can also be used as a blank space in a list.

If the Report Generator does not recognize a component when loading a setup file, it will replace the unrecognized component with the empty component. This component will appear in the outline as

```
[.] <Empty> - was "componentname"
```

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

**Insert anything  
into report?**

No.

**Filename**

crgempty

# Import File

---

Component	Import File
Category	Report Generator
Description	The Import File component imports an ASCII text file.
Attributes	This is the Import File attribute page.

Import File

File to Import

...

Import file as

☐ Plain text (ignore line breaks)

☐ Paragraphs defined by line breaks

☐ Paragraphs defined by empty rows

☒ Text (retain line breaks)

☐ Fixed-width text (retain line breaks)

☐ DocBook SGML

File Content

Please specify a file to import

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## File to Import

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 filename in this field.

### Import file as:

- **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.

# Import File

---

- **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):** 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 SGML:** Select this check box if the inserted text file is in an SGML source file. If you select this option, the format of the inserted SGML file is not changed.

## File Content

This 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 on the following options:

- **Paragraphs defined by line breaks**
- **Paragraphs defined by empty rows**

Inserts the contents of the SGML file if you select the **DocBook SGML** option. This can include text, paragraphs, tables, etc.

## Filename

crg\_import\_file

# Nest Setup File

---

Component	Nest Setup File
Category	Report Generator
Description	<p>The Nest Setup File component runs another setup file at the point where the Nest Setup File component is listed in the current setup file. The Nest Setup File component allows one setup file (. rpt) to run inside of another.</p> <p>The components of the inserted setup file enter the current setup file 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.</p>
Attributes	This is the Nest Setup File attribute page.

Nest Setup File

Setup File to Run

...

Run mode

☒ Embed new setup file's report in current report

☐ Create a separate report file

Recursion limit

1

Description of selected setup file

Warning - no setup file specified

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Setup File to Run

Enter the name of the setup file 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 filename in this field.

## Run mode

Select one of the following report options:

- **Embed new setup file’s report in current report** - Select this option to insert the nested report at the place in the original report where this component is located.
- **Create a separate report (insert nothing in current report)** - Select this option to create two separate reports, one with the original setup file and one with the nested setup file. Note that this option has the same result as specifying the following expression in the Evaluate MATLAB Expression component:

report <SetupFileName>

## Recursion limit

If you want to nest a setup file inside itself, you can set a recursion limit in this field. The recursion limit sets a limit on the number of times the setup file can run itself.

## Description of selected setup file

This field contains the text from the **Description** field in the attribute page of the Report component. See “Report Description” on page 2-19 for a discussion of the **Description** field in the Report component.

# Nest Setup File

---

## Example

In the example shown below, the **Embed new setup file's report in current report** option was selected. Setup file R2. rpt is being nested in R1. rpt.

```
[ - ] Report - R1. rpt
[ ] Chapter
  [ - ] B
    [ ] Nest Setfile - R2. rpt
    [ ] C
  [ ] D
[ - ] Report - R2. rpt
[ ] 1
[ ] 2
[ - ] Chapter
[ ] 4
[ ] 5
```

The report will execute as if the original setup file looked like this.

```
[ - ] 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 will be affected by components in the parent setup file (as shown above). See the Chapter/Subsection reference page for more details on the Chapter/Subsection component.

Insert anything into report?	Yes, but only if you select the <b>Embed new setup file's report in current report</b> option.
Filename	crgnestset

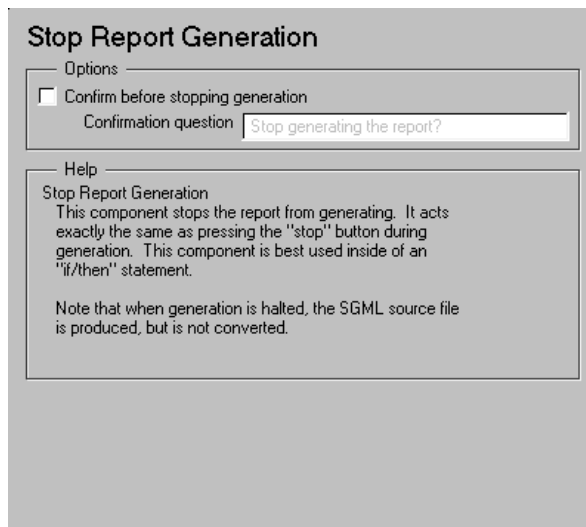
**Component** Stop Report Generation

**Category** Report Generator

**Description** The Stop Report Generation component halts report generation. This component acts like the **Stop** button 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 SGML source file is produced, but not converted.

**Attributes** This is the Stop Report Generation component attribute page.



The screenshot shows a window titled "Stop Report Generation". It has two main sections: "Options" and "Help".

**Options:** Contains a checkbox labeled "Confirm before stopping generation". Below it, a text field labeled "Confirmation question" contains the text "Stop generating the report?".

**Help:** Contains the following text:  
Stop Report Generation  
This component stops the report from generating. It acts exactly the same as pressing the "stop" button during generation. This component is best used inside of an "if/then" statement.  
  
Note that when generation is halted, the SGML source file is produced, but is not converted.

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, "Building and Editing the Setup File" for a discussion on how to add a component to the outline.

---

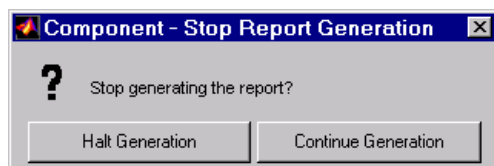
# Stop Report Generation

---

## Options

If you want the Report Generator to prompt you before it halts report generation, select the **Confirm before stopping generation**. If you select this option, the Report Generator opens a dialog box similar to the one shown below.

You can write the question that will appear in the dialog box in the **Confirmation question** field.



## Example

The setup file in this example creates a simple report that takes the 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

**Filename**

crg\_halt\_gen

<b>Component</b>	Time/Date Stamp
<b>Category</b>	Report Generator
<b>Description</b>	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.
<b>Attributes</b>	This is the Time/Date Stamp attribute page.

The screenshot shows the 'Time/Date Stamp' attribute page. It has a title bar 'Time/Date Stamp'. Below it is a checkbox 'Include text before stamp:' which is checked, with a text field containing 'Created'. Below this is a section 'Time Stamp Properties' with a checkbox 'Include current time in stamp' which is checked. It has two dropdowns: 'Time separator' set to 'Colon (:) ' and 'Time display' set to '12-hour'. There is also an unchecked checkbox 'Include seconds in time stamp'. Below this is a section 'Date Stamp Properties' with a checkbox 'Include current date in stamp' which is checked. It has three dropdowns: 'Date order' set to 'Day Month Year', 'Month display' set to 'Long (December)', and 'Date separator' set to 'Blank space ( )'. There is also a 'Year display' dropdown set to 'Long (1998)'. At the bottom is a 'Stamp Preview' section showing the output: 'Created 2:05 pm 19 May 2000'.

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

# Time/Date Stamp

---

## **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 Separator**

Use this option to select a separation marker between hours, minutes and seconds. The following options are available for the separator:

- Bl ank space ( ) - e.g., Hour Mi nute Second
- Col on (:) - e.g., Hour: Mi nute: Second
- Peri od (.) - e.g., Hour. Mi nute. Second
- None () - e.g., HourMi nuteSecond

### **Time display**

Use this option to select the appearance of the time display. The following options are available:

- 12- hour
- 24- hour

### **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 ( ) - e.g., Day Month Year
- Colon (:) - e.g., Day: Month: Year
- Slash (/) - e.g., Day/Month/Year
- Period (.) - e.g., Day. Month. Year
- None () - e.g., 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 (1998)
- Short (98)

## Stamp Preview

This field displays the time/date stamp as it will appear in the report.

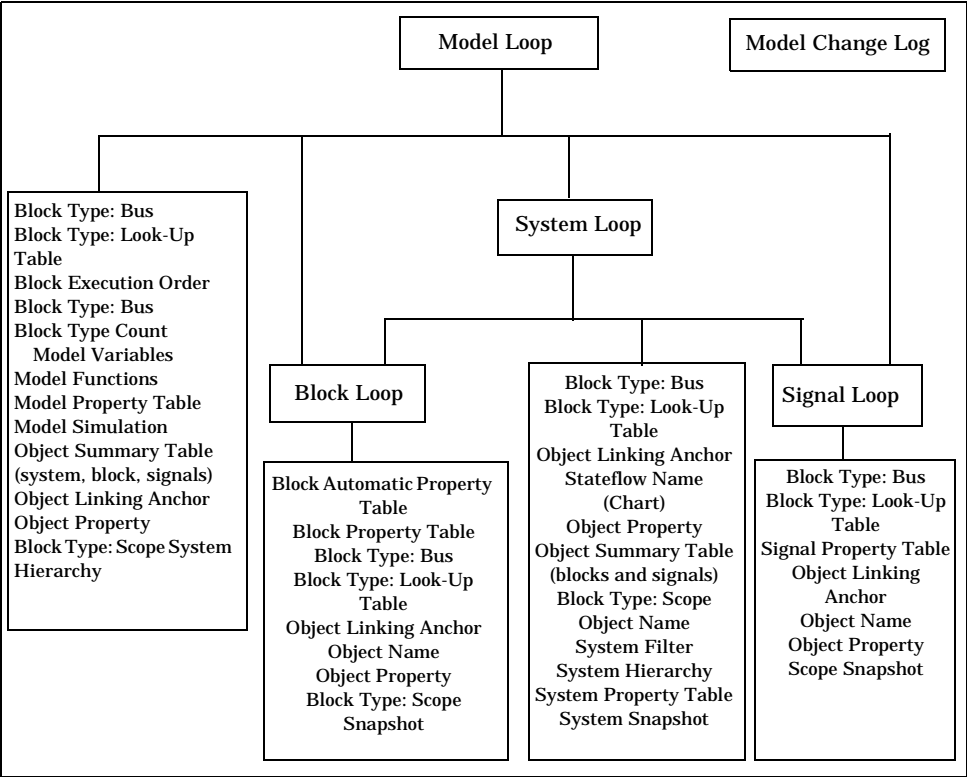
**Insert anything  
into report?** Yes. Text.

**Filename** crgtds

## Simulink Components

You can use Simulink components to include information on Simulink models in your report. Simulink components can report at four different levels of a Simulink model: model, system, block, and signal.

Simulink components have the following parent/child relationships.



**Block Automatic Property Table.** This component creates a two-column property name/property value table.

**Block Execution Order.** This component creates and inserts a list or table of all blocks in the model (displayed in order by their execution sequence) into the report. The Block Execution Order should have the Model Loop component as its parent. If you place this component inside the System Loop component, you will get same list over and over again.

**Block Loop.** This component specifies which blocks are to be included in the report. The Block Loop component can have either the Model Loop, System Loop, or Signal Loop component as its parent.

**Block Property Table.** This component inserts a table that reports on block-level property name/property value pairs.

**Block Type: Bus.** This 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 busses and signals.

**Block Type Count.** This component 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.

**Block Type: Look-Up Table.** This component reports on one-dimensional and two-dimensional look-up 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.

**Block Type: Scope Snapshot.** This component inserts a snapshot of all scope blocks and XY plots in your report. The Block Type: Scope Snapshot component can have any Simulink Looping component as its parent.

**Model Change Log.** This component constructs a model history table that displays information about each logged revision to a model. This component must be executed before the Model Simulate component.

**Model Functions.** This component inserts a table that lists all of the functions used by blocks that are reported on in the current model. The Model Functions component expects the Model Loop component as its parent.

**Model Loop.** This component specifies which models and systems are to be included in the report.

**Model Property Table.** This component creates and inserts a property name/property value table for the model specified by the Model Loop component. The Model Property Table component has the Model Loop component as its parent.

**Model Simulation.** This component simulates the current Simulink model. The Model Simulation component expects the Model Loop component as its parent.

# Simulink Components

---

**Model Variables.** This component inserts a table listing all workspace variables used by blocks that are reported on in the current model. The Model Variables component expects the Model Loop component as its parent.

**Object Linking Anchor.** This component acts like a link in a paragraph or title; when there is a link that is being followed, the Object Linking Anchor shows the Report Generator where the link should go. This component can have any Simulink looping component as its parent.

**Object Name.** This component inserts the name of the current system, model, block or signal, according to the option you specify. The Object Name component can have any Simulink looping component as its parent. This component behaves like the Text component.

**Object Property.** This component inserts a single property name/property value pair for a Simulink object as text. The Object Property component can have any Simulink looping component as its parent. This component behaves like the Text component.

**Object Summary Table.** This component inserts a table that lists several Simulink objects and their properties. The Object Summary Table component can have the Model Loop as its parent if you are creating the table for systems, blocks, or signals. If you are creating the table for blocks and signals, then the System Loop component can be its parent.

**Signal Loop.** This component runs its subcomponents 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.

**Signal Property Table.** This component inserts a property name/property value table for the signal defined by the Signal Loop component. The Signal Property Table component has the Signal Loop component as its parent.

**System Filter.** This component 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.** This component inserts a list of the names of parent and child systems in the current model; child systems are shown indented under the parent systems. The Hierarchy component can have either the Model Loop or System Loop component as its parent.

**System Loop.** This component runs its subcomponents for each system defined in the Model Loop component. The System Loop component must have the Model Loop component as its parent.

**System Property Table.** Inserts a property name/property value table for the model specified by the Model Loop component. The System Property Table component behaves like the Cell Table component. This component expects the Model Loop or System Loop component as its parent.

**System Snapshot.** This component 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.

# Block Automatic Property Table

Component	Block Automatic Property Table
Category	Simulink
Description	The Block Automatic Property Table component creates a two-column property name/property value table. The property names are chosen automatically. If the reported block is a SubSystem block, the table will contain the MaskDialogParameters; otherwise, the table contains the block's DialogParameters. This is similar to the %<SplitDialogParameters> option in the Block Property Table component. The Block Automatic Property Table component reports on the current block, as defined by the Block Loop component.
Attributes	This is the Block Automatic Property Table attribute page.

Block Automatic Property Table

Table Content

Header row ☒ No header  
☐ Block type and name  
☐ "Name" and "Value"  
☒ Display property names as prompts

Table Properties

Table title ☐ No title  
☒ Use block name for title  
☐ Other:   
☒ Show table border

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Table Content

### Header row

The header row options control the title of each column in the table. You can select these header row options in this attribute:

- **No header** if you do not want a header row (or no titles for the table columns)
- **Block type and name** if you want the header row to display the block type and the name of the block
- **“Name” and “Value”** if you want the header row to read Name in one column and Value in the other

### Display property names as prompts

Select this option to display property names as they are displayed in the graphical user interfaces. If you do not select this option, property names are displayed as they appear in the command line. For example, if you choose this option for the block type property, then the property is displayed as Block Type. If you do not choose this option, the property is displayed as BlockType.

## Table Properties

### Table title

You can select table title options in this attribute:

- **No title** if you do not want the table to be titled
- **Use block name for title** if you want to use the reported block name as the title of the table
- **Other** if you want to specify a title for the table and enter the title text in the corresponding field

# Block Automatic Property Table

---

**Show table border**  
Select this option to put a border around the property table.

**Insert anything into report?** Yes. Table.

**Filename** csl\_blk\_autotabl e



Component	Block Execution Order
Category	Simulink
Description	<p>The Block Execution Order component creates a list or table of all nonvirtual blocks in the model, showing the order in which they are executed. See the Simulink documentation for a definition and a list of virtual and nonvirtual blocks.</p> <p>You must have Real-Time Workshop to use this component. If you do not have Real-Time Workshop, then this component does not insert anything into the report.</p>
Attributes	This is the Block Execution Order attribute page.

Block Execution Order

Report Options

Link to:

☒ No linking

☐ Link to block's parent system

☐ Link to block

☒ Include block type information

Display Options

Present blocks as

☒ Numbered list

☐ Table

List title

# Block Execution Order

---

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Report Options

### Link to:

This option provides a hyperlink to another block or system in your report.

If you click on this link, it will take you to the location in the report where the block or system is reported on. If there is no such place in the report, then a hyperlink is not created during the report generation process, even if you choose this option.

- Choose **No linking** if you do not want any hyperlinks.
- Choose **Link to block's parent system** if you want the hyperlink to link to the block's parent system.
- Choose **Link to block** if you want the hyperlink to link to the block.

### Include block type information

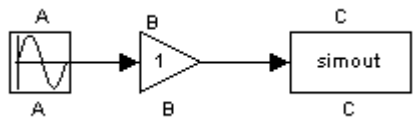
Select this option if you want to include each block's BlockType property in the list or table.

## Display Options

### Present list as

- Choose **Numbered list** to create a numbered list.
- Choose **Table** to create a table.

For example, if you have the following Simulink model or system named Demo\_System,



**Numbered list** creates the following list (note that this list includes block type information, since that option was selected for this example).

- 1. Demo\_System/A (Sin)
- 2. Demo\_System/B (Gain)
- 3. Demo\_System/C (ToWorkspace)

**Table** creates the following table.

Order	Block Name	Block Type
1	Demo_System/A	Sin
2	Demo_System/B	Gain
3	Demo_System/C	ToWorkspace

Note that the **Include blocktype information** option is selected for the examples above.

**List/Table title**

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. See “Report Format and Stylesheets” on page 2-15 to find stylesheets that have a List of Tables.

# Block Execution Order

---

**Insert anything into report?**    Yes. List or table.

**Filename**            csl sortblockl i st

Component	Block Loop
Category	Simulink
Description	The Block Loop component runs its subcomponents for each block contained in the current system, model, or signal.
Attributes	This is the Block Loop attribute page.

Block Loop

Report On

☒ Auto - All blocks in all models

☐ Manual - use block list:

Loop Options

Sort blocks

Alphabetically by block name

☐ Search for property name/property value pairs

Property name

Property value

BlockType

Scope

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

## Report On

This frame describes what the Block Loop component is operating on.

### Auto - \_\_\_\_\_

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, then selecting this option causes the Block Loop component to report on all reported blocks in the current model.
- If the parent component is another System Loop, then 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, then selecting this option causes the Block Loop component to report on all blocks connected to the current signal.
- If the parent component is the , then selecting this option causes the Block Loop component to report on the current block.
- If the Block Loop does not have either the Model Loop, System Loop, Signal Loop or Block Loop component as its parent, then selecting this option causes the Block Loop component to report on all blocks in all models.

### Manual - 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 pathname 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. Please see “%<VariableName> Notation” in the Text component reference page for more details.

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.

```
si mppend/theta dot
%<Z>
```

If you define `Z` in the workspace as the following

```
Z={' si mppend/theta' }
```

then the Report Generator will report on the following blocks.

```
si mppend/theta dot
si mppend/theta
```

## Loop Options

You can choose block sorting options and reporting options in this frame.

### Sort blocks

Use this option to select how the blocks are sorted (this option is available if you select the **Auto** - \_\_\_\_\_ option in the **Report On** section or if you select **Manual - 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 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 (req RTW)** to sort blocks by execution order. This option requires Real-Time Workshop; if you do not have Real-Time Workshop, this option will not appear in the attribute page.

### Search for 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.

## Block Loop

---

If you have Release 12 or greater of MATLAB, the **Property value** field is always read as a regular expression. Therefore, if you want to have an exact string match, you must enter the string in this way: \<string>. See `find_system` for more information on regular expressions.

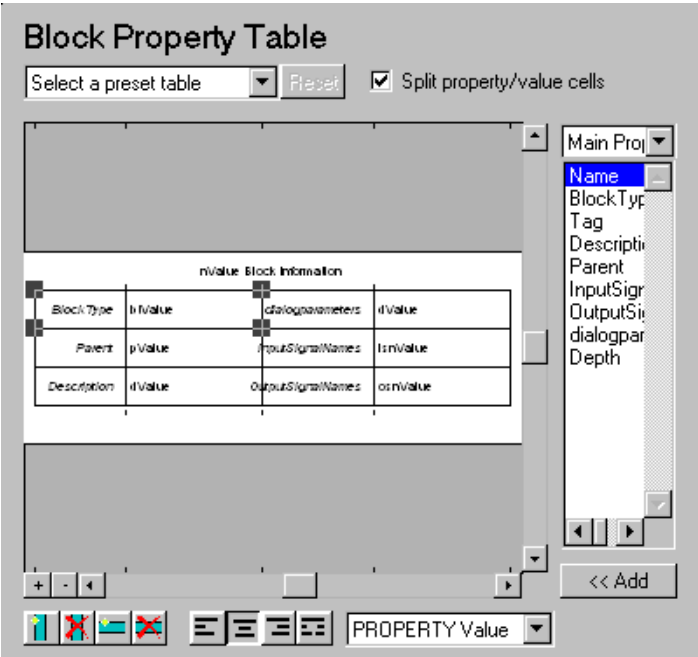
If you have an earlier release of MATLAB, the **Property value** field is always read as a string.

**Insert anything  
into report?** No.

**Filename** csl\_blk\_loop



<b>Component</b>	Block Property Table
<b>Category</b>	Simulink
<b>Description</b>	The Block Property Table component inserts a table that reports on block-level property name/property value pairs.
<b>Attributes</b>	This is the Block Property Table attribute page.



The Block Property Table component is similar to the Handle Graphics Figure Property Table component. See the Handle Graphics Figure Property Table component reference page.

# Block Property Table

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

## Example

The table below is an example of a block property table. It is the Default preset table. This property table is reporting on the Transfer Fcn\_2 block in the Aircraft Dynamics Model in the f14 demo model.

<i>BlockType</i>	TransferFcn	<i>dialogparameters</i>	Numerator Denominator
<i>Parent</i>	f14/Aircraft Dynamics Model	<i>InputSignalNames</i>	<2641.00024>
<i>Description</i>		<i>OutputSignalNames</i>	<2640.00024>

**Insert anything into report?** Yes. Table.

**Filename** csl\_block\_proptable

<b>Component</b>	Block Type: Bus
<b>Category</b>	Simulink
<b>Description</b>	The Block Type: 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 busses and signals.
<b>Attributes</b>	This is the Block Type: Bus attribute page.

**Block Type: Bus**

Show Bus Blocks

All bus blocks in all models

☐ Show bus hierarchy

List Options

Title

☐ Insert linking anchor for bus blocks

☐ Insert linking anchor for signals

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

# Block Type: Bus

---

## Show Bus Blocks

This frame describes what the component is operating on.

- If the Block Type: Bus has the Model Loop as its parent, it includes all busses in the current model.
- If the Block Type: Bus has the System Loop as its parent, it includes all busses in the current system.
- If the Block Type: Bus has the Block Loop as its parent, it includes the current block if it is a bus.
- If the Block Type: Bus has the Signal Loop as its parent, it includes all busses connected to the current signal.
- If the Block Type: Bus does not have any of the Simulink looping components as its parent, it includes all busses in all open models.

## Show Bus Hierarchy

Use this check box to specify whether or not the list should display downstream busses in a hierarchical form.

Hierarchical List:

List Title

- Bus Selector
  - signal 3
  - signal 2
  - signal 4
- Bus Selector1
  - signal 3
- Bus Selector2
  - signal 2
  - signal 1
- Bus Selector3
  - signal 2
  - signal 1

### Nonhierarchical List:

- List Title
- Bus Selector
    - signal 3
    - signal 2
    - signal 4

- List Title
- Bus Selector1
    - signal 3

- List Title
- Bus Selector2
    - signal 2
    - signal 1

- List Title
- Bus Selector3
    - signal 2
    - signal 1

### List Options

#### Title

Use this option to insert a title before each list. Note that this attribute supports the %<varname> notation.

#### Insert linking anchor for bus blocks

When this option is selected, the 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 Object 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.

# Block Type: Bus

---

**Insert linking anchor for signals**

When this option is selected, the 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 Object 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.


**Insert anything into report?**    Yes. List.

**Filename**                    csl\_block\_bus

Component	Block Type Count
Category	Simulink
Description	The Block Type Count component counts the number of each block type in the current model or system. See the Simulink documentation for a description of block type.
Attributes	This is the Block Type Count attribute page.

Block Type Count

Count Block Types

All blocks in all models

☐ All blocks in model

☐ All blocks in reported systems

Table Content

Table title

Block Type Count

☒ Show block names in table

Sort table

☒ Alphabetically by block type

☐ By number of blocks

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

# Block Type Count

---

## Count Block Types

This frame describes where block types are being counted.

- If the Block Type Count 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 has the System Loop as its parent, it reports all block types in the current system.
- The Block Loop and the Signal Loop are not valid parents for the Block Type Count.

## Table Content

### Table title

Enter the title of the table in this field.

### Show block names in table

Select this option to include 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**.

Insert anything into report? Yes. Table.

Filename csl blockcount



<b>Component</b>	Block Type: Look-Up Table
<b>Category</b>	Simulink
<b>Description</b>	The Block Type: Look-Up Table component reports on one-dimensional and two-dimensional look-up table blocks. See the Simulink documentation for information on the Look-Up Table block and the Look-Up Table (2-D) block. 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.
<b>Attributes</b>	This is the Block Type: Look-Up Table attribute page.

**Block Type: Look-Up Table**

Display 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

Figure/Table Title

- ☒ No title
- ☐ Auto: title from block name
- ☐ Manual: Look-Up Table

Image

- Image file format Automatic Handle Graphics Format
- Paper orientation Portrait
- Image size 5x4 inches

# Block Type: Look-Up Table

---

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Display Options

This frame allows you to specify which types of Look-up Table blocks are included in the report and how they are displayed. If none of the check boxes in this frame are selected, the component does not insert anything into the report.

These options display the one-dimensional and two-dimensional look-up tables, depending on what the component is operating on:

- If the Block Type: Look-Up Table has the Model Loop as its parent, it includes all look-up tables in the current model.
- If the Block Type: Look-Up Table has the System Loop as its parent, it includes all look-up tables in the current system.
- If the Block Type: Look-Up Table has the Block Loop as its parent, it includes the current block if it is a look-up table.
- If the Block Type: Look-Up Table has the Signal Loop as its parent, it includes all look-up tables connected to the current signal.
- If the Block Type: Look-Up Table does not have any of the Simulink looping components as its parent, it includes all look-up 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 drop box. See the MATLAB graphics documentation for more information 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 more information 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.

### Figure/Table Title

Choose a figure/table title option in this section:

- **No title:** Select this option if you do not want a figure/table title.
- **Auto: title from block name:** Select this option if you want the figure/table title to be the same as the block name.
- **Manual:** Select this option and type in the figure/table title in the corresponding field if you do not want the title to be the same as the block name.

Note that if you have both a figure and a table, only the figure gets a title. If you only have a figure or a table, then the figure or table gets a title.

### Image

#### Image file format

Choose the image file format (e.g., JPEG, TIFF, etc.) from this drop box. To automatically choose the format best suited for the output format that you chose in the Report component, select **Automatic Handle Graphics Format**, which is the default option. Otherwise, make sure that you choose an image format that your output viewer can read.

#### Paper orientation

You can choose **Landscape**, **Portrait**, or **Rotated** as your paper orientation.

# Block Type: Look-Up Table

---

**Image size**

Specify the image size in the form wxh (width times height) in the **Image size** field and the units in the corresponding drop box. The image size equals these dimensions.

If you change the units, the number in the image size field will automatically change to match the units.


**Insert anything into report?**      Yes. Figure and/or table.

**Filename**                      csl\_blk\_lookup

Component	Block Type: Scope Snapshot
Category	Simulink
Description	The Block Type: Scope Snapshot component inserts an image of a Simulink scope and XY graphs. If the model has not been simulated, scopes will be empty. See the Model Simulation component attribute page.
Attributes	This is the Block Type: Scope Snapshot attribute page.

Block Type: Scope Snapshot

Take Snapshots of

MODEL

All XY graphs & scopes in all models

☒ Include open scopes only

☐ Include all scopes

Snapshot Options

Image file format

Automatic Handle Graphics Format

Image size:

5x4

inches

Paper orientation

☒ Portrait

☐ Landscape

☐ Rotated

Caption

☒ No caption

☐ Automatic (use block description)

☐ Use white background and black lines

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

# Block Type: Scope Snapshot

---

## Take Snapshots of

This frame describes what the Block Type: Scope Snapshot component is operating on:

- If the Block Type: Scope Snapshot has the Model Loop as its parent or if it does not have any of the Simulink looping components as its parent, it includes all XY graphs and scopes in the current model.
- If the Block Type: Scope Snapshot has the System Loop as its parent, it includes all XY graphs and scopes in the current system.
- If the Block Type: Scope Snapshot has the Block Loop as a parent, it includes the current block if it is an XY graph or scope.
- If the Block Type: Scope Snapshot has the Signal Loop as its parent, it includes all XY graphs and scopes connected to the current signal.
- If the Block Type: 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.

## Include open scopes only

Select this option to take a snapshot of scopes that are currently open.

## Include all scopes

Select this option to take a snapshot of all scopes in the model. This option forces closed scopes to open when the report is generating.

## Snapshot Options

### Image file format

Choose the image file format (e.g., JPEG, TIFF, etc.) from this drop box. To automatically choose the format best suited for the output format that you chose in the Report component, select `Automatic`, which is the default option. Otherwise, make sure that you choose an image format that your output viewer can read.

### Image Size

Specify the image size in the form `wxh` (width times height) in the **Image Size** field and the units in the corresponding drop box. The image size equals these dimensions.

If you change the units, the number in the image size field will automatically change to match the units.

**Paper orientation**

You can choose **Landscape**, **Portrait**, or **Rotated** as your paper orientation.

**Caption**

You can choose **No caption** if you do not want a caption for the scope, or you can choose **Automatic (use block description)** if you want the caption to be the Scope Block description.

**Use white background and black lines**

Select this option to display the scope snapshot with a white background and black axes and traces. This is the best option if you are printing your output.

**Insert anything into report?**      Yes. Image.

**Filename**                      csl scopesnap

# Model Change Log

---

Component	Model Change Log
Category	Simulink
Description	<p>The Model Change Log component uses a reported model's Modified History 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:</p> <ul style="list-style-type: none"><li>• Author of each change</li><li>• Model version of the change</li><li>• Time and date of the change</li><li>• Description of the change</li></ul> <p>To see the model revision history parameters, select <b>Model Properties</b> from the model's <b>File</b> menu. This will display the <b>Block Diagram Properties</b> dialog box. The <b>Model Properties</b> tab in this dialog box contains the model revision history parameters.</p> <p>The <b>Options</b> tab in the <b>Block Diagram Properties</b> dialog box controls how the version number is incremented. The <b>History</b> tab controls history updates and lets you edit the Modified History string. You must select the <b>Prompt For Comments When Save</b> option from the <b>Modified History Update</b> drop box in order to save your revisions. This component will not run if the model's Modified History parameter is empty or if you edit the parameter in such a way that the Report Generator cannot parse it; in these cases, the Comment component will not be able to produce the table.</p> <p>For more information on the <b>Block Diagram Properties</b> dialog box and model history, see the Simulink documentation.</p> <p>Note that if your model has a long revision history, you may want to limit the number of revisions reported.</p> <p>The Model Change Log component should run before the Model Simulation component.</p>



## Attributes

This is the Model Change Log attribute page.

Model Change Log

☐ Limit displayed revisions to:

Table Columns

☐ Author name

☐ Version

☒ Date changed - Format:

☒ Description of change

Table Display

Table title:

☒ Display table border

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

### Limit displayed revisions to:

Select this option to limit the number of revisions that is displayed in the report and enter the number in the corresponding text field.

### Table Columns

You can choose the information displayed in the model revision table in this section.

# Model Change Log

**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 Display**

You can choose how the model revision history table is displayed in this section.

**Table title:**

Enter the title of the table in this field.

**Display table border**

Select this option if you want to place a border around the table.

**Example**

This is an example of a report generated from a setup file with the Model Change Log component.

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.

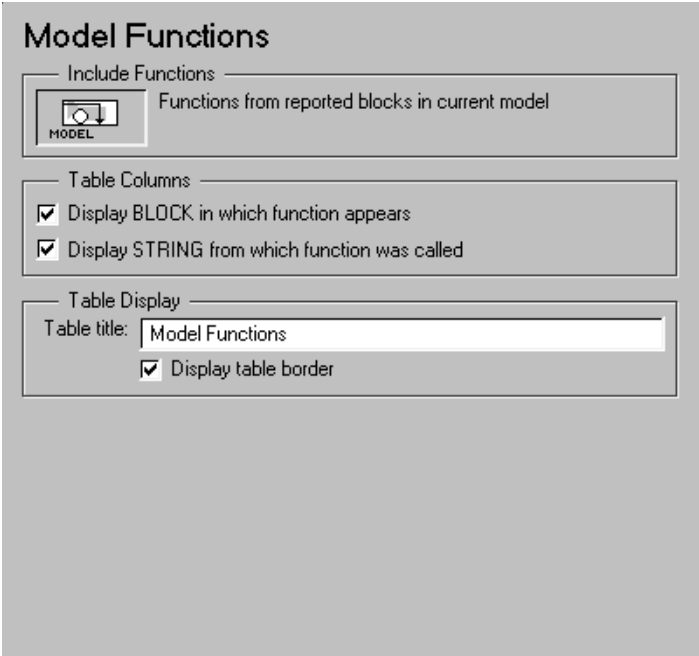
**Insert anything  
into report?**      Yes. Table.

**Filename**              csl\_md1\_changel og

# Model Functions

---

Component	Model Functions
Category	Simulink
Description	The Model Functions component shows all top-level functions used by reported blocks in a Simulink model and enters each function name in a table.
Attributes	This is the Model Functions attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

Include Functions

This context-sensitive frame shows that this component is reporting on functions in reported blocks in the current model.

Table Columns

Display BLOCK in which function appears

Select this option to include the parent blocks that contain the function.

Display STRING from which function was called

Select this option to include the M-code line that calls the function.

Table Display

Table title:

Enter the title of the table in this field.

Display table border

Select this option to insert a border around the table.

Example

The table below is an example of a model function property table. This model function table is reporting on the Dryden Wind Gust Models in the F14 demo model.

Function Name	Parent Blocks	Calling string
pi	f14/Dryden Wind Gust Models/Q-gust model , f14/Dryden Wind Gust Models/Q-gust model	pi/(4*b)*[1 0] [1 pi*Vto/(4*b)]
sqrt	f14/Dryden Wind Gust Models/W-gust model	Swg/ sqrt(a^3)*[sqrt(3)*a, 1]

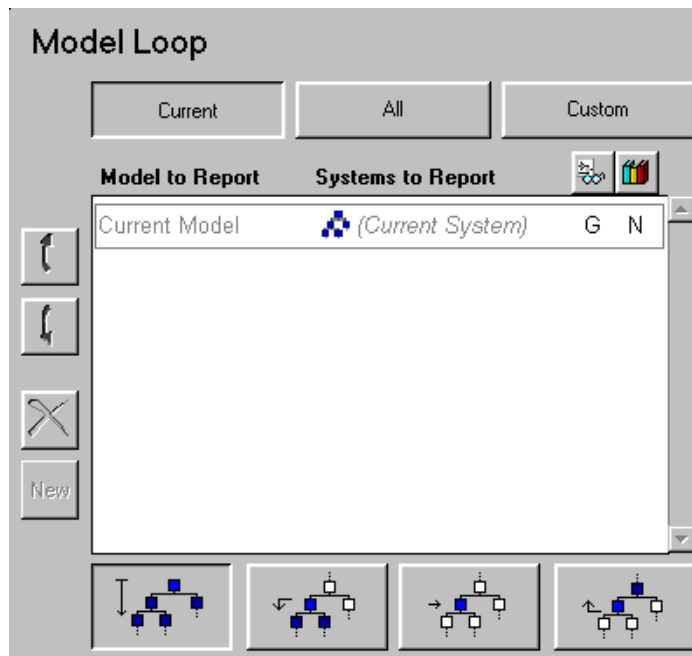
# Model Functions

---

**Insert anything  
into report?**    Yes. Table.

**Filename**            csl\_functions

<b>Component</b>	Model Loop
<b>Category</b>	Simulink
<b>Description</b>	<p>The Model Loop component runs its subcomponents for all selected Simulink models and systems. The Model Loop component can run its subcomponents for a specific model, such as the demo model F14, or it can run its subcomponents for all models or the current model.</p> <p>Because Simulink components are context-sensitive, they should be parented by the Model Loop (although the Model Loop does not have to be the immediate parent of some Simulink components).</p>
<b>Attributes</b>	This is the Model Loop attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Choosing Models to Report On

You can report on:

- The current model
- All models that are currently open
- A custom set of models and systems

The **Model to Report** column shows you what models are reported.

### Reporting on the Current Model

If you want to report on the current model, click on the **Current** button.

To find out what the current model is, type `bdroot` in the MATLAB command window.

### Reporting on All Models

If you want to report on all models that are currently open, click on the **All** button.

### Reporting on Custom Models





If you want to report on a custom set of models and systems, click on the **Custom** button.

The **Model to Report** column, which contains a list of reported models, is initially empty. Type the name of the model in this column. Press the **Esc** key when you are finished.

To add another model to the list, click on the **New** button; an empty row will appear.



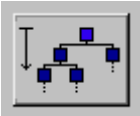
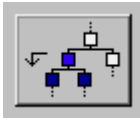
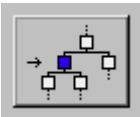
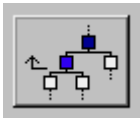
The following table shows you which button you should use to move a model up or down in the list, or to add or delete a model.

Use this button:	To do this:
	To move a model up in the list
	To move a model down in the list
	To remove a model from the list
	To add a model to the list

# Model Loop

## Choosing Systems to Report On

For each of the model reporting options (**Current**, **All** and **Custom**), you can select the systems that will be included in the report for each model. You can select the systems that are reported from the **Systems to Report** column.

Use this button	To do this
	To report on all systems in the current model
	To report on the current system in a model and all of its subsystems
	To report on the current system only
	To report on the current system and all the systems above it

You can also select the reported system through the context menu. To invoke the content menu, click on the right mouse button. The options shown in the table above are available through **Set loop type** in the context menu.

## Specifying the Current System for the Reported Model

You can specify the current system in the reported model for the following reported systems:

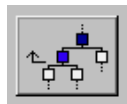
- Current system only
- Current system and all the systems above it
- Current system in a model and all of its subsystems

How to specify the current system for:

- The current model: You can choose to report on the **Top System** or the **Current System** when you are reporting on the current model. Click on **Top System** or **Current System** in the **Systems to Report** column to toggle between the two systems. You can also choose between the two options by using the context menu.
- All models: The current system for the **All Models** option is always the top system.
- A custom model: You can choose to report on the top system or any other system when you are reporting on a custom model. You can also choose multiple systems.
  - *Choosing a system:* To select the current system for a custom model, click in the **Systems to Report** column. The **Select “Current” Systems** window appears (shown in the example below). You can also invoke this window by using the context menu and choosing **Select current system(s)**.

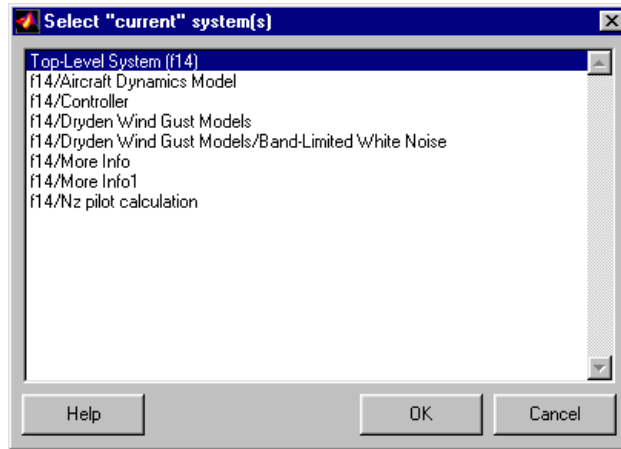
Select the system you want to include and click on **OK**. The system you select is now displayed in the **Systems to Report** column.

For example, choose the custom model option and click in the **Model to Report** column. Type f14 in the **Model to Report** column. Choose to report on the current system and all the systems above it by clicking on the button shown below.



# Model Loop

Click in the **Systems to Report** column. The **Select “Current” Systems** window, shown below, lists all of the systems in the f14 model.



Select **f14/Controller** and then click on **Ok**. The **Systems to Report** column now has the following entry: **(f14/Controller)**. F14 and F14/Controller are included in the report.

- *Choosing multiple systems:* To select multiple systems, hold down the **Shift** key and select the systems you want to report on. Click on **Ok** when you are done. The **Systems to Report** column now has the following entry: **(Multiple)**.

If you choose to report on multiple systems, and you are reporting on the current system, then the systems you have selected are included in the report.

If you choose to report on multiple systems, and you are reporting on the current system and all the systems above it, then all the systems you have selected and all the systems above them are included in the report.

If you choose to report on multiple systems, and you are reporting on the current system and all of its subsystems, then all the systems you have selected and all their subsystems are included in the report. The subsystems that are included in the report depend on the options you choose for following library links and for looking under masks. These two options are discussed in the next two sections.

Following Library Links

Simulink blocks and systems can be linked to a library. See the Simulink documentation for more information about library links.

Invoke the context menu to toggle the **Follow library links** option on and off.

The following table shows the options for reporting on blocks or systems with library links. Select the option by clicking on the option letter until it changes to the letter of your choice. These options are available if you are reporting on one of the following:

- All systems in a model
- The current system in a model and all of its subsystems

The options shown in the table below are also available through **Follow library links** in the context menu.

Select this option	To report on these systems
Y	Systems with library links
N	Systems without library links
U	Systems with unique library links (if you have more than one system with the same library link, only one of them is included in the report)

Looking Under Masks

Simulink systems can have masks. The masks can be functional or graphical. See the Simulink documentation for more information about masks.

The following table shows the options for reporting on masked systems. Select the option by clicking on the option letter until it changes to the letter of your choice. These options are available if you are reporting on one of the following:

- All systems in a model
- The current system in a model and all of its subsystems

# Model Loop

The options shown in the table below are also available through **Look under masks** in the context menu.

Select this option	To report on these systems
N	Systems with no masks
G	Systems with graphical masks Systems with no masks
F	Systems with functional masks Systems with graphical masks Systems with no masks
A	All systems

**Note** If you choose **Y** to report on systems with library links and **N** to report on systems with no masks, then the report will not include most library blocks, since most library blocks tend to be masked.

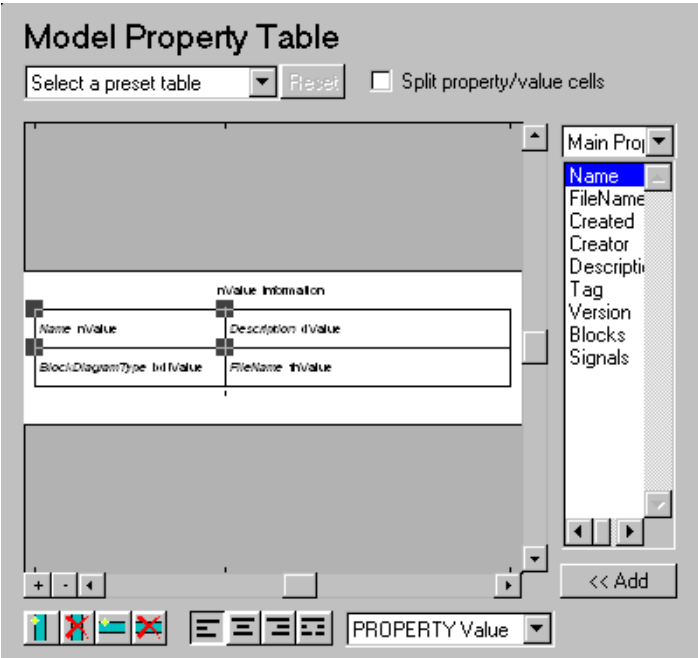
Insert anything  
into report?

No.

Filename

csl\_md1\_loop

Component	Model Property Table
Category	Simulink
Description	The Model Property Table component inserts a table which reports on model-level property-value pairs.
Attributes	This is the Model Property Table attribute page.



**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

# Model Property Table

The Model Property Table is similar to the Handle Graphics Figure Property Table. See the Handle Graphics Figure Property Table for reference.

Note that if you do not have Real-Time Workshop, **Summary Properties (req RTW)** will not be listed in this drop box, since summary properties require Real-Time Workshop. If you are using a setup file that contains a summary property, then 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.

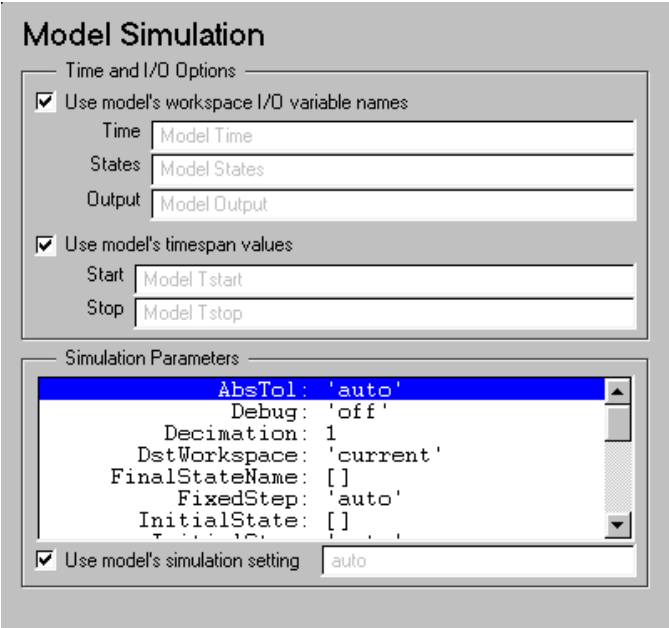
<i>Solver</i> ode45	<i>ZeroCross</i> on	<i>StartTime</i> 0.0 <i>StopTime</i> 60
<i>RelTol</i> 1e-4	<i>AbsTol</i> 1e-6	<i>Refine</i> 1
<i>InitialStep</i> auto	<i>FixedStep</i> auto	<i>MaxStep</i> auto
<i>LimitMaxRows</i> off	<i>MaxRows</i> 1000	<i>Decimation</i> 1

Insert anything into report? Yes. Table.

Filename csl \_mdl \_proptabl e



Component	Model Simulation
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	This is the Model Simulation attribute page.



**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

## Time and I/O Options

### Use model's workspace I/O variable names

Select this option to use the names of the variables specified in the **Workspace I/O** 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 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.

### 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, then the Report Generator will terminate the model simulation after 60 seconds to prevent the report from generating for an infinite amount of time.

---

## Simulation Parameters

The list box in this section of the attributes page contains all of the model's simulation parameters. To use these parameters, select **Use model's simulation setting**.

If you do not want to use these parameters, make sure that the **Use model's simulation setting** check box is not checked. Then click on the parameter you want to edit and type in the new parameter value in the field below the list box and next to the **Use model's simulation setting** check box.

**Insert anything into report?** No.

**Filename** csl si m


# Model Variables

---

Component	Model Variables
Category	Simulink
Description	<p>The Model Variables component creates a table that shows all workspace variables 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.</p> <p>For example, a Simulink Gain block might have a string <code>x</code> instead of a number. Simulink looks for a variable <code>x</code> in the workspace.</p>
Attributes	This is the Model Variables attribute page.

Model Variables

Include Variables

Workspace variables from reported blocks in current model

☐ Include Workspace I/O parameters

Table Columns

☒ Block in which variable appears

☒ String from which variable was called

☐ Size of variable (MxN)

☐ Memory requirements (bytes)

☐ Class of variable (double, string, etc.)

☒ Value in workspace

☐ RTW storage class

Table Display

Table title:

☒ Display table border

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

**Include Variables**

This context-sensitive frame shows that this component is reporting on variables in reported blocks in the current model.

**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** tab 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 will look for the variable listed in the second column in the workspace. If the component finds it, it includes the variable in the report.

LoadExternal Input	External Input
SaveTime	TimeSaveName
SaveState	StateSaveName
SaveOutput	OutputSaveName
LoadInitial State	Initial State
SaveFinal State	Final StateName

## Table Columns

The options in this section specify the information included in the table. The first column is always **Variable Name**.

### Block in which variable appears

Select this option to include the name of the block that contains the reported variable. The title of this column is **Parent Blocks**.

### String from which variable was called

Select this option to include the piece of M-code that calls the reported variable. The title of this column is **Calling String**.

### Size of variable (MxN)

Select this option to include the size of the reported variable. The title of this column is **Size**.

### Memory requirements (bytes)

Select this option to include amount of memory bytes the reported variable needs. The title of this column is **Bytes**.

### Class of variable (double, string, etc.)

Select this option to include the variable class to which the reported variable belongs. The title of this column is **Class**.

### Value in Workspace

Select this option to include the actual value of the reported variable. The title of this column is **Value**. Note that very large arrays may display as [MxN CLASS]. For example, if you have a 300-by-200 double array, it will display 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, then 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

Table Display

Table title:

Enter the title of the table in this field.

Display table border

Select this option if you want to have a border around the table.

Example

This is an example of a table created by the Model Variables component. This property table is reporting 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

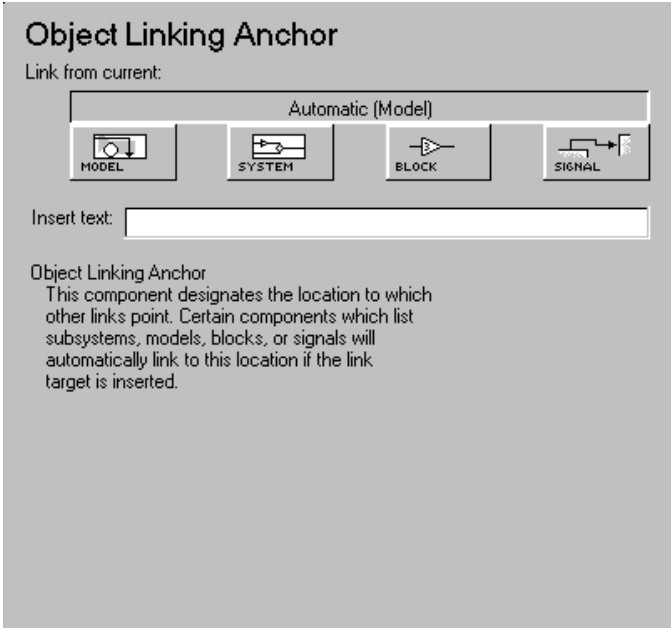
Insert anything into report? Yes. Table.

Filename csl\_variables

# Object Linking Anchor

---

Component	Object Linking Anchor
Category	Simulink
Description	The Object Linking Anchor component designates a location to which other links point. This component should have either the Model Loop, System Loop, Block Loop, or Signal Loop component as its parent.
Attributes	This is the Object Linking Anchor attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---



**Link from current:**

This option sets the current model, system, block, or signal as the linking anchor:

- Select **Automatic** to automatically select the appropriate model, system, block or signal as a linking anchor. If the Model Loop component is the parent component, then the linking anchor is set on the current model. Similarly, if the System Loop, Block Loop, or Signal Loop is the parent component, then 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.

**Insert text:**

If you want text to appear after the linking anchor, enter the text in this field.

---

**Note** You should not use the Object Linking Anchor component to create an anchor for an object that is anchored in the object summary table. See the Object Summary Table component reference page.

---

**Insert anything into report?**

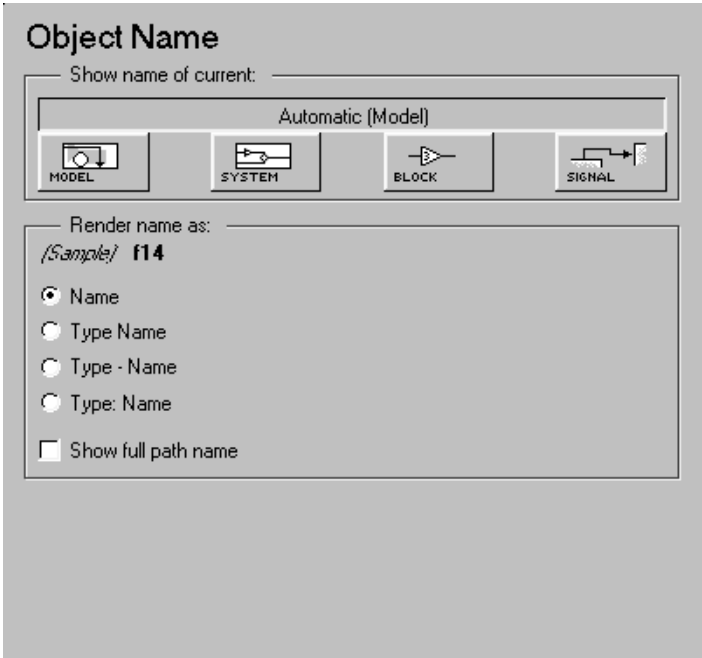
Yes. A link, and possibly text, depending on attribute choices.

**Filename**

`csllinktarget`

# Object Name

Component	Object Name
Category	Simulink
Description	<p>The Object Name component inserts the name of a Simulink model, system, block or signal into the report.</p> <p>You can use this component as the first subcomponent of a Chapter/Subsection component, which allows the current Simulink model, system block or signal name to be the chapter or section title.</p>
Attributes	This is the Object Name attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Show name of current:

- 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, then 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, then the signal <handle> is included in the report. The signal handle is a unique numerical identifier to that signal.

## Render name as:

Select one of the options listed below to display the Simulink object name in the report:

- **Name** - e.g., f14
- **Type Name** - e.g., Model f14
- **Type - Name** - e.g., Model - f14
- **Type: Name** - e.g., 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 will

# Object Name

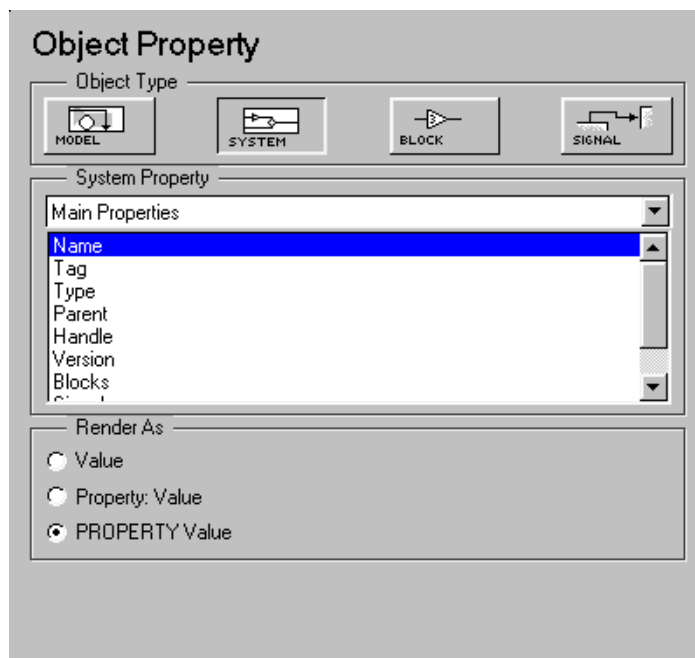
---

appear as: <Model Name>/<System Name>/<Block Name>. Note that this option is not available for models and signals.

**Insert anything into report?** Yes. Text.

**Filename** csl sysname

<b>Component</b>	Object Property
<b>Category</b>	Simulink
<b>Description</b>	The Object Property component inserts a single property name/property value pair for the current Simulink model, system, block, or signal.
<b>Attributes</b>	This is the Object Property attribute page.



**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

# Object Property

---

## Object Type

Select a Simulink object on which to report:

- Select **Model** to report on a property name/property value pair in the current model.
- Select **System** to report on a property name/property value pair in the current system.
- Select **Block** to report on a property name/property value pair in the current block.
- Select **Signal** to report on a property name/property value pair in the current signal.

## <Object> Property

This context-sensitive section contains a drop box that shows all the property types for the current Simulink model, system, block, or signal. Select the property type from this drop box. Note that if you do not have Real-Time Workshop, **Summary Properties (req RTW)** will not be listed in this drop box, since summary properties require Real-Time Workshop. If you are using a setup file that contains a summary property, then the property name appears in the report, but the property value does not.

This section also contains a list box that shows available property names for the property type you chose in the drop box. Select the property to include in the report from the list box.

## Render As

Select one of the options listed below to display the Simulink object property name/property value pair in the report:

- **Value** - displays the value of the property only (e.g., if the property is Name and the value is f14, only f14 is displayed).
- **Property: Value** - e.g., Name: f14
- **PROPERTY Value** - displays the name of the property in italics, followed by the value of the property (e.g., *Name* f14)

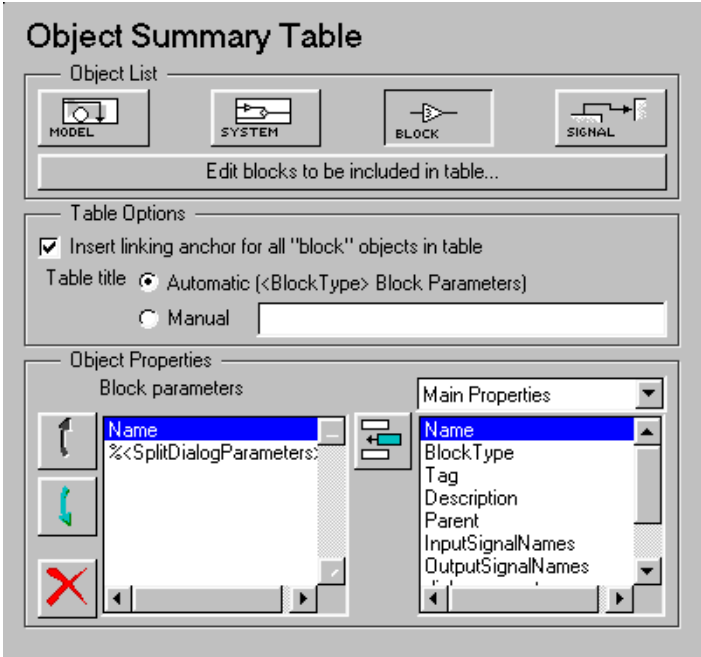
**Insert anything  
into report?**    Yes. Text.

**Filename**            csl property

# Object Summary Table

---

Component	Object Summary Table
Category	Simulink
Description	The Object Summary Table component shows properties or parameters of the selected Simulink models, systems, blocks, or signals in a table.
Attributes	This is the Object Summary Table attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

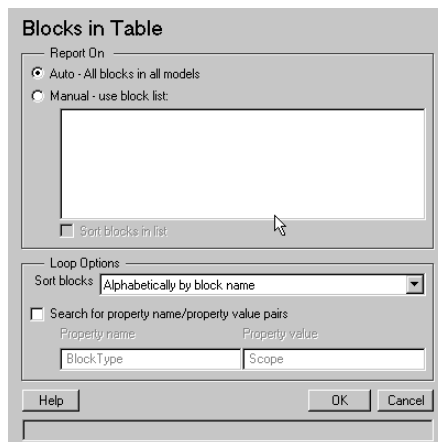


## Object List

Select a Simulink object:

- Select **Model** to create a summary table on the current model.
- Select **System** to create a summary table on the current system.
- Select **Block** to create a summary table on the current block.
- Select **Signal** to create a summary table on the current signal.

To edit blocks to be included in table, you can click on the bar. The **Edit Component Options** window appears, as shown below.



These are the same attribute options found in the Block Loop component. See the Block Loop component reference page for more information on these options.

For the Sort blocks option, note the following:

- Models can be only be sorted alphabetically.
- Systems can be sorted:
  - By the number of blocks in the system
  - Alphabetically by system name
  - By system depth

# Object Summary Table

---

- Block can be sorted:
  - Alphabetically by block name
  - Alphabetically by system name
  - By block type
  - By block depth
  - By simulation order (requires Real-Time Workshop)
- Signals can be sorted:
  - Alphabetically by signal name
  - Alphabetically by signal name (excludes signals with no name)
  - Alphabetically by system name
  - By signal depth

## Table Options

Select title and linking options for the summary table in this section.

### Insert linking anchor for all <objects> in table

This context-sensitive option inserts a linking anchor for all blocks, signals, systems, or models that are listed in the table.

Note that you should use this option if you already have an Object Linking Anchor component that creates an anchor for an object in the object summary table.

### Table title

Select a table title option:

- **Automatic (<BlockType> Block Parameters):** for example, if you are reporting on the Sum block, the table title will be the Sum Block Parameters
- **Manual:** if you want to create your own table title, select this option and enter a table title in the corresponding field. 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. Please see “%<VariableName> Notation” in the Text component reference page for more details.

## Object Properties





Select which object parameters are included in the report in this section of the attribute page.

### <Object> Parameters

This list box shows all of the object parameters that are included in the report. To add a parameter to this list, select the parameter category from the drop box on the left, and then select parameters from the list box below it. Then click on the add button, shown in the table below.

With the buttons shown in the table below, you can:

- Move a parameter up in the list
- Move a parameter down in the list
- Remove a parameter from the list
- Add a parameter to the list

Use this button	To do this
	To move a parameter up in the list
	To move a parameter down in the list
	To remove a parameter from the list
	To add a parameter to the list

# Object Summary Table

Note that if you do not have Real-Time Workshop, Summary Properties (req RTW) will not be listed in this drop box, since summary properties require Real-Time Workshop. If you are using a setup file that contains a summary property, then the property name appears in the report, but the property value does not.

**The %<SplitDialogParameters> and the dialogparameters Properties**  
The %<SplitDialogParameters> property creates columns with the dialog parameter names as headings (one column for each parameter), and the dialog parameter values listed in the table cells below it. This property is valid for blocks only and creates multiple tables, one for each block type.

The dialogparameters property creates one column with dialogparameters as a heading and lists the dialog parameter names in the table cells below it.  
For example, if your reported properties are Name and dialogparameters, then you will get the following object summary table.

Name	dialogparameters
Gain1	Gain
Gain2	Gain
Sum3	IconShape Inputs SaturateOnIntegerOverflow
T-Fcn1	Numerator Denominator

If you choose to report on the same blocks summarized in the table above but choose Name and %<SplitDialogParameters> as your reported properties, then you will get the following three object summary tables, one for each blocktype.

Name	Gain
Gain1	Mz
Gain2	5

Name	IconShape	Inputs	SaturdateOnIntegerOverflow
Sum3	round	++	0

Name	Numerator	Denominator
Sum3	1	[1, -Mq]

Example

If you are reporting on the demo model F14, and you choose the following options, then summary table shown on the next page 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)

# Object Summary Table

Name	Blocks
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
More Info	
More Info1	
Nz pilot calculation	w, q, Constant, Derivative, Derivative1, Gain1, Gain2, Product, Sum1, Pilot g force (g)

Insert anything  
into report?

Yes. Table.

Filename

csl\_summ\_table

<b>Component</b>	Signal Loop
<b>Category</b>	Simulink
<b>Description</b>	The Signal Loop component runs its subcomponents for each signal contained in the current system, model, or block.
<b>Attributes</b>	This is the Signal Loop attribute page.

The screenshot shows the 'Signal Loop' attribute page. It has a title bar 'Signal Loop'. Below it is a section 'Report on' with a dropdown menu showing 'MODEL' and a button with a magnifying glass icon. To the right of the dropdown is the text 'All signals in all models'. Below this are five checkboxes arranged in two columns: 'Block input signals', 'Block output signals', 'System input signals', 'System output signals', and 'System internal signals'. All five checkboxes are checked. Below this section is a section 'Sort Signals' with four radio button options: 'Alphabetically by signal name' (selected), 'Alphabetically by signal name (exclude empty)', 'Alphabetically by system name', and 'By signal depth'.

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

## Report On

This context-sensitive frame describes what the Signal Loop component is operating on:

- 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 deselecting 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 deselecting the corresponding option:
  - **Block input signals**
  - **Block output signals**
- If the Signal Loop does not have either 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 deselecting the corresponding option:
  - **Block input signals**
  - **Block output signals**
  - **System input signals**
  - **System output signals**
  - **System internal signals**



## 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.

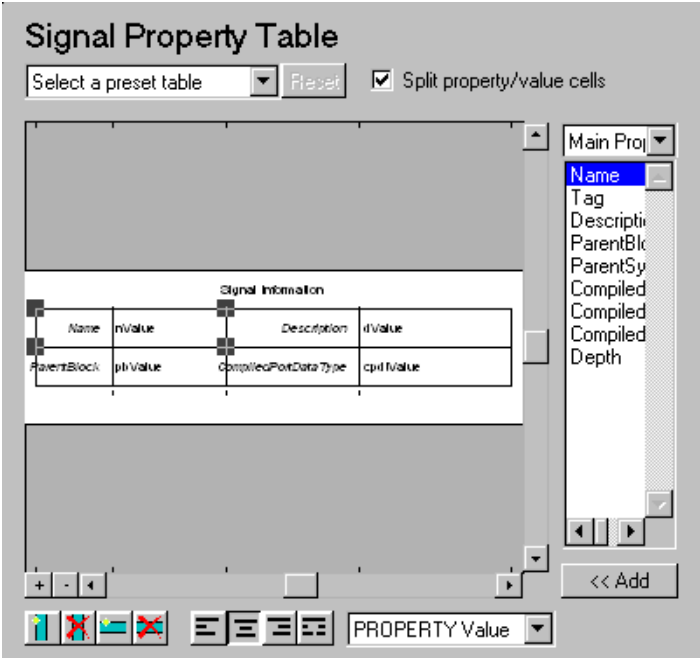
**Insert anything  
into report?**

No.

**Filename**      csl\_si g\_l oop

# Signal Property Table

Component	Signal Property Table
Category	Simulink
Description	The Signal Property Table component inserts a table that reports on signal-level property name/property value pairs into your report.
Attributes	This is the Signal Property Table attribute page.



The Signal Property Table component is similar to the Handle Graphics Figure Property Table component. See the Handle Graphics Figure Property Table component reference page.

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

### Example

The table below is an example of a signal property table. It is the **Compiled Information** preset table. This property table is reporting on signal *w* in the Aircraft Dynamics Model in the F14 demo model.

<i>Name</i>	w
<i>CompiledPortDataType</i>	auto
<i>CompiledPortWidth</i>	1
<i>CompiledPortComplexSignal</i>	-1

**Insert anything into report?** Yes. Table.

**Filename** csl\_sig\_proptable

# System Filter

---

Component	System Filter
Category	Simulink
Description	The System Filter component executes subcomponents only if the current systems meets the conditions specified in the attribute page.
Attributes	This is the System Filter attribute page.

System Filter

Only execute subcomponents:

if number of blocks in system >=

if number of subsystems >=

if system is

☐ masked

☐ unmasked

☒ either masked or unmasked

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Only execute subcomponents:

- Enter a number in the **if number of blocks in system**  $\geq$  field to execute subcomponents only if there are at least the specified number of blocks in the system. Note that if you enter 0, then there is no filtering; subcomponents are executed regardless of the number of blocks in the system.
- Enter a number in the **if number of subsystems**  $\geq$  field to execute subcomponents only if there are at least the specified number of subsystems in the system. Note that if you enter 0, then there is no filtering; subcomponents are executed regardless of the number of subsystems in the system.

## if system is

Select one of the following reporting options:

- Use the **masked** option to execute subcomponents only if the system is masked.
- Use the **unmasked** option to execute subcomponents only if the system is unmasked.
- Select the **either masked or unmasked** option to execute subcomponents regardless of whether the system is masked or unmasked.

Insert anything  
into report?

No.

Filename

csl filter

# System Hierarchy

---

Component	System Hierarchy
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	This is the System Hierarchy attribute page.

System Hierarchy

Starting System

Build list from:

☒ Current system

☐ Current model

☒ Emphasize current system

Display Systems

Show number of parents:

1

Show children to depth:

1

☐ Display peers of current system

List Style

☒ Bulleted list

☐ Numbered list

---

**Note** To view the component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Starting System

### Build list for:

- Choose **Current system** to create a hierarchy list for the current system.
- Choose **Current Model (Top-level system)** to create a hierarchy list for the current model.

### Emphasize current system

Select this option to display the name of the current system in an italicized font.

## Display Systems

### Show number of parents:

Use this field to specify the depth of parents that should be included in the hierarchy list. For example, if you specify 2 in this field and you are reporting on *Throttle*, which is a system in *Throttle* and *Manifold*, which is a system in the Simulink demo model *Engine*, then the hierarchy list contains:

```
engine
  Throttle & Manifold
    Throttle
```

Note that the starting system, *Throttle*, is emphasized by italics.

### Show children to depth:

Use this field to specify the depth of children that should be included in the hierarchy list. For example, if you are reporting on the demo model *F14* and you enter 2 in this field, then you get the following hierarchy list:

```
f14
  Aircraft Dynamics Model
  Controller
  Dryden Wind Gust Models
    Band-Limited White Noise
  More Info
  More Info1
  Nz pilot calculation
```

Note that the starting system, *f14*, is emphasized by italics.

**Display peers of current system**

Systems that have the same parent are peers. Use this option to display the peers of the current system. For example, if you are reporting on Dryden Wind Gust Models, and you select this option, then you get the following hierarchy list.

Aircraft Dynamics Model  
Controller  
*Dryden Wind Gust Models*  
Band-Limited White Noise  
More Info  
More Info1  
Nz pilot calculation

If you do not select this option, then you get the following hierarchy list.

*Dryden Wind Gust Models*  
Band-Limited White Noise

Note that the starting system, *Dryden Wind Gust Models*, is emphasized by italics.

**List Style**

**Bulleted list**

Select this checkbox to specify that the list uses a bulleted list format. This is the default.

**Numbered list**

Select this checkbox to specify that the list uses a numbered list format.

**Insert anything into report?** Yes. List.

**Filename** csl sysl i st



Component	System Loop
Category	Simulink
Description	The System Loop component runs its subcomponents for each system defined by the Model Loop. The System Loop component must have the Model Loop component as its parent.
Attributes	This is the System Loop attribute page.

System Loop

Report On

☒ Auto - All systems in all models

☐ Manual - use system list:

Sort Systems

☐ By number of blocks in system

☒ Alphabetically by system name

☐ By system depth

☐ No sort

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

## Report On

This context-sensitive frame describes what the System Loop component component is operating on.

### Auto - \_\_\_\_\_

Select this 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, then selecting this option causes the System Loop component to report on reported systems in the current model.
- If the parent component is another System Loop, then selecting this option causes the System Loop component to report on the current system.
- If the parent component is the Signal Loop, then 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, then selecting this option causes the System Loop component to report on the current block's parent system
- If the System Loop does not have either the Model Loop, System Loop, Signal Loop or Block Loop component as its parent, then selecting this option causes the System Loop component to report on all systems in all models.

### Manual - 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 pathname 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. Please see “%<VariableName> Notation” in the Text component reference page for more details.

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’ }

then the Report Generator will report on the following systems.

f14/Nz pilot calculation

f14/Controller

## Sort Systems

Use this section to select how the systems are sorted:

- 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 **Alphabetically by system name** to sort systems alphabetically by name.
- Select **By system depth** to sort systems by their depth in the model.

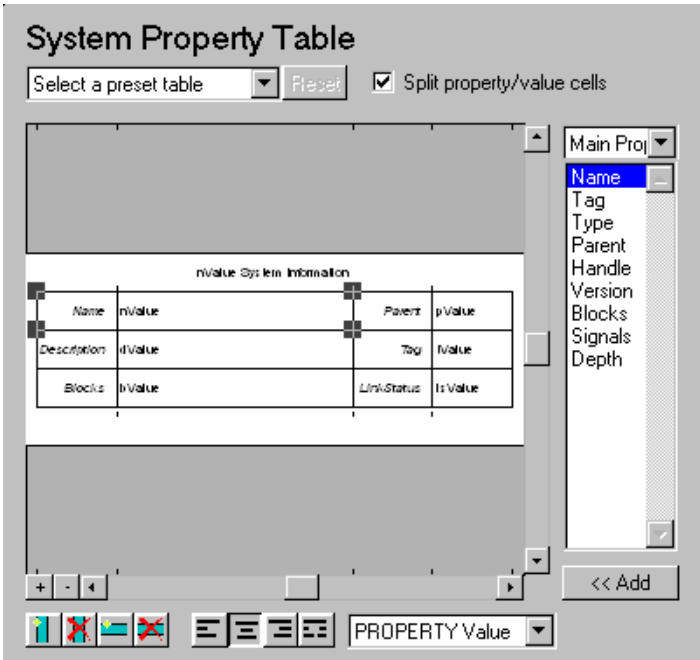
**Insert anything  
into report?**

No.

**Filename** csl\_sys\_loop

# System Property Table

Component	System Property Table
Category	Simulink
Description	The System Property Table component inserts a table that reports on system-level property name/property value pairs.
Attributes	This is the System Property Table attribute page.



The System Property Table is similar to the Handle Graphics Figure Property Table component. See the Handle Graphics Figure Property Table reference page.

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

Example

The table below is an example of a system property table. It is the **Default** preset table. This property table is reporting on signal w in the Aircraft Dynamics Model in the F14 demo model.

<i>Name</i>	Aircraft Dynamics Model	<i>Parent</i>	f14
<i>Description</i>		<i>Tag</i>	
<i>Blocks</i>	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)	<i>LinkStatus</i>	none

Insert anything  
into report?

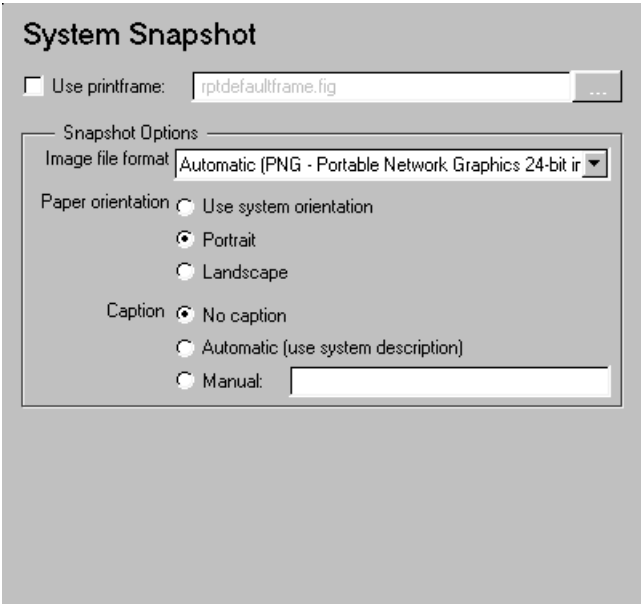
Yes. Table.

Filename csl\_sys\_proptable

# System Snapshot

---

Component	System Snapshot
Category	Simulink
Description	The System Snapshot component inserts a snapshot of the current system in your report.
Attributes	This is the System Snapshot attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

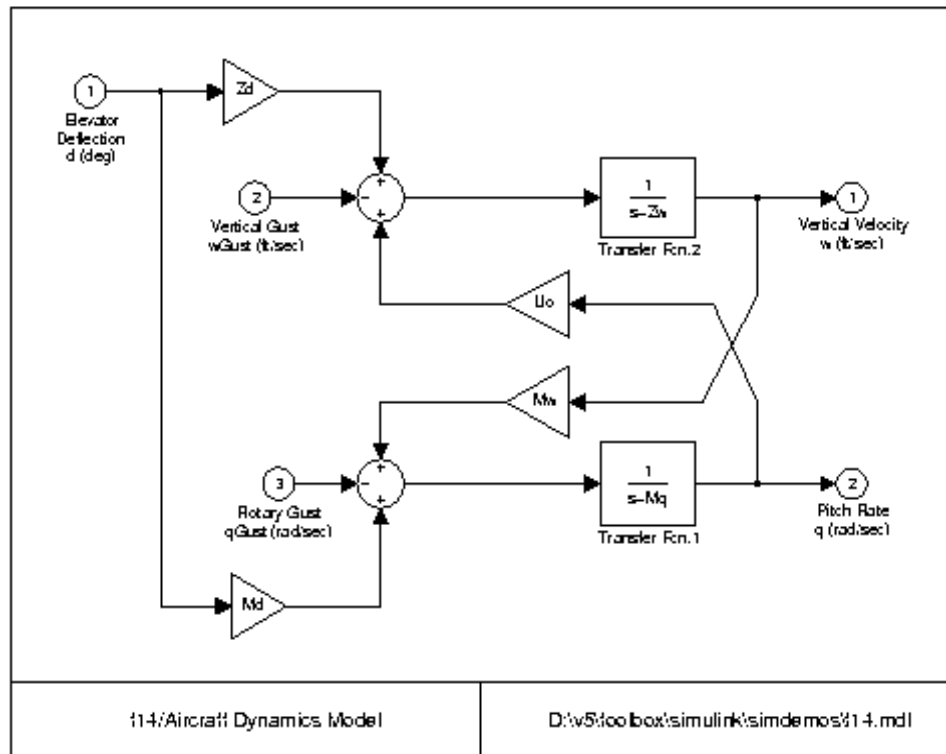
**Use printframe:**

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 Simulink reference pages for a discussion of the `frameedit` command.

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.

# System Snapshot



## Image file format

Choose the image file format (e.g., JPEG, TIFF, etc.) from this drop box. To automatically choose the format best suited for the output format that you chose in the Report component, select **Automatic**, which is the default option. Otherwise, make sure that you choose an image format that your output viewer can read.

## Paper orientation

You can choose **Landscape**, **Portrait**, or **Rotated** as your paper orientation, or you can choose the **Use system orientation** option. The **Use system orientation** option uses the `PaperOrientation` parameter in Simulink, which you can set as `Landscape`, `Portrait`, or `Rotated`.



**Caption**

You can choose **No caption**, **Automatic (use system description)**, or **Manual**: for specifying whether or not you want a caption for the image. The default is not to have a caption. If you choose **Manual**, then you need to type in the caption you want to appear.

**Insert anything into report?**      Yes. Image.

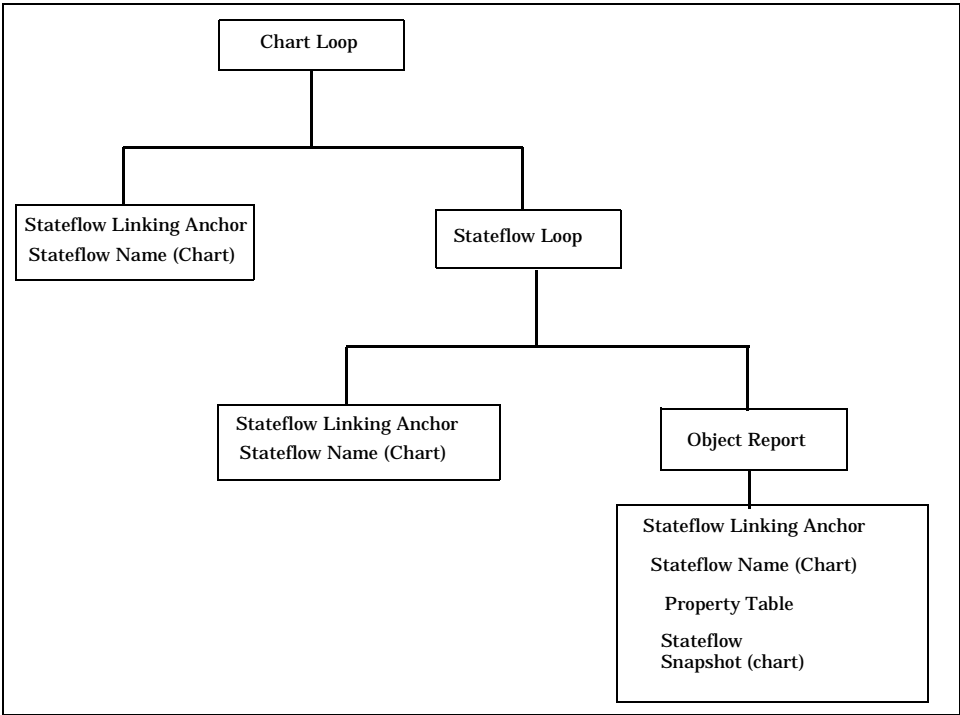
**Filename**                      csl snapshot

# Stateflow Components

## Stateflow Components

You can use Stateflow components to include information on Stateflow models in your report.

Stateflow components have the following parent/child relationships.



**Chart Loop.** This component 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. The Chart Loop cannot have any Stateflow component as its parent.

**Stateflow Linking Anchor.** This component acts like a link in a paragraph or title; when there is a link that is being followed, the Stateflow Linking Anchor shows the Report Generator where the link should go. This component must have the Stateflow Loop, Chart Loop, or Object Report component as its parent.

**Stateflow Name (Chart).** This component inserts the name of the Stateflow object specified by its parent component into the report. This component must have the Stateflow Loop, Chart Loop, or Object Report component as its parent.

**Object Report.** This component is a filter that runs its children only if the current object type (set by the parent Stateflow Loop) matches the component's object type. The Object Report component must have the Stateflow Loop component as its parent.

**Object Summary Table.** This component shows properties or parameters of the selected Stateflow objects in a table. The Object Summary Table Component can have the Chart Loop or any Simulink looping component (Model Loop, System Loop, Block Loop, or Signal Loop) as its parent.

**Property Table.** This component inserts a property name/property value table for a Stateflow object into the report. The Stateflow Property Table component must have the Object Report component as its parent.

**Stateflow Snapshot (chart).** This component inserts an image of a Stateflow object into the report. The Snapshot component must have the Object Report component as its parent and is only valid for certain object types.

**Stateflow Loop.** This component runs its children for each Stateflow object defined by its parent component. The Stateflow Loop component must have the Simulink Model Loop, System Loop, Signal Loop, Block Loop or Chart Loop component as its parent.

# Chart Loop

Component	Chart Loop
Category	Stateflow
Description	<p>The Chart Loop component runs its children for each Stateflow chart that you choose. The Chart Loop component can have either the Model Loop, System Loop, Signal Loop or Block Loop component as its parent. The Chart Loop cannot have any Stateflow component as its parents.</p> <p>Note that Stateflow objects that have already been reported on (by another loop) will not be reported again.</p>
Attributes	This is the Chart Loop attribute page.

Chart Loop

Report On

☒ Auto - All Stateflow charts in all models

☐ Manual - use chart list:

☐ Sort charts in list

Loop Options

Sort charts

Alphabetically by block name

Property name

Property value

☐ Search Stateflow:

☐ Search Simulink:

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Report On

### Auto - \_\_\_\_\_

Select this option to report on all charts in the current context. The context is set by the Chart Loop component's parent component:

- If the parent component is the Model Loop, then selecting this option causes the Chart Loop component to report on all reported Stateflow charts in the current model.
- If the parent component is the System Loop, then selecting this option causes the Chart Loop component to report on all Stateflow charts in the current system.
- If the parent component is the Signal Loop, then selecting this option causes the Chart Loop component to report on all Stateflow charts connected to the current signal.
- If the parent component is the Block Loop, then 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, then selecting this option causes the Chart Loop component to report on all Stateflow charts.

### Manual - use chart list:

Select this option to report on a list of Stateflow charts that you specify, and enter the chart names in the corresponding field. If you want to sort the charts in this list, select the **Sort charts in list** option and then select a sorting option from the **Sort charts** option (in the **Loop Options** section). Charts are reported on in the order that they are sorted. If you do not select this option, charts 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. Please see “%<VariableName> Notation” in the Text component reference page for more details and the **Manual - use block list** section in the Simulink Block Loop reference page for an example.

## Loop Options

You can choose chart sorting options and reporting options in this frame.

### Sort charts

Use this option to select how the blocks are sorted (this option is available if you select the **Auto - \_\_\_\_\_** option in the **Report On** section or if you select **Manual - use chart list** and the **Sort charts** 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 (req RTW)** to sort blocks by execution order. This option requires Real-Time Workshop; if you do not have Real-Time Workshop, this option will not appear in the attribute page.

### 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.

If you have Release 12 or later of MATLAB, the **Property value** field is read as a regular expression unless you enter a number, which will be treated as a numeric value. Therefore, if you want to have an exact string match, you must enter the string in this way: \<string>. See `find_system` for more information on regular expressions.

If you have an earlier release of MATLAB, the **Property value** field is read as a string.

### **Search Simulink:**

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.

If you have Release 12 or later of MATLAB, the **Property value** field is always read as a regular expression. Therefore, if you want to have an exact string match, you must enter the string in this way: \<string\>. See `find_system` for more information on regular expressions.

If you have an earlier release of MATLAB, the **Property value** field is always read as a string.

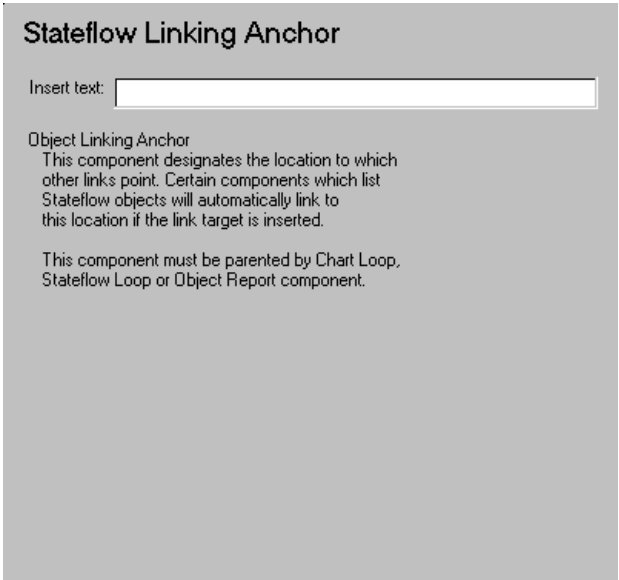
**Insert anything  
into report?**    No.

**Filename**            csf\_chart\_loop

# Stateflow Linking Anchor

---

Component	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, Stateflow Loop, or Object Report component as its parent.
Attributes	This is the Stateflow Linking Anchor attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---



**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 Object Summary Table component reference page.

---

**Insert anything into report?**

Yes. A link, and possibly text, depending on attribute choices.

**Filename**

`csflinktarget`

# Stateflow Name (Chart)

Component	Stateflow Name (Chart)
Category	Stateflow
Description	<p>The Stateflow Name (Chart) component inserts the name of the Stateflow object specified by its parent component into the report. This component must have the Stateflow Loop, Chart Loop, or Object Report component as its parent.</p> <p>You can use this component as the first subcomponent of a Chapter/Subsection component, which allows the current Stateflow object name to be the chapter or section title.</p>
Attributes	This is the Stateflow Name (Chart) attribute page.

Stateflow Name (Chart)

Render name as:

/Sample/ myChartName

☒ Name

☐ Type Name

☐ Type - Name

☐ Type: Name

☐ Show Stateflow path

☐ Show Simulink and Stateflow path

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

### Render name as:

Select one of the options listed below to display the Stateflow object name in the report:

- **Name** - e.g., Object
- **Type Name** - e.g., Object <ObjectName>
- **Type - Name** - e.g., Object - <ObjectName>
- **Type: Name** - e.g., Object: <ObjectName>

### Show Stateflow path

Select this option to show the Stateflow path for a Stateflow object. For example, if you choose this option for a transition, the transition name will appear as: <Machine Name>/<Chart Name>/<State Name>/<Transition>.

### Show Simulink and Stateflow path

Select this option to show the full Simulink and Stateflow path name for a Stateflow object. This option is only available if you choose the **Show Stateflow path** option. If you choose this option for a state, the state name will include the entire Simulink path and appear as:

<Model (or Machine) Name>/<System Name>/<Block (or Chart) Name>/<State Name>.

**Insert anything  
into report?**

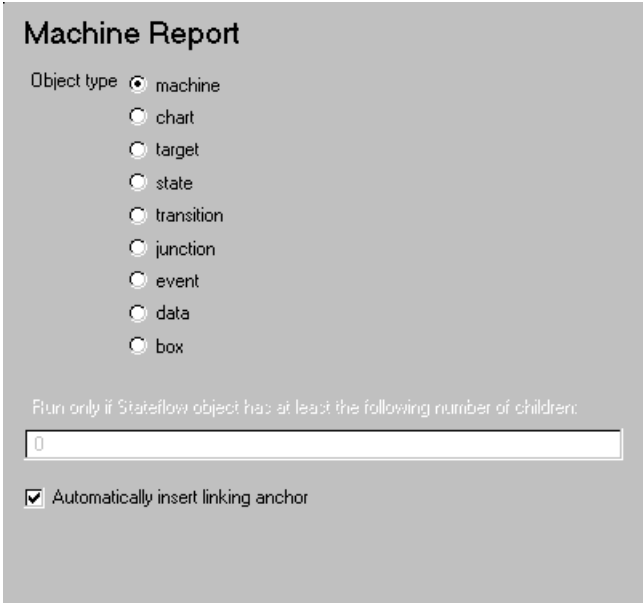
Yes. Text.

**Filename** csfobj name

# Object Report

---

Component	Object Report
Category	Stateflow
Description	The Object Report component is a filter that runs its children only if the current object type (set by the parent Stateflow Loop) matches the selected object type. The Object Report component must have the Stateflow Loop component as its parent.
Attributes	This is the Object Report attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

**Object type**

Select the Stateflow object type to include in the report. The current object type (set by the parent Stateflow 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 are reported on. If you enter a number in this field, a Stateflow object will not be included in the report unless it has at least that number of children.

**Automatically insert linking anchor:**

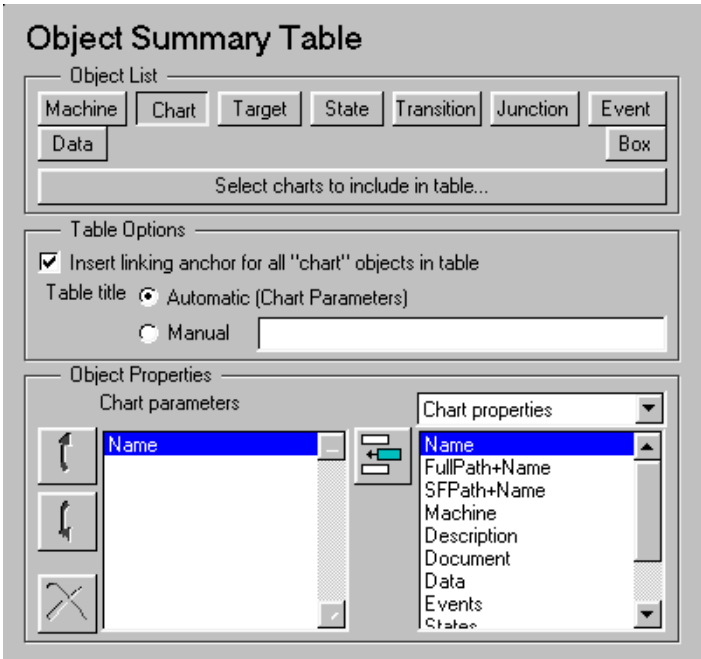
Select this option to insert a linking anchor in front of the reported object. If an anchor for this object already exists (inserted by the Stateflow Linking Anchor component), a second anchor will not be created.

**Insert anything  
into report?** No.

**Filename** csf\_obj\_report

# Object Summary Table

Component	Object Summary Table
Category	Stateflow
Description	<p>The Object Summary Table component shows properties or parameters of the selected Stateflow objects in a table. The Object Summary Table Component can have the following parents:</p> <ul style="list-style-type: none"><li>• Chart Loop</li><li>• Any Simulink looping component (Model Loop, System Loop, Block Loop, or Signal Loop)</li></ul>
Attributes	This is the Object Summary Table attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

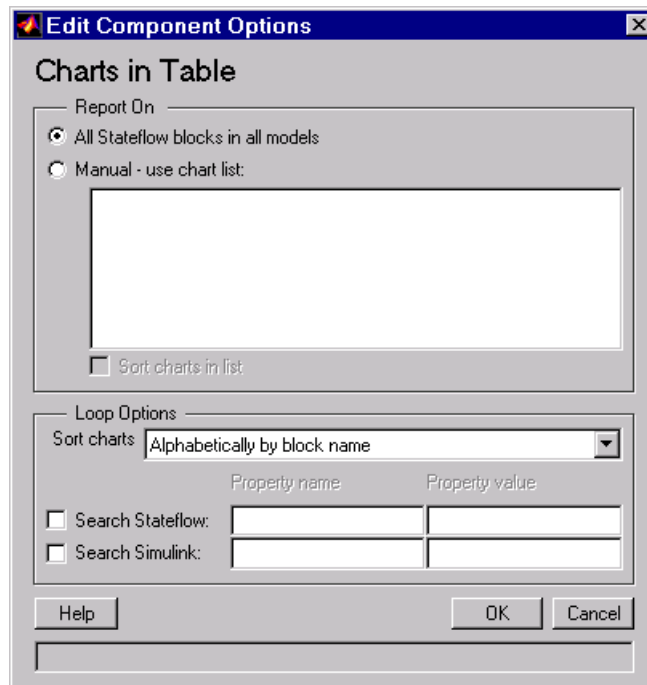
## Object List

Select a Stateflow object to report on. The Report Generator will report on the the Stateflow objects you choose that are contained in the charts you specify.

To edit the list of Stateflow charts, you can click on the edit bar (**Select charts to include \_\_\_\_**). Note that this option is not available when the Object Summary Table has the Chart Loop as its parent.

The **Edit Component Options** window appears when you click on the edit bar, as shown below.

# Object Summary Table



Note that the specifying the filtering and looping options in the **Edit Component Options** window modifies the list of charts whose children (Stateflow objects) are included in the report, not the list of Stateflow objects themselves.

## Report On

**Auto** - \_\_\_\_\_

Select this option to report on all charts in the current context. The context is set by the Chart Loop component's parent component:

- If the parent component is the Model Loop, then selecting this option causes the Object Summary Table component to report on the specified Stateflow objects in all reported Stateflow charts in the current model.



- If the parent component is the System Loop, then selecting this option causes the Object Summary Table component to report on the specified Stateflow objects in all Stateflow charts in the current system.
- If the parent component is the Signal Loop, then selecting this option causes the Object Summary Table component to report on the specified Stateflow objects in all Stateflow charts connected to the current signal.
- If the parent component is the Block Loop, then selecting this option causes the Object Summary Table component to report on the specified Stateflow objects in current Simulink block if it is a Stateflow chart.
- If the Object Summary Table does not have the Model Loop, System Loop, Signal Loop, or Block Loop component as its parent, then selecting this option causes the Chart Loop component to report on all the specified Stateflow objects in all Stateflow charts in all open models. In this case, the **Select charts to include** \_\_\_\_ option is not available.

All of the other attribute options are the same options found in the Chart Loop component. See the Chart Loop component reference page for more information on these options.

# Object Summary Table

---

## Table Options

### Insert linking anchor for all <objects> in table

This context-sensitive option inserts a linking anchor for all blocks, signals, systems, or models that are listed in the table.

Note that you should use this option if you already have an Stateflow Linking Anchor component that creates an anchor for an object in the object summary table.

### Table title

Select a table title option:

- **Automatic (<Object> Parameters):** for example, if you are reporting on a state, the table title will be the State Parameters.
- **Manual:** if you want to create your own table title, select this option and enter a table title in the corresponding field. 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. Please see “%<VariableName> Notation” in the Text component reference page for more details.

## Object Properties

Select which object parameters are included in the report in this section of the attribute page.





### <Object> Parameters

This list box shows all of the object parameters that are included in the report. To add a parameter to this list, select the parameter category from the drop box on the left, and then select parameters from the list box below it. Then click on the add button, shown in the table below.

With the buttons shown in the table below, you can:

- Move a parameter up in the list
- Move a parameter down in the list
- Remove a parameter from the list

- Add a parameter to the list

Use this button	To do this
	To move a parameter up in the list
	To move a parameter down in the list
	To remove a parameter from the list
	To add a parameter to the list

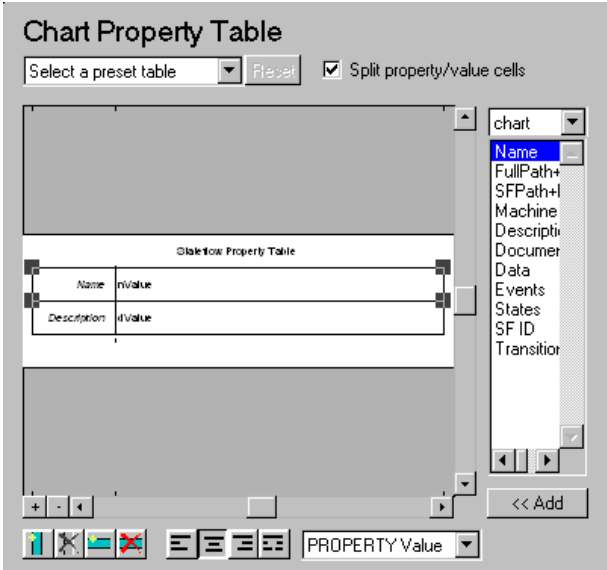
**Insert anything into report?** Yes. Table.

**Filename** csl\_summ\_table

# Property Table

---

Component	Property Table
Category	Stateflow
Description	The Property Table component inserts a property table in your report (reports on property-value pairs). This component inserts a property-value table for a Stateflow object into the report. This object is defined by the Object Report component, which must be the parent of the Property Table.
Attributes	This is the Property Table attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

The Property Table is similar to the Handle Graphics Figure Property Table. See the Handle Graphics Figure Property Table for reference.

Note that available properties and preset tables depend on the Object Report parent component.

**Insert anything into report?** Yes. Table.

**Filename** csf\_prop\_table

# Stateflow Snapshot (chart)

Component	Stateflow Snapshot (chart)
Category	Stateflow
Description	<p>The Stateflow Snapshot (chart) component inserts a snapshot of a Stateflow object (defined by the Object Report component) into the report. The Stateflow Snapshot (chart) component must have the Object Report as its parent.</p> <p>This component only executes if the selected object in the Object Report is a graphical object. The following are graphical Stateflow objects:</p> <ul style="list-style-type: none"><li>• Chart</li><li>• State</li><li>• Transition</li><li>• Frame</li></ul>
Attributes	This is the Stateflow Snapshot (chart) attribute page.

Stateflow Snapshot (chart)

Image file formatAutomatic (PNG - Portable Network Graphics 24-bit imag

Run only if Stateflow object has at least the following number of children:  
0

☒ Include callouts to describe visible objects

Image Size

☒ Attempt to shrink image to minimum font size set in Stateflow Loop  
☐ As specified (maintain aspect ratio)

Maximum size500x300points

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

### Image file format

Choose the image file format (e.g., JPEG, TIFF, etc.) from this drop box. To automatically choose the format best suited for the output format that you chose in the Report component, select **Automatic**, which is the default option. Otherwise, make sure that you choose an image format that your output viewer can read.

### 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 Object Report component. This option is not active unless the selected object in the Stateflow Object Report (the parent component) is a graphical object.

This option lets you decrease the size of the report by excluding certain images. You may want to use this option when you have a large model.

### Include callouts to describe visible objects

Choose whether or not to include visible callouts to describe visible objects. The default is to include callouts.

### Image Size

To avoid excessively large images, choose the **Attempt to shrink image to minimum legible font size (set in Stateflow Loop)** option. The image is displayed as small as possible while keeping the font no smaller than the font size you specify in the **Minimum legible font size** option in the Stateflow Loop attribute page.

To specify an image size, choose the **As specified (maintain aspect ratio)** option. If you select this option, the image is displayed according to your specifications, while the aspect ratio is maintained. Thus, the image will fit at

# Stateflow Snapshot (chart)

---

least one of the dimensions that you specify. The other dimension will change to keep the aspect ratio constant.

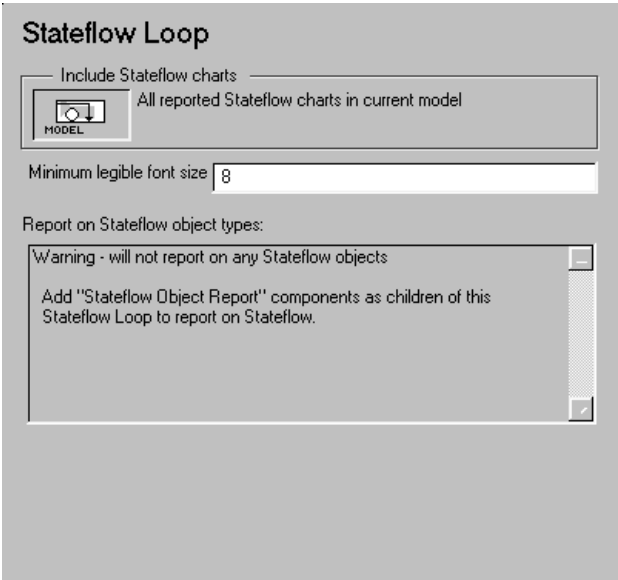
Specify the image size in the form wxh (width times height) in the **Maximum Size** field and the units in the corresponding drop box. The image size will not exceed these dimensions.

**Insert anything into report?**      Yes. Image.

**Filename**                      csf\_snapshot



Component	Stateflow Loop
Category	Stateflow
Description	<p>The Stateflow Loop component runs its children for the Simulink model, system, or block defined by its parent component. For example, if the Stateflow Loop is parented by the Simulink Model Loop, it will run its children for all Stateflow objects in the Simulink model. The Stateflow objects are specified by the Object Report, which must be a child of the Stateflow Loop.</p> <p>The Stateflow Loop component must have the Model Loop, System Loop, or Block Loop component as its parent.</p>
Attributes	This is the Stateflow Loop attribute page.



---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Include Stateflow Charts

This context-sensitive frame describes what the Stateflow Loop component is operating on:

- If the Stateflow Loop has the Model Loop as its parent, it includes all Stateflow charts in the current model.
- If the Stateflow Loop has the System Loop as its parent, it includes all Stateflow charts in the current system.
- If the Stateflow Loop has the Block Loop as a parent, it includes the current Simulink block if it is a Stateflow chart.
- If the Stateflow Loop has the Signal Loop as its parent, it includes all Stateflow chart connected to the current signal.
- If the Stateflow Loop does not have any of the Simulink looping components as its parent, it includes all Stateflow charts in all models.

## Minimum legible font size

Use this field to enter the minimum font size that can be used in an image. This attribute ensures that each reported Stateflow object with textual information has legible text in at least one generated image.

You can use this option to determine whether a certain Stateflow object will be reported on. If the font size in an object is smaller than the size you specify in this field, then that object will be included in the report.

To use this option for the Stateflow Snapshot (chart) component, choose the **Attempt to shrink image to minimum legible font size (set in Stateflow Loop)** option in the Stateflow Snapshot attribute page. The image is displayed as small as possible while keeping the font no smaller than the font size you specify in the **Minimum legible font size** option.

**Report on Stateflow object types:**

This list box shows all Stateflow object types that are included in the report. These object types are specified by child Object Report components.

For example, if the Stateflow Loop has the following four Stateflow Object Report components as children

- Stateflow machine
- Stateflow chart
- Stateflow event
- Stateflow data

then the **Report on Stateflow object types:** list box will list these Stateflow objects.

**Insert anything into report?** No.

**Filename** csf\_hier\_loop

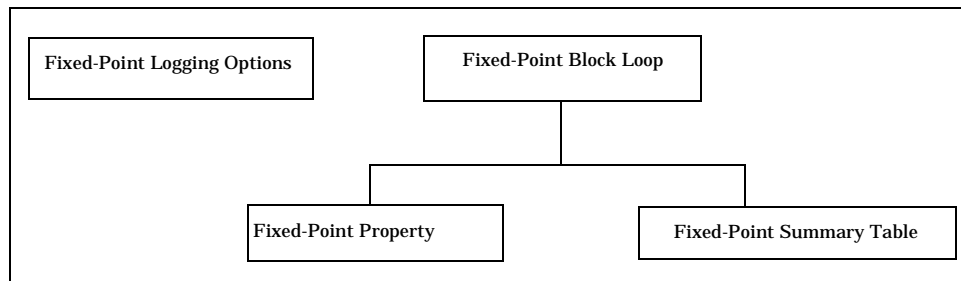
# Fixed-Point Components

---

## Fixed-Point Components

You can use Fixed-Point components to include information on fixed-point blocks in your report.

Fixed-Point components have the following parent/child relationships.



**Fixed-Point Block Loop.** This 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 will run its children for all fixed-point blocks in the Simulink model. The Fixed-Point Block Loop component may have either the Simulink Model Loop, System Loop, or Signal Loop component as its parent.

**Fixed-Point Logging Options.** This component sets fixed-point options similar to those set in the Fixed-Point Blockset Interface GUI. You can use this component before a Model Simulation component in order to control how the model behaves during simulation.

**Fixed-Point Property Table.** This component 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.** This component 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.

Component	Fixed-Point Block Loop
Category	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 will run its children for all fixed-point blocks in the Simulink model. The Fixed-Point Block Loop component may have either the Simulink Model Loop, System Loop, or Signal Loop component as its parent.
Attributes	This is the Fixed-Point Block Loop attribute page.

Fixed-Point Block Loop

Report On

☒ Auto - Fixed-Point blocks in all models

☐ Manual - use block list:

Loop Options

Sort blocks

Alphabetically by Fixed-Point block name

Report on

All Fixed-Point blocks (logged or unlogged)

☐ Search for property name/property value pairs

Property name

Property value

# Fixed-Point Block Loop

---

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Report On

This frame describes what the Fixed-Point Block Loop component is operating on.

### Auto - Fixed-Point blocks in \_\_\_\_\_

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, then this component will report on all fixed-point blocks in the current system.

If the Fixed-Point Block Loop component does not have either the Simulink Model Loop, System Loop, Block Loop, or Signal Loop component as its parent, then selecting this option causes the component to report on all fixed-point blocks in all models.

### Manual - 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** 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. Please see “%<VariableName> Notation” in the Text component reference page for more details.

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 the following.

`Z={ 'fxpdemo_feedback/D2A' }`

then the Report Generator will report on the following blocks.

A2D

D2A

### Loop Options

You can choose block sorting options and reporting options in this frame.

#### Sort blocks

Use this option to select how the blocks are sorted (this option is available if you select the **Auto - Fixed-Point blocks in \_\_\_\_\_** option in the **Report On** section or if you select **Manual - use block list** and the **Sort blocks** option):

- Select **Alphabetically by fixed-point 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 simulation order (req RTW)** to sort blocks by execution order. This option requires Real-Time Workshop; if you do not have Real-Time Workshop, this option will not appear in the attribute page.

# Fixed-Point Block Loop

---

## Report on

Use this option to select which blocks are reported on.

- Select **Logged fixed-point blocks only** to report on logged fixed-point blocks (dol og=' on' ).

To see if a fixed-point block is logged, double click on the block to open the **Block Parameters** dialog box. If the **Log minimums and maximums** check box is selected, then the fixed-point block is logged, i.e., minimum and maximum simulation values are logged to the workspace. See the Fixed-Point Blockset documentation for more information about overriding with doubles.

- Select **Unlogged fixed-point blocks only** to report on unlogged fixed-point blocks that have (dol og=' off' ).
- Select **All fixed-point blocks (logged or unlogged)** to report on all fixed-point blocks, regardless of whether they are logged.

## Search for property name/property value pairs

Select this option to report only on fixed-point blocks with certain property name/property value pairs. Enter the property name and the property value in the appropriate fields.

If you have Release 12 or greater of MATLAB, the **Property value** field is always read as a regular expression. Therefore, if you want to have an exact string match, you must enter the string in this way: \<string>. See `find_system` for more information on regular expressions.

If you have an earlier release of MATLAB, the **Property value** field is always read as a string.

Insert anything  
into report?

No.

Filename

cfp\_block\_loop

Fixed Point Logging Options



<b>Component</b>	Fixed-Point Logging Options
<b>Category</b>	Fixed-Point
<b>Description</b>	<p>The Fixed-Point Logging Options component sets fixed-point options similar to those set in the Fixed-Point Blockset Interface GUI. You can set the following options with this component:</p> <ul style="list-style-type: none"><li>• Logging options</li><li>• Doubles override</li><li>• Log mode</li></ul> <p>You can use this component before a Model Simulation component in order to control how the model behaves during simulation.</p>
<b>Attributes</b>	This is the Fixed-Point Logging Options attribute page.

The screenshot shows the 'Fixed-Point Logging Options' GUI. It has a title bar and three main sections, each with a title and a list of radio button options:

- Log Min & Max**:
  - ☐ All Fixed-Point blocks
  - ☐ No logging
  - ☒ Use block "dolog" parameter
- Doubles Override**:
  - ☐ All Fixed-Point blocks
  - ☐ No doubles
  - ☒ Use block "DbtOver" parameter
- Log Mode**:
  - ☒ Override log
  - ☐ Merge log

# Fixed-Point Logging Options

---

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Log Min & Max

You can select logging minimum and maximum options in this section. If a fixed-point block is logged, minimum and maximum simulation values are logged to the workspace. See the Fixed-Point Blockset documentation for more information about logging simulation results.

### All fixed-point blocks

Select this option to log all fixed-point blocks.

### No logging

Select this options if you do not want to any fixed-point block to be logged.

### Use block “dolog” parameter

Select this option to use the fixed-point blocks’ dol og parameters.

## Doubles Override

You can select whether you want to override the fixed-point blocks’ parameters and output data types with doubl es, i.e, represent the parameters and outputs using double precision floating-point numbers and perform all calculations with floating point arithmetic. See the Fixed-Point Blockset documentation for more information about overriding with doubles.

### All fixed-point blocks

Select this option to override the output data type with doubl es.

### No doubles

Select this option to have no output data types that are doubl es.

### Use block “DblOver” parameter

Select this option to use the fixed-point blocks’ Dbl Over parameters.

## Log Mode

Select how you want logs to be recorded in this section.

### Override log

Select this option if you want all simulation logs to overwrite the previous simulation logs.

### Merge log

Select this option if you want to append all simulation logs to the previous simulation logs.

**Insert anything  
into report?**

No.

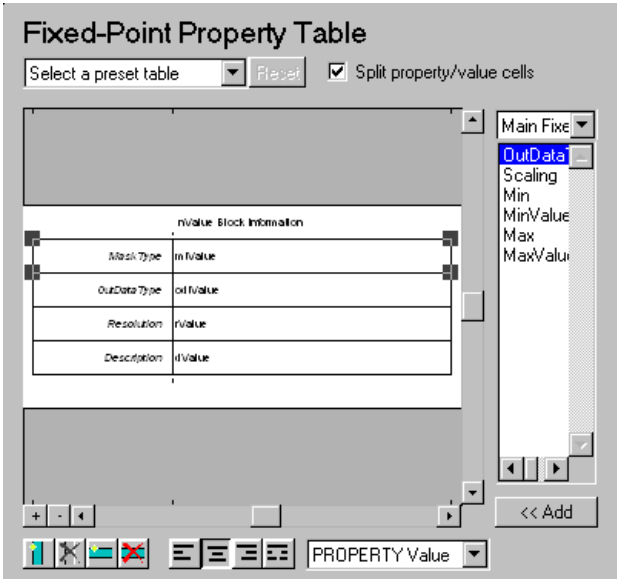
**Filename**

cfp\_opti ons

# Fixed-Point Property Table

---

Component	Fixed-Point Property Table
Category	Fixed-Point
Description	The Fixed-Point Property Table inserts a table that reports on fixed-point block property name/property value pairs.
Attributes	This is the Fixed-Point Property Table attribute page.



The Fixed-Point Property Table component is similar to the Handle Graphics Figure Property Table component. See the Handle Graphics Figure Property Table component reference page.

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

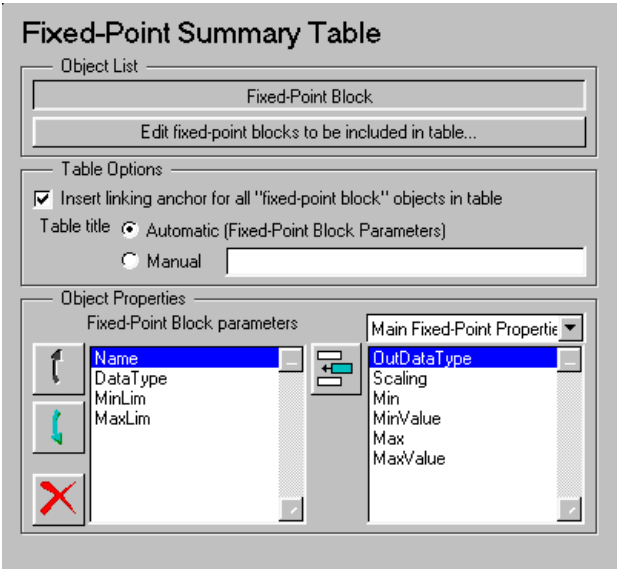
Insert anything into report? Yes. Table.

Filename cfp\_blk\_proptable

# Fixed-Point Summary Table

---

Component	Fixed-Point Summary Table
Category	Fixed-Point
Description	The Fixed-Point Summary Table component shows properties or parameters of the selected fixed-point blocks in a table.
Attributes	This is the Fixed-Point Summary Table attribute page.



The Fixed-Point Summary Table component is similar to the Simulink Object Summary Table component, except that this component reports on fixed-point blocks only; there are no other reporting options. See the Simulink Object Summary Table component reference page.

---

**Note** To view a component attribute page in the Setup File Editor, select the component in the setup file outline on the left and click on the **Options** tab. If the component is not in the setup file outline, you must add it to the outline first. See Chapter 4, “Building and Editing the Setup File” for a discussion on how to add a component to the outline.

---

## Fixed-Point Summary Table

---

**Insert anything  
into report?**    Yes. Table.

**Filename**            cfp\_summ\_table

# Fixed-Point Summary Table

---



## Symbols

% notation 6-29

## A

activating components 4-7

Add Components tab 2-9

adding components 4-4

Att. name field 5-15

Attribute data type 5-32

Attribute name 5-31

attribute UI control 5-32

attribute.m 5-26

## B

Block Automatic Property Table component  
6-108

Block Execution Order component 6-111

Block Loop component 6-115

Block Property Table component 6-119

Block Type: Look-Up Table component 6-127

Block Type:Bus component 6-121

## C

Category ID field 5-5

Category name field 5-5

Cell Table component 6-8

ch1-tutorial 1-3

changing a previously created attribute 5-33

changing the outline string 5-34

Chapter/Subsection component 6-12

Chart Loop component 6-194

child components 2-8, 4-3

Comment component 6-88

component 1-7, 1-9, 2-8, 5-2

component attributes 3-2

Component Attributes page 5-12

Component can contain subcomponents option  
5-10

Component category list box 5-5

Component Category page 5-5

Component Creation Wizard 5-2, 5-3

Component Function Name field 5-9

Component Methods page 5-19

Component Name page 5-8

compwi z command 5-3

Confirm Selections page 5-23

Control type drop box 5-14

Convert an SGML File UI 2-21

converting a generated report into another format  
2-21

converting a report 2-22

copying a component 4-16

Create custom attributes method option 5-19

Create custom outlinestring method option 5-21

Create Log File option 2-12

creating components 5-2

creating log files 2-24

crp\_randplot.m 5-26

cutting a component 4-16

## D

Data type drop box 5-13

deactivating components 4-7

Default value field 5-14

deleting components 4-6

Description field 5-10

Display conversion status messages in command  
window option 2-22

displaying component attributes 3-2

**E**

Edit menu 2-10  
Else component ({If} Else) 6-58  
Elseif component ({If} Elseif) 6-61  
Empty component 6-90  
Evaluate MATLAB Expression component 6-76  
execute.m 5-2, 5-25, 5-27

**F**

Fieldname field 5-13  
figure loop- tutorial 3-4  
Figure “For” Loop component 3-3, 6-36  
figure loop tutorial 3-3  
Figure Property Table component 3-14, 6-39  
File menu 2-10  
Fixed-Point Block Loop component 6-219  
Fixed-Point Logging Options component 6-223  
Fixed-Point Property Table component 6-226  
Fixed-Point Summary Table component 6-228  
Flow control looping component 3-3  
For Loop component 6-64  
Formatting components 6-5

**G**

Generate a Report UI 2-4  
generating a report 1-5  
Generation Status message priority level 5-38  
Generation Status tab 2-9, 2-12, 2-20  
getinfo.m 5-25, 5-27  
Graphics Figure Snapshot component 6-48  
Graphics Object Name component 6-51

**H**

halting report generation 2-11

Handle Graphics components 6-34  
Handle Graphics Parameter component 6-54  
HandleVisibility 3-7  
Help menu 2-10

**I**

Image component 6-15  
Import File component 6-92  
inherited attribute page 5-36  
Insert Variable component 6-78  
invisible figure 3-6

**L**

Link component 6-18  
List component 6-21  
list of figures 2-18  
list of tables 2-18  
list of titles 2-18  
log file 2-12, 2-24  
Logical and Flow components 6-56  
Logical If component 6-67  
looping component 3-3

**M**

MATLAB components 6-75  
MATLAB Report Generator 1-2, 4-2, 6-2  
MATLAB variables  
    using with components 6-29  
MATLAB/Toolbox Version Number component 6-81  
menu bar 2-10  
message priority levels 2-20  
Model Change Log component 6-134  
Model Functions component 6-138

Model Loop component 6-141  
Model Property Table component 6-149  
Model Simulation component 6-151  
Model Variables component 6-154  
moving components 4-8

## N

Name field 5-9  
naming the report file 2-14  
Nest Setup File component 6-96

## O

Object Linking Anchor component 6-158  
Object looping component 3-3  
Object Name component 6-160  
Object Property component 6-163  
Object Report component 6-202  
Object Summary Table component 6-166, 6-204  
Options tab 2-9, 3-2  
out. att. XXX 5-29  
out. attx. XXX 5-29  
outlinestring. m 5-25, 5-26  
output options 2-12

## P

Paragraph component 6-26  
parent component 4-3  
Parent directory field 5-9  
Print stylesheets 2-18  
Property Table (Stateflow) component 6-210  
property table components 3-14

## R

removing a component 4-16  
report 1-10  
report command 2-4  
Report Conversion UI 2-21  
Report Description field 2-19  
Report Generator 1-10  
Report Generator components 6-87  
Report Options component 1-7, 6-3  
report output 2-14  
report output format 2-15  
report output format descriptions 2-17  
rpt component 5-19  
rptconvert command 2-21, 2-23  
rptlist command 1-3, 2-5

## S

Save setup file on generation option 2-19  
Scope Snapshot component 6-131  
setup file 1-3, 1-10, 2-2, 2-7  
Setup File Editor 1-9, 2-2, 2-7  
Setup File List 1-3, 2-2, 2-5  
setup file outline 4-3  
SGML source filename 2-21  
Show status updates drop box 2-20  
Signal Loop component 6-173  
Signal Property Table component 6-176  
Simulink components 6-104  
Simulink Report Generator 1-2, 4-2, 6-2  
specifying the directory for the report file 2-14  
Stateflow components 6-192, 6-218  
Stateflow Linking Anchor component 6-198  
Stateflow Loop component 6-215, 6-219  
Stateflow Name (Chart) component 6-200  
Stateflow Snapshot (chart) component 6-212  
Stop Report Generation component 6-99

stylesheet 2-15  
stylesheet options 2-15  
subasgn. m 5-26  
subcomponent 2-8  
subsref. m 5-26  
System Filter component 6-178  
System Hierarchy component 6-180  
System Loop component 6-183  
System Property Table component 6-186  
System Snapshot component 6-188

## T

tag 3-7  
Text component 6-28  
Then component ({If} Then) 6-70  
Time/Date Stamp component 6-101  
Title Page component 6-31  
typographical conventions 1-12

## U

undo (Edit menu) 4-3  
undoing an attribute edit 3-2  
unreadable image files 2-22

## V

Variable Table component 6-84  
variables  
    using with components 6-29  
verifying component creation 5-24  
View report after converting option 2-22  
View report after generating option 1-5, 2-18  
viewing a report 1-5

## W

Web stylesheets 2-17  
While Loop component 6-73  
Window menu 2-10



