

SimBiology[®] Release Notes

How to Contact The MathWorks



www.mathworks.com
comp.soft-sys.matlab
www.mathworks.com/contact_TS.html

Web
Newsgroup
Technical Support



suggest@mathworks.com
bugs@mathworks.com
doc@mathworks.com
service@mathworks.com
info@mathworks.com

Product enhancement suggestions
Bug reports
Documentation error reports
Order status, license renewals, passcodes
Sales, pricing, and general information



508-647-7000 (Phone)



508-647-7001 (Fax)



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

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

SimBiology[®] Release Notes

© COPYRIGHT 2005–2009 by The MathWorks, Inc.

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

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

Trademarks

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

Patents

The MathWorks products are protected by one or more U.S. patents. Please see www.mathworks.com/patents for more information.

Summary by Version	1
Version 3.1 (R2009b) SimBiology Software	4
Version 3.0 (R2009a) SimBiology Software	7
Version 2.4 (R2008b) SimBiology Software	11
Version 2.3 (R2008a) SimBiology Software	22
Version 2.2 (R2007b+) SimBiology Software	25
Version 2.1.2 (R2007b) SimBiology Software	38
Version 2.1.1 (R2007a) SimBiology Software	39
Version 2.1 (R2006b+) SimBiology Software	40
Version 2.0.1 (R2006b) SimBiology Software	43
Version 2.0 (R2006a+) SimBiology Software	44
Version 1.0.1 (R2006a) SimBiology Software	49
Version 1.0 (R14SP3+) SimBiology Software	50
Compatibility Summary for SimBiology Software	56

Summary by Version

This table provides quick access to what's new in each version. For clarification, see "Using Release Notes" on page 2.

Version (Release)	New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Latest Version V3.1 (R2009b)	Yes Details	Yes Summary	Bug Reports Includes fixes	Printable Release Notes: PDF Current product documentation
V3.0 (R2009a)	Yes Details	No	Bug Reports Includes fixes	No
V2.4 (R2008b)	Yes Details	Yes Summary	Bug Reports Includes fixes	No
V2.3 (R2008a)	Yes Details	Yes Summary	Bug Reports Includes fixes	No
V2.2 (R2007b+)	Yes Details	Yes Summary	Bug Reports Includes fixes	No
V2.1.2 (R2007b)	No	No	Bug Reports	No
V2.1.1 (R2007a)	No	No	Bug Reports	No
V2.1 (R2006b+)	Yes Details	Yes Summary	Bug Reports Includes fixes	No
V2.0.1 (R2006b)	No	No	Bug Reports Includes fixes	No
V2.0 (R2006a+)	Yes Details	Yes Summary	Bug Reports Includes fixes	No
V1.0.1 (R2006a)	Yes Details	No	Bug Reports	No
V1.0 (R14SP3+)	Yes Details	No	Bug Reports	No

Using Release Notes

Use release notes when upgrading to a newer version to learn about:

- New features
- Changes
- Potential impact on your existing files and practices

Review the release notes for other MathWorks™ products required for this product (for example, MATLAB® or Simulink®). Determine if enhancements, bugs, or compatibility considerations in other products impact you.

If you are upgrading from a software version other than the most recent one, review the current release notes and all interim versions. For example, when you upgrade from V1.0 to V1.2, review the release notes for V1.1 and V1.2.

What Is in the Release Notes

New Features and Changes

- New functionality
- Changes to existing functionality

Version Compatibility Considerations

When a new feature or change introduces a reported incompatibility between versions, the **Compatibility Considerations** subsection explains the impact.

Compatibility issues reported after the product release appear under Bug Reports at The MathWorks™ Web site. Bug fixes can sometimes result in incompatibilities, so review the fixed bugs in Bug Reports for any compatibility impact.

Fixed Bugs and Known Problems

The MathWorks offers a user-searchable Bug Reports database so you can view Bug Reports. The development team updates this database at release

time and as more information becomes available. Bug Reports include provisions for any known workarounds or file replacements. Information is available for bugs existing in or fixed in Release 14SP2 or later. Information is not available for all bugs in earlier releases.

Access Bug Reports using your MathWorks Account.

About Functions Being Removed

This section lists functions removed or in the process of being removed. Functions typically go through several stages across multiple releases before being completely removed. This provides time for you to make adjustments to your code.

- **Announcement** – The release notes announce the planned removal, but there are no functional changes; the function runs as it did before.
- **Warning** – When you run the function, it displays a warning message indicating it will be removed in a future release; otherwise the function runs as it did before.
- **Error** – When you run the function, it produces an error. The error message indicates the function was removed and suggests a replacement function, if one is available.
- **Removal** – When you run the function, it fails. The error message is the standard message when MATLAB does not recognize an entry.

Functions might be in a stage for one or more releases before moving to another stage. Functions are listed in the Functions Being Removed section only when they enter a new stage and their behavior changes. For example, if a function displayed a warning in the previous release and errors in this release, it appears on the list. If it continues to display a warning, it does not appear on the list because there was no change between the releases.

Not all functions go through all stages. For example, a function's impending removal might not be announced, but instead the first notification might be that the function displays a warning.

The release notes include actions you can take to mitigate the effects of a function's removal, such as adapting your code to use a replacement function.

Version 3.1 (R2009b) SimBiology Software

This table summarizes what's new in Version 3.1 (R2009b):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as Compatibility Considerations , below. See also Summary.	Bug Reports Includes fixes	Printable Release Notes: PDF Current product documentation

New features and changes introduced in this version are:

- “Increased Performance When Repeatedly Simulating a Model” on page 4
- “Enhanced Desktop Support for Scanning Using Monte Carlo Methods” on page 5
- “Desktop Support for Copy and Paste” on page 5
- “View Status of Parameter Fitting Task During Run” on page 5
- “Improved Usability for Model Building and Debugging” on page 5
- “Unit Conversion Compatibility Considerations” on page 6
- “Functions and Properties Being Removed” on page 6

Increased Performance When Repeatedly Simulating a Model

Many analysis tasks that involve repeatedly simulating a model now run faster. These tasks include parameter fits and scans, as well as repeatedly simulating the same model using different variants or setting different values for the `InitialAmount` of species, the `Capacity` of compartments, and the `Value` of parameters. Under these conditions, repeatedly simulating the model generates any applicable warnings only the first time. To display warnings again, use the verification methods described in “Verifying that the Model Has No Warnings or Errors” in the SimBiology® User’s Guide.

Enhanced Desktop Support for Scanning Using Monte Carlo Methods

Scanning analysis now includes additional support for Monte Carlo methods. You can specify sampling using the options `multivariate normal distribution` or `latin hypercube sample with a normal distribution`. The Statistics Toolbox™ is required for this functionality. For more information, see “Scanning Analysis” in the SimBiology user’s guide documentation.

Desktop Support for Copy and Paste

The SimBiology desktop now supports copying and pasting. Use the typical keyboard shortcuts or the context menus to execute these commands.

View Status of Parameter Fitting Task During Run

You can track the status of a parameter fit when using nonlinear mixed effects with the `sbiofitstatusplot` function at the command line or by selecting an option in the SimBiology desktop. For more information, see “Obtaining the Status of Fitting” (command line) or “Obtaining the Status of Fitting” (desktop).

Improved Usability for Model Building and Debugging

The SimBiology desktop now supports:

- Dynamically updated error indicators for variants, configuring plots, and defining scans — Shows correctly defined, incorrectly defined, and warning indicators (green, red, and yellow) for additional help with model debugging.
- M-Lint indicators — Wherever MATLAB code appears in the desktop, the indicators used by M-lint also appear.
- Help for user-identified MATLAB code — Select and use the context menu to find help on functions where MATLAB code appears in the desktop.

Unit Conversion Compatibility Considerations

Previously a model with no units specified could have unit conversion on or off. Now, a model with no units specified that has `UnitConversion` on shows an error. Set `UnitConversion` off for models without units.

Functions and Properties Being Removed

For more information about the process of removing functions, see “About Functions Being Removed” in “What Is in the Release Notes” on page 2.

Function or Property Name	What Happens When You Use Function or Property?	Use This Instead	Compatibility Considerations
<code>-flat</code> as an option for <code>getstoichmatrix</code> and <code>getadjacencymatrix</code>	Errors	Not applicable	Since support for submodels has been removed, this option no longer applies.

Version 3.0 (R2009a) SimBiology Software

This table summarizes what's new in Version 3.0 (R2009a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	No	Bug Reports Includes fixes	No

New features and changes introduced in this version are:

- “New Feature to Import, Visualize, and Statistically Analyze Clinical and Experimental Data” on page 7
- “New Functionality to Create Pharmacokinetic Models” on page 8
- “New Functionality to Fit Data and Estimate Parameters Using Nonlinear Mixed Effects” on page 8
- “New Diagnostic Plots for Individual and Population Fitting Results” on page 9
- “New Project Wizard to Add Data, Create Models, and Specify Tasks” on page 9
- “New simbiology Command to Open the SimBiology Desktop” on page 10
- “Enhanced Usability Features in the SimBiology Desktop” on page 10
- “New Demo for Pharmacokinetic Modeling” on page 10

New Feature to Import, Visualize, and Statistically Analyze Clinical and Experimental Data

You can import tabular data into the SimBiology desktop or the MATLAB Workspace. The supported file types are `.xls`, `.csv`, and `.txt`.

At the command line, you can process and visualize the data using command-line functions. In the SimBiology desktop, you can filter the raw data to suppress outliers, visualize data using MATLAB plots, and calculate

statistics to analyze the data. You can further choose to plot the imported data with any analysis task.

See “Importing Data” in the SimBiology User’s Guide for more information.

New Functionality to Create Pharmacokinetic Models

You can automatically generate pharmacokinetic (PK) models by specifying number of compartments, dosing type, and method of elimination. If you plan to use the MATLAB command line, see “Creating PK Models at the Command Line” in the SimBiology User’s Guide for more information.

If you plan to use the SimBiology desktop, the new Add Model wizard lets you automatically generate PK models in the desktop. See “Creating PK Models in the SimBiology Desktop Using a Wizard” in the SimBiology User’s Guide for more information.

In addition, in the SimBiology desktop you can start by creating a new project using the new Project Wizard which also lets you add data, create models, and add analysis tasks in a SimBiology project. See “New Project Wizard to Add Data, Create Models, and Specify Tasks” on page 9 for additional information.

New Functionality to Fit Data and Estimate Parameters Using Nonlinear Mixed Effects

You can perform both individual and population fits to grouped longitudinal data.

- Individual fit — Fit data using nonlinear least squares method, estimate parameters, and calculate residuals and the estimated coefficient covariance matrix.
- Population fit — Estimate the fixed effects and the random sources of variation on parameters, using nonlinear mixed-effects models.

You can use two classes of methods to approximate the likelihood of the model:

- Methods that directly estimate parameters using the likelihood function namely, LME and ReLME.

- Methods that linearize the likelihood function, namely, first-order approximation (FO) and first-order approximation at the conditional estimates ('FOCE').

For more information, see “Parameter Fitting in Pharmacokinetic Models” in the SimBiology User’s Guide.

New Diagnostic Plots for Individual and Population Fitting Results

In the SimBiology desktop, after fitting the data, the analysis generates diagnostic plots that show:

- The predicted time courses and observations for an individual or the population
- Observed versus predicted values
- Residuals versus time, group, or predictions
- Distribution of the residuals
- A box-plot for random effects or parameter estimates from individual fitting.

For more information, see “Visualizing Parameter Fitting Results and Generating Diagnostic Plots” in the SimBiology User’s Guide.

New Project Wizard to Add Data, Create Models, and Specify Tasks

The newly added Project Wizard in the SimBiology desktop lets you:

- Add data from text files, spreadsheets, or the MATLAB Workspace.
- Create models, including automatically generate pharmacokinetic models by specifying number of compartments, dosing type, and method of elimination.
- Specify analysis tasks to add to the project.

For an example of how to use the Project Wizard, see “Modeling Using the SimBiology Graphical User Interface” in the SimBiology Getting Started Guide.

New simbiology Command to Open the SimBiology Desktop

A new function, `simbiology`, has been added for enhanced usability in opening the SimBiology desktop. `simbiology` is equivalent to the `sbidesktop` function, which is also supported.

Enhanced Usability Features in the SimBiology Desktop

The following enhancements to the SimBiology desktop are included in this release:

- **Back** and **Forward** buttons to help with navigation between desktop panes
- In **Preferences**, the ability to choose default model tasks to add to a model when loading an SBML file or importing a model from the MATLAB Workspace

New Demo for Pharmacokinetic Modeling

There is a new demo showing pharmacokinetic modeling functionality (Modeling the Population Pharmacokinetics of Phenobarbital in Neonates). To see all demos, click **SimBiology demos** or type `demo('MATLAB', 'SimBiology')` at the command prompt.

Version 2.4 (R2008b) SimBiology Software

This table summarizes what's new in Version 2.4 (R2008b):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as Compatibility Considerations , below. See also Summary.	Bug Reports Includes fixes	No

New features and changes introduced in this version are:

- “Enhanced Usability with the Redesigned Reaction Pane” on page 11
- “Additional Support for Showing Usages and Generating Reports in the SimBiology Desktop” on page 12
- “Support for Specifying Additional Inputs in Custom Plot Types” on page 13
- “Edit Graphical Models Using the New Block Property Editor” on page 14
- “Manage and Share Libraries Using the New Library Explorer” on page 15
- “Additional Options for Renaming Compartments, Species, and Parameters” on page 17
- “Change in the Random Number Generator Used During Stochastic Simulations” on page 18
- “Functions and Properties Being Removed” on page 19

Enhanced Usability with the Redesigned Reaction Pane

In the SimBiology desktop, the redesigned reaction pane consolidates the procedure to configure kinetic law, rate parameters, rate species, and reaction rates on one screen.

Work Area - My Project \ Model Session - Heterotrimeric_G_Protein_wt \ SimBiology Model \ Reactions

Build Edit Bind Unbind

Enter Reaction: Add Delete

Name	Reaction	KineticLaw	ReactionRate
1 Receptor ligand interaction	$L + R \leftrightarrow RL$	MassAction	$k_{RL} * L * R - k_{RLm} * RL$
2 R synthesis/degradation	$R \leftrightarrow \text{null}$	MassAction	$k_{Rdo} * R - k_{Rs}$
3 RL degradation	$RL \rightarrow \text{null}$	MassAction	$k_{RD1} * RL$
4 Gprotein complex formation	$Gd + Gbg \rightarrow G$	MassAction	$k_{G1} * Gd * Gbg$
5 Gprotein activation	$G + RL \rightarrow Ga + Gbg + RL$	MassAction	$k_{Ga} * G * RL$
6 Gprotein Inactivation	$Ga \rightarrow Gd$	MassAction	$k_{Gd} * Ga$

Settings | **Description**

Reaction: Reversible

$L + R \leftrightarrow RL$

KineticLaw: Expression:

Map between KineticLaw Parameters and Parameter Names:

Kinetic Law Parameter	Parameter Name	Value	Scope	ValueUnits
1 Forward Rate Parameter	k _{RL}	3.32E-18	L + R ↔ RL	
2 Reverse Rate Parameter	k _{RLm}	0.01	L + R ↔ RL	

Map between KineticLaw Species and Species Names:

Kinetic Law Species	Species Name	InitialAmount	Scope	InitialAmountUnits
1 MassAction Species	L	6.022E17	unnamed	
2 MassAction Species	R	10000.0	unnamed	

ReactionRate:

Name:

Active (Select if the reaction is enabled during the simulation.)

Additional Support for Showing Usages and Generating Reports in the SimBiology Desktop

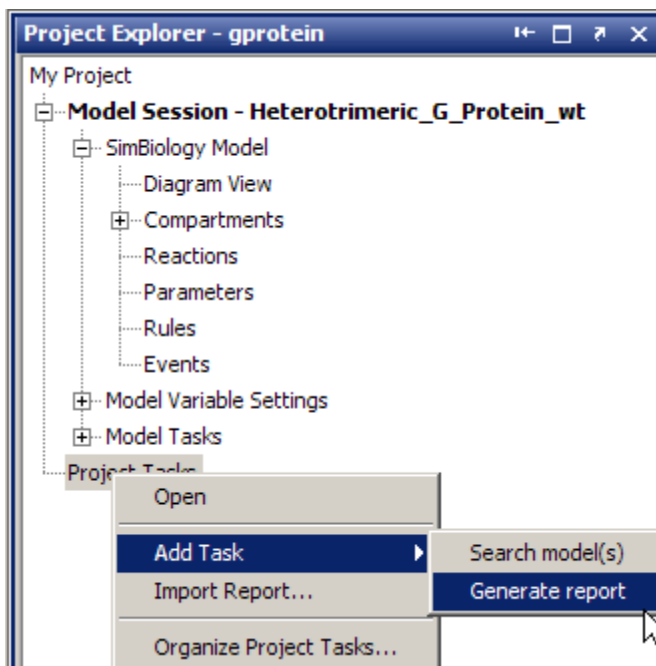
Additional Support for Showing Usages

The **Compartments** and **Species** panes now have added support for showing usages of compartments and species in a model. To show usages of a component, right-click (Windows®), or **Ctrl**+click (Macintosh®) the

compartment or species table, and select **Show Usages**. Support has also been added for showing usages from the **Diagram View**.

Additional Support for Generating Reports

You can now automatically populate report contents with one click in the **Report** pane. In the **Project Explorer**, right-click (Windows), or **Ctrl+click** (Macintosh) **Project Tasks** and select **Add Task > Generate report** to create a report. Select **Help > SimBiology Desktop Help** to see the context-sensitive help for information on how to generate reports. Click **AutoBuild** to populate the contents of your report automatically.



Support for Specifying Additional Inputs in Custom Plot Types

You can now specify additional inputs for plot types and define their types, default values, and ranges where applicable. For example, you can use this feature to extend the Time plot with an additional input to specify markers.

Compatibility Considerations

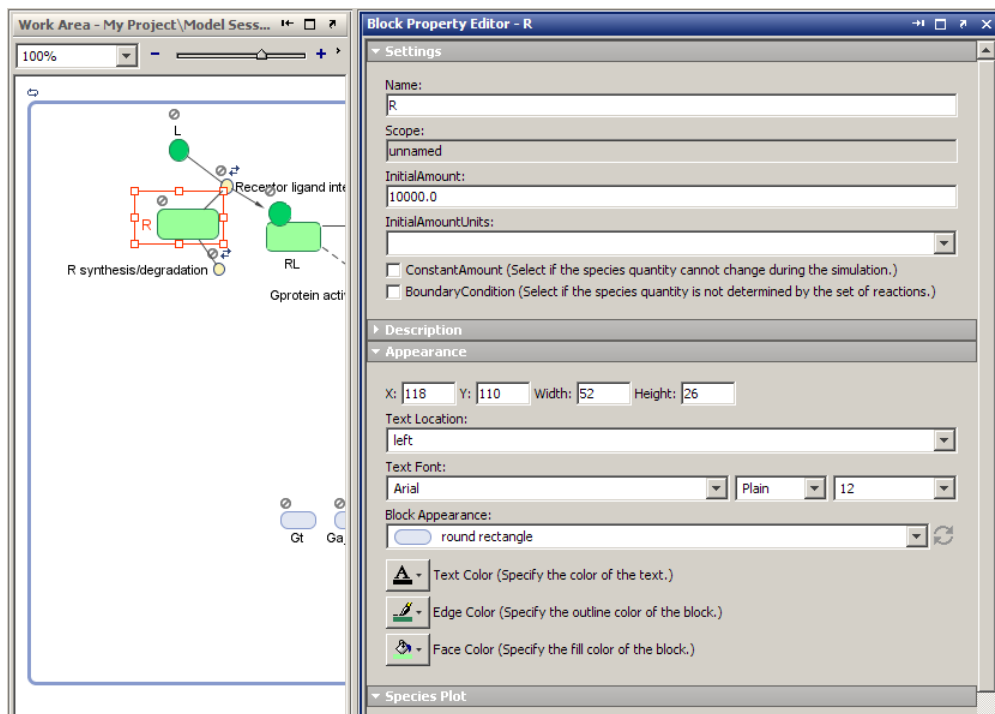
This consideration applies when you previously created custom plot types. Before, you could define the number of species or parameters that could be specified for x or y arguments. This functionality has been removed. You should modify your plot code to enforce this constraint. For an example of enforcing the constraint, see the code for the built-in plot type XY.

To access this code:

- 1** In the SimBiology desktop, select **Desktop > Library Explorer**. The **Library Explorer** opens.
- 2** Select **Plot Types**.
- 3** In the plot types table, select XY. The code section updates to show XY plot type code.

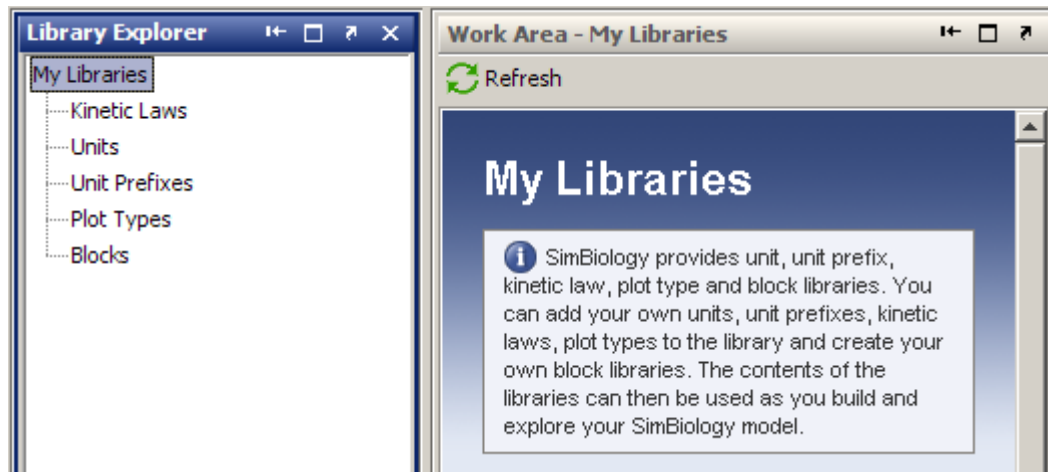
Edit Graphical Models Using the New Block Property Editor

The **Block Property Editor** is a tool that facilitates model building using the **Diagram View** pane by docking next to the diagram and allowing you to change properties of the selected block in the diagram. If you select multiple blocks, you can edit the properties that are common between each block. To open the **Block Property Editor**, select a block in the **Diagram View** pane, and then select **Diagram > Tools > Block Property Editor**.



Manage and Share Libraries Using the New Library Explorer

The **Library Explorer** adds the ability to add, modify, and share the contents of kinetic law, plot type, unit, unit prefix, and block libraries. To open the **Library Explorer**, in the SimBiology desktop, select **Desktop > Show Library Explorer**.



The **Library Explorer** shows all built-in and user-defined components, namely kinetic laws, units, unit prefixes, plot types, and blocks.

- **Kinetic Laws** — Contains kinetic laws that you can use as templates while creating a reaction rate expression.
- **Units** — Contains units that you can specify for compartment capacity, species amounts and parameter values, to do dimensional analysis and unit conversion during simulation.
- **Unit Prefixes** — Contains all unit prefixes that you can specify in combination with a valid unit for compartment capacity, species amounts and parameter values, to do dimensional analysis and unit conversion during simulation.
- **Plot Types** — Contains different types of plots that you can use with **Model Tasks** to visualize your results.
- **Blocks** — Contains blocks that you can use in the **Diagram View**.

For help, open the **Library Explorer** and select **Help > SimBiology Desktop Help** to see the context-sensitive help. Select a library in the **Library Explorer** to view information specific to the library.

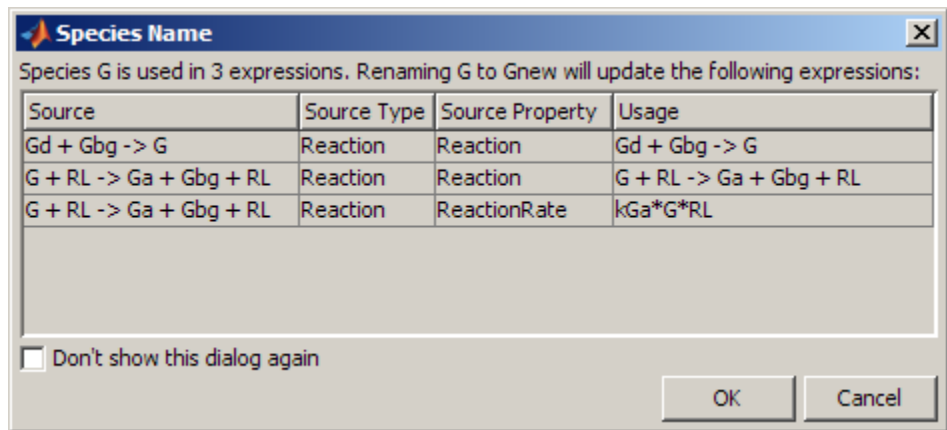
Additional Options for Renaming Compartments, Species, and Parameters

New Method for Renaming at the Command Line

The newly added rename method allows you to change the name of a compartment, species, or parameter, and update the name in expressions that refer to the component. Use the rename method instead of the set method to enable the name change and expression update.

New Options for Renaming in the SimBiology Desktop

You can specify how compartments, species, and parameter names should be updated in expressions. When you rename a compartment, species, or parameter, by default the names are set to be updated in all expressions that refer to the component being renamed. During renaming, if the component is used in one or more expressions, you will see a dialog box that tells you which expressions will be updated. For example, the following dialog box appears when you try to rename species G to Gnew.



You can choose not to see this dialog box for every instance of renaming. You can also set your preferences on whether expressions should be updated. To select preferences for renaming:

- 1 Select **File > Preferences**. The Preferences dialog box opens.

2 Click **Rename** to view and select default renaming options.

You can also specify whether to update expressions while renaming a component in the **Compartment**, **Species**, or **Parameter** panes. Right-click (Windows or Linux®) or **Ctrl+click** (Macintosh) the species, parameters, or compartment table and select **Rename**. This allows you to select whether to use the new name in all expressions referring to the component.

Compatibility Considerations

Previously, if you changed the name in a table or the diagram, expressions that used the components were not updated unless you selected **Rename in Expressions** from the context menu. Now the default is to change the name in expressions. But, you will see a dialog box that allows you to cancel the action if necessary, and you can specify the default as shown in “Additional Options for Renaming Compartments, Species, and Parameters” on page 17.

Change in the Random Number Generator Used During Stochastic Simulations

The default random number generator used during stochastic simulation has changed. The stochastic solver now uses the random numbers from the MATLAB default stream. When you run a model using a stochastic solver, and have set the `RandomState` property in the configuration set, you may see different simulation results relative to previous releases because the random numbers used may be different.

For more information about the change to the random number generator, see [Upgrade to Random Number Generator](#) in the MATLAB release notes.

Compatibility Considerations

If you never set the `RandomState` property, there should be no compatibility considerations. If however, you have previously set `RandomState` for your model and want to reproduce your previous results, type `rand('state',0);` at the command line before running the simulation. This sets the random number generator to the one used in previous releases.

Functions and Properties Being Removed

For more information about the process of removing functions, see “About Functions Being Removed” in “What Is in the Release Notes” on page 2.

Function or Property Name	What Happens When You Use Function or Property?	Use This Instead	Compatibility Considerations
sbioevent	Errors	addevent	Events must belong to a model. Replace all existing instances of sbioevent with addevent.
sbioparameter	Errors	addparameter	Parameters must belong to a model or a kinetic law. Replace all existing instances of sbioparameter with addparameter.
sbioreaction	Errors	addreaction	Reactions must belong to a model. Replace all existing instances of sbioreaction with addreaction.
<i>Model name</i> as an input argument for sbioroot	Errors	sbioselect	sbioroot does not accept a model name as an input argument. Use sbioselect to query models by name.
sbiorule	Errors	addrule	Rules must belong to a model. Replace all existing instances of sbiorule with addrule.

Function or Property Name	What Happens When You Use Function or Property?	Use This Instead	Compatibility Considerations
sbiospecies	Errors	addspecies	Species must belong to a compartment. Replace all existing instances of sbiospecies with addspecies.
sbioregisterunit	Errors	Unit object and sbioaddtolibrary	See “New Way to Add Units and Unit Prefixes” on page 34.
sbiounregisterunit	Errors	sbioremovefromlibrary	See “New Way to Add Units and Unit Prefixes” on page 34.
sbioregisterunitprefix	Errors	Unit object and sbioaddtolibrary	See “New Way to Add Units and Unit Prefixes” on page 34.
sbiounregisterunitprefix	Errors	sbioremovefromlibrary	See “New Way to Add Units and Unit Prefixes” on page 34.
BuiltInKineticLaws	Errors	BuiltInLibrary	See “Changes to the Library Structure in the Root” on page 31.
BuiltInUnitPrefixes	Errors	BuiltInLibrary	See “Changes to the Library Structure in the Root” on page 31.
BuiltInUnits	Errors	BuiltInLibrary	See “Changes to the Library Structure in the Root” on page 31.

Function or Property Name	What Happens When You Use Function or Property?	Use This Instead	Compatibility Considerations
UserDefinedKineticLaws	Errors	UserDefinedLibrary	See “Changes to the Library Structure in the Root” on page 31.
UserDefinedUnits	Errors	UserDefinedLibrary	See “Changes to the Library Structure in the Root” on page 31.
UserDefinedUnitPrefixes	Errors	UserDefinedLibrary	See “Changes to the Library Structure in the Root” on page 31.

Version 2.3 (R2008a) SimBiology Software

This table summarizes what's new in Version 2.3 (R2008a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes Summary	Bug Reports Includes fixes	No

New features and changes introduced in this version are:

- “Support for 64-Bit Microsoft Windows” on page 22
- “Functions and Properties Being Removed” on page 22

Support for 64-Bit Microsoft Windows

SimBiology software now has added support for 64-bit Windows (Win64).

Functions and Properties Being Removed

For more information about the process of removing functions, see “About Functions Being Removed” in “What Is in the Release Notes” on page 2.

Function or Property Name	What Happens When You Use Function or Property?	Use This Instead	Compatibility Considerations
sbioevent	Warns	addevent	Events must belong to a model. Replace all existing instances of sbioevent with addevent.

Function or Property Name	What Happens When You Use Function or Property?	Use This Instead	Compatibility Considerations
sbioparameter	Warns	addparameter	Parameters must belong to a model or a kinetic law. Replace all existing instances of sbioparameter with addparameter.
sbioreaction	Warns	addreaction	Reactions must belong to a model. Replace all existing instances of sbioreaction with addreaction.
<i>Model name</i> as an input argument for sbioroot	Errors	sbioselect	sbioroot does not accept a model name as an input argument. Use sbioselect to query models by name.
sbiorule	Warns	addrule	Rules must belong to a model. Replace all existing instances of sbiorule with addrule.
sbiospecies	Warns	addspecies	Species must belong to a compartment. Replace all existing instances of sbiospecies with addspecies.
sbioregisterunit	Errors	Unit object and sbioaddtolibrary	See “New Way to Add Units and Unit Prefixes” on page 34.
sbiounregisterunit	Errors	sbioremovefromlibrary	See “New Way to Add Units and Unit Prefixes” on page 34.

Function or Property Name	What Happens When You Use Function or Property?	Use This Instead	Compatibility Considerations
sbioregisterunitprefix	Errors	Unit object and sbioaddtolibrary	See “New Way to Add Units and Unit Prefixes” on page 34.
sbiounregisterunitprefix	Errors	sbioremovefromlibrary	See “New Way to Add Units and Unit Prefixes” on page 34.
BuiltInKineticLaws	Errors	BuiltInLibrary	See “Changes to the Library Structure in the Root” on page 31.
BuiltInUnitPrefixes	Errors	BuiltInLibrary	See “Changes to the Library Structure in the Root” on page 31.
BuiltInUnits	Errors	BuiltInLibrary	See “Changes to the Library Structure in the Root” on page 31.
UserDefinedKineticLaws	Errors	UserDefinedLibrary	See “Changes to the Library Structure in the Root” on page 31.
UserDefinedUnits	Errors	UserDefinedLibrary	See “Changes to the Library Structure in the Root” on page 31.
UserDefinedUnitPrefixes	Errors	UserDefinedLibrary	See “Changes to the Library Structure in the Root” on page 31.

Version 2.2 (R2007b+) SimBiology Software

This table summarizes what's new in Version 2.2 (R2007b+):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as Compatibility Considerations , below. See also Summary.	Bug Reports Includes fixes	No

New features and changes introduced in this version are:

- “Changes to the Model Structure” on page 25
- “Events” on page 27
- “Variants” on page 27
- “Support for Analysis Tasks in the Desktop” on page 28
- “Changes to the Library Structure in the Root” on page 31
- “New Features for Solvers and Simulation Settings” on page 32
- “New Plot Functions” on page 34
- “New Sensitivity Analysis Property for Species Outputs” on page 34
- “New Way to Add Units and Unit Prefixes” on page 34
- “Functions and Properties Being Removed” on page 35

Changes to the Model Structure

The following new features and changes apply to a model's structure:

- “Compartments Now Supported” on page 26
- “Submodel Support Will Be Removed” on page 26

Compartments Now Supported

SimBiology models now let you add compartments to a model, specify compartment size, simulate, and do analysis with multiple compartments. Any model containing species must have a compartment, and the species must belong to a compartment. You can also perform dimensional analysis and unit conversion accounting for the specified compartment size. For more information on compartments, refer to the following sources in the documentation:

- “Compartment” — Introduction to compartments in the SimBiology Getting Started Guide.
- `Compartment` object — Reference for compartment objects.

Submodel Support Will Be Removed

Support for submodels will be removed in a future release. You can still open models containing submodels and create submodels, but you cannot simulate or perform any analysis tasks on the model. Therefore, you should transition to using compartments where applicable.

Compatibility Considerations. If you have submodels in your previously created model, you can still open your model, but you must convert the submodels into top-level SimBiology models to be able to perform any tasks.

To convert submodels, use `sbiupdate`.

If you open a model that contains submodels in the SimBiologydesktop, the submodels are automatically converted to models and placed in separate model sessions. See `sbiupdate` for more information.

As a result of removing submodels, the `addmodel` method and `Models` property of the model object are no longer relevant. You can still access the `addmodel` method and the `Models` property for this version of the software, though it may be removed in a future release.

Events

You can now add events to SimBiology models. Use events to describe sudden changes in a model system. Events are supported only by the `sundials` solver and the stochastic solver `ssa`.

For more information on events, refer to the following:

- “Events” — Introduction to events in the SimBiology Getting Started Guide.
- “Changing Model Component Values Using Events” — Discussion about events including how events are evaluated in the SimBiology User’s Guide.
- `Event` object — Reference for event objects.
- “Sundials Solvers” — Deterministic solver for simulating models with events in the SimBiology User’s Guide.

Models containing events do not support sensitivity analysis.

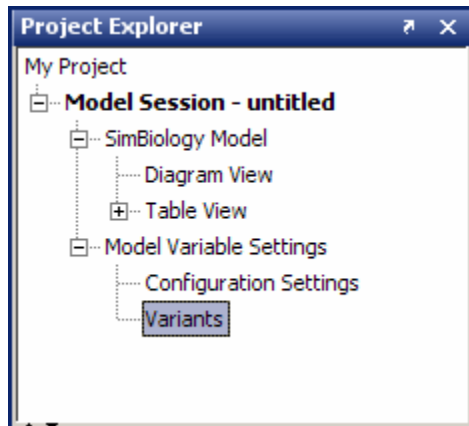
Variants

Variants let you store the names and values of model components and use the values stored in a variant as the alternate value to apply during a simulation. You can store values for species `InitialAmount`, parameter `Value`, and compartment `Capacity` in a variant.

For more information on variants, refer to the following:

- “Variants” — Introduction to variants in the SimBiology Getting Started Guide.
- `Variant` object — Reference for variant objects in the SimBiology Reference.
- “Desktop Example — Applying Changes to Parameter Value Using a Variant” — Example in a tutorial in the SimBiology Model Reference.

In the SimBiology desktop, expand **Model Variable Settings** and double-click **Variants** to open the **Variants** pane. The **SimBiology Desktop Help** updates with more information on adding and setting variants. If the help is not open in the desktop, select **Help > SimBiology Desktop Help**.



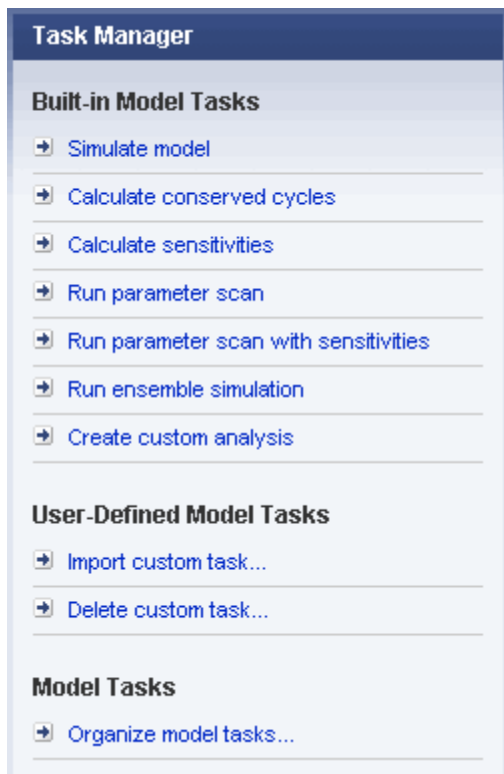
Support for Analysis Tasks in the Desktop

The SimBiologydesktop now supports adding and managing analysis tasks through the following features:

- “Task Manager” on page 28
- “Sensitivity Analysis in the Desktop” on page 29
- “Scanning and Scanning with Sensitivities in the Desktop” on page 30
- “Ensemble Simulation Runs in the Desktop” on page 30
- “Conserved Cycle Calculations in the Desktop” on page 30
- “Create Custom Analysis Tasks” on page 31
- “Generate Reports for Projects” on page 31

Task Manager

The **Task Manager** lets you add and manage simulation and analysis tasks. In the **Project Explorer**, double-click **Model Session**. The **Model Session** pane opens with the **Task Manager** listed on the right.



Click a task to add it to your model. The desktop adds the task to the **Project Explorer** and opens the task pane. For more information on setting up and running a task in the desktop, open the task pane and select **Help > SimBiology Desktop Help** to see the context-sensitive help.

Sensitivity Analysis in the Desktop

Sensitivity analysis is now supported in the desktop. Sensitivity analysis was previously available only through command line. Sensitivity analysis lets you calculate the time-dependent sensitivities of a species specified in **SpeciesOutputs** with respect to species initial conditions and parameter values.

See “Performing Sensitivity Analysis Using the Desktop” in the SimBiology User’s Guide for more information. You must have a model in the desktop for this feature to be enabled.

For more information on sensitivity analysis, see “Sensitivity Analysis” in the SimBiology User’s Guide.

Scanning and Scanning with Sensitivities in the Desktop

You can perform species and parameter scanning analysis alone or in combination with sensitivity analysis in the desktop. Scan a parameter value or a species initial amount to determine the effect of a range of values of the parameter or species.

Combine the scan with sensitivity analysis to explore the sensitivity of a species with respect to a range of values of a parameter or a species.

For information on how to add the task to a model, see “Task Manager” on page 28. For more information on setting up and running the task in the desktop, open the task pane and select **Help > SimBiology Desktop Help** to see the context-sensitive help. You need a model in the desktop for this feature to be enabled.

Ensemble Simulation Runs in the Desktop

The SimBiologydesktop now supports ensemble simulations. You can perform ensemble simulations using the stochastic solvers to gather data from multiple stochastic runs of the model.

See “Running Ensemble Simulations in the Desktop” in the SimBiology User’s Guide for more information. You need a model in the desktop for this feature to be enabled.

Conserved Cycle Calculations in the Desktop

The SimBiologydesktop now supports conserved cycle calculations. This feature lets you calculate a complete set of linear conservation relations for the species in a SimBiology model object. For an introduction, see “Moiety Conservation” in the SimBiology User’s Guide.

For information on how to add the task to a model, see “Task Manager” on page 28. For more information on setting up and running the task in the desktop, open the task pane and select **Help > SimBiology Desktop Help** to see the context-sensitive help. You need a model in the desktop for this feature to be enabled.

Create Custom Analysis Tasks

The SimBiologydesktop lets you create custom tasks that are associated with a project. You can either write new code or copy and modify task code from built-in tasks in the desktop. For more information about creating custom tasks, see “Desktop Example — Creating Custom Analysis” in the SimBiology User’s Guide.

For information on how to add the task to a model, see “Task Manager” on page 28. For more information on setting up and running the task in the desktop, open the task pane and select **Help > SimBiology Desktop Help** to see the context-sensitive help. You need a model in the desktop for this feature to be enabled.

Generate Reports for Projects

You can now generate reports for your projects. Report templates let you generate a report with specified information about a model. To generate a template, in the **Project Explorer**, click **Report Templates** to open the **Report Templates** pane. Select **Help > SimBiology Desktop Help** to see the context-sensitive help for information on how to generate reports.

Changes to the Library Structure in the Root

Built-in and user-defined libraries for units, unit prefixes, and abstract kinetic laws are now organized under two root object properties, `BuiltInLibrary` and `UserDefinedLibrary`, with subcategories for units, unit prefixes, and abstract kinetic laws. See `Root` object, `BuiltInLibrary`, and `UserDefinedLibrary` in SimBiology Reference for more information.

Compatibility Considerations

In previous versions, the libraries were organized under six properties:

- `UserDefinedKineticLaws`

- BuiltInKineticLaws
- UserDefinedUnits
- BuiltInUnits
- UserDefinedUnitPrefixes
- BuiltInUnitPrefixes

The changes to the library structure improve the organization of root object properties.

To illustrate the change using an example, previously you would access a user-defined kinetic law using the following syntax:

```
rootObj = sbioroot;  
get(rootObj, 'UserDefinedKineticLaws')
```

You must now use the following syntax:

```
rootObj = sbioroot;  
get(rootObj.UserDefinedLibrary, 'KineticLaws')
```

New Features for Solvers and Simulation Settings

The following new features and changes apply to solvers and simulation settings:

- “Support for Sundials Solvers” on page 32
- “New Property in Configuration Sets to Specify Species Dimensions” on page 33
- “SimData Object Holds All Simulation Data” on page 33

Support for Sundials Solvers

The Sundials package of solvers has been added in this release. The Sundials solvers are part of a freely available third-party package developed at Lawrence Livermore National Laboratory. Models that contain events are supported by the Sundials solvers and by the stochastic solver `ssa`. For more information, see “Sundials Solvers” in the SimBiology User’s Guide.

New Property in Configuration Sets to Specify Species Dimensions

The new property `DefaultSpeciesDimension` lets you specify whether the default species dimensions should be concentration (default) or substance. This property thus lets you specify whether the solver should account for compartment capacity. If however, you specify the species units in the `InitialAmountUnits` property, these units define the species dimension regardless of the value in `DefaultSpeciesDimension`.

SimData Object Holds All Simulation Data

The SimBiology `SimData` object now stores the data returned from any simulation. For example, the output from the `sbiosimulate` function is now stored in a `SimData` object which holds time and state data as well as metadata, such as the types and names for the logged states or the configuration set used during simulation.

You can also store data from multiple simulation runs as an array of `SimData` objects. Thus, the output of `sbioenssemblerun` is an array of `SimData` objects. See `SimData` object for more information and a list of methods and properties.

Compatibility Considerations. The `SimData` object is now the preferred container for simulation and analysis task data. Previously, simulation and analysis data were stored as time series objects. Functions that used to return time series objects now return `SimData` objects. If you have time series objects in your projects, you can convert them using `sbioupdate`. Functions that used to take a time series object as an input argument now take `SimData` object. You can use time series objects in an input argument, but you see a warning. Support for time series objects in SimBiology functions may be removed in a future version.

The `sbiogetsensmatrix` and `sbiogetnamedstate` functions are being replaced by the `SimData` object methods `getsensmatrix` and `selectbyname` respectively.

New Plot Functions

There are two new plot functions — `sbioplot` and `sbiosubplot`. Both functions let you plot data directly from the `SimData` object (see “`SimData` Object Holds All Simulation Data” on page 33).

`sbioplot` plots each simulation run for a `SimData` object or array of `SimData` objects from a model in the same figure. The plot is a time plot of each state in the object. The figure also shows a hierarchical display of all the runs in a tree, where you can choose which trajectories to show.

`sbiosubplot` plots each simulation run for a `SimData` object or array of objects into its own subplot. The subplot is a time plot of each state in the object. You can navigate through the plots in the figure window and select a subset of the plots to view.

New Sensitivity Analysis Property for Species Outputs

To set up sensitivity analysis, you must now specify an additional property called `SpeciesOutputs`. In `SpeciesOutputs`, specify the species for which you want to compute sensitivities.

Compatibility Considerations

Previously, sensitivity analysis used the species specified in `StatesToLog` as the species for which sensitivities should be calculated. `SpeciesOutputs` improves the functionality by separating the use of the properties.

If you have models from a previous version configured for sensitivity analysis, you must specify species for which you want to compute sensitivities in the `SpeciesOutputs` property. Until this property is specified, sensitivity analysis gives a warning that the `SpeciesOutputs` property is not set, and continues to use the species specified in `StatesToLog`. The use of `StatesToLog` in this context may not be available in a future version, so you should set the `SpeciesOutputs` property for your models, if applicable.

New Way to Add Units and Unit Prefixes

Units and unit prefixes are now represented by objects. You can create units and prefixes and add them to the user-defined library using the function

`sbioaddtolibrary`. See `Unit` object and `UnitPrefix` object for more information.

Compatibility Considerations

Previously, `sbioregisterunit` and `sbiounregisterunit` created and removed units respectively. These functions now produce warnings and will be removed in a future version. Use unit objects and `sbioremovefromlibrary` instead.

Similarly, `sbioregisterunitprefix` and `sbiounregisterunitprefix` created and removed unit prefixes respectively. These functions now produce warnings and will be removed in a future version. Use unit prefix objects and `sbioremovefromlibrary` instead.

Functions and Properties Being Removed

For more information about the process of removing functions, see “About Functions Being Removed” in “What Is in the Release Notes” on page 2.

Function or Property Name	What Happens When You Use Function or Property?	Use This Instead	Compatibility Considerations
<code>addmodel</code>	Warns	<code>addcompartment</code> , where applicable	See “Submodel Support Will Be Removed” on page 26.
<code>Models</code>	Still runs	<code>Compartments</code> , where applicable	See “Submodel Support Will Be Removed” on page 26.

Function or Property Name	What Happens When You Use Function or Property?	Use This Instead	Compatibility Considerations
<i>Model name</i> as an input argument for sbioroot	Warns	sbioselect	sbioroot does not accept a model name as an input argument. Use sbioselect to query models by name.
sbioregisterunit	Warns	Unit object and sbioaddtolibrary	See “New Way to Add Units and Unit Prefixes” on page 34.
sbiounregisterunit	Warns	sbioremovefromlibrary	See “New Way to Add Units and Unit Prefixes” on page 34.
sbioregisterunitprefix	Warns	Unit object and sbioaddtolibrary	See “New Way to Add Units and Unit Prefixes” on page 34.
sbiounregisterunitprefix	Warns	sbioremovefromlibrary	See “New Way to Add Units and Unit Prefixes” on page 34.
sbiogetsensmatrix	Warns	getsensmatrix	See “SimData Object Holds All Simulation Data” on page 33.

Function or Property Name	What Happens When You Use Function or Property?	Use This Instead	Compatibility Considerations
sbiogetnamedstate	Warns	selectbyname	See “SimData Object Holds All Simulation Data” on page 33.
BuiltInKineticLaws	Warns	BuiltInLibrary	See “Changes to the Library Structure in the Root” on page 31.
BuiltInUnits	Warns	BuiltInLibrary	See “Changes to the Library Structure in the Root” on page 31.
BuiltInUnitPrefixes	Warns	BuiltInLibrary	See “Changes to the Library Structure in the Root” on page 31.
UserDefinedKineticLaws	Warns	UserDefinedLibrary	See “Changes to the Library Structure in the Root” on page 31.
UserDefinedUnits	Warns	UserDefinedLibrary	See “Changes to the Library Structure in the Root” on page 31.
UserDefinedUnitPrefixes	Warns	UserDefinedLibrary	See “Changes to the Library Structure in the Root” on page 31.

Version 2.1.2 (R2007b) SimBiology Software

This table summarizes what's new in Version 2.1.2 (R2007b):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
No	No	Bug Reports	No

Version 2.1.1 (R2007a) SimBiology Software

This table summarizes what's new in Version 2.1.1 (R2007a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
No	No	Bug Reports	No

Version 2.1 (R2006b+) SimBiology Software

This table summarizes what's new in Version 2.1 (R2006b+):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as Compatibility Considerations , below. See also Summary.	Bug Reports Includes fixes	No

New features and changes introduced in this version are:

- “Printing and Exporting the Diagram” on page 40
- “Diagram Menu” on page 40
- “Block Overview Tool” on page 41
- “Miscellaneous Desktop Enhancements” on page 41

Printing and Exporting the Diagram

You can annotate and print your pathway's block diagram in SimBiology, or export the diagram using .svg, .jpeg, or .pdf file formats. For example, you can annotate the diagram with the name of the author, the date, notes, and name of the organization. You can choose to place this content as a header or footer on the diagram page.

Diagram Menu

The SimBiology desktop supports multiple diagram features and actions through the **Diagram** menu. Use the menu options to do the following:

- Copy, paste, and delete blocks.
- Select model or diagram components by category (for example, select all lines or all Species blocks).

- Filter selected portions of a model diagram to include only model or diagram components by category.
- Edit multiple blocks or lines.
- Add selected blocks to a Block Library.
- Annotate and print, or export your model diagram.
- Perform layout tasks, for example, hide and show blocks, move blocks to the front or back of a diagram, rotate a block, and pin or unpin selected blocks.
- Copy the style of a block and apply the style to a group of selected blocks.
- Reload a graphic used for a block in the Block Properties dialog box.
- Access diagram tools such as the Diagram Table View, Block Library Browser, Diagram Overview, and Block Overview.

Compatibility Considerations

There is a compatibility consideration regarding the support for editing multiple blocks or lines. In SimBiology Versions 2.0 and 2.0.1, if you selected multiple blocks and/or lines and applied an editing action such as **Hide**, **Pin**, or **Hide Name**, the action applied only to the block on which you selected the right-click (context) menu. Starting in Version 2.1, applicable editing actions are propagated to all selected blocks.

Block Overview Tool

The Block Overview tool provides a summary of key information about a particular block. Hover the mouse over a block to find information in the **Block Overview** pane. Each block includes information pertinent to that type of block, for example, a species block overview shows **Name**, **InitialAmount**, **InitialAmountUnits**, the number and list of reactions the species is involved in, and a description of any indicators shown in the diagram.

Miscellaneous Desktop Enhancements

The enhancements for the SimBiologydesktop let you do the following:

- Set your preference to open a SimBiology pane with a single-click in the **Project Explorer**. Select **File > Preferences** to open the Preferences dialog box.
- Interrupt and stop model verification by clicking **Stop** when **Verify** is running.
- Sort the available plot arguments in the **Simulation** and **Data** panes. The X and Y argument lists are now tabulated.
- View a species and any of its cloned blocks from the search results for the species.
- Select and view multiple results for **Find** and **Bookmarks** in either the diagram or the table form.

Version 2.0.1 (R2006b) SimBiology Software

This table summarizes what's new in Version 2.0.1 (R2006b):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
No	No	Bug Reports Includes fixes	No

Version 2.0 (R2006a+) SimBiology Software

This table summarizes what's new in Version 2.0 (R2006a+):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as Compatibility Considerations , below. See also Summary.	Bug Reports Includes fixes	No

New features and changes introduced in this version are:

- “Diagram Interface” on page 44
- “Find and Bookmarks in Projects” on page 45
- “Sensitivity Analysis” on page 45
- “Parameter Estimation” on page 46
- “Ensemble Simulation Runs” on page 46
- “Moiety Conservation” on page 47
- “Model Verification and Validation” on page 47
- “Simulation and Solvers” on page 47
- “New Demos for SimBiology Version 2.0” on page 48

Diagram Interface

The Diagram is a graphical user interface you can use to enter model pathways using block representations for species, reactions, and submodels. Use the Plot block to visualize simulation data during a simulation. For a tutorial, see “Modeling Using the SimBiology Diagram” in the SimBiology Getting Started Guide. You can also access video demos from a list of SimBiology demos.

Find and Bookmarks in Projects

- **Find** — Type a string in the **Find** box to quickly find matching model components in the project and in abstract kinetic laws.
- **Bookmarks** — Use complex rules to identify objects from the project, and create a custom and persistent set of desktop and project objects.

Compatibility Considerations

In Versions 1.0 and 1.0.1, SimBiology projects saved searches. In Version 2.0, the project saves bookmarks. The software converts a previous version's saved search into a bookmark. After you save a project in Version 2.0, a bookmark is saved and the old search is no longer available. If a project contains a search, you see a warning that the project file will contain a bookmark rather than a search after saving.

All functionality available in Version 1.0 searches are present in Version 2.0 bookmarks except for the ability to mix and match and and or between the search rows. If you have a multiple-row search saved with a mixture of and and or, when you load this project into Version 2.0, it is converted to either all or any based on whether the software first encounters and or or.

Sensitivity Analysis

You can perform sensitivity analysis using the following properties:

- **SensitivityAnalysis** — Configuration set property that lets you calculate the time-dependent sensitivities of all the species states defined by the **StatesToLog** property with respect to species initial conditions and parameter values.
- **SensitivityAnalysisOptions** — An object that holds the sensitivity analysis options in the configuration set object. Properties of **SensitivityAnalysisOptions** are summarized below:
 - **SpeciesInputFactors** — Specify the species with respect to which you want to compute the sensitivities of the species states in your model.
 - **ParameterInputFactors** — Specify the parameters with respect to which you want to compute the sensitivities of the species states in your model.

- **Normalization** — Specify the normalization for the calculated sensitivities.

For an introduction and an example, see “Sensitivity Analysis” in the SimBiology User’s Guide.

Parameter Estimation

The `sbioparamestim` function lets you estimate any or all parameters in your model using the experimental data you provide. The software uses the optimization functions in the MATLAB, Optimization Toolbox™, and Genetic Algorithm and Direct Search Toolbox™ software to enable parameter estimation.

Optimization Toolbox and Genetic Algorithm and Direct Search Toolbox software are not required for you to use `sbioparamestim`. If you do not have these products installed, `sbioparamestim` uses the MATLAB function `fminsearch` by default.

For an introduction and an example, see “Parameter Estimation” in the SimBiology User’s Guide.

Ensemble Simulation Runs

You can perform ensemble simulations using the stochastic solvers to gather data from multiple stochastic runs of the model. The following functions let you perform ensemble runs:

- `sbioensamplerun` — Performs multiple stochastic ensemble runs of the SimBiology model object.
- `sbioensembleplot` — Shows a 2-D distribution plot or a 3-D shaded plot of the time varying distribution of one or more specified species in the ensemble data generated by `sbioensamplerun`.
- `sbioensemblestats` — Gets mean and variance as a function of time for all the species in the ensemble data generated by `sbioensamplerun`.

Moiety Conservation

The `sbioconsmoiety` function lets you calculate a complete set of linear conservation relations for the species in a SimBiology model object.

For an introduction and an example, see “Moiety Conservation” in the SimBiology User’s Guide.

Model Verification and Validation

SimBiology software performs model verification and validation either during simulation, or when you explicitly execute the commands for verification before simulation.

Verification at the Command Line

The following new functions let you verify and validate, at the command line, that your model is ready for simulation:

- `verify` — Performs checks on a model to verify that you can simulate the model. You see stacked errors and warnings if any problems are found. To see the entire list of errors and warnings, use `sbiolasterror` and `sbiolastwarning`.
- `sbiolasterror` — Returns a SimBiology diagnostic structure array containing the last errors that are generated.
- `sbiolastwarning` — Returns a SimBiology diagnostic structure array containing the last warnings that are generated.

Verification on the SimBiology Desktop

Click the **Verify** button on the SimBiology desktop toolbar to perform verification and validation of your model. The **Output** pane opens to show the errors and warnings. You can double-click a result row to go to the location of the error or warning.

Simulation and Solvers

The following new features and changes apply to simulation settings and solvers:

- **MaxStep** — Lets you specify the upper bound on solver step size for a deterministic solver. `MaxStep` is a property of the `SolverOptions` object.
- **Implicit Tau solver settings** — For `impltau`, `AbsoluteTolerance` holds the value for convergence tolerance for the nonlinear solver that is used internally by the Implicit Tau solver. You can now specify `AbsoluteTolerance` for `impltau`. Previously, if you selected the Implicit Tau solver, the software ignored any changes to the `AbsoluteTolerance` and `RelativeTolerance` options within a configuration set and used the default values set internally.
- **UnitConversion** — Supported by both deterministic solvers and stochastic solvers. Previously `UnitConversion` was supported only by the stochastic solvers.

Implicit Tau Solver Settings Compatibility Considerations

The `RelativeTolerance` property is no longer valid for the Implicit Tau (`impltau`) solver.

When you load a file created in a previous version, the project loads the `RelativeTolerance` property. But when you save the file, the software updates the change.

Unit Conversion Compatibility Considerations

The `UnitConversion` property default is now `'false'`. If you load a SimBiology project created in a previous version into the SimBiology desktop, the `UnitConversion` setting in each model in the project remains as the saved setting. If however, you are running an M-file, you must now remember to set the `UnitConversion` property to `true` if you want the software to perform unit conversions

New Demos for SimBiology Version 2.0

There are 14 new demos for SimBiology Version 2.0. Click **SimBiology demos** or type `demo('MATLAB', 'SimBiology')` at the command prompt.

Version 1.0.1 (R2006a) SimBiology Software

This table summarizes what's new in Version 1.0.1 (R2006a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	No	Bug Reports	No

The changes introduced in this version are:

- The characters \, ^, and * in species and parameter names are now supported in rules.
- Rules are now supported in submodels.

Version 1.0 (R14SP3+) SimBiology Software

This table summarizes what's new in Version 1.0 (R14SP3+):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	No	Bug Reports	No

The features introduced in this version are:

- “Introduction” on page 50
- “Features” on page 51
- “Known Software Problems” on page 51
- “Upgrading from a Beta Release” on page 55

Introduction

SimBiology Version 1.0 (Release 14SP3+) is an integrated environment for modeling biological processes, simulating the dynamic behavior of these processes, and analyzing simulation and experimental data. Biological processes include metabolic, genetic, and signaling pathways with transform, binding, and transport reactions. You can use SimBiology software as a tool in three major areas:

- **Model** — Design and build models by entering model components with a graphical desktop interface, or use the MATLAB Command Window.
- **Simulate** — Select deterministic or stochastic solvers and observe the changes in species amounts and variable parameter values over time.
- **Analyze** — Log data from a simulation and export the data to the MATLAB workspace. Compare simulation with experimental data for parameter estimation and model validation.

Features

The features in SimBiology Version 1.0 are the following:

- **Graphical user interface** — Provides access to the command-line functionality through a graphical user interface (GUI).
- **Command-line interface** — All the features are accessible and executable from the MATLAB Command Window.
- **Data formats and projects** — Organize and save related models, simulation data, and analysis results in project files. Save user-defined kinetic laws and units. Share models by exporting SBML Level 2 files.
- **Modeling** — Create biological models by adding components that include reactions, species, parameters, kinetic laws, rules, and submodels.
- **Simulation** — Select either deterministic or stochastic solvers with dimensional analysis and unit conversion.
- **Analysis** — SimBiology software is fully integrated with MATLAB. Record data during a simulation and analyze results in MATLAB.

Known Software Problems

To view important open bugs for SimBiology Version 1.0, use the Bug Reports interface on the MathWorks Web site.

Note If you are not already logged in to your MathWorks Account, when you link to the Bug Reports interface (see below), you will be prompted to log in or create an account.

After you are logged in, use this [Bug Reports](#) link. You will see the bug report for SimBiology. The report is sorted with fixed bugs listed first, and then open bugs. You can select the Status column to list the open bugs first.

If you are viewing these release notes in PDF form on the MathWorks Web site, you can refer to the HTML form of the release notes on the MathWorks Web site and use the link provided.

Software problems include unsupported SBML features and current feature and function limitations.

Unsupported SBML Level 2 Version 1 Features

SimBiology software supports a subset of the SBML Level 2 Version 1 specification. Unsupported features include:

- Compartments — Model compartments are not supported. If an imported SBML model has a single compartment, the model is loaded as a top-level model. If the model has multiple compartments, you see a warning and the software does not load the SBML file.
- Volume — Volume is not supported and cannot be specified.
- Events — Events in an SBML file are ignored when you are importing into a project.
- Piecewise kinetics — Models with piecewise kinetics are loaded in, but the software ignores the definitions for piecewise kinetics.
- Function definitions — Models containing functional definitions are loaded, but you see a warning and the software ignores the function definitions.
- MATLAB incompatible variable names in `UnitDefinition` — Models that have variable names incompatible with MATLAB in `UnitDefinition` are not loaded and you see an error message.

Functional Limitations

Simulation and Solvers

- Stochastic solvers support only mass action kinetics, while ODE solvers support all built-in and user-defined kinetic laws.
- If you use a stochastic solver to simulate a model, the software ignores any rate, assignment, or algebraic rules if present in the model.
- If you have `DimensionalAnalysis` and `UnitConversion` on for the stochastic solvers, note that the software assumes that volume is unity during simulation. The stochastic solvers perform calculations using species units in molecules. Therefore, if you specify the species units in molecules per unit volume or moles per unit volume, the software assumes volume to be unity and converts species amounts to molecules for simulation. The results are finally plotted in the units you specified for the species.

In addition, if you have reactions with stoichiometric coefficients greater than or equal to 2, you need to convert the deterministic rate constants to stochastic rate constants. For example, $2 R \rightarrow P$ has a reaction rate of $v=k[R]^2$. If R is moles/liter, the deterministic rate constant k has units of liter/mole-second. If the unit of species concentration is molecule, then the stochastic rate constant c has units of 1/molecule-second, and $c = 2k/NV$ where N is Avogadro's Number, $6.022e23$ molecules/mole, and V is the volume of the model in liters.

- When you select the Implicit Tau solver, the software ignores any changes to `AbsoluteTolerance` and `RelativeTolerance` options within a configuration set and uses the default values that are set internally.
- By default, `StatesToLog` is set to 'all' and all variable parameters are logged. Variable parameters are those that have `ConstantValue` cleared or false. If you choose the species to log, however, you cannot log the variable parameters.
- The characters `\`, `^`, and `*` in species and parameter names are not supported in rules.

Units

- Stochastic solvers support dimensional analysis and unit conversion. ODE solvers support dimensional analysis but not unit conversion.
- You can delete a unit that is being used in a model; however, you will see an error when you try to simulate the model or export to SBML.

Submodels

- The context menus (right-click options) for the **Species**, **Reaction**, **Parameter** and **Rule** nodes that appear beneath a submodel node all act on the corresponding parent model node, and not on the submodel node. For example, if you select **Delete All Species** in the submodel **Species** node, this selection deletes the species in the parent model.
- Rules are not supported in submodels.

SBML Export Limitations

There are features in SimBiology software that are not supported in SBML. When you export a model to an SBML file, you might lose some of these features.

- Submodels are not supported by SBML export.
- The abstract kinetic law name and corresponding expression are not supported by SBML, but the associated reaction rate equation is exported to SBML.
- The properties `Tag`, `UserData`, and `Active` are not supported by SBML export.

Tips

Naming SimBiology Variables

- If you are using a species or parameter name that is not a valid MATLAB variable name, do the following:
 - Enclose the name in square brackets when writing a reaction rate equation or a rule.
 - Enter the name without brackets when you are creating the species or parameter, or when you add a reaction.

For example, enclose `[DNA polymerase+]` and `[K_DNA polymerase+]` within brackets in reaction rates and rules, but, enter `DNA polymerase+` or `K_DNA polymerase+` when creating a species, adding a reaction, or creating a parameter.

- The names `i` and `j` are reserved MATLAB characters. Because expressions in abstract kinetic laws, reaction rates, and rules are considered to be MATLAB code, the software evaluates `i` or `j` as an imaginary number and not as the value of species `i` or `j`. For example, an expression $V \cdot S^i / K$ is interpreted to have three names, `V`, `S`, and `K`, instead of four. Use brackets to protect such variables.

If a variable in a reaction rate equation or rule has the same name as a MATLAB function, the software evaluates the expression as a call to the MATLAB function. In general, when creating variable names, you should

avoid using MATLAB function names or variable names that are invalid in MATLAB.

Changing SimBiology Variable Names

- If you change the Name of a parameter you must configure all applicable elements, such as rules that use the parameter, any user-specified ReactionRate, or the kinetic law object property ParameterVariableNames. Use the method setparameter to configure ParameterVariableNames.

To update parameter names in the SimBiology graphical user interface, access each appropriate pane through the **Project Explorer**.

- If you change the Name of a species you must configure all applicable elements, such as rules that use the parameter, any user-specified ReactionRate, or the kinetic law object property SpeciesVariableNames. Use the method setspecies to configure SpeciesVariableNames.

To update species names in the SimBiology graphical user interface, access each appropriate pane through the **Project Explorer**. The software automatically updates species names for reactions that use MassAction kinetic law.

Upgrading from a Beta Release

Any projects that you created and saved with the SimBiology beta release version will not load in Version 1.0.

As a workaround, before upgrading to Version 1.0, save your models to SBML, upgrade to Version 1.0, and then import the SBML models into Version 1.0 projects. Alternatively, contact the MathWorks for help with your conversion.

Compatibility Summary for SimBiology Software

This table summarizes new features and changes that might cause incompatibilities when you upgrade from an earlier version, or when you use files on multiple versions. Details are provided in the description of the new feature or change.

Version (Release)	New Features and Changes with Version Compatibility Impact
Latest Version V3.1 (R2009b)	See the following: <ul style="list-style-type: none"> • “Unit Conversion Compatibility Considerations” on page 6 • “Functions and Properties Being Removed” on page 6
V3.0 (R2009a)	None
V2.4 (R2008b)	See the Compatibility Considerations subheadings for: <ul style="list-style-type: none"> • “Support for Specifying Additional Inputs in Custom Plot Types” on page 13 • “Additional Options for Renaming Compartments, Species, and Parameters” on page 17 • “Change in the Random Number Generator Used During Stochastic Simulations” on page 18 See “Functions and Properties Being Removed” on page 19.
V2.3 (R2008a)	See “Functions and Properties Being Removed” on page 22.

Version (Release)	New Features and Changes with Version Compatibility Impact
V2.2 (R2007b+)	<p>See the Compatibility Considerations subheadings for:</p> <ul style="list-style-type: none"> • “Submodel Support Will Be Removed” on page 26 • “Changes to the Library Structure in the Root” on page 31 • “SimData Object Holds All Simulation Data” on page 33 • “New Sensitivity Analysis Property for Species Outputs” on page 34 • “New Way to Add Units and Unit Prefixes” on page 34 <p>See “Functions and Properties Being Removed” on page 35.</p>
V2.1.2 (R2007b)	None
V2.1.1 (R2007a)	None
V2.1 (R2006b+)	See the Compatibility Considerations subheading for “Diagram Menu” on page 40.
V2.0.1 (R2006b)	None
V2.0 (R2006a+)	<p>See the Compatibility Considerations subheading for “Find and Bookmarks in Projects” on page 45 and the following:</p> <ul style="list-style-type: none"> • “Implicit Tau Solver Settings Compatibility Considerations” on page 48 • “Unit Conversion Compatibility Considerations” on page 48
V1.0.1 (R2006a)	None
V1.0 (R14SP3+)	None