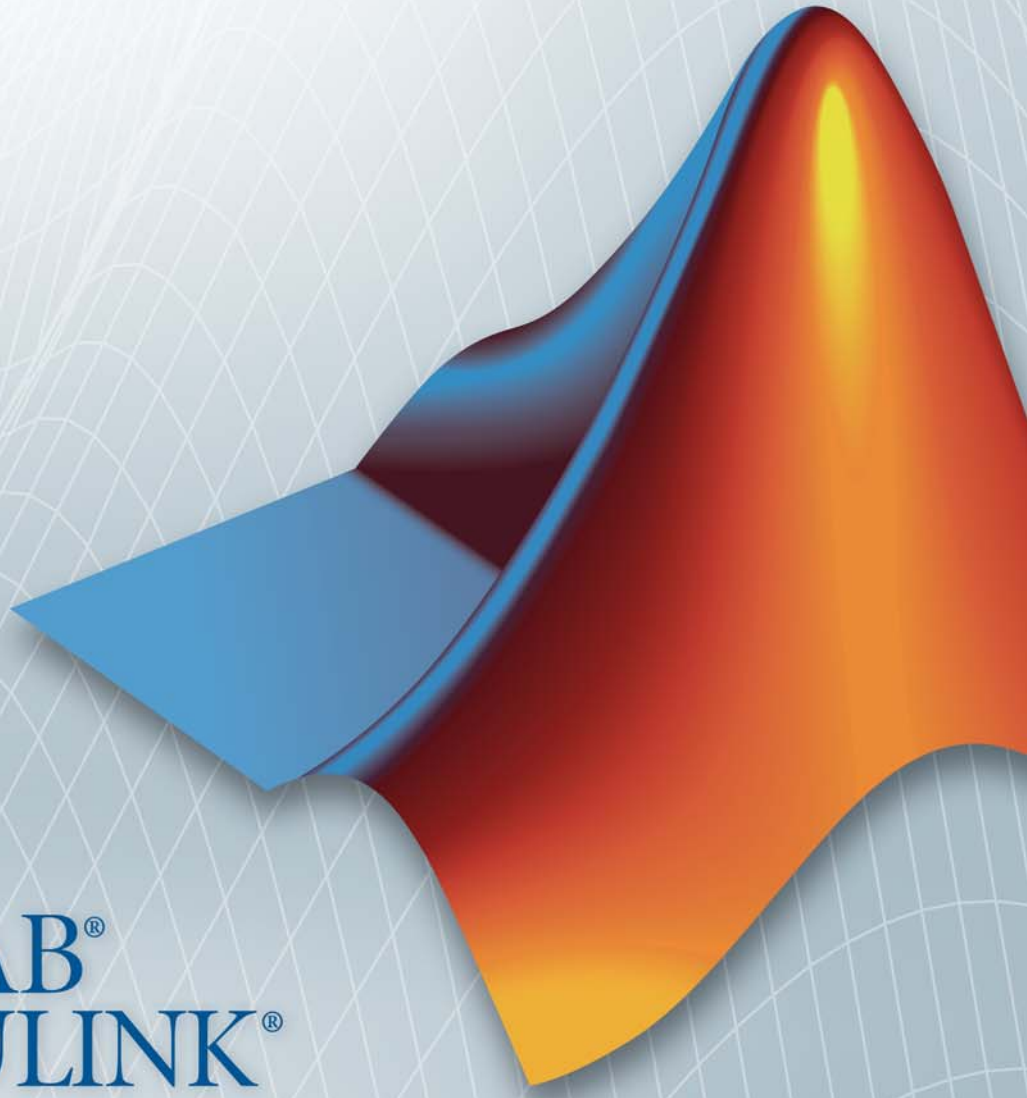
# SimMechanics™

Reference

**R2012b**



MATLAB®
&SIMULINK®

MathWorks®

### How to Contact MathWorks

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

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

508-647-7000 (Phone)

508-647-7001 (Fax)

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

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

*SimMechanics™ Reference*



#### Trademarks

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

#### Patents

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

#### Revision History

| | | |
|---|---|---|
| September 2012 | Online only | New for Version 4.1 (Release 2012b) |

# Contents

## 1 Block Reference


**Body Elements** . . . . . 1-2

**Constraints** . . . . . 1-3

**Forces and Torques** . . . . . 1-4

**Frames and Transforms** . . . . . 1-5

**Joints** . . . . . 1-6

**Utilities** . . . . . 1-7


## 2 Configuration Parameters


**SimMechanics Pane: General** . . . . . 2-2
SimMechanics Pane Overview . . . . . 2-2

**SimMechanics Pane: Diagnostics** . . . . . 2-3
Invalid visual properties . . . . . 2-4
Repeated vertices in a cross-section . . . . . 2-5
Unconnected frame port . . . . . 2-6
Redundant block . . . . . 2-7
Conflicting reference frames . . . . . 2-8
Rigidly constrained block . . . . . 2-9
Unsatisfied high priority state targets . . . . . 2-10
Overspecified targets in kinematic loops . . . . . 2-11

**SimMechanics Pane: Explorer** . . . . . 2-12

Open Mechanics Explorer on model update or simulation ..... 2-12

# 3 SimMechanics Visualization

**Visualization with Mechanics Explorer** ..... 3-2
Introduction to Mechanics Explorer ..... 3-2
Visualization Requirements ..... 3-4
Configure Mechanics Explorer for Automatic Start-Up ..... 3-4
Open Mechanics Explorer ..... 3-5

**Menu and Tool Bars** ..... 3-7
Menu Bar ..... 3-7
Toolbar ..... 3-8

**Tree View and Property Panes** ..... 3-11
Introduction to Tree View and Property Panes ..... 3-11
Browse Model ..... 3-12

**Context Menu Display Controls** ..... 3-14

**Animation Toolstrip** ..... 3-15
Requirements for Animation Recording and Playback ..... 3-15
How Animation and Playback Work ..... 3-15
Play Control Buttons ..... 3-16
Slider and Loop Controls — Simulation Progress Bar ..... 3-16
Playback Speed Control ..... 3-17
Playback Time Counter ..... 3-18

Functions—Alphabetical List
4

Index

# Block Reference

This page refers to SimMechanics™ Second-Generation. For SimMechanics First-Generation, click here.


| | |
|---|---|
| Body Elements (p. 1-2) | Represent rigid bodies with solid shapes, masses and inertias, geometries, and graphic properties |
| Constraints (p. 1-3) | Represent restrictions on how rigid bodies can move |
| Forces and Torques (p. 1-4) | Represent application of forces and torques to rigid bodies |
| Frames and Transforms (p. 1-5) | Represent frames and transforms |
| Joints (p. 1-6) | Represent ways that rigid bodies can move |
| Utilities (p. 1-7) | Configure mechanical and simulation environment |

# Body Elements

# Constraints

# Forces and Torques

# Frames and Transforms

# Joints

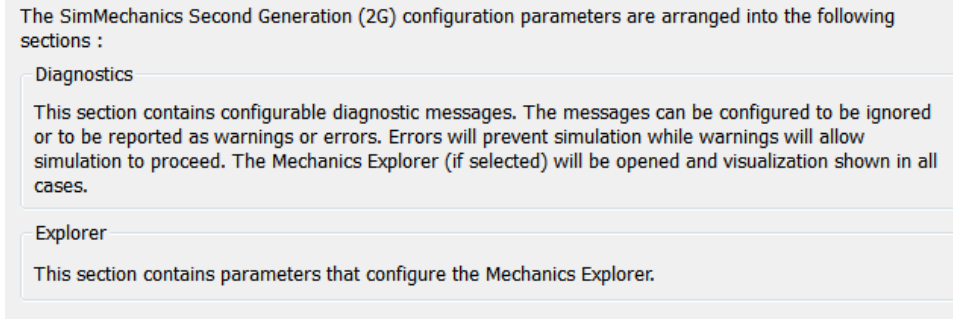
# Utilities

2

# Configuration Parameters


- "SimMechanics Pane: General" on page 2-2
- "SimMechanics Pane: Diagnostics" on page 2-3
- "SimMechanics Pane: Explorer" on page 2-12

# SimMechanics Pane: General



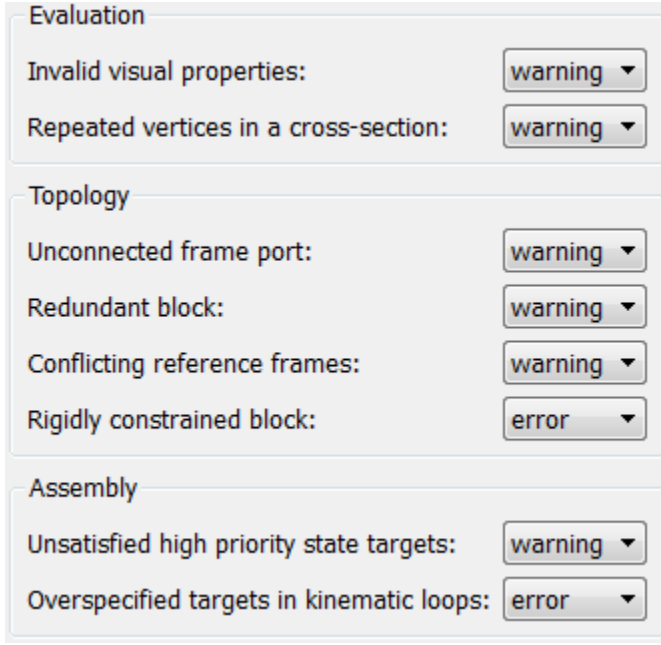

## SimMechanics Pane Overview

Configure the mechanical settings for an entire SimMechanics model.

### Configuration

- This pane appears only if your model contains at least one block from the Simscape™ product or a product based on the Simscape product, such as the SimMechanics product.
- The settings in this pane are saved only if your model contains at least one SimMechanics block.

# SimMechanics Pane: Diagnostics




| In this section... |
| --- |
| “Invalid visual properties” on page 2-4 |
| “Repeated vertices in a cross-section” on page 2-5 |
| “Unconnected frame port” on page 2-6 |
| “Redundant block” on page 2-7 |
| “Conflicting reference frames” on page 2-8 |
| “Rigidly constrained block” on page 2-9 |
| “Unsatisfied high priority state targets” on page 2-10 |
| “Overspecified targets in kinematic loops” on page 2-11 |

## Invalid visual properties

Select the diagnostic action to take if the application detects an improperly specified color vector.

### Settings

**Default:** `warning`

`none`
: The application does not check for this situation.

`warning`
: When the application detects this situation, it displays a warning.

`error`
: When the application detects this situation, it terminates the simulation and displays an error message.

### Command-Line Information

**Parameter:** `SimMechanicsInvalidVisualProperty`
**Type:** string
**Value:** `none | warning | error`
**Default:** `warning`

## Repeated vertices in a cross-section

Select the diagnostic action to take if the application detects repeated vertices in a cross-section.

### Settings

**Default:** `warning`

`none`
: The application does not check for this situation.

`warning`
: When the application detects this situation, it displays a warning.

`error`
: When the application detects this situation, it terminates the simulation and displays an error message.

### Command-Line Information

**Parameter:** `SimMechanicsCrossSectionNullEdge`
**Type:** string
**Value:** `none | warning | error`
**Default:** `warning`

## Unconnected frame port

Select the diagnostic action to take if the application detects an unconnected frame port.

### Settings

**Default:** `Warning`

`none`
: The application does not check for this situation.

`warning`
: When the application detects this situation, it displays a warning.

`error`
: When the application detects this situation, it terminates the simulation and displays an error message.

### Command-Line Information

**Parameter:** `SimMechanicsUnconnectedFramePorts`
**Type:** string
**Value:** `none | warning | error`
**Default:** `warning`

## Redundant block

Select the diagnostic action to take if the application detects a redundant block in the model.

### Settings

**Default:** `warning`

`none`
: The application does not check for this situation.

`warning`
: When the application detects this situation, it displays a warning.

`error`
: When the application detects this situation, it terminates the simulation and displays an error message.

### Command-Line Information

**Parameter:** `SimMechanicsRedundantBlock`
**Type:** string
**Value:** `none | warning | error`
**Default:** `warning`

## Conflicting reference frames

Select the diagnostic action to take if the application detects conflicting reference frames in the model.

### Settings

**Default:** `warning`

`none`
: The application does not check for this situation.

`warning`
: When the application detects this situation, it displays a warning.

`error`
: When the application detects this situation, it terminates the simulation and displays an error message.

### Command-Line Information

**Parameter:** `SimMechanicsConflictingReferenceFrames`
**Type:** string
**Value:** `none | warning | error`
**Default:** `warning`

## Rigidly constrained block

Select the diagnostic action to take if the application detects a rigidly constrained block in the model.

### Settings

**Default:** `warning`

`none`
: The application does not check for this situation.

`warning`
: When the application detects this situation, it displays a warning.

`error`
: When the application detects this situation, it terminates the simulation and displays an error message.

### Command-Line Information

**Parameter:** `SimMechanicsRigidlyBoundBlock`
**Type:** string
**Value:** `none | warning | error`
**Default:** `error`

## Unsatisfied high priority state targets

Select the diagnostic action to take if the application detects targets with unsatisfied desired states in the model.

### Settings

**Default:** `warning`

`none`
: The application does not check for this situation.

`warning`
: When the application detects this situation, it displays a warning.

`error`
: When the application detects this situation, it terminates the simulation and displays an error message.

### Command-Line Information

**Parameter:** `SimMechanicsUnsatisfiedHighPriorityTargets`
**Type:** string
**Value:** `none | warning | error`
**Default:** `warning`

## Overspecified targets in kinematic loops

Select the diagnostic action to take if the application detects overspecified targets contained in kinematic loops in the model.

### Settings

**Default:** `warning`

`none`
: The application does not check for this situation.

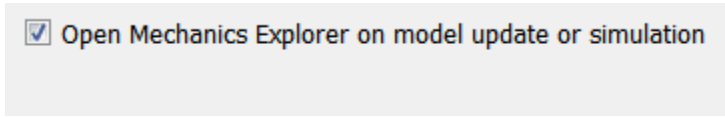
`warning`
: When the application detects this situation, it displays a warning.

`error`
: When the application detects this situation, it terminates the simulation and displays an error message.

### Command-Line Information

**Parameter:** `SimMechanicsJointTargetOverSpecification`
**Type:** string
**Value:** `none | warning | error`
**Default:** `error`

# SimMechanics Pane: Explorer




## Open Mechanics Explorer on model update or simulation

Start Mechanics Explorer when model is updated or simulated.

### Settings

**Default:** on

 On

Model Explorer starts when model is updated or simulated.

 Off

Model Explorer does not start when model is updated or simulated.

### Tip

If you clear this check box, you can start Model Explorer by selecting **Desktop > Mechanics Explorers** from the MATLAB® Command Window.

### Command-Line Information

**Parameter:** `SimMechanicsOpenEditorOnUpdate`
**Type:** string
**Value:** `'on' | 'off'`
**Default:** `'on'`

3

# SimMechanics Visualization


- "Visualization with Mechanics Explorer" on page 3-2
- "Menu and Tool Bars" on page 3-7
- "Tree View and Property Panes" on page 3-11
- "Context Menu Display Controls" on page 3-14
- "Animation Toolstrip" on page 3-15

# Visualization with Mechanics Explorer

**In this section...**

"Introduction to Mechanics Explorer" on page 3-2

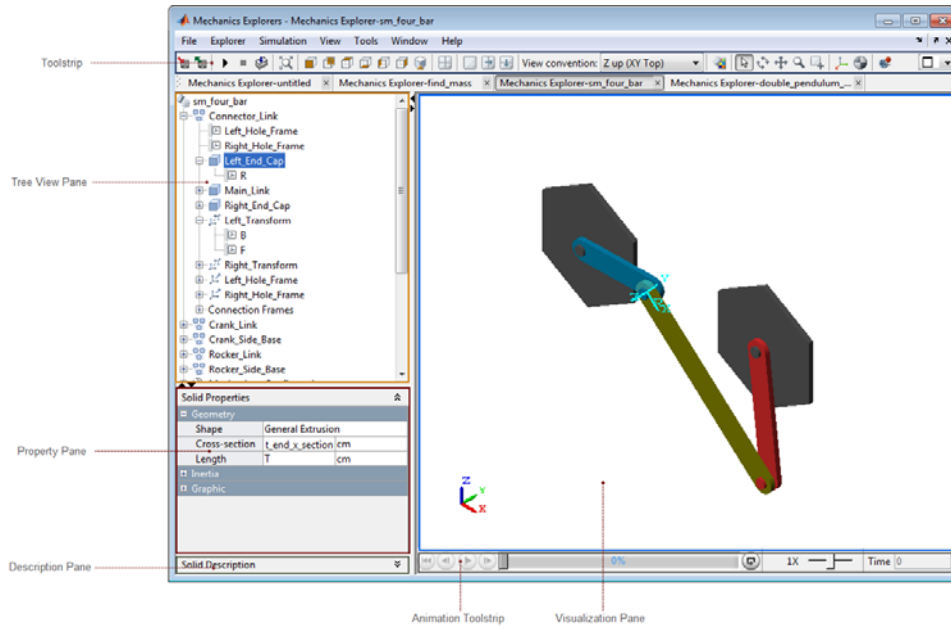
"Visualization Requirements" on page 3-4

"Configure Mechanics Explorer for Automatic Start-Up" on page 3-4

"Open Mechanics Explorer" on page 3-5

## Introduction to Mechanics Explorer

Mechanics Explorer is a tool used to visualize and animate mechanical models in SimMechanics. This tool provides you the entire visualization functionality available through SimMechanics. With it, you can visualize your model in its initial state and simulate it as a function of time.




Mechanics Explorer is the primary visualization tool of SimMechanics. This tool contains three panes:

- Visualization — Displays a 3-D representation of a SimMechanics model.
- Tree View — Displays the model component hierarchy, organized in this order: subsystems→blocks→frame ports.
- Property — Displays the properties of a block or frame port selected in the Tree View pane.
- Description — Provides a short description of the subsystem, block or frame port selected in the Tree View pane.
- Animation — Provides controls for dynamic animations.
- Toolstrip — Provides controls that change the visualization of a model

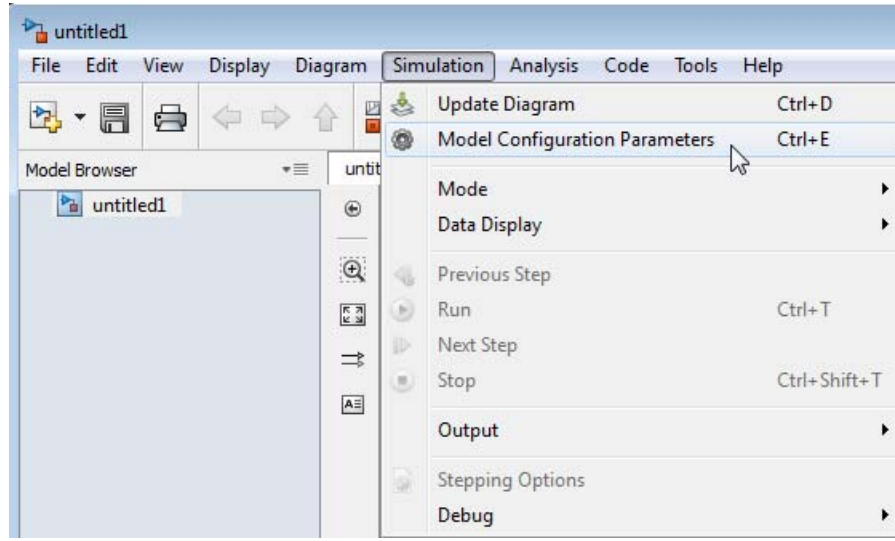
## Visualization Requirements

To visualize a model in SimMechanics, you must have Silicon Graphics OpenGL® installed in your system.

## Configure Mechanics Explorer for Automatic Start-Up

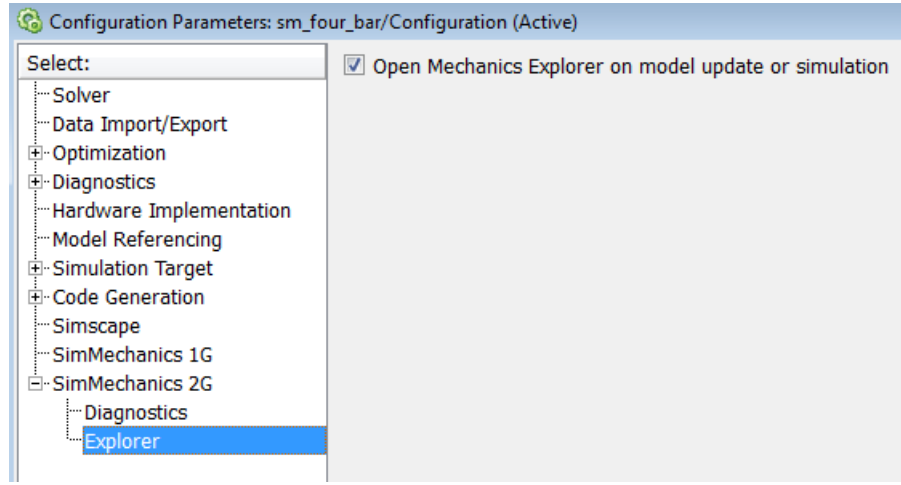
By default, when you update or simulate a model, SimMechanics opens a new Mechanics Explorer window. It is in this window that SimMechanics displays your model. If the configuration settings have changed, you may need to reconfigure Mechanics Explorer for automatic start-up. To do this:

1 In the Simulink Editor window containing your model, select **Simulation > Configuration Parameters**.




2 In the Configuration Parameters dialog box, select **SimMechanics > Explorer**.

3 Check the **Open Mechanics Explorer on model update or simulation** box.




4 Mechanics Explorer is now configured for automatic start-up. Updating or simulating a mechanical model from the Simulink Editor window causes a Mechanics Explorer window to open.

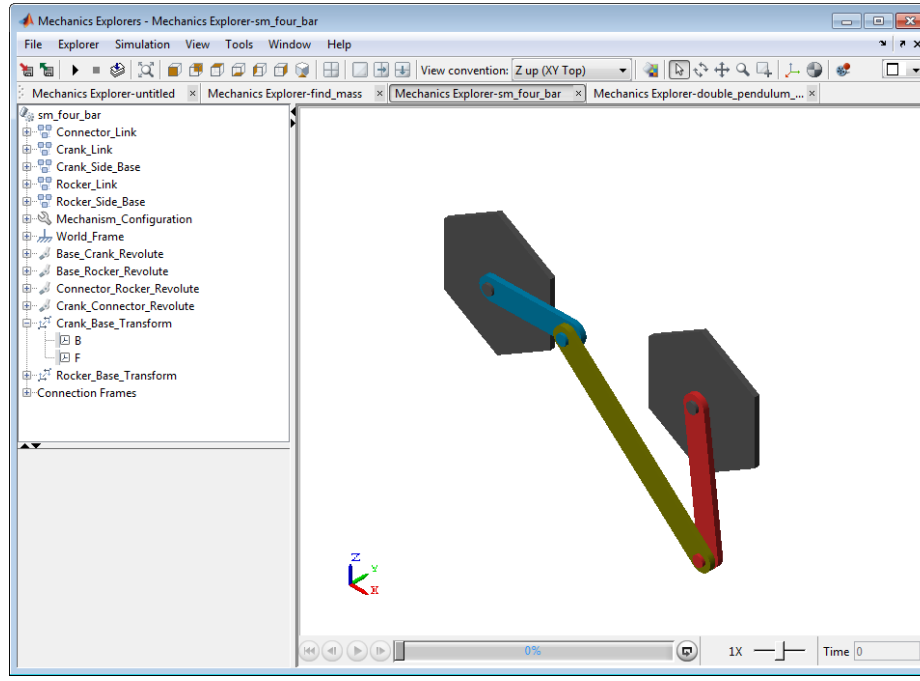
## Open Mechanics Explorer

With Mechanics Explorer configured to start automatically, you do not have to manually open a Mechanics Explorer window. Simply follow the procedure for updating or simulating a model and, if a Mechanics Explorer window is not yet open, a new window opens up.

### Open Mechanics Explorer for Model Update

1 In the Simulink Editor window containing your model, select **Simulation > Update Diagram**. Alternatively, on your keyboard, press **Ctrl+D**.

2 A Mechanics Explorer window opens, displaying your model in its initial state.




## Open Mechanics Explorer for Model Simulation

1. In the Simulink Editor window containing your model, select **Simulation > Run**. Alternatively, on your keyboard press Ctrl+T or, in the Simulink Editor toolbar, press the **Run** button ▶.

2. A Mechanics Explorer window opens. The window displays a 3-D animation of the model.

---

**Note** Each Mechanics Explorer window is named after the model it displays. E.g. Mechanics Explorer-*sm_four_bar*.

---

# Menu and Tool Bars

| In this section... |
| --- |
| “Menu Bar” on page 3-7<br>“Toolbar” on page 3-8 |

Mechanics Explorer provides you menu and tool bars for easy access to tools, including simulation and visualization tools. These two bars contain the majority of the tools which you can use with SimMechanics. Some of the tools provided by the Menu and Tool bars are also accessible in the contest-sensitive menu, which you can open by right-clicking the visualization window in Mechanics Explorer.

## Menu Bar

The Menu bar contains eight menu items, on the left end of the bar, and three desktop control icons, on the right end of the bar.


### Menu Items

| Menu Item | Function |
| --- | --- |
| File | Open and close a file, import and save data |
| Explorer | Save and close a Mechanics Explorer window |
| Simulation | Start and stop a simulation |
| View | Select and modify a model viewpoint |
| Tools | Troubleshoot a model and record an animation |
| Desktop | Dock and undock a Mechanics Explorer window, show and hide the Mechanics Explorer toolbar |

| Menu Item | Function |
|---|---|
| Window | Configure Mechanics Explorer window |
| Help | Search MathWorks documentation for Mechanics Explorer or other product help |

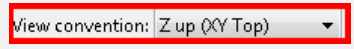
### Desktop Control Icons

| Desktop Control Icon | Function |
|---|---|
| Dock Mechanics Explorers | Open Mechanics Explorer window inside the MATLAB window |
| Undock Mechanics Explorers | Open Mechanics Explorer outside the MATLAB window |
| Close Mechanics Explorers | Close active Mechanics Explorer window |

## Toolbar

The Tool bar contains nine sections.

| Toolbar Icons | Function |
|---|---|
|  | • Save Mechanics Explorer configuration to model<br>• Restore Mechanics Explorer configuration from model |
|  | • Start simulation<br>• Stop simulation |

| Toolbar Icons | Function |
|---|---|
| | • Update model block diagram |
|  | • Fit to view<br>• Front view<br>• Back view<br>• Top view<br>• Bottom view<br>• Left view<br>• Right view<br>• Isometric view |
|  | • Split the screen into four equally sized visualization windows<br>• Use a single full-sized visualization window<br>• Split the screen into two vertical visualization windows<br>• Split the screen into two horizontal visualization windows |
|  | • Change default frame orientation. Choices include Z Up, Z Down, and Yup |
|  | • Change background color of visualization window<br>• Select model component<br>• Rotate model<br>• Pan model<br>• Zoom model<br>• Zoom model region |

| Toolbar Icons | Function |
|---|---|
|  | • Show model frames<br>• Show model Center-of-Masses |
|  | • Record movie of animation |
|  | • Tile multiple Mechanics Explorer windows into a four-matrix<br>• Tile multiple Mechanics Explorer windows into two vertical windows<br>• Tile multiple Mechanics Explorer windows into two horizontal windows<br>• Maximize active Mechanics Explorer window to occupy entire region of visualization window |

# Tree View and Property Panes

**In this section...**

"Introduction to Tree View and Property Panes" on page 3-11

"Browse Model" on page 3-12

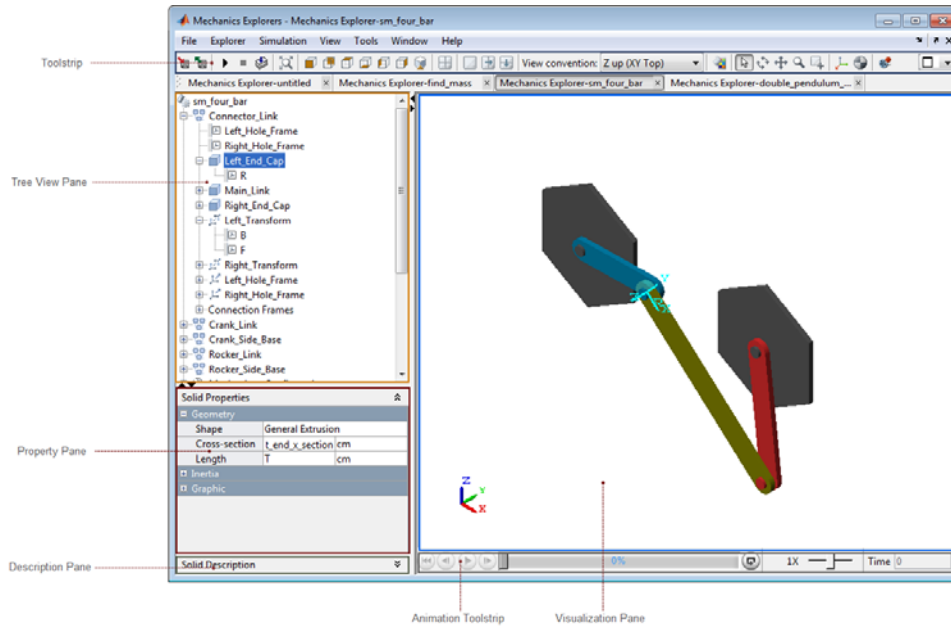
Mechanics Explorer contains **Tree View** and **Property** panes so that you can navigate a model and explore model parameters.

## Introduction to Tree View and Property Panes

By default, Mechanics Explorer displays both a **Tree View** pane and a **Property** pane. With these panes, you can navigate the subsystem and block hierarchy of a model. You can also view the properties of any block in the model.

The **Tree View** browser helps you navigate a mechanical model by organizing blocks into a hierarchical structure. You can select a subsystem in your model, and examine any of its constituent blocks. If a subsystem is in turn built from other subsystems, you can select the new subsystem of interest, and navigate to any of its constituent blocks, all from the model browser. You can also expand any single block to examine any of its frames.

The **Property** pane displays the properties of the component that you selected in the **Tree View** browser. If the component is a block, the pane displays the dialog box parameters for that block. If the component is a frame port, the pane displays other frame ports that directly connect to the selected frame port.




## Browse Model

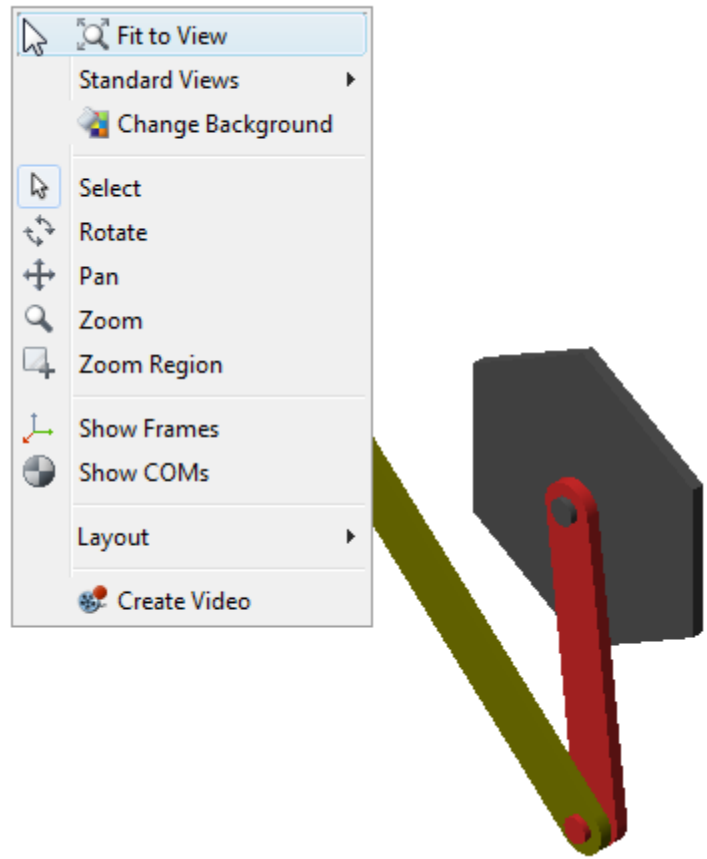
Use the **Tree View** pane to highlight a component and its associated frames in the visualization window, or to view its properties:

1. In the **Tree View** pane, select the component you wish to analyze:
   - The selected component and all associated frames are highlighted in the visualization window
   - The properties of the selected component are displayed in the property viewer
2. Click the expandable menu for the selected component, displaying its frames.
3. Select the frame you wish to examine:

- Only the selected frame is highlighted in the visualization window
- All frames which connect to the selected frame are highlighted in the property viewer

# Context Menu Display Controls

You can right-click the visualization pane to open a context menu, giving you quick access to commonly used visualization and animation tools. The following figure shows an open context menu in a Mechanics Explorer window. You can also access each of these tools from the Mechanics Explorer toolbar. For more information about context menu options, see "Menu and Tool Bars" in this reference documentation.



# Animation Toolstrip

**In this section...**

"Requirements for Animation Recording and Playback" on page 3-15

"How Animation and Playback Work" on page 3-15

"Play Control Buttons" on page 3-16

"Slider and Loop Controls — Simulation Progress Bar" on page 3-16

"Playback Speed Control" on page 3-17

"Playback Time Counter" on page 3-18

## Requirements for Animation Recording and Playback

- With SimMechanics visualization, you record simulation animations in Microsoft Audio Video Interleave® (AVI) format.
  - Animations are recorded as compressed AVI files by the MATLAB `VideoWriter` class using the default Motion JPEG codec, which is available on most operating systems.
  - SimMechanics visualization does not support uncompressed AVI recording.
- To play back AVI files, you need an AVI-compatible media application.
  - You can use an external AVI-compatible player.
  - Alternatively, use the MATLAB `VideoReader` class and its `read` method instead.

## How Animation and Playback Work

Animation is cached during model simulation. What you see during simulation is the animation playback, unless the simulation is slower than the animation caching. In that case, the animation goes no faster than the simulation can produce the cache.

Once a partial or complete animation is cached from simulation, starting the animation again plays back the cache, without running the simulation a second time.

- While simulating for the first time, you can go back to previous animation time steps (already cached), but not forward in time.
- When playing back second, third, etc., time, you can go backward and forward to any time in the cached animation.
- Animation is cached on a per-session basis. This cache is lost when you close your visualization window session.
- Creating a video as a permanent record on your file system outside of MATLAB saves the cached animation permanently.


## Play Control Buttons

Use the play control buttons to play and stop an animation, skip a step, or reset it to the beginning:

| Play Control Button | Use it to... |
|---|---|
| **Start/Stop** | Start/stop the cached animation |
| **Reset** | Move the animation playback time back to zero |
| **Forward Step** | Skip forward in time |
| **Back Step** | Skip backward in time |

## Slider and Loop Controls — Simulation Progress Bar

### Slider Control

The Slider control moves the animation playback time to an arbitrary point in the animation. The Playback time counter displays the resulting playback time.

- If the animation is not running, moving the Slider control changes the animation time from one fixed time to another.
- If the animation is running, moving the Slider control changes the animation time to whatever time you stop the slider. But the animation then continues running from that time.

Without manual control, the Slider control indicates how far in relative time the cached animation has played.

### Loop Control

You use the loop control button to automatically replay an animation from the start once it reaches the end. When you click this button, the cached animation replays indefinitely until you click the **Stop** button.

The Loop control is disabled by default. Enable looping by clicking this button. Disable looping by clicking it again.

### Simulation Progress Bar

The Simulation progress bar indicates how far in time the model simulation has proceeded, as opposed to how far in time the cached animation has played.

Fast models complete simulation before the cached animation is finished playing. Slower models cache and play the animation at a speed limited by how fast the simulation proceeds.

## Playback Speed Control

The Playback speed control adjusts the relative speed at which the animation plays back, as compared to the default speed of 1. The relative playback speed is incremented by multiples of 2 and 1/2.

Moving the Playback speed control to the left slows the playback speed, by a factor of 1/2, 1/4, etc. Moving the Playback speed control to the right speeds up the playback speed, by a factor of 2, 4, etc.

## Playback Time Counter

The Playback time counter is both an indicator and a controller of animation playback time. A valid playback time is one that lies between 0 and the full simulation time, inclusive.

- During animation, or stopped at an intermediate time between start and end, the Playback time counter displays the current or instantaneous animation time.
- Entering a valid animation time in the Playback time counter field and pressing Enter at the keyboard moves the animation playback to the animation frame closest to whatever time you entered.

4

# Functions—Alphabetical List

# smnew

## Purpose

Open new model with common SimMechanics blocks

## Syntax

```
smnew
smnew(modelname)
smnew(modelname,solver)
```

## Description

`smnew` creates a new SimMechanics model, with required and commonly used blocks already on the model canvas, and opens the `sim_lib` library. The default model name is `untitled` and the default solver is `ode45` (the recommended solver).

The new model contains the following blocks:

- World Frame
- Solid
- Rigid Transform
- Solver Configuration
- Mechanism Configuration
- PS-Simulink Converter
- Simulink-PS Converter
- Scope

`smnew(modelname)` creates a newSimMechanics model with the specified name. If you specify an invalid model name, the model name is `untitled`. The default solver is `ode45`, which is the recommended solver.

`smnew(modelname,solver)` creates a new SimMechanics model with the specified name and Simulink® solver type. If you do not specify a model name, then the model name is `untitled`.

## Input Arguments

### modelname

String specifying the name for the new model. Do not include a file extension.

**Default:** `untitled`

### solver

String specifying the name of a Simulink solver. If you choose not to use the default `ode45` solver, other recommended solver types for SimMechanics models are `ode15s`, `ode23t`, and `ode14x`. You can use other Simulink solvers, but, depending on the particular model, they may be less suitable. For more information, see "Improving Performance" and "Setting Up Solvers for Physical Models".

**Default:** `ode45`.

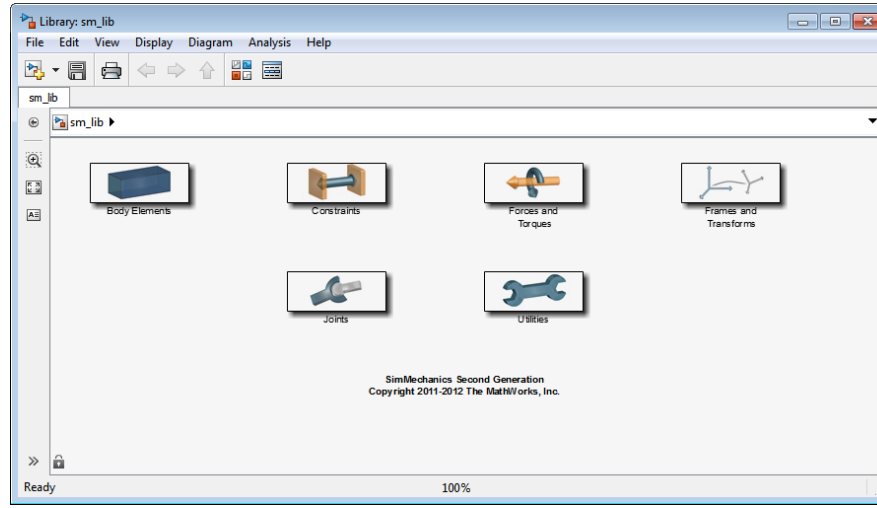
## Examples

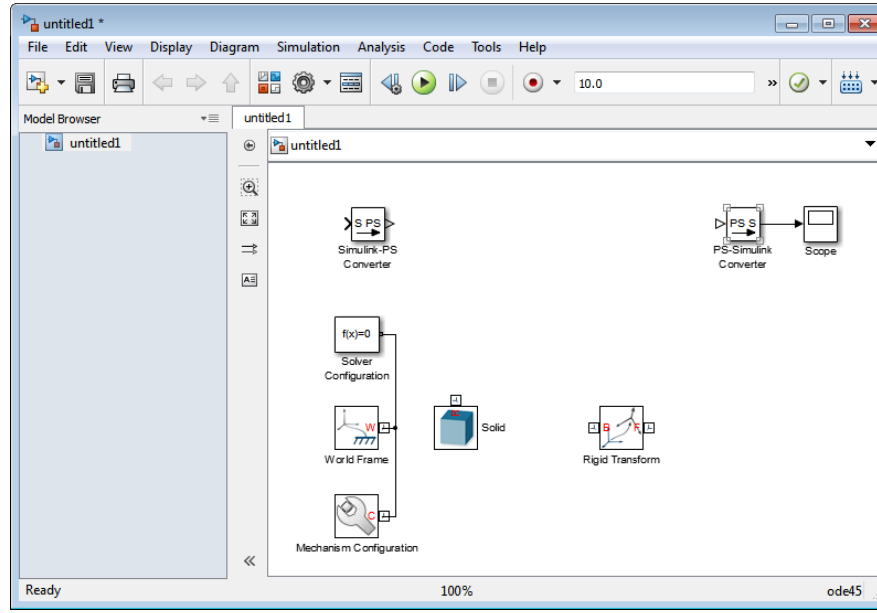
### New SimMechanics Model with Defaults

Create a new SimMechanics model, using the default model name (`untitled`) and the default solver (`ode45`).

```
smnew
```

The command opens the `sm_lib` library, as well as a SimMechanics model called `untitled`, with several commonly used blocks.


Library: sm_lib
File Edit View Display Diagram Analysis Help
sm_lib
sm_lib
Body Elements
Constraints
Forces and Torques
Frames and Transforms
Joints
Utilities
SimMechanics Second Generation
Copyright 2011-2012 The MathWorks, Inc.
Ready
100%

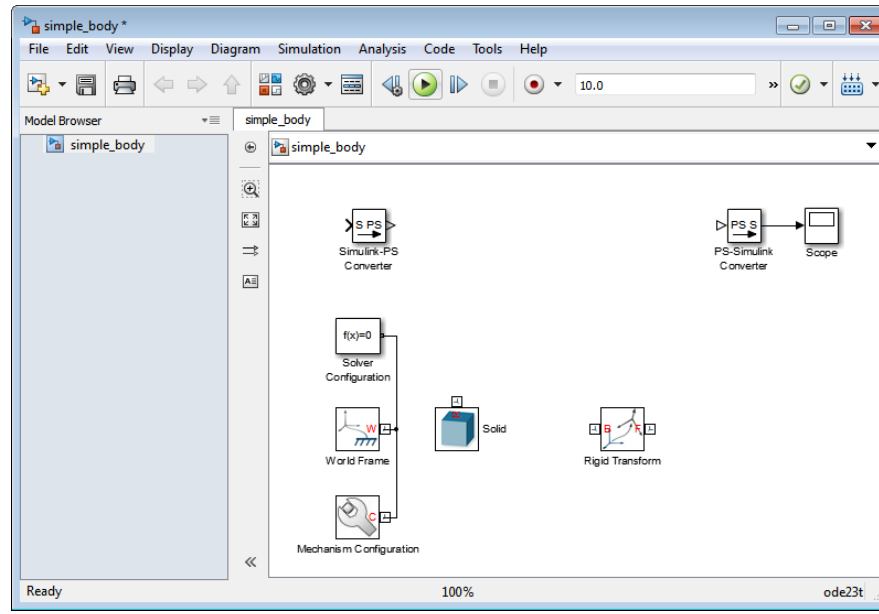

untitled1 *
File Edit View Display Diagram Simulation Analysis Code Tools Help
10.0
Model Browser
untitled1
untitled1
Simulink-PS Converter
PS-Simulink Converter
Scope
f(x)=0
Solver Configuration
World Frame
Solid
Rigid Transform
Mechanism Configuration
Ready
100%
ode45

### Named New SimMechanics Model with Nondefault Solver

Create a new SimMechanics model, named `simple_body`. Use the `ode23t` solver.

```
sm_new('simple_body','ode23t')
```

The command opens the `sm_lib` library, as well as a SimMechanics model called `simple_body`, with several commonly used blocks. The model uses the `ode23t` solver.




## See Also

`sm_lib` | `smimport`

# smimport

## Purpose

Generate new model from SimMechanics Import XML file

## Syntax

```
H = smimport('filename')
H = smimport('filename', ModelName, 'modelname')
```

## Description

`H = smimport('filename')` imports SimMechanics XML file `filename.xml` into a new Simulink model with the same name. SimMechanics generates the new model according to XML file specification, adding and connecting any required blocks. For CAD-generated XML files, the new model represents a CAD assembly.

`filename` is a string specifying the XML file name. This string must be a valid model name, using any combination of letters, digits, and underscores such that the first character is a letter and the length of the string is smaller than `namelengthmax`. If `filename` is invalid, SimMechanics generates a valid name based on `filename`.

If a model with the same name is open,`smimport` returns a warning and appends an integer to `filename` to create a unique model name. If a model with the same name exists in the MATLAB workspace or path, `smimport` returns a warning asking you to change the model name.

`filename` supports path specification relative to the MATLAB working directory. Alternatively, `filename` can contain the full path to the corresponding XML file. In the absence of a path, `smimport` searches for `filename` inside the MATLAB path.

By default, `smimport` assumes `filename` refers to an XML file in the MATLAB workspace or path. `filename` does not require an explicit XML extension. If `filename` is not an XML file, or if it is not in the MATLAB workspace or path, `smimport` returns an error.

`H = smimport('filename', ModelName, 'modelname')` imports XML file `filename` into a new SimMechanics model with name `modelname`.

## Tips

- `smimport` provides CAD integration with SimMechanics. Use SimMechanics Link™ to generate a valid SimMechanics XML file from SolidWorks™, Creo™ (Pro/Engineer™), or AutoCAD

Inventor™. Then, use `smimport` to generate a corresponding SimMechanicsmodel.

## Input Arguments

### filename

String specifying input XML file name. Replace *`filename`* with the XML file name you want to import. The `.xml` extension is optional.

This input is required.

**Default:** `''`

### Name-Value Pair Arguments

Specify optional comma-separated pairs of `Name,Value` arguments, where `Name` is the argument name and `Value` is the corresponding value. `Name` must appear inside single quotes (`' '`). You can specify several name and value pair arguments in any order as `Name1,Value1,...,NameN,ValueN`.

#### ModelName

String specifying generated SimMechanics model name.

This input is optional.

**Default:** `''`

## Examples

### Import Robot Assembly with Default Name

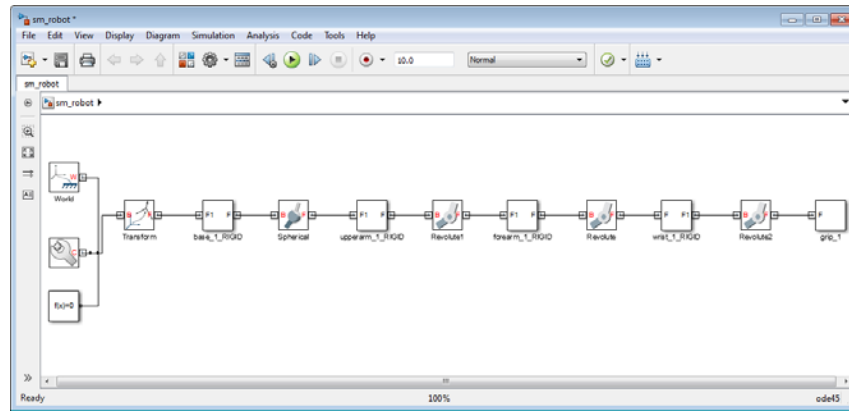
Generate robot model from SimMechanics Import XML file `sm_robot.xml`. The XML file represents a robot arm CAD assembly. You can access the XML file in the following directory:

```
matlabroot\toolbox\physmod\sm\smdemos\import\robot
```

The directory contains an additional set of STL files that specify 3-D surface geometry for all CAD parts.

1 At the command line, enter

```
smimport('sm_robot.xml');
```




2 In the new model window, select **File > Save**.

3 Enter model name and select storage directory

### Import Robot Assembly with Custom Name

You can rename the robot model at the MATLAB command line. To rename the model `robot_arm`, add the model name to the `smimport` parameters.

1 At the MATLAB command line, enter

```
smimport('sm_robot', 'ModelName', 'robot_arm');
```

2 SimMechanics software imports file `robot.xml` and generates a new model called `walkingrobot`.

## See Also

`smnew` | `sm_lib`

## Purpose

Open the SimMechanics block library

## Syntax

```
sm_lib
```

## Description

`sm_lib` opens a new window containing the complete SimMechanics block library. Use this window to access the six SimMechanics block sublibraries:

- Body Elements
- Constraints
- Forces and Torques
- Frames and Transforms
- Joints
- Utilities

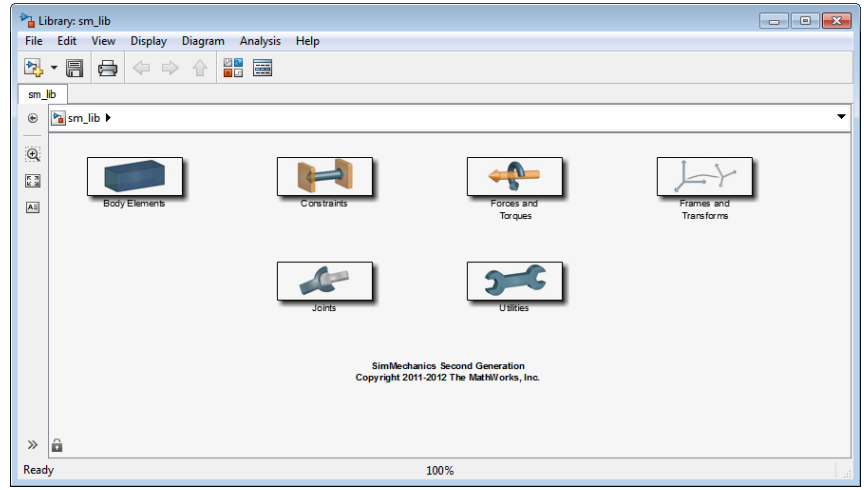
## Tips

- Use `sm_lib` to open the SimMechanics block library from the MATLAB command line.
- `sm_lib` loads only the SimMechanics block library, so you do not have to wait for other Simulink libraries to load.

## Examples

### Open block library from MATLAB Command Line

1. At the MATLAB command line, enter `sm_lib`. A new window containing the six SimMechanics sublibraries opens.




**2** Double-click a sublibrary to access any underlying blocks.

## See Also

`smnew` | `smimport`

# Index

## C

configuration parameters
pane 2-2
Conflicting reference frames: 2-8
Invalid color vector: 2-4
Open Mechanics Explorer on model update or simulation 2-12
Overspecified targets in kinematic loops: 2-11
Redundant block: 2-7
Repeated vertices in a cross-section: 2-5
Rigidly constrained block: 2-9
Unconnected frame port: 2-6
Unsatisfied desired state targets: 2-10