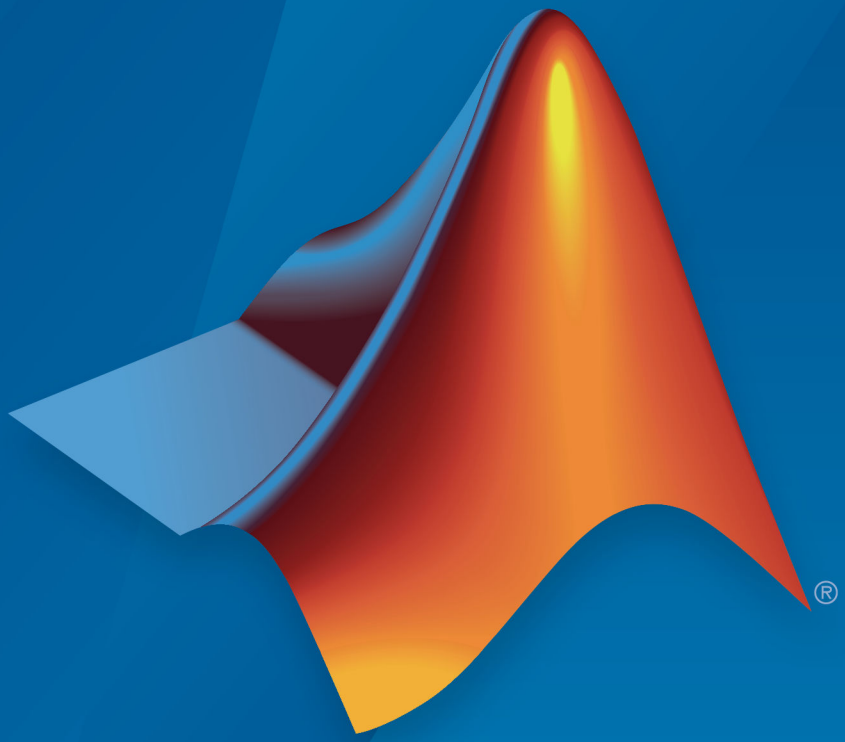
Simulink® 3D Animation™ Release Notes



MATLAB® & SIMULINK®

MathWorks®

How to Contact MathWorks

Latest news: www.mathworks.com

Sales and services: www.mathworks.com/sales_and_services

User community: www.mathworks.com/matlabcentral

Technical support: www.mathworks.com/support/contact_us

Phone: 508-647-7000

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

Simulink® 3D Animation™ Release Notes

© COPYRIGHT 2002–2019 by HUMUSOFT s.r.o. and 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

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

Contents

R2019a

Coordinate Transformation Utilities 1-2

FBX format import 1-2

UAV Competition Example 1-3

R2018b

VR RigidBodyTree block: Automate the visualization of Robotics System Toolbox rigid body tree objects in virtual reality worlds 2-2

R2018a

Improved Collision Detection: Sense collisions of virtual world objects using point clouds, raytracing, and primitive geometries 3-2

Offscreen Rendering: Use the VR to Video block as a video stream source without the need to keep the viewer window open during simulation 3-2

X3D Standard Format: Represent objects in a virtual reality 3D environment natively using the X3D ISO standard. 3-2

R2017b

Collision Detection: Sense collisions of virtual world objects using point clouds, raytracing, and primitive geometries **4-2**

R2017a

Select Mode: Select and highlight virtual world objects using mouse pointer in the 3D World Editor View pane **5-2**

R2016b

Unified Robot Description Format (URDF) Support: Import visual representations of robots into a virtual world **6-2**

Virtual Figure Rendering: Turn off rendering for improved performance **6-2**

Virtual Figures in MATLAB Live Scripts: Include code in MATLAB Live Scripts for displaying virtual reality figures **6-3**

Saving Virtual Worlds: Suppress thumbnails **6-3**

R2016a

X3D Syntax Highlighting: Identify X3D language elements in MATLAB Editor **7-2**

STL to X3D File Conversion: Convert CAD models in STL format to X3D formats 7-2

Virtual World Previews: View virtual worlds before opening files in the 3D World Editor 7-2

R2015b

X3D standard support for XML-encoded .x3d and classic VRML-encoded .x3dv file formats 8-2

Face count statistics for IndexedFaceSet and IndexedLineSet nodes to assess model complexity 8-2

Removal of legacy viewer 8-2

R2015a

Space mouse support for Mac OS X 9-2

Conversion between MATLAB Patch objects and IndexedFaceSetsConversion 9-2

Improved geometry optimization that preserves per-vortex colors of objects 9-2

Option to tune stereoscopic vision figure properties 9-2

Definition of scripts in virtual worlds in separate files 9-2

Browsing to specify file URLs in 3D World Editor 9-2

Option to save virtual worlds to use in VRML viewers 9-3

Display of event fields in 3D World Editor tree pane 9-3

More responsive navigation for large virtual worlds **9-3**

R2014b

Internal viewer sound support **10-2**

Internal viewer stereoscopic vision support **10-2**

Improved Orbisnap standalone viewer interface **10-3**

Inclusion of vr.canvas objects in panels **10-3**

Parameterized vrfigure constructor **10-3**

Navigation panel not shown in screen captures **10-3**

Blaxxun Contact VRML plug-in no longer supported **10-3**

R2014a

Import of STL and Physical Modeling XML files directly into 3D World Editor for authoring VRML scenes **11-2**

Polygon reduction option for scene objects in 3D World Editor for enhancing rendering performance **11-2**

Improved Timing of 2D Animation Recording **11-2**

R2013b

Web Viewer for remote access to virtual worlds using HTML5-enabled web browser technology **12-2**

Interactive setting of pivot points for examining scenes **12-2**

R2013a

Multiple-pane grid view in 3D World Editor for enhanced scene viewing and modeling **13-2**

Current-view indication and navigation-panel minimization in 3D World Editor, Simulink 3D Animation viewer, and vrplay animation viewer **13-2**

MATLAB Current Folder panel for opening VRML file in 3D World Editor **13-3**

Option to start 3D World Editor with previous session's layout **13-3**

Expansion of subtree in the tree structure pane of 3D World Editor **13-3**

Multiple 3D World Editor instances **13-3**

Blaxxun Contact viewer **13-4**

R2012b

VRML syntax highlighting for MATLAB Editor **14-2**

Mac OS support for Joystick Input Block and joystick object **14-2**

Multiple nodes selection and parent node creation, facilitating hierarchical scene edits in 3D World Editor **14-2**

Interactive color selection for SFColor and MFColor fields using color palette in 3D World Editor **14-2**

Use of MATLAB expressions to enter 3D World Editor's numeric field values **14-2**

Additional grids for Components library **14-2**

vredit function for opening 3D World Editor **14-3**

MATLAB buttons for editor and animation player **14-3**

Simulation Stepper support **14-3**

R2012a

VR Tracer Block Enhancements **15-2**

3D World Editor Enhancements **15-2**

stl2vrml Function Accepts Additional Units **15-2**

Animation Recording Uses MATLAB VideoWriter Technology **15-2**

New Example For Global Coordinates **15-2**

R2011b

VR Source Block **16-2**

Triad of Coordinate Axes **16-2**

Controlling Full-Screen Mode **16-2**

Global Coordinates for vrnode Objects **16-3**

3D World Editor Enhancements **16-3**
Template Files **16-3**
Add and Paste Nodes Throughout the Hierarchy **16-3**
Copy Nodes Between VRML Files **16-3**
Sensing Objects Added to Components Library **16-3**
Improved VRML Script Editing **16-4**
Improved VRML Tree Structure Pane **16-4**

Specify a Default Viewpoint From the Command Line **16-4**

New or Enhanced Examples **16-4**

Legacy Viewer **16-4**

R2011a

Geometric and Background Objects Added to 3D World Editor Library **17-2**

Switching Between Bindable Nodes **17-2**

3D World Editor Accessible from the Start Menu **17-2**

Preference Options for 3D World Editor Startup Position . . . **17-2**

3D World Editor Option to Hide Default Values **17-3**

Node Fields Are Listed Before Node Children Fields in 3D World Editor **17-3**

Setting Viewpoint Values in the 3D World Editor Based on Camera Position **17-3**

Faster Texture Handling **17-3**

New Examples **17-3**

R2010b

3D World Editor **18-2**

stl2vrml Function **18-3**

Simulink 3D Animation Viewer Supported on Linux **18-3**

R2010a

vrnode/getfield Now Returns Field values of SFNode and MFNode Types **19-2**

VRML Field Data Types **19-2**

Camera Bound to Viewpoint Shortcut Change **19-2**

Navigation Method Changes **19-2**

Updated vrphysmod **19-2**

VR Sink Block Now Supports Setting Node eventIn Events **19-3**

vrmorph Example **19-3**

Some Dialog Boxes Are No Longer Modal 19-3

R2009b

Alternate Mode of Virtual Scene Viewer Now Default 20-2

Support on the 64-Bit Intel Macintosh Platform 20-2

VRML Files Recognized in MATLAB Current Folder Browser 20-2

Changed vr.canvas Units Property Default 20-3

Improved Performance of Orbisnap in Remote Configuration 20-3

R2009a

Alternate Mode of Virtual Scene Viewer Integrated with MATLAB Figures 21-2

vrplay Updated 21-2

New Functions 21-2

New Object 21-3

Virtual Scenes Now Open Automatically 21-3

Product Files Moved 21-3

Saving and Restoring the Complete SimState 21-3

R2008b

New VR Tracer Block **22-2**

Updated vrnode Method **22-2**

New vrworld Method **22-2**

New Example **22-2**

R2008a

VR Sink Block Now Supports Matrix Signals as Input **23-2**

New vrworld Object Option **23-2**

Enhanced vrplay Shortcut Keys **23-2**

New Examples **23-2**

R2007b

New VR To Video block **24-2**

Joystick and Space Mouse Input Functionality Added to MATLAB Toolbox Interface **24-2**

Joystick and Space Mouse Input Support Added for Linux . . **24-2**

New Axes Transformation and Math Functions **24-2**

Magellan Space Mouse Block Renamed to Space Mouse Input **24-2**

R2007a

Full-Screen Mode in Virtual Reality Toolbox Viewer and Orbisnap Stand-Alone Viewer **25-2**

R2006b

VR Text Output Block Added **26-2**

Frames Per Second Added to AVI File Playback Options **26-2**

New 3Dconnexion Devices **26-2**

Joystick Input Block **26-2**

R2006a

New Utilities Library **27-2**

VR Animation Playback Controls **27-2**

R14SP3

Bug Fixes

R14SP2+

Virtual Reality Toolbox Support of MATLAB Compiler Product **29-2**

Virtual Reality Toolbox Viewer **29-2**

Virtual Reality Toolbox Preferences Dialog **29-2**

vrsetpref and vrgetpref Changes **29-2**

Space Mouse Block Enhancement **29-2**

R14SP2

Virtual Reality Toolbox Stand-Alone Viewer **30-2**

Virtual Reality Toolbox Viewer **30-2**

R2019a

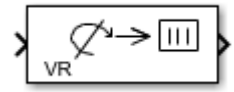
Version: 8.2

New Features

Bug Fixes

Coordinate Transformation Utilities

Simulink 3D Animation provides the following blocks in the **Simulink 3D Animation/ Utilities** library:



VR Rotation to Rotation Matrix

Convert the axis-angle representation of rotations in the VR world to a Rotation Matrix.

To apply the same transformation in MATLAB®, use the `vrrotvec2mat` function

VR to MATLAB Coordinates

Convert VR coordinates in [*x*, *y*, *z*] notation to MATLAB coordinates in [*x*, *z*, -*y*] notation.

To apply the same transformation in MATLAB, use the `vrcoordvr2m` function.

MATLAB to VR Coordinates

Convert MATLAB coordinates in [*x*, *z*, -*y*] notation to VR coordinates in [*x*, *y*, *z*] notation.

To apply the same transformation in MATLAB, use the `vrcoordm2vr` function.

FBX format import

You can import FBX files as nodes into virtual reality worlds using the Simulink 3D Animation viewer as well as the `vrimport` function.

UAV Competition Example

Simulink 3D Animation ships with an example model demonstrating a typical UAV competition scenario.

Use the `sl3dex_uav` example to view a quadcopter flying a predefined trajectory through an obstacle course. You can also switch to manual navigation using a SpaceMouse by flipping the SpaceMouse/ Predefined Navigation Switch in the model.

R2018b

Version: 8.1

New Features

Bug Fixes

VR RigidBodyTree block: Automate the visualization of Robotics System Toolbox rigid body tree objects in virtual reality worlds

The new VR RigidBodyTree block inserts graphical information present in the Robotics System Toolbox™ rigid body tree object into a virtual reality world and animates it based on the manipulator configuration. The same functionality is available through the Simulink 3D Animation MATLABinterface using the `vrinsertrobot` and `vrupdaterobot` functions.

R2018a

Version: 8.0

New Features

Improved Collision Detection: Sense collisions of virtual world objects using point clouds, raytracing, and primitive geometries

You can use collision detection to accurately model physical constraints of objects in the real world, where generally two objects cannot be in the same place at the same time. You can use collision detection node outputs to:

- Change the state of other virtual world nodes.
- Apply MATLAB algorithms to collision data.
- Drive Simulink models.

You can use the `PointPickSensor`, `LinePickSensor`, and `PrimitivePickSensor` to model sensors such as ultrasonic, lidar, and touch sensors.

For examples of using collision detection, see `vrcollisions` and `vrmaze`.

Offscreen Rendering: Use the VR to Video block as a video stream source without the need to keep the viewer window open during simulation

You can use the VR to Video block to configure a video stream. You can then close the viewer window and use the video stream elsewhere in the simulation. You no longer need to keep all the streaming windows open over the course of the simulation. Use this option in situations where you have multiple 2D video signals from virtual world synthetic cameras.

For an example on using off-screen rendering, see `vr_octavia_mirror`.

X3D Standard Format: Represent objects in a virtual reality 3D environment natively using the X3D ISO standard.

X3D is natively supported in Simulink 3D Animation. You can easily convert CAD models to X3D. To convert a .wrl file to .x3d or .x3dv, open it and save it with the new extension.

X3D also includes components that allow implementation of collision detection in Simulink 3D Animation. For more information, see Picking component.

R2017b

Version: 7.8

New Features

Bug Fixes

Collision Detection: Sense collisions of virtual world objects using point clouds, raytracing, and primitive geometries

You can use collision detection to accurately model physical constraints of objects in the real world, where generally two objects cannot be in the same place at the same time. You can use collision detection node outputs to:

- Change the state of other virtual world nodes.
- Apply MATLAB algorithms to collision data.
- Drive Simulink models.

For example, you can use geometric sensors for robotics modeling.

To set up collision detection, define collision (pick) sensors that detect when they collide with targeted surrounding scene objects. The virtual world sensors resemble real world sensors, such as ultrasonic, lidar, and touch sensors. You can add these types of collision sensors:

- `PointPickSensor` – Point clouds that detect which of the points are inside colliding geometries
- `LinePickSensor` – Ray fans or other sets of lines that detect the distance to the colliding geometries
- `PrimitivePickSensor` – Primitive geometries (such as a cone, sphere, or box) that detect colliding geometries

The `isActive` output property of a sensor becomes `TRUE` as soon as a collision occurs. If you have a model associated with the virtual reality scene, you can use a VR Source block to read the sensor `isActive` property and the current position of the object for which the sensor is defined. You can use a VR Sink block to define the behavior of the object, such as its position, rotation, or color.

For examples of using collision detection, see `vrcollisions` and `vrmaze`. For additional information about collision detection, see Detect Object Collisions.

R2017a

Version: 7.7

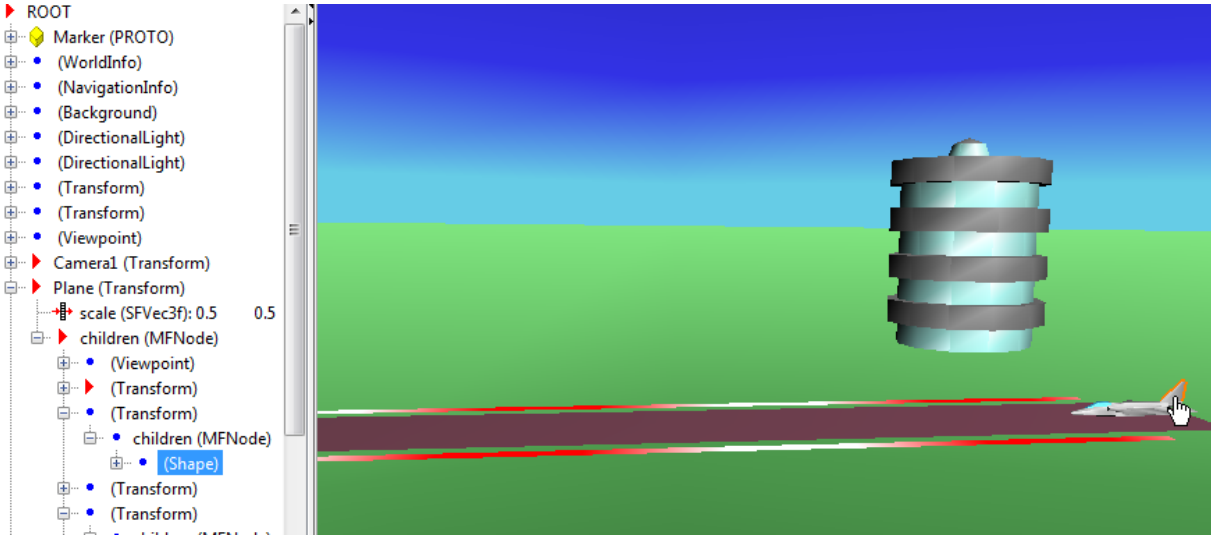
New Features

Bug Fixes

Select Mode: Select and highlight virtual world objects using mouse pointer in the 3D World Editor View pane

To select and highlight virtual world objects using the mouse pointer in the 3D World Editor view pane, use the select mode. To enable the select mode, in the 3D World Editor toolbar, click the button. Then click an object (or **Ctrl**-click multiple objects) in the 3D World pane or a node in the tree structure pane. In the view pane, selected virtual world objects are highlighted with an orange outline. When the cursor hovers over a selectable object in the view pane, the cursor shape changes to a hand symbol. If you do not want objects to be highlighted when you select them, in the view pane right-click and clear the **Rendering** > **Highlight Selected Objects** option.

For example, with **Highlight Selected Objects** enabled, if you click the tail of the plane in this virtual world, the corresponding Shape node is highlighted.



To return to the default navigation mode, click the button. To specify the default selection and highlighting behavior, use the new **View pane mouse behavior** and **Highlight selected objects** 3D World Editor preferences.

For additional information, see Import Visual Representations of Robot Models.

R2016b

Version: 7.6

New Features

Bug Fixes

Unified Robot Description Format (URDF) Support: Import visual representations of robots into a virtual world

You can import robot and robot environment 3D models defined using unified robot description format (`.urdf`) or simulation description format (`.sdf`) into a virtual world in Simulink 3D Animation. The URDF and SDF files are XML files for representing a robot model. You can import visual properties of the objects present in the URDF or SDF files into corresponding hierarchical structure of `Transform` nodes containing `IndexedFaceSet` objects with textures.

URDF and SDF files use Collada DAE and STL formats to define visual properties of scene objects. In addition to existing STL format import, the new URDF and SDF import feature also supports a subset of the Collada DAE 3D file format (geometric shape objects with textures).

To import visual representation of robots, in the 3D World Editor select **Nodes > Import From**. Then select **STL File**, **Physical Modeling XML File**, **URDF File**, **SDF File**, or **COLLADA File**. From the MATLAB command line, you can use the `vrimport` function to import `.urdf`, `.dae`, `.stl`, and `.sdf` files into a virtual world.

For examples, see Import Visual Representations of Robot Models.

Virtual Figure Rendering: Turn off rendering for improved performance

You can use the new `Rendering` property to turn off rendering for a `vrfigure` or `vr.canvas` object in the Simulink 3D Animation Viewer. Turning off rendering improves performance. For example, if your code performs batch operations on a virtual figure, you can turn off rendering for the duration of the processing.

You can also use the new **Simulink 3D Animation > World > Default Figure Rendering** preference (`DefaultFigureRendering`) to specify whether to render newly created virtual reality figure or canvas objects.

Virtual Figures in MATLAB Live Scripts: Include code in MATLAB Live Scripts for displaying virtual reality figures

You can now include code that results in displaying virtual reality figures and canvas objects in MATLAB Live Scripts and display the results in the Live Editor.

Saving Virtual Worlds: Suppress thumbnails

To suppress creating a thumbnail image used for virtual world preview, you can use the new `'-nothumbnail'` argument for the `vrworld/save` function.

To save a copy of all the resources (additional files) used by the world, so that the world can be transferred to a different computer, specify `'-export'` as an argument for the `vrworld/save` function. Prior to R2016b, you specified `'true'` to perform an export. R2016b supports that syntax for backward compatibility.

R2016a

Version: 7.5

New Features

Bug Fixes

X3D Syntax Highlighting: Identify X3D language elements in MATLAB Editor

You can display X3D syntax highlighting in the MATLAB Editor. You can specify colors for highlighting X3D syntax. See VRML and X3D Syntax Highlighting.

STL to X3D File Conversion: Convert CAD models in STL format to X3D formats

To convert CAD models in STL format (`.stl` files) to .X3D format (`.x3d` or `.x3dv` files) or VRML format (`.wrl` files), use the `stl2vrml` function.

Virtual World Previews: View virtual worlds before opening files in the 3D World Editor

In R2016a the 3D World Editor, when you save a virtual world, creates a thumbnail of the virtual world. When you open that virtual world, when you select the virtual world file you see the thumbnail preview in the file browser.

In the Simulink 3D Animation Viewer, when you select a file in the file browser, you see a thumbnail for that virtual world, if the file has been saved in R2016a.

The VR Sink, VR Source, and VR To Video blocks also provide a thumbnail preview. In the Block Parameters dialog box **Source file** parameter, selecting a file in the file browser displays a thumbnail, if available.

To generate thumbnails whenever you open a virtual world file, use this command:

```
vrsetpref('AutoCreateThumbnail','on')
```

Setting this preference to `'on'` can be helpful if you download multiple virtual worlds from the web, without saving them. Creating thumbnails on file open provides thumbnails the next time someone browses through the downloaded worlds.

R2015b

Version: 7.4

New Features

Bug Fixes

Compatibility Considerations

X3D standard support for XML-encoded .x3d and classic VRML-encoded .x3dv file formats

Simulink 3D Animation now supports the X3D standard format. The support includes XML encoded (`.x3d`) and Classic VRML encoded (`.x3dv`) files. X3D support is for versions from version 3.0 up to version 3.3. Support is for X3D files that contain components that comply to the `Immersive` profile. Nevertheless, VRML97 remains the default virtual world file format.

You can use MATLAB commands to read and save X3D files and to associate X3D files with Simulink models. For additional information about X3D support in Simulink 3D Animation, see X3D Support.

By default, the new `%e` file name token for animation recording and frame capture uses the same file extension (`.wrl`, `.x3d`, or `.x3dv`) as the extension of the virtual world that drives the animation. For details, see Define File Name Tokens.

For information about the supported X3D specification, see ISO/IEC 19775-1:2013. For information about supported X3D encoding, see ISO/IEC 19776-1.3:201x and ISO/IEC 19776-2.3:201x.

Face count statistics for IndexedFaceSet and IndexedLineSet nodes to assess model complexity

In the 3D World Editor, you can get face count statistics for nodes such as `IndexedFaceSet` and `IndexedLineSet` nodes. To get the counts, select **Nodes** > **Face Count Statistics**.

Removal of legacy viewer

Simulink 3D Animation no longer includes the legacy viewer (`internalv4`). The Simulink 3D Animation Viewer, Simulink 3D Animation Web Viewer, and Orbisnap viewer supercede the legacy viewer.

Compatibility Considerations

You can set the `DefaultViewer` preference to `internal` or `web`. If in a previous release you used the `vrsetpref` command to set `DefaultViewer` to `internalv4` or `internalv5`, starting in R2015b the value converts to `internal`.

R2015a

Version: 7.3

New Features

Bug Fixes

Space mouse support for Mac OS X

You can use space mouse input devices on Mac OS X platforms. In earlier releases, you could use a space mouse only on Windows® platforms.

Conversion between MATLAB Patch objects and IndexedFaceSetsConversion

You can use the `vrpatch2ifs` function to convert a MATLAB `Patch` object to a VRML `IndexedFaceSet` node.

You can use the `vrifs2patch` function to convert an `IndexedFaceSet` node to a `Patch` object.

Improved geometry optimization that preserves per-vortex colors of objects

The per-vertex colors of each object are now preserved.

Option to tune stereoscopic vision figure properties

You can use the `vr.utils.stereo3d` class properties to tune stereoscopic vision. For example, you can specify yellow and blue as the colors to use for the anaglyph stereoscopic effect, instead of the default red and cyan combination. You can also specify camera settings such as camera angle and camera offset.

Definition of scripts in virtual worlds in separate files

For the `url (MFString)` field in a `Script` node, you can now specify a path to a JavaScript® file, or you can continue to specify the script code in the `url` field, which was the only option in previous releases. For details, see Edit VRML Scripts.

Browsing to specify file URLs in 3D World Editor

In the 3D World Editor, you can now use a browse dialog box to specify URLs for nodes such as `ImageTexture` and `Inline`. You can continue to specify URLs by typing the path. For details, see Specify a URL.

Option to save virtual worlds to use in VRML viewers

The `vrworld/save` method now includes an optional third argument for exporting to a VRML file copies of all relatively specified resources (such as textures) for a virtual world, as well as resources from the Simulink 3D Editor library. This feature supports the use of a Simulink 3D Animation virtual world outside of Simulink 3D Animation.

Display of event fields in 3D World Editor tree pane

You can display `eventIn` and `eventOut` nodes in the 3D World Editor tree pane. You can either click the **Show Events** button or select **Tree > Show Events**.

You can perform an `IS` mapping for events in a `PROTO` declaration.

More responsive navigation for large virtual worlds

In the 3D World Editor and Simulink 3D Animation Viewer, navigation of large virtual works is more responsive than in previous releases.

R2014b

Version: 7.2

New Features

Bug Fixes

Internal viewer sound support

The following Simulink 3D Animation components now support listening to sound in virtual worlds:

- Simulink 3D Animation Viewer
- 3D Animation Player
- `vr.canvas` on a figure window

To add sound to a virtual world, use a `VRML Sound` node. See Add Sound to a Virtual World.

To listen to sound for a virtual world, use a computer that supports sound. For details, see Listen to Sound in a Virtual World.

The Orbisnap standalone viewer shipped with Simulink 3D Animation also supports sound.

Internal viewer stereoscopic vision support

Stereoscopic vision support allows you to view a virtual world using 3D effects, so that elements in the virtual world appear to come forward or back from the plane of the monitor. You can use either of these stereoscopic vision approaches:

- Anagylph — Use red/cyan 3D glasses. Viewing a virtual world in this mode causes the colors to appear as almost grayscale. This approach does not require any special computer hardware or software.
- Active stereo — Use active shutter 3D glasses. This approach preserves color effects and produces more powerful 3D effects. Active stereo requires a specially configured computer and monitor setup.

The following Simulink 3D Animation components now support stereoscopic vision:

- 3D World Editor
- Simulink 3D Animation Viewer
- 3D Animation Player
- `vr.canvas` on a figure window

For details, see View a Virtual World in Stereoscopic Vision.

The Orbisnap standalone viewer shipped with Simulink 3D Animation also supports stereoscopic vision.

Improved Orbisnap standalone viewer interface

The interface of the Orbisnap standalone viewer shipped with Simulink 3D Animation has an improved look and feel.

For more information, see Orbisnap Standalone Viewer.

Inclusion of vr.canvas objects in panels

You can now include a `vr.canvas` object as a child object in a MATLAB `uipanel` object.

Panels arrange graphical user interface components into groups. By visually grouping related controls, panels can make the user interface easier to understand. A panel can have a title and various borders.

Parameterized vrfigure constructor

You can now use the `vrfigure` constructor with name-value pairs to set property values. For example:

```
fig = vrfigure(vrlights,'Antialiasing','on')
```

For a list of properties that you can set, see the documentation for `vrfigure.get`.

Navigation panel not shown in screen captures

Screen captures of a virtual world in the Simulink 3D Animation Viewer omit the navigation panel, resulting in a cleaner image.

Blaxxun Contact VRML plug-in no longer supported

The Blaxxun Contact® VRML plug-in is no longer shipped with, or supported by, Simulink 3D Animation.

In R2014b, Simulink 3D Animation now supports sound and stereoscopic vision, features that previously required the Blaxxun Contact viewer.

R2014a

Version: 7.1

New Features

Bug Fixes

Compatibility Considerations

Import of STL and Physical Modeling XML files directly into 3D World Editor for authoring VRML scenes

You can use the 3D World Editor to import STL and Physical Modeling XML files directly into a VRML scene. STL and Physical Modeling XML files usually originate from CAD models. You can integrate content from those files into a virtual world, converting the structure into VRML.

To import an STL or Physical Modeling XML file, in the 3D World Editor, right-click the `Root` or an `MFNode` node and from the **Import From** menu item, select either **STL File** or **Physical Modeling XML File**. For details, see Import STL and Physical Modeling XML Files Directly into a VRML Virtual World.

Polygon reduction option for scene objects in 3D World Editor for enhancing rendering performance

For a node that is, or includes, an `IndexedFaceSet` node, you can improve rendering speed by reducing the number of polygons in the `IndexedFaceSet`. Choose the polygon reduction factor that produces your desired balance of rendering performance and quality.

For details, see Reduce Number of Polygons for Shapes.

Improved Timing of 2D Animation Recording

AVI file playback speed now corresponds to the original Simulink model simulation time. For details, see Record Files in the Audio Video Interleave (AVI) Format.

Compatibility Considerations

The `vrfigure` `Record2DFPS` property has a new default value of `auto`, to support this improved timing. Previously, the default was `15`.

R2013b

Version: 7.0

New Features

Bug Fixes

Web Viewer for remote access to virtual worlds using HTML5-enabled web browser technology

Use the Simulink 3D Animation Web Viewer to access virtual worlds with an HTML5-enabled Web browser. You can open a virtual world in Simulink 3D Animation on a host computer and then view it remotely in a Web browser on another computer. You do not need to install Simulink 3D Animation on the remote computer.

The Web Viewer supplements the Simulink 3D Animation Viewer. It also supplements the VRML plug-in Blaxxun Contact viewer and the Orbisnap viewer, which come with Simulink 3D Animation. Some benefits of the Web Viewer include:

- Multiplatform remote viewing of Simulink 3D Animation virtual worlds
    - Support for HTML5-enabled browsers on Microsoft® Windows, Macintosh, and Linux® platforms
- No additional software installation required on client computers
- Access to HTML5 browser features for creating customized pages with virtual reality visualization

For details, see Simulink 3D Animation Web Viewer.

Interactive setting of pivot points for examining scenes

You can set a pivot point interactively when you are in Examine mode in the 3D World Editor, Simulink 3D Animation Viewer, and the Simulink 3D Animation Web Viewer. You can use a pivot point for rotating a virtual world around, using a mouse.

For details, see Pivot Point.

R2013a

Version: 6.3

New Features

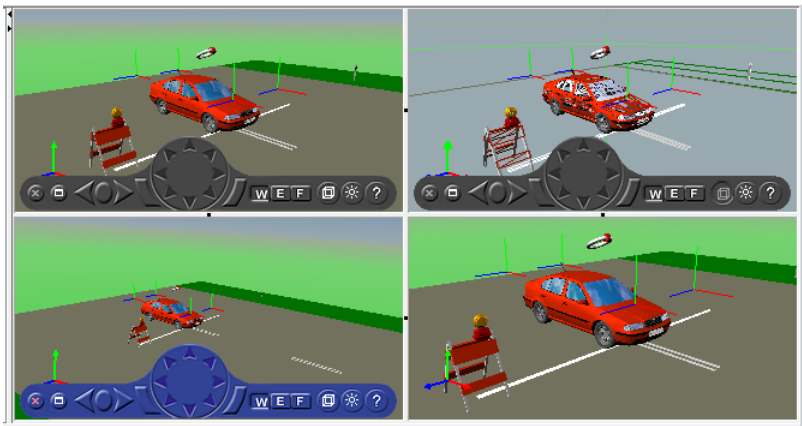
Bug Fixes

Compatibility Considerations

Multiple-pane grid view in 3D World Editor for enhanced scene viewing and modeling

You can view the virtual world in one pane, two horizontal panes, two vertical panes, or four panes. You can control the navigation of each pane independently. For example, you can set viewpoints, manipulate the triad, and specify some rendering techniques differently for each pane.

For example, you can select **View Pane > Grid View** and set viewing options independently:



For more information, see View Panes.

Current-view indication and navigation-panel minimization in 3D World Editor, Simulink 3D Animation viewer, and vrplay animation viewer

The navigation panel is blue in the active 3D World Editor view pane, Simulink 3D Animation viewer, and `vrplay` animation viewer and is gray in the inactive view panes or viewers.

You can minimize the navigation panel. Select **View > Navigation Panel > Minimized**. The minimized navigation panel appears as an icon in the lower right corner of the viewer. For more information, see Minimize the Navigation Panel.

Compatibility Considerations

Before R2013a, the red X control on the left side of the navigation panel was a toggle button to hide or display the navigation panel. Starting in R2013a, the red X control minimizes the navigation panel.

To hide the navigation panel, select **View > Navigation Panel > None**.

MATLAB Current Folder panel for opening VRML file in 3D World Editor

You can open VRML file (`.wrl`) in the 3D World Editor from the MATLAB Current Folder panel using the context menu.

Option to start 3D World Editor with previous session's layout

You can specify whether the 3D World Editor starts up with the default layout of a virtual world or opens with the layout as it appeared when you exited the virtual world previously. The layout includes settings for the view, viewpoints, navigation, and rendering.

By default, virtual worlds open with the layout that was saved at exit. To have virtual worlds open using the default layout, clear the **Preferences > Simulink 3D Animation > Editor > Preserve Layout per VRML File** preference.

Expansion of subtree in the tree structure pane of 3D World Editor

You can expand subtrees for a 3D World Editor VRML node by using the **Expand Subtree** context menu item.

Multiple 3D World Editor instances

When you open the 3D World Editor using a **File** menu item, if the existing editor window has a file open, then a separate instance of the editor opens. Having multiple instances of the 3D World Editor enables you to work on multiple virtual worlds at the same time and to copy and paste from one virtual world to another.

Before R2013a, if you already had the 3D World Editor open, then when you opened another virtual world, it replaced the initial virtual world.

Blaxxun Contact viewer

Support for the Blaxxun Contact viewer will be removed in a future release. Consider using one of the other VRML viewers described in VRML Viewers.

R2012b

Version: 6.2

New Features

Bug Fixes

VRML syntax highlighting for MATLAB Editor

The MATLAB Editor displays VRML syntax highlighting for `.wrl` files. For details, see VRML Syntax Highlighting in the MATLAB Editor.

Mac OS support for Joystick Input Block and joystick object

You can use the Joystick Input block and the joystick objects (`vrjoystick`) on all supported platforms, including Mac OS platforms.

Multiple nodes selection and parent node creation, facilitating hierarchical scene edits in 3D World Editor

In the 3D World Editor, on the **tree structure** pane, you can select multiple nodes to:

- Delete the nodes.
- Wrap the nodes as children of another node.

Interactive color selection for SFColor and MFColor fields using color palette in 3D World Editor

In the edit panel for `SFColor` and `MFColor` nodes, you can interactively select the color using the palette of colors, as an alternative to specifying the RGB values explicitly.

Use of MATLAB expressions to enter 3D World Editor's numeric field values

When you enter a numeric field value in the 3D World Editor, you can use MATLAB expressions and MATLAB variables. For example, to convert an angle from degrees to radians, enter a MATLAB expression such as `25*pi/180`.

Additional grids for Components library

The Components library includes some additional grids for more control of granularity.

vredit function for opening 3D World Editor

You can use the new `vredit` function to open a virtual world in the 3D World Editor.

MATLAB buttons for editor and animation player

You can open the 3D World Editor or the Simulink 3D Animation animation player by using MATLAB Toolstrip buttons.

From the MATLAB Toolstrip, in the **Apps** tab, click the down arrow on the right side of the tab. Scroll down to the **Simulation Graphics and Reporting** area and click either the **3D World Editor** or the **3D Animation Player** button.

Simulation Stepper support

You can use the Simulink Simulation Stepper with the VR Sink and VR Source blocks.

Note You cannot use the Simulink Simulation Stepper with the VR Tracer block.

R2012a

Version: 6.1

New Features

VR Tracer Block Enhancements

The VR Tracer block supports specification of a projection plane and projection point for better visualization of an object's trajectory. In the VR Tracer block parameters dialog box, you can access these enhancements with the new **Marker Projections** tab.

3D World Editor Enhancements

R2012a includes the following 3D World Editor enhancements:

- Full editing of VRML `PROTO` and `EXTERNPROTO` definitions.
- Faster refreshes.
- You can specify coordinate plane grid templates, which provide orientation information to improve navigation of a virtual world. To specify a grid template, select **File > New From Template > Grid**.

stl2vrml Function Accepts Additional Units

In previous releases, the `stl2vrml` function accepted only meters as the unit for the SimMechanics™ model to be converted to VRML. In R2012a, the SimMechanics model can include additional units, including a mixture of units.

Animation Recording Uses MATLAB VideoWriter Technology

Virtual world AVI animation recording now uses the MATLAB `VideoWriter` technology. The VideoWriter technology supports:

- Creating animation recording files that exceed 2 GB
- Platform-independent Motion JPEG compression

New Example For Global Coordinates

The new “Manipulator Moving a Load with Use of Global Coordinates” example (`vrmanipul_global`) illustrates the use of global coordinates in Simulink 3D Animation models. You can use global coordinates in a model in many ways, including:

- Object tracking and manipulation

- Simple collision detection
- Simulation of haptic effects

R2011b

Version: 6.0

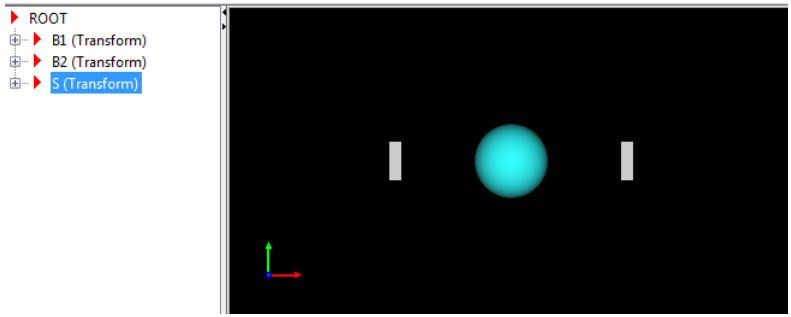
New Features

VR Source Block

The VR Source block provides access to values from virtual world fields that you specify in the Block Parameters dialog box as inputs during simulation. This functionality supports activities such as using the VR Source block to register user interactions with the virtual world. You can pass to the model those values that can affect the simulation of a model.

Triad of Coordinate Axes

To help you visualize changes in the orientation (coordinate axes) of nodes in a virtual world, the 3D World Editor **virtual world display** pane includes a triad of red, green, and blue arrows. These arrows are always parallel with global `x`, `y`, and `z` coordinate axes. The triad displays changes in orientation as you navigate in a virtual world.



Also, you can add a triad to the Simulink 3D Animation Viewer.

For details about using the triad in the 3D World Editor, see Coordinate Axes Triad.

For details about using the triad in the Simulink 3D Animation Viewer, see Display a Coordinate Axes Triad.

Controlling Full-Screen Mode

The Simulink 3D Animation viewer navigation panel now includes a full-screen mode toggle button ( ) to facilitate moving in and out of full-screen display mode.

You can also enable full-screen mode using the new **View > Fullscreen Mode** menu option.

Global Coordinates for vrnode Objects

When you read the positions and rotations of `vrnode` types such as `Transform` and `Viewpoint`, you can now use global coordinates. Global coordinates are very useful, for example, when you are tracking a robot end-effector using tracers that you create independently from the VRML hierarchy of the robot. For details, see the documentation for the `vrnode` function.

3D World Editor Enhancements

Template Files

The 3D World Editor includes template VRML files that you can use as a starting point for creating virtual reality worlds. Some examples of templates are the Earth, roads, sea, and terrains virtual world templates. For details, see Template VRML Files.

Add and Paste Nodes Throughout the Hierarchy

In the **Tree structure** pane, you can add or paste nodes at the beginning or end of the node list, as well as at the end of a node. See Adding Objects and Copying and Pasting a Node for more information.

Copy Nodes Between VRML Files

You can copy nodes between VRML files. For details, see Copying and Pasting Between Virtual Worlds.

Sensing Objects Added to Components Library

The `Components` library now includes sensing objects. Some examples of sensing objects are:

- Push buttons
- Toggle buttons
- 2D setpoint area

For information, see Add VRML Sensors to Virtual Worlds.

Improved VRML Script Editing

The 3D World Editor includes enhancements to make VRML script editing easier. You can:

- Add script interface elements by right-clicking a `Script` node.
- Edit URL code in an edit box.

For more information, see Editing VRML Scripts.

Improved VRML Tree Structure Pane

The **Tree structure** pane displays icons to visually distinguish between node field types (`field`, `eventIn`, `eventOut`, and `exposedField`).

Specify a Default Viewpoint From the Command Line

You can now specify a default viewpoint at the command line, using the `vrview` function with a pound sign (`#`), followed by the name of a viewpoint. For example:

```
vrview('vrlights.wrl#Ocean')
```

New or Enhanced Examples

The new `vrcrane_panel` example uses an operator control panel to set the setpoint. The setpoint is the desired position of the load. The example uses the VR Source block.

The example previously named `vrbuttons` is now `vrdemo_panel`. The `vrdemo_panel` example uses sensing objects that are now available in the 3D World Editor `Components` library. You can place the example control panel in your virtual worlds. You can control a simulation or visualize certain virtual world properties by using signals from virtual world sensors in a simulation model.

Legacy Viewer

The legacy viewer is not recommended. Use the Simulink 3D Animation Viewer or other supported VRML viewer instead.

R2011a

Version: 5.3

New Features

Bug Fixes

Geometric and Background Objects Added to 3D World Editor Library

The 3D World Editor library includes geometric objects, such as a Sphere object. Access the geometric objects from the **Nodes > Insert From > Components Library > Shapes** menu.

The 3D World Editor library also includes background objects, such as a two-tone blue background object. Access the background objects from the **Nodes > Insert From > Components Library > Backgrounds** menu.

Switching Between Bindable Nodes

You can switch between multiple `Bindable` nodes of the same type (`Background`, `Fog`, `NavigationInfo` or `Viewpoint` nodes) in the 3D Work Editor. By default, the first node of multiple instances of a `Bindable` node is active.

For `Background`, `Fog`, and `NavigationInfo` nodes, right-click the node that you want to make active and select the **Bind** menu item.

As you could in releases prior to R2011a, to make an instance of a `Viewpoint` node active, right-click the node that you want to be active and select the **Go to viewpoint** menu item.

3D World Editor Accessible from the Start Menu

You can open the 3D World Editor using the MATLAB **Start** button. Select **Simulink > Simulink 3D Animation > 3D World Editor**.

Preference Options for 3D World Editor Startup Position

The Simulink 3D Animation Preferences pane includes the following two options for specifying the startup position of the 3D World Editor:

- To specify the default location for the 3D World Editor, select **3D World Editor > Position**, and specify the pixel location for the upper-left and the lower-right corners (for example, `[96 120 862 960]`).

- To have the 3D World Editor open in the same location where the editor was last exited, select **3D World Editor > Save position on exit**.

3D World Editor Option to Hide Default Values

To simplify the tree view in the 3D World Editor, you can hide default VRML values. To hide default values, select the **Tree > Hide Default Values** menu item. To return to viewing default values, clear the **Hide Default Values** option.

Node Fields Are Listed Before Node Children Fields in 3D World Editor

Node fields appear at the top of the list of nodes in the tree structure pane of the 3D World Editor.

In the R2010b version of the 3D World Editor, node children appeared before the node's own fields. However, virtual worlds created in R2010b do not require any changes to work in R2011a.

Setting Viewpoint Values in the 3D World Editor Based on Camera Position

You can use the current camera position to interactively specify a viewpoint in the 3D World Editor. Right-click a Viewpoint node in the tree structure pane and select the **Copy values from current camera** option.

1. Navigate to a position in the scene that you want.
2. In the **tree structure** pane, right-click a `Viewpoint` node .
3. Select **Copy values from current camera**.

Faster Texture Handling

Simulink 3D Animation renders textures more quickly than previously. The performance improvement is especially noticeable for 64-bit Windows platforms.

New Examples

R2011a includes the following new examples:

- Vehicle Dynamics Visualization with Live Rear Mirror Image (`vr_octavia_mirror`)

  Illustrates the capability of the VR Sink block to process a video stream on input. The example also creates the impression of a live reflection.
- Vehicle Dynamics Visualization - Simulation of a Double Lane-Change Manoeuvre (`vr_octavia_2cars.mdl`)

  Illustrates multiple-object scenario visualizations.

R2010b

Version: 5.2

New Features

Bug Fixes

Compatibility Considerations

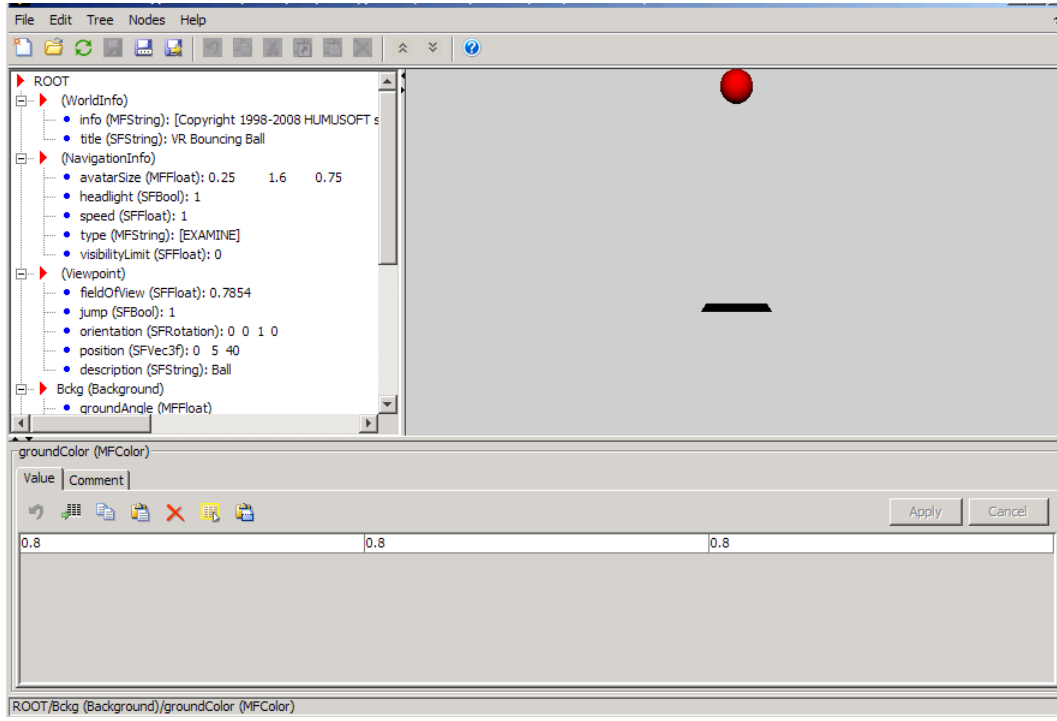
3D World Editor

R2010b introduces the 3D World Editor, which is a built-in VRML editor that works on all platforms supported for the Simulink 3D Animation product.

You can use the 3D World Editor instead of, or in addition to, the V-Realm Builder editor (which also is included with the Simulink 3D Animation product) or with another VRML virtual world editor.

The 3D World Editor includes three panes:

- Tree structure pane, which shows the hierarchy for the virtual world you are editing
- Virtual scene display pane, which reflects your edits
- Edit pane, where you can change values for node items



For details about using the new editor, see Using the 3D World Editor.

Compatibility Considerations

When you install R2010b in a different location from where any previous MATLAB releases are installed, the 3D World Editor is the default editor. If you are upgrading from a previous release, then you need to change your preferences, please read this section.

The **VRML Editor command line** preference in R2010b is now `*BUILTIN`.

Also, in R2010b the **Antialiasing** preference check box is enabled. To access the **Antialiasing** preference,

If you are upgrading from a previous release to R2010b, we recommend that you reset your preferences. You can use one of these approaches:

- Reset all your preferences to match the default preference settings for R2010b.
- Explicitly reset the **VRML Editor command line** and **Antialiasing** preferences.

To reset your preferences to the R2010b default settings, at the MATLAB command line, enter:

```
vrsetpref factory;
```

To set the individual preferences explicitly:

1. In MATLAB, select **File > Preferences > Simulink 3D Animation**
2. In the Simulink 3D Animation Preferences dialog box, set the **VRML Editor command line** preference to `*BUILTIN`.
3. In the Simulink 3D Animation preferences tree view, select the `Figure` node and then open the **Rendering** tab.
4. Clear the **Antialiasing** check box.

stl2vrml Function

You can use the new `stl2vrml` function to convert STL files to VRML format.

Simulink 3D Animation Viewer Supported on Linux

You can now use the Simulink 3D Animation viewer on Linux platforms. You can use MATLAB graphics features with this viewer.

R2010a

Version: 5.1.1

New Features

Bug Fixes

vrnode/getfield Now Returns Field values of SFNode and MFNode Types

The `vrnode/getfield` function now provides access to the `SFNode` and `MFNode` field values. This access improves virtual scene node management. It enables you to access the entire virtual world structure and get the properties of all nodes present in the scene, including unnamed nodes. This capability is most useful for advanced scene manipulations. For example, adding nodes to unnamed nodes or setting properties for these nodes. See VRML Field Data Types in the Simulink 3D Animation User's Guide.

In previous releases, this capability was available only for nodes named using the `DEF` keyword.

VRML Field Data Types

The software now supports the `SFImage` VRML field data type. This support enables dynamic object texturing and playback of video sequences in the virtual scene. See VRML Field Data Types in the Simulink 3D Animation User's Guide.

Camera Bound to Viewpoint Shortcut Change

The Camera Bound to Viewpoint shortcut is now **Ctrl+F10**. In previous releases, this shortcut was **F10**.

Navigation Method Changes

You can now select **None** for the **Navigation > Method** context menu. This option disables mouse navigation in the virtual scene viewer. The **Shift+N** keyboard command provides this option.

Updated vrphysmod

The `vrphysmod` function has been enhanced to automatically link multiple SimMechanics models imported through SimMechanics Link to a single virtual scene that contains these components. This means that if your model contains several VR Sink blocks that refer to the same VRML file, this function attempts to consolidate the animation signals of that virtual scene into one VR Sink block.

VR Sink Block Now Supports Setting Node eventIn Events

The VR Sink block now supports setting node `eventIn` events that do not have any field or `exposeField` associated with them. This improves the ability to control virtual scene, in particular using scripts and interpolators. For example, you can now define an object trajectory using a VRML PositionInterpolator. During simulation, you can define the current position of the object on the trajectory by setting the `set_fraction` `eventIn` event of that interpolator.

vrmorph Example

The `vrmorph` example has been updated to reflect support for Simulink variable-size signals.

Some Dialog Boxes Are No Longer Modal

The Capturing and Recording Parameters and Create new viewpoint dialog boxes are no longer modal dialog boxes. This means that you can access other windows while these dialog boxes are active. For example, you can now change recording parameters or create viewpoints while a simulation is running.

R2009b

Version: 5.1

New Features

Bug Fixes

Compatibility Considerations

Alternate Mode of Virtual Scene Viewer Now Default

The alternate mode of virtual scene viewer introduced in Version 5.0 (R2009a) is now the default viewer for all supported platforms (except for Linux platforms). Using `vrsetpref` to set the `DefaultViewer` property to `'internal'` enables this version of the virtual scene viewer.

To reactivate the legacy (former) viewer, use `vrsetpref` to set the `DefaultViewer` property to `'internalv4'`.

Note If you are running the Simulink 3D Animation software on a Linux platform, the default viewer continues to be the legacy viewer. The legacy viewer does not support MATLAB figures. If you have a Linux platform and want to access the MATLAB figure capability for the viewer, install the hardware-accelerated OpenGL® rendering engine first. Otherwise, the Simulink 3D Animation software cannot properly display the viewer window. You can then activate the new viewer by typing `vrsetpref('DefaultViewer','internalv5')` in the MATLAB Command Window.

Compatibility Considerations

The alternate mode of virtual scene viewer introduced in Version 5.0 (R2009a) is now the default viewer for all supported platforms (except for Linux platforms). At a minimum, this viewer provides the same functionality as the legacy viewer.

Support on the 64-Bit Intel Macintosh Platform

The Simulink 3D Animation software now supports the 64-bit Intel® Macintosh platform.

VRML Files Recognized in MATLAB Current Folder Browser

The MATLAB Current Folder browser now supports `.wrl` files. This support enables you to perform standard MATLAB file operations on the `.wrl` file in the Current Folder browser. For example, double-clicking a `.wrl` file in the Current Folder browser now starts the Simulink 3D Animation viewer for that `.wrl` file. In previous releases, double-clicking a `.wrl` file in the current folder browser started the MATLAB editor for that file.

Changed vr.canvas Units Property Default

The `vr.canvas` method `Units` property default is now `'normalized'`. In the previous release, the default was `'pixels'`. The `vrgetpref` function `DefaultCanvasUnits` property controls the default of the `vr.canvas` `Units` property.

Improved Performance of Orbisnap in Remote Configuration

The performance speed of the Simulink 3D Animation Stand-Alone Viewer (Orbisnap) has been improved.

R2009a

Version: 5.0

New Features

Bug Fixes

Alternate Mode of Virtual Scene Viewer Integrated with MATLAB Figures

Alternate mode of virtual scene viewer integrated with MATLAB figures, enables you to combine the following in the same figure:

- Virtual scenes and MATLAB graphics objects
- Multiple views of a virtual world
- Views of different virtual worlds

This feature also allows for the docking of these figure windows, each of which can contain different worlds, into split screen MATLAB figure windows.

See Viewing Virtual Worlds in the Simulink 3D Animation User's Guide for details on this alternate viewer mode, including how to activate it (use `vrsetpref` to set the `DefaultViewer` property to `'internalv5'`).

vrplay Updated

The `vrplay` function now contains virtual scene and player controls in a single MATLAB figure window.

New Functions

The Simulink 3D Animation product has the following new functions:

- `vrcadcleanup`

  Processes VRML files created by CAD tools for use with Simulink 3D Animation environment. In previous releases, you manually processed these files before using them in the Simulink 3D Animation environment.

- `vrphysmod`

  Automatically adds a Simulink 3D Animation scene to SimMechanics models created from CAD tools. In previous releases, you manually processed these files before using them in the Simulink 3D Animation environment.

New Object

This release provides a new object, `vr.canvas`. This object allows for the placement of a virtual scene view into a MATLAB figure, with full control of its position and ability to be combined with other MATLAB graphics objects into one graphical interface.

Virtual Scenes Now Open Automatically

The associated virtual scene now opens automatically when you open the model. This behavior occurs even if the Simulink 3D Animation block associated with the virtual scene is in a subsystem of the model. In previous releases, the ability to automatically open virtual scenes did not work for VR Sink blocks in subsystems.

Product Files Moved

The Simulink 3D Animation files have moved from *`matlabroot`*`\toolbox\vr` to *`matlabroot`*`\toolbox\sl3d`.

Saving and Restoring the Complete SimState

Use the new SimState feature to save the complete simulation state. Unlike the final states stored in earlier versions of Simulink, the SimState contains the complete simulation state of the model (including block states that are logged). You can then restore the state at a later time and continue simulation from the exact instant at which you stopped the simulation. See Saving and Restoring the Simulation State as the SimState in the Simulink User's Guide.

R2008b

Version: 4.8

New Features

Bug Fixes

New VR Tracer Block

New VR Tracer block enables visualization of trajectories in virtual scenes.

Updated vrnode Method

The `vrnode` method now reflects support for the VRML `USE` reference.

New vrworld Method

The `vrworld/addexternproto` method is new for `vrworld`. This function enables the addition of an `EXTERNPROTO` declaration to the virtual world.

New Example

The `vrtkoff_spacemouse` example is new in this release. It illustrates how to use a space mouse compatible device via the MATLAB interface. This example requires a space mouse or compatible device.

R2008a

Version: 4.7

New Features

Bug Fixes

VR Sink Block Now Supports Matrix Signals as Input

The Virtual Reality Toolbox™ product now supports matrix-type data input for the VR Sink block. Previously, only the MATLAB interface supported matrices (MFxx fields). The Simulink interface supported only vectors (SFxx fields). Now, both MATLAB and Simulink interfaces support matrices (MFxx fields).

New vrworld Object Option

The `vrworld` object has a new option, `new`. This option enables the creation of a new virtual world object, even if another vrworld object associated with the same VRML file already exists.

Enhanced vrplay Shortcut Keys

The `vrplay` function shortcut keys have changed as follows:

- The **Page Up** and **Page Down** keys have the same functionality as the **R** (rewind) and **F** (forward) shortcut keys, respectively. These keys perform rewind and forward operations on the animation file you are playing.
- The **p** shortcut key toggles between play and pause behavior.

New Examples

The following examples are new in this release:

- `vrmorph`
- `vr_octavia_video`
- `vrtkoff_trace`

R2007b

Version: 4.6

New Features

Bug Fixes

New VR To Video block

New VR To Video block allows RGB video stream output for further 2D video processing.

Joystick and Space Mouse Input Functionality Added to MATLAB Toolbox Interface

New `vrjoystick` and `vrspacemouse` objects can interface with joysticks and space mouse input devices.

Joystick and Space Mouse Input Support Added for Linux

Joystick and space mouse input support is added for both 32-bit and 64-bit Linux operating systems.

New Axes Transformation and Math Functions

New axes transformation and math functions are added to the MATLAB toolbox interface. These are:

- `vrdir2ori` to convert viewpoint direction to orientation.
- `vrori2dir` to convert viewpoint orientation to direction.
- `vrrotmat2vec` to convert rotation from matrix to axis-angle representation.
- `vrrotvec` to calculate a rotation between two vectors.
- `vrrotvec2mat` to convert rotation from axis-angle to matrix representation.

Magellan Space Mouse Block Renamed to Space Mouse Input

The Magellan Space Mouse block is renamed to Space Mouse Input.

R2007a

Version: 4.5

New Features

Bug Fixes

Full-Screen Mode in Virtual Reality Toolbox Viewer and Orbisnap Stand-Alone Viewer

Fullscreen mode, in which the menu bar and toolbar are switched off, is implemented in the Virtual Reality Toolbox viewer and the Orbisnap stand-alone viewer.

R2006b

Version: 4.4

New Features

Bug Fixes

Compatibility Considerations

VR Text Output Block Added

The VR Text Output block allows Simulink signal values to be displayed as text in VRML scenes.

Frames Per Second Added to AVI File Playback Options

It is now possible to specify the frames per second displayed during the playback of AVI files.

New 3Dconnexion Devices

SpacePilot and SpaceBall 5000 have been added to the list of devices supported by `Magellan Space Mouse` block.

Joystick Input Block

The Joystick Input block joystick yaw and throttle controls have switched behaviors.

Compatibility Considerations

If you have models that contain this block, and you want to use that model in a Virtual Reality Toolbox version later than 4,3, you must reorder the axes. You can do so by placing the Simulink Selector block after the Joystick Input block in the model.

R2006a

Version: 4.3

New Features

Bug Fixes

New Utilities Library

A new library `Utilities` is added to the Virtual Reality Toolbox block library `vrlib`. This library contains the blocks `Rotation Matrix to VRML Rotation`, `Viewpoint Direction to VRML Orientation`, `Rotation between 2 Vectors`, `Normalize Vector`, and `Cross Product`. See Block Reference for details.

VR Animation Playback Controls

A new `vrplay` GUI is added, which allows playback control of VRML animation files

R14SP3

Version: 4.2.1

Bug Fixes

R14SP2+

Version: 4.2

New Features

Virtual Reality Toolbox Support of MATLAB Compiler Product

The Virtual Reality Toolbox product now supports the MATLAB Compiler™ product. With this capability, you can use the MATLAB Compiler software to take MATLAB files as input and generate redistributable, stand-alone applications that include Virtual Reality Toolbox functionality, including the Virtual Reality Toolbox viewer. The Virtual Reality Toolbox product does not support the MATLAB Compiler product for the HP-UX platform.

Virtual Reality Toolbox Viewer

The following changes were made to the viewer:

- The Virtual Reality Toolbox viewer **Recording** menu has changed to support frame captures in addition to animation recording.
- The Virtual Reality Toolbox toolbar now has a camera icon that you can click to capture a frame of the current virtual scene.

Virtual Reality Toolbox Preferences Dialog

The Virtual Reality Toolbox preferences dialog, available through the MATLAB desktop **File -> Preferences** dialog, has been updated as follows:

- The **Virtual Reality Figure** pane is now organized by tabs.
- The Virtual Reality Toolbox preferences dialog now provides frame capture preferences. These are available through the **Frame Capture** tab of the **Virtual Reality Figure** pane.

vrsetpref and vrgetpref Changes

The `vrfigure` `set` and `get` functions now include the following properties to support frame capturing:

- `CaptureFileFormat`
- `CaptureFileName`

Space Mouse Block Enhancement

The Magellan Space Mouse block now allows you to specify upper and lower position coordinate limits for the mouse. The following parameters were added:

- **Limit position** — Determines whether you can limit the upper and lower positions of the mouse.
- **Lower position limit** — Position coordinates for the lower limit of the mouse.
- **Upper position limit** — Position coordinates for the upper limit of the mouse.

R14SP2

Version: 4.1

New Features

Virtual Reality Toolbox Stand-Alone Viewer

The Virtual Reality Toolbox product now includes Orbisnap. This is a free, optional, and multiplatform stand-alone VRML viewer that does not require you to have either MATLAB or Virtual Reality Toolbox products. You can use Orbisnap to

- View prerecorded WRL animation files
- Remotely view, from a client machine, a virtual world loaded in a current session of the Virtual Reality Toolbox software
- View and navigate, but not simulate, a VRML world or scene

Virtual Reality Toolbox Viewer

The Virtual Reality Toolbox viewer now allows you to interactively set the `vrfigure` maximum texture size through the **Rendering** menu.