

Three Model Viewer Version 1.2

User Guide

**Copyright © 2016 PTC Inc. and/or Its Subsidiary Companies. All Rights Reserved.**

User and training guides and related documentation from PTC Inc. and its subsidiary companies (collectively “PTC”) are subject to the copyright laws of the United States and other countries and are provided under a license agreement that restricts copying, disclosure, and use of such documentation. PTC hereby grants to the licensed software user the right to make copies in printed form of this documentation if provided on software media, but only for internal/personal use and in accordance with the license agreement under which the applicable software is licensed. Any copy made shall include the PTC copyright notice and any other proprietary notice provided by PTC. Training materials may not be copied without the express written consent of PTC. This documentation may not be disclosed, transferred, modified, or reduced to any form, including electronic media, or transmitted or made publicly available by any means without the prior written consent of PTC and no authorization is granted to make copies for such purposes.

Information described herein is furnished for general information only, is subject to change without notice, and should not be construed as a warranty or commitment by PTC. PTC assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

The software described in this document is provided under written license agreement, contains valuable trade secrets and proprietary information, and is protected by the copyright laws of the United States and other countries. It may not be copied or distributed in any form or medium, disclosed to third parties, or used in any manner not provided for in the software licenses agreement except with written prior approval from PTC.

UNAUTHORIZED USE OF SOFTWARE OR ITS DOCUMENTATION CAN RESULT IN CIVIL DAMAGES AND CRIMINAL PROSECUTION. PTC regards software piracy as the crime it is, and we view offenders accordingly. We do not tolerate the piracy of PTC software products, and we pursue (both civilly and criminally) those who do so using all legal means available, including public and private surveillance resources. As part of these efforts, PTC uses data monitoring and scouring technologies to obtain and transmit data on users of illegal copies of our software. This data collection is not performed on users of legally licensed software from PTC and its authorized distributors. If you are using an illegal copy of our software and do not consent to the collection and transmission of such data (including to the United States), cease using the illegal version, and contact PTC to obtain a legally licensed copy.

**Important Copyright, Trademark, Patent, and Licensing Information:** See the About Box, or copyright notice, of your PTC software.

**UNITED STATES GOVERNMENT RESTRICTED RIGHTS LEGEND**

This document and the software described herein are Commercial Computer Documentation and Software, pursuant to FAR 12.212(a)-(b) (OCT’95) or DFARS 227.7202-1(a) and 227.7202-3(a) (JUN’95), and are provided to the US Government under a limited commercial license only. For procurements predating the above clauses, use, duplication, or disclosure by the Government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software Clause at DFARS 252.227-7013 (OCT’88) or Commercial Computer Software-Restricted Rights at FAR 52.227-19(c)(1)-(2) (JUN’87), as applicable. 01012015

**PTC Inc., 140 Kendrick Street, Needham, MA 02494 USA**



[Software Change Log 2](#_Toc460343402)

[Introduction and Installation 2](#_Toc460343403)

[About the EXTENSION 2](#_Toc460343404)

[Installing the THREE MODEL VIEWER Extension 2](#_Toc460343405)

[Configuration and Usage 4](#_Toc460343406)

[Configuration 4](#_Toc460343407)

[Usage 5](#_Toc460343408)

[Compatibility 5](#_Toc460343409)

# Software Change Log

|  |  |  |
| --- | --- | --- |
| Version | Release Date | Changes |
| 1.0 | 1/1/2016 | Initial Release |
| 1.1 | 09/26/2016 | Added selection capabilities |
| 1.2 | 12/02/2016 | Updated Three.js to r82  Added configurable lighting  Allow bindings for model rotation  Automatically load mtl for obj files when available.  Configurable texture paths  Fixed selection bugs. |
| 1.3 | 10/01/2017 | Updated to Three.js to r83  Added an experimental sceneTree property  The rotation now happens from the center of the object  Added rotation based on quaternions  Added option to set the Y model offset  Added loading indicator  Added event for loading successful and loading error |

# Introduction and Installation

Extensibility is a core aspect of the architecture and design of ThingWorx. Partners, third parties, and general ThingWorx users can easily add new functionality into the system, seamlessly. Extensions can be in the form of Service (function/method) Libraries, Connector Templates, Widgets, and more.

This document provides installation and usage instructions for the THREE MODEL VIEWER Extension.

## About the EXTENSION

THREE MODEL VIEWER allows you to visualize 3D models in a lot of different formats.

The THREE MODEL VIEWER Extension for ThingWorx also provides advanced capabilities of using scenes created in other programs. This extension utilizes the Three.js libraries and modern web technologies like WebGL. For more information, visit <http://threejs.org/>

The THREE MODEL VIEWER Extension allows you to use the Three Model Viewer widget. This responsive widget allows visualization and interaction of the most commonly used 3D data formats.

## Installing the THREE MODEL VIEWER Extension

|  |  |
| --- | --- |
| 1. From a web browser, launch ThingWorx. 2. Log into ThingWorx as an administrator. |  |
| 1. Go to **Import/Export > Import**. |  |
| 1. Click Choose File and select [ThreeModelViewer\_ExtensionPackage].zip 2. Click **Import**.  Note: If an **Import Successful** message does not display, contact your ThingWorx System Administrator. | Note: |
| 1. Click **Yes** to refresh Composer after importing the final extension. |  |
| 1. Confirm that the Extension has been imported properly. Check the Application Log for potential problems. |  |
|  |  |

# Configuration and Usage

The following model formats are supported:

|  |  |
| --- | --- |
| **Format** | **Observations** |
| **dae** | Loaded as a 3D model. Also loads textures |
| **3mf** | Loaded as a 3D model |
| **amf** | Loaded as a 3D model |
| **awd** | Loaded as a 3D scene |
| **babylon** | Loaded as a 3D scene |
| **ctm** | Loaded as a 3D model |
| **fbx** | Loaded as a 3D model |
| **gltf** | Loaded as a 3D model |
| **kmz** | Loaded as a 3D model |
| **playcanvas** | Loaded as a 3D model |
| **json** | Three.js native file format. Depending on the file, loaded as scene or model |
| **md2** | Loaded as a 3D model |
| **obj** | Loaded as a 3D model. Materials are not supported |
| **ply** | Loaded as a 3D model |
| **stl** | Loaded as a 3D model |
| **vtk** | Loaded as a 3D scene |
| **wrl** | Loaded as a 3D model |
| **assimpjson** | Loaded as a 3D model |
| **assimp** | Loaded as a 3D model |
| **sea** | Loaded as a 3D scene. Also includes animations |

## 

## 

## Configuration

The widget offers the following **properties** that can be changed:

* **ModelUrl:** the URL where the 3D data can be found.
* **ModelType:** the type of the model to load. If set on the default, “Auto-Detect”, the ModelURL must contain the file name and the extension that you are trying to load. This extension is used to determine what kind of file it is. For example, a link like <http://example.com/avatar.dae> is explicit that this is a Collada file. If the ModelType is set, then only files of that type can be loaded.
* **CameraControls:** Enable orbiting, dollying (zooming), and panning.
* **CameraAutoRotate:** Enable auto-rotating of the camera around the target.
* **DrawAxisHelpers:** Draw Axis Helpers to visualize the the 3 axes in a simple way. The X axis is red. The Y axis is green. The Z axis is blue.
* **DrawGridHelpers:** Draw Grid Helpers on the ground.
* **ResetSceneOnModelChange:** Reset the scene whenever the model changes (delete the old models).
* **AddLightsToSceneFiles:** For the files loaded as scene files, add the default lights (ambient light and directional lights in each corner of the scene)**.**
* **BackgroundStyle:** The background of the widget. Opacity is supported.
* **EnableSelection:** The user can select subcomponents of the 3D model.
* **SelectedItem:** The id currently selected item in the scene. Depending on the input file type, this can be an 3D object or a primitive geometry.
* **SelectedItemName**: The name currently selected item in the scene. May not be available for all the models.
* **TexturePath:** If textures are requested, what is the path to get them. If null, defaults to the folder where the scene is stored.
* **Rotation X, Rotation Y, Rotation Z:** If loading a model, specify its rotation. We consider the center bounding box as the center of rotation.
* **ShowStats:** Show render statistics (FPS, memory, CPU).
* **LightIntensity:** The intensity of the light. Use a value between 0 and 1.
* **SceneTree:** EXPERIMENTAL FEATURE: A tree of all the elements in the scene. Does not support selection. For big models the generated tree can be very big!
* **Quaternion:** Rotation Quaternion for the model. Represented as comma sepparated X,Y,Z,W. Disabled by default.
* **ModelYOffset:** Positions the model on a Y offset vs the grid.
* **EnableQuaternionRotation:** Use Quaternions for rotation rather than eulers. This sets the rotation of the model using the Quaternion property.

The widget also has the following events:

* **LoadDone**: Called when the loading of a model finishes. This can be called multiple times for a single file, if the file also refers to other external files.
* **LoadError**: Called if the loading of a file fails.

## Usage

A general use-case for this widgets is when you want to display 3D models coming from a URL inside a ThingWorx mashup. This widget allows to bind the URL to the **ModelUrl** property and view the model at runtime. If the **ModelUrl** property changes, then the new model is loaded in.

At runtime, the user can also interact with the model by controlling the camera. It has set of controls performs orbiting, dollying (zooming), and panning.

The user can also enable selection, so when a part of the model is clicked, information about that part can be displayed.

The following commands are available:

* **Orbit** - left mouse / touch: one finger move
* **Zoom** - middle mouse, or mousewheel / touch: two finger spread or squish
* **Pan** - right mouse, or arrow keys / touch: three finger swipe
* **Click** – select the highlighted part.

The widget is also compatible with mobile devices, and supports touch events.

# Compatibility

This extension was tested for compatibility with the following ThingWorx Platform version(s) and Operating System(s). Please note that some model formats are not compatible with certain browsers.

|  |  |
| --- | --- |
| ThingWorx Platform Version | ThingWorx 7.1.0 |
| OS | Windows 7, Service Pack 1, iOS 10, Android 5, OSX 10.12 |
| Browser | Chrome 53, Firefox 48, |