

Frequently Asked Questions

The frequently asked questions about OpenSim are broken up into several categories:

- [Installation and Compatibility](#)
- [Visualization and Graphics](#)
- [Importing and Exporting Data](#)
- [Muscle Models](#)
- [Plotting](#)
- [Scaling and Customizing Models](#)
- [Generating Simulations](#)
- [Applying OpenSim](#)
- [Scripting](#)
- [OpenSim vs SIMM](#)

Installation and Compatibility

Q. Which OpenSim installer should I use?

A. As of release 3.3, we provide two Windows only installers (32bit and 64 bit). In general you should use the 64-bit installer on a 64-bit machine and a 32-bit on a 32-bit machine, with some exceptions (below). You can find what version of Windows you're running by following the Windows documentation found [here](#).

Exceptions: You may need to run 32-bit version on 64-bit machine if you're using Matlab 32-bit, or if you want to use a feature (e.g. recording movies) that doesn't work on 64-bit machines out-of-the-box.

Windows 10: OpenSim 3.3 was released before Windows 10 so we were unable to test the variety of Windows configurations (Home, Professional, etc.). We are currently testing on a Windows 10 Pro, 64-bit installation and have had no problems using OpenSim 3.3, although some users reported failing to open the view/visualization window. We are still investigating. If you have issues using Windows 10, please send a message to opensim@stanford.edu and include a description of your machine configuration and the errors you are experiencing.

Q. When trying to start OpenSim, I get a 'JVM creation failed' error.

A. 32-bit OpenSim on a 64-bit machine may encounter an issue where the GUI's Java Virtual machine (JVM) incorrectly detects the machine as 64-bit and fails to start because it requests a large chunk of memory. To solve this, close a few other applications to free up some memory (Google Chrome is particularly resource heavy) and restart OpenSim. Once OpenSim has started you can re-open any closed applications. A forum thread on this issue can be found [here](#).

Q. Does OpenSim work on a Mac?

A. The name OpenSim refers to two distinct entities, the OpenSim libraries (API), which are a C++ library of classes used for modeling and simulation of biological structures, and the OpenSim application (GUI) which builds on the OpenSim API. The only distribution of the GUI is on Windows (32Bit and 64Bit as of 3.1). Mac users who want to use the GUI, have successfully run OpenSim using Bootcamp, Virtual Machines, or other environments that simulate Windows on a Mac. Please see [Running OpenSim on Mac OS X or Linux using a Windows Virtual Machine](#) for more information.

The API (headers, libraries and Doxygen documentation) is available for Windows (32Bit and 64Bit as of 3.1), Mac (32Bit and 64Bit), & Linux (32Bit and 64Bit) and is available on the [OpenSim Downloads](#) page. To use the API, users will need to write a main program or a plugin or use the various tools provided as command line executables to perform modeling/simulation.

Mac and Linux users will need to add the installation-folder/bin to their environment (LD_LIBRARY_PATH on linux, DYLD_LIBRARY_PATH on mac) and can use the command line utilities or build main programs using the API following instructions similar to those given in the developer's guide for Windows.

If you're interested in helping to build a GUI distribution on Mac, Linux or any other platform, please contact the development team at opensim@stanford.edu.

Q. Why do I see a big red ball when I launch OpenSim?

A. Please see the info on this user forum post: https://simtk.org/forum/message.php?msg_id=5418

Q. I installed the application, but when I launch it I get "Required library failed to load. Check that the dynamic library osimJavaJNI is in your PATH". What could be wrong?

A. You likely picked not to add OpenSim to your PATH while doing installation. If this is the case you can fix it by either reinstalling or adding the directory where you installed OpenSim to your PATH Environment variable. To check if OpenSim is already in the PATH, open a command prompt shell on windows and type `C:\>echo %PATH%` You should see the installation-directory/bin in the list of directories.

Q. Do you need Matlab to run OpenSim?

A. No. You only need Matlab to use the Matlab scripting. If you don't have Matlab, you can still use the application or GUI, the GUI or python scripting shell, and the API.

Q: Do I have to uninstall the previous version of OpenSim before installing a new version?

A. No. You can keep old installations of OpenSim to compare your results.

Q: Where can I download previous releases of OpenSim?

A. You can access all previous releases through the [OpenSim Downloads Page](#). Directly under the current version is the drop down menu 'List all previous releases'.

Q: Can I run OpenSim from a USB drive?

A. If you would like to run OpenSim on a Windows machine, but you don't have install privileges, you can try running OpenSim from a USB drive. This may be useful for instructors of biomechanics courses, for example, allowing students to run OpenSim on library computers. To set up the drive, insert the USB drive and install OpenSim, selecting a path the USB drive as your install directory. Note that you must do this on a Windows machine. After installing you can then run OpenSim from another Windows machine. You will likely need to update the Geometry folder under Edit -> Preferences, to the appropriate path on the new machine.

Visualization and Graphics

Q. Why can't I launch the GUI's visualizer in Windows?

A: The OpenSim 3.3 GUI has been unable to launch the 3D visualizer for some Windows users. If you are experiencing this problem, it is likely that a few core Microsoft libraries are missing from your Windows 10 installation. To fix this, you must install these missing libraries using the correct Microsoft VC Redistributable Installer. Download the Visual C++ 2010 Redistributable installer for [64-bit](#) or [32-bit](#) machines, close OpenSim GUI if it's open, run the installer, and reopen the OpenSim GUI.

Update: The latest [OpenSim Installer](#) (OpenSim-3.3.0-win64VC13P.exe) has been updated to include the Visual C++ Redistributable installer.

Q. I imported a model from SIMM but the bones don't show up what's wrong?

A. You likely are missing the bone files that go with your model, try reimporting with the bones located in a directory under your .jnt file.

Q. How do I tell OpenSim where to locate the bone/geometry files?

A. Open Edit/Preferences and edit the entry for "Geometry Path" which is a semicolon separated list of directories and add the desired path. See [User Preferences](#) for more information.

Q. What are .vtp files?

A. These are the geometry files corresponding to the bones and other objects used by your model. The ".vtp" format is a format used by VTK which is the component used by OpenSim for visualization.

Q. OpenSim loads many models in the same graphics window, how do I know which one I'm working on now?

A. Although many models are loaded, only a single model can be marked as "Current". This is indicated by the dropdown in the application toolbar and also in the navigator.

Q. I can't pick some objects in the Graphics window, is this a bug?

A. Picking is restricted to "Current model" (by default non-current models are dimmed out). If you want to pick on a non-current model you'll have to make it current first.

Q. I can't pick some Muscle Points in the Graphics window why?

A. Moving Muscle Points are special Muscle Points that have Splines describing their trajectories so they can't be picked and dragged from the GUI.

Importing and Exporting Data

Q. Would it be possible to open a C-3D file directly in OpenSim?

A. This is a popular feature request, and we are working on getting it implemented for a future release of OpenSim. For utilities that other users have developed, see the [OpenSim Utilities](#) library.

Q. Can one extract the muscle force magnitudes, points of origin and direction of action?

A. While this is not currently a pre-packaged analysis in OpenSim, it is definitely possible to extract this information with a plug-in or a custom API main program. See the [OpenSim Utilities](#) library for tools other researchers have been working on.

Q: Is it possible to get position over time data from a model? For example can I track a point on the shank of the leg as the model is moving.?

A. Yes! If you have a motion or simulation you can use the Analysis tool to run a Kinematics or PointKinematics analysis. See the [Analyses](#) chapter for more information.

Muscle Models

Q. What muscle models are included in OpenSim and what do you have to develop yourself? Are there some generic muscle models available I can use to get going?

A. There are many muscle models implemented in OpenSim.

One of the most commonly used muscles is the Thelen2003Muscle, based on the model described in Thelen, 2003.

Thelen DG. Adjustment of muscle mechanics model parameters to simulate dynamic contractions in older adults. J Biomech Eng. 2003; 125(1):70-7.

We've also added some new muscle models. Watch [Matt Millard's webinar](#) to find out more.

Q. Do OpenSim muscle models account for the delay within activation dynamics in your model? If yes, how?

A. Yes, the muscle models in OpenSim model the electro-mechanical delay between muscle excitation (or the electrical neural signal sent to the motor-neuron junction) and activation (i.e., the level of force production). For more details on the theory behind our muscle models, check out this great review paper by Felix Zajac: Zajac FE. Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control. Crit Rev Biomed Eng. 1989, 17(4):359-411.

Q. Is there any way that you can incorporate stretch activation of muscle to try and get the model to predict the activation of muscle before foot strike?

A. Sure! You could develop a custom controller in the OpenSim API that is based upon the stretch of muscle or tendons in the model and map that back to predicted activations.

Q. Is the orientation (i.e., pennation angle) or constitution of fibers (e.g., slow vs. fast) taken into account in the muscle models?

A. Yes, we do model the orientation, or pennation angle, of muscle fibers, but no, we do not currently model slow versus fast twitch fibers. For more details on the theory behind these muscle models, check out this great review paper by Felix Zajac: Zajac FE. Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control. Crit Rev Biomed Eng. 1989;17(4):359-411.

Plotting

Q. How can I import data into OpenSim Plotter?

A. Under Y-Quantity, select "Load file..." and then you can read data files in the storage or motion format.

Q. Can I export data from a plot into a ASCII file for processing by Matlab or use in Excel?

A. Under the "curves list", right click on the top level node and select "Export Data...". OpenSim will prompt you for a file name to save the data into. This file can be later re-imported into OpenSim or used by other applications.

Scaling and Customizing Models

Q. What anthropometric database is used for model scaling?

A. OpenSim does not use a database for scaling. The model is scaled based on measurements of experimental markers. See [How Scaling Works](#) for more information.

Q. When you are scaling a generic model to match a particular subject, how do you account for the difference in bone length, like, if someone has a proportionally shorter tibia?

A. Scaling in OpenSim scales each body segment independently based on measurements of markers placed on that segment. See the section on [Scaling](#) for more details.

Q. Why have some studies with OpenSim scaled the model's muscle strength? Does this reflect the model's development from cadaver data?

A. Yes, the optimal force or max strength of the muscles in the model used in many studies are based on Delp, 1990 (see reference below), which determined muscle parameters from cadavers. While the strength of the model was not increased to study running (Hamner et al., 2010), other researchers have noticed that the "strength" (i.e., optimal force) of the muscles has to be increased for movements with higher forces and accelerations, like running and cutting, or for pathological gaits, such as crouch gait. Delp SL et al., An interactive graphics-based model of the lower extremity to study orthopedic surgical procedures. IEEE Trans Biomed Eng. 1990; 37(8):757-67.

Q. Are muscle forces or strength scaled based on the subject's weight (compared to the generic model weight)?

A. No, the scale tool in OpenSim does not automatically scale muscle forces based on the geometric based scaling (i.e., scaling based on distance measurements between markers) or the mass scaling. However, you can scale muscle forces in the model based manually if you have a criterion for doing this. See the [Scripting](#) chapter for info about how to do this with a script.

Q. How can you incorporate BIODEX data into your simulations to scale models?

A. BIODEX data can be used to determine the strength, or torque generating capacity of your subject. Researchers have used this information to modify the optimal muscle forces used in their model. For an example, check out the paper by Anderson & Pandy from 2001. They explain, "Values of peak isometric muscle force and tendon slack length were adjusted so that the maximum isometric torque angle curves for each joint in the model matched average torque-angle curves measured for the five subjects." Anderson FC, Pandy MG. Dynamic optimization of human walking. J Biomech Eng. 2001;123(5):381-90.

Generating Simulations

Q. What are typical settings that are modified during the RRA?

A. RRA is an optimization algorithm whose goal is to minimize the non-physical "residual" forces acting on a dynamic simulation. The settings modified for RRA are the inputs to the optimization algorithm, which include the excitation limits (i.e., min/max controls) for the residual actuators along with the tracking weights for each of the coordinates (i.e., joint angles). See [Getting Started with RRA](#) for more information.

Q. Can you briefly describe available ways to compute muscle forces in OpenSim besides CMC?

A. Yes, OpenSim has another tool called [Static Optimization](#), which has been used by numerous researchers to estimate muscle forces. Check out this paper describing static optimization: Anderson FC, Pandy MG. Static and dynamic optimization solutions for gait are practically equivalent. J Biomech. 2001;34(2):153-61.

Q. Does CMC take synergistic muscles into account?

A. No, we do not explicitly model synergistic muscle groups. Computed muscle control treats each muscle as an independently controlled actuator. However, our results show that muscles are still activated in the observed synergistic groups.

Applying OpenSim

Q. How can I find a good model for my research study?

See our [Musculoskeletal Models](#) page for a list of OpenSim models that we've developed at Stanford, as well as a link to the library of models from our team and other researchers on Simtk.org. Please also see our page on [Evaluating a Musculoskeletal Model](#). It has a list of questions to review when deciding what model to use in your research.

Q. Has anyone used Opensim to simulate pathological gait, like stroke patients or OA patients? What is the difficulty of simulating pathological gait?

A. Yes, students and researchers in the Neuromuscular Biomechanics Lab at Stanford have created muscle-actuated simulations of individuals with cerebral palsy with a crouch gait. Check out a recent study led by Kat Steele. Steele, K.M., Seth, A., Hicks, J.L., Schwartz, M.S., Delp, S.L. Muscle contributions to support and progression during single-limb stance in crouch gait. Journal of Biomechanics, vol 43, pp 2099-2105 and other papers on the [Neuromuscular Biomechanics Lab Page](#). See the [OpenSim Website](#) for more research applications.

Q. I am new to these kinds of simulations and would like to make a new model of a simple experimental set up (in this case an inverted pendulum). What software could I use to generate this for dynamic analysis?

A. For dynamic simulations of rigid body systems, like a double pendulum, check out Simbody(TM), which is the rigid-body dynamics engine used by OpenSim. It is open-source and freely available with a well-tested and stable API. The project page is <https://simtk.org/home/simbody>.

Q: How useful do you feel Opensim currently and in the future will be at smoothly importing, calculating, and reporting clinical gait data?

A. Importing data can sometimes be time consuming and require troubleshooting. The most difficult part is setting up a custom workflow that fits our lab and needs. See [Preparing Your Data](#) for more information about this process. We've added [Scripting](#) to make it easier to customize your workflows. Other users have also contributed their tools in the [OpenSim Utilities](#) library. We also plan to continue to improve our usability going forward, as one of our top priorities.

Q: Can you simulate contact forces (e.g. between the foot and a platform) in OpenSim?

Yes. You can model Hunt Crossley Contact Forces and add contact geometry. See the [OpenSim Doxygen](#) for more information. Check out the XML Browser in the GUI (under Help) to find the xml tags to add the force to your model.

Scripting

Q. Does OpenSim support python scripting? (or only MATLAB?)

A. Yes, OpenSim version 3.2 and later supports scripting in Python, Jython (an implementation of Python in Java), and MATLAB. In the OpenSim application, the scripting shell is a Jython shell. All scripting capabilities use a technique (called wrapping) to invoke the OpenSim C++ API from Python, Java, or MATLAB.

Q. Is there full documentation on how the API is exposed to Scripting?

A. You should be able to leverage the extensive [Doxygen documentation](#) of the OpenSim API to use it in Scripts. You'll also find the examples included with the distribution as well as the help page on Scripting ([Common Scripting Commands](#)) helpful.

Q. Is the MATLAB interface only for 32 bit MATLAB or will it work for 64 bit as well?

A. Starting with OpenSim 3.1, we provide both 32-bit and 64-bit versions of the software. You can use OpenSim with 64-bit Matlab by downloading and installing the 64-bit version of the OpenSim GUI. This assumes you are using Windows. Read more in [Scripting with Matlab](#).

Q. Should OpenSim be running when doing scripting from another language like MATLAB/python?

A. No, the OpenSim application and the scripting interface both utilize the same libraries, but are unrelated otherwise. While scripting in the GUI, you have access to all the API and some GUI functionality (e.g. control of view, plotter, loading models, motions, etc.). While in MATLAB you have access to the API and you can use the MATLAB tools for plotting and the API Visualizer. You can also use the scripting interface from a standalone Python or Jython shell but you have access to neither the GUI functionality nor MATLAB's. See [Common Scripting Commands](#) for more information.

Q. Can OpenSim scripting control all of the GUI's functionality?

A. You can invoke almost any menu command and many other functions (e.g. Plotting) from the GUI scripting layer, but not all dialogs/tools/options are exposed. Please consult the page [Common Scripting Commands#GUIScriptingShell](#) for specific examples.

Q. Do you provide a debug mode when scripting?

A. You can turn on more verbose messages so you get more feedback while running scripts by calling `modeling.setDebugLevel(3)`, but for syntax errors you usually get the error message displayed in the scripting shell directly.

Q. How should I do batch processing for the OpenSim tools (e.g. IK, ID, CMC) with a script?

In general, we strongly discourage doing raw parsing of XML files in a Matlab script. This approach is not robust and does not take advantage of the OpenSim API in Matlab. Instead you should instantiate tools/objects from setup files and let the code do the parsing internally. The XML format changes with different versions of the software and the version number of the documents written by newer versions of the software may even be different from the one you read due to backward compatibility built into the code and assumptions about default values, etc. You'll keep changing your code endlessly rather than rely on a robust API that makes calls to set/get values (rather than parse XML files). In essence the assumptions about the contents of the XML files and the XML file header are all built into the code that is exercised by the API but ignored by any XML parsing code. Violating these assumptions will break the functionality downstream.

An example using parsing (not recommended for OpenSim 3.0):

```

% get initial and inal time from first and last entry in time column
initial_time = trcData(1,1);
final_time=trcData(end,1);

xmlDoc.getElementsByTagName('IKTool').item(0).getAttributes.item(0).setValue(name);
xmlDoc.getElementsByTagName('marker_file').item(0).getFirstChild.setNodeValue([trialsForIKPath markerFile{1}]);
xmlDoc.getElementsByTagName('output_motion_file').item(0).getFirstChild.setNodeValue([results_folder '\ ' name
'_ik.mot']);
xmlDoc.getElementsByTagName('time_range').item(0).getFirstChild.setNodeValue([num2str(initial_time) ' ' num2str
(final_time)]);

```

An example of calling the API (recommended):

```

% Get trc data to determine time range
markerData = MarkerData(markerDataFileName);

% Get initial and intial time
initial_time = markerData.getStartFrameTime();
final_time = markerData.getLastFrameTime();

% Setup the ikTool for this trial
ikTool.setName(name);
ikTool.setMarkerDataFileName(fullpath);
ikTool.setStartTime(initial_time);
ikTool.setEndTime(final_time);

```

The code snippet above is take from the example "setupAndRunIKBatchExample.m".

OpenSim vs SIMM

Q. What is the relationship between SIMM and OpenSim

A. Please [review this PDF](#) (updated March 2010)