



# modo 401

## Software Development Kit (SDK)

June 2009

**Version [Build]:** 401 [31381]

**Date published:** June 18, 2009

**Availability:** Open source download provided to permanent licensees

### About the SDK

The modo 401 SDK provides an application programming interface (API) and source code samples that enable developers to read and write modo 401 scene, image, and movie files, for compatibility with other applications, as well as plug-in scripting support, with an example Lua script interpreter.

The SDK is based on COM, with C++ wrappers for many of the published APIs, and is available for both Mac OS X and Windows 32-bit and 64-bit versions of modo 401.

### What is included

- Header files and source code to build the example plug-ins and the C++ wrappers.
- Sample code and detailed documentation for LXO file importers and exporters.
- Scene loaders and savers for COLLADA Digital Asset Exchange (.DAE), Autodesk FBX (.FBX), Wavefront Object (.OBJ), and VideoScape (.GEO) files
- Scene savers for HPGL Plotter (.plt) files
- Image loaders and savers for Portable Network Graphics (.PNG) and simple raw (.RAW) files
- Movie savers for Apple QuickTime (.MOV) and Windows Media (.WMV) files.
- A script interpreter for the Lua programming language.

The COLLADA, Wavefront Object, and PNG plug-ins all provide sample UI forms that appear in the Preferences window after they are installed.

The SDK provides projects for Visual Studio 2008 on Windows, and for XCode 3 on Mac OS X.

Many sample plug-ins are provided in C++, with liberal open source licensing that allows developers to use the code in their own applications and products.

### What you can do with the SDK

You can create loaders or savers for scenes or images, and savers for movies. You can also create a custom script interpreter for Lua or other programming languages.

Loaders implement a recognition method that scans the file to determine if it's of a known type, and a load method which allows the loader to import data from the file into an object (scene or image) provided by modo.

Savers implement a verify method which can display information or warnings about the chosen file format, and a save

# modo 401

## Software Development Kit (SDK)



method, which reads the state of a scene or image object and writes that to the file. With API calls to the SDK you can access any parameter known to modo about the scene. Surface normals are supported, both per-vertex and per-polygon. Developers have access to topology information (e.g., what edges are connected to a given edge). If a new image saver is registered, then that format will appear in the "Save As" dialog for saving renders. You can also define your own layered image saver. The modo 401 renderer can generate a depth buffer and other effect buffers, which can be saved as an image or as part of a layered image.

Scene loaders can import vertices and polygons as dynamic or static (trisurf) meshes. Static meshes have lower memory requirements and therefore enable loading larger scenes. Additional API support is provided to access polygonal faces, subdivision surfaces, curves of various types, patches, and text polygons.

Plug-ins can access "user" preference settings to control their behavior. You can also use modo forms to design your plug-in's preferences panel user interface.

### Applications and Benefits

Direct access to modo data is provided to developers who need a customizable way to interface modo to other 3D and 2D applications.

- Export data from modo into the rest of your pipeline or tool suite. Control how modo data looks to your application, for seamless downstream use in your asset pipeline.
- Dump the contents of an LXO file into human-readable form.
- Import data from another application into modo. Read a proprietary in-house format or tackle an industry-standard format.
- Copy the contents of one LXO to another, applying custom operations as needed.

### Limitations

- Errors encountered during loading or saving can be sent to a log, however loaders and savers cannot open dialogs, fire commands or otherwise interact with the user.
- Plug-ins cannot read out GL tessellation or displacement — to save those users have to freeze the mesh first.
- For selection state, the SDK only allows scene plug-ins to determine the state of vertex maps, which is used by formats that only support a single UV map, to determine which map the user wants to save.
- Saver plug-ins can "pipe" data to another application by writing to a pipe, but this wouldn't be a live connection.