# Table of Contents

1 Introduction ................................................................. 7
   Games data translators ................................................. 7
   Introduction ............................................................ 7
   Translator processing has two basic stages internally .......... 7
   Common options for the internal data base ...................... 8
   Run a games translator ............................................... 8
   Customize and rebuild translators ................................ 8
   Build the necessary translators .................................... 9
   Installing on different platforms .................................. 10
   Load translators in Maya ............................................. 11

2 VRML2 Translator .......................................................... 13
   Games data translators ................................................. 13
   VRML2 ..................................................................... 13
   Overview of VRML2 ...................................................... 13
   VRML2 Features .......................................................... 13
   Files required to use this plug-in. ................................. 14
   Limitations ............................................................... 14
   Install the VRML2 translator ......................................... 15
   Set VRML2 properties in Maya ....................................... 15
   VRML2 Plug-ins. .......................................................... 15
   Link Objects to Web Addresses with VRML2Link .............. 16
   Create preset VRML viewpoints ...................................... 17
   vrm2TagsWin attribute descriptions ............................... 17
   Import a VRML2 file using wrl2ma ............................... 18

Menus ............................................................................ 20
File .............................................................................. 20
   File > Export All > VRML2 ........................................... 20
   Animation Options ...................................................... 20
   Export Options ........................................................... 21
   Texture Options ........................................................ 22
   VRML2 options .......................................................... 23
# Table of Contents

## 3 RTG Utility and File Format

- Developers
  - Games data translators
  - RTG utility and file format
    - Overview of RTG translator
    - RTG Features
    - Files required to use this plug-in
    - Install the RTG translator
  - Menus
    - File
      - File > Export All > RTG
        - RTG translator options
          - RTG Options
          - Animation Options
          - Export Options
          - Texture Options

## 4 Game Exchange 2 Translator

- Developers
  - Games data translators
  - Games exchange 2 translator
    - Overview of GE2 translator
    - GE2 Features
    - Files required to use this plug-in
    - Install the GE2 translator
  - Menus
    - File
      - File > Export All > GE2
        - Domain Options
        - Animation Options
        - Export Options
        - Texture Options
        - Other Options
        - Debug Output
Table of Contents

5 MDt API ......................................................... 41

Developer

Games data translators ......................................... 41
Games export toolkit (MDt API) ................................. 41
Overview of Games export toolkit. .............................. 41
Incorporate functions/ideas ...................................... 41
Use the MDt library. ............................................. 42
Find API function documentation ................................ 42
Use the MDt API .................................................. 43
  Structure of the internal database. ............................ 43
  Animation ....................................................... 45
Minimal Program example ....................................... 46
Basic Program example .......................................... 46
1 Introduction

Developer Games data translators

Introduction

This documentation provides information on how to use the Maya game translators.

Important note
Since Maya 4.0, we encourage you to use Maya’s API directly. Several new functions have been added to API since the MDT layer was created for Maya 1.0 which means the need to use the MDT layer has reduced. For Macintosh OS X, the source code has also been moved to: /Applications/Alias/Maya 6.0/devkit/obsolete/games.

Maya game translators are provided for licensed Alias customers. The source code for the translators as well as the MDtApi is provided so that the supplied translators can be modified for site-specific customizations.

The game translators, VRML2, RTG and GE2, are written with an API interface layer (MDtApi). The MDtApi functions are implemented using the Maya API. MDtApi does not replace the Maya API in any form. The Dt functions are meant to ease the re-use of source code between the translators and different platforms and to provide example API source code.

The game translators are File Translator plug-ins accessed from the File > Export All or File > Export Selection menu items.

Translator processing has two basic stages internally

- Creating an internal database of information that allows for relatively easy access to the Maya scene information. This database can be considered as a grouping mechanism.
- Running the translator using the information from the data base.

This process divides into separate options that you need to define. The first set of options deals with the setup of the internal database, and the second with the actual options for the translator that you are going to run.
Common options for the internal data base

Animation options
  including frame ranges, the level of requested animation

Export Options
  including hierarchy information, tessellation parameters, verbose activity reporting

Texture Options
  including shader evaluation parameters

Translator-specific options are described for each translator.

Run a games translator

The Maya games translators include VRML2 RTG and GE2.

Running a translator involves two processes:

• Loading the translator from the Window > Settings/Preferences > Plug-in Manager window.
• Starting the translation from either the File > Export All or File > Export Selection menu items or from within a MEL script, using the commands:

  file -rename output_filename
  file -exportAll
  or
  file -exportSelected

Customize and rebuild translators

The following shows you how to recompile and/or rebuild translators.

To recompile the translators:

1  Before you begin, you need to have a C++ compiler, such as:
   (IRIX and Linux) SGI C++ 7.3.1
   (Windows) Microsoft VC++ 6.0
   (Mac OS X) Metrowerks’ CodeWarrior 7.1

2  You can recompile the translator by creating a local work area and populate it with the following directory tree:
   MDt/MDtApi
   MDt/include
MDt/lib
MDt/vrml2Export
MDt/rtgExport
MDt/ge2Export

3 Copy the respective directories from $MAYA_LOCATION/devkit/obsolete/games.
Maya contains Makefiles for CodeWarrior, SGI, and MS IDE project files that can be used to rebuild the translators.

| Note | You may have to edit the Makefiles and customize the project files for specific Maya installation areas. |

4 Build the MDtApi library first. This creates a static library used for the other translators. It is possible to create this as a LIB (Mac OS X), DSO (IRIX and Linux), or DLL (Windows) file, but for the moment it is easier to create a static library for updates and changes to the translators.

(IRIX and Linux) The libMDtApi.a will be copied to the lib directory and referenced from there.

(Windows) The libMDtApi.lib is found in the Release subdirectory and is referenced from there.

(Mac OS X) The libMDtApi.lib is found in the CFMSupport folder and is referenced from there.

Build the necessary translators

| Note | For Mac OS X, you do not need to build any translators, since the libraries are already provided. |

1 Copy the translator plug-in file to the user’s Maya work area into the plug-ins directory. There should be a work area with a directory structure such as:

$HOME/maya/prefs
$HOME/maya/projects
$HOME/maya/plug-ins
$HOME/maya/scripts
Installing on different platforms

The direct translators for games are installed as a standard part of the Maya software under devkit/games.

The game translator directory structure is described in the following depending on which platform Maya is running.

Note
If the plug-ins or scripts directories are not present they should be created by the user.

2 Unload the current version of the plug-in from Windows > Settings/Preferences > Plug-in Manager.

3 Compile the new version of the plug-in.

4 Place the new translator plug into the plug-ins directory, and the MEL option script into the scripts directory.

5 Reload the plug-in from Windows > Settings/Preferences > Plug-in Manager.

6 If the MEL option script has been modified, source it again so that the changes are updated within Maya. Else the changes will not be seen.

7 Run the translator.

Note
Replacing a plug-in without first unloading it will cause Maya to crash on IRIX or Linux, and will fail to copy the new version of the plug-in on Windows and Mac OS X.

The translator binary image .lib (Mac OS X), .so (IRIX and Linux) or .mll (Windows) files are in:

(IRIX and Linux) /usr/aw/maya/bin/plug-ins/
(Windows) C:\Program Files\Alias\Maya6.0\bin\plug-ins
(Mac OS X) /Applications/Maya 6.0/Application Support/plug-ins

The MEL script files used for user options are in:

(IRIX and Linux) /usr/aw/maya/scripts/others/

Note
In the following directory locations it is assumed that for Windows, Maya is installed on drive C:. If this is not the case, change the “c:” references to the drive where it is installed.
Load translators in Maya

After you add the games plug-ins to the list of plug-ins, you still need to load the plug-ins to use them.

1. Select Windows > Settings/Preferences > Plug-in Manager to open the Plug-in Manager window and display the list of all known plug-ins.

2. Find the plug-in you need and click the loaded check box to load the plug-in. If you only turn on the auto load, you must restart Maya to load the plug-in.
Introduction | 1
Developer > Load translators in Maya
VRML2 Translator

Overview of VRML2

The VRML2 translator lets you export Maya scene files into VRML2 format. NURBS surfaces are tessellated to polygons and then output.

VRML2 Features

Outputs
- polygon geometry
- NURBS surfaces - tessellated into either triangles or quads
- hierarchy structure if wanted
- VRML2 primitives (if tagged)
- shader parameters
- textures - SGI RGBA image format
- rigid TRS animation
- vertex animation
- Camera information
- Light information

Commands

There is a MEL command included in the VRML2 translator. This command, `vrml2Tags`, let you define dynamic attributes onto objects that the translator queries and uses to define the output file.

Attributes

The following dynamic attributes can be defined and used with this translator.
- `userAnimated` boolean: On = force vertex animation processing
- `VRML2Primitive`: long, cube/plane

Note

The VRML2 translator does not output all of the Maya feature set.
VRML2 Billboards: long, camera facing/Y facing/Z facing

Files required to use this plug-in
The files required for use of this plug-in are:
- vrml2Export.lib (Mac OS X), vrml2Export.so (IRIX and Linux), or vrml2Export.mll (Windows)
- vrml2TranslatorOpts.mel

Source code can be found in:
- (IRIX and Linux) /devkit/obsolete/games/vrml2Export
- (Windows) Program Files\Alias\Maya 6.0\devkit\obsolete\games\vrml2Export
- (Mac OS X) Applications/maya 6.0/devkit/obsolete/games/vrml2Export

Limitations
- On Mac OS X, the VRML2 Exporter does not export textures.
- The range for the light intensity in VRML2 is [0 to 1.0]. Maya doesn’t have a limit. The translator assumes a scaling of [0-100] and divides down the Maya setting.
- No checks are made for negative scales. VRML2 defines scale to be greater than 0.
- Only SGI RGB/RGBA file textures are output. Microsoft’s VRML2 viewer does not support this format. Cosmoplayer does read the texture files on both IRIX and Linux and Windows systems. You may have to set up the file associations on the Windows platform to run Netscape instead of Explore if the auto launch feature is used on Windows systems.
- You need to turn on Sample for Texture Options to get proper rotation of textures.
- There may be memory leaks in the translator. This is because not all of the intermediate internal structures are deleted (for instance, animating NURBS surfaces will create a noticeable memory leak with Maya 1.0 due to the tesselation).
- Vertex animation is based on the connection of the inMesh attribute, so this will cause more geometry to be considered animated that may be. For final results, you may want to clean up the history of non-animated tweaks.
A check for dynamic attribute called *userAnimated* to force usage has been added. Use the Add Attribute command to add *userAnimated* as a boolean flag and then set it ON for those objects that you want to force the vertex animation to recognize and use. This will mostly be for those objects that don’t have history and an *inMesh* connection.

- Instanced geometry is output multiple times but doesn’t get reused.

### Install the VRML2 translator

You install the VRML2 translator plug-in using the Plug-in Manager.

1. Select **Windows > Settings/Preferences > Plug-in Manager** to display the list of all known plug-ins.

2. Click the loaded check box beside `vrml2Export.so` (IRIX and Linux), or `vrml2Export.mll` (Windows), or `vrml2Export.lib` (Mac OS X) to load the plug-in.

   If you only turn on the auto load, you must restart Maya to load the plug-in.

### Set VRML2 properties in Maya

**VRML2 Plug-ins**

The `vrml2Tags` command lets you attach tags to objects that have no effect in Maya as dynamic Attributes, but will be added to the VRML2 file.

The `vrml2Tags` command is a MEL script file `vrml2Tags.mel` which is found in the scripts/other directory.

**To set VRML2 properties:**

1. Select the objects you want to create with VRML2 attributes.

2. Type `vrml2Tags` in the Command Line to run the `vrml2Tags` command. The following window displays.
3 Change the settings of the vrml2 tags in this window and click the Add button.

4 These attributes are now available for further changes in the Extra Attributes section of the object’s Attribute Editor.

See the following for attribute descriptions and workflows.

**Link Objects to Web Addresses with VRML2Link**

The Link and Message attributes lets you create, verify, or remove hypertext links.

**To link objects to web addresses**

1 Select the object or objects you want to link.

2 Execute the vrml2Tags MEL command to display the current VRML2 tags information. Type `vrml2Tags` in the Command Line.

3 Press the Add button at the bottom of the window.

The attributes are now accessible from the selected object’s Attribute Editor’s Extra Attribute section.

4 In the vrml2 Link box, enter a Universal Resource Locator (URL). For example:

http://www.aw.sgi.com

You can also add a description of the destination using the form:

**URL**|**DESCRIPTION**

Where URL is the destination, the * (asterisk) is a separator, and DESCRIPTION is a description that will be displayed in the VRML browser when you point at the link.

For example:

http://www.alias.com*Alias Web Page

5 When you output the scene using File > Export All, the objects become active VRML links.
Create preset VRML viewpoints

Viewpoints are predefined camera positions and views in a VRML world. Viewpoints help viewers navigate your VRML worlds (for example, allowing them to return to an entrance point, or view close-ups of objects).

To create VRML viewpoints

1. Create a new camera.
2. Name the camera. Use a name that contains only valid VRML2 characters. This means do not use dashes or [ ] {}. For example, a valid VRML2 camera name would be EntranceView.
3. On the URL in the vrml2 Link field, type the name of the viewpoint this object will trigger, preceded by #. For example:

   #EntranceView

vrml2TagsWin attribute descriptions

The following describes the attributes in the vrml2TagsWin. Whatever you set in the vrmlTags window displays and becomes editable in the Extra Attributes section of the object’s Attribute Editor.

vrml2 Billboard menu

The VRML2 billBoard options include the standard VRML2 options, None, X-Rotating, Y-Rotating, and Screen Aligned.

vrml2 primitive menu

The VRML2 primitive options include the standard VRML2 options, None, Box, Cone, Cylinder, Sphere, and Elevation Grid.

vrml2 Sensor menu

The VRML2 sensor options include the standard VRML2 options, None, Cylinder, Sphere, Plane, Proximity, Touch, and Visibility.

vrml2 Collision menu

The VRML2 collision options include the standard VRML2 options, None and Object.

Do All selected checkBox

This checkbox performs the Add/Update operation on the entire current selection list. If it is not turned on, the current item in the selection list as shown at the top of the vrml2TagsWin window is used.

This is only useful if multiple objects are currently selected.
Add and Update check box
This checkbox combines the add and update functions at the same time. When turned off, the Add operation creates the dynamic attributes on the objects and uses the default settings. When turned on, the currently displayed settings are used to assign to the attributes.

Add button
Click this button to create default dynamic attributes on the objects in the current selection list. See “Do All selected check box” and “Add and Update check box” options for details.

Delete button
Click this button to delete the VRML2 dynamic attributes on the objects in the current selection list. See “Do All selected check box” and “Add and Update check box” options for details.

Selection stepping controls
The four stepping control buttons let you select which item to use for the individual object of interest. This is only useful if multiple objects are currently selected.

The buttons from left to right are first object, previous object, next object, and last object.

The current object of interest is listed in the first line of the vrml2Tags window.

Close button
Click this button to close the VRML2 tags window.

Import a VRML2 file using wrl2ma
The wrl2ma stand-alone program lets you convert VRML2 files to Maya ASCII.

The vrml2 importers include:
(IRIX and Linux and Mac OS X) wrl2ma
(Windows) wrl2ma.exe

The conversion includes:
• geometry
• normals
• texture coordinates
• color per vertex
- shader parameters (Lambert and Phong)
- texture file references (in-line textures are not supported)

**Note**  Animation is not supported with this version.

**To use wrl2ma**

Specify the input .wrl file and the output .ma file, and also specify the extensions you want.

For example, type something similar to the following in a system shell or terminal:

```
wrl2ma -i Porsc911.wrl -o Porsc911.ma
```

This example converts the VRML file Porsc911.wrl to a Maya 6.0 ASCII file named Porsc911.ma. The command provides the following options:

<table>
<thead>
<tr>
<th>Options</th>
<th>What it does</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>Prints the help file</td>
</tr>
<tr>
<td>-i &lt;inputFile&gt;</td>
<td>Specifies the input file to convert</td>
</tr>
<tr>
<td>-o &lt;outputFile&gt;</td>
<td>Specifies the output file to save to</td>
</tr>
<tr>
<td>-m</td>
<td>Assume input file was output by Maya</td>
</tr>
<tr>
<td>-v</td>
<td>Prints verbose messages</td>
</tr>
<tr>
<td>-d</td>
<td>Uses a debugging aid to see how the libvrmll97 scene parser parses the original VRML file.</td>
</tr>
</tbody>
</table>
**VRML2 Translator | 2**
Developer > File > Export All > VRML2

**Menus**

**File**

File > Export All > VRML2

**Animation Options**

**Range Control options**

**Loop**

- Enables the VRML2 option to loop the animation on playback.

**Time Slider**

- Uses the start/end range from the time slider instead of from the settings in this window.

**Enabled**

- Click to quickly disable any animation without resetting the ranges.

**Start/End**

- The Start value specifies the frame to start the output from (integer values). The End value specifies the frame to end the output.

**Step**

- This value defines the number of animated frames by which the animation is sampled between exported frames.

**Frames per sec**

- This value specifies the VRML2 playback frame rate.

**Animate**

- Click to turn on the following options. A check mark displays if the option is on and the option you select will animate.

**Vertices**

- Animates vertex animation at each frame (can be slow for NURBS geometry). This option is off by default.

**Transf**

- Animates transformation animation at each frame. This option is on by default.
VRML2 Translator | 2
Developer > File > Export All > VRML2

Materials
Animates shader material parameter animation. This option is on by default.

Lights
Animates light parameter animation. This option is on by default.

Cameras
Controls camera process information.

Keyframe using Anim Curves
Output by Keyframes on Animation curves. This option is off by default. (Note: This is a future option and is not currently implemented.)

Export Options
These options determine what elements you want to export.

Hierarchy
World
A World hierarchy tells the VRML 2 translator not to export hierarchy. This means that all vertices use world space.

Flat
A Flat hierarchy means one level of hierarchy. Selecting Flat means that transformations use world space and vertices use local space. There is no hierarchy/parenting information. This is the default Hierarchy selection.

Full
A Full hierarchy exports the full DAG tree hierarchy. This means that transformations and vertices use local space. Full hierarchy/parenting information is available.

Joints
Allows NULL chains to be output for hierarchy information, mostly for skeletons/joints.

Export
All
Exports all scene information (everything visible in the scene). This is the default Export selection.
VRML2 Translator | 2
Developer > File > Export All > VRML2

Picked
Exports everything picked in the scene (items lower in the DAG tree from the DAG nodes of picked objects are not included).

Active
Exports everything picked in the scene as well as any objects that are below them in the DAG tree.

Tessellation
Tri
Tessellates NURBS surfaces into triangles. This is the default Tessellation selection.

Quad
Tessellates NURBS surfaces into quads.

Include
Cameras
Allows Camera information to be collected for the database. This option is on by default.

Lights
Allows Light information to be collected for the database. This option is on by default.

Debug Info
Geo/Mat
Outputs a lot of information during the export process. Shape/material information is output here.

Cameras
Controls camera process information.

Lights
Controls light process information.

Texture Options
Evaluate
Uses convertSolidTx to create image files for both file and procedural textures. If turned off, Maya reads in the file textures as they are, and only converts the procedural textures.
Sample

Uses the 2D texture placement transformation information to generate the file image for output. For instance, the checker texture image consists of a checkerboard of 8x8 checkers. If Sample is off, the UV repeats are set to 1 internally, UV Rotate is set to 0.0 internally, and then the texture is processed. The result is that the checker image will be 2x2 (assuming the default 4 repeat).

Original

For file texture images, turn on to use a reference to the original file texture as defined in the texture node.

Resolution settings

Set the following X and Y texture resolution swatch sizes by entering a value and pressing Enter (IRIX and Linux and Windows) or Return (Mac OS X), or by dragging the slider.

X Tex Res/ Y Tex Res

Specifies the X or Y swatch size used for procedural textures.

Max X Tex Res/ Max Y Tex Res

Specifies the Max X or Y swatch size used for all textures. File textures larger than the size specified here are rescaled to the maximum size when the texture is computed with the convertSolidTx command.

Texture Search path

This is a list of directories to search for file textures if not found in the default list of directories obtained with workspace -q -rd .

VRML2 options

Navigation

This is the VRML2 navigation list. These settings control the initial set-up of the VRML browser (or viewer) when it opens the .wrl file. Click to turn on the following items (a check mark displays) to suit your needs.

Walk

Walk navigation is used to explore a virtual world on foot or in a vehicle that rests or hovers above the ground. It is strongly recommended that Walk navigation defines the up vector in the Y direction and provides some form of terrain and gravity to produce a walking or driving experience. If on, the browser supports collision detection. This option is off by default.
Examine

Examine navigation is used to view individual objects and often includes (but does not require) the ability to spin around an object and move the viewer closer or further away. This option is on by default.

Fly

Fly navigation is similar to Walk navigation except that terrain and gravity can be disabled or ignored. This option is off by default.

Any

If the Any navigation mode is selected, the browser chooses the navigation mode that best suits the content and provides a user interface that lets you change the navigation mode dynamically. If off, you cannot change the navigation mode and the browser only uses the modes specified in the list. This option is on by default.

None

None navigation disables and removes all browser-specific navigation user interface. This means that you can only navigate using mechanisms provided in the scene, such as Anchor nodes or scripts that include loadURL(). This option is off by default.

Options: Headlight

This is the VRML2 headlight control option. When turned on, the viewer casts light on the scene. This option is on by default.

Navigation speed

The value in this box is the VRML2 navigation speed setting. Enter a new value and press Enter (IRIX, Linux and Windows)/Return (Mac OS X) or drag the slider if you want to change the initial speed of the viewer.

Float precision

This is the number of digits used after the decimal point.

Export

Click to turn on the following options. A check mark displays if the option is on and the Export option you select will export to VRML2.

Normals

Exports normals. This option is off by default.

Opposite

Flips the direction of generated normals (not normally needed). This option is off by default.
Textures
Exports textures. This option is on by default.

Long Lines
Allows long lines in output file.
Turn off to break the VRML text file into short lines (each value on a separate line). For instance, turn this off if you want to edit the output file with a text editor that does not handle lines longer than approximately 80 characters well (such as vi). This option is off by default.

Verbose
Acquires feedback during the translation. This option is off by default.

Launch viewer
If turned on, on completion of the translator Maya launches Netscape (IRIX and Linux) or an associated program for files with the .wrl extension (Windows). This option is off by default, and not available on Mac OSX.

Compressed
Runs the gzip program to compress the output file. The file extension will still be .wrl. On Windows, the gzip.exe program should be located in the execution path. This option is off by default.

Reversed
Reverses the winding for NURBS surfaces that are single-sided and have the Opposite turned on. This option is on by default.

ColorPerVertex
Enables the output of color per vertex if it is available for objects being output. This option is off by default.

Texture path
The textures are saved in the directory you specify in this box.

Run script when done
Click the Browse button to select a script or enter the name of the script to run after the translation is complete. A system() call is made so the executable command should be found on the PATH statement.

Append file name to script
This option determines whether or not the output file name is appended to the end of the command line submitted. This option is on by default.
VRML2 Translator | 2
Developer > File > Export All > VRML2
3 RTG Utility and File Format

Developer Games data translators

RTG utility and file format

Overview of RTG translator

The RTG translator provides a method to output the Maya scene information generically. The source code is included so that site-specific changes or customizations can be done.

RTG Features

Outputs
- polygon geometry
- NURBS surfaces - tessellated into either triangles or quads
- hierarchy structure if wanted
- shader parameters
- textures - SGI RGBA image format
- rigid TRS animation

Commands

There are no MEL commands included with the RTG translator.

Attributes

There are no specific Maya attributes queried in the RTG translator.

Files required to use this plug-in

The files required for use of this plug-in are:
- rtgExport.lib (Mac OS X), rtgExport.so (IRIX and Linux), or rtgExport.mll (Windows)
- rtgTranslatorOpts.mel

Source code can be found in: devkit/obsolete/games/rtgExport

Install the RTG translator

You install the RTG translator plug-in using the Plug-in Manager.

1 Select Windows > Settings/Preferences > Plug-in Manager to display the list of all known plug-ins.
2  To load the plug-in, click the loaded check box beside rtgExport.so, rtgExport.mll, or rtgExport.lib (depending on your operating system). If you only turn on the auto load, you must restart Maya to load the plug-in.

**Menus**

**File**

**File > Export All > RTG**

**RTG translator options**

**RTG Options**

**Base options**
Click to turn on the following options. A check mark displays if the option is on and the option you select will be enabled during output.

**Verbose**
Acquires feedback during the translation. This option is off by default.

**Reversed**
Reverses the winding for NURBS surfaces that are single-sided and have the Opposite turned on. This option should be turned on by default.

**Image Format**
Choose either SGI RGB or Alias PIX from the list provided.

**Format**

**IRIX and Linux/DOS**
Select one of these options to determine whether the output ASCII files are written in IRIX and Linux `<lf>` or DOS `<cr>`<lf> format. Click to turn on the following options. A check mark displays if the option is on and the format option you select will output to rtg.

**V1.8 Compatible**
Enables some of the PA RTG 1.8 output formats.

**V Normals**
Outputs Vertex Normals.
P Normals
   Outputs Polygon Normals.

V Colors
   Outputs Vertex Colors.

UV Coords
   Outputs UV coordinates.
   **Games Translators** Outputs UV coordinates.

indexCnt
   Outputs index counters at the beginning of each list entry.

Degrees
   Outputs the transformation angle in degrees instead of radians.

Materials
   Outputs shading group Material properties.

MultiTexture
   Outputs separate texture images for those material properties that are texture mapped, including: diffuse, bump, translucence, specular color, cosinePower etc.

MDecomp
   Decomposes the transformation matrix into components of T/R/S.

Pivots
   Outputs pivot information.

Transforms
   Outputs transformation information.

Local Xform
   Outputs the local transformation matrix [4x4].

Animation
   Outputs T/R/S animation.

All Nodes
   Forces all nodes to output animation.
Animation Options

Range Control options

Time Slider

Uses the start/end range from the time slider instead of the settings in this window.

Enabled

Click to quickly disable any animation without resetting the ranges.

Start/End

The Start value specifies the frame to start the output from (integer values). The End value specifies the frame to end the output.

Step

This value defines the number of animated frames by which the animation is sampled between exported frames.

Export Options

These options determine what elements you want to export.

Hierarchy

World

A World hierarchy tells the RTG translator not to export hierarchy. This means that all vertices use world space.

Flat

A Flat hierarchy means one level of hierarchy. Selecting Flat means that transformations use world space and vertices use local space. There is no hierarchy/parenting information. This is the default Hierarchy selection.

Full

A Full hierarchy exports the full DAG tree hierarchy. This means that transformations and vertices use local space. Full hierarchy/parenting information is also available.

Joints

Allows NULL chains to be output for hierarchy information, mostly for skeletons/joints.
Export

All
Exports all scene information (everything visible in the scene). This is the default Export selection.

Picked
Exports everything picked in the scene (items lower in the DAG tree from the DAG nodes of picked objects are not included).

Active
Exports everything picked in the scene as well as any objects that are below them in the DAG tree.

Tessellation

Tri
Tessellates NURBS surfaces into triangles. This is the default Tessellation selection.

Quad
Tessellates NURBS surfaces into quads.

Include

Cameras
Allows Camera information to be collected for the database. This option is on by default.

Lights
Allows Light information to be collected for the database. This option is on by default.

Debug Info

Geo/Mat
Outputs a lot of information during the export process. Shape/material information is output here.

Cameras
Controls camera process information.

Lights
Controls light process information.
Texture Options

Texture control

Evaluate

Uses convertSolidTx to create image files for both file and procedural textures. If turned off, Maya reads in the file textures as they are, and only converts the procedural textures.

Sample

Uses the 2D texture placement transformation information to generate the file image for output. For instance, the checker texture image consists of a checker board of 8x8 checkers. If Sample is off, the UV repeats are set to 1 internally, UV Rotate is set to 0.0 internally, and then the texture is processed. The result is that the checker image will be 2x2 (assuming the default 4 repeat).

Original

For file texture images, turn on to use a reference to the original file texture as defined in the texture node.

Resolution settings

Set the following X and Y texture resolution swatch sizes by entering a value and pressing Enter (IRIX, Linux and Windows)/Return (Mac OS X) or dragging the slider.

X Tex Res/Y Tex Res

Specifies the X or Y swatch size used for procedural textures.

Max X Tex Res/Max Y Tex Res

Specifies the Max X or Y swatch size used for all textures. If the size of file texture images is larger than the maximum specified here, then the file textures will be scaled down to this maximum setting when the convertSolidTx command is processed.

Texture Search path

This is a list of directories to search for file textures if not found in the default list of directories obtained with workspace -q -rd.
4 Game Exchange 2 Translator

Developer Games data translators

Games exchange 2 translator

Overview of GE2 translator

The GE2 translator lets you export Maya scene files into GE2 format.

GE2 Features

The following lists GE2 “Outputs” and “Files required to use this plug-in”.

Outputs

- polygon geometry
- NURBS surfaces - tessellated into either triangles or quads
- hierarchy structure if wanted
- shader parameters
- textures - possible formats are: SGI RGBA, Alias wirefile, bmp, png, ppm, and tif
- TRS animation
- vertex animation

Note GE2 export does not properly produce .bmp format image files for converted textures. We suggest you specify .sgi or .tif format instead, and convert the textures to .bmp.

Commands

Use the ge2tags script to assign ge2 properties to Maya shaders. This command adds a set of extra attributes to all Maya shaders, and directs the Attribute Editor to display the new attributes in a tabbed, easy-to-manage format.

Attributes

The ge2tags script assigns extra attributes to shaders to correspond with ge2 Material attributes. These can be edited and queried just as any other attributes.
Files required to use this plug-in

The files required for use of this plug-in are:
- ge2Export.lib (Mac OSX), ge2Export.so (IRIX and Linux), or ge2Export.mll (Windows)
- ge2TranslatorOpts.mel
- ge2tags.mel
- ge2MatAttr.mel
- AElambertGe2Revision.mel
- Source code can be found in: devkit/obsolete/games/ge2Export

Install the GE2 translator

You install the vrml translator plug-in using the Plug-in Manager.

To load the plug-ins

1. Select Windows > Settings/Preferences > Plug-in Manager to display the list of all known plug-ins.
2. Click the loaded or auto load check box to load one of the ge2Export.mll, or ge2Export.so, or ge2Export.lib plug-ins.

Menus

File

File > Export All > GE2

Domain Options

Click to turn on the options. A check mark displays if the option is on and the option you select will be enabled during output. The checked domains will be filled with the material attributes.

Format data for domain:

- Options include: GL/DirectX, SonyPSX, N64, Custom

Animation Options

Enable Animation

Click to quickly disable any animation without resetting the ranges.
Start/End

The Start value specifies the frame to start the output from (integer values). The End value specifies the frame to end the output.

Step

The Step option is only available when the Keyframe using Anim Curves option is turned off. This value defines the number of animated frames by which the animation is sampled between exported frames.

Animate options

Click to turn the following options on or off.

Vertices

If on, Vertices exports vertex animation.

Transforms

If on, Transforms exports TRS animation.

Lights

If on, Lights animates the lights in the scene.

Camera

If on, Camera animates the camera in the scene.

Keyframe using options

Click to turn the following options on or off.

Anim Curves

If on, Anim Curves extracts keyframes from the AnimCurves on the shapes. If this option is selected, Step is set to 1 and disabled to ensure keyframes will fall on exported frames. If off, Step is enabled and you can change the default value.

Sampling

When on, Maya adaptively samples the animation. If this option is selected, Sample By and Tolerance are enabled. Frames are sampled at the given rate and exported as keyframes if geometry does not meet tolerance requirements.

Sample By

You must select Sampling in the Keyframe Using section to enable this option. This value represents the sample rate for adaptive sampling.
Tolerance
You must select Sampling in the Keyframe Using section to enable this option. This value determines whether sampled frames are exported as keys.

Animation Displacement options
These options are only enabled if you select Vertices in the Animate section.
- Relative to Current Frame specifies relative vertex displacements.
- Absolute generates absolute vertex displacement.

Export Options
These options determine what elements you want to export and how to export them.

Hierarchy

World
A World hierarchy tells the GE2 translator not to export hierarchy. This means that all vertices use world space.

Flat
A Flat hierarchy means one level of hierarchy. Selecting Flat means that transformations use world space and vertices use local space. There is no hierarchy/parenting information. This is the default Hierarchy selection.

Full
A Full hierarchy exports the full DAG tree hierarchy. This means that transformations and vertices use local space. Full hierarchy/parenting information is exported.

Export

All
Exports all scene information (everything visible in the scene). This is the default Export selection.

Selected
Exports everything picked in the scene (items lower in the DAG tree from the DAG nodes of picked objects are not included).

Active
Exports everything picked in the scene as well as any objects that are below them in the DAG tree.
Tessellation

Tri

Tessellates NURBS surfaces into triangles. This is the default Tessellation selection.

Quad

Tessellates NURBS surfaces into quads.

Export

If these options are on, (by default, all the Export options are on), Maya exports information for lights, cameras, textures, geometry, and normals.

Export options

The following options are off by default.

Reverse winding

If on, Maya enables reverse winding on polygons. Default winding is counter-clockwise.

Flip Normals

If on, Maya reverses the direction of normals. This option is off by default.

Texture Options

Texture Control

Uses convertSolidTx to create image files for both file and procedural textures. If turned off, Maya reads in the file textures as they are, and only converts the procedural textures.

Use 2d and 3d texture placement

Uses the 2D texture placement transformation information to generate the file image for output. For instance, the checker texture image consists of a checker board of 8x8 checkers. If off, the UV repeats are set to 1 internally, UV Rotate is set to 0.0 internally, and then the texture is processed. The result is that the checker image will be 2x2 (assuming the default 4 repeat).

Use convertSolidTx on file textures

This is a future option and is not currently implemented.

Use original file textures

If on, Maya reads in the file textures as they are, and only converts the procedural textures. If off, both procedural and file textures are converted.
Resolution settings
Set the following X and Y texture resolution swatch sizes by entering a value and pressing Enter (IRIX and Linux and Windows)/Return (Mac OS X) or dragging the slider.

X Tex Res/Y Tex Res
Specifies the X or Y swatch size used for procedural textures.

Max X Tex Res/Max Y Tex Res
Specifies the Max X or Y swatch size used for all textures. If the size of file texture images is larger than the maximum specified here, then the file textures will be scaled down to this maximum setting when the convertSolidTx command is processed.

Texture Search path
This is a list of directories to search for file textures if not found in the default list of directories obtained with workspace -q -rd.

Other Options

Float precision
Select a menu item to print out floats with the number of decimal places you choose.

Format Output
Tabs
If on, output is tabbed for readability. This option is on by default.

Comments
If on, output is commented (numbered indices, etc.). This option is off by default.

Debug Output
The following options are off by default.

Geometry
If on, Maya outputs information pertaining to geometry and anything attached to geometry (including materials) during the export process. A lot of information is generated during this process.

Cameras
If on, Maya outputs information about cameras during the export process.
Lights
If on, Maya outputs information about lights is during the export process.

Run script when done
Specify a script (and command line options) to be run after export to convert ge2 files to another file format or otherwise post-process the files.

Append file name to script
If on, the appropriate file (.grp or .gaf file) will be appended to the script you specify in Run script when done before invoking.
Game Exchange 2 Translator | 4
Developer > File > Export All > GE2
The MDt API is an implementation of the GAME EXPORT toolkit for Maya. It is a wrapper layer around the Maya API. Its main purposes are:

- To enable reuse of common code in multiple translators.
- Ease conversion of PA game translators over to Maya.
- Provide a level of grouping information obtained from the Maya API.
- Provide an additional source of Maya API examples.

### Important note
We encourage you to use Maya's API directly. Several new functions have been added to API since the MDT layer was created for Maya 1.0 which means the need to use the MDT layer has reduced. The source code has also been moved to: devkit/obsolete/games.

### Note
The MDt Api DOES NOT replace the Maya API. MDt Api functions work by using the Maya API.

### Incorporate functions/ideas
The MDt Api compiles into a static library which the translators can be linked with. The files that make up the MDt Api library are:

- **MDtShape.cpp**  
  geometry grouping (vertices, normals, colors, tessellation)

- **MDtMaterial.cpp**  
  shaders, generating texture images, material properties

- **MDtLayer.cpp**  
  miscellaneous control functions/startup/cleanup

- **MDtLight.cpp**  
  light parameters
Use the MDt library

The MDt library works by walking through the DAG structure of the scene and creating internal data tables that are then referenced by the MDt functions. This process can take a bit of time for large files or NURBS surfaces. If it takes too much time to generate the internal data tables, using the Maya API directly may better suit your needs.

Once the data tables are created, the translator is then able to access the tables and output the data in whatever format it needs to. References to the original Maya objects are stored (temporarily) so they can be referenced to get additional data without having to walk the DAG tree again.

Start with MDt functions and then use Maya API to obtain information that is not cached in the MDt data tables.

Tip

It is not recommended that you start with the Maya API and add in the MDt functions. The MDt functions can be used to quickstart or prototype a translator, and then use the knowledge gained to refine/customize the translator.

Find API function documentation

The documentation for the MDt API is integrated into the source files themselves where possible.
Use the MDt API

The MDt API is an example of how to use the Maya API. It is not the only method of obtaining Maya scene information from Maya. It is a combination of Maya API functions and executing MEL commands to get access to the data.

MDt API is not meant to replace Maya API in any way. It is grown and functions are added when deemed that something is of common interest to multiple uses. For example, not all of the Shader attributes are cached in the database. Those that are static and found to be used repeatedly are cached, but the API allows for the original Maya Object to be referenced to obtain additional data that is needed. It can be thought of as a higher level grouping function of common routines.

The two-stage process of generating the internal database and then running a translator with that data is a very general implementation. The scanning of the DAG tree and populating the internal database takes time. It is possible to write faster translators using the Maya API functions directly. The approach used for the VRML2 and other MDt based translators was to reduce the amount of work that had to be redone for each translator, and to reuse what was possible from pre-existing translators.

The VRML2, RTG, and GE2 game translators have been implemented as “file translators”. They could also be implemented as normal Maya MPxCommand plug-ins.

Structure of the internal database

The database that is generated by the walking of the DAG tree is controlled by user options that are set/defined in the option box of the file translators.

The information gathered for the database consists of:
- Shapes, Groups, Materials, Textures, Lights, and Cameras
- Controls

These options are described in the sections for each of the games translators using the MDt API.

There is a lot of flexibility in the settings that can be set. The source code is also supplied so that additional options can be added for local customizations.

The options are implemented in the file MDtLayer.cpp and the header file MDtExt.h. It is possible to add additional control flags to the source code and also modify the MEL option scripts to set/reset the options.
This would also then modify the MDt files that implement the particular area of interest for the option. The translator Option MEL script file would then be modified to pass the values of the options to the translator itself.

Shapes
Shapes are the basic transformation/joint/polygonal/NURBS geometry objects. Each “shape” has an associated number of parameters.

- Transformations
- Materials/Groups
- Parents

Each shape also has an associated list of Groups. If the shape is a pure transformation/joint node then the number of groups will be Zero (for instance, no shader assignments). This usually only occurs when the hierarchy is set to FULL, or all of the transformations are flattened out, and geometry is associated with all of them.

Some API functions that operate using the Shape index are:

- `DtShapeGetCount()` - returns the number of shape nodes found.
- `DtShapeGetMatrix(idx, *matrix)` - returns the transformation matrix for shape idx, in the variable matrix

In general it is faster to use the geometry from the ByShape based functions rather than those of the ByGroup family. There isn’t a second copy of the data generated and passed by to the caller.

The Shape functions are found in the source file MDtShape.cpp.

Groups
Groups are subsets of its Shape, that are defined by multiple shader assignments to it. For a NURB object there would be at most 1 group. For polygonal objects there would be 1 group for each shader that is assigned to that object. If there are no shaders assigned to the polygonal object then the number of groups will be zero. In this case it is most likely to be a transformation node of some kind.

The ByGroup functions need to allocate memory to generate the lists that are returned. Since the API functions don’t know when that memory is no longer used, it is up to you to free it when done with the returned lists.

The `MDtShape.cpp` file implements these functions.

Materials
Materials are the shaders found in the DAG tree walk.
For each shader that is found, a reference is stored for it, and can be accessed by either “name” or ID number. From the reference it is possible to use the Maya API directly to obtain the shader parameters of interest.

The `MDtMaterial.cpp` file implements these functions.

Textures
Textures are the textures mapped for the shaders. By default, only the color and transparency textures are evaluated and a combined rgba file image is generated in memory. It is possible to enable extra processing to output all or most of the textures mapped on a shader, this is part of the “MultiTexture” option.

The `MDtMaterial.cpp` file implements these functions

Lights
Lights information is gathered and made available. Similarly to Materials, a reference is temporarily saved to access the light parameters of interest.

The `MDtLights.cpp` file implements these functions

Camera
Camera information is gathered and made available. Similarly to the Materials, a reference is temporarily saved to access the camera parameter of interest.

The `MDtCamera.cpp` file implements these functions

Utility
There are some utility functions that allow for decomposing matrices, and other general functions. This are found in the `MDtLayer.cpp` file along with the control options for the DAG walk processing

Animation
Obtaining animation data out of the MDt functions is the same as getting the non-animation data out.

When the DAG tree is walked for generating the internal database, the references that are saved are used to obtain the geometry/material information that is needed.

Iterating over a range of animation, it is then accomplished with the `DtFrameSet(frame)` call which sets the current frame time. Maya automatically updates its dependency graph.

For each frame, `DtFrameSet` then updates the parameters that are wanted. Currently, this is mostly the geometry (vertices, normals, uvs) so the same function calls are reused to get the new frames geometry data.
The transformation data is obtained by referencing the transformation object. This data is not cached.

We recommend that you iterate over the animation range as few times as possible. See the following example:

```c
for ( animation = start, to end )
    for ( shape = 0, to last shape )
        get and save TRS data for shape
        next shape
    next frame
```

**Minimal Program example**

The bare minimum plug-in to use the MDt functions consist of:

- initializing the internal database
- walking the Maya scene graph and populating the database
- freeing the internal database

The following code snippet shows the functions that would need to be called. This snippet doesn’t actually output anything. There would normally be additional calls to customize the method of populating the internal database and actual calls to retrieve and output the data itself.

This is just a sample.

```c
// Initialize the internal database with a scene "name"
DtExt_SceneInit( "basic" );
// Now we can setup the database from the wire file geometry
DtExt_dbInit();
// Clean house.
DtExt_CleanUp();
```

**Basic Program example**

The following is a basic Dt based plug-in that queries the selected objects and outputs the name of the shape (mesh) and any “color” texture file names that are mapped to the shader.

```c
#include <maya/MSimple.h>
#include <MDt.h>
#include <MDtExt.h>
DeclareSimpleCommand( basic, "Alias - Example", "2.0");

void outputFileNames()
{
    char *shapeName;
    char *mtlName;
    char *texName;
```
// 1st thing find out how many shapes (meshes) are in this scene
// currently being looked at.
int numShapes = DtShapeGetCount();
// For each shape, check out what is assigned as the color texture
// There may not be a texture associated with material, or the texture
// may be a procedural texture and so have no file name.
for ( int shape=0; shape < numShapes; shape++ )
{
    // Find the name of the current shape that we are looking at.
    DtShapeGetName( shape, &shapeName );

    // Find out the number of groups (materials) assigned to this shape
    numGroups = DtGroupGetCount( shape );
    // For each group (material) check out to see if there is a file texture
    // associated with the material and output its name.
    // It is possible that the number of groups will be zero.
    // In this case there would be no materials assigned:
    // 1) Joint or other Transform node
    // 2) user deleted all of the materials including the default shading group
    // 3) removed the connections between the mesh and the shading groups.
    // Usual reason is #1, a Joint or straight Transform node.
    (when using
    // full hierarchy mode
    for ( int group=0; groups < numGroups; group++ )
    {
        // For the current shape+group combination find out its material
        DtMtlGetName( shape, group, &mtlName );

        // For the current material, find out its texture file name if any
        DtTextureGetFileName( mtlName, &texName );

        // output any information that is wanted about what we found above.
        printf( "for shape %s(%d) group %s(%d), texture filename is %s\n",
.shaTempName, shape, mtlName, group, texName

texName : "(NULL)" );
}
}
// main doIt function for the plugin command.
MStatus basic::doIt( const MArgList& args )
{
// Initialize the Dt database
DtExt_SceneInit( "basic" );
// using the default settings, as no initializer
// functions are called here
// Walk the dag and fill in the internal database
DtExt_initdb();

// walk thru shapes found and output the texture file names
outputFileNames();

// Clean up the allocated memory and internal storage
DtExt_CleanUp();
// return from the plugin command
return MS::kSuccess;
}
Index

Numerics
2D/3D texture placement
   GE2 Texture Option .................. 37

A
Active
   GE2 Export Option .................. 36
   RTG Export Option .................. 31
   VRML2 Export Option ................. 22
Add and Update checkBox
   VRML2 .............................. 18
All
   GE2 Export Option .................. 36
   RTG Export Option .................. 31
   VRML2 Export Option ................. 21
All Nodes .............................. 29
   RTG Export Option .................. 29
Animation .............................. 29
   RTG Export Option .................. 29
animation
   displace for GE2 ..................... 36
   MDt API description .................. 45
   sample for GE2 ....................... 35
   tolerance, set for GE2 ............... 36
animation playback
   loop, for VRML2 ...................... 20
Any
   VRML2 Export Option ................. 24
append file name
   to scripts, for GE2 .................. 39
   to scripts, for VRML2 ................ 25

B
browse
   scripts, for GE2 ..................... 39
   scripts, for VRML2 .................. 25
ByGroups
   MDt API description .................. 44

C
Camera
   GE2 Animate Option .................. 35
   RTG Export Option, debug info ........ 31
   RTG Export Option, Include ........... 31
camera
   animate, for GE2 .................... 35
Cameras
   GE2 debug info ...................... 38
   VRML2 debug info .................... 21, 22
   VRML2 Export Option, Include ........ 22
cameras
   MDt API description .................. 45
color
   per vertex, export
      VRML2 225
ColorPerVertex
   VRML2 Export Option .................. 25
Comments
   GE2 Export Option ................... 38
Compressed
   VRML2 Export Option .................. 25
Custom
   GE2 Export Domain option ............. 34

D
debug info
   Camera for RTG ...................... 31
   Cameras for GE2 ..................... 38
   Cameras for VRML2 ................... 21, 22
   Geo/Mat for RTG ..................... 31
   Geo/Mat for VRML2 ................... 22
geometry for GE2 ...................... 38
   Lights for GE2 ....................... 39
   Lights for RTG ....................... 31
   Lights for VRML2 .................... 22
Degrees
   RTG Export Option, Format ........... 29
digits
   number of, for VRML2 output .......... 24
Displacement
   GE2 Animation Option .................. 36
Index

Domain Options
GE2 Output options .......................... 34

E
Enabled
   RTG Animation Option .................. 30
   VRML2 Animation Option ............... 20
Evaluate
   RTG Export Option, Texture ........... 32
   VRML2 Export Option, Texture ........ 22
Examine
   VRML2 Output Option ................... 24

F
file textures
   use original, for GE2 .................... 37
Flat
   GE2 Export Option, Hierarchy .......... 36
   RTG Export Option, Hierarchy .......... 30
   VRML2 Export Option, Hierarchy ........ 21
flip
   normals, for VRML2 ..................... 24
   normals, using GE2 ...................... 37
Flip Normals
   GE2 Export Option ...................... 37
Float precision
   for GE2 ................................. 38
float precision
   for VRML2 output ...................... 24
Fly
   VRML2 Output Option ................... 24
Frames per sec
   VRML2 Animation Option ............... 20
Full
   GE2 Export Option, Hierarchy .......... 36
   RTG Export Option, Hierarchy .......... 30
   VRML2 Export Option, Hierarchy ........ 21

G
games translators
   build .................................... 9
   GE2, install ............................ 34
   installing ............................. 10
   recompile ............................. 8
   RTG, install ........................... 27
   running ............................... 8
   VRML2, install ......................... 15
GE2
   browse scripts for ..................... 39
   Cameras debug info ..................... 38
   Domain Options ........................ 34
   features ............................... 33
   file name, append to scripts .......... 39
   Geometry debug info ................... 38
   install translator ..................... 34
   Lights debug info ...................... 39
   Output Options ........................ 34
   Texture Search Path .................... 38
   translator options .................... 34
GE2 Animate Option
   Camera .................................. 35
   Vertices ................................ 35
GE2 Animation Option ..................... 35
Start/End ................................ 35
GE2 Displacement Option
   Animation option ....................... 36
GE2 Export Option
   Active .................................. 36
   All ...................................... 36
   Flat hierarchy ........................... 36
   Flip Normals ............................ 37
   Full hierarchy ........................... 36
   Reverse winding ......................... 37
   Selected ............................... 36
   Tri Tessellation ........................ 37
   World hierarchy ......................... 36
GE2 Keyframe Option
   Sample animation ....................... 35
   Tolerance, set for animation .......... 36
GE2 Texture Option
   2D/3D texture placement ............... 37
   textures, set resolution for .......... 38
   use original file textures ............. 37
Geo/Mat
  RTG Export Option, debug info .................. 31
  VRML2 debug info .............................. 22
Geometry
  GE2 debug info ................................. 38
GL/DirectX
  GE2 Output Domain option ...................... 34
gzip
  compress output file
    VRML225

Headlight
  control, for VRML2 output ....................... 24
hierarchy
  Flat, GE2 Export Option ......................... 36
  Flat, RTG Export Option ......................... 30
  Flat, VRML2 Export Option ...................... 21
  Full, GE2 Export Option ......................... 36
  Full, RTG Export Option ......................... 30
  Full, VRML2 Export Option ...................... 21
  World, GE2 Export Option ....................... 36
  World, RTG Export Option ....................... 30
  World, VRML2 Export Option .................... 21

iffreader.cpp .................................. 42
iffwriter.cpp .................................. 42
importing
  VRML2 ........................................... 18
IndexCnt ......................................... 29
  RTG Output Option ................................ 29
IRIX and Linux/DOS
  RTG Output Option, Format ....................... 28

Joints
  RTG Export Option ............................... 21, 30

Launch viewer
  VRML2 Export Option ............................. 25
lighting
  control, for VRML2 output ....................... 24
Lights
  GE2 debug info ................................. 39
  RTG Export Option, debug info .................. 31
  RTG Export Option, Include ..................... 31
  VRML2 Animate Option ........................... 21, 35
  VRML2 debug info ............................... 22
  VRML2 Export Option, Include ................... 22
lights
  animate, for VRML2 .............................. 21, 35
  MDt API description ............................ 45
Local Xform ...................................... 29
  RTG Output Option ............................... 29
Long Lines
  VRML2 Export Option ............................. 25
Loop
  VRML2 Animation Option ......................... 20

Materials ........................................ 29
  RTG Output Option ................................ 29
  VRML2 Animate Option ............................ 21
materials
  MDt API description ............................. 44
  shader, animate, for VRML2 ....................... 21
MDdecomp ........................................ 29
  RTG Output Option ................................ 29
Index

MDt API
- animation, description .................................. 45
- ByGroups, description .................................... 44
- cameras, description ...................................... 45
- controls, description ...................................... 43
- description .................................................... 41
- groups, description ........................................ 44
- lights, description ......................................... 45
- materials, description ..................................... 44
- program example, basic ................................... 46
- program example, minimal ................................ 46
- shapes, description ........................................ 44
- textures, description ...................................... 45
- using ................................................................ 43

MDt library
- using ............................................................ 42
- MDt.h ............................................................ 42
- MDtCamera.cpp .............................................. 42, 45
- MDCnetwork.cpp ............................................. 42
- MDtExt.h ........................................................ 42, 43
- MDtLayer.cpp ................................................ 41, 43
- MDtLight.cpp ................................................. 41
- MDtLights.cpp ................................................ 45
- MDtMaterial.cpp ............................................. 41, 45
- MDtShape.cpp ................................................. 41, 44
- MultiTexture ................................................... 29
- RTG Output Option .......................................... 29

O
- Opposite
  - VRML2 Export Option .................................... 24
- output
  - editing, for VRML2 ...................................... 25

P
- P Normals
  - RTG Output Option, Format .............................. 28
- Picked
  - RTG Export Option ........................................ 31
  - VRML2 Export Option .................................... 22
- Pivots .......................................................... 29
- RTG Output Option .......................................... 29
- playback
  - animation, for VRML2 ................................... 20
- polygons
  - flip normals, using GE2 ................................. 37
  - reverse winding, with GE2 .............................. 37

Q
- Quad
  - RTG Export Option, Tessellation ................. 31, 37
  - VRML2 Export Option, Tessellation ........... 22

R
- Resolution
  - GE2 Texture Option, set for textures ........... 38
  - RTG Export Option, set for textures ............ 32
  - VRML2 Export Options, set for textures ....... 23
- Reverse winding
  - GE2 Export Option ....................................... 37
- Reversed
  - RTG Output Option, basic ............................ 28
  - VRML2 Export option .................................... 25
- RTG
  - features .................................................. 27
  - install translator ........................................ 27
  - translator options ....................................... 27

Games Translators
52
Index

RTG Animation Option .................................. 30
   Enabled ............................................. 30
   Time Slider ........................................ 30
RTG Export Option
   Active .............................................. 31
   All .................................................... 31
   Camera debug info ................................. 31
   Camera, include .................................. 31
   Evaluate texture .................................. 32
   Flat hierarchy ..................................... 30
   Full hierarchy ..................................... 30
   Geo/Mat debug info ................................. 31
   Joints ................................................ 21, 30
   Lights debug info .................................. 31
   Lights, include ..................................... 31
   Picked ............................................... 31
   Quad Tessellation .................................. 31, 37
   Sample texture ..................................... 32
   Texture Search Path ................................. 32
   textures, set resolution for ........................ 32
   Tri Tessellation ..................................... 31
   World hierarchy ..................................... 30
RTG Output Option .................................. 28, 29
   Degrees format ..................................... 29
   IRIX and Linux/DOS formats ...................... 28
   P Normals format .................................... 28
   Reversed, basic ..................................... 28
   V Colors format ..................................... 28
   V Normals format .................................... 28
   Verbose, basic ....................................... 28

S
Sample
   GE2 Keyframe Option ................................. 35
   RTG Export Option, Texture ........................ 32
   VRML2 Export Option, Texture ...................... 23, 32

Sample By
   GE2 Keyframe Option ................................. 35

scripts
   append to, for GE2 .................................. 39
   append to, for VRML2 ................................ 25
   browse for GE2 ...................................... 39
   browse for VRML2 .................................... 25
search path
   for textures, GE2 ..................................... 38
   for textures, RTG Export ............................ 32
   for textures, VRML2 Export .......................... 23, 25

Selected
   GE2 Export Option ..................................... 36
   shader materials
     animate, for VRML2 .................................. 21
   SonyPSX
     GE2 Output Domain option ......................... 34
   Start/End ............................................. 30
     GE2 Animation Option ............................... 35
     RTG Animation Option ............................... 30
     VRML2 Animation Option ............................ 20
   Step ..................................................... 30, 35
     GE2 Animation Option ............................... 35
     RTG Animation Option ............................... 30
     VRML2 Animation Option ............................ 20

T
Tabs
   GE2 Output option ..................................... 38
Texture Search path
   GE2 ..................................................... 38
   RTG Export Options .................................. 32
   VRML2 Export Options ................................. 23, 25
Textures
   VRML2 Export Option .................................. 25
textures
   evaluate for RTG ...................................... 32
   evaluate for VRML2 .................................... 22
   export, for VRML2 ...................................... 25
   file, use original, for GE2 ............................ 37
   MDT API description ................................... 45
   place 2D/3D for GE2 .................................... 37
   sample for RTG .......................................... 32
   sample for VRML2 ....................................... 23, 32
   search path for GE2 .................................... 38
   search path for RTG Export ............................ 32
   search path for VRML2 Export ........................ 23, 25
Time Slider
   RTG Animation Option ................................. 30
   VRML 2 Animation Option ............................. 20

Games Translators

53
Index

Tolerance
   GE2 Keyframe Option .................................. 36
Transf
   VRML2 Animate Option ................................ 20, 35
transformations
   animate, for VRML2 ................................... 20, 35
Transforms ................................................. 29
   RTG Output Option ..................................... 29
translator
   install GE2. ........................................... 34
   install RTG ............................................. 27
   install VRML2 ......................................... 15
translators
   build, for games ....................................... 9
   installing, for games ................................. 10
   recompile, for games ................................. 8
   running, for games ................................... 8
   VRML2, introduction ................................ 13
Tri
   GE2 Export Option, Tessellation .................... 37
   RTG Export Option, Tessellation .................. 31
   VRML2 Export Option, Tessellation .............. 22

U
UI
   remove navigation, for VRML2 ...................... 24
   UV Coords ............................................. 29
   RTG Output Option ................................... 29

V
V Colors
   RTG Output Option, Format .......................... 28
V Normals
   RTG Output Option, Format .......................... 28
V1.8 Compatible
   RTG Output Option, Format .......................... 28
V1.8 Compatible format ................................ 28
Verbose
   RTG Output Option, basic ............................ 28
   VRML2 Export Option ................................ 25
Vertices
   GE2 Animate Option .................................... 35
   VRML2 Animate Option ............................... 20

Games Translators
54

vertices
   animate, for GE2 ....................................... 35
   animate, for VRML2 ................................... 20
viewer
   launch, for VRML2 .................................... 25
viewpoints
   creating with VRML2 ................................ 17
VRML 2 Animation Option
   Time Slider .............................................. 20
VRML Output Option
   Any ....................................................... 24
   Examine ............................................... 24
   Fly ..................................................... 24
   Walk .................................................... 23
VRML2
   attributes ................................................ 13
   browse scripts for .................................... 25
   Cameras debug info .................................. 21, 22
   commands .............................................. 13
   features ................................................ 13
   file name, append to scripts ........................ 25
   files ..................................................... 14
   Float precision option ............................... 24
   float precision setting ............................... 24
   for UNIX ............................................... 18
   for Windows .......................................... 18
   Geo/Mat debug info ................................ 22
   Headlight options .................................... 24
   install translator ..................................... 15
   introduction .......................................... 13
   lighting, for output .................................. 24
   Lights debug info .................................... 22
   Navigation speed option ............................. 24
   navigation speed, output for ........................ 24
   options ................................................ 19
   Output navigation check box ...................... 23
   remove navigation UI ................................ 24
   translator options ................................... 15
   viewpoints, creating ................................. 17
   VRML2Link, linking objects with .................. 16
   vrlm2Tags command, setting ...................... 15
VRML2 Animate Option
   Lights .................................................... 21, 35
   Materials .............................................. 21
   Transf .................................................. 20, 35
   Vertices ................................................ 20
### VRML2 Animation Option
- **Enabled** .................................................. 20
- **frames per sec** ........................................... 20
- **Loop** ....................................................... 20
- **Start/End** .................................................. 20
- **Step** ......................................................... 20

### VRML2 Export Option
- **Active** ..................................................... 22
- **All** .......................................................... 21
- **Cameras, include** ......................................... 22
- **ColorPerVertex** ........................................... 25
- **Compressed** ................................................ 25
- **Evaluate texture** .......................................... 22
- **Flat hierarchy** ............................................. 21
- **Full hierarchy** ............................................. 21
- **Launch viewer** ............................................ 25
- **Lights, include** ........................................... 22
- **Long Lines** ................................................ 25
- **Normals** ..................................................... 24
- **Opposite** .................................................... 24
- **Picked** ....................................................... 22
- **Quad Tessellation** ......................................... 22
- **Sample texture** ........................................... 23, 32
- **Texture Search Path** ...................................... 23, 25
- **Textures** ..................................................... 25
- **textures, set resolution for** ............................. 23
- **Tri Tessellation** ............................................ 22
- **Verbose** ...................................................... 25
- **World hierarchy** .......................................... 21

### VRML2 Output Option
- **Reversed** ................................................... 25

### vrml2Tags
- **command** .................................................. 15
- **setting** .................................................... 15
- **window** ..................................................... 15

### vrml2TagsWin
- **Add and Update checkbox** ............................... 18
- **Add button** ................................................ 18
- **attributes, descriptions** ................................ 17
- **Close button** ............................................. 18
- **Delete button** ............................................. 18
- **Do All selected checkbox** ................................ 17
- **stepping control buttons** ............................... 18
- **vrml2 Billboard menu** .................................... 17
- **vrml2 Sensor menu** ....................................... 17

### vrml2tagsWin
- **vrml2 Collision menu** .................................... 17

### vrmls2TagsWin
- **vrml2 primitive menu** .................................... 17

### W
- **Walk**
  - **VRML2 Output Option** ................................ 23

### World
- **GE2 Export Option, Hierarchy** .......................... 36
- **RTG Export Option, Hierarchy** .......................... 30
- **VRML2 Export Option, Hierarchy** ....................... 21

### wrl2ma
- **UNIX VRML2 import** ..................................... 18

### wrl2ma.exe
- **Windows VRML2 import** .................................. 18