Contents

Abbreviations.................................................................................. 11
Overview .......................................................................................... 13
Installation .................................................................................... 14
  Microsoft® Windows® ................................................................. 14
  Linux ............................................................................................. 14
Window Areas ................................................................................... 15
  Menu Bar ....................................................................................... 15
  Tool Icons ..................................................................................... 15
  Other Icons .................................................................................. 15
    File Icons ................................................................................... 16
    Edit Icons .................................................................................. 16
    Selection Mode Icons ............................................................... 17
    Miscellaneous Icons ................................................................. 17
    Palette Icons .............................................................................. 19
Model View ...................................................................................... 21
Tool Area ......................................................................................... 21
Tree View ......................................................................................... 21
Coordinates and Selections Field .................................................. 21
Status Area ....................................................................................... 21
Mouse Handling ............................................................................. 22
  Mouse Movement ........................................................................ 22
  Mouse Selection ......................................................................... 22
  Mouse Coordinate Indication ...................................................... 23
  Miscellaneous Mouse Functions ................................................. 23
  Space Navigator™ Support ......................................................... 23
File Dialogs ...................................................................................... 24
Windows File Dialogs ........................................................................ 24
Linux File Dialogs ............................................................................. 25
Menu .................................................................................................. 26
File Menu .......................................................................................... 26
Edit Menu ......................................................................................... 27
View Menu ....................................................................................... 28
Selection Menu .................................................................................. 29
Mode Menu ........................................................................................ 29
Tools Menu ....................................................................................... 30
Palettes Menu ................................................................................... 30
Script Menu ....................................................................................... 31
Misc Menu ........................................................................................ 31
Help Menu ........................................................................................ 31
Select By Attributes ........................................................................ 32
Shortcuts .......................................................................................... 33
Tools .................................................................................................. 36
XFORM (Transformation tools) ............................................................ 36

- Move .......................................................................................... 36
- Rotate ......................................................................................... 37
- Scale ............................................................................................ 38
- Soft Selection ................................................................................ 39
- Apply Transformation Matrix ...................................................... 40
- Eliminate Transformation Matrices ............................................. 40
CREATE (Creation tools) ................................................................. 41

- Create Node .............................................................................. 41
- Create Polygon .......................................................................... 41
- Create Disk .............................................................................. 41
- Create Sphere ........................................................................... 41
- Create External Reference ......................................................... 42
- Create Light Point ..................................................................... 42

SURFACE (Surface appearance tools) .......................................... 42

- Apply Color .................................................................................. 42
- Read Color From Node ................................................................. 43
- Apply Texture .............................................................................. 43
- Read Texture From Node .............................................................. 43
- Apply Texture Mapping ............................................................... 43
- Read Texture Mapping From Node ................................................ 43
- Apply Material ............................................................................ 43
- Read Material From Node ............................................................ 44
- Apply Shader ................................................................................ 44
- Read Shader From Node ............................................................... 44
- Remove Texture .......................................................................... 44
- Remove Texture Mapping ............................................................ 44
- Remove Material ......................................................................... 44
- Remove Shader ........................................................................... 44

TEXMAP (Texture mapping tools) .................................................. 45
Map Texture ................................................................. 45
Move Texture Map .......................................................... 46
Rotate Texture Map .......................................................... 47
Scale Texture Map .......................................................... 48
Modify UV ................................................................. 49

MODIFY (Modification tools) ........................................ 50

Mirror ................................................................. 50
Reverse Polygon .......................................................... 50
Slice ................................................................. 51
Split/Triangulate .......................................................... 51
Extrude ................................................................. 51
Revolve ................................................................. 52
Subdivide ................................................................. 52
Simplify ................................................................. 53
Merge Polygons .......................................................... 54
Quantify ................................................................. 54
Shade ................................................................. 54

ATTR (Attribute tools) ................................................ 55

Apply Attributes .......................................................... 55
Apply Comment .......................................................... 70

Modes ................................................................. 71

Vertex ................................................................. 71
Edge ................................................................. 71
Polygon ............................................................. 71
Object ............................................................... 71
Group ............................................................... 71
LOD ................................................................. 71
DOF ................................................................. 71
Switch ............................................................. 72
Light Point ....................................................... 72
Light Point System .............................................. 72
External Reference .............................................. 72

Tree View .......................................................... 73

Node Colors ...................................................... 74

Palettes ............................................................ 75
Color Palette ...................................................... 75
Texture Palette .................................................. 76
Texture Attributes ............................................. 78
Texture Mapping Palette ..................................... 82
Material Palette ................................................ 84
Light Point Palette ............................................. 85
Light Point Appearance Attributes ....................... 86
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Light Point Animation Attributes</td>
<td>88</td>
</tr>
<tr>
<td><strong>Shader Palette</strong></td>
<td>90</td>
</tr>
<tr>
<td><strong>Command List</strong></td>
<td>92</td>
</tr>
<tr>
<td><strong>Preview DOF</strong></td>
<td>93</td>
</tr>
<tr>
<td><strong>Preview Animation</strong></td>
<td>94</td>
</tr>
<tr>
<td><strong>Settings</strong></td>
<td>95</td>
</tr>
<tr>
<td>Preferences</td>
<td>95</td>
</tr>
<tr>
<td>Viewport Menu</td>
<td>97</td>
</tr>
<tr>
<td>Viewport Preferences</td>
<td>100</td>
</tr>
<tr>
<td>Viewport Preferences</td>
<td>100</td>
</tr>
<tr>
<td><strong>License Handling</strong></td>
<td>102</td>
</tr>
<tr>
<td>Node-locked and USB dongle licenses</td>
<td>102</td>
</tr>
<tr>
<td>Floating licenses</td>
<td>103</td>
</tr>
<tr>
<td>Time limited license</td>
<td>104</td>
</tr>
<tr>
<td><strong>Installed Contents</strong></td>
<td>105</td>
</tr>
<tr>
<td>Built-in Help System</td>
<td>105</td>
</tr>
<tr>
<td>Tutorial</td>
<td>105</td>
</tr>
<tr>
<td>Samples</td>
<td>105</td>
</tr>
<tr>
<td>Sample Models</td>
<td>105</td>
</tr>
<tr>
<td>Scripts</td>
<td>105</td>
</tr>
<tr>
<td>Macros</td>
<td>105</td>
</tr>
<tr>
<td><strong>Help</strong></td>
<td>106</td>
</tr>
<tr>
<td><strong>Scripts</strong></td>
<td>107</td>
</tr>
<tr>
<td>Script Environment</td>
<td>107</td>
</tr>
<tr>
<td>Script Samples</td>
<td>108</td>
</tr>
</tbody>
</table>
Script Reference ......................................................................................................................... 110

General Script Functions ............................................................................................................. 111
File Script Functions .................................................................................................................. 113
Clipboard/Edit Script Functions ................................................................................................. 115
Selection Script Functions .......................................................................................................... 116
Transformation Script Functions ................................................................................................. 121
Creation Script Functions ........................................................................................................ 127
Surface Script Functions ........................................................................................................... 129
Texture Mapping Script Functions .............................................................................................. 131
Modification Script Functions .................................................................................................... 133
Attribute Script Functions .......................................................................................................... 136
Reporting Script Functions ........................................................................................................ 139
Grid Script Functions ................................................................................................................ 139
Color Palette Script Functions ................................................................................................... 141
Texture Palette Script Functions ................................................................................................. 143
Texture Palette Script Functions ................................................................................................. 143
Material Palette Script Functions ............................................................................................... 146
Texture Mapping Palette Script Functions ................................................................................ 150
Light Point Palette Script Functions .......................................................................................... 152
Shader Palette Script Functions ................................................................................................. 157

Macros ........................................................................................................................................... 163

Macro Editor ................................................................................................................................. 163
Macro Samples ............................................................................................................................. 164
Sharing Macros ............................................................................................................................. 166

Troubleshooting .......................................................................................................................... 167

Technical Specifications .............................................................................................................. 169

OpenFlight® File Format .............................................................................................................. 169
Texture File Formats .................................................................................................................... 170
Import File Formats ..................................................................................................................... 170
Export File Formats .................................................................................................................... 171
Environment Variables ................................................................................................................ 171

Setting an Environment Variable .................................................................................................. 171
Environment Variable Names ........................................................................................................ 172

System Configuration .................................................................................................................. 173

Development Tools ...................................................................................................................... 174
Index .................................................................................................................. 175
Remo 3D™ v2.8 User’s Guide

Copyright © 2018 Remograph AB.
All rights reserved.

All nationally or internationally recognized trademarks and trade names referenced are properties of their respective owners and are hereby acknowledged. This software is based in part on the work of the Independent JPEG Group and the FLTK project (www.fltk.org). Remo 3D™ is based in part on the work of the FLTK project (http://www.fltk.org) and uses Pthreads (license described in LGPLv2). libungif (license described in GIF_LICENSE.txt), Boost (license described in BOOST.txt) and the International Components for Unicode (license described in ICU.txt). Portions of this software are copyright © 2007 The FreeType Project (www.freetype.org). All rights reserved. This software contains Autodesk® FBX® code developed by Autodesk, Inc. Copyright 2014 Autodesk, Inc. All rights, reserved. Such code is provided "as is" and Autodesk, Inc. disclaims any and all warranties, whether express or implied, including without limitation the implied warranties of merchantability, fitness for a particular purpose or non-infringement of third party rights. In no event shall Autodesk, Inc. be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of such code.

The information in this user’s guide is subject to change without notice at any time. No responsibility is assumed by Remograph for its use.

END USER LICENSE AGREEMENT

This end user license agreement is a legal agreement between Remograph and you (either an individual or, if purchased or otherwise acquired by or for an entity, an entity). Remograph owns all intellectual property rights in the Remo 3D software and permits you to use the Remo 3D software only in accordance with the terms of this agreement.

Please read this agreement carefully before installing, starting or otherwise using the Remo 3D software. Any of the above actions indicate that you have read all of the terms and conditions of this agreement, understand them, and agree to be legally bound to those terms and conditions. Should you not accept the terms and conditions of this agreement, then do not install, start or otherwise use the Remo 3D software.

The Remo 3D software and user's guide are provided "as is" without warranty of any kind, either expressed or implied, including without limitation, any warranty of merchantability and fitness for a particular purpose.

Remograph has tested the Remo 3D software extensively in a number of different setups, but you assume the entire risk as to the results and performance of the software. Remograph assumes no responsibility for the accuracy or application of or errors or omissions in the software.

You acknowledge that Remograph has no control over your use or exploitation of the software. You therefore agree to indemnify and hold Remograph harmless against any costs, claims, demands, expenses and liabilities of whatsoever nature by any third party arising from such use or exploitation.

In no event shall Remograph be liable for any direct, indirect, special, incidental or consequential damages arising out of the use or inability to use the Remo 3D software, even if Remograph has been advised of the likelihood of such damages occurring. Remograph shall not be liable for any loss, damages or costs, arising out of, but not limited to, lost profits or revenue, loss of use of the software or the media, loss of data or equipment, the costs of recovering the software, the media, data or equipment, the cost of substitute software, media, data or equipment or claims by third parties, or other similar costs.

Since local legislation in some countries does not allow any exclusion or limitation of implied warranties or limitation of liability for incidental or consequential damages; the above limitations or exclusions may apply to you only partly or not at all.

In any event, if any statute implies warranties or conditions not stated in this license agreement, your entire liability under any provision of this license agreement shall be limited to the amount actually paid by you to license the software.

Remograph AB
Norrbervägen 58
SE-589 61 Sturefors
SWEDEN

Internet: www.remograph.com
E-mail: info@remograph.com
sales@remograph.com
support@remograph.com
## Abbreviations

This section describes the abbreviations used in this documentation.

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3DS</td>
<td>3D Studio</td>
</tr>
<tr>
<td>ATTR</td>
<td>ATTRibute</td>
</tr>
<tr>
<td>BMP</td>
<td>BitMaP</td>
</tr>
<tr>
<td>COL</td>
<td>COLOr</td>
</tr>
<tr>
<td>DDS</td>
<td>Direct Draw Surface</td>
</tr>
<tr>
<td>DLL</td>
<td>Dynamic Link Library</td>
</tr>
<tr>
<td>DOF</td>
<td>Degree-Of-Freedom</td>
</tr>
<tr>
<td>EXE</td>
<td>EXEcutable file in Microsoft® Windows®</td>
</tr>
<tr>
<td>FLT</td>
<td>FLighT (OpenFlight® file format extension)</td>
</tr>
<tr>
<td>GIF</td>
<td>Graphics Interchange Format</td>
</tr>
<tr>
<td>HSV</td>
<td>Hue-Saturation-Value</td>
</tr>
<tr>
<td>ID</td>
<td>IDENTification</td>
</tr>
<tr>
<td>INT</td>
<td>INTensity</td>
</tr>
<tr>
<td>INTA</td>
<td>INTensity-Alpha</td>
</tr>
<tr>
<td>IR</td>
<td>Infra-Red</td>
</tr>
<tr>
<td>IRISGL</td>
<td>Integrated Raster Imaging System Graphics Library</td>
</tr>
<tr>
<td>IVE</td>
<td>OpenSceneGraph deprecated binary file format</td>
</tr>
<tr>
<td>JPEG</td>
<td>Joint Photographic Experts Group</td>
</tr>
<tr>
<td>LOD</td>
<td>Level-Of-Detail</td>
</tr>
<tr>
<td>MAC</td>
<td>Media Access Control</td>
</tr>
<tr>
<td>MAT</td>
<td>MATerial</td>
</tr>
<tr>
<td>N/A</td>
<td>Not Available</td>
</tr>
<tr>
<td>NAD</td>
<td>North American Datum</td>
</tr>
<tr>
<td>NE</td>
<td>North-East</td>
</tr>
<tr>
<td>OpenGL</td>
<td>Open Graphics Library</td>
</tr>
<tr>
<td>OSG</td>
<td>OpenSceneGraph deprecated ASCII file format</td>
</tr>
<tr>
<td>OSGB</td>
<td>OpenSceneGraph binary file format</td>
</tr>
<tr>
<td>OSGT</td>
<td>OpenSceneGraph text file format</td>
</tr>
<tr>
<td>OSGX</td>
<td>OpenSceneGraph XML file format</td>
</tr>
<tr>
<td>P3D</td>
<td>3D model file format used by VBS2</td>
</tr>
<tr>
<td>PBM</td>
<td>Portable Bit Map</td>
</tr>
<tr>
<td>PGM</td>
<td>Portable Gray Map</td>
</tr>
<tr>
<td>PNG</td>
<td>Portable Network Graphics</td>
</tr>
<tr>
<td>PNM</td>
<td>Portable aNy Map</td>
</tr>
<tr>
<td>PPM</td>
<td>Portable Pixel Map</td>
</tr>
<tr>
<td>RCP</td>
<td>Remo Color Palette</td>
</tr>
<tr>
<td>RGB</td>
<td>Red-Green-Blue</td>
</tr>
<tr>
<td>RGBA</td>
<td>Red-Green-Blue-Alpha</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td>RLP</td>
<td>Remo Light Point Palette</td>
</tr>
<tr>
<td>RMP</td>
<td>Remo Material Palette</td>
</tr>
<tr>
<td>RSP</td>
<td>Remo Shader Palette</td>
</tr>
<tr>
<td>RTP</td>
<td>Remo Texture Palette</td>
</tr>
<tr>
<td>RXP</td>
<td>Remo Texture Mapping Palette</td>
</tr>
<tr>
<td>SGI®</td>
<td>Silicon Graphics Incorporated</td>
</tr>
<tr>
<td>SHP</td>
<td>ESRI SHApe file format</td>
</tr>
<tr>
<td>STL</td>
<td>StereoLithography</td>
</tr>
<tr>
<td>SW</td>
<td>South-West</td>
</tr>
<tr>
<td>TAR</td>
<td>Tape ARchive (archive file format)</td>
</tr>
<tr>
<td>TEX</td>
<td>TEXTure</td>
</tr>
<tr>
<td>TGA</td>
<td>TarGA</td>
</tr>
<tr>
<td>TIFF</td>
<td>Tagged Image File Format</td>
</tr>
<tr>
<td>UTM</td>
<td>Universal Transverse Mercator</td>
</tr>
<tr>
<td>VBS2</td>
<td>Virtual BattleSpace 2</td>
</tr>
<tr>
<td>WGS</td>
<td>World Geodetic System</td>
</tr>
<tr>
<td>XML</td>
<td>eXtensible Markup Language</td>
</tr>
<tr>
<td>ZIP</td>
<td>Compressed archive file format</td>
</tr>
</tbody>
</table>
Overview

Remo 3D™ is an effective tool for creating and modifying 3D models intended for realtime visualization. Remo 3D's primary file format is OpenFlight® and it allows for importing from and exporting to different file formats.

As opposed to many other 3D modeling products that are primarily intended for rendering, Remo 3D offers the user full control of the model scene graph and allows for modification of features like degree-of-freedom nodes, level-of-detail nodes, switch nodes, etc. Furthermore, the process of creating models in Remo 3D allows for focusing on the individual polygons and vertices. All in all, this makes Remo 3D exceptionally suitable for creating and modifying 3D models intended for realtime visualization.
Installation

This section describes the installation of Remo 3D™ v2.8 on your system.

**Microsoft® Windows®**

Perform the following steps to install Remo 3D™ v2.8 in Microsoft® Windows®:

- Unpack the downloaded ZIP file.
- Run the unpacked installation EXE file and follow the instructions.

Remo 3D™ v2.8 can be installed both with and without administrative privileges. Administrators can install for all users or for the current user only. The application can be uninstalled from the Program menu, by executing uninstall.exe in the Remo 3D installation folder or from the Windows Control Panel.

You can also install and uninstall Remo 3D™ v2.8 non-interactively from the command prompt with the following commands (/S is for silent):

- `remo3d_v2.8_64_install.exe /S /INSTDIR=<installation_folder>`
- `"<installation_folder>\uninstall.exe" /S`

This can be useful for automatic installation scripts. Here are examples of installing and uninstalling from the command-line:

- `remo3d_v2.8_64_install.exe /S /INSTDIR=C:\Program Files\Remo3D`
- `"C:\Program Files\Remo3D\uninstall.exe" /S`

**Linux**

Perform the following steps to install Remo 3D™ v2.8 in Linux:

- Unpack the downloaded gzipped TAR file with one of the following commands:
  ```bash
  tar xvzf remo3d_v2.8_64_linux.tar.gz
  
  or
  
gunzip remo3d_v2.8_64_linux.tar.gz; tar xvf remo3d_v2.8_64_linux.tar
  ```

- The unpacked `remo3d_v2.8_linux` folder is the installed Remo 3D™ v2.8 folder in which `remo3d` is the executable file.
Window Areas

This section describes the different areas of the Remo 3D™ graphical user interface.

Menu Bar

The menu bar is located at the top of the Remo 3D window. It is described in the Menu section.

Tool Icons

Along the left border of the Remo 3D window is a scrollable area containing tool icons. These are described in the Tools chapter.

Other Icons

Below the menu bar is an area containing icons for file operations, edit operations, selection modes and palettes.
**File Icons**

![New](image)

By clicking this icon a new model is created, called untitled.flt.

![Open](image)

By clicking this icon the user can choose to open one or more OpenFlight® files. The general functionality of the file dialog is described in the File Dialogs section.

Note that OpenFlight® files also can be opened by drag and drop, i.e. dragging the files from a file manager and dropping them on to the Remo 3D window. Another way of opening models is by specifying their file names on the command-line:

```
remo3d <filename1> <filename2> ...
```

or for instance like this in Microsoft® Windows®:

"C:\Program Files\Remo3D\remo3d.exe" <filename1> <filename2> ...

![Save](image)

By clicking this icon the current model is saved to disk, using the same name as before and overwriting the old model.

**Edit Icons**

![Cut](image)

By clicking this icon the selected primitives are copied to the clipboard and deleted from the model. The old contents of the clipboard are erased.

![Copy](image)

By clicking this icon the selected primitives are copied to the clipboard. The old contents of the clipboard are erased. Note that the current pivot point is also copied and restored when pasting.
Paste

By clicking this icon the primitives in the clipboard are added to the current parent in the model. The current pivot point is also changed to the pivot point saved together with the primitives in the clipboard.

Undo

By clicking this icon the latest command is reversed. Commands that respond to undo/redo are all tools or palette operations that modify the model, left- or right-shifting of nodes, all types of selection operations, cutting and pasting of geometry, drag and drop operations, moving of the pivot point and setting the parent node. Undoable and redoable commands can be viewed in the Command List described in the Command List section.

Redo

By clicking this icon the latest command that were undone is re-activated. Commands that respond to undo/redo are all tools or palette operations that modify the model, left- or right-shifting of nodes, all types of selection operations, cutting and pasting of geometry, drag and drop operations, moving of the pivot point and setting the parent node. Undoable and redoable commands can be viewed in the Command List described in the Command List section.

Selection Mode Icons

The mode icons are described in the Modes chapter.

Miscellaneous Icons

The miscellaneous icons are described in the following sections.

Shrink Polygons

To be able to select separate vertices sharing the same position, see thin polygons and such, it can be convenient to temporarily shrink all polygons by clicking this icon. The shrinking is done by moving all vertices toward the center of their polygons. The distance moved is 20% of the distance between the vertex and the polygon center.
Show Normals

This icon activates and deactivates visualization of vertex normals, if any.

Report Statistics

This icon activates a statistics report for the sub-trees of the selected nodes. The statistics include numbers of vertices, polygons, objects, groups, LODs, switches, DOFs, light point systems, light points and external references in the selected sub-trees as well as minimum, maximum and center coordinates together with the size of the selected geometry. It also reports the area of all polygons in the selected sub-tree. Note that the number of polygons reported might be higher than the number of polygons selected, since some of the selected polygons might have polygons (sub-faces) beneath them.

Report Differences

By clicking this icon the user can choose to compare the currently selected sub-tree with either the sub-tree contained in the current clipboard (put there by a Copy or Cut operation) or with another OpenFlight® file. In the case of comparing with an OpenFlight® file, the header node of the current model should be selected in order for the report to make any sense. The user can also specify if the revision date or version of the file (not the OpenFlight® version) to compare with should be ignored or not.

The result is presented in a message box, reporting the first difference encountered. This could be e.g. different node types, node names, attributes or different numbers of children.

Snap To Grid

When indicating coordinates grid snapping can be used with a tolerance of 25% of the current grid spacing.

Show Vertices Large

This icon activates and deactivates visualization of large vertices.

Show Vertex Numbers

This icon activates and deactivates visualization of vertex numbers, starting with 0 for the first vertex in a polygon or light point node.
Palette Icons

Current Color (COL)
This icon shows the currently chosen color. Clicking it will open the color palette described in the Color Palette section and let the user select the current color or edit the color palette.

Current Texture (TEX)
This icon shows the currently chosen texture. Clicking it will open the texture palette described in the Texture Palette section and let the user select the current texture or edit the texture palette.

Current Texture Mapping (TXM)
This icon shows the currently chosen texture mapping. Clicking it will open the texture mapping palette described in the Texture Mapping Palette section and let the user select the current texture mapping or edit the texture mapping palette.

Current Material (MAT)
This icon shows the currently chosen material. Clicking it will open the material palette described in the
Material Palette section and let the user select the current material or edit the material palette.

**Current Light Point (LPT)**

Clicking this icon will open the light point palette described in the Light Point Palette section and let the user select the current light point appearance and animation or edit the light point palette.

**Current Shader (SHD)**

Clicking this icon will open the shader palette described in the
Shader Palette section and let the user edit the shader palette.

Model View

The area to the left and beneath the icons is the model view, where the model is three-dimensionally visualized. The model view contains tabs so that several models can be in memory simultaneously. With Ctrl+TAB and Ctrl+Shift+TAB the user can switch forwards and backwards between these models. A model can be presented in one or more viewports. These viewports, as well as the tree view, can be resized with respect to each other by clicking and moving their borders.

Tool Area

The area to the right of the model view is the tool area. This is used by tools that need user interface components like buttons, input and output fields.

Tree View

The area beneath the model view is the tree view, which is described in the Tree View chapter.

Coordinates and Selections Field

The area beneath the tree view is used for displaying the current and previous coordinates, the delta in X, Y and Z and the distance from the previous coordinate. It is also used for displaying a list of the names of the currently selected primitives together with the number of primitives selected. Furthermore, the name of the current parent node is presented on a button. To change the current parent, select the new parent node and click the parent button or press Ctrl+P.

This field also contains three convenient buttons for aligning the grid to the XY, XZ or YZ plane.

Status Area

The area at the bottom of the Remo 3D window is the status area. This is used for displaying informative text of the current action. If the mouse cursor is located over a node in the tree view, the complete node name is also displayed in the status area.
**Mouse Handling**

This section describes the different ways of controlling Remo 3D using the mouse. Note that except for an ordinary mouse, the 3Dconnexion Space Navigator™ is supported for controlling the model view as described in the Space Navigator™ Support section.

**Mouse Movement**

The table below describes movement of the model using the mouse.

Note that the sensitivity of panning, zooming and rotating can be set in the preferences described in the Preferences section or by using the Sensitivity dialog available from the View menu.

<table>
<thead>
<tr>
<th>Mouse Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Right mouse button</td>
<td>Rotates view around the current center point, horizontally and vertically.</td>
</tr>
<tr>
<td>Shift + Right mouse button</td>
<td>Zooms view in (move mouse downwards) and out (move mouse upwards).</td>
</tr>
<tr>
<td>Ctrl + Right mouse button</td>
<td>Pans view.</td>
</tr>
<tr>
<td>Scroll wheel</td>
<td>Zooms view in (scroll downwards) and out (scroll upwards).</td>
</tr>
<tr>
<td>Shift + Scroll wheel</td>
<td>Zooms view in and out in finer steps.</td>
</tr>
</tbody>
</table>

**Mouse Selection**

The table below describes selection of geometry using the mouse.

Note that sometimes selecting small primitives, e.g. one vertex, by one click can be difficult. In those cases, try selecting with a small rectangular area instead.

<table>
<thead>
<tr>
<th>Mouse Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clicking left mouse button</td>
<td>Selects the nearest primitive.</td>
</tr>
<tr>
<td>Clicking and dragging left mouse button</td>
<td>Selects primitives within a rectangular area, even those behind.</td>
</tr>
<tr>
<td>Holding Shift and clicking left mouse button</td>
<td>Adds the nearest primitive to the existing selection.</td>
</tr>
<tr>
<td>Holding Alt and clicking left mouse button</td>
<td>Removes the nearest primitive from the existing selection.</td>
</tr>
<tr>
<td>Holding Shift and clicking and dragging left mouse button</td>
<td>Adds non-selected primitives within the rectangular area to the existing selection.</td>
</tr>
<tr>
<td>Holding Alt and clicking and dragging left mouse button</td>
<td>Removes selected primitives within the rectangular area from the existing selection.</td>
</tr>
</tbody>
</table>
Mouse Coordinate Indication

The table below describes coordinate indication using the mouse. Indicating coordinates is required for controlling certain tools, defining the grid as three points, translating the grid, etc.

<table>
<thead>
<tr>
<th>Mouse Combination</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Middle mouse button</td>
<td>Indicates a coordinate in the grid. Snaps to grid points if snap is active.</td>
</tr>
<tr>
<td>Ctrl + Middle mouse button</td>
<td>Moves pivot point of current selection to a coordinate in the grid. Snaps to grid points if snap is active.</td>
</tr>
<tr>
<td>Shift + Middle mouse button</td>
<td>Indicates a coordinate, snapping to the center point of an existing primitive. The current primitive mode decides which type of primitive to which to snap.</td>
</tr>
<tr>
<td>Ctrl + Shift + Middle mouse button</td>
<td>Moves pivot point of current selection to a coordinate, snapping to the center point of an existing primitive. The current primitive mode decides which type of primitive to which to snap.</td>
</tr>
</tbody>
</table>

Miscellaneous Mouse Functions

The table below describes miscellaneous use of the mouse buttons.

<table>
<thead>
<tr>
<th>Mouse Combination</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ctrl + Alt + Right mouse button</td>
<td>Opens a pop-up menu with various settings for the currently focused viewport. This menu is the same as Viewport in the View menu and is described in detail in the Viewport Menu section. The menu can also be reached by pressing the Menu button on the keyboard.</td>
</tr>
</tbody>
</table>

Space Navigator™ Support

Remo 3D supports model view manipulation with the 3Dconnexion Space Navigator™. This support requires that the 3Dconnexion driver is installed. The pan, zoom and rotate configuration from the 3Dconnexion control panel will be used and the sensitivity is controlled as described in the View Menu and Preferences sections. For more information about this device, see www.3dconnexion.com.
File Dialogs

This section describes the different file dialogs used throughout the Remo 3D graphical user interface. The file dialogs will have different appearances depending on which operating system you are on.

In both Microsoft® Windows® and Linux, the file dialog for saving OpenFlight® files will contain a drop-down menu list of available OpenFlight® versions.

Windows File Dialogs

In Microsoft® Windows®, an ordinary file dialog matching the current Windows version is used, except for the addition of the OpenFlight® version in the save dialog. No further description of file dialogs in Microsoft® Windows® is needed.
Linux File Dialogs

In Linux, a common file dialog independent of the Linux distribution or window system is used. The functionality of this file dialog is described below.

When opening all file dialogs in Remo 3D for Linux (e.g. for opening or saving of OpenFlight® models, textures or palettes), they will show the latest folder visited in that particular dialog in the current Remo 3D session.

The favorites list always contains a File Systems favorite which will show the available file systems in the file list.

Automatic completion is used when typing the filename. The file type can be chosen in the Files Of Type drop-down menu at the bottom of the dialog.

In file dialogs intended for saving or exporting files, folders can be created by clicking the New Folder icon in the top right corner of the file dialog. Navigation to the upper level in the folder hierarchy can be performed by clicking the Up One Level icon to the left of the New Folder icon.

The folder currently visited can be saved to a list of favorites by choosing Add To Favorites from the Favorites drop-down menu at the top of the file dialog. A favorite folder can be navigated to again by choosing it from the Favorites drop-down menu. The favorites can also be moved and deleted by choosing Manage Favorites from the same drop-down menu. The favorites are remembered until the next session of Remo 3D.
Menu

This section describes all items in the Remo 3D menu.

File Menu

The table below describes the contents of the File menu.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>Creates a new model, called untitled.flt.</td>
</tr>
<tr>
<td>Open...</td>
<td>Lets the user open one or more OpenFlight® files. The general functionality of the file dialog is described in the File Dialogs section.</td>
</tr>
<tr>
<td></td>
<td>Note that OpenFlight® files also can be opened by dragging them from a file manager and dropping them on to the Remo 3D window. Another way of opening models is by specifying their file names on the command-line: remo3d &lt;filename1&gt; &lt;filename2&gt; ...</td>
</tr>
<tr>
<td>Revert...</td>
<td>Opens the current model from disk again.</td>
</tr>
<tr>
<td>Revert Externals</td>
<td>Opens all external references from disk again.</td>
</tr>
<tr>
<td>Revert Textures</td>
<td>Reloads all textures from disk again.</td>
</tr>
<tr>
<td>External References...</td>
<td>Shows a list of all externally referenced models, uniquely by file name, allowing for enabling and disabling. Use the Apply button to confirm. If the Load External References setting described in the Preferences section is disabled, all external references in this dialog will disabled to start with.</td>
</tr>
<tr>
<td>Open Recent</td>
<td>Contains up to four recent 3D model files that can be opened directly.</td>
</tr>
<tr>
<td>Save</td>
<td>Save the current model to disk, using the same name as before and overwriting the old model.</td>
</tr>
<tr>
<td>Save As...</td>
<td>Lets the user choose a filename and location for saving the opened model to disk. The OpenFlight® version can also be selected. If a file with the chosen name already exists, the user can choose to overwrite the file or not. The general functionality of the file dialog is described in the File Dialogs section.</td>
</tr>
<tr>
<td>Import...</td>
<td>Lets the user import one or more files of different 3D file formats. The general functionality of the file dialog is described in the File Dialogs section. Each imported file results in a new untitled model containing the geometry from the imported file. The import file formats supported are listed in the Technical Specifications section. Embedded textures and shaders, as well as externally referenced files, can be converted and written to disk. Remo 3D uses OpenSceneGraph plugins for importing different file formats. These plugins can be controlled with a number of settings specified as an option string. This string can be set</td>
</tr>
</tbody>
</table>
using the REMO3D_IMPORT_EXPORT_OPTIONS environment variable.

Note that 3D model files supported for import also can be imported by dragging them from a file manager and dropping them on to the Remo 3D window. Another way of importing models is by specifying their file names on the command-line:

```bash
remo3d <filename1> <filename2> ...
```

**Export As...**

Lets the user choose a filename for exporting the current model to another 3D file format. If a file with the chosen name already exists, the user can choose to overwrite the file or not. The general functionality of the file dialog is described in the File Dialogs section. The export file formats supported are listed in the Technical Specifications section.

Remo 3D uses OpenSceneGraph plugins for exporting different file formats. These plugins can be controlled with a number of settings specified as an option string. This string can be set using the REMO3D_IMPORT_EXPORT_OPTIONS environment variable.

**Close**

Closes the currently selected model. If changes have been made, the user is given the opportunity to save the model.

**Close All Models**

Closes all models. If changes have been made to a model, the user is given the opportunity to save that model.

**Exit**

Exits the program. All models are closed. If changes have been made to a model, the user is given the opportunity to save that model.

---

### Edit Menu

The table below describes the contents of the Edit menu.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Undo</td>
<td>Reverses the latest command. Commands that respond to undo/redo are all tools or palette operations that modify the model, left- or right-shifting of nodes, all types of selection operations, cutting and pasting of geometry, drag and drop operations, moving of the pivot point and setting the parent node. Undoable and redoable commands can be viewed in the Command List described in the Command List section.</td>
</tr>
<tr>
<td>Redo</td>
<td>Re-activates the latest command that were undone. Commands that respond to undo/redo are all tools or palette operations that modify the model, left- or right-shifting of nodes, all types of selection operations, cutting and pasting of geometry, drag and drop operations, moving of the pivot point and setting the parent node. Undoable and redoable commands can be viewed in the Command List described in the Command List section.</td>
</tr>
<tr>
<td>Cut</td>
<td>Copies selected primitives to the clipboard and deletes them from the model. The old contents of the clipboard are erased.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies selected primitives to the clipboard. The old contents of</td>
</tr>
</tbody>
</table>
the clipboard are erased. The current pivot point is also saved to the clipboard.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paste</td>
<td>Adds the primitives in the clipboard to the current parent in the model. The pivot point is changed to the pivot point saved together with the primitives in the clipboard.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes selected primitives.</td>
</tr>
<tr>
<td>Set Parent</td>
<td>Changes the current parent to the currently selected node.</td>
</tr>
<tr>
<td>Change Parent</td>
<td>Moves the selected nodes to become children of the current parent.</td>
</tr>
<tr>
<td>Shift Far Left</td>
<td>Shifts the selected nodes all the way to the left in the tree structure.</td>
</tr>
<tr>
<td>Shift Left</td>
<td>Shifts the selected nodes one step to the left in the tree structure.</td>
</tr>
<tr>
<td>Shift Right</td>
<td>Shifts the selected nodes one step to the right in the tree structure.</td>
</tr>
<tr>
<td>Shift Far Right</td>
<td>Shifts the selected nodes all the way to the right in the tree structure.</td>
</tr>
<tr>
<td>Preferences...</td>
<td>Opens the preferences window, described in the Preferences section.</td>
</tr>
</tbody>
</table>

### View Menu

The table below describes the contents of the View menu.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Next Model</td>
<td>Switches to the next model among the tabs.</td>
</tr>
<tr>
<td>Previous Model</td>
<td>Switches to the previous model among the tabs.</td>
</tr>
<tr>
<td>More Detail</td>
<td>Shows the next more detailed LOD level.</td>
</tr>
<tr>
<td>Most Detail</td>
<td>Shows the most detailed LOD level.</td>
</tr>
<tr>
<td>Less Detail</td>
<td>Shows the next less detailed LOD level.</td>
</tr>
<tr>
<td>Least Detail</td>
<td>Shows the least detailed LOD level.</td>
</tr>
<tr>
<td>Show All LODs</td>
<td>Shows all LODs simultaneously.</td>
</tr>
<tr>
<td>Sensitivity...</td>
<td>Lets the user set the sensitivity for panning, zooming and rotating the view. This input dialog is non-modal which means that it is possible to open the dialog and continue working with the rest of the Remo 3D graphical user interface.</td>
</tr>
<tr>
<td>Multi-texture</td>
<td>In this sub-menu, the user can choose which of the eight multi-texture layers that are to be visible. It’s also possible to enable or disable all layers or only enable the base layer 0.</td>
</tr>
<tr>
<td>Show Gizmos</td>
<td>Toggle if gizmos should be shown or not.</td>
</tr>
<tr>
<td>Command List...</td>
<td>Opens the command list where all commands executed so far can be viewed. The command list is described in the Command List section.</td>
</tr>
<tr>
<td>Preview DOF...</td>
<td>Opens a DOF preview dialog described in the Preview DOF section.</td>
</tr>
<tr>
<td>Preview Animation...</td>
<td>Opens an animation preview dialog described in the</td>
</tr>
</tbody>
</table>
Viewport

This sub-menu is only available when a viewport is focused. This menu is described in Viewport Menu in the Viewport Menu section.

Selection Menu

The table below describes the contents of the Selection menu.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>Selects all primitives of the currently chosen type. Selected primitives of other types are deselected.</td>
</tr>
<tr>
<td>None</td>
<td>Deselects all primitives.</td>
</tr>
<tr>
<td>Invert Selection</td>
<td>Selects all primitives of the currently chosen type except for those already selected. Selected primitives of other types are deselected.</td>
</tr>
<tr>
<td>Select By Name...</td>
<td>Selects a node with a specified name. The user can specify the name, whether the search should be case sensitive or not and if the selection should be added to the previous selection or created as a new selection.</td>
</tr>
<tr>
<td></td>
<td>To select vertices, add a /Vx suffix to the polygon or light point node name, where x is the zero-based vertex number. To select edges, add a /Ex suffix, where x is the zero-based edge number.</td>
</tr>
<tr>
<td>Select By Attributes...</td>
<td>Selects all primitives matching a given condition of desired attributes. Select By Attributes is described in more detail in the Select By Attributes section.</td>
</tr>
<tr>
<td>Select Polygons By Number Of Vertices...</td>
<td>Selects all polygons with a given number of vertices.</td>
</tr>
<tr>
<td>Select Sliver Polygons</td>
<td>Selects all polygons in the model (or in the selected subtree) having an angle smaller than the current tolerance between two edges. Those polygons are considered to be slivers. The current tolerance is described in the Preferences section.</td>
</tr>
<tr>
<td>Select Parents</td>
<td>Selects the parents of the currently selected nodes.</td>
</tr>
<tr>
<td>Select Children</td>
<td>Selects the child nodes of the currently selected nodes.</td>
</tr>
<tr>
<td>Select Previous</td>
<td>Switches to the latest previous selection. This will essentially toggle between the two latest selections.</td>
</tr>
</tbody>
</table>

Mode Menu

The table below describes the contents of the Mode menu.

<table>
<thead>
<tr>
<th>Mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vertex Mode</td>
<td>Sets vertices to be interactively selectable in the model view. Vertices are corner points of polygon or light point nodes.</td>
</tr>
<tr>
<td>Edge Mode</td>
<td>Sets edges to be interactively selectable in the model view. Edges are virtual structures representing pairs of consecutive vertices.</td>
</tr>
<tr>
<td>Polygon Mode</td>
<td>Sets polygons to be interactively selectable in the model view.</td>
</tr>
</tbody>
</table>
Polygons are surface nodes, each one containing a number of vertices.

**Object Mode**  
Sets objects to be interactively selectable in the model view. Objects are hierarchy grouping nodes containing only polygon or light point nodes.

**Group Mode**  
Sets groups to be interactively selectable in the model view. Groups are general hierarchy grouping nodes.

**LOD Mode**  
Sets LODs to be interactively selectable in the model view. LODs are level-of-detail hierarchy grouping nodes switched on or off depending on viewing distance.

**DOF Mode**  
Sets DOFs to be interactively selectable in the model view. DOFs are degree-of-freedom hierarchy grouping nodes allowing for transformation of contained geometry.

**Switch Mode**  
Sets switches to be interactively selectable in the model view. Switches are hierarchy grouping nodes switching contained geometry on or off with a number of switch masks.

**Light Point Mode**  
Sets light points to be interactively selectable in the model view. Light points are light point nodes, each one containing a number of vertices.

**Light Point System Mode**  
Sets light point system nodes to be interactively selectable in the model view. Light point system nodes are hierarchy grouping nodes containing only light points.

**External Reference Mode**  
Sets external references to be interactively selectable in the model view. External references are nodes referencing external OpenFlight® files.

## Tools Menu

The Tools menu contains all tools described in the Tools chapter.

## Palettes Menu

The table below describes the contents of the Palettes menu.

<table>
<thead>
<tr>
<th>Palette</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Color Palette...</td>
<td>Opens the color palette of the currently focused model and lets the user select the current color or edit the color palette.</td>
</tr>
<tr>
<td>Texture Palette...</td>
<td>Opens the texture palette of the currently focused model and lets the user select the current texture or edit the texture palette.</td>
</tr>
<tr>
<td>Texture Mapping Palette...</td>
<td>Opens the texture mapping palette of the currently focused model and lets the user select the current texture mapping or edit the texture mapping palette.</td>
</tr>
<tr>
<td>Material Palette...</td>
<td>Opens the material palette of the currently focused model and lets the user select the current material or edit the material palette.</td>
</tr>
<tr>
<td>Light Point Palette...</td>
<td>Opens the light point palette of the currently focused model and lets the user select the current light point appearance and animation or edit the light point palette.</td>
</tr>
</tbody>
</table>
Shader Palette... Opens the shader palette of the currently focused model and lets the user edit the shader palette.

Script Menu

The table below describes the contents of the Script menu.

<table>
<thead>
<tr>
<th>Script Editor...</th>
<th>Opens the Script Editor described in the Script Environment section.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Macro Editor...</td>
<td>Opens the Macro Editor described in the Macro Editor section.</td>
</tr>
<tr>
<td>Macros</td>
<td>This sub-menu contains available macros, described in the Macros chapter.</td>
</tr>
</tbody>
</table>

Misc Menu

The Misc menu contains the same functionalities as described in the Miscellaneous Icons section.

Help Menu

The table below describes the contents of the Help menu.

<table>
<thead>
<tr>
<th>Contents...</th>
<th>Opens the help window described in the Help chapter.</th>
</tr>
</thead>
<tbody>
<tr>
<td>About Remo 3D...</td>
<td>Shows short information about the current version of the program and how to get in contact with Remograph.</td>
</tr>
<tr>
<td>Purchase...</td>
<td>Opens the purchase dialog described in the License Handling section.</td>
</tr>
</tbody>
</table>
Select By Attributes

This dialog is used for selecting all primitives matching a given condition of desired attributes.

The total condition will consist of a number of expressions. To construct an expression, click on a desired attribute label in the list to the right, select an operator (Equal To, Not Equal To, Greater Than or Less Than) and specify a desired value of the chosen attribute.

Select By Attributes will only work with one selection mode at a time, one can’t select e.g. vertices and polygons at the same time this way. That would require using the Add To Selection check box described later. The selection mode can be changed with the Mode buttons in the top of this dialog. Note that this will reset the current condition. If the primitives currently selected share the same value of a certain attribute, this will be the default for the desired value. For light points, the user has to choose between “ordinary” inline light points or indexed light points.

An expression can be added to the total condition by clicking the ADD button or the OR button, representing the logical operator with which the expression will be added. The latest expression added can be removed with the Remove Last button. With the Add To Selection check box, the user can also specify if the selection should be added to the previous selection or created as a new selection.

The last condition used will remain after applying it. If the dialog is closed and opened again the condition will still be the same as before, in which case the previously used selection mode will also be activated. The Clear button will reset the condition.
# Shortcuts

This section lists all keyboard shortcuts available in Remo 3D.

<table>
<thead>
<tr>
<th>Shortcut</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ctrl+n</td>
<td>New</td>
</tr>
<tr>
<td>Ctrl+o</td>
<td>Open...</td>
</tr>
<tr>
<td>Ctrl+s</td>
<td>Save</td>
</tr>
<tr>
<td>Shift+Ctrl+s</td>
<td>Save As...</td>
</tr>
<tr>
<td>Ctrl+i</td>
<td>Import...</td>
</tr>
<tr>
<td>Ctrl+e</td>
<td>Export As...</td>
</tr>
<tr>
<td>Ctrl+w</td>
<td>Close Model</td>
</tr>
<tr>
<td>Shift+Ctrl+w</td>
<td>Close All Models</td>
</tr>
<tr>
<td>Ctrl+q</td>
<td>Exit</td>
</tr>
<tr>
<td>Ctrl+z</td>
<td>Undo</td>
</tr>
<tr>
<td>Ctrl+y</td>
<td>Redo</td>
</tr>
<tr>
<td>Ctrl+x</td>
<td>Cut</td>
</tr>
<tr>
<td>Ctrl+c</td>
<td>Copy</td>
</tr>
<tr>
<td>Ctrl+v</td>
<td>Paste</td>
</tr>
<tr>
<td>DELETE/BACKSPACE</td>
<td>Delete</td>
</tr>
<tr>
<td>Ctrl+p</td>
<td>Set Parent</td>
</tr>
<tr>
<td>Ctrl+r</td>
<td>Change Parent</td>
</tr>
<tr>
<td>Ctrl+TAB</td>
<td>Next Model</td>
</tr>
<tr>
<td>Ctrl+Shift+TAB</td>
<td>Previous Model</td>
</tr>
<tr>
<td>Ctrl+m</td>
<td>More Detail</td>
</tr>
<tr>
<td>Shift+Ctrl+m</td>
<td>Most Detail</td>
</tr>
<tr>
<td>Ctrl+l</td>
<td>Less Detail</td>
</tr>
<tr>
<td>Shift+Ctrl+l</td>
<td>Least Detail</td>
</tr>
<tr>
<td>Ctrl+t</td>
<td>Sensitivity...</td>
</tr>
<tr>
<td>Ctrl+g</td>
<td>Show Gizmos</td>
</tr>
<tr>
<td>Ctrl+a</td>
<td>Select All</td>
</tr>
<tr>
<td>Ctrl+d</td>
<td>Select None</td>
</tr>
<tr>
<td>Alt+Ctrl+0</td>
<td>Toggle multi-texture layer 0 (base).</td>
</tr>
<tr>
<td>Alt+Ctrl+1</td>
<td>Toggle multi-texture layer 1.</td>
</tr>
<tr>
<td>Alt+Ctrl+2</td>
<td>Toggle multi-texture layer 2.</td>
</tr>
<tr>
<td>Alt+Ctrl+3</td>
<td>Toggle multi-texture layer 3.</td>
</tr>
<tr>
<td>Alt+Ctrl+4</td>
<td>Toggle multi-texture layer 4.</td>
</tr>
<tr>
<td>Alt+Ctrl+5</td>
<td>Toggle multi-texture layer 5.</td>
</tr>
<tr>
<td>Alt+Ctrl+6</td>
<td>Toggle multi-texture layer 6.</td>
</tr>
<tr>
<td>Alt+Ctrl+7</td>
<td>Toggle multi-texture layer 7.</td>
</tr>
<tr>
<td>F1</td>
<td>Help</td>
</tr>
<tr>
<td>Shift+V</td>
<td>Vertex Mode</td>
</tr>
<tr>
<td>Shift+E</td>
<td>Edge Mode</td>
</tr>
</tbody>
</table>
Shift+F | Polygon (face) Mode  
Shift+O | Object mode  
Shift+G | Group mode  
Shift+L | LOD mode  
Shift+D | DOF mode  
Shift+S | Switch mode  
Shift+P | Light Point Mode  
Shift+T | Light Point System Mode  
Shift+R | External Reference Mode  

s | Shrink Polygons  
n | Show Normals  
p | Perspective  
o | Orthographic  
Z | Top view  
z | Bottom view  
Y | Front view  
y | Back view  
X | Right view  
x | Left view  
J | View toward and below grid  
j | View toward and above grid  
Q | Store current view  
q | Recall stored view  
v | In viewport: Fit view to the current selection, changing the point of rotation and zoom.  
| In tree view: Scroll to the selected node. If there are more than one node selected, pressing v again will cycle trough them.  
SPACE | Fit model in view and possibly recalculate automatic pan and zoom sensitivity  
w | Toggle between solid, scribed and wireframe modes  
t | Toggle texture on/off  
r | Toggle shader visualization on/off  
l | Toggle lighting on/off  
m | Toggle lighting mode between head light and sky light  
g | Toggle grid on/off  
a | Align grid to three vertices  
f | Align grid to a polygon (face)  
+ | Increase grid size  
- | Decrease grid size  
Shift+'+' | Double grid size  
Shift+'-' | Half grid size  
e | Grid spacing
<table>
<thead>
<tr>
<th>Key</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>d</td>
<td>Grid offset</td>
</tr>
<tr>
<td>h</td>
<td>Hide selected geometry</td>
</tr>
<tr>
<td>i</td>
<td>Isolate selected geometry</td>
</tr>
<tr>
<td>H</td>
<td>Unhide selected geometry</td>
</tr>
<tr>
<td>U</td>
<td>Unhide all geometry</td>
</tr>
<tr>
<td>c</td>
<td>Viewport preferences</td>
</tr>
<tr>
<td>MENU</td>
<td>Viewport menu</td>
</tr>
</tbody>
</table>
Tools

Generally, the behavior of a tool depends on the current mode, the current selection and any coordinates indicated by the user. A tool is active until another tool is activated, until the user deactivates the tool or until a tool that doesn’t require any user interface (e.g. Apply Color) has finishes its task. After e.g. moving a number of selected primitives the same primitives can be moved again just by indicating a new coordinate.

A new set of primitives can also be selected and the tool can be applied to them without having to restart the tool. A tool can be deselected so that no tool is currently active.

XFORM (Transformation tools)

Move

This tool moves the currently selected primitives so that the pivot point of the selection is moved to the new coordinate indicated by pressing the middle mouse button. A new position can be assigned to the pivot point prior to the movement using Ctrl + Middle mouse button.

The user can choose to freeze the X, Y or Z axis to avoid moving in the corresponding directions. An exact movement can also be performed by manually specifying the coordinate in the x, y and z input boxes and clicking the Move button. This way of moving can be relative or absolute depending on if the Relative check box is enabled or not. If the movement was relative and the Move button is clicked again, the same relative movement will be repeated.

Another way of moving the selected primitives is to check the Move To Point check box. This will make each of the selected primitives move to the specified point and can be useful for example when moving several vertices to the same position. Since the movement in this mode will always be absolute, the Relative check box is disabled.

By setting the Snap Movement to a value above zero, the relative movement will be snapped to an even multiple of the specified number.

The move tool can also be controlled by a gizmo. When pressing down the middle mouse button on one of the arrows or planes between the arrows, the selected geometry can be moved in the arrow’s direction or in the plane defined by the arrows.
A convenient feature of the Move tool is the Plant option. This will simply place the selected primitives exactly at the position clicked at, provided that the click hits another existing geometry. This can be useful when e.g. planting trees on terrain, with the pivot at the bottom of the tree geometry and not in its center point.

When planting, one often copies selected primitives to the clipboard, pastes the primitives and then plants them, all in a repeated process. Make sure that the desired pivot point is correct when copying the primitives, since the pivot is saved together with the geometry in the clipboard.

By enabling the Apply Matrix check box, the movement will be applied as a transformation matrix to the selected primitives that support this, instead of modifying the coordinates. Nodes that don't support transformation matrices (vertices, edges, polygons, light points, headers and DOF nodes) will not be affected and a warning will be shown. This type of transformation will be the same as running the Apply Transformation Matrix tool.

See the separate description of Soft Selection below.

**Rotate**

This tool rotates the currently selected primitives around a specified axis. The axis can quickly be chosen as X, Y or Z by using the convenient X, Y and Z buttons. An existing edge can also be selected as rotation axis by clicking the Pick Rotation Edge button and selecting one of the existing edges in the model using the left mouse button.

The current rotation axis is visualized as a yellow line. Note that when picking an edge to define the direction of the rotation axis, it's only the direction and not the position of the rotation axis which is defined. To actually force the rotation axis to be the same as the picked edge, it has to be translated by moving the pivot point to the picked edge’s center using the Edge mode and Shift + Ctrl + Middle mouse button. The pivot point can also be moved to any coordinate on the picked edge by indicating a coordinate either on the grid or snapped to a vertex.

The actual rotation can be performed by clicking and dragging the middle mouse button up and down in the model view. An exact angle (in degrees) can also be specified in the Angle input box in which case the Rotate button must be clicked to confirm the rotation. If the Rotate button is clicked again, a new rotation with the same angle is performed.
By setting the Snap Angle to a value above zero, the relative rotation will be snapped to an even multiple of the specified angle in degrees.

The Rotate Normals check box controls whether or not to rotate the vertex normals to correspond to the vertex position rotation.

The rotate tool can also be controlled by a gizmo. When pressing down the middle mouse button on one of the boxes on the rotation circles, the selected geometry can be rotated around the axis that goes through the center of the rotation circle.

By enabling the Apply Matrix check box, the rotation will be applied as a transformation matrix to the selected primitives that support this, instead of modifying the coordinates. Nodes that don't support transformation matrices (vertices, edges, polygons, light points, headers and DOF nodes) will not be affected and a warning will be shown. This type of transformation will be the same as running the Apply Transformation Matrix tool.

See the separate description of Soft Selection below.

---

Scale

This tool scales the currently selected primitives from the current pivot point. The user can choose to freeze the X, Y or Z axis to avoid scaling in the corresponding direction. The actual scaling can be performed by clicking and dragging the middle mouse button up and down in the model view.

An exact scaling factor (in percent) can also be specified in the Scale input box in which case the Scale button must be clicked to confirm the scaling. If the Scale button is clicked again, a new scaling with the same factor is performed.

By setting the Snap Scale to a value above zero, the relative scale will be snapped to an even multiple of the specified percentage.

The scale tool can also be controlled by a gizmo. When pressing down the middle mouse button on one of the arrows or planes between the arrows, the selected geometry can be scaled in the direction of the arrow or in the plane by dragging the mouse in the arrow's direction or within the plane. When scaling with the plane, the scaling can be uniform in both directions or individual in each direction of the axes that make up the plane. This is controlled by a setting described in the Preferences section.
By enabling the Apply Matrix check box, the scale will be applied as a transformation matrix to the selected primitives that support this, instead of modifying the coordinates. Nodes that don’t support transformation matrices (vertices, edges, polygons, light points, headers and DOF nodes) will not be affected and a warning will be shown. This type of transformation will be the same as running the Apply Transformation Matrix tool.

See the separate description of Soft Selection below.

**Soft Selection**

The Move, Rotate and Scale tools can all use Soft Selection when performing the transformations. Soft Selection is used to smoothly vary the weight of each vertex or primitive. The weight is varied according to a weight function; there are currently six different weight functions:

- Linear
- Sine
- Square Root
- Square
- Cube Root
- Cube

The weight is also bound by a radius. When the distance between the vertex/primitive to the pivot point is greater than this radius, the vertex/primitive will not be affected by the transformation at all. From the pivot point up to the radius, the weight will be calculated according to the weight function.

To activate Soft Selection in the Move, Rotate and Scale tools, check the Activate check box. When Soft Selection is active, the weight of each selected vertex/primitive can be shown by checking the Show Visualization check box. The colors used to visualize the weight of the vertex/primitive range from blue to red, where blue is zero weight and red is full weight.
The weights of the Soft Selection are recalculated when a new selection is made. They are also recalculated when the pivot point is independently moved or any of the Soft Selection parameters are changed.

**Apply Transformation Matrix**

With this tool transformation matrices can be applied, inspected and edited. Transformation matrices can be applied to group nodes, object nodes, LOD nodes, external references, light point system nodes and switch nodes. New transformations can be added by selecting the transformation type and clicking the Add Before or the Add After button.

Transformation matrices can also be applied to polygon and light point nodes, but the transformation matrices are only used to transform replications; without a replication the transformation matrices have no effect. If a group has a transformation matrix and a replication, the transformation matrix is used to transform the replications.

The transformation matrices are applied to the geometry from the top and downwards in the list. The list of transformations can be edited by moving them upwards or downwards in the list with the Move Up and Move Down buttons or removing them with the Remove button. The details of a transformation can be edited when it’s selected. All changes to the transformation matrices must be confirmed by clicking the Apply button.

Note that the details of Rotate And/Or Scale To Point transformations and Put transformations can’t be edited in the current version of Remo 3D.

**Eliminate Transformation Matrices**

This tool eliminates all transformation matrices recursively beneath and including the selected node. Vertex coordinates are transformed accordingly so that the geometry appears the same. Note that external references and instances will not always respond well to this, since it’s impossible to transform their vertices correctly.
CREATE (Creation tools)

Create Node

This tool allows for creation of object, group, LOD, DOF, light point system or switch nodes. The new node is placed beneath the current parent node.

Create Polygon

With this tool the user can create a new polygon. The coordinates of the new vertices are indicated using the middle mouse button. They can be snapped to previous coordinates by using the vertex mode and Shift + Middle mouse button. They can also be snapped to the centers of other primitives, like polygons or edges in the same manner.

When the coordinates for all vertices have been indicated, the polygon is created by clicking the Done button or by pressing the Enter key. The last vertex created can be removed by clicking the Remove Last button.

When clicking the Done button or pressing the Enter key, the vertices might be rotated automatically to a reverse order, to make the polygon face the viewer in the current viewport. This can be changed later by using the Reverse Polygon tool. The polygon is placed beneath the current parent node.

Create Disk

With this tool the user can create a disk polygon. The disk polygon is attached to the current parent node. A center coordinate is indicated with the middle mouse button, as well as a position on the disk border. A third point, called span, must be in the disk plane and is needed to complete the disk. If Align To Grid is enabled, the span point isn’t needed. Instead the disk will be created in the plane defined by the current grid.

The number of segments can be changed before or after indicating the three coordinates. Any of the three coordinates can be changed by clicking their radio buttons and indicating new coordinates. A disk is completed by clicking the Done button or pressing the Enter key.

Create Sphere

With this tool the user can create a sphere. The sphere is made up of a number of polygons put beneath an object node. The object node is attached to the current parent node. A center coordinate is indicated with the middle mouse button, as well as a position on the sphere border.
The number of rows and segments can be changed before or after indicating the two coordinates. Any of the two coordinates can be changed by clicking their radio buttons and indicating new coordinates. A sphere is completed by clicking the Done button or pressing the Enter key.

**Create External Reference**

This tool creates one or more external references. Simply select one or more OpenFlight® files in the file dialog. The general functionality of the file dialog is described in the File Dialogs section. The reference nodes are placed beneath the current parent node.

Note that if the Load External References setting described in the Preferences section is disabled, the model won’t be loaded until it’s enabled in the External References dialog available from the File menu.

When loading a model with external references, the REMO3D_FILE_PATH environment variable is used if the external referenced files aren’t found directly.

**Create Light Point**

This tool creates a string of light points. The coordinates of the new vertices are indicated using the middle mouse button. They can be snapped to previous coordinates by using the vertex mode and Shift + Middle mouse button. They can also be snapped to the centers of other primitives, like polygons or edges in the same manner.

By enabling the Indexed check box, indexed light points will be created instead of ordinary “inline” light points. The indexed light point that will be created will refer the light point palette with the currently active appearance and animation indices which can be picked in the Light Point Palette described in the Light Point Palette section.

When the coordinates for all vertices have been indicated, the light point string is completed by clicking the Done button or pressing the Enter key. The last vertex created can be removed by clicking the Remove Last button. The light point node is placed beneath the current parent node.

**SURFACE (Surface appearance tools)**

**Apply Color**

This tool applies the current color to all polygons in the selected geometry. If vertices or edges are selected, the color is applied to the vertices.
Read Color From Node

This tool reads the color from the polygons in the selected geometry and applies it as the current color. If vertices or edges are selected, the color is read from the vertices. This tool only works if all selected primitives have the same color.

Apply Texture

This tool applies the current texture to all polygons in the selected geometry.

Note that the texture will not be mapped; it’s only the polygon texture index that is modified. For mapping a texture to the geometry, use one of the Map Texture or Apply Texture Mapping tools.

Read Texture From Node

This tool reads the texture from the polygons in the selected geometry and applies it as the current texture. This tool only works if all selected polygons have the same texture.

Note that the texture mapping will not be read from the node, it’s only the polygon texture index that is retrieved.

Apply Texture Mapping

This tool applies the current texture mapping to all polygons in the selected geometry. This will modify the texture mapping index of the polygons and it will also modify the UV coordinates of the vertices according to the new mapping.

Note that the texture index will not be changed, only the texture mapping from the texture mapping palette will be modified.

Read Texture Mapping From Node

This tool reads the texture mapping index (pointing at the texture mapping palette) from the polygons in the selected geometry and applies it as the current texture mapping. This tool only works if all selected polygons have the same texture mapping index.

Apply Material

This tool applies the current material to all polygons in the selected geometry.
Read Material From Node

This tool reads the material from the polygons in the selected geometry and applies it as the current material. This tool only works if all selected polygons have the same material.

Apply Shader

This tool applies the current shader to all polygons in the selected geometry.

Read Shader From Node

This tool reads the shader from the polygons in the selected geometry and applies it as the current shader. This tool only works if all selected polygons have the same shader.

Remove Texture

This tool removes the texture and detail texture, as well as the texture coordinates and texture mappings from all polygons in the selected geometry.

Remove Texture Mapping

This tool removes all texture mapping indices from all polygons in the selected geometry.

Note that it’s only the texture mapping indices that will be set to -1, the UV coordinates of the vertices won’t be modified. This means that the appearance will be similar to what it was before, but the texture mapping will now be controlled only by the UV coordinates, which will imply a different appearance for e.g. four-point mappings. If the texture and mapping should be removed completely, use the Remove Texture tool.

Remove Material

This tool removes the material from all polygons in the selected geometry.

Remove Shader

This tool removes the shader from all polygons in the selected geometry.
TEXMAP (Texture mapping tools)

Map Texture

With this tool, a texture can be mapped on to a number of selected polygons or all polygons beneath the selected sub-trees. The currently chosen texture is applied to the polygons and texture coordinates are applied to their vertices in different ways depending on if planar, cylindrical or spherical mapping is chosen. If the user chooses to add this mapping to the texture mapping palette, an index for the new mapping created will also be applied to the polygons.

The Map Texture tool supports multi-texturing. To apply a base texture, layer 0 should be used. The other seven layers are used for multi-texturing. When done with one layer the button Texture New Layer should be clicked. If the Texture New Layer button isn’t clicked and a new layer is chosen, the previous layer will be restored to its previous texture.

The texture mapping is controlled by indicating up to four geometry coordinates with the middle mouse button. With the First, Second, Third and Fourth radio buttons, the user can switch between these geometry coordinates. As a default, they are automatically activated one by one, starting with the first. The geometry coordinates have different impact on the texture mapping depending on the mapping type and if adding to palette is chosen or not. This is described in more detail in the following sections. For planar mappings (3-point and 4-point), each geometry coordinate corresponds to a texture coordinate \((u, v)\) which can be manually modified by the user, either by entering the values in the text boxes or by clicking and dragging the corresponding points in the texture image below. By clicking the Full View button, a window with a full size view of the texture image is opened, also allowing for modifying the \((u, v)\) coordinates in the same way.

These \((u, v)\) coordinates are horizontal \((u)\) and vertical \((v)\) positions in the texture image and range from 0 to 1. A \(u\)-value of 0 means the left-most position while a \(u\)-value of 1 means the right-most position. Likewise, a \(v\)-value of 0 means the bottom position and a \(v\)-value of 1 means the top position. If new geometry is selected, the corresponding texture coordinates will be reset to their default values, unless the Freeze check box is enabled. Both the geometry coordinates and the corresponding texture coordinates will be reset if a new texture is chosen.

If planar (3-point or 4-point) mapping is chosen, the first, second and third geometry coordinates are lower left, lower right and upper right points of the virtual plane (at least the default corresponding texture coordinates imply these corner points). The texture coordinates applied to the vertices will be interpolated between the texture coordinates to which these coordinates correspond. In the case of 4-point mapping, a fourth geometry point is added, representing the upper left point of the mapping. This fourth geometry point will actually be projected into the plane described by the first, second and third points. Its corresponding texture coordinate will contribute to the mapping and the result could be a non-linear mapping that is useful for non-orthogonal texture images. Note that a 4-point texture mapping needs to be added to the texture mapping palette in order to work in the expected way. When adding to palette, the OpenGL TexGen functionality will be used, resulting in a continuous non-linear mapping. Therefore, adding to palette will be the default when choosing 4-point mapping. It’s still
possible to use 4-point mapping without adding it to the texture mapping palette and in some cases that might be sufficient.

Cylindrical and spherical mappings behave differently depending on whether the Add To Palette check box is enabled or not, as described in the following sections. This is due to the need for supporting the OpenFlight® texture mapping format when adding to palette is chosen. If we’re not adding the texture mapping palette, a more advanced texture mapping is allowed.

**Mapping texture and adding to palette**

For cylindrical mapping when adding to palette, the first geometry coordinate is the bottom center of the virtual cylinder. The second geometry coordinate is the top center of the virtual cylinder. Repetitions is the number of times the texture width is wrapped around the cylinder.

For spherical mapping when adding to palette, the one and only geometry coordinate is the virtual sphere origin. Repetitions is the number of times the texture is wrapped around the sphere, both horizontally and vertically.

**Mapping texture without adding to palette**

For cylindrical mapping when not adding to palette, the first geometry coordinate is the virtual cylinder origin. The second geometry coordinate should be placed at the virtual cylinder surface and the line between the first and the second coordinate defines the zero-line for the horizontal angle. The third geometry coordinate should be placed directly above the second geometry coordinate and thereby defines the height of the texture mapping. The three geometry coordinates together define the position and orientation of the virtual cylinder. Horizontal Repetitions is the number of times the texture width is wrapped around the cylinder.

For spherical mapping when not adding to palette, the first geometry coordinate is the virtual sphere origin. The second geometry coordinate should be placed at the virtual sphere surface and the line between the first and the second coordinate defines the zero-line for the horizontal angle. The third geometry coordinate should be placed directly above the second geometry coordinate and the line between the second and the third geometry coordinates defines the zero-line for the vertical angle. The three geometry coordinates together define the position and orientation of the virtual sphere. Horizontal Repetitions is the number of times the texture width is wrapped around the sphere. Vertical Repetitions is the number of times the texture height is wrapped around the sphere.

**Move Texture Map**

This tool moves the texture on the currently selected primitives. The texture can be moved using the pivot point of the selection relative to a new coordinate indicated by pressing the middle mouse button. A new position can be assigned to the pivot point prior to the movement using Ctrl + Middle mouse button.
The user can choose to freeze the X, Y or Z axis to avoid moving the texture in the corresponding directions. An exact movement can also be performed by manually specifying the coordinate in the x, y and z input boxes and clicking the Move button. If the Move button is clicked again, the same relative movement will be repeated.

The move texture map tool can also be controlled by a gizmo. When pressing down the middle mouse button on one of the arrows or planes between the arrows, the texture on the selected geometry can be moved in the arrow’s direction or in the plane defined by the arrows.

To control for which multi-texture layer the texture map is to be transformed, the Texture Layer combo-box can be used.

Note that all texture mapping indices on the polygons in the selected geometry will be reset, since changing the mappings of the texture mapping palette is not the intention of this tool.

**Rotate Texture Map**

This tool rotates the texture on currently selected primitives around a specified axis. The axis can quickly be chosen as X, Y or Z by using the convenient X, Y and Z buttons. An existing edge can also be selected as rotation axis by clicking the Pick Rotation Edge button and selecting one of the existing edges in the model using the left mouse button.

The current rotation axis is visualized as a yellow line. Note that when picking an edge to define the direction of the rotation axis, it’s only the direction and not the position of the rotation axis which is defined. To actually force the rotation axis to be the same as the picked edge, it has to be translated by moving the pivot point to the picked edge’s center using the Edge mode and Shift + Ctrl + Middle mouse button. The pivot point can also be moved to any coordinate on the picked edge by indicating a coordinate either on the grid or snapped to a vertex.

The actual texture rotation can be performed by clicking and dragging the middle mouse button up and down in the model view. An exact angle (in degrees) can also be specified in the Angle input box in which case the Rotate button must be clicked to confirm the rotation. If the Rotate button is clicked again, a new rotation with the same angle is performed.

The rotate texture map tool can also be controlled by a gizmo. When pressing down the middle mouse button on one of the boxes on the rotation circles, the texture on the selected geometry can be rotated around the axis that goes through the center of the rotation circle.

To control for which multi-texture layer the texture map is to be transformed, the Texture Layer combo-box can be used.

Note that all texture mapping indices on the polygons in the selected geometry will be reset, since changing the mappings of the texture mapping palette is not the intention of this tool.
Scale Texture Map

This tool scales the texture on the currently selected primitives from the current pivot point. The user can choose to freeze the X, Y or Z axis to avoid scaling in the corresponding direction. The actual scaling can be performed by clicking and dragging the middle mouse button up and down in the model view.

An exact scaling factor (in percent) can also be specified in the Scale input box in which case the Scale button must be clicked to confirm the scaling. If the Scale button is clicked again, a new scaling with the same factor is performed.

The scale texture map tool can also be controlled by a gizmo. When pressing down the middle mouse button on one of the arrows or planes between the arrows, the texture on the selected geometry can be scaled in the direction of the arrow or in the plane by dragging the mouse in the arrow’s direction or within the plane. When scaling with the plane, the scaling can be uniform in both directions or individual in each direction of the axes that make up the plane. This is controlled by a setting described in the Preferences section.

To control for which multi-texture layer the texture map is to be transformed, the Texture Layer combo-box can be used.

Note that all texture mapping indices on the polygons in the selected geometry will be reset, since changing the mappings of the texture mapping palette is not the intention of this tool.
Modify UV

This tool offers an alternative way of modifying an existing texture mapping. Choose the multi-texture layer and press the Modify UV button to present the texture mapping of the currently selected geometry, projected into the current texture space. This is only possible when the selected geometry has exactly one texture mapped.

In the Modify UV window, left-clicking and dragging vertices will result in modifying their texture coordinates correspondingly. Each release of the mouse button and each click on the mirror icons (see below) will produce an undoable command. Undo/redo of Modify UV commands can be performed with Ctrl+Z and Ctrl+Y respectively. Note that any palette texture mapping will be inactivated for the modified polygons.

Vertices can be selected with a box selection as well, by clicking and dragging the left mouse button. Selected vertices are marked in orange and surrounded by a orange/black rectangle with handles in the corners and edge centers (more about these handles below). Vertices can be added and removed from the selection by holding Shift and Alt respectively.

The entire texture perimeter, i.e. the [0, 1] texture coordinate interval in both directions, is marked with an green/black rectangle.

The view can be moved by clicking and dragging Ctrl + right mouse button, and zoomed by clicking and dragging Shift + right mouse button or using the mouse wheel, all in the same manner as in the viewport.

To be able to separately modify neighboring vertices whose texture coordinates coincide, there is a Shrink button available. This can also be activated with the 's' key.
There are buttons for Fit Selected and Fit Texture, that will zoom and pan the current view to the selected geometry and the entire texture (the green/black rectangle) respectively. These can also be activated with the ‘v’ and space keys respectively.

To be able to view the result in the model view (move the Modify UV window to see the model view behind), there is a Fit Viewport button that will pan, rotate and zoom the model view to the selected geometry. If certain vertices are selected and marked orange in the Modify UV view, the model view will be focused on only these corresponding vertices and they will also be orange in the model view.

When multiple vertices have been selected so that they are marked by the orange/black rectangle, they can be scaled or rotated in the UV space by left-clicking and dragging the corner and edge handles. Scale and rotate mode can be switched with the icons in the top left corner of the Modify UV window, scale mode is default.

The selected vertices can also be mirrored horizontally or vertically in the UV space by clicking one of the two mirror icons to the right of the scale/rotate mode icons.

The Freeze U and Freeze V check boxes in the top right corner will cause the U and V directions to freeze when moving or scaling the vertices in the UV space.

The snap check box is enabled by default and will make selected vertices with identical texture coordinates snap to other vertices when moved.

The Modify UV dialog can be closed with the Close button, the cross button in the upper right corner or by pressing Escape.

**MODIFY (Modification tools)**

**Mirror**

This tool mirrors the selected geometry with respect to the plane described by the grid in the focused viewport. Note that matrix transformed geometry will be mirrored as if it wasn’t transformed.

**Reverse Polygon**

This tool reverses a number of selected polygons or all polygons beneath the selected sub-trees. The side from which the vertices of a polygon are ordered counter-clockwise is defined as the upside of the polygon. With this in mind, this tool can be used to define the upsides and downsides of polygons.
Slice

This tool allows the user to slice the selected geometry along the grid plane or along planes orthogonal to the grid lines. Note that slicing will be performed as if the grid would be infinite. Normals and texture coordinates are interpolated for new vertices created by the slice operation. After the slice operation, all faces that were actually sliced are automatically selected.

Split/Triangulate

With this tool, polygons and edges can be split in different ways. When selecting an edge and activating this tool, the edge is split in two halves. If a polygon is selected, the tool will triangulate the polygon into a number of triangles. This works for concave as well as convex polygons. If two vertices in the same polygon are selected, the polygon will be split along a new edge between the vertices. Normals and texture coordinates are interpolated for new vertices created by the split operation.

Extrude

With this tool, selected geometry can be extruded, i.e. raised to a certain level in a certain direction (or along a certain extrusion axis). For example, it can be used to create a box out of a quad polygon. An extruded polygon is moved along the extrusion axis to the desired level and new polygons (walls) are created along its borders. The extruding operation is defined by a number of parameters described below. The actual extrusion (with an implicated undoable extrusion command) is performed when a position is indicated or the Apply Parameters button is clicked. If a position is indicated any changed parameters will also be applied.

The extrusion axis can be defined by indicating a position in the model view. This way of defining the axis is chosen by selecting the ‘Free’ radio button. The axis is then defined as the vector from each selected polygon to the indicated position. The ‘Shared Axis’ can be used here and then the axis for extrusion will be an axis from the pivot point to the indicated point.

The extrusion axis can also be defined by the polygon normal (chosen by selecting the ‘Polygon Normal’ radio button), in which case the extrusion level can be specified by typing a distance (positive or negative) in the ‘Height’ field or indicating a position in the model view by pressing the middle mouse button. Note that the indicated position is not necessarily the actual position to which the extrusion is made, since the polygon normal is used as extrusion axis. Instead, the indicated position is projected onto the polygon normal.

The extrusion axis can also be defined only by indicating a position in the model view. This way of defining the axis is chosen by selecting the ‘Free’ radio button. The axis is then defined as the vector from each selected polygon to the indicated position. The ‘Shared Axis’ can be used here as well.
For both types of extrusions, ‘Keep Floor’ controls whether floor polygons are to be created in the original polygon positions or not. The floor polygons are essentially reversed copied of the original polygons selected for extrusion.

The extrude tool can also be controlled by a gizmo. When pressing down the middle mouse button on one of the arrows or planes between the arrows, the selected geometry can be extruded in the arrow’s direction or in the plane defined by the arrows.

**Revolve**

With this tool, selected geometry can be revolved about a specified axis. The axis can quickly be chosen as X, Y or Z by using the convenient X, Y and Z buttons. An existing edge can also be selected to define the direction of the revolve axis by clicking the Pick Revolve Edge button and selecting one of the existing edges in the model using the left mouse button.

The current revolve axis is visualized as a yellow line. Note that when picking an edge to define the direction of the revolve axis, it’s only the direction and not the position of the revolve axis which is defined. To actually force the revolve axis to be the same as the picked edge, it has to be translated by moving the pivot point to the picked edge’s center using the Edge mode and Shift + Ctrl + Middle mouse button. The pivot point can also be moved to any coordinate on the picked edge by indicating a coordinate either on the grid or snapped to a vertex.

The revolving process is performed by specifying a total revolve angle (in degrees) and a number of steps in the Angle and Steps input boxes and clicking the Revolve button. If the Revolve button is clicked again, a new revolving process with the same angle will be performed.

**Subdivide**

This tool allows the user to increase complexity of the selected geometry and create smoother looking objects. Each polygon will be subdivided into several new polygons. The subdivision uses the Catmull-Clark algorithm. The new subdivided geometry will be shown in orange together with the original geometry. The resulting polygons from the subdivision will always be quads.

To control the subdivision, the number of iterations can be specified.

If the Subdivide Borders check box is enabled, the borders of the geometry will also be subdivided; otherwise they will be unmodified.

To create the new subdivided geometry, press the Done button.
This tool allows the user to reduce complexity of the selected geometry. The reduction of the geometry is performed by an edge collapse method. To be able to simplify the geometry, all polygons in the selection need to be triangles (the Split/Triangulate tool can be used to split non-triangular polygons into triangles). The original geometry will be shown while the simplification is performed to make it easier to see the differences.

In the selected geometry faces that are siblings will be grouped and simplified together, e.g. two different object nodes will be simplified independently of each other.

There are two parts of the selected geometry that can be simplified, the boundary of the selection and the interior. The level of reduction for the interior is controlled by the Simplify Level slider and for the boundary the Boundary Simplify Level slider is used. Both sliders range from 0 to 100 percent reduction, where 0 means no reduction at all and 100 percent means maximum reduction. To control the order in which the edges are collapsed, the Edge Weight Ratio slider can be used for the interior and the Boundary Edge Weight Ratio can be used for the boundary.

The Edge Weight Ratio slider goes from Edge Length to Normal Deviation, where Edge Length will remove the shortest edges first and Normal Deviation will remove the edges that will cause the least deviation of the normals on the affected polygons. It is possible to place the slider somewhere between Edge Length and Normal Deviation, this will cause the edge weight to be a mix between the Edge Length and the Normal Deviation. One way of looking at the different methods of calculating edge weights is that Edge Length will remove detail while Normal Deviation will try to preserve detail.

The Boundary Edge Weight Ratio Slider ranges from Edge Length to Edge Angles, where Edge Length will remove the shortest edges first and Edge Angles will remove the edges with the smallest angle deviation toward the adjacent edges. It is possible to place the slider somewhere between Edge Length and Edge Angles, this will cause the edge weight to be a mix between the Edge Length and the Edge Angles.

It is possible to control the new vertex position when an interior edge is collapsed, this is done by checking the radio buttons in the Vertex Weight box. Equal implies that all vertices have equal weight and the new vertex position will be in the middle of the edge that is collapsed. Normal Deviation will consider the change of the adjacent polygons when calculating the weight of the vertices, so the new vertex position can be shifted toward one of the ends of the edge. When an interior edge that has one of its vertices on the boundary is collapsed, the new vertex position will be the same as the previous vertex on the boundary. When a boundary edge is collapsed, the new vertex position is the same as the vertex with the largest weight in the collapsed edge.

There are three different ways of coloring the simplified geometry, this is controlled by checking the radio buttons in the Coloring box.

- Uniform will set the same color on all edges.
- Vertex Weight will color the vertices according to the vertex weight. Interior vertices will be colored from red to yellow, where the red vertices have the smallest weights. Boundary vertices will be colored from blue to green, where the blue vertices have the smallest weights.
• Edge Weight will color the edges according to the edge weight. Interior edges will be colored from red to yellow, where the red edges have the smallest weights. Boundary edges will be colored from blue to green, where the blue edges have the smallest weights.

In the Number Of Faces output box the current number of polygons will be shown.

To create the new simplified geometry, press the Done button.

**Merge Polygons**

This tool merges all selected polygons so that each collection of connected polygons is exchanged for one polygon made up of the outer edges of the collection.

Note that this tool only works for polygons located in the same plane and sharing the same edges, measured with the tolerance described in the Preferences section. Also note that polygons with different attributes, such as different textures, materials or colors, will not be merged with this tool.

**Quantify**

This tool quantifies all vertex positions in the selected geometry with a specified quantification value. Quantification simply means to change the x, y and z of a vertex to the nearest value divisible by the specified quantification value. The quantification is performed when the Quantify button is clicked. This can for example be useful for snapping free-hand indicated coordinates to a specified accuracy. It can also be used for fixing erroneous coordinate inaccuracies in a model.

Note that some of the Remo 3D tools can introduce inaccuracies in vertex positions. This problem will be addressed in future versions of Remo 3D, but until then the Quantify tool can be used to fix these inaccuracies.

**Shade**

With this tool, normals are calculated for the vertices in the selected geometry. If flat shading is selected, the normals will be set to their polygon normals, creating a flat appearance when lighting is applied to the geometry. If smooth shading is selected, the normals will be interpolated, creating a smooth appearance.

Note that the polygons must be set to use the Vertex Color Lit or Polygon Color Lit light modes in order to make the lighting work with the calculated normals. This can be done with the Attribute tool or directly in this tool with the convenient Update Light Mode check box and the Light Mode pull-down menu.

When using smooth shading, an input box for entering a crease angle will be activated. The crease angle is used to specify a threshold angle for when the angle between polygon normals is too large to be smoothed. E.g. if you specify a crease angle of 90°...
degrees all angles between polygon normals that are less than 90 degrees will be smoothed.

Also note that when smooth shading is selected, different results are produced when selecting adjacent geometries and shading them at the same time, than when selecting and shading them one at a time. This is because the interpolation will take both geometries into consideration.

The actual creation of normals is performed when clicking the Shade button.

**ATTR (Attribute tools)**

**Apply Attributes**

With this tool, attributes for the selected primitives can be viewed and modified. Note that attributes for primitives of different types can't be modified at once. Attributes not having the same value in all the selected primitives are indicated as gray.

The new attributes are confirmed by clicking the Apply button. The modified attributes, including those that were not the same from the beginning, are assigned to all selected primitives. The unmodified attributes are completely left alone, even those that differ among the primitives. This allows the user to modify a certain attribute for a number of primitives while leaving the other attributes unchanged.

If the current model is in indexed color mode (RGB Mode is unchecked in the header node attributes), color for polygons and vertices can be changed by editing the color index (0 - 1023) and intensity (0 - 127) or by clicking the color box next to the index and intensity attributes. If the color box is clicked, the corresponding palette will emerge and let the user select the desired color. The same applies for polygon textures.

If the model is in RGB mode, the color of polygons and vertices can be changed by either editing the red, green and blue values (0 - 255) or by clicking the color box next to the RGB values. In that case, a color chooser dialog will appear and let the user choose a new RGB color.

When unchecking the RGB mode header node attribute and clicking the Apply button, each polygon and vertex color will be translated to a color index and intensity which will give the nearest corresponding RGB color.

Note that not all attributes are considered when visualizing the geometry in Remo 3D. However, all attributes will be saved in the OpenFlight® file.

The attribute tool is activated either by clicking its icon or by double-clicking any primitive in the model view or in the tree view. The following is a description of the attributes of each primitive type:

**Header Attributes**

The tables below describe the header attributes.
The General tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The header node name.</td>
</tr>
<tr>
<td>Coordinate Units</td>
<td>The unit used for this model’s coordinates (meters, kilometers, feet, inches or nautical miles).</td>
</tr>
<tr>
<td>Save Vertex Normals</td>
<td>Enabled if vertex normals should be saved in the OpenFlight® file (not used in Remo 3D, always saved).</td>
</tr>
<tr>
<td>New Textured Polygon White</td>
<td>Enabled if created polygons with textures are to be rendered white.</td>
</tr>
<tr>
<td>RGB Mode</td>
<td>Enabled if polygon and vertex colors are defined as RGB instead of indexed.</td>
</tr>
<tr>
<td>Format Version</td>
<td>The OpenFlight® format version. This will always be 16.2 for files created in Remo 3D v2.8.</td>
</tr>
<tr>
<td>Database Version</td>
<td>A version number for the current model. This number will increase each time the file is saved.</td>
</tr>
<tr>
<td>Revision Date</td>
<td>The date and time for the last time this file was saved.</td>
</tr>
<tr>
<td>Next IDs – Polygon</td>
<td>The ID of the next polygon to be created.</td>
</tr>
<tr>
<td>Next IDs – Object</td>
<td>The ID of the next object to be created.</td>
</tr>
<tr>
<td>Next IDs – Group</td>
<td>The ID of the next group to be created.</td>
</tr>
<tr>
<td>Next IDs – LOD</td>
<td>The ID of the next LOD to be created.</td>
</tr>
<tr>
<td>Next IDs – DOF</td>
<td>The ID of the next DOF to be created.</td>
</tr>
<tr>
<td>Next IDs – Switch</td>
<td>The ID of the next switch to be created.</td>
</tr>
<tr>
<td>Next IDs – Light Point</td>
<td>The ID of the next light point to be created.</td>
</tr>
<tr>
<td>Next IDs – Light Source</td>
<td>The ID of the next light source to be created (not used by Remo 3D).</td>
</tr>
</tbody>
</table>

The Projection tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Projection Type</td>
<td>The type of projection used in this model (Flat Earth, Trapezoidal, Round Earth, Lambert, UTM, Geodetic or Geocentric).</td>
</tr>
<tr>
<td>Earth Ellipsoid Model</td>
<td>The ellipsoid used in this model (WGS 1984, WGS 1972, Bessel, Clarke 1866 or NAD 1927).</td>
</tr>
<tr>
<td>SW Corner Latitude</td>
<td>The south-west corner latitude of the area covered by this model (decimal degrees).</td>
</tr>
<tr>
<td>SW Corner Longitude</td>
<td>The south-west corner longitude of the area covered by this model (decimal degrees).</td>
</tr>
<tr>
<td>NE Corner Latitude</td>
<td>The north-east corner latitude of the area covered by this model (decimal degrees).</td>
</tr>
<tr>
<td>NE Corner Longitude</td>
<td>The north-east corner longitude of the area covered by this model (decimal degrees).</td>
</tr>
</tbody>
</table>
### Group Attributes

The table below describes the group attributes.

*Note that the visibility of a group node’s children can be modified manually by ordinary hiding and showing of the child nodes. However, applying attributes for a group with a Forward or Backward sequence animation type will reset the visibility of the group’s children so that only the first or last child is visible.*

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Replications</td>
<td>The number of times to replicate this group. A value of 0 is equal to no replications and the group will be visualized as normal. The replications are transformed by a transformation matrix on the group.</td>
</tr>
<tr>
<td>Sequence</td>
<td>The type of sequence animation used for this group’s children (None, Forward or Backward).</td>
</tr>
<tr>
<td>Loop Count</td>
<td>The number of sequence animation loops, 0 means looping forever.</td>
</tr>
<tr>
<td>Loop Duration</td>
<td>The duration of each loop (seconds), 0 means as fast as possible.</td>
</tr>
<tr>
<td>Last Frame Duration</td>
<td>The duration of the last loop (seconds), 0 means the same duration as previous loops.</td>
</tr>
<tr>
<td>Swing</td>
<td>Enabled if sequence animation should swing back in the opposite direction after playing the last loop (which is not repeated).</td>
</tr>
</tbody>
</table>

### Object Attributes

The table below describes the object attributes.
<table>
<thead>
<tr>
<th>Name</th>
<th>The object node name.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transparency</td>
<td>The transparency value of the geometry beneath this object node (0.0 - 1.0).</td>
</tr>
<tr>
<td>Flat Shaded</td>
<td>Enabled if the geometry beneath this object node is to be flat shaded (not visualized in Remo 3D).</td>
</tr>
<tr>
<td>Shadow</td>
<td>Enabled if the geometry beneath this object node defines the shadow object of the group above.</td>
</tr>
<tr>
<td>Illuminated</td>
<td>Enabled if the geometry below this object node can be illuminated (not used in Remo 3D, defined by polygons).</td>
</tr>
<tr>
<td>Display In Daylight</td>
<td>Enabled if the geometry below this object node will be displayed in daylight.</td>
</tr>
<tr>
<td>Display At Dusk</td>
<td>Enabled if the geometry below this object node will be displayed at dusk.</td>
</tr>
<tr>
<td>Display At Night</td>
<td>Enabled if the geometry below this object node will be displayed at night.</td>
</tr>
</tbody>
</table>

**Polygon Attributes**

The tables below describe the polygon attributes.

The General tab contains the following attributes:

<table>
<thead>
<tr>
<th>Name</th>
<th>The polygon node name.</th>
</tr>
</thead>
<tbody>
<tr>
<td>IR Color Code</td>
<td>A color code for infra-red visualization in the receiving system.</td>
</tr>
<tr>
<td>IR Material Code</td>
<td>A material code for infra-red classification in the receiving system.</td>
</tr>
<tr>
<td>Feature ID</td>
<td>A feature identification code.</td>
</tr>
<tr>
<td>Surface Material Code</td>
<td>A classification code defining the material of this polygon.</td>
</tr>
<tr>
<td>Replications</td>
<td>The number of times to replicate this polygon. A value of 0 is equal to no replications and the polygon will be visualized as normal. The replications are transformed by a transformation matrix on the polygon.</td>
</tr>
<tr>
<td>Terrain</td>
<td>Enabled if this polygon is defined as terrain.</td>
</tr>
<tr>
<td>Footprint</td>
<td>Enabled if this polygon is defined as the footprint of the model (e.g. used for connecting the model to the terrain).</td>
</tr>
</tbody>
</table>

The Drawing tab contains the following attributes:

<p>| Draw Mode | Defines how this polygon is drawn (Solid |</p>
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>One Side, Solid Both Sides, Wireframe closed or Wireframe unclosed)</td>
<td>Light Mode Defines how this polygon is lit (Polygon Color (Flat), Vertex Color (Gouraud), Polygon Color Lit or Vertex Color Lit).</td>
</tr>
<tr>
<td>Defines the billboard and alpha behavior of this polygon (Fixed/No Alpha, Fixed/Alpha, Axis/Alpha or Point/Alpha).</td>
<td>Billboard/Alpha</td>
</tr>
<tr>
<td>The line style index of this polygon (not used in Remo 3D).</td>
<td>Line Style</td>
</tr>
<tr>
<td>The transparency of this polygon (0.0 – 1.0).</td>
<td>Transparency</td>
</tr>
<tr>
<td>Enabled if this polygon is to be rendered as white if it's textured.</td>
<td>Textured Polygon White</td>
</tr>
<tr>
<td>Enabled if this polygon is to be hidden.</td>
<td>Hidden</td>
</tr>
</tbody>
</table>

The Color tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>The color palette index of the primary color (0 – 1023, -1 if none). Disabled if RGB color mode is used.</td>
<td>Primary Color – Index</td>
</tr>
<tr>
<td>The intensity of the primary color (0 – 127). Disabled if RGB color mode is used.</td>
<td>Primary Color – Intensity</td>
</tr>
<tr>
<td>The red value of the primary color (0 – 255). Disabled if indexed color mode is used.</td>
<td>Primary Color – Red</td>
</tr>
<tr>
<td>The green value of the primary color (0 – 255). Disabled if indexed color mode is used.</td>
<td>Primary Color – Green</td>
</tr>
<tr>
<td>The blue value of the primary color (0 – 255). Disabled if indexed color mode is used.</td>
<td>Primary Color – Blue</td>
</tr>
<tr>
<td>The color palette index of the secondary color (0 – 1023, -1 if none). Disabled if RGB color mode is used. Secondary color isn’t visualized in Remo 3D.</td>
<td>Secondary Color – Index</td>
</tr>
<tr>
<td>The intensity of the secondary color (0 – 127). Disabled if RGB color mode is used. Secondary color isn’t visualized in Remo 3D.</td>
<td>Secondary Color – Intensity</td>
</tr>
<tr>
<td>The red value of the secondary color (0 – 255). Disabled if indexed color mode is used. Secondary color isn’t visualized in Remo 3D.</td>
<td>Secondary Color – Red</td>
</tr>
<tr>
<td>The green value of the secondary color (0 – 255). Disabled if indexed color mode is used. Secondary color isn’t visualized in Remo 3D.</td>
<td>Secondary Color – Green</td>
</tr>
<tr>
<td>The blue value of the secondary color (0 – 255). Disabled if indexed color mode is used. Secondary color isn’t visualized in Remo 3D.</td>
<td>Secondary Color – Blue</td>
</tr>
</tbody>
</table>
The Surface tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Texture – Layer</td>
<td>The multi-texture layer for which the attributes below are modified (Layer 0 – Layer 7). Layer 0 is the base texture layer.</td>
</tr>
<tr>
<td>Texture – Index</td>
<td>The texture palette index of this layer’s texture (-1 if not used).</td>
</tr>
<tr>
<td>Texture – Multi-texture Effect</td>
<td>A multi-texture effect index (not visualized in Remo 3D).</td>
</tr>
<tr>
<td>Texture – Multi-texture Data</td>
<td>A multi-texture data number (not visualized in Remo 3D).</td>
</tr>
<tr>
<td>Texture – Texture Mapping Index</td>
<td>The texture mapping index of this layer’s texture (-1 if not used). Note that a texture can be mapped to a polygon either by the texture coordinates of the polygon’s vertices, or by this index into the texture mapping palette.</td>
</tr>
<tr>
<td>Detail Texture – Index</td>
<td>The texture palette index of this polygon’s detail texture (-1 if not used). Note that detail textures are normally only supported on SGI® hardware. They are not visualized in Remo 3D.</td>
</tr>
<tr>
<td>Material – Index</td>
<td>The material palette index of this polygon’s material (-1 if not used).</td>
</tr>
<tr>
<td>Shader – Index</td>
<td>The shader palette index of this polygon’s shader (-1 if not used).</td>
</tr>
</tbody>
</table>

**Light Point Attributes**

This section describes the attributes of “ordinary” inline light points. Indexed light point attributes are described in the Indexed Light Point Attributes section.

Note that the color of each light point vertex is defined by the vertex color. The light point attributes only define the back color of a bidirectional light point.

Also note that the visualization of a light point in Remo 3D is only based upon the light point vertex color, the minimum and maximum pixel sizes, the directional type, the lobe horizontal and vertical angles and the lobe roll angle. Other attributes are applied to the light point in the OpenFlight® file but won’t affect the visualization in Remo 3D.

The tables below describe the inline light point attributes.

The General tab contains the following attributes:
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The light point node name.</td>
</tr>
<tr>
<td>Display Mode</td>
<td>Defines how this light point is displayed (Raster, Calligraphic or Either).</td>
</tr>
<tr>
<td>Feature ID</td>
<td>A feature identification code.</td>
</tr>
<tr>
<td>Surface Material Code</td>
<td>A classification code defining the material of this light point.</td>
</tr>
<tr>
<td>Front Intensity</td>
<td>Intensity scalar for front color.</td>
</tr>
<tr>
<td>Back Intensity</td>
<td>Intensity scalar for back color.</td>
</tr>
<tr>
<td>Replications</td>
<td>The number of times to replicate this light point. A value of 0 is equal to</td>
</tr>
<tr>
<td></td>
<td>no replications and the light point will be visualized as normal. The</td>
</tr>
<tr>
<td></td>
<td>replications are transformed by a transformation matrix on the light point.</td>
</tr>
<tr>
<td>Back Color</td>
<td>Enabled if back color is used for bidirectional light points.</td>
</tr>
<tr>
<td>Back Color – Index</td>
<td>The color palette index of the back color (0 – 1023, -1 if none). This</td>
</tr>
<tr>
<td></td>
<td>color is always indexed, even if the model is in RGB mode.</td>
</tr>
<tr>
<td>Back Color – Intensity</td>
<td>The intensity of the back color (0 – 127). This color is always indexed,</td>
</tr>
<tr>
<td></td>
<td>even if the model is in RGB mode.</td>
</tr>
<tr>
<td>Display In Daylight</td>
<td>Enabled if this light point will be displayed in daylight.</td>
</tr>
<tr>
<td>Display At Dusk</td>
<td>Enabled if this light point will be displayed at dusk.</td>
</tr>
<tr>
<td>Display At Night</td>
<td>Enabled if this light point will be displayed at night.</td>
</tr>
</tbody>
</table>

The Direction tab contains the following attributes:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Directional Type</td>
<td>Defines the directions of this light point’s emitting light (Omnidirectional,</td>
</tr>
<tr>
<td></td>
<td>Unidirectional or Bidirectional).</td>
</tr>
<tr>
<td>Horizontal Lobe Angle</td>
<td>The horizontal angle of the light lobe (degrees).</td>
</tr>
<tr>
<td>Vertical Lobe Angle</td>
<td>The vertical angle of the light lobe (degrees).</td>
</tr>
<tr>
<td>Lobe Roll Angle</td>
<td>The roll angle of the light lobe (degrees).</td>
</tr>
<tr>
<td>Falloff Exponent</td>
<td>Exponential value controlling the falloff between center and edge of the</td>
</tr>
<tr>
<td></td>
<td>light lobe. Linear falloff = 1.0.</td>
</tr>
<tr>
<td>Ambient Intensity</td>
<td>The ambient intensity of the light point when viewed off-axis.</td>
</tr>
</tbody>
</table>
The Anim (animation) tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Period</td>
<td>The animation period (seconds).</td>
</tr>
<tr>
<td>Phase Delay</td>
<td>The animation phase delay (seconds).</td>
</tr>
<tr>
<td>Enabled Period</td>
<td>The part of the animation period where the light point is switched on (seconds).</td>
</tr>
<tr>
<td>Rotation Axis – i</td>
<td>The i (x) value of the rotation axis.</td>
</tr>
<tr>
<td>Rotation Axis – j</td>
<td>The j (y) value of the rotation axis.</td>
</tr>
<tr>
<td>Rotation Axis – k</td>
<td>The k (z) value of the rotation axis.</td>
</tr>
<tr>
<td>Flashing</td>
<td>Enabled if the light point is flashing.</td>
</tr>
<tr>
<td>Rotating</td>
<td>Enabled if the light point is rotating.</td>
</tr>
<tr>
<td>Counter Clockwise</td>
<td>Enabled if the light point rotation is defined counter-clockwise.</td>
</tr>
</tbody>
</table>

The Callig (calligraphic) tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Significance</td>
<td>Priority used by raster/calligraphic hybrid systems for deciding which light points that should be switched to raster mode at high visualization loads (0.0 – 1.0).</td>
</tr>
<tr>
<td>Drawing Order</td>
<td>Calligraphic draw order index.</td>
</tr>
<tr>
<td>Minimum Defocus</td>
<td>Lower calligraphic defocus limit (0.0 – 1.0).</td>
</tr>
<tr>
<td>Maximum Defocus</td>
<td>Upper calligraphic defocus limit (0.0 – 1.0).</td>
</tr>
<tr>
<td>Quality</td>
<td>Calligraphic quality (Low, Medium or High).</td>
</tr>
<tr>
<td>Proximity Occulting</td>
<td>Enabled if the intensity of adjacent light points are to be decreased when viewed from a distance.</td>
</tr>
</tbody>
</table>

The Misc (miscellaneous) tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range Mode</td>
<td>Method for calculating the distance between the viewer and the light point (Depth Buffer or Slant Range).</td>
</tr>
<tr>
<td>Randomize Intensity</td>
<td>The random intensity level (Off, Low, Medium or High).</td>
</tr>
<tr>
<td>Fog Scale</td>
<td>The level of light point visibility through fog. Values above 0.0 imply punching through the fog.</td>
</tr>
<tr>
<td>Fog Intensity</td>
<td>Intensity factor controlling the light point alpha.</td>
</tr>
<tr>
<td>Size Difference Threshold</td>
<td>Light point visibility threshold in pixels.</td>
</tr>
<tr>
<td>Transparent Pixel Size</td>
<td>Size in pixels where light points start becoming transparent.</td>
</tr>
<tr>
<td>Transparent Falloff Exponent</td>
<td>Exponential value controlling the transparent falloff. Linear falloff = 1.0.</td>
</tr>
</tbody>
</table>
### Transparent Scale
Scale factor controlling the transparent falloff.

### Transparent Clamp
Minimum transparent falloff.

### Minimum Pixel Size
Minimum diameter in pixels.

### Maximum Pixel Size
Maximum diameter in pixels.

### Actual Size
Diameter in database units (e.g. meters).

### Perspective Mode
Enabled if perspective fading is to be used.

### Reflective Only
Enabled if light point is reflective instead of emissive.

### Fading
Enabled if perspective fading calculations are to be used.

### Fog Punch
Enabled if light point is to punch through fog.

## Indexed Light Point Attributes
The table below describes the indexed light point attributes.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The indexed light point node name.</td>
</tr>
<tr>
<td>Appearance Index</td>
<td>The light point palette index for the appearance of the indexed light point.</td>
</tr>
<tr>
<td>Animation Index</td>
<td>The light point palette index for the animation of the indexed light point.</td>
</tr>
<tr>
<td>Calligraphic Draw Order</td>
<td>Calligraphic draw order index.</td>
</tr>
<tr>
<td>Replications</td>
<td>The number of times to replicate this indexed light point. A value of 0 is equal to no replications and the indexed light point will be visualized as normal. The replications are transformed by a transformation matrix on the indexed light point.</td>
</tr>
</tbody>
</table>

## Light Point System Attributes
The table below describes the light point system attributes.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The light point system node name.</td>
</tr>
<tr>
<td>Intensity</td>
<td>The intensity of the light points beneath this light point system node. This will not affect the visualization in Remo 3D but might be used in the target runtime system.</td>
</tr>
<tr>
<td>Animation State</td>
<td>The animation state of the light points beneath this light point system node (On, Off or Random). This will not affect the visualization in Remo 3D but might be used in the target runtime system.</td>
</tr>
</tbody>
</table>
### Enabled

| Enables or disables the light points beneath this light point system node. This will not affect the visualization in Remo 3D but might be used in the target runtime system. |

---

### Vertex Attributes

The tables below describe the vertex attributes.

The Coordinates tab contains the following attributes:

<table>
<thead>
<tr>
<th>Name</th>
<th>The vertex name, automatically created by its parent polygon or light point name plus the vertex number. This attribute is read-only.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coordinate – x</td>
<td>The x value of the vertex coordinate.</td>
</tr>
<tr>
<td>Coordinate – y</td>
<td>The y value of the vertex coordinate.</td>
</tr>
<tr>
<td>Coordinate – z</td>
<td>The z value of the vertex coordinate.</td>
</tr>
<tr>
<td>Texture Coordinate</td>
<td>Enabled if the vertex has a texture coordinate.</td>
</tr>
<tr>
<td>Texture Coordinate – u</td>
<td>The horizontal u value of the vertex texture coordinate.</td>
</tr>
<tr>
<td>Texture Coordinate – v</td>
<td>The vertical v value of the vertex texture coordinate.</td>
</tr>
<tr>
<td>Normal</td>
<td>Enabled if the vertex has a normal.</td>
</tr>
<tr>
<td>Normal – i</td>
<td>The i (x) value of the vertex normal vector.</td>
</tr>
<tr>
<td>Normal – j</td>
<td>The j (y) value of the vertex normal vector.</td>
</tr>
<tr>
<td>Normal – k</td>
<td>The k (z) value of the vertex normal vector.</td>
</tr>
<tr>
<td>Hard Edge</td>
<td>Enabled if the parent polygon normal is to be used for shading (not used by Remo 3D).</td>
</tr>
<tr>
<td>Frozen Normal</td>
<td>Enabled if the vertex normal is fixed (not used by Remo 3D).</td>
</tr>
</tbody>
</table>

The Color tab contains the following attributes:

| Color – Index | The color palette index of the vertex color (0 – 1023, -1 if none). Disabled if RGB color mode is used. |
| Color – Intensity | The intensity of the vertex color (0 – 127). Disabled if RGB color mode is used. |
| Color – Red | The red value of the vertex color (0 – 255). Disabled if indexed color mode is used. |
| Color – Green | The green value of the vertex color (0 – 255). }
### LOD Attributes

The table below describes the LOD attributes.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switch Range – Switch In (far range)</td>
<td>The far range at which the LOD is switched in, in database units (e.g. meters).</td>
</tr>
<tr>
<td>Switch Range – Switch Out (near range)</td>
<td>The near range at which the LOD is switched out, in database units (e.g. meters).</td>
</tr>
<tr>
<td>Switch Range – Transition</td>
<td>The interval in which the LOD is faded, in database units (e.g. meters).</td>
</tr>
<tr>
<td>Use Previous Slant Range</td>
<td>If this attribute is enabled, the target runtime system can assume the same slant range (distance from viewer to LOD center) as this node’s left sibling.</td>
</tr>
<tr>
<td>Additive LODs Below</td>
<td>This indicates that the LOD nodes beneath this LOD node are additive, i.e. that they add details rather than replace details.</td>
</tr>
<tr>
<td>LOD Center – x</td>
<td>The x value of the LOD center coordinate.</td>
</tr>
<tr>
<td>LOD Center – y</td>
<td>The y value of the LOD center coordinate.</td>
</tr>
<tr>
<td>LOD Center – z</td>
<td>The z value of the LOD center coordinate.</td>
</tr>
<tr>
<td>Freeze LOD Center</td>
<td>Enabled if the LOD center is to be frozen and not recalculated in the receiving system.</td>
</tr>
</tbody>
</table>

Note that the LOD center can be retrieved from the current pivot point by clicking the Get From Pivot button.

### DOF Attributes

The tables below describe the DOF attributes.

The General tab contains the following attributes:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Origin X</td>
<td>The x value of the origin of the DOF local coordinate system.</td>
</tr>
<tr>
<td>Origin Y</td>
<td>The y value of the origin of the DOF local coordinate system.</td>
</tr>
<tr>
<td>Origin Z</td>
<td>The z value of the origin of the DOF local coordinate system.</td>
</tr>
</tbody>
</table>
The Origin, Point On X Axis and Point In XY Plane can be retrieved from the current pivot point by clicking the Get From Pivot button.

The Pick... buttons can be used to pick the points with the middle mouse button in the viewports. When the Origin Pick... button is checked and a point is picked it automatically switches to pick the Point On X Axis and then switches to pick the Point in XY Plane.

The Translate tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum X</td>
<td>The x value of the minimum translation.</td>
</tr>
<tr>
<td>Minimum Y</td>
<td>The y value of the minimum translation.</td>
</tr>
<tr>
<td>Minimum Z</td>
<td>The z value of the minimum translation.</td>
</tr>
<tr>
<td>Current X</td>
<td>The x value of the current translation.</td>
</tr>
<tr>
<td>Current Y</td>
<td>The y value of the current translation.</td>
</tr>
<tr>
<td>Current Z</td>
<td>The z value of the current translation.</td>
</tr>
<tr>
<td>Maximum X</td>
<td>The x value of the maximum translation.</td>
</tr>
<tr>
<td>Maximum Y</td>
<td>The y value of the maximum translation.</td>
</tr>
<tr>
<td>Maximum Z</td>
<td>The z value of the maximum translation.</td>
</tr>
<tr>
<td>Increment X</td>
<td>The x value of the translation increment.</td>
</tr>
<tr>
<td>Increment Y</td>
<td>The y value of the translation increment.</td>
</tr>
<tr>
<td>Increment Z</td>
<td>The z value of the translation increment.</td>
</tr>
<tr>
<td>Limit X</td>
<td>Enabled if translation in the x direction is limited.</td>
</tr>
</tbody>
</table>
Limit Y
Enabled if translation in the y direction is limited.

Limit Z
Enabled if translation in the z direction is limited.

The Rotate tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum X Rotation (Pitch)</td>
<td>The minimum rotation around the x axis in degrees.</td>
</tr>
<tr>
<td>Minimum Y Rotation (Roll)</td>
<td>The minimum rotation around the y axis in degrees.</td>
</tr>
<tr>
<td>Minimum Z Rotation (Yaw)</td>
<td>The minimum rotation around the z axis in degrees.</td>
</tr>
<tr>
<td>Current X Rotation (Pitch)</td>
<td>The current rotation around the x axis in degrees.</td>
</tr>
<tr>
<td>Current Y Rotation (Roll)</td>
<td>The current rotation around the y axis in degrees.</td>
</tr>
<tr>
<td>Current Z Rotation (Yaw)</td>
<td>The current rotation around the z axis in degrees.</td>
</tr>
<tr>
<td>Maximum X Rotation (Pitch)</td>
<td>The maximum rotation around the x axis in degrees.</td>
</tr>
<tr>
<td>Maximum Y Rotation (Roll)</td>
<td>The maximum rotation around the y axis in degrees.</td>
</tr>
<tr>
<td>Maximum Z Rotation (Yaw)</td>
<td>The maximum rotation around the z axis in degrees.</td>
</tr>
<tr>
<td>Increment X Rotation (Pitch)</td>
<td>The rotation increment around the x axis in degrees.</td>
</tr>
<tr>
<td>Increment Y Rotation (Roll)</td>
<td>The rotation increment around the y axis in degrees.</td>
</tr>
<tr>
<td>Increment Z Rotation (Yaw)</td>
<td>The rotation increment around the z axis in degrees.</td>
</tr>
<tr>
<td>Limit X Rotation (Pitch)</td>
<td>Enabled if rotation around the x axis is limited.</td>
</tr>
<tr>
<td>Limit Y Rotation (Roll)</td>
<td>Enabled if rotation around the y axis limited.</td>
</tr>
<tr>
<td>Limit Z Rotation (Yaw)</td>
<td>Enabled if rotation around the z axis limited.</td>
</tr>
</tbody>
</table>

The Scale tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum X Scale</td>
<td>The x value of the minimum scale.</td>
</tr>
<tr>
<td>Minimum Y Scale</td>
<td>The y value of the minimum scale.</td>
</tr>
<tr>
<td>Minimum Z Scale</td>
<td>The z value of the minimum scale.</td>
</tr>
<tr>
<td>Current X Scale</td>
<td>The x value of the current scale.</td>
</tr>
<tr>
<td>Current Y Scale</td>
<td>The y value of the current scale.</td>
</tr>
<tr>
<td>Current Z Scale</td>
<td>The z value of the current scale.</td>
</tr>
<tr>
<td>Maximum X Scale</td>
<td>The x value of the maximum scale.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Maximum X Scale</td>
<td>The x value of the maximum scale.</td>
</tr>
<tr>
<td>Maximum Y Scale</td>
<td>The y value of the maximum scale.</td>
</tr>
<tr>
<td>Maximum Z Scale</td>
<td>The z value of the maximum scale.</td>
</tr>
<tr>
<td>Increment X Scale</td>
<td>The x value of the scale increment.</td>
</tr>
<tr>
<td>Increment Y Scale</td>
<td>The y value of the scale increment.</td>
</tr>
<tr>
<td>Increment Z Scale</td>
<td>The z value of the scale increment.</td>
</tr>
<tr>
<td>Limit X Scale</td>
<td>Enabled if scaling in the x direction is limited.</td>
</tr>
<tr>
<td>Limit Y Scale</td>
<td>Enabled if scaling in the y direction is limited.</td>
</tr>
<tr>
<td>Limit Z Scale</td>
<td>Enabled if scaling in the z direction is limited.</td>
</tr>
</tbody>
</table>

### Switch Attributes

The table below describes the switch attributes.

<table>
<thead>
<tr>
<th>Name</th>
<th>The switch node name.</th>
</tr>
</thead>
</table>

The rest of the switch attributes user interface controls the switch masks of the switch node. Masks can be created and removed by using the Add Before, Add After and Delete buttons. The area below these buttons lists the currently available masks. A switch mask is represented by its index number followed by a radio button indicating if this is the currently active mask, directly followed by a number of check boxes, one for each child of the switch node. If a check box is enabled, its corresponding child will be visible when its switch mask is active. The picture below shows an example of the attributes of a switch node having two children and three switch masks. In the first mask (0) only the first child is visible, in the second mask (1) the second child is visible and in the third mask (2) both children are visible. The currently active mask is the second mask (1).

When editing a switch mask, the check boxes can be manually enabled or disabled. To enable or disable all check boxes in one mask, the All On and All Off buttons can be used. The Invert button can be used to invert the status of the check boxes for the current mask. If other masks than the currently active mask are edited, don’t forget to reset the currently active mask again by clicking its radio button before applying the switch attributes.

*Note that the visibility of a switch node’s children can be modified manually by ordinary hiding and showing of the child nodes. However, applying attributes for a switch node will reset the visibility of the switch node’s children according to the currently selected switch mask.*
External Reference Attributes

The table below describes the external reference attributes.

<table>
<thead>
<tr>
<th>Filename</th>
<th>The OpenFlight® filename of the external reference.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use External Color Palette</td>
<td>Enabled if indexed polygon and vertex colors in the externally referenced geometry will be fetched from the color palette in the external file. Otherwise they will be fetched from the current model's color palette.</td>
</tr>
<tr>
<td>Use External Material Palette</td>
<td>Enabled if materials in the externally referenced geometry will be fetched from the material palette in the external file. Otherwise they will be fetched from the current model’s material palette.</td>
</tr>
<tr>
<td>Use External Texture Palette</td>
<td>Enabled if textures and texture mappings in the externally referenced geometry will be fetched from the texture and texture mapping palettes in the external file. Otherwise they will be fetched from the current model’s texture and texture mapping palettes.</td>
</tr>
<tr>
<td>Use External Line Style Palette</td>
<td>Enabled if line styles in the externally referenced geometry will be fetched from the line style palette in the external file. Otherwise they will be fetched from the current model’s line style palette (not used in Remo 3D).</td>
</tr>
<tr>
<td>Use External Sound Palette</td>
<td>Enabled if sounds in the externally referenced geometry will be fetched from the sound palette in the external file. Otherwise they will be fetched from the current model’s sound palette (not used in Remo 3D).</td>
</tr>
<tr>
<td>Use External Light Source Palette</td>
<td>Enabled if light sources in the externally referenced geometry will be fetched from the light source palette in the external file. Otherwise they will be fetched from the current model’s light source palette (not used in Remo 3D).</td>
</tr>
<tr>
<td>Use External Light Point Palette</td>
<td>Enabled if indexed light points in the externally referenced geometry will be fetched from the light point palette in the external file. Otherwise they will be fetched from the current model’s light point palette.</td>
</tr>
</tbody>
</table>

The external reference filename also can be specified by clicking the Browse (…) button to next to the Filename field. Note that if the Load External References setting described
in the Preferences section is disabled, the model won’t be loaded until it’s enabled in the External References dialog available from the File menu.

**Apply Comment**

The comment tool lets the user add, change or remove a comment on any node. If all nodes selected don’t have the same comments, this is indicated by erasing the comment field and making it gray. New or modified comments are confirmed by clicking the Apply button. If a comment is modified and even if the comments of the selected nodes were not the same, it is assigned to all selected nodes. If the comment is left unmodified, no assignment is made, not even for the differing nodes.
Modes

A mode represents the type of primitive that can currently be selected or snapped to. The following modes are available:

Vertex

Vertices are corner points of polygon or light point nodes.

Edge

Edges are virtual structures representing pairs of consecutive vertices.

Polygon

Polygons are surface nodes, each one containing a number of vertices.

Object

Objects are hierarchy grouping nodes containing only polygon or light point nodes.

Group

Groups are general hierarchy grouping nodes.

LOD

LODs are level-of-detail hierarchy grouping nodes switched on or off depending on viewing distance.

DOF

DOFs are degree-of-freedom hierarchy grouping nodes allowing for transformation of contained geometry.
Switch

Switches are hierarchy grouping nodes switching contained geometry on or off with a number of switch masks.

Light Point

Light points are light point nodes, each one containing a number of vertices.

Light Point System

Light point systems are hierarchy grouping nodes containing only light points.

External Reference

External references are nodes referencing external OpenFlight® files.
Tree View

This section describes the tree view which shows the hierarchy of the model.

Navigate in the tree view by using the scroll bars located to the right and at the bottom (note that the scroll bars are only shown if they are needed). The view can also be moved by clicking and dragging Ctrl + right mouse button in the same manner as in the viewport. A node is visualized by a box labeled by the node’s name. Different node types have different colors as described in the Node Colors section. Polygon nodes have a small box in their top right corner that shows the color of the polygon. Nodes having transformation matrices are labeled with an ‘M’ in their bottom right corner. Replicated nodes are labeled with an ‘R’ in their bottom right corner. Instances are visualized with a small arrow pointing downwards at the referred nodes. Vertices are shown as separate boxes, even though they are not actually nodes.

The nodes can be selected by clicking on them with the left mouse button. Shift-clicking on a node will add that node to the current selection. Alt-clicking on an already selected node will deselect that node. Selecting nodes within a rectangular area in the tree view is also possible by clicking the left mouse button outside the nodes and dragging a box around the nodes that are to be selected. Holding Shift while creating selection box will add non-selected nodes within the rectangular area to the selection. Holding Alt while making the selection box will remove selected nodes within the rectangular area from the current selection. If the cursor is dragged outside the tree view when using the box selection, the scroll bar will be scrolled toward the cursor if possible. The further from the tree view the cursor is moved, the faster the scroll will be.

Pressing ‘v’ in the tree view will cycle through the selected nodes. Pressing ‘V’ will cycle backwards. This can be useful when e.g. selecting some geometry in the model view, changing focus by clicking the right mouse button (not the left mouse button since that would imply a new selection) in the tree view and pressing ‘v’ to find out where the selected node exists in the hierarchy.

Drag and drop of nodes is supported by clicking on the selected nodes with the left mouse button and dragging them to their new parent. As with the box selection, if the cursor is dragged outside the tree view, the scroll bars will be scrolled toward the cursor.

To the left are buttons for expanding and contracting sub-trees beneath the currently selected node(s). This expansion/contraction will be performed incrementally, i.e. multiple clicks on the expand button will expand the selected sub-tree to gradually lower levels. Expanding and contracting can also be performed by clicking the middle mouse button on a node. This expansion/contraction will only expand or contract the clicked node and will not affect the nodes in the sub-tree beneath. By holding down Shift and
clicking with the middle mouse button on a selected node, all selected nodes will be contracted or expanded depending on the state of the clicked node.

There is also a slider that can be used to zoom the tree view in and out. The zoom slider has a snapping center position, allowing the user to easily find the default zoom level. The view can also be zoomed by clicking and dragging Shift + right mouse button in the same manner as in the viewport.

The left button area also contains buttons for shifting the selected nodes to the left or to the right, one step at a time (Shift Left/Right) or all the way to the left or to the right (Shift Far Left/Right). This can be performed on a number of selected nodes of different types and will produce undoable commands.

The right button area contains buttons for selecting the parents or children of the currently selected primitives. There is also a button for selecting the previously selected nodes. This button will toggle between the two latest selections.

Nodes that are hidden in the last viewport focused are colored light gray in the tree view. Invisible LOD nodes are also colored with the same light gray color. Node types not fully supported in Remo 3D will be visualized with a medium gray color. They can still be cut, copied, pasted, deleted, dragged and dropped, but their attributes can’t be modified.

The attribute tool can be started for a selected node by double-clicking it. When the cursor is located over a node in the tree view the full name of that node will be displayed in the status area.

**Node Colors**

- The database header node has a dark gray color.
- Group nodes have a blue-purple color.
- Object nodes have a light blue color.
- LOD nodes have a light green color.
- DOF nodes have a red-purple color.
- Switch nodes have a pink color.
- External reference nodes have an orange color.
- Light point nodes have a yellow color.
- Light point system nodes have a cyan/turquoise color.
- Polygon nodes have a blue-green color.
- Vertices have a red-orange color.
Palettes

This section describes all palettes that can be used in Remo 3D.

Color Palette

This window presents the color palette with its 1024 color indices. The color for each index can be edited by selecting the color, choosing Edit Color from the Edit menu and picking a new color.

The color picking is done manually by indicating the hue, saturation and value (lightness) with the mouse or typing the RGB or HSV values. The RGB values can be given as floats, bytes or hexadecimal.

The color to be picked (e.g. the model’s current color or a polygon color - depending on from where the color palette was activated) is defined by choosing a color index and setting a color intensity. By clicking the OK button, the color is set and the palette window is closed. By clicking the Close button or choosing Close from the File menu, no color is applied and the palette window is closed.

If the current model uses indexed color mode, the result from a picked color is a color index and an intensity. If RGB color mode is used (controlled by a header node attribute), the result is a color defined by red, green and blue values. Picking the current color from the palette by clicking the current color button in the main user interface or picking a new color for a polygon or a vertex is supported both in the indexed and RGB
color modes. Picking a light point back color always results in an indexed color, even if the model is in RGB mode.

A color palette can be opened from disk by choosing Open Palette from the File menu and saved to disk by choosing Save Palette As from the File menu. Color palette files have an RCP (Remo 3D Color Palette) extension.

The general functionality of the file dialogs used in the color palette is described in the File Dialogs section.

**Texture Palette**

This window presents the texture palette with textures represented as images regularly distributed within the window.

The images are presented with a maximum size of 128x128. If an image has to be scaled down, the image can be opened in a window showing it in its original size by choosing Full Size from the View menu. A list of all textures with their indices, file paths and sizes can be viewed by choosing Texture List from the View menu.

The texture to be picked (e.g. the current texture or a polygon texture - depending on from where the texture palette was activated) is defined by choosing an image representing a certain texture index. By clicking the OK button, the texture is set and the palette window is closed. By clicking the Close button or choosing Close from the File menu, no texture is applied and the palette window is closed.
An existing texture can be removed by selecting the texture and choosing Remove from the Edit menu. A new texture can be loaded into the palette by choosing Load Texture from the File menu and choosing the filename of the texture that is to be loaded. Supported image formats are SGI® (RGB, RGBA, INT, INTA), BMP, DDS, GIF, JPEG, PNG, PNM (PNM, PPM, PGM, PBM), TGA and TIFF. To replace a texture, select the texture, choose Replace Texture from the File menu and choose which file with which the texture shall be replaced. From the Edit menu all textures that are not used in the current model can be removed by choosing Remove Unused.

When selecting Attributes from the Edit menu, a dialog with texture attributes is presented to the user. Through this dialog, texture attributes can be created, edited and saved. Texture attributes are saved to a file with the same name as the texture plus an '.attr' extension. The different texture attributes are described in more detail in the Texture Attributes section below.

A texture’s index can be changed by selecting the texture, choosing Change Index from the Edit menu and specifying a new index. To remove the path of a texture and only keep the filename, select the texture and choose Remove Path from the Edit menu. All texture paths can be changed by choosing Change All Paths from the Edit menu and specifying a new folder. All texture paths can also be removed by choosing Remove All Paths from the Edit menu. By enabling the Use Relative Paths menu item, all paths currently in the texture palette will be converted to relative and all loaded textures after that will use a relative path. By disabling the Use Relative Paths menu item, all paths will be converted to absolute. All relative paths will be relative to the directory of the model.

Since the vertical orientation of DDS and PNM image files can be different, this can be controlled with Flip DDS/PNM Images from the Edit menu. Enabling or disabling flipping will reload all DDS and PNM textures, but it will not change any textures or models on disk. The default is that flipping is enabled.

A texture palette can be opened from disk by selecting Open Palette from the File menu and saved to disk by selecting Save Palette As from the File menu. Texture palette files have an RTP (Remo 3D Texture Palette) extension.

To reload all textures from disk, Refresh from the View menu can be used.

When loading a model with textures, the REMO3D_FILE_PATH environment variable is used if the texture files aren’t found directly.

The general functionality of the file dialogs used in the texture palette is described in the File Dialogs section.
Texture Attributes

The tables below describe the texture attributes.

The General tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filename (read-only)</td>
<td>The filename of the corresponding texture image file.</td>
</tr>
<tr>
<td>Internal Format</td>
<td>The internal format used on the graphics card for this texture, one of the following:</td>
</tr>
<tr>
<td>Default</td>
<td>Default</td>
</tr>
<tr>
<td>IA_12A_4</td>
<td>(12 bits intensity, 4 bits alpha)</td>
</tr>
<tr>
<td>IA_8</td>
<td>(8 bits intensity, 8 bits alpha)</td>
</tr>
<tr>
<td>RGB_5</td>
<td>(5 bits red, 6 bits green, 5 bits blue)</td>
</tr>
<tr>
<td>RGBA_4</td>
<td>(4 bits red, 4 bits green, 4 bits blue, 4 bits alpha)</td>
</tr>
<tr>
<td>IA_12</td>
<td>(12 bits intensity, 12 bits alpha)</td>
</tr>
<tr>
<td>RGBA_8</td>
<td>(8 bits red, 8 bits green, 8 bits blue 8 bits alpha)</td>
</tr>
<tr>
<td>RGBA_12</td>
<td>(12 bits red, 12 bits green, 12 bits blue, 12 bits alpha)</td>
</tr>
<tr>
<td>I_16</td>
<td>(16 bits intensity)</td>
</tr>
<tr>
<td>RGB_12</td>
<td>(12 bits red, 12 bits green, 12 bits blue)</td>
</tr>
<tr>
<td>RG_X</td>
<td>X-component (in width direction) of pivot point for rotating textures.</td>
</tr>
<tr>
<td>Y-component (in height direction) of pivot point for rotating textures.</td>
<td></td>
</tr>
</tbody>
</table>

The Geometry tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Earth Model</td>
<td>The earth model associated with the coordinates of this model (WGS84, WGS72, Bessel, Clark 1866 or NAD27).</td>
</tr>
<tr>
<td>Projection</td>
<td>The projection associated with the coordinates of this model (Flat Earth, Lambert Conic, UTM or Undefined).</td>
</tr>
<tr>
<td>Image Origin</td>
<td>The image origin (Lower Left or Upper Left).</td>
</tr>
<tr>
<td>Texture Size – u (read-only)</td>
<td>The texture image width in pixels.</td>
</tr>
<tr>
<td>Texture Size – v (read-only)</td>
<td>The texture image height in pixels.</td>
</tr>
<tr>
<td>Real World Size – u</td>
<td>The real-world width in database units (e.g. meters).</td>
</tr>
<tr>
<td>Real World Size – v</td>
<td>The real-world height in database units (e.g. meters).</td>
</tr>
<tr>
<td>Pivot – x</td>
<td>X-component (in width direction) of pivot point for rotating textures.</td>
</tr>
<tr>
<td>Pivot – y</td>
<td>Y-component (in height direction) of pivot point for rotating textures.</td>
</tr>
<tr>
<td>Up Vector – x</td>
<td>X-component (in width direction) of an orientation vector, only for backwards compatibility.</td>
</tr>
<tr>
<td>Up Vector – y</td>
<td>Y-component (in height direction) of an orientation</td>
</tr>
</tbody>
</table>
The Filters tab contains the following attributes:

<table>
<thead>
<tr>
<th>Minification Filter</th>
<th>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td>POINT (nearest pixel)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
<tr>
<td>BILINEAR (weighted average of four nearest pixels)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
<tr>
<td>MIPMAP (obsolete)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
<tr>
<td>MIPMAP_POINT (nearest pixel in closest mipmap level)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
<tr>
<td>MIPMAP_LINEAR (nearest pixel in average between two closest mipmap levels)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
<tr>
<td>MIPMAP_BILINEAR (weighted average of four nearest pixels in closest mipmap level)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
<tr>
<td>MIPMAP_TRILINEAR (weighted average of two weighted averages of four nearest pixels in two closest mipmap levels)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
<tr>
<td>None (uses system default, usually BILINEAR)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
<tr>
<td>BICUBIC (spline-weighted average of 16 nearest pixels)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
<tr>
<td>BILINEAR_GEQUAL (can be used for real-time shadows)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
<tr>
<td>BILINEAR_LEQUAL (can be used for real-time shadows)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
<tr>
<td>BICUBIC_GEQUAL (can be used for real-time shadows)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
<tr>
<td>BICUBIC_LEQUAL (can be used for real-time shadows)</td>
<td>The filter used for decreasing the resolution of the texture when viewed from afar, one of the following:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Magnification Filter</th>
<th>The filter used for increasing the resolution of the texture when viewed at a close range (POINT, BILINEAR, None, BICUBIC, SHARPEN, ADD_DETAIL, MODULATE_DETAIL, BILINEAR_GEQUAL, BILINEAR_LEQUAL, BICUBIC_GEQUAL, BICUBIC_LEQUAL).</th>
</tr>
</thead>
<tbody>
<tr>
<td>POINT (nearest pixel)</td>
<td>The filter used for increasing the resolution of the texture when viewed at a close range (POINT, BILINEAR, None, BICUBIC, SHARPEN, ADD_DETAIL, MODULATE_DETAIL, BILINEAR_GEQUAL, BILINEAR_LEQUAL, BICUBIC_GEQUAL, BICUBIC_LEQUAL).</td>
</tr>
<tr>
<td>BILINEAR (weighted average of four nearest pixels)</td>
<td>The filter used for increasing the resolution of the texture when viewed at a close range (POINT, BILINEAR, None, BICUBIC, SHARPEN, ADD_DETAIL, MODULATE_DETAIL, BILINEAR_GEQUAL, BILINEAR_LEQUAL, BICUBIC_GEQUAL, BICUBIC_LEQUAL).</td>
</tr>
<tr>
<td>None (uses system default, usually BILINEAR)</td>
<td>The filter used for increasing the resolution of the texture when viewed at a close range (POINT, BILINEAR, None, BICUBIC, SHARPEN, ADD_DETAIL, MODULATE_DETAIL, BILINEAR_GEQUAL, BILINEAR_LEQUAL, BICUBIC_GEQUAL, BICUBIC_LEQUAL).</td>
</tr>
<tr>
<td>BICUBIC (spline-weighted average of 8 nearest pixels)</td>
<td>The filter used for increasing the resolution of the texture when viewed at a close range (POINT, BILINEAR, None, BICUBIC, SHARPEN, ADD_DETAIL, MODULATE_DETAIL, BILINEAR_GEQUAL, BILINEAR_LEQUAL, BICUBIC_GEQUAL, BICUBIC_LEQUAL).</td>
</tr>
<tr>
<td>SHARPEN (extrapolates from weighted averages of four nearest pixels in the two highest mipmap levels)</td>
<td>The filter used for increasing the resolution of the texture when viewed at a close range (POINT, BILINEAR, None, BICUBIC, SHARPEN, ADD_DETAIL, MODULATE_DETAIL, BILINEAR_GEQUAL, BILINEAR_LEQUAL, BICUBIC_GEQUAL, BICUBIC_LEQUAL).</td>
</tr>
<tr>
<td>ADD_DETAIL (adds the detail texture specified in the polygon attributes, normally only supported in SGI® and not visualized in Remo 3D)</td>
<td>The filter used for increasing the resolution of the texture when viewed at a close range (POINT, BILINEAR, None, BICUBIC, SHARPEN, ADD_DETAIL, MODULATE_DETAIL, BILINEAR_GEQUAL, BILINEAR_LEQUAL, BICUBIC_GEQUAL, BICUBIC_LEQUAL).</td>
</tr>
<tr>
<td>MODULATE_DETAIL (modulates the detail texture specified in the polygon attributes, normally only supported in SGI® and not visualized in Remo 3D)</td>
<td>The filter used for increasing the resolution of the texture when viewed at a close range (POINT, BILINEAR, None, BICUBIC, SHARPEN, ADD_DETAIL, MODULATE_DETAIL, BILINEAR_GEQUAL, BILINEAR_LEQUAL, BICUBIC_GEQUAL, BICUBIC_LEQUAL).</td>
</tr>
<tr>
<td>BILINEAR_GEQUAL (can be used for real-time shadows)</td>
<td>The filter used for increasing the resolution of the texture when viewed at a close range (POINT, BILINEAR, None, BICUBIC, SHARPEN, ADD_DETAIL, MODULATE_DETAIL, BILINEAR_GEQUAL, BILINEAR_LEQUAL, BICUBIC_GEQUAL, BICUBIC_LEQUAL).</td>
</tr>
<tr>
<td>BILINEAR_LEQUAL (can be used for real-time shadows)</td>
<td>The filter used for increasing the resolution of the texture when viewed at a close range (POINT, BILINEAR, None, BICUBIC, SHARPEN, ADD_DETAIL, MODULATE_DETAIL, BILINEAR_GEQUAL, BILINEAR_LEQUAL, BICUBIC_GEQUAL, BICUBIC_LEQUAL).</td>
</tr>
<tr>
<td>BICUBIC_GEQUAL (can be used for real-time shadows)</td>
<td>The filter used for increasing the resolution of the texture when viewed at a close range (POINT, BILINEAR, None, BICUBIC, SHARPEN, ADD_DETAIL, MODULATE_DETAIL, BILINEAR_GEQUAL, BILINEAR_LEQUAL, BICUBIC_GEQUAL, BICUBIC_LEQUAL).</td>
</tr>
<tr>
<td>BICUBIC_LEQUAL (can be used for real-time shadows)</td>
<td>The filter used for increasing the resolution of the texture when viewed at a close range (POINT, BILINEAR, None, BICUBIC, SHARPEN, ADD_DETAIL, MODULATE_DETAIL, BILINEAR_GEQUAL, BILINEAR_LEQUAL, BICUBIC_GEQUAL, BICUBIC_LEQUAL).</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Magnification Filter Alpha</th>
<th>The alpha-specific filter used for increasing the resolution of the texture when viewed at a close range, see possible values for magnification filter above.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Magnification Filter Color</th>
<th>The color-specific filter used for increasing the resolution of the texture when viewed at a close range, see possible values for magnification filter above.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Environment Type</th>
<th>The method for blending texture color with the color of the underlying polygon or other textures, one of the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODULATE (multiplicative integration between texture and vector, only for backwards compatibility.</td>
<td></td>
</tr>
</tbody>
</table>
### polygon color
- **BLEND** (average between primary and secondary polygon color, weighted by texture value)
- **DECAL** (replaces with texture color; polygon color blended according to texture alpha)
- **REPLACE/COLOR** (replaces with texture color)
- **ADD** (adds texture color to polygon color)

### Wrap Method u
The method for wrapping this texture in the u (width) direction, one of the following:
- Repeat (image copied to fill textured polygons)
- Clamp (edge pixels repeated to fill textured polygons)
- None
- Mirror (like repeat but every other image is mirrored)

### Wrap Method v
The method for wrapping this texture in the v (height) direction, one of the following:
- Repeat (image copied to fill textured polygons)
- Clamp (edge pixels repeated to fill textured polygons)
- None
- Mirror (like repeat but every other image is mirrored)

### Wrap Method u, v
The method for wrapping this texture if any of the direction-specific methods above are set to None; one of the following:
- Repeat (image copied to fill textured polygons)
- Clamp (edge pixels repeated to fill textured polygons)
- Mirror (like repeat but every other image is mirrored)

### The Detail tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Use Detail</strong></td>
<td>True if the relationship between detail and main texture is to be specified. Note that this has no effect on the visualization in Remo 3D. It's purely an IRISGL feature used on SGI® hardware.</td>
</tr>
<tr>
<td><strong>j</strong></td>
<td>The relative detail texture width, assumed to be 4.</td>
</tr>
<tr>
<td><strong>k</strong></td>
<td>The relative detail texture height, assumed to be 4.</td>
</tr>
<tr>
<td><strong>m</strong></td>
<td>The main texture width, valid values are 4, 8, 16, 32 and 64 (higher values imply wrapping).</td>
</tr>
<tr>
<td><strong>n</strong></td>
<td>The main texture height, valid values are 4, 8, 16, 32 and 64 (higher values imply wrapping).</td>
</tr>
<tr>
<td><strong>Scramble</strong></td>
<td>Not used, only for backwards compatibility.</td>
</tr>
</tbody>
</table>

### The MIP tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Use MIP Kernel</strong></td>
<td>True if a specific 8x8 convolution kernel is to be used for calculating mipmaps (image copies of decreasing power-of-two sizes to reduce aliasing artifacts). Note that this has no effect on the visualization in Remo 3D.</td>
</tr>
</tbody>
</table>
It's purely an IRISGL feature used on SGI hardware. Mipmaps will be automatically pre-calculated even without this kernel.

**Kernel Value (1-8)** These eight values form a vertical matrix and a horizontal matrix according to below. The convolution kernel is generated by multiplying these matrices.

\[
\begin{array}{cccc}
1 & 2 \\
3 & 4 & X & 1 2 3 4 5 6 7 8 \\
5 & 6 \\
7 & 8
\end{array}
\]

The Control tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Control Point</td>
<td>True if the texture level-of-detail scale and minification/magnification is to be specifically controlled. Note that this has no effect on the visualization in Remo 3D. It's purely an IRISGL feature used on SGI hardware.</td>
</tr>
<tr>
<td>LOD 0</td>
<td>The first texture level-of-detail value corresponding to the first texture scale value.</td>
</tr>
<tr>
<td>LOD 1</td>
<td>The second texture level-of-detail value corresponding to the second texture scale value.</td>
</tr>
<tr>
<td>LOD 2</td>
<td>The third texture level-of-detail value corresponding to the third texture scale value.</td>
</tr>
<tr>
<td>LOD 3</td>
<td>The fourth texture level-of-detail value corresponding to the fourth texture scale value.</td>
</tr>
<tr>
<td>LOD 4</td>
<td>Not used.</td>
</tr>
<tr>
<td>LOD 5</td>
<td>Not used.</td>
</tr>
<tr>
<td>LOD 6</td>
<td>Not used.</td>
</tr>
<tr>
<td>LOD 7</td>
<td>Not used.</td>
</tr>
<tr>
<td>Scale 0</td>
<td>The first texture scale value corresponding to the first texture level-of-detail value.</td>
</tr>
<tr>
<td>Scale 1</td>
<td>The second texture scale value corresponding to the second texture level-of-detail value.</td>
</tr>
<tr>
<td>Scale 2</td>
<td>The third texture scale value corresponding to the third texture level-of-detail value.</td>
</tr>
<tr>
<td>Scale 3</td>
<td>The fourth texture scale value corresponding to the fourth texture level-of-detail value.</td>
</tr>
<tr>
<td>Scale 4</td>
<td>Not used.</td>
</tr>
<tr>
<td>Scale 5</td>
<td>Not used.</td>
</tr>
<tr>
<td>Scale 6</td>
<td>Not used.</td>
</tr>
<tr>
<td>Scale 7</td>
<td>Not used.</td>
</tr>
<tr>
<td>Clamp</td>
<td>The maximum texture scale value.</td>
</tr>
</tbody>
</table>

The Tile tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Tile</td>
<td>True if a sub-section of the texture is to be visualized</td>
</tr>
</tbody>
</table>
and the rest completely transparent. Note that this has no effect on the visualization in Remo 3D.

| Lower Left - u | U-component (in width direction) of the lower-left corner of the visible part of the texture. |
| Lower Left – v | V-component (in height direction) of the lower-left corner of the visible part of the texture. |
| Upper Right - u | U-component (in width direction) of the upper-right corner of the visible part of the texture. |
| Upper Right – v | V-component (in height direction) of the upper-right corner of the visible part of the texture. |

Texture Mapping Palette

This window presents the texture mapping palette with texture mappings represented as type-specific (3-point, 4-point, cylindrical or spherical) images regularly distributed within the window. These images are represented with the currently chosen texture but the mappings will work with any texture.

The texture mapping to be picked (e.g. the current texture mapping or a polygon texture mapping - depending on from where the texture mapping palette was activated) is defined by choosing an image representing a certain texture mapping index. Clicking in the area outside the texture mapping images will mean selecting no texture mapping, resulting in a -1 index. By clicking the OK button, the texture mapping is set and the palette window is closed. By clicking the Close button or choosing Close from the File menu, no texture mapping is applied and the palette window is closed.
Texture mappings are created through the Map Texture Tool described in the Map Texture section. An existing texture mapping can be removed by selecting the texture mapping and choosing Remove from the Edit menu, or using the Delete button. All texture mappings that are not used in the current model can be removed by choosing Remove Unused from the Edit menu.

When selecting one of the texture mappings, its attributes are shown to the right. Depending on the type of texture mapping selected, different attributes can be edited. For mappings of all types, an optional name can be edited. For planar mappings (3-point and 4-point), the texture coordinates \((u, v)\) corresponding to the three or four geometry coordinates can be edited, either by moving the three or four points in the texture image as visualized in the dialog, by moving the same points in a full-size view of the texture image after clicking the Full View button, or by editing the exact \((u, v)\) values for each of the three or four points. For cylindrical and spherical mappings, the number of repetitions can be edited. Future versions of Remo 3D will allow editing the geometry points as well. Note that the actual modification (and corresponding undoable command) of the texture mapping attributes will be performed when leaving a text field, e.g. focusing another part of the user interface or leaving the field by pressing Tab or Return.

A texture mapping’s index can be changed by selecting the texture mapping, choosing Change Index from the Edit menu and specifying a new index.

A texture mapping palette can be opened from disk by selecting Open Palette from the File menu and saved to disk by selecting Save Palette As from the File menu. Texture mapping palette files have an RXP (Remo 3D teXture mapping Palette) extension.

The general functionality of the file dialogs used in the texture mapping palette is described in the File Dialogs section.
Material Palette

This window presents the material palette with materials represented as images of material spheres regularly distributed within the window.

The material to be picked (e.g. the current material or a polygon material - depending on from where the material palette was activated) is defined by choosing a sphere representing a certain material index. By clicking the OK button, the material is set and the palette window is closed. By clicking the Close button or choosing Close from the File menu, no material is applied and the palette window is closed.

A new material can be created by choosing New from the Edit menu. A default material will then be created and added at the end of the palette. Each material can be edited by selecting the material and clicking one of the ambient, diffuse, specular or emissive color boxes and choosing a new color. A color is chosen by indicating the hue, saturation and value (lightness) with the mouse or typing the RGB or HSV values. The RGB values can be given as floats, bytes or hexadecimal. The material’s shininess and alpha parameters can also be changed. When changing these colors and parameters immediate feedback will be given in the visualization of the selected material in the palette.

A material’s index can be changed by selecting the material, choosing Change Index from the Edit menu and specifying a new index. An existing material can be removed by selecting the material and choosing Remove from the Edit menu. To remove all unused materials, choose Remove Unused from the Edit menu.

A material palette can be opened from disk by selecting Open Palette from the File menu and saved to disk by selecting Save Palette As from the File menu. Material palette files have an RMP (Remo 3D Material Palette) extension.
The general functionality of the file dialogs used in the material palette is described in the File Dialogs section.

**Light Point Palette**

This window presents the light point palette with one list for appearances and one list for animations.

The appearance and animation to be picked are defined by choosing the corresponding rows in the appearance and animation list. The user can switch between the lists with the Appearance and Animation radio buttons to the left. By clicking the OK button, the current light point appearance and animation indices are set and the palette window is closed. By clicking the Close button or choosing Close from the File menu, no appearance or animation is applied and the palette window is closed.

A new appearance or animation can be created by switching to the appearance or animation list with the radio buttons and choosing New from the Edit menu. A default appearance or animation will then be created and added at the end of the list. Each appearance and animation can be edited by selecting the appearance or animation in the list and modifying the attributes appearing to the right. Note that the actual modification (and corresponding undoable command) will be performed when leaving a text field, e.g. focusing another part of the user interface or leaving the field by pressing Tab or Return. The attributes are described in the following two sub-sections:
Light Point Appearance Attributes

The tables below describe the light point appearance attributes.

The General tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The light point appearance name.</td>
</tr>
<tr>
<td>Display Mode</td>
<td>Defines how this light point is displayed (Raster, Calligraphic or Either).</td>
</tr>
<tr>
<td>Feature ID</td>
<td>A feature identification code.</td>
</tr>
<tr>
<td>Surface Material Code</td>
<td>A classification code defining the material of this light point.</td>
</tr>
<tr>
<td>Front Intensity</td>
<td>Intensity scalar for front color.</td>
</tr>
<tr>
<td>Back Intensity</td>
<td>Intensity scalar for back color.</td>
</tr>
<tr>
<td>Back Color</td>
<td>Enabled if back color is used for bidirectional light points.</td>
</tr>
<tr>
<td>Back Color – Index</td>
<td>The color palette index of the back color (0 – 1023, -1 if none). This color is always indexed, even if the model is in RGB mode.</td>
</tr>
<tr>
<td>Back Color – Intensity</td>
<td>The intensity of the back color (0 – 127). This color is always indexed, even if the model is in RGB mode.</td>
</tr>
<tr>
<td>Display In Daylight</td>
<td>Enabled if this light point will be displayed in daylight.</td>
</tr>
<tr>
<td>Display At Dusk</td>
<td>Enabled if this light point will be displayed at dusk.</td>
</tr>
<tr>
<td>Display At Night</td>
<td>Enabled if this light point will be displayed at night.</td>
</tr>
<tr>
<td>Conform To Terrain</td>
<td>Enabled if the light point is to be adapted in height to always be on top of the terrain. This is not visualized in Remo 3D.</td>
</tr>
</tbody>
</table>

The Direction tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Directional Type</td>
<td>Defines the directions of this light point’s emitting light (Omnidirectional, Unidirectional or Bidirectional).</td>
</tr>
<tr>
<td>Horizontal Lobe Angle</td>
<td>The horizontal angle of the light lobe (degrees).</td>
</tr>
<tr>
<td>Vertical Lobe Angle</td>
<td>The vertical angle of the light lobe (degrees).</td>
</tr>
<tr>
<td>Lobe Roll Angle</td>
<td>The roll angle of the light lobe (degrees).</td>
</tr>
<tr>
<td>Falloff Exponent</td>
<td>Exponential value controlling the falloff between center and edge of the light lobe. Linear falloff = 1.0.</td>
</tr>
<tr>
<td>Ambient Intensity</td>
<td>The ambient intensity of the light point when viewed off-axis.</td>
</tr>
</tbody>
</table>

The Callig (calligraphic) tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Significance</td>
<td>Priority used by raster/calligraphic hybrid systems for deciding which light points that should be switched to raster mode at high visualization loads (0.0 – 1.0).</td>
</tr>
<tr>
<td>Minimum Defocus</td>
<td>Lower calligraphic defocus limit (0.0 – 1.0).</td>
</tr>
<tr>
<td>Maximum Defocus</td>
<td>Upper calligraphic defocus limit (0.0 – 1.0).</td>
</tr>
</tbody>
</table>
Quality | Calligraphic quality (Low, Medium or High).
---|---
Proximity Occulting | Enabled if the intensity of adjacent light points are to be decreased when viewed from a distance.

The Misc (miscellaneous) tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range Mode</td>
<td>Method for calculating the distance between the viewer and the light point (Depth Buffer or Slant Range).</td>
</tr>
<tr>
<td>Randomize Intensity</td>
<td>The random intensity level (Off, Low, Medium or High).</td>
</tr>
<tr>
<td>Fog Scale</td>
<td>The level of light point visibility through fog. Values above 0.0 imply punching through the fog.</td>
</tr>
<tr>
<td>Fog Intensity</td>
<td>Intensity factor controlling the light point alpha.</td>
</tr>
<tr>
<td>Size Difference Threshold</td>
<td>Light point visibility threshold in pixels.</td>
</tr>
<tr>
<td>Transparent Pixel Size</td>
<td>Size in pixels where light points start becoming transparent.</td>
</tr>
<tr>
<td>Transparent Falloff Exponent</td>
<td>Exponential value controlling the transparent falloff. Linear falloff = 1.0.</td>
</tr>
<tr>
<td>Transparent Scale</td>
<td>Scale factor controlling the transparent falloff.</td>
</tr>
<tr>
<td>Transparent Clamp</td>
<td>Minimum transparent falloff.</td>
</tr>
<tr>
<td>Minimum Pixel Size</td>
<td>Minimum diameter in pixels.</td>
</tr>
<tr>
<td>Maximum Pixel Size</td>
<td>Maximum diameter in pixels.</td>
</tr>
<tr>
<td>Actual Size</td>
<td>Diameter in database units (e.g. meters).</td>
</tr>
<tr>
<td>Perspective Mode</td>
<td>Enabled if perspective fading is to be used.</td>
</tr>
<tr>
<td>Reflective Only</td>
<td>Enabled if light point is reflective instead of emissive.</td>
</tr>
<tr>
<td>Fading</td>
<td>Enabled if perspective fading calculations are to be used.</td>
</tr>
<tr>
<td>Fog Punch</td>
<td>Enabled if light point is to punch through fog.</td>
</tr>
</tbody>
</table>

The Misc2 (miscellaneous continued) tab contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visibility Range</td>
<td>The range within which the light point is visible, in database units (e.g. meters).</td>
</tr>
<tr>
<td>Fade Range Ratio</td>
<td>The ratio of the visibility range within which the light point fades away (0.0 – 1.0).</td>
</tr>
<tr>
<td>Fade In Duration</td>
<td>The fade-in time interval for the light point, in seconds.</td>
</tr>
<tr>
<td>Fade Out Duration</td>
<td>The fade-out time interval for the light point, in seconds.</td>
</tr>
<tr>
<td>LOD Range Ratio</td>
<td>Certain receiving systems support changing the light point visualization to a textured polygon at close range, as a higher level-of-detail representation. This is the ratio (0.0 – 1.0) of the visibility range within which this higher level-of-detail is active.</td>
</tr>
<tr>
<td>LOD Scale</td>
<td>Certain receiving systems support changing the light point visualization to a textured polygon at close range, as a higher level-of-detail representation. This is the size of the textured polygon relative to the light point size.</td>
</tr>
</tbody>
</table>
Texture - Index

Certain receiving systems support changing the light point visualization to a textured polygon at close range, as a higher level-of-detail representation. This is the texture palette index of higher level-of-detail texture (-1 if not used).

**Light Point Animation Attributes**

The table below describes the light point animation attributes.

The General tab contains the following attributes:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The light point animation name.</td>
</tr>
<tr>
<td>Animation Type</td>
<td>The type of animation (Flashing Sequence, Rotating, Strobe or Morse Code).</td>
</tr>
<tr>
<td>Period</td>
<td>The animation period (seconds).</td>
</tr>
<tr>
<td>Phase Delay</td>
<td>The animation phase delay (seconds).</td>
</tr>
<tr>
<td>Enabled Period</td>
<td>The part of the animation period where the light point is switched on (seconds).</td>
</tr>
<tr>
<td>Rotation Axis – i</td>
<td>The i (x) value of the rotation axis.</td>
</tr>
<tr>
<td>Rotation Axis – j</td>
<td>The j (y) value of the rotation axis.</td>
</tr>
<tr>
<td>Rotation Axis – k</td>
<td>The k (z) value of the rotation axis.</td>
</tr>
<tr>
<td>Flashing</td>
<td>Enabled if the light point is flashing.</td>
</tr>
<tr>
<td>Rotating</td>
<td>Enabled if the light point is rotating.</td>
</tr>
<tr>
<td>Counter Clockwise</td>
<td>Enabled if the light point rotation is defined counter-clockwise.</td>
</tr>
</tbody>
</table>

The Morse Code tab contains the following attributes:

| Morse Code Timing     | The type of Morse code timing (Standard or Farnsworth). |
| Word Rate             | The Morse code word rate in seconds.                  |
| Character Rate        | The Morse code character rate in seconds.             |
| Morse Code            | The Morse code text.                                  |

The Sequence tab contains the following attributes:

| Sequence Number       | The number of the sequence state for which the attributes below are modified. |
| State                 | The state of the current sequence (On, Off or Color Change). This means that this sequence will turn on, turn off or change the color of the light point. |
| Duration              | The duration in seconds of this sequence.             |
| Color Change Index    | The color palette index of the sequence color (0 – 1023, -1 if none). This color is always indexed, even if the model is in RGB mode. |
Color Change Intensity | The intensity of the primary color (0 – 127). This color is always indexed, even if the model is in RGB mode.

Sequence states can be added and removed with the Add and Remove buttons in the Sequence tab of the Light Point Animation attributes. New sequence states are added to the end of the list. When clicking the Remove button, the current sequence state, as specified by the Sequence Number drop-down menu, will be removed. The following sequence states will be re-numbered to achieve a continuous list of sequence state indices.

A light point appearance or animation index can be changed by selecting the appearance or animation list row, choosing Change Index from the Edit menu and specifying a new index. An existing appearance or animation can be removed by selecting the appearance or animation and choosing Remove from the Edit menu. To remove all unused appearances or animations, choose Remove Unused from the Edit menu.

A light point palette can be opened from disk by selecting Open Palette from the File menu and saved to disk by selecting Save Palette As from the File menu. Light point palette files have an RLP (Remo 3D Light point Palette) extension.

The general functionality of the file dialogs used in the light point palette is described in the File Dialogs section.
Shader Palette

This window presents the shader palette with all shaders defined for this model.

In the Shaders output, the index and the name of the shaders are presented. By clicking on the shaders, details about them are shown. There are two types of shaders; OpenGL Shading Language and Cg.

For the OpenGL Shading Language shaders, two types of shader program files can be added; vertex program files and fragment program files. Vertex and fragment program files can be added and removed with their respective Add and Remove buttons.

For the Cg shaders, one vertex program and one fragment program file can be specified. A vertex and a fragment profile number and vertex and fragment program entry points can also be specified.

Note that the actual modification (and corresponding undoable command) of the information in the Cg shader text fields will be performed when leaving a text field, e.g. focusing another part of the user interface or leaving the field by pressing Tab or Return.

Shader palettes can be saved and opened from the File menu. Shader palette files have an RSP (Remo 3D Shader Palette) extension. New shaders can be created with Ctrl+n or from the Edit menu by choosing New. To remove shaders from the shader palette, use
Remove from the Edit menu or use the Delete button on the keyboard. Unused shaders can also be removed by selecting Remove Unused from the Edit menu. To change the index for a shader, Change Index from the Edit menu can be used. A shader’s name can be changed by selecting Change Name from the Edit menu.

If shaders need to be reloaded from disk, Refresh from the Edit menu can be used.

When loading a model with shaders, the REMO3D_FILE_PATH environment variable is used if the shader files aren't found directly.

The general functionality of the file dialogs used in the shader palette is described in the File Dialogs section.
Command List

This section describes the command list which lists the executed undoable and redoable commands with the latest command at the top of the list.

If a command has been undone, this is visualized by putting its command name within parentheses and changing the font to italic. These parentheses are removed if the same command is redone.

Script commands are commands that are only present when a script is being recorded. This makes it possible to undo commands that are recorded to a script but that are not usually undoable.

When executing a macro, a Macro Command will be added to the command list. The name of the macro will be displayed after the Macro Command.

The command list window is non-modal which means that it is possible to open the command list and continue working with the rest of the Remo 3D graphical user interface.
Preview DOF

This section describes the DOF preview dialog which lets you preview the different transformations of a selected DOF node.

The DOF preview dialog only works on one DOF node at a time. The different transformation types for a DOF node can also be previewed only one at a time. The current transformation type is chosen by one of the radio buttons in the upper part of the dialog. A transformation type radio button is only active if the corresponding transformation’s minimum value is smaller than or equal to its maximum value, its current value is between the minimum and the maximum value and its increment value is not zero.

To preview the chosen transformation the slider below the transformation type radio buttons can be moved. Its left-most position corresponds to the minimum value of the chosen transformation and the right-most position represents the maximum value. When a transformation type is chosen, the slider will be located at its current value. When moving the slider and releasing the mouse button it will return to the current value. While previewing the current selection visualization will be hidden.

The DOF preview dialog is non-modal which means that it is possible to open the DOF preview dialog and continue working with the rest of the Remo 3D graphical user interface.
Preview Animation

This section describes the animation preview dialog which lets you preview the sequence animation of a selected group node.

The animation preview dialog will only open if exactly one group node with an animation sequence is selected. Click the Start button to start the animation and the Stop button to cancel the current animation. Forward and backward sequences will animate correctly. The Loop Count, Loop Duration, Last Frame Duration and Swing attributes of the selected group node controls the animation. An animation with Loop Count above 0 will loop until finished and then stop. If the Loop Count is 0 it will repeat until Stop is clicked. While animating the current selection visualization will be hidden.
Settings

This section describes the different ways that the user can customize the appearance and behavior of Remo 3D.

Preferences

This dialog lets the user select a number of preferences for the Remo 3D application. These preferences can be saved to disk and will automatically be loaded the next time Remo 3D is started.

In the Viewports tab, the number of viewports and their arrangement can be selected. Viewports are separate views of the same model. The user can choose from one, two, three or four viewports.

In the Control tab, the sensitivities with which panning, zooming and rotation of the model view are performed can be selected. When pan or zoom sensitivity is set to Auto, an attempt to set a suitable sensitivity depending on the size of the model is made. This automatic sensitivity calculation can be repeated later by pressing space in order to zoom to the model. Note that the sensitivity can also be controlled with the Sensitivity dialog available from the View menu.

In this tab, the selection tolerance in pixels can also be set. When selecting by clicking in the model view, this tolerance is added to the indicated window pixel in order to make it easier to aim at the primitives that are to be selected.

Finally, the control tab lets the user specify whether changing the viewpoint to top, bottom, front, back, right or left should automatically zoom to the model’s extent or not.
In the Gizmos tab, the user can specify if gizmos should be shown for tools supporting them. The size of the gizmos can also be specified, defined as the number of pixels the gizmos will occupy on screen. A check box for enabling uniform scaling when using the scale gizmo is also available. When this is checked the scale will be the same in both directions when scaling with the planes between the axes.

In the Misc tab, the user can specify the tolerance with which certain geometry operations are performed. In the current version of Remo 3D, this tolerance is used when slicing geometry, performing cylindrical or spherical texture mapping and when merging polygons. The tolerance is specified in model units, e.g. meters.

The maximum number of undo commands to keep can be specified in the Max Commands input. This setting can have an impact on the memory usage of Remo 3D. Setting it to a high value will make Remo 3D use more memory.

The Screen Capture Size inputs are used to specify the size of the screenshots captured with Capture Screen from the Viewport Menu. The image will be cropped if the specified aspect ratio differs from the viewport aspect ratio. If any or both of the specified width and height are 0, the screen capture will have the exact size of the viewport.

The Parameter Precision sets the number of decimals to use for floating-point numbers when written in text, e.g. in Select By Attributes or when recording a script.

The user can also specify whether to convert values representing N/A (not available) to zero or not. Certain values in an OpenFlight® file are sometimes specified as -1e+008 (-100 000 000), which is meant to represent a N/A value. To avoid actually interpreting these values as -1e+008 and instead convert them to 0 (zero) when opening an OpenFlight® file, this check box can be set. The following values are affected by the N/A setting:
- The from-point in a translating matrix transformation
- The origin, point on X axis and point in XY plane of a DOF node

The Automatic Texture Reload check box controls if textures should be automatically reloaded when changed on disk.

The Automatic External Reference Reload check box controls if external references should be automatically reloaded when changed on disk.

The Automatic Shader Reload check box controls if shaders should be automatically reloaded when changed on disk.

The Always Save To Latest Version check box is used to enable or disable saving to the latest OpenFlight® version. If it is checked, the Save command will save to the latest supported OpenFlight® version, otherwise it will save to the version the file was opened with or the last version it was saved to.

By enabling or disabling the Activate Splash Screen check box, the user can control whether the Remo 3D splash screen will be shown or not when starting Remo 3D.

The Load External References check box controls whether externally referenced models are to be loaded from the start or not. If disabled, they won’t be loaded directly when loading the parent model, when creating them with the Create External Reference tool or when specifying their model file name in the Attributes tool. Separate externally referenced models (per file name) can be enabled in the External References dialog available from the File menu.

The preferences can be reset to the default preferences by clicking the Defaults button. To apply the given preferences, click the Apply button. The preferences can be saved to disk by clicking the Save button, which will also apply the preferences. Saved preferences will automatically be loaded the next time Remo 3D is started.

**Viewport Menu**

This menu is reached by choosing Viewport from the View menu, by clicking Ctrl + Alt + right mouse button in a viewport or by pressing the Menu keyboard button in a viewport. It contains a number of settings for the currently focused viewport.

The table below describes the contents of the viewport menu.

<table>
<thead>
<tr>
<th>Viewpoint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perspective</td>
<td>Draws model in perspective view. This will be indicated with the letter P in the top right corner of the viewport.</td>
</tr>
<tr>
<td>Orthographic</td>
<td>Draws model in orthographic view, i.e. with a straight projection. This will be indicated with the letter O in the top right corner of the viewport.</td>
</tr>
<tr>
<td>Top (Along -Z)</td>
<td>Moves viewpoint to the Z axis, viewing in a negative Z direction and zooming out to the model’s extent. Most models are created so that this will show the top of the model.</td>
</tr>
<tr>
<td>Bottom (Along Z)</td>
<td>Moves viewpoint to the Z axis, viewing in a positive Z direction and zooming out to the model’s extent. Most models are</td>
</tr>
</tbody>
</table>
created so that this will show the bottom of the model.

<table>
<thead>
<tr>
<th>Front (Along -Y)</th>
<th>Moves viewpoint to the Y axis, viewing in a negative Y direction and zooming out to the model’s extent. Most models are created so that this will show the front of the model.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Back (Along Y)</td>
<td>Moves viewpoint to the Y axis, viewing in a positive Y direction and zooming out to the model’s extent. Most models are created so that this will show the back of the model.</td>
</tr>
<tr>
<td>Right (Along -X)</td>
<td>Moves viewpoint to the X axis, viewing in a negative X direction and zooming out to the model’s extent. Most models are created so that this will show the right side of the model.</td>
</tr>
<tr>
<td>Left (Along X)</td>
<td>Moves viewpoint to the X axis, viewing in a positive X direction and zooming out to the model’s extent. Most models are created so that this will show the left side of the model.</td>
</tr>
<tr>
<td>Above Grid</td>
<td>Moves viewpoint toward and above the current grid.</td>
</tr>
<tr>
<td>Below Grid</td>
<td>Moves viewpoint toward and below the current grid.</td>
</tr>
<tr>
<td>Store View</td>
<td>Stores the current view for later use.</td>
</tr>
<tr>
<td>Recall View</td>
<td>Recalls the stored view.</td>
</tr>
<tr>
<td>Fit To Selection</td>
<td>Fits the view to the current selection, changing the zoom and point of rotation.</td>
</tr>
<tr>
<td>Fit In View</td>
<td>Moves the viewpoint backwards so that the complete model is visible.</td>
</tr>
</tbody>
</table>

**Appearance**

<table>
<thead>
<tr>
<th>Scribed</th>
<th>Draws model with wireframe polygons on top of solid polygons.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solid</td>
<td>Draws model with solid polygons.</td>
</tr>
<tr>
<td>Wireframe</td>
<td>Draws model with wireframe polygons.</td>
</tr>
<tr>
<td>Draw Wireframe Backface</td>
<td>Draws the back-face of the wireframe overlay.</td>
</tr>
<tr>
<td>Texture On</td>
<td>Toggles texture visualization on/off.</td>
</tr>
<tr>
<td>Shader On</td>
<td>Toggles shader visualization on/off.</td>
</tr>
</tbody>
</table>

**Lighting**

<table>
<thead>
<tr>
<th>Lighting On</th>
<th>Toggles the lighting on/off.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head Light</td>
<td>Switches lighting mode to head light, i.e. the light will emanate from the viewpoint.</td>
</tr>
<tr>
<td>Sky Light</td>
<td>Switches lighting mode to sky light, i.e. the light will emanate from above.</td>
</tr>
</tbody>
</table>

**Grid**

<table>
<thead>
<tr>
<th>Show Grid</th>
<th>Enables or disables the grid. When the grid is enabled this will be indicated with a small grid symbol in the top left corner of the viewport.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Align To XY</td>
<td>Aligns the grid to the XY-plane with the origin in (0,0,0). Note that this can be accomplished by clicking the XY button in the coordinates and selection field as well.</td>
</tr>
<tr>
<td>Align To XZ</td>
<td>Aligns the grid to the XZ-plane with the origin in (0,0,0). Note</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Align To YZ</td>
<td>Aligns the grid to the YZ-plane with the origin in (0,0,0). Note that this can be accomplished by clicking the YZ button in the coordinates and selection field as well.</td>
</tr>
<tr>
<td>Align To Vertices</td>
<td>Aligns the grid to three vertices (e.g. three vertices of a polygon; cancel by choosing again). None of the three indicated coordinates are propagated to the active tool.</td>
</tr>
<tr>
<td>Align To Polygon (Face)</td>
<td>Aligns the grid to a selected polygon.</td>
</tr>
<tr>
<td>Translate Origin</td>
<td>Translates the grid origin to an indicated coordinate (cancel by choosing again). The indicated coordinate is not propagated to the active tool.</td>
</tr>
<tr>
<td>Z-buffer</td>
<td>Draws the grid using normal Z buffer with respect to other geometry.</td>
</tr>
<tr>
<td>Always On Top</td>
<td>Draws the grid on top of other geometry.</td>
</tr>
<tr>
<td>Never On Top</td>
<td>Draws the grid behind other geometry.</td>
</tr>
<tr>
<td>Increase Grid Size</td>
<td>Increases the grid size by one grid line.</td>
</tr>
<tr>
<td>Decrease Grid Size</td>
<td>Decreases the grid size by one grid line.</td>
</tr>
<tr>
<td>Double Grid Size</td>
<td>Increases the grid size to twice the current number of grid lines.</td>
</tr>
<tr>
<td>Half Grid Size</td>
<td>Decreases the grid size to half the current number of grid lines.</td>
</tr>
<tr>
<td>Grid Spacing...</td>
<td>Lets the user select the distance between each grid line. This input dialog is non-modal which means that it is possible to open the dialog and continue working with the rest of the Remo 3D graphical user interface.</td>
</tr>
<tr>
<td>Grid Offset...</td>
<td>Lets the user select an offset with which the grid should be translated in its orthogonal direction. This input dialog is non-modal which means that it is possible to open the dialog and continue working with the rest of the Remo 3D graphical user interface.</td>
</tr>
</tbody>
</table>

### Hiding

- **Hide Selected**: Hides the selected geometry.
- **Isolate Selected**: Hides all geometry but the selected.
- **Unhide Selected**: Unhides the selected geometry.
- **Unhide All**: Unhides all geometry.

### Capture Screen...

- Captures a screenshot of the current viewport. The size of the screenshot is determined by the size specified in the Preferences dialog.

### Viewport Preferences...

- Opens the viewport preferences window described in the Viewport Preferences section.
Viewport Preferences

This dialog lets the user select a number of preferences for the current viewport. These preferences can be saved to disk and will automatically be loaded the next time a viewport is created.

In the Background tab, top and bottom colors can be chosen to form a color gradient or a solid background color. The colors can be specified as red, green and blue or hue, saturation and value (lightness). Preset background colors can be chosen by clicking the Sky, Gray, Black or White button.

The Frustum tab lets the user specify the viewing frustum including the near clip and far clip distances as well as the field-of-view angle. The default is to automatically calculate the near clip and far clip distances from the extent of the visible geometry.

Small feature culling can also be enabled and disabled by checking and unchecking the Small Feature Culling check box. The pixel size threshold for the small feature culling can also be specified. When enabled, small feature culling will not display objects that occupy fewer pixels on screen than the specified threshold.
In the Light tab, ambient, diffuse and specular colors can be specified for the lighting of the scene. The colors can be specified as red, green and blue or hue, saturation and value (lightness).

The Visual Cues tab contains settings for the grid, the visualization of large vertices, normals and the orientation axes. The grid can be customized by specifying the colors of the grid lines and axes, as well as the spacing between the grid lines and which line number that is to be drawn with a thick line style.

Large vertex visualization can be customized by specifying inner and outer colors as well as the size of the vertex markings.

The visualization of normals can be customized with color and scale. It’s also possible to only show normals for the current selection and toggle if normals should be visible for replications, instances and external references.

The orientation X, Y and Z axes can be visualized in one of the model view’s four corners. It can also be turned off.

The viewport preferences can be reset to the default preferences by clicking the Defaults button. To apply the given preferences, click the Apply button. The viewport preferences can also be saved to disk by clicking the Save button, which will also apply the viewport preferences. Saved viewport preferences will automatically be used for all viewports created, e.g. when the next time Remo 3D is started, when a new model is opened or when the number of viewports is changed.
License Handling

There are three types of licenses in Remo 3D; node-locked, USB dongle, and floating. Node-locked and USB dongle licenses can be time-limited for evaluation of the product.

Node-locked and USB dongle licenses

The node-locked and USB dongle licenses in Remo 3D are based on a hardware fingerprint and a license key. The hardware fingerprint is generated either from the MAC address of the Ethernet card in the computer which is running Remo 3D or from the serial number of a USB memory stick working as a dongle, depending on which type of license you purchase. To obtain a valid license key, the hardware fingerprint must be included in the purchase form on www.remograph.com or sent to sales@remograph.com.

To unlock a demo version of Remo 3D, the license key should be entered in the Purchase dialog which can be found under the Help menu. This dialog is also where the hardware fingerprint can be obtained. When the License File radio button is activated, depending on which type of license you intend to purchase, click the Pick Network Adapter or Pick USB Dongle button. Select a network adapter (choose a network adapter that will be active when running Remo 3D) or a connected USB memory stick for which the hardware fingerprint should be generated. If a USB memory stick is selected, this device will be your USB dongle and will have to be connected to the computer running Remo 3D.

Note that in Microsoft® Windows® 10/8/7, the USB memory stick also has to be mounted in order to generate a hardware fingerprint and verify the license when running Remo 3D. However, when inserting a USB memory stick it is usually mounted automatically.

The license key will by default be stored in the same folder as the Remo 3D executable file. This can be changed by entering a filename in which the license should be stored. To use an alternative location of the license file, i.e. not alongside the Remo 3D executable file, the Purchase dialog will also need to be used.

![Purchase Remo 3D dialog](image)
file and named license.dat, use the environment variable REMO3D_LICENSE_FILE to specify the license file to use.

Floating licenses

When using a floating license, a license server is needed. A license server program is included in the Remo 3D download and is called remoLicenseServer. The server program is available in Linux and Microsoft® Windows® 10/8/7.

The remoLicenseServer program has the following syntax:

```
remoLicenseServer -g | -u | [-i] -l <number of licenses> -n <license key> [-f <license file>] [-p <port number>] [-o <log file>] [-v]
-g Prints all possible hardware fingerprints to the console.
-u Stop and uninstall Windows Service (only available in Microsoft Windows).
-i Install and start Windows Service (only available in Microsoft Windows).
-l Specifies the number of licenses the server should use.
-n Specified the license key.
-f Specifies the path and name of the license file written and used internally.
-p Specifies the port number.
-o Specifies a log file (default is stdout).
-v Verbose mode.
```

The server is unlocked with a license key generated from a hardware fingerprint and the number of licenses. A hardware fingerprint can be obtained by specifying -g to remoLicenseServer, then a list of hardware fingerprints will be output to the console. The license key is locked to a certain number of licenses, so when starting remoLicenseServer, make sure the correct number of licenses is specified.

In Microsoft® Windows®, the license server can be run as a service. Install Remo 3D normally on the server computer and run remoLicenseServer.exe from a command prompt, specifying the -i switch. The service will then be installed and started automatically with the options that you specify with the other parameters. To stop and uninstall the service, specify only the -u switch to remoLicenseServer.exe. It is recommended to send the server log messages to a specific log file with the -o switch, since this is the only way of seeing what the service does. It is also recommended to use the -v switch for verbose output. A license file for internal use will be written to the same folder as where the remoLicenseServer executable resides. If another destination for this file is preferred, e.g. due to privilege issues, this can be specified with the -f switch.

When running Remo 3D, the server needs to be specified in the purchase dialog, check the License Server radio button, enter the server name in the License Server field and the server port in the License Server Port.
The license server and port will by default be stored in the same folder as the Remo 3D executable file. This can be changed by entering a filename in which the license server and port should be stored. To use an alternative location of the license server file, i.e. not alongside the Remo 3D executable file and named licenseserver.dat, use the environment variable REMO3D_LICENSE_SERVER_FILE to specify the license server file to use.

The license server and port can also be specified with environment variables by setting REMO3D_LICENSE_SERVER to the server name and REMO3D_LICENSE_SERVER_PORT to the server port.

**Time limited license**

A time limited license can be used for evaluation of Remo 3D. You need to generate a node-locked or dongle hardware fingerprint to be used for the time limited license, as described in the Node-locked and USB dongle licenses section above. Enter this hardware fingerprint when requesting a demo license from our website at www.remograph.com and you will instantly receive a license key.

When you have received the time limited license key, press the License File radio button in the Purchase dialog and enter the key in the License Key field. Remo 3D will then be unlocked for 30 days from the day when the license key was generated. When starting Remo 3D, a message box will appear that states how long time is left for the time limited license. If the time is up, the message box will state for how long the time limited license has been exceeded.

The license key will by default be stored in the same folder as the Remo 3D executable file. This can be changed by entering a filename in which the license should be stored. To use an alternative location of the license file, i.e. not alongside the Remo 3D executable file and named license.dat, use the environment variable REMO3D_LICENSE_FILE to specify the license file to use.
Installed Contents

Remo 3D is installed with a comprehensive built-in help system, a tutorial complete with necessary models and textures, sample model files, scripts and macros.

Built-in Help System

The help system is described in the Help chapter.

Tutorial

With the tutorial you will learn how to use a majority of the tools in Remo 3D. The models and textures necessary for performing the tutorial steps, as well as the tutorial document, are available in the tutorial folder.

Samples

The samples folder contains the following sample material:

Sample Models

The samples/EdgedSign, samples/Simthetiq and samples/Cantaloupe folders contain some nice OpenFlight® models from EdgedSign, Simthetiq Inc. and cantaloupe|3d.

Scripts

The samples/scripts folder contains a number of sample command-line scripts described in the Script Samples section.

Macros

The samples/macros folder contains sample macros described in the Macro Samples section.
Help

The Remo 3D help will open up in a separate window. If not closed manually, it will remain open until Remo 3D is closed.

The help system is based on the Eclipse platform running in a Java™ environment. Therefore, it requires that Java™ is installed on the computer, with a path set to the java executable. If not, the help window will not open. However, the same information is available in the user’s guide that is installed with Remo 3D.

You can navigate through the contents in the left pane of the help window, use the Search functionality at the top of the window or the Index available in the bottom left part of the window. You can also print pages and set personal bookmarks that will be saved until the next session.
Scripts

All tasks available manually in Remo 3D are possible to automate through a script. The script language is based on Lua (www.lua.org) with an addition of several Remo 3D-specific functions.

Script Environment

A script can be written in your favorite text editor or in the Script Editor available from the Script menu. Either way, this is where you execute the script and where you can see the results.

A script can also be executed directly from the command-line by giving it as a parameter to the remo3d executable file, like this:

remo3d <script filename> <arguments>

or for instance like this in Microsoft® Windows®:

"C:\Program Files\Remo3D\remo3d.exe" <script filename> <arguments>

In Microsoft® Windows®, a separate console window will open when executing a script from the command-line. To be able to see any printed information in this console, the execution ends with a pause where the user has to press a key to continue. When executing multiple scripts from e.g. a batch file this is not ideal, so to avoid having to press a key between each script, an environment variable called
REMO3D_SCRIPT_CONFIRMATION can be set to OFF (or off) by doing the following (any other value or no environment variable at all will result in confirmations):

Start -> Settings -> Control Panel -> System -> Advanced -> Environment Variables -> New

Enter REMO3D_SCRIPT_CONFIRMATION as variable name and OFF as variable value.

To direct the text output from a script to a log file rather than the console, use the REMO3D_LOG environment variable to point at a log file, e.g. set it to C:\Temp\log.txt. The log file doesn’t have to exist before the script is executed. When REMO3D_LOG is defined, no console window will open while running a command-line script in Microsoft® Windows®.

When executing a script from the command-line, the arg table will be filled with the given command-line arguments, just like with normal Lua command-line usage. The remo3d executable path goes into index -1 of the arg table, the script filename goes into index 0, the first script argument goes into index 1, and so on. The following command line would e.g. generate the following arg table:

remo3d script.lua a b

arg[-1] = "remo3d"
arg[0] = "script.lua"
arg[1] = "a"
arg[2] = "b"

A script executed from the Script Editor will operate in the current Remo 3D environment, i.e. the currently open models will be available to the script and any immediate commands from the script will be operating on the current model. A script executed from the command-line will have to open or create a model explicitly.

Script Samples

The Remo 3D installation contains a number of script samples. See the Script Environment section for information about how to execute a command-line script. The samples/scripts folder contains the following command-line scripts:
**convert.lua**

Remo 3D command-line script for converting between all model file formats supported by Remo 3D.

**Usage:**

```
convert.lua <input filename> <output filename>
```

- `<input filename>` is the source model, has to be OpenFlight or format supported for import in Remo 3D.
- `<output filename>` is the target model, has to be OpenFlight or format supported for import in Remo 3D.

---

**flt2txt.lua**

Remo 3D command-line script for printing nodes as an indented scene graph in a new text file.

**Usage:**

```
flt2txt.lua <model filename> <text filename>
```

- `<model filename>` is the input OpenFlight model file.
- `<text filename>` is the output text file.

---

**fltConvert.lua**

Remo 3D command-line script for converting between OpenFlight versions.

**Usage:**

```
fltConvert.lua <input filename> <output filename> <version>
```

- `<input filename>` is the source OpenFlight file.
- `<output filename>` is the target OpenFlight file.
- `<version>` is the new OpenFlight version as an integer, e.g. 1600 or 1570.

---

**listExternals.lua**

Remo 3D command-line script for printing the file names of all external references in a given model file.

**Usage:**

```
listExternals.lua <model filename>
```

- `<model filename>` is the input OpenFlight model file.
listTextures.lua

Remo 3D command-line script for printing the file names of all textures in a given model file.

Usage:
listTextures.lua <model filename>

<model filename> is the input OpenFlight model file.

setTextureFilters.lua

Remo 3D command-line script for setting minification and magnification filters on a given texture image.

Usage:
setTextureFilters.lua <texture filename> <minFilter> <magFilter> [magFilterAlpha] [magFilterColor]

<texture filename> is the texture image file, has to be supported by Remo 3D.

<minFilter> is the minification filter given as one of the strings in the corresponding Minification Filter combo-box in the Texture Attributes dialog in Remo 3D, e.g. MIPMAP_TRILINEAR.

<magFilter> is the magnification filter given as one of the strings in the corresponding Magnification Filter combo-box in the Texture Attributes dialog in Remo 3D, e.g. BILINEAR.

[magFilterAlpha] and [magFilterColor] are optional, see above for instructions. Note that if magFilterColor is given, then magFilterAlpha must be given as well.

The samples/macros folder contains a number of sample macros described in the Macro Samples section.

Script Reference

The following sections describe all Remo 3D-specific functions that you can use in combination with the functions from all standard Lua libraries. For more information about the functionality behind the functions, see the corresponding descriptions of menu options or tools.

All actions available to the user interactively in Remo 3D can be performed in a script with corresponding functions. These actions include tools or palette operations that modify the model, left- or right-shifting of nodes, all types of selection operations, cutting and pasting of geometry, moving of the pivot point and setting the parent node. The retrieval (get/report) functions available will report properties, characteristics or attributes that are visible to the user in the Remo 3D user interface. In additions to these functions, there are functions for traversing the tree structure (remo.traverse) and converting the current selection to another mode (remo.convertSelection).

These are some general facts to be aware of when writing scripts for Remo 3D:

- The “remo.” prefix is used with every script function.
While indices generally begin with 1 in Lua, the Remo 3D-specific indices are zero-based.
- File paths given as strings in a script, e.g. to functions such as `remo.openModel`, can use both forward slashes (/) and backslashes (\) as separators, regardless of the current operating system. Note that backslashes have to be represented as duplicates (\\) in hard-coded strings.
- It is prohibited to create or import multiple models with the same name, since you wouldn’t be able to switch to them in a deterministic way.
- Be sure to use the `print` function in Lua to achieve text output from your script, the `io.write` function will not produce any text in the Script Editor output dialog or the console. Note that the `print` function always ends with a new-line, to avoid this the `REMO3D_NO_LINE_FEED` environment variable can be used.
- The version of the Lua library shipped with Remo 3D is 5.2.3.

**General Script Functions**

This section describes all general and miscellaneous script functions.

---

**remo.getVersion()**

Reports the current Remo 3D version.

**returns:** The Remo 3D version as a number, e.g. 2.0 or 2.3.

**remo.changePivot(x, y, z)**

Changes the position of the pivot point to a given point.

**x, y, z:** The coordinate at which the new pivot should be located.

**remo.getPivot()**

Reports the current pivot point.

**returns:** The current pivot point as three values (x,y,z).

**remo.getNumChildren()**

Reports the number of children for a selected node. One and only one selected node is allowed. Vertices will not be counted, so the children of polygons and light points will only be other polygons (sub-faces) or light points.

**returns:** The number of children for the selected node as an integer.

**remo.getNumVertices()**

Reports the number of vertices for a selected polygon or light point node. One and only one selected polygon or light point node is allowed.

**returns:** The number of vertices for the selected polygon or light point node.
remo.executeMacro(scriptFile, paramName1, paramValue1, ...)

Executes a macro with parameters.

scriptFile: The script file to execute.

paramNameX: The name of the parameter number X.

paramValueX: The value for the parameter number X.

remo.traverse(preCallback, postCallback)

Traverses the selected nodes recursively (depth-first), executing given callback functions for each node. Vertices are also traversed. The selection is cleared after the traversal. See below for details. (*)

preCallback: The pre-callback function (as a function first-class value). The node being traversed is selected and this callback is executed before traversing the sub-tree beneath the node. Can be specified as nil, but at least one of the pre- or post-callback functions must be defined. See below for details. (*)

postCallback: The post-callback function (as a function first-class value). The node being traversed is selected and this callback is executed after traversing the sub-tree beneath the node. Can be specified as nil, but at least one of the pre- or post-callback functions must be defined. See below for details. (*)

remo.setTolerance(tolerance)

Sets the tolerance used by different tools and script functions. This tolerance is described in the Preferences section.

tolerance: The desired tolerance.

remo.getTolerance()

Returns the tolerance used by different tools and script functions. This tolerance is described in the Preferences section.

returns: The current tolerance.

(*) The pre- and post-callback functions used in the remo.traverse function are given the selection mode of the current node as a string, one of the following:

"VERTEX"
"POLYGON"
"OBJECT"
"GROUP"
"LOD"
"DOF"
"EXTERNAL"
"SWITCH"
"LIGHTPOINT"
"LIGHTPOINTSYSTEM"
"HEADER"
The pre-callback function must return a boolean specifying whether the traversal should continue downwards or not. Note that the post-callback function is executed on a node whose pre-callback returned false, even if the traversal below wasn’t performed. Also note that if the current model is changed in the callback functions (e.g. another model is temporarily opened), it has to be changed back with `remo.setCurrentModel` before leaving the function.

Below is an example that prints the type and name of each node, indented according to the tree structure. Vertex names are avoided by returning false from the pre-callback after printing the names of polygon nodes and light point nodes.

```lua
indent = 0

function preCallback(selectionMode)
    print(string.rep("  ", indent) .. selectionMode .. ": " .. remo.getAttributes("Name"))
    indent = indent + 1
    return (selectionMode ~= "POLYGON" and selectionMode ~= "LIGHTPOINT")
end

function postCallback(selectionMode)
    indent = indent - 1
end

remo.selectAll("HEADER")
remo.traverse(preCallback, postCallback)
```

**File Script Functions**

This section describes script functions related to model files.

---

**remo.newModel(filename)**

Creates a new (unsaved) OpenFlight model and makes it the current model.

**filename**: The filename of the new (unsaved) OpenFlight model.

**returns**: true if the new operation was successful.

---

**remo.openModel(filename)**

Opens an OpenFlight model and makes it the current model.

**filename**: The filename of the OpenFlight model that should be opened.

**returns**: true if the open operation was successful.

---

**remo.revertModel()**

Reverts the current OpenFlight model.

**returns**: true if the revert operation was successful.
remo.saveModel(version)

Saves the current model to an OpenFlight file with the current model's filename.

version (optional, default latest supported): The version of the saved file as an integer, e.g. 1570 or 1620.

returns: true if the save operation was successful.

remo.saveModelAs(filename, version)

Saves the current model to an OpenFlight file with a given filename.

filename: The filename to which the OpenFlight model should be saved.

version (optional, default latest supported): The version of the saved file as an integer, e.g. 1570 or 1620.

returns: true if the save operation was successful.

remo.isModelUnsaved()

Reports whether the current model is unsaved or not.

returns: true if the current model is unsaved.

remo.importModel(sourceFilename, targetFilename, textureOutputPath, shaderOutputPath, shaderOutputBaseName, externalOutputPath)

Imports geometry to a new (unsaved) model and makes it the current model.

sourceFilename: The filename from which to import.

targetFilename: The filename of the new model with the imported geometry.

textureOutputPath (optional, default empty): The output path where embedded textures will be written. If not given or empty, no embedded textures will be written to disk.

shaderOutputPath (optional, default empty): The output path where embedded shaders will be written. If not given or empty, no embedded shaders will be written to disk.

shaderOutputBaseName (optional, default empty): The base name of embedded shaders written to disk. If not given or empty, no embedded shaders will be written to disk.

externalOutputPath (optional, default empty): The output path where externally referenced files will be written. If not given or empty, no externally referenced files will be written to disk.

returns: true if the import operation was successful.
remo.exportModel(filename)
Exports the current model to a file. The file extension determines the output file format.
filename: The filename to which to export.
returns: true if the export operation was successful.
remo.closeModel()
Closes the current model.
remo.setCurrentModel(filename)
Sets the current model to the model with the given filename. If no such model exists, this does nothing.
filename: The filename of the model that is to be made current.
returns: true if the current model switch succeeded.
remo.getCurrentModelFilename()
Reports the filename of the current model.
returns: The current model filename as a string.
remo.saveAllModels()
Saves all open models.
remo.closeAllModels()
Closes all open models.

**Clipboard/Edit Script Functions**

This section describes script functions related to the clipboard and general editing.

remo.cut()
Cuts the currently selected nodes to the clipboard.
remo.copy()
Copies the currently selected nodes to the clipboard.
remo.paste()
Pastes the contents of the clipboard under the current parent.
remo.delete()

Deletes the currently selected nodes.

remo.setParent()

Sets the parent to the currently selected node.

remo.changeParent()

Moves the currently selected nodes to the current parent.

remo.getParent()

Reports the name of the current parent node.

returns: The name of the current parent node as a string.

remo.shiftLeft()

Shift selected nodes one step to the left.

remo.shiftRight()

Shift selected nodes one step to the right.

remo.shiftFarLeft()

Shift selected nodes all the way to the left.

remo.shiftFarRight()

Shift selected nodes all the way to the right.

**Selection Script Functions**

This section describes script functions related to selecting geometry and nodes in different ways.
remo.selectAll(selectionMode)

Selects all primitives of a given mode.

selectionMode: The selection mode as a string, one of the following:
"VERTEX"
"EDGE"
"POLYGON"
"OBJECT"
"GROUP"
"LOD"
"DOF"
"EXTERNAL"
"SWITCH"
"LIGHTPOINT"
"LIGHTPOINTSYSTEM"
"HEADER"

returns: true if any primitive was selected.

remo.selectByName([table | name1, name2, ...], caseSensitive, addToSelection)

Selects all primitives with at least one specific name (give a table OR separate names as parameters)

table: A table of names, given as [name1, name2, ...

nameX: One of the names to search for; could be one (name1 only) or an arbitrary number of names.

caseSensitive (optional, default false): true if the match should be case sensitive.

addToSelection (optional, default false): true if the selected nodes should be added to the current selection.

returns: true if any primitive was selected.

remo.selectNone()

Clears all selections.
remo.invertSelection(selectionMode)

Inverts the selection of a given mode.

selectionMode: The selection mode as a string, one of the following:

"VERTEX"
"EDGE"
"POLYGON"
"OBJECT"
"GROUP"
"LOD"
"DOF"
"EXTERNAL"
"SWITCH"
"LIGHTPOINT"
"LIGHTPOINTSYSTEM"
"HEADER"

remo.selectByAttributes(selectionMode, conditionString, addToSelection, indexedLightPoint)

Selects all primitives matching a given condition of desired attributes (see below for details). (*)

selectionMode: The selection mode as a string, one of the following:

"VERTEX"
"POLYGON"
"OBJECT"
"GROUP"
"LOD"
"DOF"
"EXTERNAL"
"SWITCH"
"LIGHTPOINT"
"LIGHTPOINTSYSTEM"

conditionString: The string representing the expression for selecting nodes, more information below. (*)

addToSelection (optional, default false): true if selection should be added to the existing selection.

indexedLightPoint (optional, default false): true if light points should be of indexed type, false if inline.

returns: true if any primitive was selected.
remo.getNumSelectedPrimitives(selectionMode)

Reports the number of selected primitives of a given mode.

**selectionMode**: The selection mode as a string, one of the following:

- "VERTEX"
- "EDGE"
- "POLYGON"
- "OBJECT"
- "GROUP"
- "LOD"
- "DOF"
- "EXTERNAL"
- "SWITCH"
- "LIGHTPOINT"
- "LIGHTPOINTSYSTEM"
- "HEADER"

**returns**: The number of selected primitives of the given mode.

remo.getTotalNumSelectedPrimitives()

Reports the number of selected primitives of all modes.

**returns**: The total number of selected primitives.

remo.getSelectionNames()

Reports the names of all selected primitives.

**returns**: A table containing the names of the selected primitives.

remo.convertSelection(selectionMode)

Converts the selection to a given mode. E.g., if a number of objects and groups are selected and
**selectionMode** is "POLYGON", this will select all polygons beneath the objects and groups.

**selectionMode**: The selection mode as a string, one of the following:

- "VERTEX"
- "EDGE"
- "POLYGON"
- "OBJECT"
- "GROUP"
- "LOD"
- "DOF"
- "EXTERNAL"
- "SWITCH"
- "LIGHTPOINT"
- "LIGHTPOINTSYSTEM"

remo.selectSliverPolygons()

Selects all polygons in the model (or in the selected sub-tree) having an angle smaller than the
current tolerance between two edges. Those polygons are considered to be slivers. The current
tolerance is described in the Preferences section.
remo.selectByNumberOfVertices(number, selectFromCurrent)

Selects all polygons in the model (or in the selected sub-tree) having a given number of vertices.

number: The desired number of vertices.

selectFromCurrent (optional, default true): True if selection is to be performed within the already selected geometry.

remo.selectParents()

Selects the parents of the currently selected nodes.

remo.selectChildren()

Selects the child nodes of the currently selected nodes.

remo.selectPrevious()

Selects the previously selected nodes. This will essentially toggle between the two latest selections.

(*) The condition string given to the remo.selectByAttributes function needs some special attention:

- A condition is made up of a number of expressions connected with AND or OR (case insensitive) relations.
- An expression is made up of three parts; an attribute name given as it appears in the corresponding label in the attribute tool (case insensitive), a relation and a value (case insensitive for strings).
- A relation is either = (equal to), < (less than), > (greater than) or != (not equal to).

The following are examples of different condition strings:

"Coordinate x < 19 AND normal enabled = 1"

"billboard/alpha = Fixed / Alpha and draw mode = Solid one side"

There are special rules for the attribute names, described at the asterisk (*) in the Attribute Script Functions section.
Transformation Script Functions

This section describes script functions for transformation tools.

```
remo.move(x, y, z, absolute, moveToPoint, softSelectionActive, radius,
weightFunction, weightMode, freezeX, freezeY, freezeZ, applyMatrix)
```

Moves the currently selected geometry.

x, y, z: The coordinates for the movement.

absolute (optional, default false): true if the movement should be made absolute, false if relative.

moveToPoint (optional, default false): true if all selected nodes should be moved to the same point.

(if moveToPoint is set to true the coordinates will always be treated as absolute)

softSelectionActive (optional, default false): true if soft selection should be active.

radius (optional, default 10.0): The soft selection radius.

weightFunction (optional, default "LINEAR"): The soft selection weight function as a string, one of the following:
  "LINEAR"
  "SQUARE"
  "SQUAREROOT"
  "CUBE"
  "CUBEROOT"

weightMode (optional, default "WEIGHT_PER_VERTEX"): The soft selection weight mode as a string, one of the following:
  "WEIGHT_PER_VERTEX"
  "WEIGHT_PER_PRIMITIVE"

freezeX, freezeY, freezeZ (optional, default false): Set to true to freeze movement along certain axes.

applyMatrix (optional, default false): Set to true to apply as transformation matrix.
remo.rotate(angle, axisX, axisY, axisZ, normal, softSelectionActive, radius, weightFunction, weightMode, applyMatrix)

Rotates the currently selected geometry.

angle: The angle to rotate in degrees.

axisX, axisY, axisZ: The axis to rotate around.

normal: true if the normals should be rotated.

softSelectionActive (optional, default false): true if soft selection should be active.

radius (optional, default 10.0): The soft selection radius.

weightFunction (optional, default "LINEAR"): The soft selection weight function as a string, one of the following:
"LINEAR"
"SQUARE"
"SQUAREROOT"
"CUBE"
"CUBEROOT"

weightMode (optional, default "WEIGHT_PER_VERTEX"): The soft selection weight mode as a string, one of the following:
"WEIGHT_PER_VERTEX"
"WEIGHT_PER_PRIMITIVE"

applyMatrix (optional, default false): Set to true to apply as transformation matrix.
remo.scale(scaleX, scaleY, scaleZ, softSelectionActive, radius, weightFunction, weightMode, applyMatrix)

Scales the currently selected geometry.

scaleX: The scale along the x-axis.
scaleY: The scale along the y-axis.
scaleZ: The scale along the z-axis.
softSelectionActive (optional, default false): true if soft selection should be active.
nradius (optional, default 10.0): The soft selection radius.

weightFunction (optional, default "LINEAR"): The soft selection weight function as a string, one of the following:
  "LINEAR"
  "SQUARE"
  "SQUAREROOT"
  "CUBE"
  "CUBEROOT"

weightMode (optional, default "WEIGHT_PER_VERTEX"): The soft selection weight mode as a string, one of the following:
  "WEIGHT_PER_VERTEX"
  "WEIGHT_PER_PRIMITIVE"

applyMatrix (optional, default false): Set to true to apply as transformation matrix.
remo.addTransformationMatrix(type, ...)
(rest of parameters depending on type as follows:)
TRANSFER:
..., fromX, fromY, fromZ, deltaX, deltaY, deltaZ, index)
SCALE:
..., centerX, centerY, centerZ, scaleX, scaleY, scaleZ, index)
ROTATE_ABOUT_EDGE:
..., point1X, point1Y, point1Z, point2X, point2Y, point2Z, angle, index)
ROTATE_ABOUT_POINT:
..., centerX, centerY, centerZ, axisX, axisY, axisZ, angle, index)
GENERAL_MATRIX:
..., m00, m01, m02, m03,
    m10, m11, m12, m13,
    m20, m21, m22, m23,
    m30, m31, m32, m33, index)

Applies a transformation matrix of a given type to the selected node. This will only work with a single (one and only one) node selected. Transformation matrices can be applied to group nodes, object nodes, LOD nodes, external references, light point system nodes and switch nodes. Transformation matrices can also be applied to polygon and light point nodes, but the transformation matrices are only used to transform replications; without a replication the transformation matrices have no effect. If a group has a transformation matrix and a replication, the transformation matrix is used to transform the replications.

type: The type of transformation as a string, one of the following:
"TRANSFER"
"SCALE"
"ROTATE_ABOUT_EDGE"
"ROTATE_ABOUT_POINT"
"GENERAL_MATRIX"

fromX, fromY, fromZ: The reference ‘from’ position of a translation. Not used in the calculation.
deltaX, deltaY, deltaZ: The translation distance.
centerX, centerY, centerZ: The center of a scale or a rotate-about-point transformation.
scaleX, scaleY, scaleZ: The scale factors in X, Y and Z.
point1X, point1Y, point1Z: The ‘from’ point of the edge about which to rotate.
point2X, point2Y, point2Z: The ‘to’ point of the edge about which to rotate.
angle: The angle in degrees with which the rotation is performed.
axisX, axisY, axisZ: The axis around which the rotation is to be performed.
mXY: The value at column X and row Y of the general transformation matrix.

index (optional, default last): The zero-based index at which the new transformation matrix is to be added.

returns: true if a transformation matrix was added.
remo.getNumTransformationMatrices()

Reports the number of transformation matrices applied to the selected node. This will only work with a single (one and only one) node selected.

returns: The number of transformation matrices for the selected node.

remo.getTransformationMatrixType(index)

Reports the type of transformation matrix at a given index for the selected node.

index: The zero-based index at which the requested transformation matrix is located.

returns: The type of transformation as a string, one of the following:
  "TRANSLATE"
  "SCALE"
  "ROTATE_ABOUT_EDGE"
  "ROTATE_ABOUT_POINT"
  "ROTATE_SCALE_TO_POINT"
  "PUT_TRANSFORM"
  "GENERAL_MATRIX"
(or an empty string if no transformation matrix is at the given index)

remo.eliminateTransformationMatrices()

Eliminates all transformation matrices recursively in the currently selected sub-tree and modifies all coordinates accordingly instead.
remo.getTransformationMatrix(index)

Reports the attributes of the transformation matrix at a given index for the selected node.

index: The zero-based index at which the requested transformation matrix is located.

returns: Multiple values:

Always the type of transformation as a string, one of the following:
- "TRANSLATE"
- "SCALE"
- "ROTATE_ABOUT_EDGE"
- "ROTATE_ABOUT_POINT"
- "ROTATE_SCALE_TO_POINT"
- "PUT_TRANSFORM"
- "GENERAL_MATRIX"
(or an empty string if no transformation matrix is at the given index)

Different values depending on type:

TRANSLATE:
fromX, fromY, fromZ: The reference 'from' position of the translation. Not used in the calculation.
deltaX, deltaY, deltaZ: The translation distance.

SCALE:
centerX, centerY, centerZ: The center of the scale transformation.
scaleX, scaleY, scaleZ: The scale factors in X, Y and Z.

ROTATE_ABOUT_EDGE:
point1X, point1Y, point1Z: The 'from' point of the edge about which to rotate.
point2X, point2Y, point2Z: The 'to' point of the edge about which to rotate.
angle: The angle in degrees with which the rotation is performed.

ROTATE_ABOUT_POINT:
centerX, centerY, centerZ: The center of the rotation.
axisX, axisY, axisZ: The axis around which the rotation is to be performed.
angle: The angle in degrees with which the rotation is performed.

ROTATE_SCALE_TO_POINT:
centerX, centerY, centerZ: The center of the rotate-scale-to-point transformation.
refX, refY, refZ: The reference point of the rotate-scale-to-point transformation.
toX, toY, toZ: The 'to' point of the rotate-scale-to-point transformation.
overallScaleFactor: The overall scale factor of the rotate-scale-to-point transformation.
axisScaleFactor: The scale factor in axis direction of the rotate-scale-to-point transformation.
angle: The angle in degrees with which the rotation is performed.

PUT_TRANSFORM:
fromOriginX, fromOriginY, fromOriginZ: The 'from' point origin coordinate.
fromAlignX, fromAlignY, fromAlignZ: The 'from' point align coordinate.
fromTrackX, fromTrackY, fromTrackZ: The 'from' point track coordinate.
toOriginX, toOriginY, toOriginZ: The 'to' point origin coordinate.
toAlignX, toAlignY, toAlignZ: The 'to' point align coordinate.
toTrackX, toTrackY, toTrackZ: The 'to' point track coordinate.

GENERAL_MATRIX:
m00, m01, m02, m03, m10, m11, m12, m13, m20, m21, m22, m23, m30, m31, m32, m33:
The general transformation matrix values.
remo.clearTransformationMatrices()

Clears all transformation matrices from the selected node.

returns: true if any transformation matrices were cleared.

remo.removeTransformationMatrix(index)

index: The zero-based index at which the requested transformation matrix is located.

returns: true if a transformation matrix was removed.

remo.moveTransformationMatrix(index, newIndex)

index: The zero-based index at which the requested transformation matrix is located.

newIndex: The new zero-based index for the transformation matrix.

remo.getAccumulatedTransformationMatrix()

Reports the accumulated transformation matrix for the selected node.

returns: The accumulated transformation matrix as separate values: 
m00, m01, m02, m03, m10, m11, m12, m13, m20, m21, m22, m23, m30, m31, m32, m33

Creation Script Functions

This section describes script functions for creation tools.

remo.create(selectionMode)

Creates a primitive of given type.

selectionMode: The selection mode as a string, one of the following: 
"OBJECT"
"GROUP"
"LOD"
"DOF"
"SWITCH"
"LIGHTPOINTSYSTEM"

returns: The name of the new node, empty string if no node was created.
remo.createPolygon([table | x1, y1, z1, x2, y2, z2, ... , xn, yn, zn])

Creates a polygon (give a table OR separate coordinates as parameters)

table: A table of coordinates, given as [x1, y1, z1, x2, y2, z2, ...]

x1, y1, z1: Coordinate for the first point in the polygon.

x2, y2, z2: Coordinate for the second point in the polygon.

xn, yn, zn: Coordinate for the last point in the polygon.

returns: The name of the new polygon, empty string if no polygon was created.

remo.createDisk(centerX, centerY, centerZ, borderX, borderY, borderZ, normalX, normalY, normalZ, numSegments)

Creates a disk polygon.

centerX, centerY, centerZ: The center coordinate for the disk.

borderX, borderY, borderZ: The border coordinate for the disk.

normalX, normalY, normalZ: The normal for the disk.

numSegments: The number of segments in the disk.

returns: The name of the created disk polygon, empty string if no disk was created.

remo.createSphere(centerX, centerY, centerZ, borderX, borderY, borderZ, numRows, numSegments)

Creates a sphere.

centerX, centerY, centerZ: The center coordinate.

borderX, borderY, borderZ: The border coordinate.

numRows: The number of rows in the sphere.

numSegments: The number of segments in the sphere.

returns: The name of the created sphere object node, empty string if no sphere was created.
**Remo 3D™ v2.8 User’s Guide**

---

**remo.createLightPoint(indexed, x1, y1, z1, x2, y2, z2, ... , xn, yn, zn)**

Creates a light point node with one or more light point vertices.

indexed: true if an indexed light point is wanted, false if ordinary (inline).

x1, y1, z1: The coordinate for the first light point.

x2, y2, z2: The coordinate for the second light point.

xn, yn, zn: The coordinate for the last light point.

returns: The name of the new light point node, empty string if no light point node was created.

---

**remo.createExternals(filename1, filename2, ...)**

Creates one or more external references.

filenameX: The filename for the external number X to be created.

returns: true if any external reference nodes were created.

---

**Surface Script Functions**

This section describes script functions for surface appearance tools.

---

**remo.applyColor()**

Applies the current color to the currently selected nodes.

returns: true if any color was applied.

---

**remo.readColor()**

Reads the color from the currently selected node(s). The retrieved color will be the current color.

returns: true if a color was retrieved.

---

**remo.applyTexture()**

Applies the current texture to the currently selected nodes.

returns: true if any texture was applied.

---

**remo.readTexture()**

Reads the texture from the currently selected node(s). The retrieved texture will be the current texture.

returns: true if a texture was retrieved.
remo.applyTextureMapping()
Applies the current texture mapping to the currently selected nodes.
returns: true if any texture mapping was applied.

remo.readTextureMapping()
Reads the texture mapping from the currently selected node(s). The retrieved texture mapping will be the current texture mapping.
returns: true if a texture mapping was retrieved.

remo.applyMaterial()
Applies the current material to the currently selected nodes.
returns: true if any material was applied.

remo.readMaterial()
Reads the material from the currently selected node(s). The retrieved material will be the current material.
returns: true if a material was retrieved.

remo.applyShader()
Applies the current shader to the currently selected nodes.
returns: true if any shader was applied.

remo.readShader()
Reads the shader from the currently selected node(s). The retrieved shader will be the current shader.
returns: true if a shader was retrieved.

remo.removeTexture()
Removes all texture and texture mapping indices from the currently selected nodes.
returns: true if any texture or texture mapping was removed.

remo.removeTextureMapping()
Removes all texture mapping indices from the currently selected nodes.
returns: true if any texture mapping was removed.
remo.removeMaterial()

Removes all material indices from the currently selected nodes.

returns: true if any material was removed.

remo.removeShader()

Removes all shader indices from the currently selected nodes.

returns: true if any shader was removed.

Texture Mapping Script Functions

This section describes script functions for texture mapping.

remo.mapTexture(type, layer, addToPalette, ...
(rest of parameters depending on type as follows:)
THREEPOINT:
..., x1,y1,z1, x2,y2,z2, x3,y3,z3, u1,v1, u2,v2, u3,v3)
FOURPOINT:
..., x1,y1,z1, x2,y2,z2, x3,y3,z3, x4,y4,z4, u1,v1, u2,v2, u3,v3, u4,v4)
CYLINDRICAL, no palette:
..., x1,y1,z1, x2,y2,z2, x3,y3,z3, horizRepetitions)
CYLINDRICAL, with palette:
..., x1,y1,z1, x2,y2,z2, horizRepetitions)
SPHERICAL, no palette:
..., x1,y1,z1, x2,y2,z2, x3,y3,z3, horizRepetitions, vertRepetitions)
SPHERICAL, with palette:
..., x1,y1,z1, repetitions)

Maps the current texture on to the currently selected nodes.

type: The texture mapping type as a string, one of the following:
"THREEPOINT"
"FOURPOINT"
"CYLINDRICAL"
"SPHERICAL"

layer: The texture layer (0 - 7).

addToPalette: true if the created texture mapping should be added to the texture mapping palette.

x1,y1,z1: The first (origin) coordinate of a 3-point or 4-point mapping,
or the center of a non-palette cylindrical mapping (addToPalette = false),
or the lower center of a palette cylindrical mapping (addToPalette = true),
or the center of a spherical mapping.
x2, y2, z2: The second (alignment) coordinate of a 3-point or 4-point mapping, or the perpendicular surface coordinate of a non-palette cylindrical or spherical mapping (addToPalette = false), or the upper center of a palette cylindrical mapping (addToPalette = true).

x3, y3, z3: The third (shear) coordinate of a 3-point or 4-point mapping, or the vertical perpendicular coordinate of a non-palette cylindrical or spherical mapping (addToPalette = false).

x4, y4, z4: The fourth (perspective) coordinate of a 4-point mapping.

u1, v1: The texture coordinate corresponding to the first (origin) coordinate of a 3-point or 4-point mapping.

u2, v2: The texture coordinate corresponding to the second (alignment) coordinate of a 3-point or 4-point mapping.

u3, v3: The texture coordinate corresponding to the third (shear) coordinate of a 3-point or 4-point mapping.

u4, v4: The texture coordinate corresponding to the fourth (perspective) coordinate of a 4-point mapping.

horizRepetitions: The number of horizontal repetitions for a cylindrical or non-palette spherical mapping (addToPalette = false).

vertRepetitions: The number of vertical repetitions for a non-palette spherical mapping (addToPalette = false).

repetitions: The number of horizontal and vertical repetitions for a palette spherical mapping (addToPalette = true).

returns: true if any texture was mapped.

remo.moveTexmap(x, y, z, layer, freezeX, freezeY, freezeZ)

Moves the texture coordinates of the currently selected geometry.

x, y, z: The coordinates for the movement.

layer (optional, default 0): The texture layer to be moved.

freezeX, freezeY, freezeZ (optional, default false): Set to true to freeze movement along certain axes.

remo.rotateTexmap(angle, axisX, axisY, axisZ, layer)

Rotates the texture coordinates of the currently selected geometry.

angle: The angle to rotate in degrees.

axisX, axisY, axisZ: The axis to rotate around.

layer (optional, default 0): The texture layer to be rotated.
**Remo 3D™ v2.8 User’s Guide**

---

**remo.scaleTexmap(scaleX, scaleY, scaleZ, layer)**

Scales the texture coordinates of the currently selected geometry.

scaleX: The scale along the x-axis.

scaleY: The scale along the y-axis.

scaleZ: The scale along the z-axis.

layer (optional, default 0): The texture layer to be moved.

---

**remo.modifyUV(layer, vertexName1, u1, v1, vertexName2, u2, v2, ...)**

Modifies the texture coordinate of one or more given vertices.

layer: The texture layer (0 - 7).

vertexNameX: The name of a vertex, e.g. p19/V2.

uX, vX: The new texture coordinate for the given vertex.

*NOTE: Since the modifyUV function doesn't use the current selection, this is not the most efficient way of modifying a texture mapping from script, merely a necessary equivalent to be able to record actions in the Modify UV tool. Consider using the texture mapping transformation functions described above or the applyAttributes function instead.*

---

**Modification Script Functions**

This section describes script functions for modification tools.

---

**remo.mirror()**

Mirrors the selected nodes with respect to the grid plane.

returns: true if any geometry was mirrored.

---

**remo.reverse()**

Reverses the polygons in the current selection.

returns: true if any polygons were reversed.
remo.slice(sliceAlongGridPlane)

Slices the currently selected geometry either along the grid plane or along the grid lines.

sliceAlongGridPlane: Set to true to slice along the grid plane. Set to false to slice along the grid lines, like a cookie-cutter.

returns: true if any geometry was sliced.

remo.split()

Splits selected edges in halves, triangulates selected polygons and splits polygons between selected vertices.

returns: true if any geometry was split.

remo.extrude(keepFloor, ...
(rest of parameters depending on if user wants a point or a height:)
POINT:
..., x, y, z, sharedAxis)
HEIGHT:
..., height)

Extrudes (elevates and builds walls) the currently selected polygons to a POINT or a along the polygon normals to a specific HEIGHT.

keepFloor: true if the floor should be kept.

x, y, z: The point to which the extrusion is to be performed.

sharedAxis: Use the extrusion axis from the pivot point of the selected polygons to the indicated point.

height: The height with which the extrusion will be performed.

returns: true if any geometry was extruded.

remo.revolve(x, y, z, angle, numSteps, revolveNormals)

Revolves the currently selected polygons around a specific axis. Note that the revolve origin is the current pivot point.

x, y, z: The axis to revolve around.

angle: The angle to revolve.

numSteps: The number of steps to divide the revolve into.

revolveNormals: true if the normals should be revolved.

returns: true if any geometry was revolved.
remo.subdivide(iterations, subdivideBorders)

Increases the complexity of the selected geometry by subdividing the current polygons.

iterations: The number of subdividing iterations.

subdivideBorders (optional, default false): true if borders should be subdivided.

returns: true if any geometry was subdivided.

remo.simplify(level, weightRatio, boundaryLevel, boundaryWeightRatio, vertexWeightEqual)

Reduces the complexity of the selected geometry.

level: The simplify level (0.0 - 1.0), where 1.0 means maximum reduction and 0.0 means no reduction.

weightRatio: The ratio (0.0 - 1.0) with which to blend between the edge length and the normal deviation. 0.0 means using only edge length and 1.0 means using only normal deviation.

boundaryLevel: The simplify level for the boundary edges (0.0 - 1.0), where 1.0 means maximum reduction and 0.0 means no reduction.

boundaryWeightRatio: The ratio (0.0 - 1.0) with which to blend between the edge length and the edge angles. 0.0 means using only edge length and 1.0 means using only edge angles.

vertexWeightEqual: true if the vertex weight is to be equal, otherwise the weight will be based on the normal deviation.

returns: true if any geometry was simplified.

remo.merge()

Merges all selected polygons that are adjacent and in the same plane.

returns: true if any polygons were merged.

remo.quantify(quantification)

Quantifies all selected vertices to a specific value.

quantification: The value to quantify all values with.

returns: true if any vertices were quantified.
remo.shade(flat, updateLightMode, vertexColorLit, creaseAngle)

Smooth or flat shades polygons in the selected geometry.

flat: true if the polygons should be flat shaded.

updateLightMode: true if the polygon light mode should be updated.

vertexColorLit (optional, default false): true if the polygon light mode should be set to 'vertex color lit', false if 'polygon color lit'. Both require updateLightMode to be true.

creaseAngle (optional, default 180): The crease angle to use when smooth shading.

returns: true if any polygons were shaded.

---

**Attribute Script Functions**

This section describes script functions for attribute tools.

```plaintext
remo.applyAttributes(name1, attribute1, name2, attribute2, ...)

(remo.setAttributes is an alias that works exactly in the same way)

Applies a number of attributes to the selected geometry.

nameX: The name of the attribute number X as a string, as it appears in the corresponding label in the attribute tool (case-insensitive). See below for more information. (*)

attributeX: The value of the attribute number X that is to be applied, can be of different types (string, boolean or number) depending on the attribute.

returns: true if any attribute was applied.

remo.getAttributes(name1, name2, ...)

Reports the values for a number of given attributes of the selected geometry.

nameX: The name of the attribute number X as a string, as it appears in the corresponding label in the attribute tool (case-insensitive). See below for more information. (*)

returns: A number of values corresponding to the given attribute names, in the same order. The values can be of different types (string, boolean or number) depending on the attribute. If a value is undefined or different among the selected primitives, it’s set to nil.

remo.addSwitchMask(maskString, maskIndex)

Adds a switch mask to the selected switch node.

maskString: A string containing 1 for an enabled child and 0 for a disabled child, e.g. "1101".

maskIndex (optional): If given, specifies the index of the new mask - otherwise added last.
Remo 3D™ v2.8 User’s Guide

**remo.deleteSwitchMask(maskIndex)**

Deletes a given switch mask from the selected switch node.

*maskIndex*: The zero-based mask index.

**remo.setSwitchMask(maskIndex, maskString)**

Modifies an existing switch mask on the selected switch node.

*maskIndex*: The zero-based index for the existing switch mask.

*maskString*: A string containing 1 for an enabled child and 0 for a disabled child, e.g. "1101".

**remo.setActiveSwitchMaskIndex(maskIndex)**

Sets the currently active switch mask on the selected switch node.

*maskIndex*: The zero-based index for the new currently active mask.

**remo.getNumSwitchMasks()**

Reports the number of switch masks for the selected switch node.

*returns*: The number of switch masks.

**remo.getSwitchMask(maskIndex)**

Reports a given switch mask for the selected switch node.

*maskIndex*: The zero-based mask index.

*returns*: The requested switch mask as a string containing 1 for an enabled child and 0 for a disabled child, e.g. "1101".

**remo.getSwitchState(maskIndex, childNumber)**

Reports the switch state for a given child in a given switch mask on the selected switch node.

*maskIndex*: The zero-based mask index.

*childNumber*: The zero-based child number.

*returns*: The switch state (true or false) for the given child in the given mask.

**remo.getCurrentSwitchMaskIndex()**

Reports the index for the currently active switch mask for the selected switch node.

*returns*: The zero-based index for the currently active mask.
**remo.applyComment(comment)**

(remo.setComment is an alias that works exactly in the same way)

Applies a comment to the currently selected nodes.

**comment**: The comment string to set on the selected nodes.

**returns**: true if a comment was applied.

---

**remo.getComment()**

Reports the comment of the currently selected node(s).

**returns**: The comment string of the currently selected node(s). Empty if not equal among the selected nodes.

---

(*) The labels given to apply/setAttributes and getAttributes should, as described above, be given as they appear in the corresponding label in the attribute tool. However, there are some special cases:

- Attributes placed in groups in the attribute tool will normally get the group name as prefix, e.g. "Primary Color Index". The exceptions from this rule are multi-texture attributes, LOD switch range attributes and the Freeze LOD center attribute.
- Attributes representing the check box in the top right corner of a group, used for enabling or disabling the complete group (e.g. Normal and Texture Coordinate for vertex attributes, Back Color for light point attributes) will get an "Enabled" suffix, e.g. "Normal Enabled".
- Additional information within parentheses or brackets at the end of a label will be excluded.
- To specify the multi-texture layer index of a multi-texture-specific attribute, a case-insensitive "(layer X)" suffix has to be added to the attribute label name, where X is the zero-based multi-texture layer index. An example is "Texture Index (layer 1)".
- Special labels are used for the switch attributes:
  - "Mask Count" - replaces all masks with the new number of fully enabled masks.
  - "Active Mask" - the currently active zero-based mask index.
  - "Child Enabled (mask M, child C)" - the state (true/false) for the given child C in the given mask M.
- It's recommended to use the special switch functions described above for more control of the switch attributes.

The same rules apply to the attribute list labels in the Select By Attributes window in Remo 3D, which makes that list a good reference when writing your scripts.
**Reporting Script Functions**

This section describes script functions for producing reports about the geometry.

---

```javascript
remo.reportStatistics(name1, name2, ...)  
```

Reports a number of statistic values.

*nameX*: The name of the statistic number X as a string, as it appears in the corresponding label in the statistics tool (case-insensitive).

*returns*: A number of values corresponding to the given statistic names, in the same order. The values are of number type. If no result is applicable, e.g. because of an empty selection, zeroes are returned.

---

```javascript
remo.reportDifferences(source, ignoreRevisionDate, ignoreDatabaseVersion, filename)  
```

Reports the differences between the currently selected sub-tree and either the sub-tree contained in the current clipboard (put there by a Copy or Cut operation) or with another OpenFlight file.

*source*: The type of source, one of the following:
- "CLIPBOARD"
- "FILE"

*ignoreRevisionDate* (optional, default true): true if the revision date of the file to compare with should be ignored.

*ignoreDatabaseVersion* (optional, default true): true if the database version (not the OpenFlight version) of the file to compare with should be ignored.

*filename*: The name of the file to compare with (if source is "FILE", otherwise ignored)

*returns*: The first difference encountered, as a descriptive string. The difference can be e.g. different node types, node names, attributes or different numbers of children.

---

**Grid Script Functions**

This section describes script functions handling the current grid.

```javascript
remo.setXYGrid()  
```

Sets the grid to be aligned to the XY plane with the origin at (0,0,0).

```javascript
remo.setXZGrid()  
```

Sets the grid to be aligned to the XZ plane with the origin at (0,0,0).
remo.setYZGrid()

Sets the grid to be aligned to the YZ plane with the origin at (0,0,0).

remo.setGridSpacing(spacing)

Changes the spacing between the grid lines.

spacing: The spacing between the grid lines.

remo.addGridOffset(offset)

Offsets the grid in the direction of the grid normal.

offset: The offset distance.

remo.setGridOrigin(originX, originY, originZ)

Changes the grid origin, the grid will still be aligned as before but translated so that the grid origin is at the new origin position.

originX, originY, originZ: The new origin position.

remo.setGridPointOnAxis(pointOnAxisX, pointOnAxisY, pointOnAxisZ)

Changes the grid point on axis, the grid will be rotated around the axis from the origin to the third point so that the point on axis lies on one of the grid axes.


remo.setGridThirdPoint(thirdPointX, thirdPointY, thirdPointZ)

Changes the grid third point, the grid will be rotated around the axis from the origin to the point on axis so that the third point lies on the grid plane.

thirdPointX, thirdPointY, thirdPointZ: The new third point position.

remo.getGridOrigin()

Reports the grid origin.

returns: The grid origin as three values (x,y,z).

remo.getGridPointOnAxis()

Reports the grid point on axis.

returns: The grid point on axis as three values (x,y,z).

remo.getGridThirdPoint()

Reports the grid third point.

returns: The grid third point as three values (x,y,z).
remo.alignGridToPolygon()

Aligns the grid to the currently selected polygon, if any.

returns: true if alignment was successful.

---

**Color Palette Script Functions**

This section describes script functions related to the color palette.

---

**remo.setCurrentColorIndex(index)**

Sets the current color index (excluding the intensity) for the current model.  
(if indexed colors aren't used for this model, this will do nothing)

index: The new zero-based index (0 - 1023 or -1 if none).

---

**remo.setCurrentColorIntensity(intensity)**

Sets the current color intensity for the current model.  
(if indexed colors aren't used for this model, this will do nothing)

intensity: The new intensity (0 - 127).

---

**remo.getCurrentColorIndex()**

Reports the current color index (excluding the intensity) for the current model.

returns: The current zero-based color index (0 – 1023 or -1 if none).

---

**remo.getCurrentColorIntensity()**

Reports the current color intensity for the current model.

returns: The current color intensity (0 - 127).

---

**remo.setCurrentColorRGB(red, green, blue)**

Sets the current RGB color (if RGB colors are used for this model) for the current model.  
(if RGB colors aren't used for this model, this will do nothing)

red: The red value (0-255) for the new current RGB color.

green: The red value (0-255) for the new current RGB color.

blue: The red value (0-255) for the new current RGB color.
**Remo 3D™ v2.8 User’s Guide**

---

**remo.getCurrentColorRGB()**

Reports the current RGB color (if RGB colors are used for this model) for the current model. If RGB colors aren't used for this model, this will report a completely white (255,255,255) color.

returns: Three integers (0-255) corresponding to the red, green and blue values of the current RGB color.

---

**remo.openColorPalette(filename)**

Opens a palette of colors from disk.

filename: The filename of the color palette on disk.

returns: true if the palette was opened correctly.

---

**remo.saveColorPalette(filename)**

Saves the current color palette to a file on disk.

filename: The filename of the saved palette.

returns: true if the palette was saved correctly.

---

**remo.setPaletteColor(index, red, green, blue)**

Sets the RGB values of a color in the color palette.

index: The zero-based index of the color to be modified (0 - 1023).

red: The red value (0 - 255) for the new palette RGB color.

green: The green value (0 - 255) for the new palette RGB color.

blue: The blue value (0 - 255) for the new palette RGB color.

returns: true if a color was changed.

---

**remo.getPaletteColor(index)**

Reports the RGB values of a color in the color palette.

index: The zero-based index of the color to be reported (0 - 1023).

returns: Three integers (0-255) corresponding to the red, green and blue values of the palette color.
**Texture Palette Script Functions**

This section describes script functions related to the texture palette.

---

**remo.loadTexture(filename)**

Loads a texture into the texture palette.

*filename*: The texture filename.

*returns*: The index of the new texture, -1 if loading failed.

---

**remo.loadTextures(filename1, filename2, ...)**

Loads a number of textures into the texture palette.

*filenameX*: The texture filename number X.

---

**remo.replaceTexture(index, filename)**

Replaces the texture at a specified index with a new texture with a given filename.

*index*: The zero-based palette index of the texture to be replaced.

*filename*: The filename of the new texture.

*returns*: true if replacing was successful, i.e. if the index was valid and a new texture was loaded.

---

**remo.getMaxTextureIndex()**

Reports the maximum texture index, useful e.g. when looping through the texture palette.

*returns*: The maximum texture index.

---

**remo.getTextureIndex(filename)**

Reports the palette index of a texture with a specified filename.

*filename*: The texture filename (exactly as it appears in the OpenFlight file).

*returns*: The index of the texture, -1 if not found.

---

**remo.getTextureFilename(index)**

Reports the filename of a texture with a specified palette index.

*index*: The zero-based palette index of the texture.

*returns*: The filename of the texture, empty string if not found.
remo.getTextureSize(index)

Reports the width and height in pixels of a texture with a specified palette index.

index: The zero-based palette index of the texture.

returns: Two integers corresponding to the width and height of the texture; (-1, -1) if not found.

remo.setCurrentTextureIndex(index)

Sets the current texture index for the current model.

index: The new zero-based palette index (-1 if none).

remo.getCurrentTextureIndex()

Reports the current texture index for the current model.

returns: The current texture index (-1 if none).

remo.changeTextureIndex(oldIndex, newIndex)

Changes the palette index of a given texture.

oldIndex: The old zero-based palette index of the texture.

newIndex: The new zero-based palette index of the texture.

returns: true if the index change was successful.

remo.openTexturePalette(filename)

Opens a palette of textures from disk.

filename: The filename of the texture palette on disk.

returns: true if the palette was opened correctly.

remo.saveTexturePalette(filename)

Saves the current texture palette to a file on disk.

filename: The filename of the saved palette.

returns: true if the palette was saved correctly.

remo.removeTextureFromPalette(index)

Removes the texture at the specified index from the palette.

index: The zero-based palette index of the texture that is to be removed.

returns: true if a texture was removed.
remo.removeUnusedTexturesFromPalette()

Removes all unused textures from the palette.

returns: true if any texture was removed.

remo.removeTexturePath(index)

Removes the path for the texture with a specified index.

index: The zero-based palette index of the texture.

returns: true if the path removal was successful.

remo.removeAllTexturePaths()

Removes the paths for all textures in the palette.

remo.changeAllTexturePaths(path)

Changes the path for all textures in the palette.

path: The new path (except the filename).

remo.makeTexturePathsRelative()

Update all texture paths in the texture palette to be relative to the current model file path.

remo.makeTexturePathsAbsolute()

Update all texture paths in the texture palette to be absolute, by appending the model file path to the relative paths.

remo.applyTextureAttributes(index, name1, attribute1, name2, attribute2, ...)

(remo.setTextureAttributes is an alias that works exactly in the same way)

Applies a number of texture attributes to the texture with the selected index and saves the corresponding attribute file.

index: The zero-based palette index of the texture for which attributes are to be applied.

nameX: The name of the texture attribute number X as a string, as it appears in the corresponding label in the texture attribute dialog (case-insensitive). More information about the rules of attribute label names can be found at the asterisk (*) in the Attribute Script Functions section.

attributeX: The value of the texture attribute number X that is to be applied, can be of different types (string, boolean or number) depending on the texture attribute.

returns: true if any texture attribute was applied.
remo.getTextureAttributes(index, name1, name2, ...)

Reports the values of a number of given texture attributes for the texture with a specified index.

index: The zero-based palette index of the texture from which attributes are to be retrieved.

nameX: The name of the texture attribute number X as a string, as it appears in the corresponding label in the texture attribute dialog (case-insensitive). More information about the rules of attribute label names can be found at the asterisk (*) in the Attribute Script Functions section.

returns: A number of values corresponding to the given texture attribute names, in the same order. The values can be of different types (string, boolean or number) depending on the texture attribute. If a value is undefined, it’s set to nil.

**Material Palette Script Functions**

This section describes script functions related to the material palette.

remo.newMaterial()

Creates a new material in the palette.

returns: The palette index of the new material.

remo.getMaxMaterialIndex()

Reports the maximum material index, useful e.g. when looping through the material palette.

returns: The maximum material index.

remo.setCurrentMaterialIndex(index)

Sets the current material index for the current model.

index: The new zero-based index (-1 if none).

remo.getCurrentMaterialIndex()

Reports the current material index for the current model.

returns: The current material index (-1 if none).
remo.changeMaterialIndex(oldIndex, newIndex)
Changes the palette index of a given material.
oldIndex: The old zero-based index of the material.
newIndex: The new zero-based index of the material.
returns: true if the index change was successful.

remo.openMaterialPalette(filename)
Opens a palette of materials from disk.
filename: The filename of the material palette on disk.
returns: true if the palette was opened correctly.

remo.saveMaterialPalette(filename)
Saves the current material palette to a file on disk.
filename: The filename of the saved palette.
returns: true if the palette was saved correctly.

remo.removeMaterialFromPalette(index)
Removes the material at the specified index from the palette.
index: The zero-based index of the material that is to be removed.
returns: true if a material was removed.

remo.removeUnusedMaterialsFromPalette()
Removes all unused materials from the palette.
returns: true if any material was removed.

remo.setMaterialAmbient(index, red, green, blue)
Sets the ambient color of the material with the given index.
index: The zero-based index of the material.
red: The red component (0 - 255).
green: The green component (0 - 255).
blue: The blue component (0 - 255).
**remo.getMaterialAmbient(index)**

Reports the ambient color of the material with the given index.

index: The zero-based index of the material.

returns: Three integers (0 - 255) corresponding to the red, green and blue values of the requested color.

** remo.setMaterialDiffuse(index, red, green, blue)**

Sets the diffuse color of the material with the given index.

index: The zero-based index of the material.

red: The red component (0 - 255).

green: The green component (0 - 255).

blue: The blue component (0 - 255).

** remo.getMaterialDiffuse(index)**

Reports the diffuse color of the material with the given index.

index: The zero-based index of the material.

returns: Three integers (0 - 255) corresponding to the red, green and blue values of the requested color.

** remo.setMaterialSpecular(index, red, green, blue)**

Sets the specular color of the material with the given index.

index: The zero-based index of the material.

red: The red component (0 - 255).

green: The green component (0 - 255).

blue: The blue component (0 - 255).

** remo.getMaterialSpecular(index)**

Reports the specular color of the material with the given index.

index: The zero-based index of the material.

returns: Three integers (0 - 255) corresponding to the red, green and blue values of the requested color.
remo.setMaterialEmissive(index, red, green, blue)
Sets the emissive color of the material with the given index.
index: The zero-based index of the material.
red: The red component (0 - 255).
green: The green component (0 - 255).
blue: The blue component (0 - 255).

remo.getMaterialEmissive(index)
Reports the emissive color of the material with the given index.
index: The zero-based index of the material.
returns: Three integers (0 - 255) corresponding to the red, green and blue values of the requested color.

remo.setMaterialShininess(index, shininess)
Sets the shininess of the material with the given index.
index: The zero-based index of the material.
shininess: The shininess as a number (0.0 - 128.0).

remo.getMaterialShininess(index)
Reports the shininess of the material with the given index.
index: The zero-based index of the material.
returns: The shininess as a number (0.0 - 128.0).

remo.setMaterialAlpha(index, alpha)
Sets the alpha of the material with the given index.
index: The zero-based index of the material.
alpha: The alpha as a number (0.0 - 1.0).

remo.getMaterialAlpha(index)
Reports the alpha of the material with the given index.
index: The zero-based index of the material.
returns: The alpha as a number (0.0 – 1.0).
Texture Mapping Palette Script Functions

This section describes script functions related to the texture mapping palette.

```javascript
remo.getMaxTextureMappingIndex()
```

Reports the maximum texture mapping index, useful e.g. when looping through the texture mapping palette.

returns: The maximum texture mapping index.

```javascript
remo.setCurrentTextureMappingIndex(index)
```

Sets the current texture mapping index for the current model.

index: The new zero-based index (-1 if none).

```javascript
remo.getCurrentTextureMappingIndex()
```

Reports the current texture mapping index for the current model.

returns: The current texture mapping index (-1 if none).

```javascript
remo.changeTextureMappingIndex(oldIndex, newIndex)
```

Changes the palette index of a given texture mapping.

oldIndex: The old zero-based index of the texture mapping.

newIndex: The new zero-based index of the texture mapping.

returns: true if the index change was successful.

```javascript
remo.openTextureMappingPalette(filename)
```

Opens a palette of texture mappings from disk.

filename: The filename of the texture mapping palette on disk.

returns: true if the palette was opened correctly.

```javascript
remo.saveTextureMappingPalette(filename)
```

Saves the current texture mapping palette to a file on disk.

filename: The filename of the saved palette.

returns: true if the palette was saved correctly.
remo.removeTextureMappingFromPalette(index)

Removes the texture mapping at the specified index from the palette.

index: The zero-based index of the texture mapping that is to be removed.

returns: true if a texture mapping was removed.

remo.removeUnusedTextureMappingsFromPalette()

Removes all unused texture mappings from the palette.

returns: true if any texture mapping was removed.

remo.setTextureMappingName(index, name)

Sets the name of the texture mapping at the specified index.

index: The zero-based palette index of the texture mapping to be renamed.

name: The new name of the texture mapping.

remo.getTextureMappingName(index)

Reports the name of the texture mapping with the specified index.

index: The zero-based palette index of the texture mapping.

returns: The texture mapping name as a string.

remo.setTextureMappingUV(index, number, u,v)

Sets the texture coordinate of a given point in the texture mapping at the specified index.

index: The zero-based palette index of the texture mapping.

number: The point number (1-4).

u,v: The texture coordinate corresponding to the numbered point.

remo.getTextureMappingUV(index, number)

Reports the texture coordinate corresponding to a given point in the texture mapping with the given index.

index: The zero-based palette index of the texture mapping.

number: The point number (1-4).

returns: The requested texture coordinate as two numbers (u, v).
**remo.setTextureMappingRepetitions(index, repetitions)**

Sets the number of repetitions of the texture mapping at the specified index; only valid for cylindrical and spherical mappings.

- **index**: The zero-based palette index of the texture mapping for which the number of repetitions is to be set.
- **repetitions**: The new number of repetitions.

**remo.getTextureMappingRepetitions(index)**

Reports the number of repetitions for the texture mapping with the specified index.

- **returns**: The number of repetitions as a number (0.0 if inapplicable).

**remo.getTextureMappingType(index)**

Reports the type of the texture mapping at the specified index.

- **index**: The zero-based palette index of the texture mapping.
- **returns**: The texture mapping type as a string (empty if invalid), one of the following:
  - "THREEPOINT"
  - "FOURPOINT"
  - "CYLINDRICAL"
  - "SPHERICAL"

---

**Light Point Palette Script Functions**

This section describes script functions related to the light point palette.

**remo.setCurrentLightPointAppearanceIndex(index)**

Sets the current light point appearance index for the current model.

- **index**: The new zero-based index (-1 if none).

**remo.setCurrentLightPointAnimationIndex(index)**

Sets the current light point animation index for the current model.

- **index**: The new zero-based index (-1 if none).

**remo.getCurrentLightPointAppearanceIndex()**

Reports the current light point appearance index for the current model.

- **returns**: The current light point appearance index (zero-based, -1 if none).
remo.getCurrentLightPointAnimationIndex()
Reports the current light point animation index for the current model.
returns: The current light point animation index (zero-based, -1 if none).

remo.changeLightPointAppearanceIndex(oldIndex, newIndex)
Changes the palette index of a given light point appearance.
oldIndex: The old zero-based index of the light point appearance.
newIndex: The new zero-based index of the light point appearance.
returns: true if the index change was successful.

remo.changeLightPointAnimationIndex(oldIndex, newIndex)
Changes the palette index of a given light point animation.
oldIndex: The old zero-based index of the light point animation.
newIndex: The new zero-based index of the light point animation.
returns: true if the index change was successful.

remo.openLightPointPalette(filename)
Opens a palette of light points from disk.
filename: The filename of the light point palette on disk.
returns: true if the palette was opened correctly.

remo.saveLightPointPalette(filename)
Saves the current light point palette to a file on disk.
filename: The filename of the saved palette.
returns: true if the palette was saved correctly.

remo.newLightPointAppearance()
Creates a new light point appearance in the palette.
returns: The palette index of the new light point appearance.

remo.newLightPointAnimation()
Creates a new light point animation in the palette.
returns: The palette index of the new light point animation.
**remo.getMaxLightPointAppearanceIndex()**

Reports the maximum light point appearance index, useful e.g. when looping through the light point palette.

returns: The maximum light point appearance index.

**remo.getMaxLightPointAnimationIndex()**

Reports the maximum light point animation index, useful e.g. when looping through the light point palette.

returns: The maximum light point animation index.

**remo.removeLightPointAppearanceFromPalette(index)**

Removes the light point appearance at the specified index from the palette.

index: The zero-based index of the light point appearance that is to be removed.

returns: true if a light point appearance was removed.

**remo.removeLightPointAnimationFromPalette(index)**

Removes the light point animation at the specified index from the palette.

index: The zero-based index of the light point animation that is to be removed.

returns: true if a light point animation was removed.

**remo.removeUnusedLightPointAppearancesFromPalette()**

Removes all unused light point appearances from the palette.

returns: true if any light point appearance was removed.

**remo.removeUnusedLightPointAnimationFromPalette()**

Removes all unused light point animations from the palette.

returns: true if any light point animation was removed.
remo.applyLightPointAppearanceAttributes(index, name1, attribute1, name2, attribute2, ...)
(remo.setLightPointAppearanceAttributes is an alias that works exactly in the same way)

Applies a number of attributes to the light point appearance with the selected index.

index: The zero-based palette index of the light point appearance for which attributes are to be applied.

nameX: The name of the attribute number X as a string, as it appears in the corresponding label in the light point palette dialog (case-insensitive). See below for more information. (*)

attributeX: The value of the attribute number X that is to be applied, can be of different types (string, boolean or number) depending on the attribute.

returns: true if any attribute was applied.

remo.applyLightPointAnimationAttributes(index, name1, attribute1, name2, attribute2, ...)
(remo.setLightPointAnimationAttributes is an alias that works exactly in the same way)

Applies a number of attributes to the light point animation with the selected index.

index: The zero-based palette index of the light point animation for which attributes are to be applied.

nameX: The name of the attribute number X as a string, as it appears in the corresponding label in the light point palette dialog (case-insensitive). See below for more information. (*)

attributeX: The value of the attribute number X that is to be applied, can be of different types (string, boolean or number) depending on the attribute.

returns: true if any attribute was applied.

remo.getLightPointAppearanceAttributes(index, name1, name2, ...)

Reports the values of a number of given attributes for the light point appearance with a specified index.

index: The zero-based palette index of the light point appearance from which attributes are to be retrieved.

nameX: The name of the attribute number X as a string, as it appears in the corresponding label in the light point palette attribute dialog (case-insensitive). See below for more information. (*)

returns: A number of values corresponding to the given attribute names, in the same order. The values can be of different types (string, boolean or number) depending on the attribute. If a value is undefined, it’s set to nil.
Remo 3D™ v2.8 User’s Guide

remo.getLightPointAnimationAttributes(index, name1, name2, ...)

Reports the values of a number of given attributes for the light point animation with a specified index.

index: The zero-based palette index of the light point animation from which attributes are to be retrieved.

nameX: The name of the attribute number X as a string, as it appears in the corresponding label in the light point palette attribute dialog (case-insensitive). See below for more information. (*)

returns: A number of values corresponding to the given attribute names, in the same order. The values can be of different types (string, boolean or number) depending on the attribute. If a value is undefined, it’s set to nil.

remo.addLightPointSequenceState(index)

Adds a new sequence state to the end of the sequence states list in the light point animation with the given index.

index: The zero-based index of the light point animation.

returns: The zero-based sequence index of the added sequence state, -1 if nothing was added.

remo.removeLightPointSequenceState(index, sequenceIndex)

Removes a sequence state with a given index from the light point animation with the given index.

sequenceIndex: The zero-based sequence index of the light point animation.

index: The zero-based index of the light point animation.

returns: true if a sequence state was removed.

remo.getNumLightPointSequenceStates(index)

Reports the number of sequence states in the light point animation with the given index.

index: The zero-based index of the light point animation.

returns: The number of sequence states, 0 if none or inapplicable.

(*)The labels given to apply/setLightPointAppearance/AnimationAttributes and getLightPointAppearance/AnimationAttributes should, as described above, be given as they appear in the corresponding label in the light point palette dialog. However, there are some special cases:

- Attributes placed in groups in the light point palette dialog will get the group name as prefix, e.g. "Back Color Index".
- Attributes representing the check box in the top right corner of a group, used for enabling or disabling the complete group (e.g. Back Color) will get an "Enabled" suffix, e.g. "Back Color Enabled".
Additional information within parentheses or brackets at the end of a label will be excluded.

To specify the sequence state index of a sequence-specific attribute for a light point animation (i.e. State, Duration, Color Change Index or Color Change Intensity), a case-insensitive ",(sequence X)" suffix has to be added to the attribute label name, where X is the zero-based sequence index. An example is "State (sequence 2)".

**Shader Palette Script Functions**

This section describes script functions related to the shader palette.

---

`remo.setCurrentShaderIndex(index)`

Sets the current shader index for the current model.

**index**: The new zero-based index (-1 if none).

---

`remo.getCurrentShaderIndex()`

Reports the current shader index for the current model.

**returns**: The current shader index (-1 if none).

---

`remo.changeShaderIndex(oldIndex, newIndex)`

Changes the palette index of a given shader.

**oldIndex**: The old zero-based index of the shader.

**newIndex**: The new zero-based index of the shader.

**returns**: true if the index change was successful.

---

`remo.setShaderName(index, name)`

Sets the name of a given shader.

**index**: The zero-based index of the shader.

**name**: The new name of the shader as a string.

**returns**: true if the name change was successful.

---

`remo.getShaderName(index)`

Reports the name of a given shader.

**index**: The zero-based index of the shader.

**returns**: The name of the shader as a string.
remo.openShaderPalette(filename)
Opens a palette of shaders from disk.
filename: The filename of the shader palette on disk.
returns: true if the palette was opened correctly.
remo.saveShaderPalette(filename)
Saves the current shader palette to a file on disk.
filename: The filename of the saved palette.
returns: true if the palette was saved correctly.
remo.newShader(name)
Creates a new shader in the palette.
name: The name of the new shader.
returns: The palette index of the new shader.
remo.getMaxShaderIndex()
Reports the maximum shader index, useful e.g. when looping through the shader palette.
returns: The maximum shader index.
remo.removeShaderFromPalette(index)
Removes the shader at the specified index from the palette.
index: The zero-based palette index of the shader that is to be removed.
returns: true if a shader was removed.
remo.removeUnusedShadersFromPalette()
Removes all unused shaders from the palette.
returns: true if any shader was removed.
remo.setShaderType(index, type)
Sets the type of a given shader.
index: The zero-based palette index of the shader to set the shader type for.
type: A case-insensitive string with the type of the shader, can be either "GLSL" or "Cg".
remo.getShaderType(index)

Reports the type of a given shader.

index: The zero-based palette index of the shader to get the shader type for.

returns: The shader type as a string, either "GLSL" or "Cg".

remo.addOpenGLShaderVertexProgram(index, filename)

Adds an OpenGL shader vertex program to a shader with a specified index.

index: The zero-based palette index of the shader to add the vertex program to.

filename: The filename of the vertex program.

remo.getOpenGLShaderVertexPrograms(index)

Reports the OpenGL shader vertex programs of a shader with a specified index.

index: The zero-based palette index of the shader to get the vertex programs from.

returns: A table containing filenames of the vertex programs.

remo.removeOpenGLShaderVertexProgram(index, vertexProgramIndex)

Removes an OpenGL shader vertex program from a shader with a specified index.

index: The zero-based palette index of the shader to remove the vertex program from.

vertexProgramIndex: The zero-based index of the vertex program to remove.

remo.addOpenGLShaderFragmentProgram(index, filename)

Adds an OpenGL shader fragment program to a shader with a specified index.

index: The zero-based palette index of the shader to add the fragment program to.

filename: The filename of the fragment program.

remo.getOpenGLShaderFragmentPrograms(index)

Reports the OpenGL shader fragment programs of a shader with a specified index.

index: The zero-based palette index of the shader to get the fragment programs from.

returns: A table containing filenames of the fragment programs.
remo.removeOpenGLShaderFragmentProgram(index, fragmentProgramIndex)

Removes an OpenGL shader fragment program from the shader with index.

index: The zero-based palette index of the shader to remove the fragment program from.

fragmentProgramIndex: The zero-based index of the fragment program to remove.

remo.setCgShaderVertexProgram(index, filename)

Sets the Cg shader vertex program for a shader with a specified index.

index: The zero-based palette index of the shader to set the vertex program for.

filename: The filename of the vertex program.

remo.getCgShaderVertexProgram(index)

Reports the Cg shader vertex program for a shader with a specified index.

index: The zero-based palette index of the shader to get the vertex program from.

returns: The filename of the vertex program.

remo.setCgShaderFragmentProgram(index, filename)

Sets the Cg shader fragment program for a shader with a specified index.

index: The zero-based palette index of the shader to set the fragment program for.

filename: The filename of the fragment program.

remo.getCgShaderFragmentProgram(index)

Reports the Cg shader fragment program for a shader with a specified index.

index: The zero-based palette index of the shader to get the fragment program from.

returns: The filename of the fragment program.

remo.setCgShaderVertexProgramProfile(index, profile)

Sets the Cg shader vertex program profile for a shader with a specified index.

index: The zero-based palette index of the shader to set the vertex program profile for.

profile: The profile of the vertex program as an integer.
remo.getCgShaderVertexProgramProfile(index)

Reports the Cg shader vertex program profile for a shader with a specified index.

index: The zero-based palette index of the shader to get the vertex program profile from.

returns: The profile of the vertex program as an integer.

remo.setCgShaderFragmentProgramProfile(index, profile)

Sets the Cg shader fragment program profile for a shader with a specified index.

index: The zero-based palette index of the shader to set the fragment program profile for.

profile: The profile of the fragment program as an integer.

remo.getCgShaderFragmentProgramProfile(index)

Reports the Cg shader fragment program profile for a shader with a specified index.

index: The zero-based palette index of the shader to get the fragment program profile from.

returns: The profile of the fragment program as an integer.

remo.setCgShaderVertexProgramEntryPoint(index, entryPoint)

Sets the Cg shader vertex program entry point for a shader with a specified index.

index: The zero-based palette index of the shader to set the vertex program entry point for.

entryPoint: The entry point of the vertex program as a string.

remo.getCgShaderVertexProgramEntryPoint(index, entryPoint)

Reports the Cg shader vertex program entry point for a shader with a specified index.

index: The zero-based palette index of the shader to get the vertex program entry point from.

returns: The entry point of the vertex program as a string.

remo.setCgShaderFragmentProgramEntryPoint(index, entryPoint)

Sets the Cg shader fragment program entry point for a shader with a specified index.

index: The zero-based palette index of the shader to set the fragment program entry point for.

entryPoint: The entry point of the fragment program as a string.
remo.getCgShaderFragmentProgramEntryPoint(index, entryPoint)

Reports the Cg shader fragment program entry point for a shader with a specified index.

index: The zero-based palette index of the shader to get the fragment program entry point from.

returns: The entry point of the fragment program as a string.

remo.removeShaderPaths(index)

Removes the paths for a given shader, leaving only the shader source file names.

index: The zero-based index of the shader.
Macros

In order to make a script more user-friendly and appear more or less as a regular Remo 3D tool, it can be wrapped in a macro. A macro will appear in the Macros sub-menu under the Script menu.

![Cylinder Tool](image)

If a macro takes parameters, they will appear in the tool area to the right together with an Execute button, like a regular tool. If no parameters are defined, the tool will be executed immediately.

Macro Editor

A macro is defined as a name, a script (described in the Scripts chapter) and a number of parameters of different types. Macros are handled in the Macro Editor available from the Script menu. From here you can add a new macro, remove an existing macro, import or export macros and edit the name, script file name and parameters of an existing macro.

![Macro Editor](image)

Macro parameters are completely optional; a macro doesn’t have to take any parameters. A parameter is defined by a name and a type (String, Boolean, Number, File or Folder) and is essentially just a variable definition into the script that will be executed.
So if you define a number parameter named x in your macro and set it to 19 when executing the macro, it’s the same writing $x = 19$ in your script.

A file or folder parameter can be specified manually in the parameter text box before activating the macro, or with a browse (...) button next to it.

Note that parameters are preferably named as variable names, for a natural integration with the script using them. They will be modified when presented in the user interface, with capital initial letter and space before any following capital letters. So a parameter named `leftImage` will e.g. be presented as "Left Image".

**Macro Samples**

Remo 3D is delivered with a number of macro samples. Try to import them from the Macro Editor and you will discover that they are not only useful as samples, they also really extend the functionality of Remo 3D in a simple but efficient way. Each macro has its own configuration file that you can import, or you can choose to import all at once with the `all.cfg` file. It’s however recommended to import only the macros you need for the moment, to avoid filling up the Macros menu in Remo 3D. The `samples/macros` folder contains the following sample macros:

---

**box.lua**

Remo 3D script for creating a box, exposed in Remo 3D as a macro with the box.cfg file.

Parameters:
- **Size**: Side length
- **Shaded**: True if shaded

---

**cone.lua**

Remo 3D script for creating a cone, exposed in Remo 3D as a macro with the cone.cfg file.

Parameters:
- **Radius**: Radius of base disk
- **Height**: Height of the cone
- **Segments**: Number of segments of base disk
- **Shaded**: True if shaded
createBlueprintBox.lua

Remo 3D script for creating temporary geometry with given blueprints as base for modeling, exposed Remo 3D as a macro with the createBlueprintBox.cfg file.

Parameters:
- Cross: True if cross-type, false if box-type
- Length: Model length (Y direction)
- Width: Model width (X direction)
- Height: Model height (Z direction)
- FrontImage: File name of front blueprint image
- RearImage: File name of rear blueprint image
- LeftImage: File name of left blueprint image
- RightImage: File name of right blueprint image
- TopImage: File name of top blueprint image
- BottomImage: File name of bottom blueprint image

*NOTE: If rear image is not given, the front image will be used for the front. The same applies to the left and right images, as well as the top and bottom images.*

cylinder.lua

Remo 3D script for creating a cylinder, exposed in Remo 3D as a macro with the cylinder.cfg file.

Parameters:
- Radius: Radius of base disk
- Height: Height of the cylinder
- Segments: Number of segments of base disk
- Shaded: True if shaded

eliminateDoubleVertices.lua

Remo 3D script for removing all consecutive vertices with identical coordinates in all polygons in the selected geometry (or if none selected, in the complete model), exposed in Remo 3D as a macro with the eliminateDoubleVertices.cfg file.

pyramid.lua

Remo 3D script for creating a pyramid, exposed in Remo 3D as a macro with the pyramid.cfg file.

Parameters:
- Base: Side Length of base square
- Height: Height of the pyramid
- Shaded: True if shaded

selectConcavePolygons.lua

Remo 3D script for selecting all concave polygons beneath the currently selected node (or if none selected, in the complete model), exposed in Remo 3D as a macro with the selectConcavePolygons.cfg file.

selectEmptyPolygons.lua

Remo 3D script for selecting all polygons without vertices beneath the currently selected node (or if none selected, in the complete model), exposed in Remo 3D as a macro with the selectEmptyPolygons.cfg file.
selectNonPlanarPolygons.lua

Remo 3D script for selecting all non-planar polygons beneath the currently selected node (or if none selected, in the complete model), exposed in Remo 3D as a macro with the selectNonPlanarPolygons.cfg file.

Parameters:
Tolerance: Tolerance used when measuring distance from plane to vertex.

Sharing Macros

Since macros can be imported and exported in the Macro Editor, they are easy to share between users.

The Export button in the Macro Editor will export all macros currently in the dialog, no matter if they have been saved or not (they are not saved until the OK button has been clicked). A new macro configuration file is specified and the corresponding Lua script files will also be copied to the same destination folder.

The Import button in the Macro Editor will import the macros in the specified macro configuration file, so that they are added to the existing macros currently in the dialog. Nothing is saved until you click the OK button.

Macros are saved in your personal macro configuration file, macro.cfg.

In Microsoft® Windows® 10/8/7, the macro configuration file is found here:
\<WINDRIVE>\Users\<USER>\AppData\Roaming\Remo3D\macro.cfg

where <WINDRIVE> is the Microsoft® Windows® installation drive, e.g. C: and <USER> is your user name.

In Linux, the macro configuration file is found here:
<HOME>/.remo3d/macro.cfg
where <HOME> is your home directory.

However, since import and export of macros are trivial as described above, it’s normally not necessary to handle the macro configuration file itself.
Troubleshooting

This section lists a number of troubles and their solutions.

Remo 3D crashes without explanation, e.g. when loading a texture into the texture palette.

Check that the PATH environment variable is not set to a path where OpenSceneGraph DLLs with a version other than the OpenSceneGraph DLLs installed with Remo 3D are located.

Remo 3D crashes at start-up or at tool execution, or doesn’t show the model view of a loaded file.

Try removing the Remo 3D configuration file.

In Microsoft® Windows® 10/8/7, the configuration file is found here:
<WINDRIVE>\Users\<USER>\AppData\Roaming\Remo3D\remo3d.cfg

where <WINDRIVE> is the Microsoft® Windows® installation drive, e.g. C: and <USER> is your user name.

In Linux, the configuration file is found here:
<HOME>/.remo3d/remo3d.cfg

where <HOME> is your home directory.

A tool introduces inaccuracy in vertex positions.

This problem will be addressed in future versions of Remo 3D, but until then the Quantify tool can be used to fix these inaccuracies.

A shortcut doesn’t work.

A text box in e.g. an active tool can be focused and stops the shortcut from working. Try changing the focus by closing the tool or pressing TAB until no text box is focused.

Remo 3D crashes at loading of a model with materials.

Try installing the latest version of your graphics driver.

Remo 3D hangs under Linux, possibly with X warning messages.

Try setting the environment variable REMO3D_ENABLE_X_THREADING to 1.

Remo 3D hangs for about 5 seconds directly after it has been started.

An old 3Dconnexion Space Navigator™ driver may be installed, without any 3Dconnexion device connected. Try connecting the device or install the newest 3Dconnexion driver.
Cached search results and bookmarks in the Remo 3D help system don’t work.

You have probably installed a new version of Remo 3D and the cache needs to be removed. Remove the config.ini, org.* and .metadata files in the Remo 3D user folder.

In Microsoft® Windows® 10/8/7, the Remo 3D user folder is found here:
\<WINDRIVE>\Users\<USER>\AppData\Roaming\Remo3D\n
where <WINDRIVE> is the Microsoft® Windows® installation drive, e.g. C: and <USER> is your user name.

In Linux, the Remo 3D user folder is found here:
\<HOME>/.remo3d/
where <HOME> is your home directory.

The Remo 3D built-in help system won’t start when choosing Help... from the Help menu.

The installed Java Runtime Environment is probably 32-bit only, even if you're on a 64-bit operating system. Try installing the 64-bit version from http://www.java.com/en/download/manual.jsp.

Remo 3D for Linux won't load images with a certain file format, or the program terminates with a warning about missing dependencies for a certain image format, e.g. GIF, JPG, PNG or TIF.

Install the proper libraries, e.g. by doing the following in Ubuntu for the JPG image format (refer to other Linux documentation for other distributions and image formats):

```
sudo apt-get install libjpeg62
```
Technical Specifications

In this section you will find various technical specifications for Remo 3D™ v2.8.

OpenFlight® File Format

The OpenFlight® versions supported by Remo 3D v2.8 are 15.7, 15.8, 16.0, 16.1 and 16.2. There are however some exceptions from this support; these are listed below:

Note: This means that the user doesn’t have full control of these features in Remo 3D. However, when loading or saving an existing OpenFlight® file containing these features, they will not be removed or modified. Many of these features may be implemented in future versions of Remo 3D. Some of non-supported node types can still be cut, copied, pasted, deleted, dragged and dropped, but their attributes can’t be modified.

15.7:

- Mesh Nodes
- Morph Vertex Lists
- BSP Planes
- Sound
- Roads
- Clip Regions
- Text
- CAT
- Extension Nodes
- Curves
- Vectors
- Bounding Volumes
- Name Tables
- Eyepoint/Trackplane
- Linkage Palette
- Light Source Palette
- Line Style Palette
- Continuation Records
- Relative priority
- Geo-specific control points (in texture attributes)
- Sub-textures (in texture attributes)

15.8:

- Indexed String
Texture File Formats

The following texture file formats are supported by Remo 3D™ v2.8:

- SGI® image file format (.rgb, .rgba, .int, .inta)
- Microsoft® Windows® bitmap (.bmp)
- Direct Draw Surface (.dds)
- Graphics Interchange Format (.gif)
- Joint Photographic Experts Group (.jpg)
- Portable Network Graphics (.png)
- Portable any map (.pnm, .ppm, .pgm, .pbm)
- Targa (.tga)
- Tagged Image File Format (.tif)

Import File Formats

The following import file formats are supported by Remo 3D™ v2.8:

- 3DC Point Cloud (.3dc)
- 3D Studio MAX® (.3ds)
- AC3D™ (.ac)
- COLLADA™ 1.4.1 (.dae)
- Autodesk® FBX® (.fbx)
- Autodesk® DXF (.dxf)
  
  *Note: After importing a DXF file, some of the polygons may have to be reversed by using the Reverse Polygon tool, since the direction (clockwise or counter-clockwise) of polygons is not uniquely defined in the DXF file format.*

- LightWave® (.lwo / .lws)
- Quake Character Models (.md2)
- Alias® Wavefront OBJ (.obj)
- OpenSceneGraph 3.6.3 Deprecated ASCII (.osg)
- OpenSceneGraph 3.6.3 Deprecated Binary (.ive)
- OpenSceneGraph 3.6.3 Archive (.osga)
- OpenSceneGraph 3.6.3 Binary (.osgb)
- OpenSceneGraph 3.6.3 Text (.osgt)
- OpenSceneGraph 3.6.3 XML (.osgx)
- VBS2 P3D (.p3d)
  
  *Note: Textures are generally referenced with absolute paths to where a P3D model was located in the VBS2 system. This may require the use of “Change All Paths” in the texture palette after importing a P3D file.*

- Stanford Triangle Format (.ply)
- StereoLithography Binary (.stl)
- StereoLithography ASCII (.sta)
ESRI Shape (.shp)

*Note: Attributes will be imported as polygon comments, one line of text per attribute, according to the format* `<Name>=<Value>`, *e.g. Perimeter=65.*

Direct X (.x)

When importing files into Remo 3D, embedded textures and shaders, as well as externally referenced files, can be converted and written to disk.

Remo 3D uses OpenSceneGraph plugins for importing different file formats. These plugins can be controlled with a number of settings specified as an option string. This string can be set using the REMO3D_IMPORT_EXPORT_OPTIONS environment variable.

Export File Formats

The following export file formats are supported by Remo 3D™ v2.8:

- 3D Studio MAX® (.3ds)
- AC3D™ (.ac)
- COLLADA™ 1.4.1 (.dae)
- DOT (.dot)
- Autodesk® FBX® (.fbx)
- Alias® Wavefront OBJ (.obj)
- OpenSceneGraph 3.6.3 Deprecated ASCII (.osg)
- OpenSceneGraph 3.6.3 Deprecated Binary (.ive)
- OpenSceneGraph 3.6.3 Binary (.osgb)
- OpenSceneGraph 3.6.3 Text (.osgt)
- OpenSceneGraph 3.6.3 XML (.osgx)
- StereoLithography Binary (.stl)
- VBS2 P3D (.p3d)

*Note: When exporting a P3D file, textures are automatically converted to the TGA file format and put in a data sub-folder which is created in the same location as the exported file. Material RVMAT files are also written to the same data sub-folder. In the P3D file, the textures and materials are referenced locally to the data sub-folder, e.g. “data\texture.tga”. Generally, it will be necessary to change the texture and material paths to absolute paths according to the folder where the model is located in the target VBS2 system. The paths can be changed in the Oxygen 2 tool available in the VBS2 Development Suite.*

Remo 3D uses OpenSceneGraph plugins for exporting different file formats. These plugins can be controlled with a number of settings specified as an option string. This string can be set using the REMO3D_IMPORT_EXPORT_OPTIONS environment variable.

Environment Variables

Remo 3D supports a number of environment variables that allows the user to control certain parts of the tool.

**Setting an Environment Variable**

An environment variable `<name>` can be set to a `<value>` by doing the following:
Linux, tcsh:

setenv <name> <value>

Linux, bash:

export <name>=<value>

Microsoft® Windows®:

Start -> Settings -> Control Panel -> System -> Advanced -> Environment Variables -> New

Enter <name> as variable name and <value> as the variable value.

Environment Variable Names

The environment variables used by Remo 3D™ v2.8 are:

**REMO3D_FILE_PATH**
File path used for loading textures, external references and shaders, multiple paths can be specified separated with semi-colons (;).

**REMO3D_IMPORT_EXPORT_OPTIONS**
Options string to the OpenSceneGraph plugins used for import and export.

**REMO3D_LICENSE_FILE**
Path to the license file.

**REMO3D_LICENSE_SERVER**
The license server address to connect to.

**REMO3D_LICENSESERVER_PORT**
The license server port to connect to.

**REMO3D_LICENSESERVER_FILE**
Path to the license server file.

**REMO3D_ENABLE_X_THREADING**
Can be set to avoid threading issues in Linux.

**REMO3D_SCRIPT_CONFIRMATION**
Can be set to OFF to avoid having to press a key for the script to terminate in Microsoft® Windows®.

**REMO3D_NO_LINE_FEED**
Can be set to ON to avoid automatic line feed after each print call in a script.

**REMO3D_LOG**
Path to a log file where all output can be directed form a script. When REMO3D_LOG is
defined, no console window will open while running a command-line script in Microsoft® Windows®.

**REMO3D_SPACENAV DEVICE**
Path to the Space Navigator™ device in Linux.

## System Configuration

Minimal system configuration for Remo 3D™ v2.8:

- Microsoft® Windows® 10/8/7 64-bit or Linux 64-bit operating system
- Hardware accelerated 3D graphics card
- Minimum 1366x768 screen resolution
- Mouse with three buttons or wheel button
- Java™ Runtime Environment 64-bit (at least Java 6 / JRE 1.6 recommended for help system)

Recommended system configuration for Remo 3D™ v2.8:

- Microsoft® Windows® 10 64-bit or Linux 64-bit operating system
- Intel® i5 or i7 processor
- 16 GB RAM
- NVIDIA® hardware accelerated 3D graphics card
- Minimum 1920x1080 screen resolution
- Mouse with three buttons or wheel button
- Java™ Runtime Environment 64-bit (at least Java 6 / JRE 1.6 recommended for help system)
Development Tools

Remo 3D™ was developed using the following excellent tools and libraries:

**OpenSceneGraph**

[OpenSceneGraph](www.openscenegraph.org)

**OpenGL**

[OpenGL](www.opengl.org)

**FLTK**

[FLTK](www.fltk.org)

**Lua**

[www.lua.org](www.lua.org)
Index

—3—

3D Studio MAX .................................. 169, 170
3DC Point Cloud ......................... 169
3Dconnexion ............................... 22

—A—

Abbreviations ................................ 11
About Remo 3D .............................. 30
Above Grid ................................. 97
AC3D ..................................... 169, 170
Actual Size .................................. 62, 86
administrative privileges ................. 14
Alias ........................................ 169, 170
Align To Polygon (Face) ............... 98
Align To Vertices ......................... 98
Align To XY ................................ 97
Align To XZ ................................ 97
Align To YZ ................................ 98
alpha ......................................... 83
Always On Top .............................. 98
Always Save To Latest Version .......... 96
ambient ...................................... 83, 100
Ambient Intensity ......................... 60, 85
Animation Index ............................ 62
animation preview ......................... 93
Animation State ............................ 62
Animation Type ............................ 87
Appearance .................................. 97
Appearance Index ......................... 62
Apply Attributes .......................... 54
Apply Color ................................ 41
Apply Comment ............................ 69
Apply Material ............................ 42
Apply Matrix ................................ 36, 37, 38
Apply Shader ................................ 43
Apply Texture ................................ 42
Apply Texture Mapping .................. 42
Apply Transformation Matrix .......... 39
arguments .................................... 107
ATTR (Attribute tools) .................. 54
Attribute Script Functions ............. 135
attributes ................................... 54, 76
Autodesk .................................. 169, 170
Automatic External Reference Reload .. 96
Automatic Shader Reload ............... 96
Automatic Texture Reload .............. 96
Back (Along Y) .............................. 97
Back Color .................................. 60, 85
Back Intensity ............................. 60, 85
Background .................................. 99
base layer ................................... 27
Below Grid .................................. 97
BICUBIC ..................................... 78
BILINEAR .................................... 78
Billboard/Alpha ............................ 58
BLEND .......................................... 78
BMP ............................................. 76
Bottom (Along Z) .......................... 96
Built-in Help System ...................... 104

—B—

Calligraphic Draw Order ............... 62
Capture Screen ........................... 98
Catmull-Clark ............................. 51
Cg ............................................. 89
Change All Paths ......................... 76
Change Index .............................. 76, 82, 83, 88, 90
Change Parent ............................ 27
Character Rate ............................ 87
Clamp ......................................... 79, 80
clipboard ................................... 16, 17
Clipboard/Edit Script Functions .... 114
Close .......................................... 26
Close All Models .......................... 26
COLLADA .................................... 169, 170
color .......................................... 41, 42, 63, 74
Color Change Index ....................... 87
Color Change Intensity .................. 88
Color Palette .............................. 19, 29, 74
Color Palette Script Functions ....... 140
command ...................................... 17, 91
Command List ............................. 27, 91
cmd-and-line ............................... 16, 106
comment ...................................... 69
color ........................................... 31
Conform To Terrain ...................... 85
colord ......................................... 106
Contents ..................................... 30
contracting ................................... 72
Control Point ............................. 80
correlation kernel ......................... 79
Coordinate .................................. 63
Coordinate Units .......................... 55
coordinates ................................. 20
Coordinates and Selections Field ...... 20

175
Copy .......................................... 16, 26
corner points .................................. 70
Counter Clockwise .......................... 61, 87
CREATE (Creation tools) ................. 40
Create Disk .................................. 40
Create External Reference ............... 41
Create Light Point ......................... 41
Create Node .................................. 40
Create Polygon ............................... 40
Create Sphere ............................... 40
Creation Script Functions ................. 126
Current Color (COL) ......................... 19
Current Light Point (LPT) ................ 19
Current Material (MAT) ................. 19
current parent ............................... 20
Current Shader (SHD) ...................... 19
Current Texture (TEX) .................... 19
Current Texture Mapping (TXM) ......... 19
Cut ........................................... 16, 26

—D—

Database Offset X .......................... 56
Database Offset Y .......................... 56
Database Version .......................... 55
DDS .......................................... 76
DECAL ........................................ 78
Decrease Grid Size .......................... 98
Defaults ..................................... 96, 100
degree-of-freedom .......................... 70
Delete ........................................ 27
Detail Texture ............................... 59, 79
Development Tools ......................... 173
diffuse ........................................ 83, 100
Direct Draw Surface ....................... 169
Direct X ....................................... 170
Directional Type ............................ 60, 85
disk ............................................ 40
Display At Dusk ............................. 57, 60, 85
Display At Night ............................ 57, 60, 85
Display In Daylight ....................... 57, 60, 85
Display Mode ............................... 60, 85
distance ....................................... 20
DOF ............................................ 70
DOF Attributes .............................. 64
DOF Mode ..................................... 29
DOF preview .................................. 92
DOT (.dot) .................................... 170
Double Grid Size ........................... 98
double-clicking ............................. 54, 73
drag and drop ............................... 16, 72
Draw Mode .................................. 57
Draw Wireframe Backface ................ 97
Drawing Order ............................... 61
Duration ...................................... 87

—E—

DXF ............................................ 169

Earth Ellipsoid Model ...................... 55
Earth Model ................................... 77
Edge ........................................... 70
Edge Mode .................................... 28
Edit Color .................................... 74
Edit Icons .................................... 16
Edit Menu .................................... 26
Eliminate Transformation Matrices ....... 39
Embedded textures .......................... 25
emissive ....................................... 83
Enabled ....................................... 63
Enabled Period .............................. 61, 87
Environment .................................. 78
Environment Variable Names ............. 171
Environment Variables .................... 170
ESRI .......................................... 170
evaluation .................................... 103
Exit ............................................ 26
expanding ...................................... 72
Export As ..................................... 26
Export File Formats ....................... 170
expression ..................................... 31
External Format (read-only) ................ 77
External Reference ......................... 71
External Reference Attributes ............ 68
External Reference Mode .................. 29
external references ........................ 41
External References ....................... 25
Extrude ....................................... 50

—F—

Fade In Duration ............................ 86
Fade Out Duration .......................... 86
Fade Range Ratio ........................... 86
Fading ......................................... 62, 86
Falloff Exponent ............................ 60, 85
far clip ......................................... 99
far range ...................................... 64
FBX ............................................ 169, 170
Feature ID .................................... 57, 60, 85
field-of-view ................................... 99
File Dialogs .................................. 23
File Icons .................................... 16
File Menu ..................................... 25
File Script Functions ...................... 112
Fit In View .................................... 97
Fit To Selection ............................. 97
Flashing ....................................... 61, 87
Flat Shaded ................................... 57
<table>
<thead>
<tr>
<th>Flipped DDS/PNM Images</th>
<th>Image Origin</th>
</tr>
</thead>
<tbody>
<tr>
<td>floating</td>
<td>.76</td>
</tr>
<tr>
<td>Floating licenses</td>
<td>.101</td>
</tr>
<tr>
<td>FLTK</td>
<td>.173</td>
</tr>
<tr>
<td>Fog Intensity</td>
<td>.61, 86</td>
</tr>
<tr>
<td>Fog Punch</td>
<td>.62, 86</td>
</tr>
<tr>
<td>Fog Scale</td>
<td>.61, 86</td>
</tr>
<tr>
<td>Footprint</td>
<td>.57</td>
</tr>
<tr>
<td>Format Version</td>
<td>.55</td>
</tr>
<tr>
<td>fragment program</td>
<td>.89</td>
</tr>
<tr>
<td>freeze</td>
<td>.35, 46</td>
</tr>
<tr>
<td>Freeze LOD Center</td>
<td>.64</td>
</tr>
<tr>
<td>Front (Along -Y)</td>
<td>.97</td>
</tr>
<tr>
<td>Front Intensity</td>
<td>.60, 85</td>
</tr>
<tr>
<td>Frozen Normal</td>
<td>.63</td>
</tr>
<tr>
<td>Frustum</td>
<td>.99</td>
</tr>
<tr>
<td>Full Size</td>
<td>.75</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>General Script Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>GIF</td>
</tr>
<tr>
<td>Gizmo</td>
</tr>
<tr>
<td>Graphics card</td>
</tr>
<tr>
<td>Graphics driver</td>
</tr>
<tr>
<td>Graphics Interchange Format</td>
</tr>
<tr>
<td>Grid</td>
</tr>
<tr>
<td>Grid Offset</td>
</tr>
<tr>
<td>Grid Script Functions</td>
</tr>
<tr>
<td>Grid Spacing</td>
</tr>
<tr>
<td>Group</td>
</tr>
<tr>
<td>Group Attributes</td>
</tr>
<tr>
<td>Group Mode</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Half Grid Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hard Edge</td>
</tr>
<tr>
<td>hardware fingerprint</td>
</tr>
<tr>
<td>Head Light</td>
</tr>
<tr>
<td>Header Attributes</td>
</tr>
<tr>
<td>Help</td>
</tr>
<tr>
<td>Help Menu</td>
</tr>
<tr>
<td>hidden</td>
</tr>
<tr>
<td>Hide Selected</td>
</tr>
<tr>
<td>Hiding</td>
</tr>
<tr>
<td>hierarchy</td>
</tr>
<tr>
<td>Horizontal Lobe Angle</td>
</tr>
<tr>
<td>HSV</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Illuminated</th>
</tr>
</thead>
<tbody>
<tr>
<td>.57</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Image Origin</th>
</tr>
</thead>
<tbody>
<tr>
<td>.77</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Import</th>
</tr>
</thead>
<tbody>
<tr>
<td>.25</td>
</tr>
<tr>
<td>Import File Formats</td>
</tr>
<tr>
<td>inaccuracies</td>
</tr>
<tr>
<td>increase complexity</td>
</tr>
<tr>
<td>Increase Grid Size</td>
</tr>
<tr>
<td>Increment</td>
</tr>
<tr>
<td>Indexed Light Point Attributes</td>
</tr>
<tr>
<td>Installation</td>
</tr>
<tr>
<td>Installed Contents</td>
</tr>
<tr>
<td>INT</td>
</tr>
<tr>
<td>INTA</td>
</tr>
<tr>
<td>Intel</td>
</tr>
<tr>
<td>Intensity</td>
</tr>
<tr>
<td>Internal Format</td>
</tr>
<tr>
<td>Invert Selection</td>
</tr>
<tr>
<td>IR Color Code</td>
</tr>
<tr>
<td>IR Material Code</td>
</tr>
<tr>
<td>Isolate Selected</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Java</th>
</tr>
</thead>
<tbody>
<tr>
<td>.105, 172</td>
</tr>
<tr>
<td>Joint Photographic Experts Group</td>
</tr>
<tr>
<td>JPEG</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Large vertices</th>
</tr>
</thead>
<tbody>
<tr>
<td>.100</td>
</tr>
<tr>
<td>Last Frame Duration</td>
</tr>
<tr>
<td>Least Detail</td>
</tr>
<tr>
<td>Left (Along X)</td>
</tr>
<tr>
<td>Less Detail</td>
</tr>
<tr>
<td>level-of-detail</td>
</tr>
<tr>
<td>License Handling</td>
</tr>
<tr>
<td>license key</td>
</tr>
<tr>
<td>licenseserver.dat</td>
</tr>
<tr>
<td>Light Mode</td>
</tr>
<tr>
<td>Light Point</td>
</tr>
<tr>
<td>Light Point Animation Attributes</td>
</tr>
<tr>
<td>Light Point Appearance Attributes</td>
</tr>
<tr>
<td>Light Point Attributes</td>
</tr>
<tr>
<td>Light Point Mode</td>
</tr>
<tr>
<td>Light Point Palette</td>
</tr>
<tr>
<td>Light Point Palette Script Functions</td>
</tr>
<tr>
<td>Light Point System</td>
</tr>
<tr>
<td>Light Point System Attributes</td>
</tr>
<tr>
<td>Light Point System Mode</td>
</tr>
<tr>
<td>light points</td>
</tr>
<tr>
<td>lighting</td>
</tr>
<tr>
<td>LightWave</td>
</tr>
<tr>
<td>Limit</td>
</tr>
<tr>
<td>Line Style</td>
</tr>
</tbody>
</table>
Linux ........................................ 14, 172
Linux File Dialogs .................................. 24
Load Texture ........................................ 76
Lobe Roll Angle ...................................... 60, 85
LOD ................................................... 70
LOD Attributes ..................................... 64
LOD Center ........................................... 64
LOD Mode ........................................... 29
LOD Range Ratio ................................... 86
LOD Scale ............................................ 86
Loop Count ........................................... 56
Loop Duration ...................................... 56
Lua ................................................... 106, 173
—M—
macro .................................................. 162
macro configuration file ......................... 165
Macro Editor ......................................... 162
Macro Samples ...................................... 163
Macros ............................................. 104, 162
Magnification Filter ............................... 78
Map Texture ........................................ 44
Mapping texture and adding to palette 45
Mapping texture without adding to palette 45
mask ................................................... 67
matching ............................................ 31
material .............................................. 42, 43, 59, 83
Material Palette .................................... 19, 29, 83
Material Palette Script Functions ............ 145
Max Commands ....................................... 95
Maximum Defocus .................................. 61, 85
Minimum Defocus .................................. 61, 85
Minimum Pixel Size ................................. 62, 86
Menu ................................................. 25
Menu Bar ............................................. 15
Merge Polygons ..................................... 53
Microsoft ............................................ 171, 172
Microsoft® Windows® .............................. 14
Minification Filter ................................... 78
MIP Kernel .......................................... 79
mipmap .............................................. 78
Mirror ................................................ 49, 79
Misc Menu .......................................... 30
Miscellaneous Icons ............................... 17
Miscellaneous Mouse Functions ............ 22
Mode Menu .......................................... 28
Model View .......................................... 20
Modes ............................................... 70
Modification Script Functions ............... 132
MODIFY (Modification tools) .................. 49
Modify UV ........................................... 48
MODULATE .......................................... 78
More Detail .......................................... 27
Morse Code ......................................... 87
Morse Code Timing ................................. 87
Most Detail ......................................... 27
Mouse Coordinate Indication ................. 22
Mouse Handling ..................................... 21
Mouse Movement ................................... 21
Mouse Selection .................................... 21
Move .................................................. 35
Move Texture Map .................................. 45
Multi-texture ........................................ 27
Multi-texture Data .................................. 59
Multi-texture Effect ............................... 59
multi-texturing ..................................... 44
—N—
N/A (not available) ................................. 95
NE Corner Latitude ................................. 55
NE Corner Longitude ............................... 55
near clip ............................................. 99
near range ......................................... 64
Network Adapter .................................... 101
Never On Top ....................................... 98
New ..................................................... 16, 25
New Textured Polygon White .................... 55
Next IDs ............................................. 55
Next Model .......................................... 27
Node Colors ......................................... 73
node-locked ........................................ 101
Node-locked and USB dongle licenses 101
Normal .............................................. 63
normals ............................................. 100
numbers ............................................. 18
NVIDIA ............................................ 172
—O—
OBJ ................................................... 169, 170
Object ................................................. 70
Object Attributes .................................. 56
Object Mode ......................................... 29
Open .................................................... 16, 25
Open Recent ........................................ 25
OpenFlight ......................................... 16, 168
OpenFlight® File Format ......................... 168
OpenGL ............................................. 173
OpenGL Shading Language ..................... 89
OpenSceneGraph ................................. 166, 169, 170, 173
orientation axes ................................... 100
Origin ................................................. 64
Origin Latitude ..................................... 56
Origin Longitude ................................... 56
Orthographic .................................... 96
Other Icons ..................................... 15
Overview ......................................... 13

—P—

P3D ...................................... 169, 170
Palette Icons .................................... 19
Palettes ........................................... 74
Palettes Menu .................................. 29
panning ..................................... 21, 94
Parameter Precision .......................... 95
Paste ........................................ 17, 27
PATH ............................................ 166
PBM ................................................ 76
Period ........................................ 61, 87
Perspective ....................................... 96
Perspective Mode................................ 62, 86
PGM .............................................. 76
Phase Delay .................................... 61, 87
pivot ........................................ 17, 77
pivot point ...................... 17, 22, 35, 45
pixel size threshold ......................... 99
Plant ........................................... 36
PNG .............................................. 76
PNM .............................................. 76
POINT ............................................. 78
Point In XY Plane.............................. 65
Point On X Axis............................... 65
Polygon ......................................... 40, 70
Polygon Attributes ............................ 57
Polygon Mode.................................. 28
Portable any map ............................... 169
Portable Network Graphics .............. 169
PPM .............................................. 76
Preferences ................................. 27, 94
Preview Animation .......................... 93
Preview Animation.......................... 27
Preview DOF .................................. 27, 92
Previous Model ............................... 27
Primary Color ................................ 58
primitive ...................................... 70
Projection ....................................... 77
Projection Type ................................. 55
Proximity Occulting ......................... 61, 86
Purchase ........................................ 30
Purchase dialog ................................ 101

—Q—

Quake Character Models .................. 169
Quality .......................................... 61, 86
Quantify ........................................ 53
radius ............................................ 38
Randomize Intensity ..................... 61, 86
Range Mode ..................................... 61, 86
RCP (Remo 3D Color Palette) ............. 75
Read Color From Node ...................... 42
Read Material From Node .................. 43
Read Shader From Node ..................... 43
Read Texture From Node .................... 42
Read Texture Mapping From Node ........ 42
Real World Size .............................. 77
Recall View .................................... 97
Redo .......................................... 17, 26
reduce complexity ......................... 52
Reflective Only .............................. 62, 86
REMO3D_ENABLE_X_THREADING........... 171
REMO3D_FILE_PATH ....................... 41, 76, 90
REMO3D_FILE_PATH File path ............. 171
REMO3D_IMPORT_EXPORT_OPTIONS171
REMO3D_LICENSE_FILE .................... 171
REMO3D_LICENSE_SERVER ................. 103
REMO3D_LICENSE_SERVER ................. 171
REMO3D_LICENSE_SERVER_FILE ......... 171
REMO3D_LICENSE_SERVER_PORT ...... 103
REMO3D_LICENSE_SERVER_PORT ....... 171
REMO3D_LOG ................................ 171
REMO3D_NO_LINE_FEED ................. 171
REMO3D_SCRIPT_CONFIRMATION...... 107
REMO3D_SCRIPT_CONFIRMATION .......... 171
REMO3D_SPACENAV_DEVICE .......... 172
remoLicenseServer ......................... 102
Remove All Paths ............................ 76
Remove Material ............................ 43
Remove Path .................................. 76
Remove Shader ............................... 43
Remove Texture .............................. 43
Remove Texture Mapping .................. 43
Remove Unused .............................. 76, 83, 88, 90
Repeat ........................................ 79
Replace Texture ............................. 76
Replications ................................. 56, 57, 60, 62
Report Differences ......................... 18
Report Statistics ............................ 18
Reporting Script Functions ............... 138
Reverse Polygon ............................. 49
Revert ......................................... 25
Revert Externals ............................. 25
Revert Textures .............................. 25
Revision Date .................................. 55
Revolve ....................................... 51
RGB ............................................ 74, 76, 83
RGB Mode ..................................... 55
RGBA ........................................... 76
Right (Along -X) ............................. 97
### Remo 3D™ v2.8 User’s Guide

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>RLP (Remo 3D Light point Palette)</td>
<td>88</td>
</tr>
<tr>
<td>RMP (Remo 3D Material Palette)</td>
<td>83</td>
</tr>
<tr>
<td>Rotate</td>
<td>36</td>
</tr>
<tr>
<td>Rotate Texture Map</td>
<td>46</td>
</tr>
<tr>
<td>rotating</td>
<td>21, 61, 87</td>
</tr>
<tr>
<td>rotation</td>
<td>94</td>
</tr>
<tr>
<td>Rotation Axis</td>
<td>61, 87</td>
</tr>
<tr>
<td>RSP (Remo 3D Shader Palette)</td>
<td>89</td>
</tr>
<tr>
<td>RTP (Remo 3D Texture Palette)</td>
<td>76</td>
</tr>
<tr>
<td>RVMAT</td>
<td>170</td>
</tr>
<tr>
<td>RXP (Remo 3D texture mapping Palette)</td>
<td>82</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>sample</td>
<td>104</td>
</tr>
<tr>
<td>Sample Models</td>
<td>104</td>
</tr>
<tr>
<td>Samples</td>
<td>104</td>
</tr>
<tr>
<td>Save</td>
<td>16, 25</td>
</tr>
<tr>
<td>Save As</td>
<td>25</td>
</tr>
<tr>
<td>Save Vertex Normals</td>
<td>55</td>
</tr>
<tr>
<td>Scale</td>
<td>37</td>
</tr>
<tr>
<td>Scale Texture Map</td>
<td>47</td>
</tr>
<tr>
<td>Scramble</td>
<td>79</td>
</tr>
<tr>
<td>Screen Capture Size</td>
<td>95</td>
</tr>
<tr>
<td>screen resolution</td>
<td>172</td>
</tr>
<tr>
<td>Scribed</td>
<td>97</td>
</tr>
<tr>
<td>script</td>
<td>106, 162</td>
</tr>
<tr>
<td>Script Editor</td>
<td>106</td>
</tr>
<tr>
<td>Script Environment</td>
<td>106</td>
</tr>
<tr>
<td>Script Menu</td>
<td>30</td>
</tr>
<tr>
<td>Script Reference</td>
<td>109</td>
</tr>
<tr>
<td>Script Samples</td>
<td>107</td>
</tr>
<tr>
<td>Scripts</td>
<td>104, 106</td>
</tr>
<tr>
<td>Secondary Color</td>
<td>58</td>
</tr>
<tr>
<td>Select By Attributes</td>
<td>28, 31</td>
</tr>
<tr>
<td>Select By Name</td>
<td>28</td>
</tr>
<tr>
<td>Select Children</td>
<td>28</td>
</tr>
<tr>
<td>Select Parents</td>
<td>28</td>
</tr>
<tr>
<td>Select Polygons By Number Of Vertices</td>
<td>28</td>
</tr>
<tr>
<td>Select Previous</td>
<td>28</td>
</tr>
<tr>
<td>Select Sliver Polygons</td>
<td>28</td>
</tr>
<tr>
<td>Selection Menu</td>
<td>28</td>
</tr>
<tr>
<td>Selection Mode Icons</td>
<td>17</td>
</tr>
<tr>
<td>Selection Script Functions</td>
<td>115</td>
</tr>
<tr>
<td>sensitivity</td>
<td>94</td>
</tr>
<tr>
<td>Sensitivity</td>
<td>27</td>
</tr>
<tr>
<td>Sequence</td>
<td>56</td>
</tr>
<tr>
<td>Sequence Number</td>
<td>87</td>
</tr>
<tr>
<td>service</td>
<td>102</td>
</tr>
<tr>
<td>Set Parent</td>
<td>27</td>
</tr>
<tr>
<td>Setting an Environment Variable</td>
<td>170</td>
</tr>
<tr>
<td>Settings</td>
<td>94</td>
</tr>
<tr>
<td>SGI</td>
<td>76</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shade</td>
<td>53</td>
</tr>
<tr>
<td>shader</td>
<td>43, 59, 89</td>
</tr>
<tr>
<td>Shader On</td>
<td>97</td>
</tr>
<tr>
<td>Shader Palette</td>
<td>19, 30, 89</td>
</tr>
<tr>
<td>Shader Palette Script Functions</td>
<td>156</td>
</tr>
<tr>
<td>Shadow</td>
<td>57</td>
</tr>
<tr>
<td>Shape</td>
<td>170</td>
</tr>
<tr>
<td>Sharing Macros</td>
<td>165</td>
</tr>
<tr>
<td>Shift Far Left</td>
<td>27</td>
</tr>
<tr>
<td>Shift Far Right</td>
<td>27</td>
</tr>
<tr>
<td>Shift Left</td>
<td>27</td>
</tr>
<tr>
<td>Shift Right</td>
<td>27</td>
</tr>
<tr>
<td>shifting</td>
<td>73</td>
</tr>
<tr>
<td>shininess</td>
<td>83</td>
</tr>
<tr>
<td>shortcut</td>
<td>166</td>
</tr>
<tr>
<td>Shortcuts</td>
<td>32</td>
</tr>
<tr>
<td>Show All LODs</td>
<td>27</td>
</tr>
<tr>
<td>Show Coordinate System</td>
<td>65</td>
</tr>
<tr>
<td>Show Gizmos</td>
<td>27</td>
</tr>
<tr>
<td>Show Grid</td>
<td>97</td>
</tr>
<tr>
<td>Show Normals</td>
<td>18</td>
</tr>
<tr>
<td>Show Vertex Numbers</td>
<td>18</td>
</tr>
<tr>
<td>Show Vertices Large</td>
<td>18</td>
</tr>
<tr>
<td>Shrink Polygons</td>
<td>17</td>
</tr>
<tr>
<td>Significance</td>
<td>61, 85</td>
</tr>
<tr>
<td>Simplify</td>
<td>52</td>
</tr>
<tr>
<td>Size Difference Threshold</td>
<td>61, 86</td>
</tr>
<tr>
<td>Sky</td>
<td>99</td>
</tr>
<tr>
<td>Sky Light</td>
<td>97</td>
</tr>
<tr>
<td>Slice</td>
<td>50</td>
</tr>
<tr>
<td>Small Feature Culling</td>
<td>99</td>
</tr>
<tr>
<td>Snap To Grid</td>
<td>18</td>
</tr>
<tr>
<td>Soft Selection</td>
<td>38</td>
</tr>
<tr>
<td>Solid</td>
<td>97</td>
</tr>
<tr>
<td>Space Navigator™ Support</td>
<td>22</td>
</tr>
<tr>
<td>specular</td>
<td>83, 100</td>
</tr>
<tr>
<td>sphere</td>
<td>40</td>
</tr>
<tr>
<td>split</td>
<td>50</td>
</tr>
<tr>
<td>Split/Triangulate</td>
<td>50</td>
</tr>
<tr>
<td>Stanford Triangle Format (.ply)</td>
<td>169</td>
</tr>
<tr>
<td>State</td>
<td>87</td>
</tr>
<tr>
<td>statistics</td>
<td>18</td>
</tr>
<tr>
<td>Status Area</td>
<td>20</td>
</tr>
<tr>
<td>Stereolithography</td>
<td>169, 170</td>
</tr>
<tr>
<td>Store View</td>
<td>97</td>
</tr>
<tr>
<td>Subdivide</td>
<td>51</td>
</tr>
<tr>
<td>sub-section</td>
<td>80</td>
</tr>
<tr>
<td>SURFACE (Surface appearance tools)</td>
<td>41</td>
</tr>
<tr>
<td>Surface Material Code</td>
<td>57, 60, 85</td>
</tr>
<tr>
<td>Surface Script Functions</td>
<td>128</td>
</tr>
<tr>
<td>SW Corner Latitude</td>
<td>55</td>
</tr>
<tr>
<td>SW Corner Longitude</td>
<td>55</td>
</tr>
<tr>
<td>SW Database Coordinate X</td>
<td>56</td>
</tr>
<tr>
<td>SW Database Coordinate Y</td>
<td>56</td>
</tr>
</tbody>
</table>
Swing .............................................56
Switch............................................71
Switch Attributes............................67
Switch In.........................................64
switch masks...................................67
Switch Mode....................................29
Switch Out.......................................64
Switch Range...................................64
System Configuration.......................172

—T—
Tagged Image File Format...............169
Targa.............................................169
Technical Specifications..................168
Terrain............................................57
TEXTMAP (Texture mapping tools)........44
texture..........................................42, 43, 59, 75, 87
Texture Attributes..........................76, 77
Texture Coordinate..........................63
Texture File Formats.......................169
Texture List.....................................75
texture mapping.............................42, 43, 44, 81
Texture Mapping Index.....................59
Texture Mapping Palette....................19, 29, 81
Texture Mapping Palette Script Functions
..................................................149
Texture Mapping Script Functions........130
Texture On.......................................97
Texture Palette..............................19, 29, 75
Texture Palette Script Functions.........142
Texture Size....................................77
Textured Polygon White.....................58
TGA.............................................76
TIFF.............................................76
Tile.................................................80
time limited license.........................103
tolerance...........................................95
Tool Area.........................................20
Tool Icons.......................................15
Tools...............................................29, 35
Tools Menu.......................................29
Top (Along -Z)..................................96
transformation matrix.......................39
Transformation Script Functions.........120
Transition.........................................64
Translate Origin................................98
Transparency.................................57, 58
Transparent Clamp.........................62, 86
Transparent Falloff Exponent.............61, 86
Transparent Pixel Size.....................61, 86
Transparent Scale............................62, 86
trees.................................................36
triangulate.......................................50

Troubleshooting .............................166
Tutorial...........................................104

—U—
Undo..............................................17, 26
Unhide All........................................98
Unhide Selected..............................98
Up Vector.........................................77
USB dongle......................................101
Use Current Rotation.......................65
Use Current Translation....................65
Use External Color Palette.................68
Use External Light Point Palette.........68
Use External Light Source Palette.......68
Use External Line Style Palette..........68
Use External Material Palette.............68
Use External Sound Palette...............68
Use External Texture Palette..............68
Use Relative Paths.........................76

—V—
VBS2............................................169, 170
Vertex.............................................70
Vertex Attributes............................63
Vertex Mode.....................................28
vertex program...............................89
Vertical Lobe Angle..........................60, 85
vertices...........................................18
View Menu.......................................27
Viewpoint.........................................96
Viewport..........................................28
Viewport Menu.................................28, 96
Viewport Preferences.......................98, 99
Visibility Range..............................86
Visual Cues.......................................100

—W—
Window Areas..................................15
Windows .........................................171, 172
Windows File Dialogs.......................23
Wireframe........................................97
Word Rate.........................................87
Wrap...............................................79

—X—
XFORM (Transformation tools).............35
Z-buffer ........................................ 98
zoom ........................................... 73
zooming ........................................ 21, 94