

# What's New in Maya

Version 6.5



## Legal Notices

Maya®, Version 6.5

© Copyright 2005 Alias Systems Corp. ("Alias"). All rights reserved.

Graph Layout Toolkit

© Copyright 1992-1997 Tom Sawyer Software, Berkeley, California. All rights reserved.

All documentation ("Documentation") is copyrighted ©2005 Alias and contains proprietary and confidential information of Alias. The Documentation is protected by national and international intellectual property laws and treaties. All rights reserved. Use of the Documentation is subject to the terms of the license agreement that governs the use of the software product to which the documentation pertains (the "Software"). The authorized licensee of the Software is hereby authorized to print no more than one (1) hardcopy of any Documentation provided in digital format per valid license of the Software held by such licensee. Except for the foregoing, the Documentation may not be translated, copied or duplicated in any form (physically or electronically), in whole or in part, without the prior written consent of Alias.

Alias, the swirl logo and Maya are registered trademarks and the Maya logo, Conductors, Trax, IPR, Maya Shockwave 3D Exporter and MEL are trademarks of Alias in the United States and/or other countries worldwide. FBX is a registered trademark of Systèmes Alias Québec Inc. SGI, IRIX, Open GL and Silicon Graphics are registered trademarks of Silicon Graphics, Inc. in the United States and/or other countries worldwide. mental ray and mental images are registered trademarks of mental images GmbH & CO. KG. in the United States and/or other countries. Lingo, Macromedia, Director, Shockwave and Macromedia Flash are trademarks or registered trademarks of Macromedia, Inc. Wacom is a trademark of Wacom Co., Ltd. NVidia is a registered trademark and Gforce is a trademark of NVidia Corporation. Linux is a registered trademark of Linus Torvalds. Intel and Pentium are registered trademarks of Intel Corporation. Red Hat is a registered trademark of Red Hat, Inc. ActiveX, Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Mac, Macintosh and QuickTime are trademarks of Apple Computer, Inc., registered in the United States and other countries. Adobe, Adobe Illustrator, Photoshop and Acrobat are either registered trademarks or trademarks of Adobe Systems Incorporated. UNIX is a registered trademark, licensed exclusively through X/Open Company, Ltd. AutoCAD, Discreet Logic, Inferno and Flame are either registered trademarks or trademarks of Autodesk, Inc. in the USA and/or other countries. OpenFlight is a registered trademark of MultiGen Inc. Java is a registered trademark of Sun Microsystems, Inc. RenderMan is a registered trademark of Pixar Corporation. Softimage is either a registered trademark or trademark of Avid Technology, Inc. in the United States and/or other countries.

All other trademarks mentioned herein are the property of their respective owners.

All images © Copyright Alias unless otherwise noted.



# Table of Contents

<b>1 About Maya 6.5</b> .....	<b>13</b>
What's new and improved in Maya 6.5 .....	13
<b>2 File referencing</b> .....	<b>15</b>
About file referencing improvements .....	15
General file referencing improvements .....	15
File reference edits .....	15
File reference node architecture .....	16
Selective preload of file references .....	17
Maya API .....	17
File referencing MEL update .....	17
Grouping file references .....	18
Multilevel file references .....	18
Create Reference options .....	18
Reference Editor .....	23
Preload Reference Editor .....	27
Proxy references .....	27
Working with proxy references .....	31
Proxy Options .....	42
<b>3 Performance improvements</b> .....	<b>47</b>
About Maya performance improvements .....	47
General .....	48
Opening scenes with large polygonal data sets .....	48
Shaded and wireframe large scenes display draw and refresh .....	49
Polygonal objects with per face shader assignments .....	49
Particle sprite display .....	49

## Table of Contents

Polygon color per vertex draw	49
Hardware shader drawing	49
NURBS shaded display	50
Subdivision surfaces components display	50
Scene view speedup: Use default material	50
Binary save over a network	51
Import of polygon .obj files (Windows)	51
Animation	51
Skinned geometry playback	51
Skeleton draw	52
Precomputation of triangles for faster playback	52
Character Setup and Deformers	52
Wrap deformer setup and playback	52
Update for Sculpt deformations	53
Cluster deformations on polygonal models	53
Soft Modification Tool	53
Smooth skin weights in the Component Editor	53
Modeling	54
Polygons Reduce	54
Polygon editing with color per vertex	54
UV Texture Editor	54
Polygons > Combine	55
Polygons > Mirror Geometry	55
Polygons > Smooth Proxy	55
Polygons > Cleanup	55
Edit Polygons > Subdivide	55
Edit Polygons > Merge Vertices	55
Edit Polygons > Delete Vertex	55
Edit Polygons > Delete Faces	56
Edit Polygons > Separate	56

## Table of Contents

Edit Polygons > Colors .....	56
Edit Polygons > Normals .....	56
Subdivision surfaces display .....	56
Soft Modification Tool .....	56
General rendering .....	57
New user preference for maximum hardware texture resolution .....	57
Improved normal mapping .....	57
Swatch updating in Hypershade .....	57
mental ray for Maya rendering .....	57
Improved Object Instancing .....	57
Animation detection .....	58
Painting .....	58
Artisan tools .....	58
3D Paint Tool .....	59
Fluid Effects .....	59
Ocean shader playback .....	59
Fluids Solver .....	59
Fur .....	60
Hair .....	60
Cloth .....	60
Nodes and Attributes .....	60
PolyBlindData node .....	60
API .....	61
Plug-in Manager updating .....	61
MFnMesh::create .....	61
MFnMesh acceleration intersection methods .....	62
Iteration over several MFn types .....	62

## Table of Contents

<b>4 Other improvements</b> .....	<b>63</b>
What else is new and improved? .....	63
<b>General</b> .....	<b>65</b>
Alias DirectConnect included with Maya .....	65
FBX plug-in included with Maya .....	66
CGFx plug-in (Windows) .....	66
Subdivision surfaces heads-up display .....	67
Improved Layer Editor navigation .....	68
Maya Web browser improvements .....	68
Improved wheel mouse support .....	69
Scaling a light or camera locator .....	69
Environment variable expansion in Maya file browser .....	70
Examples .....	70
Preference .....	71
Other notes .....	72
Create new layouts in the Component Editor .....	73
Adding custom shared shelves .....	73
Virtual Memory limit of Maya (Windows) .....	74
<b>Modeling</b> .....	<b>74</b>
Soften extruded polygon edges and faces .....	74
Soften split polygons .....	75
Polygon Bevel improvements .....	77
Merge Vertices .....	77
Subdivide Ngons .....	78
Smoothing Angle .....	78
Mitering Angle .....	79
UV Texture Editor improvements .....	80

## Table of Contents

Image > Selected Images improved .....	80
Dim Image .....	81
Subdivision surface improvements .....	81
<b>Animation .....</b>	<b>82</b>
New animation clip per channel offset controls .....	82
Channel Offsets .....	83
Copy Channel Offsets .....	84
New channel state controls .....	85
Make channels keyable/nonkeyable from the Channel Box ..	85
Improved Channel Control Editor .....	86
How do I make channels keyable or nonkeyable in Maya 6.5?	87
Improved animation-constraint blending .....	89
New Lock Output attribute .....	90
Custom playback speed .....	91
<b>Character Setup and Deformers .....</b>	<b>92</b>
Improved Copy Smooth Skin Weights .....	92
How do I copy smooth skin weights between polygonal meshes in Maya 6.5? .....	93
Bind skin to non-joint objects .....	95
Improved Smooth Bind Options .....	95
Prevent attachment of zero-weighted smooth skin influences ..	96
Maintain Max Influences .....	97
Add influence objects at any pose .....	97
Related topics .....	98
Automatic joint radii according to bone length .....	98
New Joint Tool settings .....	98
New Joint Radius attribute .....	99

## Table of Contents

<b>Rendering</b> . . . . .	<b>99</b>
Improved Hypershade navigation . . . . .	99
New flags for the Command Line Renderer . . . . .	100
Maya vector rendering . . . . .	102
New maximum resolution for vector rendering . . . . .	102
Maya hardware rendering. . . . .	102
Unlimited file texture size . . . . .	102
New attribute for Hardware Shader Plug-in . . . . .	102
<b>mental ray for Maya rendering</b> . . . . .	<b>103</b>
Summary of major improvements. . . . .	103
Rendering with mental ray for Maya and mental ray satellite. . . . .	103
Overview . . . . .	103
Master machine setup (Maya) . . . . .	104
Improved Final Gather method . . . . .	106
Tips to retune scenes with the new Final Gather method . . . . .	106
New Rapid Scanline method . . . . .	108
New shader library . . . . .	108
Improved displacement approximation in mental ray for Maya . . . . .	109
Improved approximation on Subdivision surfaces. . . . .	110
New Render Stats for image-based lighting (IBL). . . . .	110
New image-based lighting (IBL) attributes . . . . .	110
Improved Baking . . . . .	111
New light and texture shaders . . . . .	112
New Dynamic Attributes . . . . .	113
mental ray for Maya support of native Render Stats . . . . .	115
Texture bake-set Final Gather Quality . . . . .	115
New Vertex bake-set functionality . . . . .	116

## Table of Contents

New object-specific render attributes in the Attribute Editor . .	117
New Quality Presets . . . . .	118
New Caustics and Global Illumination settings . . . . .	119
New Anti-aliasing Quality settings . . . . .	119
New Memory and Performance option . . . . .	120
New Translation Performance options . . . . .	121
New Translation Customization options . . . . .	121
New Preview settings . . . . .	122
New Shadows settings . . . . .	123
New Motion Blur Calculation settings . . . . .	124
New Final Gather settings. . . . .	125
New Diagnostics option . . . . .	126
New Displacement Overrides setting. . . . .	126
New Custom Entities options . . . . .	127
New support for region rendering . . . . .	127
New support for Area Lights. . . . .	128
Improved on demand translation. . . . .	128
Improved Custom Scene Text . . . . .	129
Extension to mental ray shading language. . . . .	129
New file texture filter types. . . . .	129
New Framebuffer Settings . . . . .	130
New Framebuffer options . . . . .	130
Improved preview rendering . . . . .	130
New attribute for Texture Bake-sets. . . . .	130
New attribute for Use Background shader . . . . .	131
New image output formats. . . . .	131
IPR (Interactive Photorealistic Rendering) improvements . . . . .	131

## Table of Contents

<b>Dynamics</b> .....	<b>132</b>
Support for renaming per particle fields .....	132
Specifying which particle systems to cache .....	132
<b>Painting</b> .....	<b>132</b>
Artisan improvements .....	132
3D Paint Tool improvements .....	132
<b>Maya Unlimited</b> .....	<b>133</b>
Fluid Effects improvements .....	133
outFoam attribute .....	133
New Solver Quality attribute .....	133
Fur improvements .....	133
Fur equalizer maps improvements .....	133
Hair improvements .....	135
Cloth improvements .....	136
Visualizing Cloth collision object attributes .....	137
Painting Cloth collision attributes .....	138
Debugging Cloth collisions .....	140
Editing Cloth constraints membership .....	142
Using expressions with Cloth .....	142
<b>Maya Web browser</b> .....	<b>144</b>
Maya browser can get data from Maya .....	144
<b>Utilities</b> .....	<b>146</b>
FCheck improvements on Mac OS X .....	146
Additional utilities on Mac OS X .....	146
<b>Environment Variables</b> .....	<b>146</b>
Improvements to IMF .....	146

## Table of Contents

<b>MEL</b> .....	<b>147</b>
New MEL commands and flags .....	147
General .....	147
File Referencing and Namespaces .....	148
Modeling .....	148
Animation .....	149
Paint Effects .....	149
Cloth .....	150
PhotoShop .....	150
<b>API and Devkit</b> .....	<b>150</b>
API and Devkit improvements .....	150
Performance improvements .....	151
New classes and header files .....	151
Important changes to existing classes .....	153
Examples of new plug-in functionality .....	177
Other significant changes .....	178
<b>5 Performance tips</b> .....	<b>181</b>
Getting the most out of Maya .....	181
Best practices for performance .....	181
Interactive drawing .....	181
Hardware Shaders .....	183
Memory .....	185
Animation .....	185
Fur .....	186
Miscellaneous .....	186
Modeling .....	186
Dynamics .....	187
Cloth .....	187
Rendering .....	187

## Table of Contents

Artisan .....	187
Paint Effects .....	187
Optimize your scene for playback .....	188
Display and UI Elements .....	188
Modeling .....	191
Skeletons .....	192
Skinning .....	192
Deformers .....	193
Animation .....	194
<b>Index .....</b>	<b>195</b>

# 1

# About Maya 6.5

## What's new and improved in Maya 6.5

The main focus of Maya 6.5 is the ability to handle larger and more complex scenes than ever before. We've approached this from two directions: performance enhancements throughout the software, and new and improved tools for breaking your scene into manageable pieces through file referencing. As well, there are new features in this release. For details, see:

- Chapter 2, "File referencing"
- Chapter 3, "Performance improvements"
- Chapter 4, "Other improvements"

We've also provided a list of some best practices to get the most out of using Maya. See Chapter 5, "Performance tips."

All features are common to all supported platforms unless otherwise indicated. Some features—Fluid Effects, Cloth, Fur, Hair, and Live—are only found in Maya Unlimited.

---

**Note**

You can also find all this information in the Maya Help. Please check the information in the Maya Help as it may be more up to date.

---

## **1 | About Maya 6.5**

### **> What's new and improved in Maya 6.5**

# 2

# File referencing

## About file referencing improvements

File referencing lets you assemble multiple objects, shading materials, animation and so on, into a scene without importing the files into the scene. This lets you segment digital assets in your scenes to better coordinate production on complex scenes and work more collaboratively.

This release includes architecture enhancements to file referencing as well as many new features such as *proxy references*. Proxy references let you substitute one or more file references by creating a set of substitute references, known as *proxies*.

In this chapter you'll find descriptions of these new features and improvements:

- “General file referencing improvements” on this page
- “Proxy references” on page 27

### Related topics

- ❖ “*Managing complex scenes*” in the Files and Organization chapter of the *Basics* guide.
- ❖ “*About file referencing*” in the Files and Organization chapter of the *Basics* guide.

## General file referencing improvements

### File reference edits

In Maya 6.0 and earlier versions, the representation of edits to file references had a number of limitations.

## 2 | File referencing

### > General file referencing improvements

- The user workflows that supported multi-level references were limited. Edits to references could be lost depending on the unload/load order that the user performed, as well as the unload/load state of the files lower in the referencing hierarchy. As a result, nodes might have appeared ungrouped or disconnected after an unload/load operation on a lower level file reference. (In most instances, this was remedied by unloading everything and reloading the reference from the top down to restore the edits.)
- Edits were not maintained between reference files when you switched between references using the Replace Reference command. This resulted in the loss of data when switching between different references.

We have restructured the representation of edits for file referencing to address these limitations. The following changes were made for this release:

- “File reference node architecture” on this page.
- “Selective preload of file references” on page 17
- “Maya API” on page 17

These architecture changes are described in more detail below.

### File reference node architecture

The representation of edits that get stored on the reference node of the parent scene has been modified for this release of Maya.

A reference node is created in the parent scene file when a file is initially referenced. The reference node stores any modifications that occur in the parent scene to the referenced data.

Previously, when a referenced file was unloaded, edits that were made to the referenced file (for example, modified attribute values, connections, and so on) were stored in a number of attributes (comprised of arrays of strings) within the reference node.

## 2 | File referencing

### > General file referencing improvements

Now the parent scene's reference node stores the edits with a single complex data attribute, and the edit strings that previously existed are no longer available.

### Selective preload of file references

To improve the performance of selective loading of reference files, Maya now saves references more consistently compared to previous versions. Any edits that have been applied to a reference file will be stored in the parent scene's reference node.

You will experience improved performance when performing a selective load of a referenced file, since unnecessary reference files (which previously required their existing edits to be resolved and then applied to their related references) are no longer pre-loaded.

### Maya API

MFileIO now lets you create, load, and unload references, and provides a number of export options that were previously unavailable.

### File referencing MEL update

The `file` and `namespace` commands have been extended to add support for managing namespaces. The `-cleanReference` flag for the `file` command has been extended. The `reference` command now has a new `-editCommand` flag which lets you query the edits on an unloaded reference node. The `reference` command has been extended to allow the query of a reference file associated with a reference node. For more information, see "New MEL commands and flags" in Chapter 4.

## 2 | File referencing

### > General file referencing improvements

#### Grouping file references

We have improved how Maya handles the grouping of file references. Previously, when a file reference was grouped and the referenced file itself was changed or modified in some way, the grouping was not maintained. Grouping is now maintained in all cases.

Grouping is also maintained between proxy references that reside within the same proxy set.

#### Multilevel file references

We have improved how Maya handles the grouping of multilevel file references, also referred to as nested references. Previously, Maya did not recognize the grouping that occurred when multilevel (nested) file references were used. This is now maintained in all cases.

#### Create Reference options

##### Lock option

A new Lock option has been added to the Create Reference options window. The Lock option lets you lock all of the nodes and attributes for a file when that file is first referenced into the scene. You can also lock an existing reference using the Reference Editor.

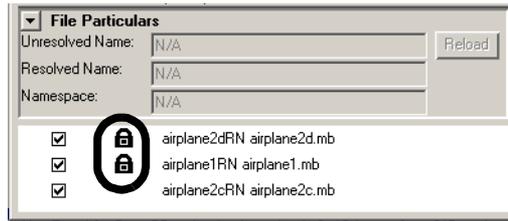


Locking file references is useful for collaborative work environments when files need to have their nodes and attributes locked to avoid accidental modifications by the users that reference the files.

In the Reference Editor, a lock icon appears beside the listed reference whenever a file reference is locked.

## 2 | File referencing

### > General file referencing improvements



If you reference a file without turning on the Lock option, any pre-existing locked attributes in a referenced file remain unchanged. That is, if the file being referenced has locked attributes when it gets referenced, they remain locked.

You can load and unload a locked reference file and the locked attributes will remain unchanged. If you need to modify any nodes or attributes on a locked reference file you can unlock and later re-lock the reference file from within the Reference Editor.

**Note** If edits are applied to an unlocked reference file at the parent scene level, those edits are not locked when the file reference is locked once again because the edits exist only in the parent scene. If you want the edits to be saved to the reference file so they can be locked, see "File > Save Reference Edits" on page 25.

#### To lock or unlock an existing file reference

- 1 In the Reference Editor, select the file reference you want to lock or unlock by clicking on its name.
- 2 In the Reference Editor, do one of the following
  - Select Reference > Lock Reference
  - Select Reference > Unlock Reference.

## 2 | File referencing

### > General file referencing improvements

With the exception of a few specific attributes and node types, all nodes and attributes for a file reference are locked by default. The attributes and node types that are excluded by the Lock option are specified by two MEL procedures. Each procedure outputs a string array that is used by the Lock option to specify the attributes to be excluded. The first procedure specifies which individual attributes to exclude. The second procedure specifies which attributes, by node type, to exclude. That is, the entire set of attributes for a given node type are excluded from locking by simply declaring the node type in the array.

Regardless of which attributes are excluded, the Lock option always locks all of the nodes for the referenced file.

You can override the MEL procedures that ship with Maya by creating your own customized MEL procedures. Your MEL procedures must be named exactly the same as the ones that ship with Maya. You must additionally ensure the files are placed in the Maya script path.

For more information on working with MEL procedures in Maya, see the *MEL and Expressions* guide.

The following examples describe the MEL procedures for excluding attributes and attributes by node type that were shipped with Maya. They can be used as the basis for creating your own customized MEL procedures.

---

**Note**

While these procedures are fully customizable through MEL, users working in a team production environment should be cautious when changing these procedures in mid-production as the change could inadvertently affect the overall pipeline and the data.

---

## 2 | File referencing

### > General file referencing improvements

#### Example 1: MEL procedure for excluding attributes

This example shows you how to create your own MEL procedure to specify which attributes get excluded by the file referencing lock operation. The MEL procedure provided in this example is the same procedure that is distributed with Maya.

- 1 Create a MEL file using a text editor, and name the MEL file: *getLockReferenceExcludedAttributes.mel*.
- 2 Add the following text to the file, ensuring you follow the exact syntax:

```
global proc string[] getLockReferenceExcludedAttributes()
{
// Return a string array containing a list of attributes
// to be skipped during locking of a referenced file. The
// listed attributes locked state will remain the same as
// in the referenced file.

string $lockReferenceExcludedAttributes[];

$lockReferenceExcludedAttributes[0] = "visibility";

return $lockReferenceExcludedAttributes;
}
```

**Note** Add or remove any attributes you want excluded by the locking operation by editing this text as necessary. That is, add or remove any `$lockReferenceExcludedAttributes[n]` lines as necessary using the same syntax as in the above example, also ensuring the array indices `[n]` are consecutive.

- 3 Save the MEL file and place it in your Maya script path.

## 2 | File referencing

### > General file referencing improvements

If you named the MEL file exactly the same as the default one that is distributed with Maya, the Lock option uses this new customized script for locking file references the next time Maya starts.

For more information on sourcing MEL scripts and working with MEL procedures in general, see the *MEL and Expressions* guide.

#### Example 2: MEL procedure for excluding node types

This example shows you how to create your own MEL procedure to specify which attributes, by node type, get excluded by the file referencing lock operation. The MEL procedure provided in this example is the same procedure that is distributed with Maya.

- 1 Create your MEL file using your favorite text editor, and name the MEL file: *getLockReferenceExcludedNodeTypes.mel*.
- 2 Add the following text to the file, ensuring you follow the exact syntax:

```
global proc string[] getLockReferenceExcludedNodeTypes ()
{
// Return a string array containing a list of node types
// whose attributes should be skipped during locking of a
// referenced file.

string $lockReferenceExcludedNodeTypes[];

$lockReferenceExcludedNodeTypes[0] = "lightLinker";
$lockReferenceExcludedNodeTypes[1] =
"displayLayerManager";
$lockReferenceExcludedNodeTypes[2] = "displayLayer";
$lockReferenceExcludedNodeTypes[3] =
"renderLayerManager";
$lockReferenceExcludedNodeTypes[4] = "renderLayer";

return $lockReferenceExcludedNodeTypes;
}
```

## 2 | File referencing

### > General file referencing improvements

#### Note

Add or remove any node types you want excluded by the locking operation by editing this text as necessary. That is, add or remove any `$lockReferenceExcludedNodeTypes [n]` lines as necessary using the same syntax as in the above example, also ensuring the array indices [n] are consecutive.

- 3 Save the MEL file and place it in your Maya script path.

If you named the MEL file the same as the default one that is shipped with Maya, the Lock option uses this new customized script for locking file references the next time Maya starts.

For more information on sourcing MEL scripts and working with MEL procedures in general, see the *MEL and Expressions* guide.

#### Proxy Tag Options

For more information, see "Proxy Tag options" on page 29

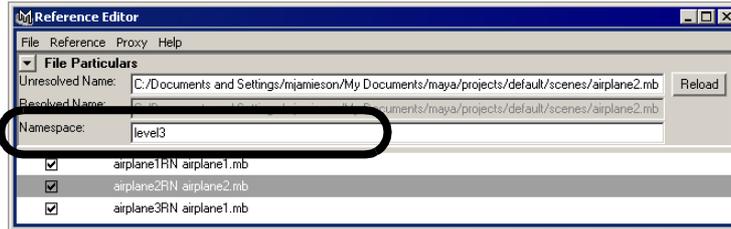
#### Reference Editor

##### Namespace editing

You can now edit the namespace for an existing file reference. Using namespaces ensures that each node gets assigned a unique name when referenced into a scene and avoids conflicts when multiple files are referenced. The new Namespace editing feature appears in the upper section of the Reference Editor.

## 2 | File referencing

### > General file referencing improvements



Editing a file reference's namespace is useful if you change your mind after the file reference is created, or inadvertently experience a namespace name clash. Editing the namespace is also useful if your project's production pipeline naming conventions change in the middle of your production work.

#### To edit the namespace for a file reference

- 1 In the Reference Editor, select the file reference whose namespace you want to edit.  
The current namespace for the selected file reference appears in the Namespace field.
- 2 Double-click within the Namespace field to select the text for the current namespace.
- 3 Type the text you want for the namespace for the file reference and click Enter.

#### Notes

- The file reference must have been created using the namespaces option for its namespace to be editable. That is, you cannot edit references that were created using the older file renaming prefix convention.
- A namespace cannot be edited if the file reference is unloaded.
- The new namespace you enter cannot already exist.

## 2 | File referencing

### > General file referencing improvements

- The namespace for a reference can only be edited for references that are made from the currently open scene. That is, you cannot edit namespaces for references that are nested within other file references.

#### File > Save Reference Edits

You can now save edits made within the parent scene to the referenced file using *Save Reference Edits*. Because edits made to a file reference reside within a reference node in the parent scene that references the file, Save Reference Edits transfers the edits that were made for a referenced file from the parent scene and applies them directly to the referenced file. As a result, the edits no longer reside in the parent scene.

Saving the edits to the referenced file is useful if you want the edits to permanently exist in the referenced file. In this way, the edits are propagated to all users of the reference. For example, if a tree were referenced into a parent scene at the wrong scale, it could be correctly scaled within the parent scene, and then have its scale attributes permanently saved back to the reference file.

#### To save edits directly to a file reference

- 1 In the Reference Editor, select the file reference to which you want the reference edits to be saved.
- 2 Select File > Save Reference Edits.

The edits for the selected file reference are transferred from the parent scene file and applied to the referenced file.

#### Notes

- Edits that are applied in the parent scene, specifically `setAttr`, `deleteAttr`, and `addAttr`, as well as commands that create and remove connections within the same referenced file, specifically `connectAttr` and `disconnectAttr`, are saved as part of the Save Reference Edits operation. Nodes and connections from the parent scene that affect only nodes from the referenced file will be saved

## 2 | File referencing

### > General file referencing improvements

to the reference file. For example, animation curves, construction history, and textures that affect only the reference file will be saved into the reference file. DAG nodes such as shapes from the main scene will never be saved to the reference file, with the exception of new construction history shapes that are parented beneath transforms from the reference file.

- Only edits to references that exist one level below the parent scene can be saved. If the reference is nested deeper in the referencing hierarchy (that is, the file reference is a grandchild of the parent scene) the edits cannot be saved. Additionally, if the reference contains nested references the edits will not be saved.
- Saving reference edits is not possible if the file reference is unloaded. You must first load the file reference before saving the edits.
- Just like any file input/output operation, you cannot undo the Save Reference Edits command.
- Nodes that are automatically created for a scene (such as default cameras and default shaders) are written to the child file during the Save Reference Edits operation.
- If you require new DAG nodes to be written out as part of the save reference edits operation, you can import the referenced file so all of the items reside in the scene, and then select only those imported items as well as any new nodes, and export the selection as a reference again. In this way, all of the edits to the nodes and attributes will get written to the exported file reference.

Importing and then exporting a reference has the potential to introduce namespace conflicts for other scenes that subsequently reference the exported reference. It is recommended that you remove any recently imported nodes from their old namespace prior to exporting a reference. For more information, see *“Editing a namespace”* in the *Basics* guide.

## Edit Menu

The Edit Menu has been removed from the Reference Editor. The Remove Reference menu item has been relocated to the Reference Menu. The Select File Contents menu item has been relocated to the File Menu.

## Preload Reference Editor

The Preload Reference Editor includes features to support the selective loading of proxy references introduced in this release. The Preload Reference Editor lets you choose which file references and proxy references to load or unload before you open a scene containing file references. You can select whether you want the proxy to be loaded or unloaded when the scene is opened, or simply select the proxy to be active so that it is set to load whenever you load or unload the reference after the scene is opened.

For more information on selectively preloading a proxy reference, see “Working with proxy references” on page 31.

## Proxy references

Proxy References lets you substitute one or more file references by creating a set of possible substitute references, known as *proxies*, for a given file reference. Proxy references are files that you create to visually or spatially stand in for an existing file reference. In most cases, proxy references are used to temporarily simplify complex scenes by substituting simpler versions of the objects into the scene. This allows production work to proceed without the overhead of the complex components of the original scene. When a scene is simplified in this manner, Maya’s performance improves.

Proxy references reduce the visual clutter in a scene by allowing you to focus on the elements you need to work on while still maintaining a spatial context of the scene’s elements.

## 2 | File referencing

### > Proxy references

For example, a user may wish to substitute a scene that contains primitive objects that exist in the same position, size, and scale as a similar scene that contains more complex and detailed versions of furniture for an office. By substituting a simple primitive-based proxy version of the scene, the user can obtain better interactive performance when dollying, tumbling, or playing back animation. Because, the primitive objects also represent where the more detailed versions of the furniture elements exist spatially in the original file reference, the user can more easily concentrate on animating their character or camera.

To create a proxy reference for a scene you must first create a scene that contains the proxy reference data. This could involve creating a scene that generally matches the contents of a pre-existing scene but contains simplified versions of your characters, props, and assets. The items in this scene would be set up and positioned exactly the same way as your more complex version of the scene, so that a spatial correlation is created between the *high* and *low* detail versions of the scenes. Once the high detail version of the scene is referenced, the low detail version can be added as its proxy reference using the Reference Editor.

A proxy reference can only be created for an existing file reference. That is, a proxy cannot exist by itself.

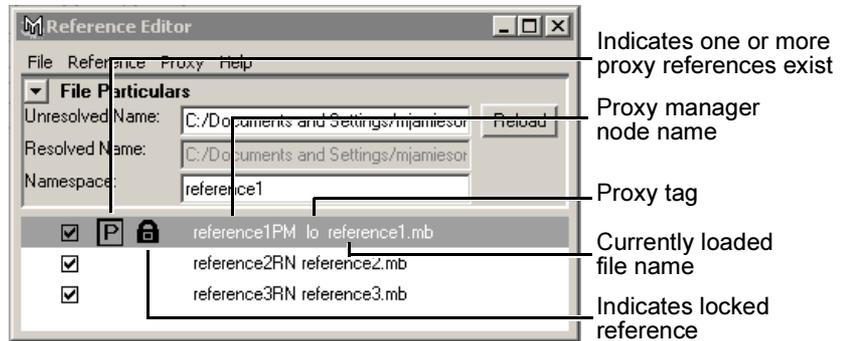
When a proxy reference is first created a *proxy set* is also created. When the proxy set is created, the original file reference as well as the selected proxy reference are organized under a new proxy manager node. The user can then select which file (the original reference or the proxy reference) they want to load in the scene using the Reference Editor or by right-clicking an object in the scene and making the appropriate selection from the marking menu.

When a proxy reference is added to an existing file reference, the original reference becomes a proxy reference within the new proxy set. In the Reference Editor, a P icon is displayed to indicate the

## 2 | File referencing

### > Proxy references

existence of one or more proxies for the reference, and the name displayed in the Reference Editor reflects the proxy manager that is created.



Any edits that occur to a proxy reference are independent from the other proxy references in the same set except when the original reference is created with grouping turned on. In that case, all subsequent proxy references that get added to the file reference are in the same group. In this way, you can use the group to transform the proxy within the scene, reload a different proxy, and have the transformation maintained.

When the second last proxy reference is removed from the proxy set and only the original file reference remains, Maya collapses the proxy set. The file reference gets updated in the Reference Editor to reflect this change.

You can differentiate between proxy references for a given file reference by labeling the proxies so they appear with a unique tag in the Reference Editor.

### Proxy Tag options

Proxy Tag options have been added to the Create Reference options window as part of the new proxy reference feature. (The Proxy Tag options also appear in the Add Proxy options window in the Reference Editor.)

## 2 | File referencing

### > Proxy references

Proxy Tags are unique names you can assign to both file and proxy references to more easily manage those references from within the Reference Editor. The Proxy Tag option lets you specify the label/tag you want applied to a file reference or proxy reference when it is first created. The proxy tag appears in the list of proxy references within the Reference Editor. Proxy tags let you globally load, unload, or switch between proxies.



Maya keeps track of, and can distinguish between, the last proxy tag used for a file reference, and the last proxy tag used for a proxy reference. This ability streamlines the tagging process regardless of your preferred workflow.

For example, you may want to tag multiple file references in succession with a tag named *hiRes* when you first create each one. In this case, you need only type the tag name once and it is automatically assigned to successive file reference tags. If you want to tag multiple proxies for those same references, you only need type in the proxy tag name for the proxy once, and the proxy tag will be remembered for successive proxies.

Alternatively, you may want to create and tag one file reference named *hiRes* and then immediately create and tag its corresponding proxy reference named *loRes*. You can then create the next file reference and it will automatically be assigned the tag *hiRes*, then create its proxy reference, and it will automatically be assigned the tag *loRes*. Because Maya can distinguish between the most recent file reference and proxy reference tags specified, this alternating tagging workflow is possible. For more information see "Proxy Options" on page 42.

Proxy Tags can be used for many other proxy situations; for example, *modelFur* vs. *modelNoFur*, or *SceneDynamics* vs. *SceneNoDynamics*, and so on.

## Notes

- If a file reference has not been assigned a unique proxy tag prior to the creation of the first proxy in the scene, the file reference will be assigned a proxy tag named *original* to differentiate the original file reference from the first proxy. Once a tag has been specified for a file reference, it will continue to be used as the default file reference tag until another is specified. That is, Maya only uses the default *original* tag if the user has not previously explicitly specified a tag for a file reference.
- If a proxy tag is not specified when the first proxy reference is created in the scene, Maya will automatically apply a unique proxy tag based on the name of the reference node. Once a proxy tag has been specified for a proxy reference, it will continue to be the default tag for proxy references until another is specified. That is, Maya only uses a default proxy tag name when the user has not previously specified an explicit tag name for a proxy reference.

To label a proxy reference, see “To add a file as a proxy reference” on page 33.

## Working with proxy references

Proxy references let you substitute different files in the scene that have been previously organized within a proxy set. You must have an existing file reference in order to add one or more corresponding proxy references to the proxy set.

### Example: Creating proxies for a file reference

A generalized workflow for creating proxies for a file reference is as follows:

- 1** Create the files you want to use as proxy references.

## 2 | File referencing

### > Proxy references

For example, if you have a full resolution version of your scene, you can create simpler, lower resolution copies of this scene. That is, the other versions contain simpler versions of the objects comprise the scene but have the same names etc. As a result, you might have three versions of the scene: a high resolution version called *Scene\_HiRes.ma*, a medium resolution version called *Scene\_MedRes.ma*, and a low resolution version *Scene\_LowRes.ma*.

- 2 Create a file reference for the full resolution version of the scene, and give it a proxy tag name of *HiRes*.
- 3 In the Reference Editor, select the file reference you just created and then add one of the other files as a proxy reference using the Reference Editor's Proxy menu.

In this example, when you add the file *Scene\_MedRes.ma* as a proxy, ensure you apply a proxy tag named *MedRes*.

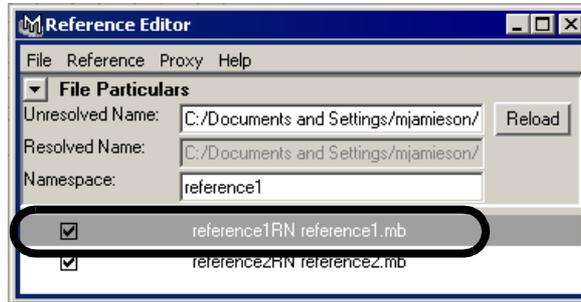
- 4 Repeat step 3 for the *Scene\_LowRes.ma* file to add it to the proxy set ensuring you apply the appropriate proxy tag name (*LowRes*).
- 5 You can load or unload multiple proxy references in the scene simultaneously by selecting the file references within the Reference Editor, and then selecting the appropriate proxy tag name (provided you have been consistent with your proxy tag naming).

For more information on specific steps that support the above workflow, please refer to the following:

- "To add a file as a proxy reference" on page 33
- "To load or switch a proxy reference" on page 35
- "To remove a proxy reference" on page 38
- "To reload multiple proxy references" on page 39
- "To selectively preload a proxy reference" on page 40

**To add a file as a proxy reference**

- 1 In the Reference Editor, select the file reference to which you want to add the proxy reference.



- 2 In the Reference Editor, select Proxy > Add Proxy > .  
The Proxy Options window appears.
- 3 In the Proxy Options window, set the Proxy Tag Options by either typing a text string that you want as the label/tag for the proxy reference, or choose an existing tag from the text string drop-down menu.



## 2 | File referencing

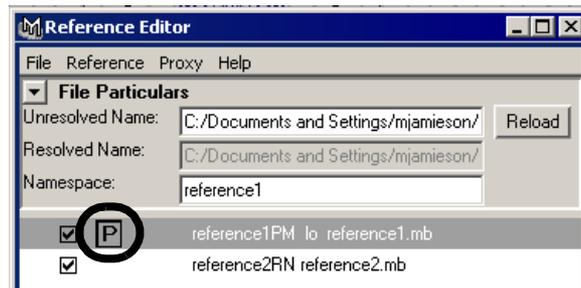
### > Proxy references

- 4 When you have finished setting the Proxy Tag options click the Proxy button.

The file browser appears showing the current default project scene directory (or the last directory that was accessed by the file browser when referencing a file).

- 5 In the file browser, navigate to the file you want to specify as the proxy reference and select it.
- 6 Click Proxy to add the selected file as a proxy reference to the existing file reference.

The proxy file is added to the list of available proxies within the proxy set.



In the Reference Editor, the selected file reference's name updates to display a P icon immediately to the right of the load/unload check box. The P indicates that one or more proxy references exists for the file reference.

When Proxy > Add Proxy is first used with an existing file reference, a proxy set is created for that file reference. The proxy set contains the original file reference as well as the new proxy reference. That is, all of the file references are members of the proxy set. The proxy set implicitly defines the various proxy files for that file reference.

For more information on proxy tag options see "Proxy Tag options" on page 29.

**Note** When you add a proxy reference to an existing file reference, it is not immediately loaded into the scene. For more information, see the next section.

### **To load or switch a proxy reference**

- 1** In the Reference Editor, select the file reference that corresponds to the proxy reference you want to load into the scene.
- 2** Select Proxy > Reload Proxy As.  
A drop-down menu appears.
- 3** Select the desired proxy file from the list of proxy references.  
Maya immediately loads the selected proxy reference.

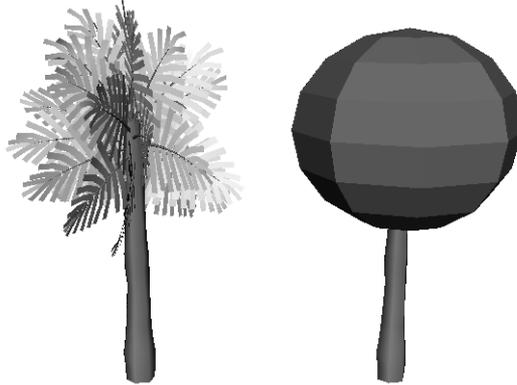
**Note** When a proxy reference is selected for reload, Maya first checks to see if the current proxy is loaded. If it is loaded, Maya unloads it before automatically loading the selected proxy reference. It is not possible to switch between unloaded proxy references, and have the reference remain unloaded. When a proxy tag appears gray in the list, it indicates that it is the currently loaded proxy reference.

### **Example 1: Adding and loading a proxy reference**

This example shows you how to add a proxy reference using the Reference Editor, identify it with a unique tag, and then load it into the scene.

## 2 | File referencing

### > Proxy references



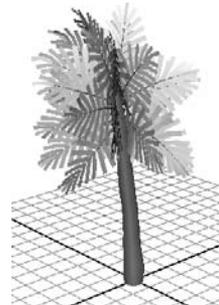
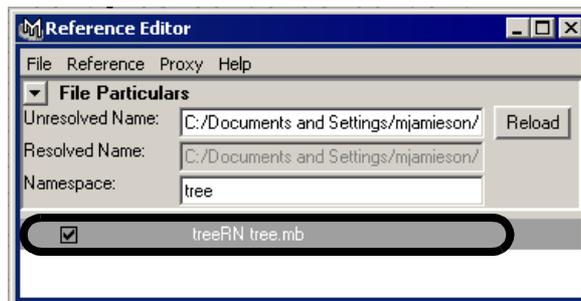
File reference of tree model  
1,800 polygon faces

Proxy reference  
170 polygon faces

A scene contains a file reference to a scene containing a tree model. The referenced tree will be substituted with a simpler proxy reference.

- 1 In the Reference Editor, from the list of file references, select the existing file reference for the detailed model of the tree.

In this example, the file reference is named *tree*.



- 2 In the Reference Editor, select Proxy > Add Proxy > .

## 2 | File referencing > Proxy references



The Proxy Options window appears.

- 3 In the Proxy Options window, set the Proxy Tag Options by typing `loRes` in the text string field.



- 4 Click the Proxy button.

The file browser appears showing the current default project scene directory (or the last directory that was accessed by the file browser when referencing a file).

- 5 In the file browser, select the *treeSimple* scene you want as the proxy reference.
- 6 Click Proxy to add the selected file as a proxy reference to the existing file reference.

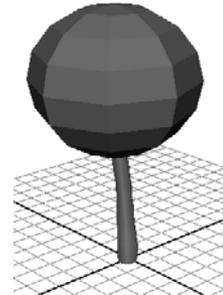
The proxy file is added to the list of available proxies within the proxy set. This is indicated by the P icon in the Reference Editor. The proxy file does not appear within the scene until you load it.

## 2 | File referencing

### > Proxy references



- 7 In the Reference Editor, with the existing file reference still highlighted, select Proxy > Reload Proxy As.
- 8 Select *loRes* from the pop-up menu that appears.



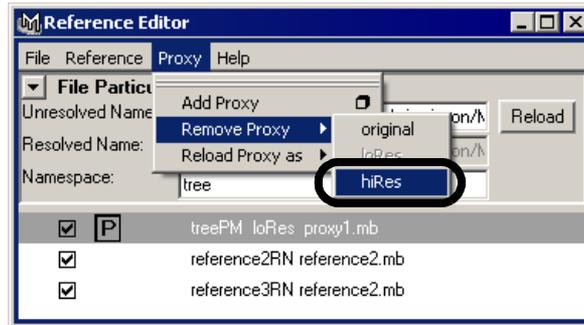
Maya updates the file reference by loading the *treeSimple* proxy reference into the scene.

### To remove a proxy reference

- 1 In the Reference Editor, select the existing file reference that contains the proxy reference you want to remove.
- 2 Select Proxy > Remove Proxy and select the proxy file from the list of proxy references that appear in the pop-up menu.

## 2 | File referencing

### > Proxy references



The proxy file is removed from the list of available proxies within the proxy set.

When a proxy tag appears in gray within the list of available proxies, it indicates that it is currently loaded. It is not possible to remove a loaded proxy. To remove the currently loaded proxy you must first reload any other proxy reference from the list so that the proxy you want to remove becomes available within the list.

When only a single reference remains in the proxy set, Maya automatically removes the proxy set for the file reference. The P icon beside the listed reference is also removed from the Reference Editor.

#### To reload multiple proxy references

- 1 In the Reference Editor, select the file references containing the proxy references you want to reload.
- 2 Select Proxy > Reload Proxy As.  
A drop-down menu appears.
- 3 Select the desired proxy tag from the list of proxy tags.  
The selected file references update by loading the proxy references associated with the proxy tag you selected.

## 2 | File referencing

### > Proxy references

#### Note

The list of proxy tags that appear is generated from all of the proxy tags currently in use for the proxy sets selected.

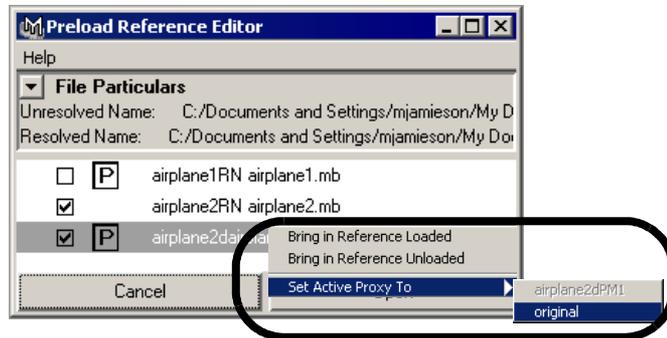
A proxy for a specific proxy set is only reloaded when the selected proxy tag matches one of the available tags for that set. For example, if multiple proxy sets are selected and the *high* proxy tag is chosen for reload, only those proxy references that have the *high* tag will be reloaded. Any proxy set that does not include the *high* tag will remain unchanged.

#### To selectively preload a proxy reference

- 1 Select File > Open Scene > .  
The Open Options window appears.
- 2 In the Referencing options section, turn on Selective Preload, then click Open.  
The file browser appears, listing the default scene directory for your project.
- 3 Select the file that contains the file and proxy references you want to selectively preload, then click Open.  
The Preload Reference Editor appears. The Preload Reference Editor lists all available file references for the scene. Depending on how the Referencing options were set, some file references will appear loaded or unloaded in the Preload Reference Editor. When a P icon appears beside a particular file reference it indicates that one or more proxy references exist for that reference.
- 4 Right-click the file reference that contains the proxy reference you want to have loaded when the scene is opened so that the Preload Reference Editor pop-up menu appears.

## 2 | File referencing > Proxy references

- 5 From the pop-up menu, choose Set Active Proxy To and select the desired proxy reference from the list of proxy tags in the drop-down list.



The selected file reference name updates in the Preload Reference Editor to indicate that the proxy reference is set to be loaded.

- 6 So the reference loads, turn on the checkmark for the file reference. (You can set any other file references at this time as desired.)
- 7 Click Open.  
The scene opens as per the Preload Reference settings.

### Note

It is possible to set the load or active state for multiple proxies simultaneously when using the Preload Reference Editor. The list of proxy tags that appear in the drop-down list when multiple file references are selected is generated from all of the proxy tags currently in use for the proxy sets associated with the selected file references.

## 2 | File referencing

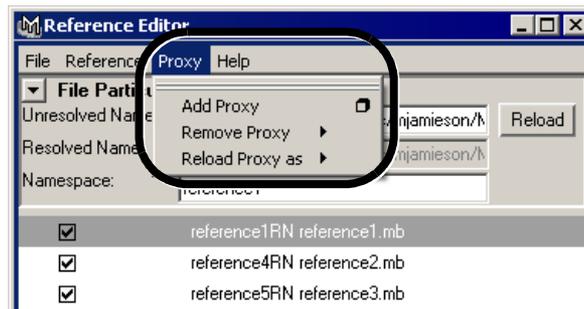
### > Proxy references

## Proxy Options

The following updates and options are new to the Reference Editor in support of the new proxy reference features.

**Note** The Proxy Options menu items are also available from the context-sensitive menu in the Reference Editor. To display this menu, select a file reference and right-click the item.

## Proxy Menu



### Add Proxy

#### Add Proxy >

Adds a proxy reference to the currently selected file reference. When you select Proxy > Add Proxy >  you can specify the file type to add and set the Proxy Tag Options. See Proxy Options window below.

If a proxy reference does not exist for the file reference, a proxy set is created for the proxy references for that file reference. If a proxy reference exists, the new proxy becomes a member of the existing proxy set for that reference.

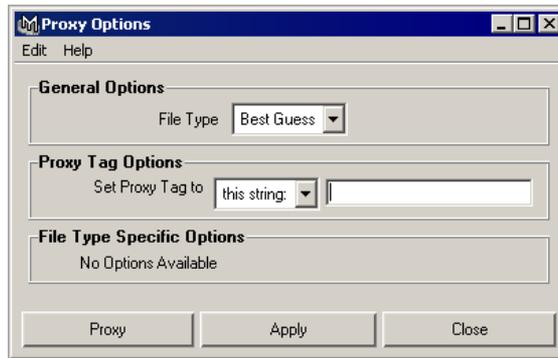
## 2 | File referencing

### > Proxy references

When a proxy reference is created, the listed file reference updates to display a P icon to indicate that one or more proxy references exist for that reference.

For information on how proxy references are tagged, see “Proxy Tag Options” on page 43.

The Proxy Options (□) window is used to set the following options:



**General Options** Select the file format you want to use as a default for the next time you add a proxy reference. If you have a project set up, when you open a scene, the browser points to the directory containing files of that type. On Windows and Mac OS X, it also sets the filter to display only files of the selected type.

**Proxy Tag Options** For more details about the Proxy Tag Options see “Proxy Tag options” on page 29.

## 2 | File referencing

### > Proxy references

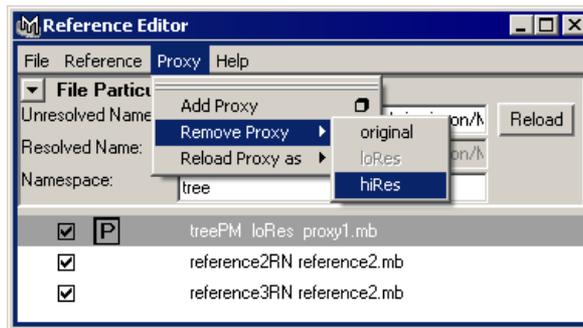
#### File Type Specific Options

Depending on the File Type you select, various File Type Specific Options are displayed here. See File > Open Scene in the *Basics* guide for more information.

#### Remove Proxy

Deletes the proxy reference that is selected from the submenu. The proxy reference is deleted from the proxy set.

The list of available proxy references is displayed in the menu based on their proxy tags. When a proxy tag appears in gray, it indicates that it is currently loaded.



When a proxy reference is removed from a proxy set, and only the original file reference remains, the proxy set is removed and the reference returns to a normal reference state. In the Reference Editor, the P icon is removed from the listed file reference, and the listed name of the file reference is updated.

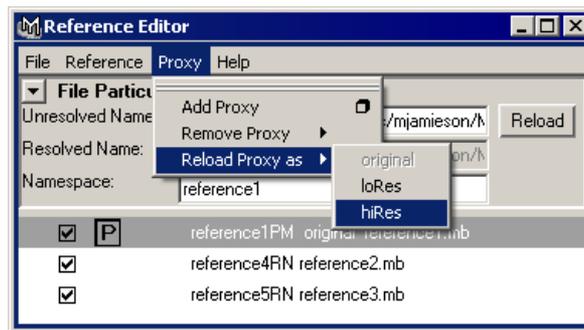
## 2 | File referencing

### > Proxy references

**Note** It is not possible to remove a loaded proxy. To remove the currently loaded proxy you must first reload any other proxy reference from the list so that the proxy you want to remove becomes available within the list for removal.

#### Reload Proxy As

Loads the proxy reference that is selected from the submenu into the scene.



Reload Proxy As is used to switch between proxy references for a given file reference. The list of available proxy references is displayed in the menu based on their proxy tags. When a proxy tag appears in gray, it indicates that it is currently loaded.

#### Related topics

- ❖ “File > Open Scene” in the *Basics* guide.
- ❖ “File > Reference Editor” in the *Basics* guide.

**2 | File referencing**  
**> Proxy references**

# 3

# Performance improvements

## About Maya performance improvements

We've made various performance improvements to Maya 6.5, which are all described in this chapter. In particular, we'd like to draw your attention to the following significant performance improvements, which in some cases are thousands of times faster:

- "Binary save over a network" on page 51
- "Import of polygon .obj files (Windows)" on page 51
- "Wrap deformer setup and playback" on page 52
- "Update for Sculpt deformations" on page 53
- "Soft Modification Tool" on page 53
- "Polygon editing with color per vertex" on page 54
- "UV Texture Editor" on page 54
- "Polygons > Combine" on page 55
- "Polygons > Mirror Geometry" on page 55
- "Polygons > Cleanup" on page 55
- "Edit Polygons > Subdivide" on page 55
- "Edit Polygons > Delete Vertex" on page 55
- "Edit Polygons > Delete Faces" on page 56
- "3D Paint Tool" on page 59
- "Fluids Solver" on page 59

Keep in mind the following about the performance improvements:

- The performance improvements detailed in this chapter were achieved on the scenes we tested. Since Maya scenes vary widely, your results may differ. For additional information on speeding up your work, please see "Getting the most out of Maya" on page 181.

## 3 | Performance improvements

### > General

- We've pushed the limit on the size and complexity of the scene you can work with.
- While you may notice small improvements in some areas, the sum of all the improvements may produce significant results.
- In previous releases of Maya you may have met challenges with some features and workflows, but in Maya 6.5 you may find the ease of working with Maya is significantly increased.

For details, see the following sections:

- "General" on page 48
- "Animation" on page 51
- "Character Setup and Deformers" on page 52
- "Modeling" on page 54
- "General rendering" on page 57
- "mental ray for Maya rendering" on page 57
- "Painting" on page 58
- "Fluid Effects" on page 59
- "Fur" on page 60
- "Hair" on page 60
- "Cloth" on page 60
- "Nodes and Attributes" on page 60
- "API" on page 61

## General

### Opening scenes with large polygonal data sets

We have made improvements in the loading of Maya ASCII format scenes comprised of large polygon data sets that contain UV information. These types of scenes now load faster.

## Shaded and wireframe large scenes display draw and refresh

We have made improvements to the display refresh and tumbling speed for shaded geometry, polygon wireframe objects, and subdivision surfaces wireframe objects. The speedup varies a great deal depending on the complexity of the scene. You will see the greatest improvement in large scenes (1000s of objects or large objects with more than 50,000 faces), which may be as much as three to four times as fast; scenes with only a few objects may be a small amount faster.

## Polygonal objects with per face shader assignments

Polygonal objects which have per face shader assignments refresh faster in the interactive scene view when in smooth shaded, or smooth shaded and textured display mode. You can also see the speed improvement when you use Isolate Select for polygonal object faces.

## Particle sprite display

We have made improvements to the display of particle sprites. The speedup can be very large for cases where a single texture has been applied to sprites. Smaller improvements have been made in cases where 2D scale or twists have been applied to sprites.

## Polygon color per vertex draw

Scenes with polygonal surfaces where color per vertex was used are much faster when drawing or refreshing.

## Hardware shader drawing

The performance of drawing hardware shaders on objects which do not deform has been improved.

## 3 | Performance improvements

### > General

#### NURBS shaded display

The speed of NURBS shaded display and tumbling has been improved, especially on Windows.

#### Subdivision surfaces components display

To improve the speed of working with subdivision surfaces in Standard mode, Maya now draws components (vertices or faces) as points by default to indicate finer levels of the base mesh. There is a user preference to switch back to the previous behavior (drawing components as numbers).

#### **To use the previous behavior (drawing components as numbers)**

- 1** Select Window > Settings/Preferences > Preferences.  
The Preferences window opens.
- 2** Select the Subdivs category from the list, and select Numbers beside Component display.

#### Scene view speedup: Use default material

In the Shading menu of the scene view, there is a new *Use default material* menu item.

This option applies the default shading material (Lambert, unless you have changed the default) to all objects in the scene. This is much faster than assigning shaders individually; as well, redrawing is faster. When this option is selected, Hardware Texturing and X-ray are disabled. As well, transparency and shadows are not supported in this mode.

## Binary save over a network

The speed of saving Maya binary files (.mb) over a network has been improved dramatically. You may notice some very large speedups. However, the performance of a networked file write is dependent on the structure of the file being saved; as well, the performance of the network influences the results.

## Import of polygon .obj files (Windows)

We have modified the way that polygon .obj files are read into Maya for Windows. You should notice a large speedup of file import operations involving polygon .obj files.

## Animation

### Skinned geometry playback

- Skinned geometry plays back faster when its unused influences are removed. You can now use the Remove Unused Influences option at the time of bind, or when optimizing your scene, to speed up the playback of your skinned geometry. See "Prevent attachment of zero-weighted smooth skin influences" on page 26.
- Skinned polygonal meshes now play back faster in shaded mode when their triangles are precomputed. See "Precomputation of triangles for faster playback" on page 52. High resolution skinned polygonal meshes now play back at faster rates in shaded mode in this version of Maya.
- Skinned geometry with user normals that are being deformed now plays back faster. However, playback of skinned geometry is still faster when you turn *off* the deformation of its user normals.

For tips on how to improve the overall playback of your scene, see "Getting the most out of Maya" on page 181.

### 3 | Performance improvements

#### > Character Setup and Deformers

## Skeleton draw

Simple and complex skeletons now draw at a faster rate.

## Precomputation of triangles for faster playback

You can now precompute the triangles for a shaded polymesh character using the Reuse Triangles attribute and improve playback performance.

Reuse Triangles precomputes the triangles for a shaded character, and then reuses the triangles during playback rather than recomputing them for every frame. Shaded characters comprised of quadrilateral polygons (quads) benefit the most when this option is turned on.

### To turn on Reuse Triangles for optimized playback

- 1 Select the polygon mesh you want precomputed for optimized playback.
- 2 In the Attribute Editor, select the *n*Shape tab for the poly mesh, where *n* is the name of the poly mesh.
- 3 In the Mesh Controls section, turn on Reuse Triangles.

## Character Setup and Deformers

### Wrap deformer setup and playback

It is now much faster to set up wrap deformers with Max Distance values other than 0. Also, the playback speed of wrap deformers with a Max Distance value of 0 (or with Use Max Distance turned off) has increased. To further improve the performance of wrap deformations, the default Max Distance value for wrap deformers has been changed to 1.

## Update for Sculpt deformations

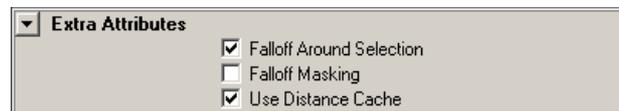
Sculpt deformations created by secondary NURBS objects now update faster. The speed at which the updates occur is determined by the size of the region (the smaller the region, the faster the update speed) and the number of control points (the smaller the number of control points, the faster the update speed) being deformed.

## Cluster deformations on polygonal models

Cluster deformations on large complex polygonal models (for example, models that have many UVs, colors, blind data and so on) now play back faster when in wireframe mode.

## Soft Modification Tool

Soft Modifications that use the Falloff Around Selection option now occur much faster, particularly on dense meshes when a significant number of components have been selected. In addition, a new Use Distance Cache option caches the influence distance information to improve interactive performance when modifying the falloff curve, falloff radius, or the position of the manipulator. The Use Distance Cache option will slightly increase the memory used by the softMod node and can be turned off from the Extra Attributes section of the softMod Attribute Editor.



## Smooth skin weights in the Component Editor

It is now significantly faster to change selections in the Component Editor when the Hide Zero Columns option is *on*. This makes editing smooth skin weights for complex characters in the Component Editor more efficient.

## **3 | Performance improvements**

### **> Modeling**

## **Modeling**

A number of performance improvements have been made to the polygon modeling tools for Maya 6.5, specifically when working with large polygon data sets. In particular, the speed in which Maya selects data for many of the tools has been improved.

### **Polygons Reduce**

Polygons > Reduce now works faster for all polygon meshes. Reduce automatically reduces the number of polygons in a selected mesh by a user-specified percentage.

### **Polygon editing with color per vertex**

The performance for polygon editing tools that modify the number of faces in a poly mesh that contains color per vertex information has been improved, for example, the Extrude Face command.

### **UV Texture Editor**

- Polygons > Rotate UVs works faster.
- Polygons > Scale UVs works faster
- Polygons > Cut UVs works faster.
- Polygons > Copy UVs works faster.
- Refresh Current UV values works faster.
- Select > Select Shell Border works faster.
- Sew UVs works much faster now when it's selected from the UV Texture Editor toolbar.

## **Polygons > Combine**

Polygons > Combine now works faster for polygon meshes that contain color per vertex information. In previous versions of Maya, polygon meshes that contained color per vertex information could take a long time to be combined into one mesh.

## **Polygons > Mirror Geometry**

Polygons > Mirror Geometry works much faster and is now implemented as a command with a single node. You can now change the mirror properties on the single node. Previously it was necessary to make edits on multiple nodes.

## **Polygons > Smooth Proxy**

Polygon meshes that contain UV information have improved interactive performance when working in Smooth Proxy mode.

## **Polygons > Cleanup**

Polygons > Cleanup works much faster when you pick faces.

## **Edit Polygons > Subdivide**

After picking faces, when you select Edit Polygons > Subdivide >  to open the Polygon Subdivide Faces Options box, it performs much faster when working with large data sets.

## **Edit Polygons > Merge Vertices**

Edit Polygons > Merge Vertices works faster.

## **Edit Polygons > Delete Vertex**

Edit Polygons > Delete Vertex works much faster.

### **3 | Performance improvements**

#### **> Modeling**

#### **Edit Polygons > Delete Faces**

Edit Polygons > Delete Faces works much faster.

#### **Edit Polygons > Separate**

Edit Polygons > Separate works faster, specifically when separating large polygon meshes that contain color per vertex information. It is now possible to separate large data sets with thousands of meshes output much faster, with a reduction in the memory required to do so.

#### **Edit Polygons > Colors**

The Remove operation in the Apply Color command removes color per vertex information faster on large polygon meshes.

#### **Edit Polygons > Normals**

We've improved the performance of several of the Edit Polygons > Normals submenu items: Average Normals, Set to Face, and Reverse. Dramatic improvements will be experienced mostly when working with large data sets.

#### **Subdivision surfaces display**

The display for subdivision surfaces is improved for wireframe, wire on shaded, and component display modes. Users will notice improved interactive performance (dolly, tumbling, and so on) when working with complex subdivision surfaces.

#### **Soft Modification Tool**

See "Soft Modification Tool" on page 53.

## General rendering

### New user preference for maximum hardware texture resolution

A new user preference, Max. Texture Display Res., is now available. Use this preference to control the maximum hardware texture resolution to improve performance when the available texture memory on your graphics card is small. The default value for the Max. Texture Display Res. is based on the maximum resolution supported by your graphics card (up to 4K). You can find this preference in the Display section of the Preferences window (Window > Settings/Preferences > Preferences). Reducing the value of the Max. Texture Display Res. improves performance, but reduces quality.

### Improved normal mapping

The time required to generate a normal map has been significantly reduced. The larger the normal map, the greater the performance improvement.

### Swatch updating in Hypershade

The time required to update material swatches in Hypershade has been significantly reduced in some cases, especially when resizing swatches.

## mental ray for Maya rendering

### Improved Object Instancing

The handling of scenes with large amounts of instances has been optimized, leading to improved performance.

### 3 | Performance improvements

#### > Painting

## Animation detection

To improve mental ray for Maya's animation detection, the new Optimize Animation Detection option is now available in the Render Global Settings window, mental ray tab, in the Translation > Performance section. This option is turned off by default.

When the Optimize Animation Detection option is turned on, the processing of non-animated geometry is significantly optimized because mental ray for Maya detects animated nodes prior to processing the scene. This is especially useful for scenes that contain many static objects and only a few simply animated objects.

**Note** Animation detection is limited in several ways, as it currently can only detect key frame animations, but not those generated by expressions and pre/post RenderMel scripts.

## Painting

### Artisan tools

- We've improved the performance of Artisan when painting and tumbling with color feedback on. Results vary depending on the size of the model. The larger the model (single skin), the better the performance change. This improves workflow efficiency when using any Artisan tool, especially with large data sets.
- We've made performance improvements to Artisan sculpting overall, and especially when flooding. This improves efficiency when using Artisan for sculpting, especially with large data sets.
- We've improved the performance of importing attribute maps on polygons using the Sculpt Polygons Tool.

## 3 | Performance improvements

### > Fluid Effects

- The Artisan Tool Settings editor now opens much faster. This improves workflow efficiency when using the Artisan or 3D Paint tools.

### 3D Paint Tool

- We've improved the performance of 3D Paint, especially with large and multiple textures, and have extended the maximum texture size to 4K. The 3D Paint Tool is now much easier to use with production quality/resolution textures.
- We've made improvements to the quality of the 3D Paint Tool's brush strokes. The strokes now have the appearance of an airbrush style and are smoother, especially where a stroke crosses itself. We've also changed the default setting of the tool's Stroke > Stamp Spacing from 0.1 to 0.05 and the default Flood Color from black to white.

## Fluid Effects

### Ocean shader playback

We've improved the playback speed of the Ocean Shader in shaded mode.

### Fluids Solver

Simulating fluids in the interactive view is now much faster. Rendering will also be faster if the fluid is not pre-cached. The speedup is most significant when working with large 3D grids using the linear solver.

## 3 | Performance improvements

### > Fur

## Fur

When using the new Rapid Scanline renderer for mental ray, you may find that if raytracing is turned off, Fur is rendered about twice as fast as the regular renderer. If raytracing is turned on, Fur is rendered more slowly, but at better quality than the regular renderer.

## Hair

We've noticeably sped up the creation and playback of hair caches for NURBS Curve output hair systems.

## Cloth

We've improved the performance of drawing Cloth constraints. This will be most noticeable when many Cloth vertices are constrained. This improves the interactive performance while working with Cloth. Results vary depending on the size of the model. The larger the model, the better the performance change. If a model is very small, the performance change may not be noticeable as the maximum frame rate is achieved.

## Nodes and Attributes

### PolyBlindData node

File operations for scenes containing poly meshes with blind data and operations within the Blind Data Editor are now improved in many cases as a result of changing the behavior of the `blindDataEntriesAreNew` attribute on the `polyBlindData` node.

In previous versions of Maya the default setting was `False`. Maya now automatically sets the value of the attribute to `True` or `False` whenever it is appropriate.

For example, if there is only one blind data node of a particular blind data type for a poly mesh, then the attribute will be set to True. If more than one blind data node exists for a particular blind data type, for a particular poly mesh, then the first occurrence of the node will be set to True. The other occurrences in the meshes history will have this node attribute set to False. Generally there will only be one node of this type for a given mesh unless the user has added to the construction history in between modifications to blind data.

## API

### Plug-in Manager updating

The Plug-in Manager now updates more quickly on subsequent opens. For example, opening a directory with hundreds of entries now takes a fraction of the time it did in previous versions of Maya. As a part of this change, the `MAYA_PLUG_IN_PATH` is scanned only once when you first open the window. As a result, if you create a new plug-in on the `MAYA_PLUG_IN_PATH` while Maya is running, it will not show up in the Plug-in Manager. To access newly created plug-ins, do one of the following:

- Click the Refresh button located in the Plug-in Manager window to update the list. This causes the directories to be scanned for changes to the window's contents.
- Use the `loadPlugin` MEL command.
- Restart Maya.

### MFnMesh::create

Creating the polygons of a large mesh using `MFnMesh::create()` is now faster.

## 3 | Performance improvements

### > API

#### MFnMesh acceleration intersection methods

We have added a number of new methods to the MFnMesh class to help you find intersections more quickly:

- `closestIntersection()`
- `anyIntersections`
- `allIntersections`

See “API and Devkit improvements” in the Other improvements chapter for the full list.

#### Iteration over several MFn types

The following classes now accept `MIteratorType` objects that allows iteration over multiple `MFn::Type` values:

- `MItDag`
- `MItDependencyGraph`
- `MItDependencyNodes`

Plug-ins that take advantage of this feature can scan the DAG or DG faster since multiple passes are not required.

# 4

# Other improvements

## What else is new and improved?

In this chapter you'll find new features and improvements made to Maya, which are not performance related.

General improvements:

- "Alias DirectConnect included with Maya" on page 65
- "FBX plug-in included with Maya" on page 66
- "CGFx plug-in (Windows)" on page 66
- "Subdivision surfaces heads-up display" on page 67
- "Improved Layer Editor navigation" on page 68
- "Maya Web browser improvements" on page 68
- "Improved wheel mouse support" on page 69
- "Scaling a light or camera locator" on page 69
- "Environment variable expansion in Maya file browser" on page 70
- "Create new layouts in the Component Editor" on page 73
- "Adding custom shared shelves" on page 73
- "Virtual Memory limit of Maya (Windows)" on page 74

Modeling improvements:

- "Soften extruded polygon edges and faces" on page 74
- "Soften split polygons" on page 75
- "Polygon Bevel improvements" on page 77
- "UV Texture Editor improvements" on page 80
- "Subdivision surface improvements" on page 81

## 4 | Other improvements

### > What else is new and improved?

Animation improvements:

- "New animation clip per channel offset controls" on page 82
- "New channel state controls" on page 85
- "Improved animation-constraint blending" on page 89
- "Custom playback speed" on page 91

Character Setup improvements:

- "Improved Copy Smooth Skin Weights" on page 92
- "Bind skin to non-joint objects" on page 95
- "Prevent attachment of zero-weighted smooth skin influences" on page 96
- "Maintain Max Influences" on page 97
- "Add influence objects at any pose" on page 97
- "Automatic joint radii according to bone length" on page 98

Rendering improvements:

- "Improved Hypershade navigation" on page 99
- "New flags for the Command Line Renderer" on page 100
- "Maya vector rendering" on page 102
- "Maya hardware rendering" on page 102
- "mental ray for Maya rendering" on page 103

Other improvements:

- "Support for renaming per particle fields" on page 132
- "Specifying which particle systems to cache" on page 132
- "Artisan improvements" on page 132
- "3D Paint Tool improvements" on page 132
- "Fluid Effects improvements" on page 133
- "Fur improvements" on page 133

## 4 | Other improvements

### > Alias DirectConnect included with Maya

- "Hair improvements" on page 135
- "Cloth improvements" on page 136
- "Maya browser can get data from Maya" on page 144
- "FCheck improvements on Mac OS X" on page 146
- "Additional utilities on Mac OS X" on page 146
- "Improvements to IMF" on page 146
- "New MEL commands and flags" on page 147
- "API and Devkit improvements" on page 150

## General

### Alias DirectConnect included with Maya

Maya 6.5 includes Alias DirectConnect. For this release of Maya, Alias DirectConnect is available on Windows and Mac OS X only.

Alias DirectConnect is a family of products that let you import CAD data into Alias software. Each DirectConnect product lets you import a specific format of CAD file into Maya. For example:

- Alias® DirectConnect for STEP lets you import STEP files (.stp or .step).

Alias DirectConnect for STEP is included with and automatically installed with Maya.

#### To import a CAD file into Maya

- 1 Make sure the Alias DirectConnect plugin is loaded (Window > Settings/Preferences > Plug-in Manager; make sure the DirectConnect plug-in is checked).
- 2 Select File > Import.

## 4 | Other improvements

### > FBX plug-in included with Maya

**Note** A CAD file's parts and assemblies are imported as groups.

For more information on Alias DirectConnect (including supported CAD formats, system requirements, purchasing information, limitations and troubleshooting information), go to [www.alias.com/directconnect](http://www.alias.com/directconnect).

## FBX plug-in included with Maya

FBX is a platform-independent 3D interchange and authoring format which is designed to allow access to 3D content from all major 3D vendors and platforms. FBX supports all major 3D data elements, as well as 2D, audio, and video media elements. FBX can be used as an interchange format between 3D content providers, motion capture systems, 3D modeling/animation/rendering packages, 3D scanners (hardware), camera tracking systems, and other applications.

The Maya FBX plug-in allows you to read and write your scenes using the FBX file interchange format. Once in FBX format, data can be easily transferred to other programs (such as Alias MotionBuilder).

For more information about FBX, see the documentation that comes with the FBX plug-in, available as a PDF document in the docs directory of Maya.

## CGFx plug-in (Windows)

The CGFx plug-in is more accessible in Maya 6.5. Several important improvements and bug fixes have been made to the CGFx plug-in and we are making it available to you in two different formats.

## 4 | Other improvements

### > Subdivision surfaces heads-up display

- A compiled version of the plug-in is available. This may be useful to you if you don't have access to a compiler. The most recent version is available by choosing Download Bonus Tools from the Web from the Help menu of Maya.
- The source code continues to be available as part of the Maya 6.5 Development Kit, in the devkit directory under your Maya installation directory (by default, this is `C:\Program Files\Alias\Maya6.5\devkit\plug-ins`). You will need to compile this plug-in to use it; for more information about compiling and using plug-ins, see the API Guide.

You will require additional libraries from nVidia to use the CGFx plug-in. See [www.nvidia.com](http://www.nvidia.com) for details.

Please check the Maya section of the Alias Support Web site for updates to the CGFx plugin.

## Subdivision surfaces heads-up display

The Heads Up Display submenu now includes a *Subdiv Details* option to let you view subdivision surface information over top of a scene view. To turn on the heads-up display, select Display > Heads Up Display > Subdiv Display. Two lines of information get displayed in the scene view:



### Subdiv current level

Displays the level setting for the currently selected components of the subdivision surface. This option is particularly useful when subdivision components are displayed as points rather than numbers and the level information is required.

## 4 | Other improvements

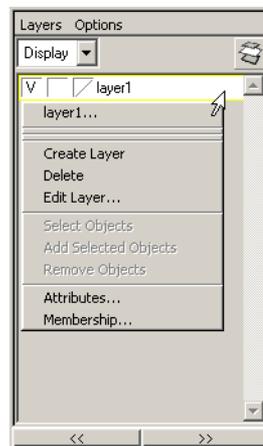
### > Improved Layer Editor navigation

#### Subdiv mode

Displays the mode for the currently selected subdivision surface. That is, Standard or Polygon Proxy mode.

## Improved Layer Editor navigation

We have improved navigation in the layer editor by including the layer name in the pop-up menu (when you right-click a layer).

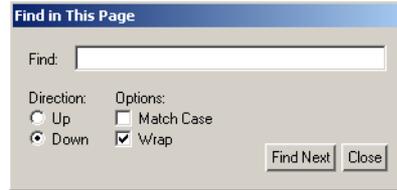


## Maya Web browser improvements

You can now search for text in the current page by choosing File > Find in This Page. You can choose to make your search case-sensitive, change the direction of the search, and choose whether the search wraps at the end of the page.

## 4 | Other improvements

### > Improved wheel mouse support



## Improved wheel mouse support

We have improved the support for scrolling wheels on mouse input devices.

In addition to the previous functionality (dolly, zoom in Hypershade and other windows and editors, scrolling in some windows), you can now scroll in most editors with scroll bars, such as the Attribute Editor, Outliner, and Channel Box. As well, you can use Shift+scroll wheel to scroll horizontally.

There is now a preference to enable or disable wheelmouse scrolling functionality (Window > Settings/Preferences > Interface > Devices).

## Scaling a light or camera locator

You can scale light and camera icons to help you see cameras in scenes where the scene size or camera perspective masks these locator icons. This is particularly useful when you have geometry parented under these objects and you want the locator to scale, but not the geometry.

The new attribute name is `locatorScale` and is present on all cameras and certain lights. It only affects the following types of lights:

- Directional light
- Spot light
- Area light
- Volume light

## 4 | Other improvements

### > Environment variable expansion in Maya file browser

Other light sources are drawn as bitmaps and have a fixed size regardless of scene scale.

#### **To scale a light or camera locator icon**

- 1** Select the light or camera icon you want to scale.
- 2** In the Attribute Editor, go to the shape tab for the light type (for example, for `directionalLight`, go to the `directionalLightShape` tab).
- 3** In the Object Display section, enter a value in the Locator Scale text box.

## Environment variable expansion in Maya file browser

The Maya file browser can now optionally display the expansion of environment variables in path names entered into the path name text box (for example, in the Reference Editor).

### Examples

This feature allows you to create scene files where the name of the environment variable stays bound into the scene file. This can be useful in a number of cases: for example, you can use the environment variable to switch between high, medium and low-res versions of referenced files just by changing the value of the environment variable and restarting Maya.

For example, if you put your high-resolution objects in reference files in `/usr/project25/models/hi-res` and low-resolution objects in `/usr/project25/models/lo-res` and then create an environment variable such as `$PROJ25_MODELS` which is set to `/usr/project25/models/hi-res`, then set the option in Maya to not expand environment variables (see below). If you then type a name such as `$PROJ25_MODELS/penguin.ma` into the file browser when referencing

## 4 | Other improvements

### > Environment variable expansion in Maya file browser

in an object, you will get the high-resolution version of `penguin.ma`. If you save your scene, it will contain the string `$PROJ25_MODELS/penguin.ma` instead of the fully-expanded path name.

If you now change `$PROJ25_MODELS` to point to the lo-resolution directory, restart Maya and reload the scene, it will now look for `penguin.ma` in the lo-resolution folder.

You can also create a portable scene file which is independent of directories and folders. The environment variables are set by the user to point to the location of any data which the scene accesses.

## Preference

We have added a new preference which controls if the Maya File Browsers expand environment variables in path names. In the Preferences editor (Window > Settings/Preferences > Preferences), select User Interface. In the File Browser section, there are two options for Environment Variables:

### Retain

Environment variables typed into Maya File Browser path name textfields are displayed unexpanded. When accepting the selected filename in a Maya File Browser, the unexpanded path name is passed to the recipient in unexpanded form. For example, if the file browser were invoked on a file texture node and a path name such as `$IMAGES/sgi/mandrill.gif` is entered, the unexpanded name `$IMAGES/sgi/mandrill.gif` is displayed in the file texture's Image Name textfield. Should the recipient field be exportable, that field is output in unexpanded form.

### Expand

Environment variables are automatically expanded by the Maya File Browsers. This is the default setting for the preference and also the behavior consistent with previous Maya releases. For example, if the environment variable `$IMAGES` is set to `/usr/images`, and the path name `$IMAGES/sgi/mandrill.rgb` is typed

## 4 | Other improvements

### > Environment variable expansion in Maya file browser

into the path name textfield of a Maya File Browser and the user accepts the current selection, the typed path name changes to the expanded name; that is, `/usr/images/sgi.mandrill.rgb`.

The expansion of environment variables when Retain is selected happens internally with each field which invokes the File Browser. For example, when specifying a filename for a file texture, the node must now internally find a filename which contains environment variables. Should the recipient of File Browser output be unable to handle environment variables in the filenames it accepts, as a workaround, manually expand the environment variable.

**Note** Some file browsers used by Maya are standard platform-specific file browsers and not the Maya File Browser. Only the Maya File Browser is able to handle unexpanded path names.

### Other notes

Environment variables are typically defined externally prior to starting Maya, but can also be set via the `putenv` command.

The internal Maya environment variable expansion code handles `$` as an environment variable delimiter on all platforms, as well as the standard combinations of `%` and `{}` on Windows, and `~` on IRIX and Linux.

The Up button in the browser only expands the current path name if necessary. For example, if the path name is `$IMAGES/sgi/` and you click Up, the path name changes to `$IMAGES/`. Clicking Up again expands the current path name first and `/usr` becomes the current selection.

## Create new layouts in the Component Editor

You can now create custom layouts in the Component Editor.

In a Component Editor tab, select columns and select Options > Remember This Layout. You are prompted to name this layout. A tab appears with the name and column layout you selected.

To remove the custom layout, select Options > Delete Current Layout.

## Adding custom shared shelves

We have added a feature that allows you to create custom shared shelves. You can use this feature to create custom shared shelves for different groups who need access to different sets of shared commands; for example, when working on different stages of a production.

Previously, Maya stored and retrieved shelves only from the default shelf directory, for example, `$USER/maya/6.0/prefs/shelves`.

The new environment variable `MAYA_SHELF_PATH` allows you to specify the directories that Maya searches to access the shelves. You can store shelves in a location accessible by different groups who then set the `MAYA_SHELF_PATH` variable for their workstation.

You can specify more than one directory using `MAYA_SHELF_PATH`, separated by colons; for example, `Production/shelf:Shot/shelf:MyDir/shelf` specifies three different shelf directories. At startup, Maya searches each directory in the order specified to instantiate shelves. After searching the directories specified in `MAYA_SHELF_PATH`, Maya continues to add shelves from the default shelf directory. Once a shelf exists, a shelf with the same name in the subsequent searched directories is ignored.

When a new shelf is created, it is always saved in the default shelf directory. To share it, move it to the desired directory (that is, one of the ones specified by `MAYA_SHELF_PATH`).

## 4 | Other improvements

### > Virtual Memory limit of Maya (Windows)

To restrict access to a shelf directory, disable the write permission of that directory. A warning appears when a user attempts to write into a restricted shelf directory (while exiting or saving all shelves).

## Virtual Memory limit of Maya (Windows)

Under Windows 2000/XP, the normal application memory partition is 2GB for system and 2GB for applications. For Maya, this means that the largest virtual memory that can be obtained is 2GB. Portions of this address space is used by MFC (Microsoft Foundation Classes), as well as graphic card drivers. The effective virtual memory size of Maya is approximately 1.6GB.

Under Windows XP SP2, there is a system boot option that partitions the memory into 1GB system and 3GB user. This option is /3GB.

Microsoft Reference:

- [www.microsoft.com/whdc/system/platform/server/PAE/PAEmem.msp](http://www.microsoft.com/whdc/system/platform/server/PAE/PAEmem.msp)  
Windows XP SP1 may have problems starting if /3GB is used. See Microsoft knowledgebase article: 328269
- [support.microsoft.com/default.aspx?scid=kb;en-us;328269](http://support.microsoft.com/default.aspx?scid=kb;en-us;328269)

Maya 6.5 is linked with the /LARGEADDRESSAWARE linker option. This effectively adds approximately 1GB to the virtual memory that Maya can grow to when used with the /3GB boot option.

## Modeling

### Soften extruded polygon edges and faces

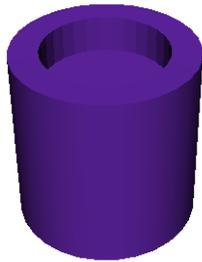
The new *Smoothing Angle* option lets you automatically smooth the edges of your extruded polygonal geometry. Previously, the edges of extruded geometry were hard by default, and to soften the edges you had to use Edit Polygons > Normals > Soften/Harden.

## 4 | Other improvements

### > Soften split polygons

Now when you extrude your polygonal geometry, you can specify whether or not the edges of your extruded geometry are *soft* or *hard*. For example, if you are extruding part of a character's face, and you want the extruded edges to be soft, you can set the Smoothing Angle to a *high* value such as 180. Or if you are extruding part of a building model, and you want the extruded edges to be hard, you can set the Smoothing Angle to a *low* value such as 0.

The Smoothing Angle option is located in the Extrude Face/Edge Option windows and in the Poly Extrude Face/Edge History sections of each polyExtrudeEdge/polyExtrudeFace node. The default Smoothing Angle value for extruded faces/edges is 30.



Extrude Face  
Smoothing Angle = 0



Extrude Face  
Smoothing Angle = 180

## Soften split polygons

The new *Smoothing Angle* option in the Split Polygon Tool Settings lets you automatically smooth the edges of your split polygonal geometry. Previously, the edges of split geometry were hard by default, and to soften the edges you had to use Edit Polygons > Normals > Soften/Harden.

## 4 | Other improvements

### > Soften split polygons

Now when you split your polygonal geometry, you can specify whether or not the edges of your split geometry are *soft* or *hard*. A *high* Smoothing Angle value like 180 makes your edges soft or rounded, and a *low* Smoothing Angle value like 0 makes your edges hard or sharp.

#### Note

When the angle between your split polygonal faces is 0:

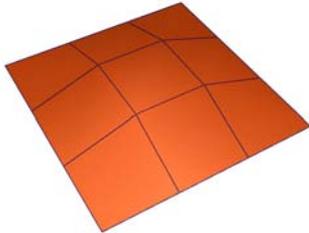
- You can make the shared edges *hard* by setting their Smoothing Angles to 0.0.
- You can not see the effect of your split polygonal faces Smoothing Angles in the scene view.

The Smoothing Angle option is located in the Split Polygon Tool settings and in the Poly Split History section of each polySplit node. The default Smoothing Angle value for split polygons it is 0.

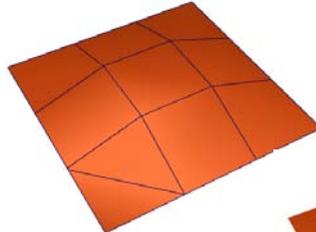
## 4 | Other improvements

### > Polygon Bevel improvements

Before  
Split Polygon



After  
Split Polygon



Split Polygon Tool  
Smoothing Angle = 0



Split Polygon Tool  
Smoothing Angle = 180

## Polygon Bevel improvements

### Merge Vertices

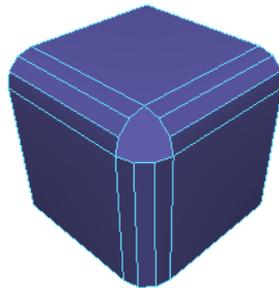
The polygon Bevel feature now incorporates attributes for merging vertices. Vertices are merged automatically when beveling an edge without having to use the Merge Vertices feature separately. When you merge vertices, coincident edges and their associated UVs are also merged automatically (within a specified threshold). The merge vertices attributes are on by default and can be edited from both the Channel Box and within the Cleanup section of the *polyBevel* Attribute Editor. For more information on merge vertices, see “Edit Polygons > Merge Vertices” in the *Polygonal Modeling* guide.

## 4 | Other improvements

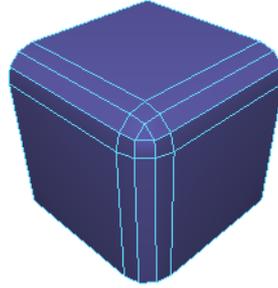
### > Polygon Bevel improvements

#### Subdivide Ngons

The polygon Bevel feature now provides an attribute for subdividing any faces that have large numbers of edges as a result of doing a bevel operation with more than one segment. The Subdivide Ngons option is on by default and can be controlled from both the Channel Box and the Cleanup section of the *polyBevel* Attribute Editor.



Polygon Bevel  
Subdivide Ngons = Off



Polygon Bevel  
Subdivide Ngons = On

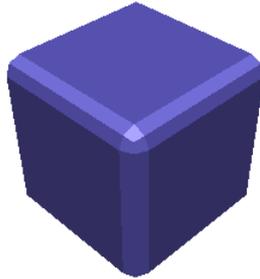
#### Smoothing Angle

PolygonBevel now incorporates a *Smoothing Angle* attribute to control the hardness of beveled edges. You can specify whether you want the recently beveled edges to appear hard or soft when shaded by setting the Smoothing Angle attribute value. The Smoothing Angle attribute can be edited from either the Channel Box or within the Poly Bevel History section of the *polyBevel* Attribute Editor.

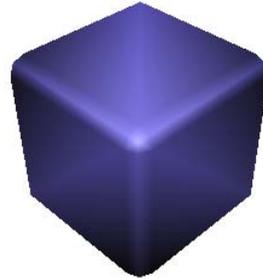
If you want the beveled edges to be soft, you can set the Smoothing Angle to a high value such as 180. If you want the beveled edges to be hard, you can set the Smoothing Angle to a low value such as 0.

## 4 | Other improvements

### > Polygon Bevel improvements



Polygon Bevel  
Smoothing Angle = 0



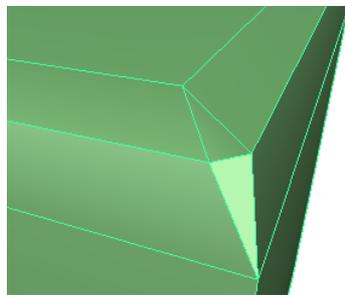
Polygon Bevel  
Smoothing Angle = 180

In general, if the angle between two shared edges is greater than the value specified by the Smoothing Angle attribute, the beveled edge will be shaded to appear hard. The default Smoothing Angle value for polygon Bevel is 30 degrees.

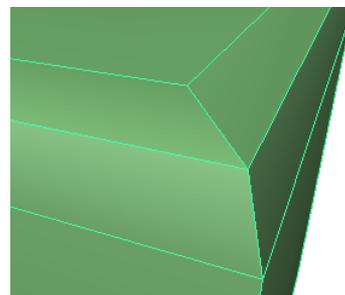
### Mitering Angle

The polygon Bevel tool now incorporates a *Mitering Angle* attribute to control how two intersecting beveled edges get joined when an intersecting non-beveled edge is involved.

You can specify whether you want the recently beveled edges to be mitered or not by setting the Mitering Angle value as required.



Bevel on corner  
Miter Angle = 0



Bevel on corner  
Miter Angle = 180

## 4 | Other improvements

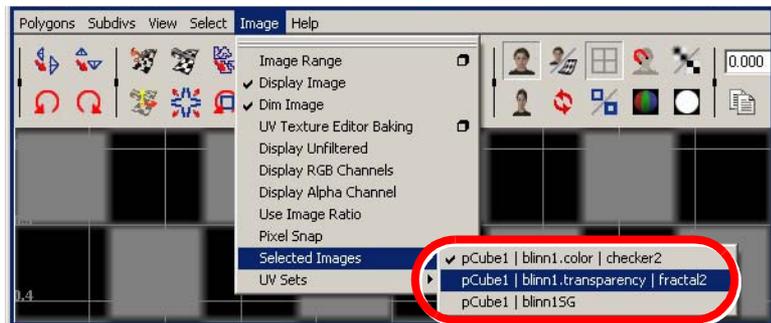
### > UV Texture Editor improvements

The Mitering Angle attribute can be edited from either the Channel Box or within the Cleanup section of the *polyBevel* Attribute Editor.

If the angle between the two beveled edges is greater than the specified mitering angle, the beveled edges will not be mitered. The Mitering Angle feature is on by default and is set to 180 degrees.

## UV Texture Editor improvements

### Image > Selected Images improved



The Image > Selected Images submenu now lists all of the images available for display as the background within the UV Texture Editor for the selected polygon meshes.

The Selected Images submenu displays texture channels for Maya native software shaders and hardware shader plug-ins (including CgFX shaders).

## 4 | Other improvements

### > Subdivision surface improvements

#### Note

In previous versions it was possible to choose a specific Texture Channel for a given shader (for example, color, transparency, and so on). However, this did not provide access to specific textures and was only accessible on some shaders through the Hardware Texture settings within the Attribute Editor. You can access this original functionality by selecting the shading group entry from the Selected Images pull-down menu. For example, for a shader named *blinn1*, the shading group entry in the pull-down menu would be named *blinn1SG*.

## Dim Image

The UV Texture Editor now provides a Dim Image feature to reduce the brightness of the currently displayed background image. Dimming the background image lets you more easily view and select components in the UV Texture Editor's view.

### To dim the background image in the UV Texture Editor

- 1 In the UV Texture Editor, select Image > Dim Image.
- 2 To return the display of the image to normal brightness select the Image > Dim Image to toggle it off.

## Subdivision surface improvements

A new *Select Coarser Components* feature has been added for subdivision surface work to let you move to a coarser level in the subdivision surface hierarchy while maintaining an active selection. For example, when editing a face or vertex at one level, and you select Subdivision Surfaces > Select Coarser Components, the related subdivision surface component at the next coarser level is selected.

## 4 | Other improvements

### > New animation clip per channel offset controls

This Select Coarser Components feature is also available by pressing the right mouse button on the subdivision surface and selecting Select Coarser from the marking menu or by pressing Ctrl + Up.

## Animation

### New animation clip per channel offset controls

In Maya 6.5, rather than offsetting whole clips, you can now offset each clip's individual channels with the new *Channel Offsets*. The Channel Offsets provide you with a more open-ended way of specifying how Maya interprets channel values within and between clips in the Trax Editor.

In previous versions of Maya, you had to place animation channels in separate clips under a subcharacter if you wanted them to have different offsets. For example, you can now treat animation control objects in world space like root nodes by giving them absolute rotation channels and relative translation channels. This lets you correctly offset your control objects without using subcharacters.

#### Absolute and relative offsets

You can set the offset of a clip channel to *absolute* or *relative*.

Absolute channels do not accumulate motion. This means that when the offset of a clip channel is absolute, then that channel *does not add* the motion of the equivalent channel in the previous clip to its own animation.

#### Note

Rotation channels are the only clip channels that have absolute offsets by default.

## 4 | Other improvements

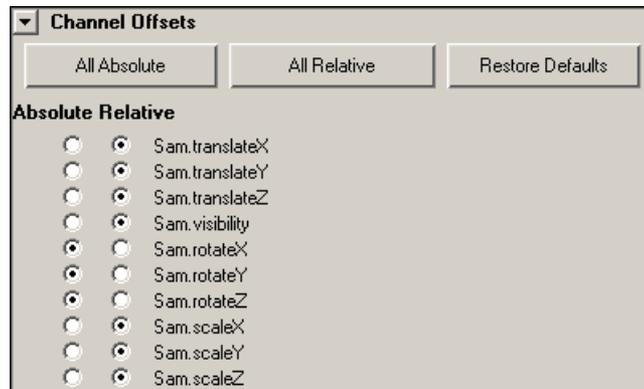
### > New animation clip per channel offset controls

Relative channels accumulate motion. This means that when the offset of a clip channel is relative, then that channel *adds* the motion on the equivalent channel in the previous clip to its own animation.

For more information on absolute and relative clip channel offsets, see “Offset clip channels” in the Nonlinear Animation chapter of the *Animation* guide.

## Channel Offsets

The Channel Offsets are located on each animClip node in the Attribute Editor.



### Absolute and Relative radio buttons

The Absolute and Relative radio buttons let you set the offsets for each clip's channels. When an Absolute radio button is turned on, the offset of the channel beside the radio button is *absolute*. When a Relative radio button is turned on, the offset of the channel beside the radio button is *relative*.

## 4 | Other improvements

### > New animation clip per channel offset controls

Each channel name beside the radio buttons has a prefix. This prefix is the name of the object to which the animation channels belong. For example, if the animation in a clip belongs to an object named *Sam*, then the prefix *Sam.* appears before each of the clip's channels in the Channel Offsets.

#### All Absolute

The All Absolute button lets you set the offsets of all the channels for the current clip to *absolute*.

#### All Relative

The All Relative button lets you set the offsets of all the channels for the current clip to *relative*.

#### Restore Defaults

The Restore Defaults button lets you revert the current clip's Channel Offsets to their original settings.

## Copy Channel Offsets

The new *Copy Channel Offsets* menu item in the Trax Editor lets you apply the current channel offset settings from one clip to another. This is useful when you want multiple clips to have the same offset settings. Copy Channel Offsets is located under the Edit menu in the Trax menu bar.

For more information on clip channel offsets, see “*Offset clip channels*” in the Nonlinear Animation chapter of the *Animation* guide.

### **To copy the channel offset settings from one clip to another clip**

- 1** Select the clip that has the channel offset settings you want to copy. This clip is the source clip.
- 2** Shift+click the clip to which you want to apply the source clip's channel offset settings. This clip is the destination clip.

## 4 | Other improvements

### > New channel state controls

#### Note

- The source and destination clips must belong to the same character.
- The source and destination clips must have common channels.
- You cannot copy clip channel offset settings from one source clip to multiple destination clips.

**3** In the Trax menu bar, select Edit > Copy Channel Offsets.

The destination clip now has the same channel offset settings as the source clip.

## New channel state controls

You can now display *nonkeyable* channels in the Channel Box. This lets you easily access nonkeyable channels from the Channel Box without running the risk of accidentally keying them. Previously, you could only display channels in the Channel Box by making them keyable.

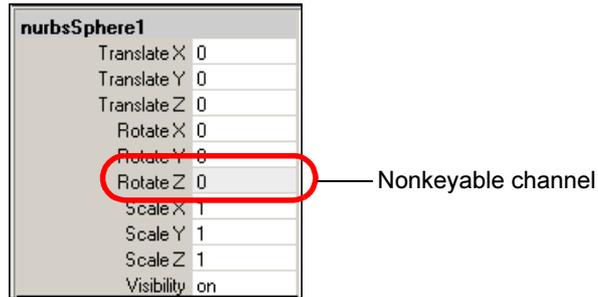
## Make channels keyable/nonkeyable from the Channel Box

You can now change the state of channels in the Channel Box from keyable to nonkeyable using the new *Make Selected Keyable* and *Make Selected Nonkeyable* options located in the Channel Box's Channels menu.

Channels that are nonkeyable appear *light gray* in the Channel Box.

## 4 | Other improvements

### > New channel state controls



When keying an object, the state of its channels affects which of the object's channels are animated. The *keyable* and *nonkeyable* channel states let you specify which of an object's channels are (keyable) or are not (nonkeyable) keyed when using the following methods to animate your object:

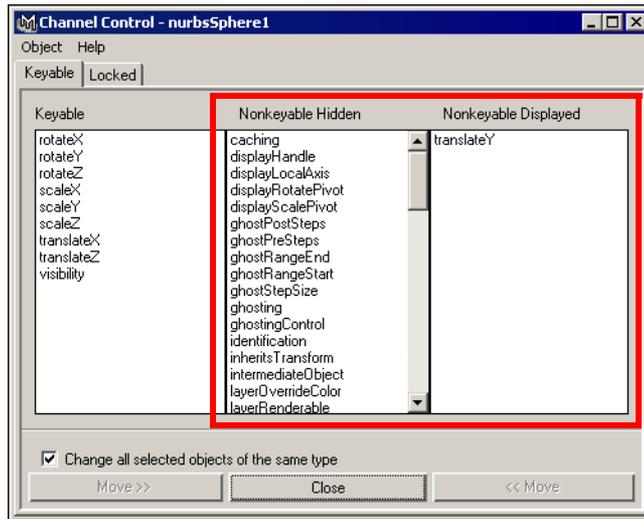
- Autokey
- Animate > Set Key (shortcut: S key)

## Improved Channel Control Editor

In Maya 6.5, two new columns have been added to the Channel Control Editor: *Nonkeyable Hidden* and *Nonkeyable Displayed*.

## 4 | Other improvements

### > New channel state controls



When a channel's name is in the Keyable or Nonkeyable Displayed column, the channel appears in the Channel Box. When a channel's name is in the Nonkeyable Hidden column, the channel does not appear in the Channel Box.

## How do I make channels keyable or nonkeyable in Maya 6.5?

### To make a channel keyable or nonkeyable from the Channel Box

- 1 Select the object that has the channel you want to make keyable or nonkeyable.
- 2 In the Channel Box, select the channel you want to make keyable or nonkeyable.
- 3 Do one of the following:
  - Right-click the channel.  
The Channel Box's Channels pop-menu appears.

## 4 | Other improvements

### > New channel state controls

- Select *Channels* in the Channel Box's menu bar.

The Channel Box's Channels menu appears.

#### 4 Do one of the following:

- If you want to make the selected channel keyable, select Make Selected Keyable.
- If you want to make the selected channel nonkeyable, select Make Selected Nonkeyable.

The selected channel now appears light gray in the Channel Box.

### **To make a channel keyable or nonkeyable from the Channel Control Editor**

#### 1 Select the object that has the channel you want to make keyable or nonkeyable.

#### 2 In the Maya main menu bar, select Window > General Editors > Channel Control.

The Channel Control Editor appears with all the selected object's channels loaded.

#### 3 Select the channel you want to make keyable or nonkeyable.

#### 4 Click the or button to do one of the following:

- To make a channel keyable, and have it appear in the Channel Box, move the selected channel to the Keyable column.
- To make a channel nonkeyable, and have it *not* appear in the Channel Box, move the selected channel to the Nonkeyable Hidden column.
- To make a channel nonkeyable, and have it appear in the Channel Box, move the selected channel to the Nonkeyable Displayed column.

## 4 | Other improvements

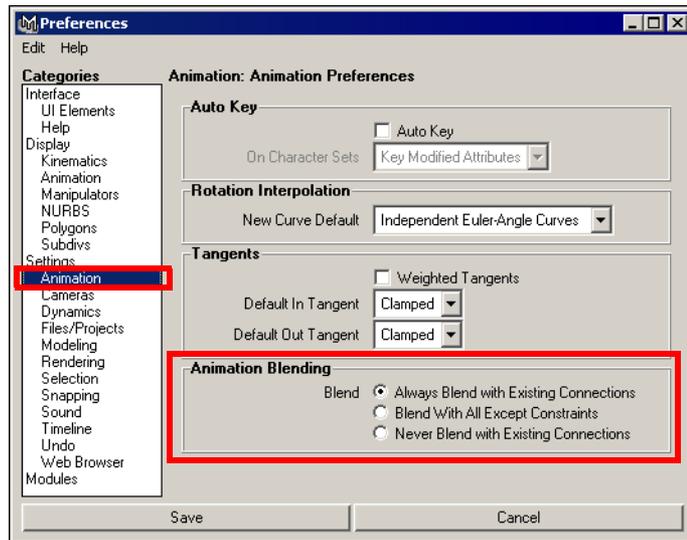
### > Improved animation-constraint blending

## Improved animation-constraint blending

The new *Animation Blending* preferences in the Maya Preferences window let you specify the types of blends that can occur between object connections. For example, when you turn on the *Always Blend with Existing Connections* Animation Blend preference, you can then both animate and constrain a single object.

### Note

In Maya 6.5, the Keys category in the Maya Preferences window has been renamed to *Animation*.



### Always Blend with Existing Connections

When this preference is on:

- You can move or key objects that have existing connections such as animation and constraints.

## 4 | Other improvements

### > Improved animation-constraint blending

- When you key or constrain an object, Maya inserts a `pairBlend` node between the object's existing connections and the new keys or constraints.
- The Lock Output attribute is *off* by default for all new constraints.

Always Blend with Existing Connections is on by default.

#### Blend With All Except Constraints

When this preference is on:

- You can only key objects that are not constrained.
- When you key an object, Maya inserts `pairBlend` nodes between the object's existing connections and the new keys.
- Objects with connections cannot be constrained.
- The Lock Output attribute is *on* by default for all new constraints.

#### Never Blend with Existing Connections

When this preference is on:

- You can only key objects that have no existing connections.
- Maya does not insert `pairBlend` nodes to blend object connections.
- The Lock Output attribute is *on* by default for all new constraints.

#### New Lock Output attribute

The new *Lock Output* constraint attribute lets you specify whether or not you can directly manipulate an object's constrained channels. For example, when an orient constraint's Lock Output attribute is off, you can change the orientation of the constrained object without having to rotate the driver object.

## 4 | Other improvements

### > Custom playback speed

When a constrained object's Lock Output attribute is *on*, you can only manipulate its constrained channels by transforming its driver object. When a constrained object's Lock Output attribute is *off*, you can directly manipulate its constrained channels, without having to transform its driver object.

The Lock Output attribute is located in the Constraint Attributes section of each constraint node.

**Note** The Animation Blending preferences affect the default state of the Lock Output attribute.

## Custom playback speed

In Maya 6.5, you can now easily specify a custom playback speed (in frames per second) for your scene using the new *Other Speed* setting. For example, if you specify an Other Speed value of 72, then your scene's animation will play back at 72 fps.

Previously, to specify a custom playback speed, you had to set a multiplier value for the Real-time playback speed.

The Other Speed setting is located in Window > Settings/Preferences > Preferences > Timeline category > Playback section. For more information, see "*Timeline preferences*" in the Preferences and Customization chapter of the *Basics* guide. The Other Speed setting is only available when you select *Other* from the Playback Speed drop-down list.

## 4 | Other improvements

### > Improved Copy Smooth Skin Weights

## Character Setup and Deformers

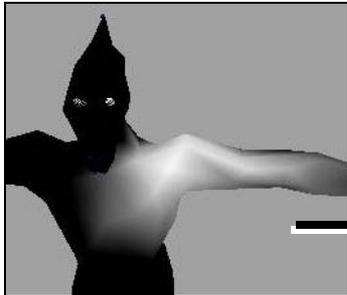
### Improved Copy Smooth Skin Weights

Now when you copy smooth skin weights from one skinned polygonal mesh (source skin) to another (destination skin), the new *Use Smooth Skin Weight Copy* option improves the results of the copy.

Use Smooth Skin Weight Copy lets you smooth the distribution of copied weights on the destination skin. Use Smooth Skin Weight Copy is most useful when copying smooth skin weights from a low resolution polygonal mesh to a high resolution polygonal mesh.

Use Smooth Skin Weight Copy is located in Skin > Edit Smooth Skin > Copy Skin Weights > .

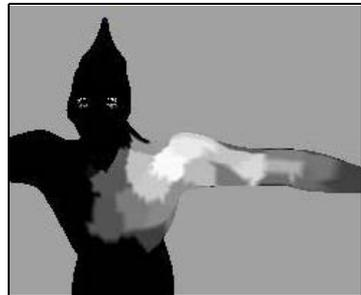
Low resolution mesh



High resolution meshes



Use Smooth Skin Weight Copy *on*



Use Smooth Skin Weight Copy *off*

Character by Rick Kogucki

## How do I copy smooth skin weights between polygonal meshes in Maya 6.5?

### Tips

For best results when copying smooth skin weights, note the following:

- Your source and destination skins should be positioned closely together in your scene. For example, the skinned meshes you are copying weights to and from should be located at the same position in Maya's coordinate space.
- Your source and destination skins should look similar. For example, copying the smooth skin weights from one medium sized biped model to another produces much better results than copying smooth skin weights from a small biped model to a large biped model.
- When the source and destination skins are bound to different skeletons, make sure that the way they are parented (if applicable) is consistent. For example, if the source skin's skeleton is grouped under a node, make sure that the destination skin's skeleton is also grouped under a node. Otherwise, Maya may make the wrong associations between joints, and this can produce undesirable results in the copy.

### 1 Do one of the following:

- Select the smooth skin polygonal mesh (or group of meshes) you want as the source skin, and then select the polygonal mesh (or group of meshes) you want as the destination skin.
- Select the polygonal mesh vertices you want as the source skin vertices, and then select the polygonal mesh vertices you want as the destination skin vertices.

## 4 | Other improvements

### > Improved Copy Smooth Skin Weights

- 2 Select Skin > Edit Smooth Skin > Copy Skin Weights > .

The Copy Skin Weights Options window appears.

- 3 Do one of the following:

- Turn *on* Use Smooth Skin Weight Copy (Use Smooth Skin Weight Copy is on by default) for best results when copying smooth skin weights from a low resolution polygonal mesh to a high resolution polygonal mesh.

#### Note

The Copy Skin Weights operation is slower when Use Smooth Skin Weight Copy is turned *on* than when it is turned *off*.

- Turn *off* Use Smooth Skin Weight Copy for a faster, less accurate smooth skin weights copy.

#### Warning

If Use Smooth Skin Weight Copy is *off* when copying smooth skin weights from a low resolution polygonal mesh to a high resolution polygonal, then the results of your Copy Skin Weights operation will have undesirable aliasing.

- 4 Do one of the following:

- Click Copy to copy the smooth skin weights from the source skin to the destination skin.  
The options window closes.
- Click Apply to copy the smooth skin weights from the source skin to the destination skin.  
The options window remains open.
- Click Close to close the options window without copying the skin weights.

## Bind skin to non-joint objects

With the new *Object Hierarchy* option in the Smooth Bind Options window, you can now bind geometry to any transform node or hierarchy of nodes such as an empty group node or a locator. For example, you can now create a deformation effect by binding a polygonal mesh to locator nodes that are connected to the CVs of a spline curve. In previous versions of Maya, you could only bind geometry to hierarchies of joints.

**Note** When using the Object Hierarchy option, you can only select joints or objects that cannot be skinned (for example, group nodes or locators—not pieces of geometry) as the initial influences in the bind.

## Improved Smooth Bind Options

Bind to



- Joint Hierarchy** In Maya 6.5, the Complete Skeleton Bind to option has been renamed to *Joint Hierarchy*. For details on this option, see Skin > Bind Skin > Smooth Bind in the Skinning chapter of the *Character Setup and Deformers* guide.
- Object Hierarchy** When this option is selected, the selected deformable geometry is bound to the entire hierarchy of the selected joint or non-joint transform node, from the top node down through the entire node hierarchy. If there are

## 4 | Other improvements

### > Prevent attachment of zero-weighted smooth skin influences

any joints in the node hierarchy, they are also included in the bind. With this Bind to option, you can bind entire pieces of geometry to nodes like groups or locators.

#### Bind Method



#### Closest In Hierarchy

In Maya 6.5, the Closest Joints bind method has been renamed to *Closest In Hierarchy*. For details on this option, see Skin > Bind Skin > Smooth Bind in the Skinning chapter of the *Character Setup and Deformers* guide.

## Prevent attachment of zero-weighted smooth skin influences

Now when you smooth skin bind your geometry, you can turn *on* the new *Remove Unused Influences* option to prevent any weighted influences that would receive a zero weighting from being included in the bind. You can turn *off* Remove Unused Influences if you want to be able to set or paint weights for all influences (for example, all the joints in a skeleton) after the bind.

The Remove Unused Influences option is located in the Smooth Bind Options window. For more information on these options, see Skin > Bind Skin > Smooth Bind >  in the *Character Setup and Deformers* guide.

The new *Remove Unused Skin Influences* option in the Optimize Scene Size Options window also lets you remove any zero weighted influences from your character's skin bind. For more information on the Optimize Scene Size window, see File > Optimize Scene Size.

## 4 | Other improvements

### > Maintain Max Influences

These new options are useful when you want to reduce the number of calculations for your scene to increase your scene's playback speed. For additional ways you can increase your scene's playback and interactive speed, see "*Optimize your scene for playback*" in the *Animation* guide.

## Maintain Max Influences

The new *Maintain Max Influences* option lets you control whether or not the Max Influences smooth bind setting is maintained after you bind your geometry. When Maintain Max Influences is *on*, your smooth skinned geometry cannot have, at any time, a number of influences *greater* than that specified by Max Influences.

For example, if Max Influences is set to 3, and you then paint or set weights for a fourth joint, one of the weights of the other three joints is set to 0 to maintain the total number of weighted influences specified by Max Influences. This limits the redistribution of weights to a specific number of influences, and ensures that the primary joints are the ones that receive the weights.

Maintain Max Influences is located in Skin > Bind Skin > Smooth Bind >  and in the Smooth Skin Attributes section of the skinCluster node.

**Note** If you turn on Maintain Max Influences in the Attribute Editor, your skin's weights will not be modified until you reassign its weights by clicking the Update Weights button.

## Add influence objects at any pose

You can now add influence objects to smooth skinned geometry at any pose—not just at the bind pose.

## 4 | Other improvements

### > Automatic joint radii according to bone length

However, adding an influence object to smooth skinned geometry when *not* at the bind pose produces slightly different results than when at the bind pose. For example, if you add an influence object to geometry that is not at its bind pose, then some deformation may occur at the location of the influence due to a redistribution of weights in that area.

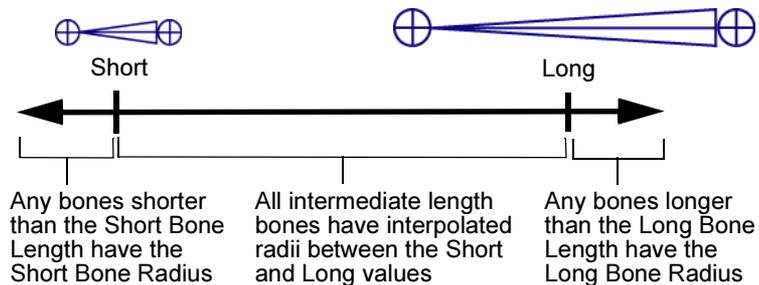
### Related topics

- ❖ “Bind pose” in the *Character Setup and Deformers* guide.
- ❖ “Smooth skin influence objects” in the *Character Setup and Deformers* guide.

## Automatic joint radii according to bone length

Now when you create joints, the radii of your joints are determined by the length of their bones. This means that when you draw a joint chain, the radius of all the joints in the chain automatically adjust to the length of their bones. For example, short bones like those in the human hand generate small joints and long bones like those in the human leg generate much larger joints than those in the hand.

### New Joint Tool settings



## 4 | Other improvements

### > Improved Hypershade navigation

#### Bone Radius Settings

Short Bone Length	Sets the bone length at which a bone is <i>short</i> .
Short Bone Radius	Sets the bone radius of <i>short</i> bones. This is the minimum bone radius.
Long Bone Length	Sets the bone length at which a bone is <i>long</i> .
Long Bone Radius	Sets the bone radius of <i>long</i> bones. This is the maximum bone radius.

#### New Joint Radius attribute

The new *Radius* attribute lets you set the radius of individual joints. The radius attribute is located in the Joint section of the joint node.

## Rendering

### Improved Hypershade navigation

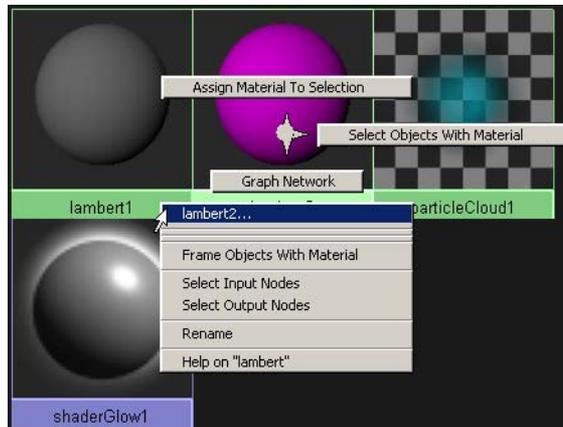
We have improved navigation in Hypershade by including the bin name in the pop-up menu (when you right-click a bin). In addition, you can rename an existing bin by double-clicking it.

## 4 | Other improvements

### > New flags for the Command Line Renderer



We've also improved navigation in Hypershade by including the swatch name in the pop-up menu (when you right-click a swatch).



## New flags for the Command Line Renderer

We've introduced the following new flags for the `Render` command:

## 4 | Other improvements

### > New flags for the Command Line Renderer

Flag Name	Definition
-sel	Selects individual objects, groups or sets.
-l	Selects display layers or render layers.
-log	Specifies a render log file and location. The log file contains the standard output and standard error information.
-lic	<p>Checks out the best available license for mental ray for Maya. Two options are available for this flag, <code>mu</code> and <code>mc</code>.</p> <p>The <code>mu</code> option checks for a mental ray for Maya Unlimited license. If the license is available, it is used, and the maximum number of satellites is set to 8. If the license is not available, a mental ray for Maya Complete license is used, and the maximum number of satellites is set to 2.</p> <p>The <code>mc</code> option checks out a mental ray for Maya Complete license. The maximum number of satellites is set to 2.</p> <p>This flag is for use with mental ray's Satellite rendering.</p>

## 4 | Other improvements

### > Maya vector rendering

For more information, see “Rendering with mental ray for Maya and mental ray satellite” on page 103 and “*Command line renderer*” in the Maya Help.

## Maya vector rendering

### New maximum resolution for vector rendering

Vector rendering now supports rendering for resolutions up to 3200x3200 pixels. This lets you use bitmap renders created with the Vector Renderer for High Definition and standard film (2K) production.

## Maya hardware rendering

### Unlimited file texture size

We’ve removed the 1K file texture size limit in Maya 6.5. You can now create and use file textures larger than 1K on Windows, IRIX, and Linux.

**Note** File textures larger than 1K are not supported on Mac OS X.

### New attribute for Hardware Shader Plug-in

A new attribute, Ignore Hardware Shader, is now available. Use this attribute to ignore a hardware shader on a particular object without reassigning shaders. When Ignore Hardware Shader is on, the object appears with the untextured default of the hardware shader that is applied to it.

The Ignore Hardware Shader attribute is available on the shape node of polygon and NURBS objects in the Attribute Editor, in the Object Display section. You can turn the attribute on or off per-object.

## 4 | Other improvements

### > Summary of major improvements

To apply the Ignore Hardware Shader attribute to multiple polygon or NURBS objects, Shift+click the objects in your scene, then select Display > Object Display > Ignore Hardware Shader. To show the hardware shader, Shift+click the objects in your scene, then select Display > Object Display > Use Hardware Shader.

## mental ray for Maya rendering

### Summary of major improvements

In Maya 6.5, we've integrated mental ray version 3.4. This version is focused on performance and large scene handling. Major improvements to mental ray for Maya may cause some of your existing scenes to render differently. These improvements include:

- "Improved Final Gather method" on page 106
- "New Rapid Scanline method" on page 108
- "Improved displacement approximation in mental ray for Maya" on page 109
- "Improved approximation on Subdivision surfaces" on page 110
- "New file texture filter types" on page 129

For more information, see the detailed documentation for each topic.

## Rendering with mental ray for Maya and mental ray satellite

### Overview

Mental ray for Maya now supports a new form of network rendering: *mental ray satellite rendering*. The new network rendering capability, based on mental ray satellite technology, lets you install a version of mental ray (satellite) on any networked computer. This special version of mental ray resides on that machine and waits for rendering tasks from Maya. It supplements the mental ray rendering capabilities in

## 4 | Other improvements

### > Rendering with mental ray for Maya and mental ray satellite

Maya with additional rendering power provided by the networked computers' CPUs. It can only receive these tasks from Maya and does not have the ability to receive .mi files for rendering; it also cannot be a master for other mental ray slave machines.

A mental ray distributed rendering can speed up all of the following tasks:

- interactive rendering (through the Maya interface)
- IPR rendering with mental ray for Maya
- interactive batch rendering (a batch render started by Maya)
- command-line rendering
- pre-lighting (baking to textures and vertices)

The mental ray for Maya plug-in that is integrated in Maya always lets you render on up to 4 local CPUs. With Maya Complete, mental ray satellite rendering can take place on 2 additional remote CPUs. With Maya Unlimited, mental ray satellite rendering can take place on 8 additional remote CPUs.

The rendering process is typically initiated from the *master* machine (the machine running Maya or where the command-line render starts from). The *slave* machines (satellite rendering computers) process rendering tasks they receive over the network from the master and send back results. The master machine handles task distribution, load balancing, and the collection of received rendering tasks which make up the rendered image.

### Master machine setup (Maya)

Once you have installed Maya, you do not need to install anything else on the master machine. Mental ray satellites are enabled (licensed) and controlled by the mental ray for Maya technology in Maya.

## 4 | Other improvements

### > Rendering with mental ray for Maya and mental ray satellite

On the master machine, the `maya.rayhosts` file controls which slaves participate in the render, limited by the number of satellites permitted by Maya (2 for Maya Complete, 8 for Maya Unlimited). For more information, see the “Network Rendering” chapter in the *Rendering guide*, especially “*Set up a master machine with mental ray for Maya or mental ray for Maya Satellite*” in the *Rendering guide*, which describes how to set up the rayhosts file on your system.

You must specify the port number for satellite rendering in the `maya.rayhosts` file. You must specify it in the rayhosts file as well as the host name. The port number for mental ray for Maya Satellite is 7103. For example, the rayhost file might look like this:

```
host1:7103
host2:7103
```

You can set the port number to a different value if you are using a different port on both master and slave machines.

For more information about installing mental ray on slave machines and setting up slave machines, see “*Slave machine installation*” in the *Installation and Licensing guide*.

#### Note

A full version of mental ray for Maya Standalone provides these additional features:

- The ability to be a master for an unlimited number of mental ray Standalone slave CPUs.
- The ability to accept rendering tasks from Maya or render `.mi` files offline.

You can purchase mental ray Standalone from Alias at [www.alias.com](http://www.alias.com).

## 4 | Other improvements

### > Improved Final Gather method

## Improved Final Gather method

The mental ray for Maya Final Gather processing method has been improved to produce better quality shading effects. As a result, lower settings can typically be used to reduce processing time. In addition, final gather now supports multiple diffuse bounces to achieve more realistic results (see "Secondary Diffuse Bounces" on page 126).

We've also made other improvements to Final Gather including:

- "New object-specific render attributes in the Attribute Editor" on page 117
- "New Final Gather settings" on page 125

## Tips to retune scenes with the new Final Gather method

The following tips may help you retune your final gather settings to work with the new final gather method. These tips may also improve performance and quality in your scenes.

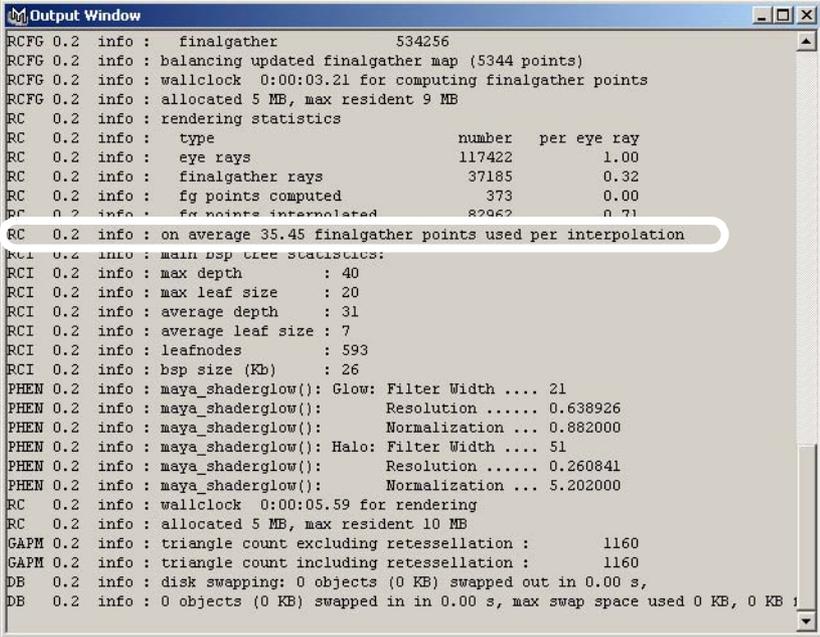
- Adjust Final Gather Rays  
Because the new final gather method is more efficient, you should use significantly fewer final gather rays. Typically, only one tenth of the original number of final gather rays are now required. You can adjust Final Gather Rays in the Render Global Settings window, mental ray tab, Final Gather section.
- Adjust Final Gather radii  
If the mental ray for Maya statistics on final gathering show more than 30 to 50 final gather points per interpolation (of the Min Radius and Max Radius), you should reduce the radii size to improve performance.

## 4 | Other improvements

### > Improved Final Gather method

#### To see statistics and adjust final gather radii

- 1 In the Render Global Settings window, mental ray tab, Translation section, select Info Messages from the Export Verbosity drop-down list.
- 2 Render your scene.
- 3 In the Maya Output window check for the final gather points statistic:



```
Output Window
RCFG 0.2 info : finalgather 534256
RCFG 0.2 info : balancing updated finalgather map (5344 points)
RCFG 0.2 info : wallclock 0:00:03.21 for computing finalgather points
RCFG 0.2 info : allocated 5 MB, max resident 9 MB
RC 0.2 info : rendering statistics
RC 0.2 info : type number per eye ray
RC 0.2 info : eye rays 117422 1.00
RC 0.2 info : finalgather rays 37185 0.32
RC 0.2 info : fg points computed 373 0.00
RC 0.2 info : fg points interpolated 82962 0.71
RC 0.2 info : on average 35.45 finalgather points used per interpolation
RCI 0.2 info : main bsp tree statistics:
RCI 0.2 info : max depth : 40
RCI 0.2 info : max leaf size : 20
RCI 0.2 info : average depth : 31
RCI 0.2 info : average leaf size : 7
RCI 0.2 info : leafnodes : 593
RCI 0.2 info : bsp size (Kb) : 26
PHEN 0.2 info : maya_shaderglow(): Glow: Filter Width .... 21
PHEN 0.2 info : maya_shaderglow(): Resolution ..... 0.638926
PHEN 0.2 info : maya_shaderglow(): Normalization ... 0.882000
PHEN 0.2 info : maya_shaderglow(): Halo: Filter Width .... 51
PHEN 0.2 info : maya_shaderglow(): Resolution ..... 0.260841
PHEN 0.2 info : maya_shaderglow(): Normalization ... 5.202000
RC 0.2 info : wallclock 0:00:05.59 for rendering
RC 0.2 info : allocated 5 MB, max resident 10 MB
GAPM 0.2 info : triangle count excluding retessellation : 1160
GAPM 0.2 info : triangle count including retessellation : 1160
DB 0.2 info : disk swapping: 0 objects (0 KB) swapped out in 0.00 s,
DB 0.2 info : 0 objects (0 KB) swapped in in 0.00 s, max swap space used 0 KB, 0 KB
```

- 4 Adjust the Min Radius and Max Radius in the Render Global Settings window, mental ray tab, Final Gather section, if necessary.

## 4 | Other improvements

### > New Rapid Scanline method

## New Rapid Scanline method

The mental ray for Maya Rapid Scanline rendering method has been optimized for use with motion blur and fur. It uses a new algorithm to achieve finer shading and improve performance. To tune the new Rapid Scanline method, we've added the following new settings:

### Visibility Samples

This value indicates the number of samples used for anti-aliasing. The default value is 0, and the maximum value is 8. A value of 0 defaults to the mental ray core value (4).

### Shading Quality

This value indicates the number of shading samples per image pixel. The default value is 1, and the maximum value is 4.

You can find these new settings in the Render Global Settings window, mental ray tab, in the Raytracing section.

You can also use new quality presets for Rapid Scanline, see "New Quality Presets" on page 118 for more information.

### Note

Rapid scanline should not be used for scenes that require ray tracing or final gathering.

## New shader library

A new shader library called `subsurface.mi`, enabling subsurface scattering is available. To see the contents of this library, click the information (i) button in the Shader Manager. This library is loaded by default.

The `subsurface.mi` library includes fast, non-physical subsurface scattering shaders (*misss\_fast\_\**), and physically correct subsurface scattering shaders (*misss\_physical*).

## 4 | Other improvements

### > Improved displacement approximation in mental ray for Maya

The physically correct subsurface scattering shaders provide physically accurate results by using true volumetric computations. They work better in scenes where photon tracing is required.

The non-physical subsurface scattering shaders do not provide physically accurate results, but instead provide quick, consistent, and visually pleasing results. The non-physical subsurface scattering shaders provide a way to efficiently render human skin, especially for shallow (near surface) scattering.

For more information, see the *mental ray Shaders Guide* available from the Maya help.

## Improved displacement approximation in mental ray for Maya

mental ray for Maya's handling of displacement approximation settings (derived from Maya), has been improved in the following ways:

- Extreme values in Maya now result in high-quality fine displacement in mental ray.
- Common values in Maya now result in fast fine displacement with reasonable quality.
- Standard (non-feature based) displacement is now supported.

### Note

Some scenes may render differently due to the above improvements. To achieve the desired results, turn on feature based displacement on the shape nodes.

## 4 | Other improvements

### > Improved approximation on Subdivision surfaces

## Improved approximation on Subdivision surfaces

mental ray for Maya now uses a single approximation value for base surface tessellation, and displacement tessellation on subdivision surfaces. As a result, some scenes may render differently.

## New Render Stats for image-based lighting (IBL)

We've added the following new Render Stats (previously called Visibility) for IBL.

### Visible as Environment

When on, the surface is visible to environment rays (direct reflection of the environment). This attribute is off by default, unless you created the IBL node from the mental ray Render Global Settings window.

### Visible in Secondary Reflections

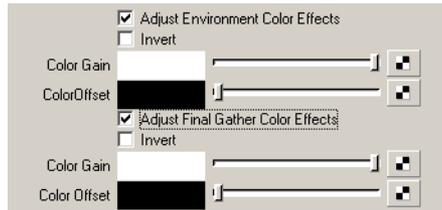
When on, the surface is visible to secondary reflection rays. This attribute is off by default, unless you created the IBL node from the mental ray Render Global Settings window.

## New image-based lighting (IBL) attributes

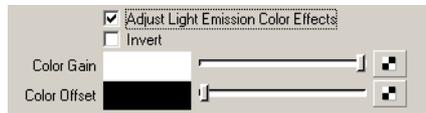
Use the new IBL attributes to adjust the environment, final gather, light emission, and photon emission Color Gain and Color Offset contributions. These attributes are off by default.

Modify environment color effects and final gather color effects in the Attribute Editor for the IBL shape node, in the Render Stats section.

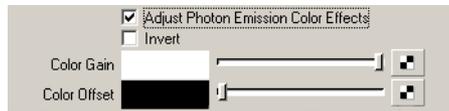
## 4 | Other improvements > Improved Baking



Modify the light emission color effects in the Attribute Editor for the IBL shape node, in the Light Emission section.



Modify the photon emission color effects in the Attribute Editor for the IBL shape node, in the Photon Emission section.



## Improved Baking

We've made the following baking improvements:

- To better represent the functionality, we've renamed Incoming Illumination to Light Only mode. The Light Only mode is supported directly from a shader so it doesn't have the overhead of assigning a Lambert material and recovering an existing shading network.
- A new Occlusion color mode is now available, including the following attributes that let you bake ambient occlusion:
  - Occlusion Rays

## 4 | Other improvements

### > New light and texture shaders

This attribute determines the number of occlusion rays to trace per sample point. Increasing the number of occlusion rays improves quality, but reduces performance speed. The default is 64.

- Occlusion Falloff

This attribute determines the maximum length of an occlusion ray. Rays longer than this value are not considered for occlusion.

For texture bake-sets, if Final Gather is not used, occlusion is computed for all sample points. Although this takes time, the result is very sharp light maps. If Final Gather is used, and the final gather quality is greater than 0, the occlusion is pre-baked into a final gather map. This final gather map can then be interpolated during rendering providing quick results at a reasonable quality.

- Vertex baking is improved by the introduction of the new Optimize Vertex Sharing option.

Optimize Vertex Sharing ensures that there is no duplication of vertices, and improves the quality of baking to vertices in many cases. This option is turned on by default in the Render Global Settings window, mental ray tab, in the Translation > Performance section.

## New light and texture shaders

The following new shaders are available:

- Ambient Occlusion shader (mib\_amb\_occlusion)

This shader is used to calculate how ambient light affects the scene. Use it to add more realism in a more efficient way than traditional global illumination.

- Blackbody light utility shader (mib\_blackbody)
- CIE D Illuminant light utility shader (mib\_cie\_d)

## 4 | Other improvements

### > New Dynamic Attributes

Use these shaders in addition to other light shader nodes to mimic real lights. You can find these shaders in the Hypershade, mental ray node, Textures section or Lights section. For more information, see “Base Shaders” in the *mental ray Shaders Guide* in the Maya Help.

## New Dynamic Attributes

These new dynamic attributes are Maya-specific, and are designed to work in conjunction with custom shaders. Some of these attributes are not available in the Maya user interface. Once you have created the attribute, you can tweak the value in the Extra Attributes section in the Attribute Editor.

### Export Shadow Shader

Use this attribute to have transparent shadows even when photons are used. The attribute functions locally and overrides the `shadowEffectsWithPhotons` attribute (the Direct Illumination Shadow Effects setting in the Render Global Settings), on a per-material basis, so that shadow shaders are exported even when photons are turned on

Create this attribute as follows:

```
addAttr -ln miExportShadowShader -at bool phong1SG;
```

### Force Displacement

For a description of this attribute, see “New Displacement Overrides setting” on page 126.

Create this attribute as an override on the shape node as follows:

```
addAttr -ln miMaxDisplace -at "float" pSphereShape1
```

### Disable Animation Detection

For a description of this attribute, see “Animation detection” on page 58.

Create this attribute as an override on the shape node as follows:

## 4 | Other improvements

### > New Dynamic Attributes

```
addAttr -ln miAnimated -at bool group1;
```

#### Disable DG Cycle Detection

Use this attribute to turn off the detection of cycles in the dependency graph when very complex shading networks are used, to improve performance. Cycles in shading networks may cause mental ray for Maya to become unstable in some cases. Set this attribute to false to disable the DG cycle detection.

Create this attribute as follows:

```
addAttr -at bool -ln "nodeCycleCheck" mentalrayGlobals
```

#### Force On-demand Translation of Geometry

For a description of this attribute, see "Improved on demand translation" on page 128.

Create this attribute as an override on the shape node as follows:

```
addAttr -ln "miPlaceholder" -at bool pSphereShape1
```

#### Custom Phenomenon for Translation

Use this attribute to attach a replacement material, usually a material phenomenon node, to replace the current shading group upon translation. This attribute lets you use more complex phenomena.

Create this attribute as follows:

```
addAttr -ln "miMaterial" -at message shadingGroup1
```

Connect this attribute as follows:

```
connectAttr customPhenomenon.message shadingGroup1.miMaterial
```

## 4 | Other improvements

### > mental ray for Maya support of native Render Stats

#### Disable Custom Node Automatic Light Linking

Use this attribute to turn off automatic light linking on a per-node basis (set the attribute to false). This is useful when the same material is used by several objects that have different light linking. In this case, disable light linking with this attribute, then manually connect the light nodes to the shader's light attributes.

Create this attribute as follows:

```
addAttr -ln "miLightLink" -at bool mib_illum_lambert1
```

#### Per-light Shadow Map Bias

Use this attribute to specify shadow map bias on a per-light basis when the mental ray for Maya Shadow Map attribute (in the Attribute Editor > mental ray > Shadow Maps section for the light) is turned on.

Create this attribute as follows:

```
addAttr -ln "smapBias" -at "float" pointLightShape1
```

## mental ray for Maya support of native Render Stats

mental ray for Maya now supports all Render Stats attributes with the exception of the Smooth Shading attribute. In particular, Visible in Reflection, Visible in Refraction, and Visibility are now supported. The Render Stats attributes are located in the Attribute Editor, Render Stats section, for the shape node.

For more information, see *"Render Stats"* in the *Rendering Guide*.

## Texture bake-set Final Gather Quality

For Texture bake-sets, the maximum value for the Final Gather Quality attribute has been increased to 2. The default value is 1.

## 4 | Other improvements

### > New Vertex bake-set functionality

## New Vertex bake-set functionality

Vertex bake-sets have been enhanced with the introduction of new attributes and improvements to existing attributes.

### Vertex bake-set attributes

#### Bake Color (new)

Turn this attribute off when you do not want to bake the color channel. When this option is on, you can set Min Color and Max Color values, and the Color Blending method. This attribute is on by default.

#### Bake Alpha (improved)

Turn this option on to bake the alpha channel (equivalent to Maya's Bake Transparency option). When this option is turned on, you can set Min Alpha and Max Alpha values, and the Alpha Blending method. This attribute is off by default.

#### Clamp Min and Clamp Max (improved)

Turn on these attributes to clamp the minimum and maximum color and alpha values so that the values are forced to be within the set range.

#### Min Alpha (new)

The lower clamp limit for the alpha channel.

#### Max Alpha (new)

The upper clamp limit for the alpha channel.

#### Occlusion Rays and Occlusion Falloff (new)

For more information about these attributes see "Improved Baking" on page 111.

## 4 | Other improvements

### > New object-specific render attributes in the Attribute Editor

#### Color Blending (improved)

Merges existing vertex colors with the ones just baked, if any.  
Select a merge method from the Color Blending drop-down list.

#### Alpha Blending (improved)

Merges existing vertex alphas with the ones just baked, if any.  
Select a merge method from the Alpha Blending drop-down list.

## New object-specific render attributes in the Attribute Editor

In the Attribute Editor, mental ray section, the following new attributes are now available:

#### Visible In Transparency

This attribute indicates whether the object shows through other transparent objects (if ray depth has not reached its limit). This attribute is on by default.

When this attribute is turned off, the object becomes invisible behind transparent objects.

#### Transmit Transparency

This attribute indicates whether the object receives transmitted transparent rays from other objects. This attribute is on by default.

When this attribute is turned off, the object becomes opaque or does not show transparency effects.

#### Final Gather Cast

This attribute indicates whether the object casts final gather rays, allowing the object to contribute to the final gather lighting in the scene. This attribute is on by default.

## 4 | Other improvements

### > New Quality Presets

#### Final Gather Receive

This attribute indicates whether the object receives final gather rays. When this attribute is turned off, the object is not affected by final gathering. This attribute is on by default.

**Note** The above attributes are only available on the shape node of NURBS, Subdiv, Mesh objects, and Fur Feedback nodes.

#### Min Max Object Sample Limits

Use this attribute to set specific sample limits on a per-object basis. The values you select must be within the global sample limits (which are available in the Render Global Settings window). The default Min Sample Level value is 0, and the default Max Sample Level value is 2. The Min Max Object Sample Limits attribute is off by default.

## New Quality Presets

We've added new quality presets for mental ray for Maya in the Render Global Settings Window, mental ray tab, Quality Presets dropdown list.

#### Custom

Lets you specify the mental ray for Maya quality settings independently.

#### DraftRapidMotion

Gives you a relatively good indication as to what the scene will look like, when using Rapid Scanline, while taking the least amount of processing time.

## 4 | Other improvements

### > New Caustics and Global Illumination settings

#### PreviewRapidMotion

Slightly better than Draft (when using Rapid Scanline), takes a little more processing time, but achieves a good balance between quality and time.

#### ProductionRapidMotion

Use this when testing the final rendering, or rendering the final image(s) when using Rapid Scanline.

For more information, see "New Rapid Scanline method" on page 108.

## New Caustics and Global Illumination settings

The following new settings are available for caustics and global illumination in the Render Global Settings window, mental ray tab, Caustics and Global Illumination section:

#### Caustic Scale

Use this setting to control the influence of indirect illumination effects for caustics. You can select a color with the Color Chooser or use the slider to set the Caustic Scale value.

#### Global Illum Scale

Use this setting to control the influence of indirect illumination effects for global illumination. You can select a color with the Color Chooser or use the slider to set the Global Illum Scale value.

These settings are off by default.

## New Anti-aliasing Quality settings

A new Sample Defaults section, containing the following new settings, is available in the Render Global Settings window, mental ray tab:

## 4 | Other improvements

### > New Memory and Performance option

#### Min Object Samples

This setting specifies the minimum number of samples taken for objects. This value is clamped by global sample settings, so that if the Min Object Samples value exceeds the global sample settings, it is ignored. This lets you keep global sample settings low. The default value is -128.

#### Max Object Samples

This setting specifies the maximum number of samples taken for objects. This value is clamped by global sample settings, so that if the Max Object Samples value exceeds the global sample settings, it is ignored. This lets you keep global sample settings low. The default value is 127.

#### Volume Samples

This setting specifies the default value for the number of volume samples for any volume effects in Maya shaders. The default value is 1.

These settings can also be adjusted on a per-object basis. This is useful in improving rendering speed by increasing anti-aliasing quality only where necessary.

For more information, see the *mental ray User Manual*, available from the Maya Help.

## New Memory and Performance option

The Separate Shadow Bsp option is now available in the Render Global Settings window, mental ray tab, Memory and Performance section. This option enables mental ray for Maya to use a secondary Bsp tree for objects that have low-detail shadow stand-ins to improve performance.

## New Translation Performance options

The following new options are available in the Render Global Settings window, mental ray tab, Translation > Performance section.

### Optimize Non-animated Display Visibility (previously Prune Invisible Parts)

This option ignores non-animated invisible scene entities during translation so that they are not part of the final rendered scene. This option is on by default.

### Prune Objects Without Material

This option ignores objects without materials during translation so that they are not part of the final rendered scene. This option is on by default.

### Export Tessellated Polygons

This option translates all polygon meshes as triangulated in Maya. This allows more efficient use of memory so that larger scenes with many polygons render with less memory usage.

This option is off by default.

### Smooth Polygon Derivatives

This option calculates derivatives by taking into account vertex sharing to decide if texture seams can be ignored. This calculation avoids artifacts due to UV seams.

This option is off by default.

## New Translation Customization options

The following new options are available in the Render Global Settings window, mental ray tab, Translation > Customization section.

## 4 | Other improvements

### > New Preview settings

#### Render Shaders With Filtering

This option causes mental ray shaders to perform filtering in individual shaders. This option reduces texture and bump mapping artifacts, and is on by default.

Render Shaders With Filtering requires that the Export Polygon Derivatives option be turned on (Render Global Settings window, mental ray tab, Translation > Performance section).

#### Optimize Raytrace Shadows

This option optimizes the algorithm used to assign mental ray shadow shaders to materials. When turned on, mental ray determines whether shadow shaders are necessary before assigning the shader. This option is on by default.

## New Preview settings

The following new settings are available in the Render Global Settings window, mental ray tab, Preview section.

#### Preview Convert Tiles

This option enables a plug-in conversion procedure for previewing image tiles when the primary framebuffer type is of a type that is not supported by Maya's Render View (for example, floating point framebuffers, or color framebuffers with components larger than 8 bit).

The conversion procedure involves the clipping and rescaling (desaturation) of colors for display. This option is on by default.

#### Preview Tonemap Tiles and Tonemap Scale

The Preview Tonemap Tiles option previews image tiles for floating-point framebuffer types where the color values exceed the typical 0-1 range (because the color values are RGB). This option is on by default.

## 4 | Other improvements

### > New Shadows settings

The Tonemap Scale value is used to rescale the color data of image tiles prior to clipping. This is useful when rendering high dynamic range images. The final rendered image is not affected by this value. The default value is 1.

## New Shadows settings

The following new settings are available for shadows in the Render Global Settings window, mental ray tab.

### Detail (Shadow Maps drop-down list)

The Detail shadow map option is a combination of features from standard shadow maps and raytraced shadows, meaning that it collects more information about shadow-casting objects.

Unlike the standard shadow map, a detail shadow map also takes into account surface and lighting properties, such as transparency. Detail shadow maps store a list of depth values together with the light transmission coefficients at each depth. This provides similar quality shadows as raytraced shadows, but at processing times similar to depth map shadows.

Detail shadow maps may require more time to calculate because they compute and store more per-pixel information.

#### Note

Detail shadow maps are more sensitive to the Softness attribute (in the Attribute Editor, Shadow Maps section for the light shape node). A large Softness value results in a penumbra spread well beyond the shadow area.

## 4 | Other improvements

### > New Motion Blur Calculation settings

#### Merge (Rebuild drop-down list)

The Merge option specifies that shadow maps should be loaded from files, if available. The default shadow map calculations are still performed and the existing points are overwritten by the recomputed points, but only if the new points are closer to the light source.

This option is useful for building shadow maps for use in multi-pass rendering because it allows shadow maps from a previous render pass to be reused for the current pass. Only shadow map changes are recalculated, not the entire shadow map.

#### Shadow Map Bias

This option applies the specified Shadow Map Bias value to all light sources that do not have their own biases. This adds a slight offset to the shadow depths, resulting in a slightly shifted shadow. This option is useful in tuning shadows in specific cases, such as when rendering Fur.

The bias value should be smaller than the smallest distance between a shadow caster and a shadow receiver. However, bias values that are too small may cause self-shadowing.

See also, "Per-light Shadow Map Bias" on page 115.

For more information, see the *mental ray User Manual*, available from the Maya Help.

## New Motion Blur Calculation settings

The following new options are available in the Render Global Settings window, *mental ray* tab, Motion Blur > Calculation section:

- Custom Motion Offsets

## 4 | Other improvements

### > New Final Gather settings

Turn on this option if you want to set values for Motion Back Offset and Static Back Offset. Use these custom motion offsets to define the time steps where motion blur information is captured. This option is off by default.

- **Motion Back Offset**  
This value determines the start point of the time interval used for motion blurring. It is an offset to the current time in frames. The default value is 0.5, and corresponds to Maya.
- **Static Object Offset**  
This value determines the time used to render static objects. The default value is 0, and corresponds to Maya.

## New Final Gather settings

The following new settings are available for final gather in the Render Global Settings window, mental ray tab, Final Gather section:

### View (Radii in Pixel Size)

This option causes the Min Radius and Max Radius of final gather rays to be calculated in pixel size, rather than in object space. This allows you to set the visual quality in pixel size, without knowing the object or scene bounds.

### Scale

The Scale value allows you to easily control the intensity and color of the final gather contribution on a global scene level. You can use the Color Chooser or use the slider to set the Scale value.

### Freeze (Rebuild Final Gather drop-down list)

The Freeze option stops any new data from being written to the final gather file. This option causes mental ray for Maya to compute the final gather solution just once for a full sequence of animation. It is useful to reduce light flickering in your animation.

## 4 | Other improvements

### > New Diagnostics option

#### Secondary Diffuse Bounces

Turn on the Secondary Diffuse Bounces option to allow multiple diffuse bounces for final gathering. This option controls whether indirect diffuse lighting contributes to final gather, up to a sum of the Final Gather Reflection and Final Gather Refraction values. This option is off by default.

For more information, see *“Render Global Settings: mental ray tab”* in the *Rendering* guide.

#### New Diagnostics option

The Diagnose Finalgather option (Render Global Settings window, mental ray tab, Diagnostics section) renders final gathering points in green for initial raster-space, and in red for render-time final gathering points. This is useful in fine tuning final gather settings to distinguish between view dependant and non-view dependant results to better distribute final gather points. This option is off by default.

#### New Displacement Overrides setting

The Max Displace setting (Render Global Settings window, mental ray tab, Overrides > Displacement section) specifies the maximum displacement applied to object control points in a normal direction. This provides control over the otherwise automated displacement range to better focus tessellation where most needed. Set this value if you have any displaced objects in your scene.

A Max Displace value that is too large results in a correct image, but takes more time and uses more memory. If the Max Displace value is too small, parts of the displaced object may be clipped. The default value of 0 means the setting is not active.

A warning message appears if a displacement shader returns a value greater than the Max Displace value. This can result in rendered geometry appearing clipped.

## New Custom Entities options

There are new Custom Entities (Render Global Settings window, mental ray tab) attributes designed to improve compositing for Maya post-effects:

### Pass Custom Alpha Channel

This option passes the mental ray alpha component of the final color as the alpha channel, ignoring the Maya alpha component. This is useful when a custom shader is producing an alpha value. This option is off by default.

### Pass Custom Depth Channel

This option overrides the Maya depth channel calculation with the default mental ray depth channel calculation. This option is useful when you want to revert to using the mental ray depth calculation, instead of the default Maya calculation. This option is off by default.

### Pass Custom Label Channel

This option passes label data untouched, rather than allowing adjustment for Maya shaders. This option is off by default.

Use these options to take advantage of alternate channel computations when writing custom shaders.

For more information, see “Render Global Settings: mental ray tab” in the *Rendering* guide, or see the *mental ray Shaders Guide*, available from the Maya Help.

## New support for region rendering

mental ray for Maya batch rendering now supports user-defined region rendering. Use the new `-reg` flag, for example:

```
render -r mr -reg 0 100 0 100 scene.ma
```

## 4 | Other improvements

### > New support for Area Lights

where `-reg 0 100 0 100` indicates the region to be rendered in pixels (left, right, bottom, top).

The above command renders the lower left 100 x100 pixel region of the scene.

## New support for Area Lights

The Maya Area light is now supported by mental ray for Maya rendering.

## Improved on demand translation

To improve mental ray for Maya's on demand translation, the new Export Objects On Demand option is available in the Render Global Settings window, mental ray tab, in the Translation > Performance section. This option is off by default.

Use the Export Objects On Demand option to control the processing of objects in your scene. This option is particularly effective in scenes that have objects beyond the view of the camera. In this case, mental ray does not process the objects beyond the camera view, therefore reducing processing time.

When Export Objects On Demand is on, it includes a Threshold value that lets you tune on demand translation for objects. Objects with a number of vertices or controlled vertices greater than the threshold value are not translated until a ray hits the bounding box. A value of 0 results in all objects processed on demand. In this case, translation is quick, but render time may not improve. You can select larger objects for on demand translation by raising the Threshold value.

#### Note

This option is not supported for Fur.

## Improved Custom Scene Text

The following new custom entities are available for custom scene text for exported .mi files:

### Root Group Text

The Root Group Text control can be used to customize the final root group specifying all cameras and objects to be rendered. This control is applied to the content of the root group, not the whole root group section (like other types of scene text).

### Render Text

The Render Text control can be used to customize render commands for renderable cameras. For example, it can be used to perform operations between renderings, like file operations.

These custom entities should only be used by advanced mental ray for Maya users. For more information, see “*Render Global Settings: mental ray tab*” in the *Rendering* guide, or see the *mental ray Shaders Guide*, available from the Maya Help.

## Extension to mental ray shading language

A new flag, `photonvol`, is available for shader `apply` statements. The `photonvol` flag indicates a photon volume shader in a material or camera. This flag is used to write custom shaders or phenomena, and is available in the .mi file that describes your shader. For more information, see the *mental ray User Manual*, available from the Maya Help.

## New file texture filter types

You can now use Quadratic, Quartic, and Gaussian filters on 2D file textures when rendering with mental ray for Maya.

## 4 | Other improvements

### > New Framebuffer Settings

In previous releases of Maya, these texture filtering methods were automatically converted to mipmap filtering. Older scenes may render differently now. To revert to the previous behavior, you should explicitly edit all the file texture nodes and set the filtering method to mipmap.

For more information on file texture filters, see *"File"* in the *Shading* guide.

## New Framebuffer Settings

We've made the following improvements to Framebuffer settings for mental ray for Maya:

### New Framebuffer options

The Framebuffer options are now organized into two sections: Primary Framebuffer and User Framebuffer.

The Primary Framebuffer section contains existing attributes, including Data Type, Gamma, and Colorclip.

Use the new User Framebuffer options to select one or more user-defined framebuffers for use with your custom shaders. You can use up to eight user framebuffers.

To enable a particular framebuffer, click the corresponding checkbox.

### Improved preview rendering

You can now preview render 16-bit and 32-bit framebuffer types.

## New attribute for Texture Bake-sets

The Final Gather Reflect attribute is available for texture bake-sets. This attribute determines the reflectivity of an object when precomputing final gather points for light mapping. This simplifies the simulation of reflective objects whose texture maps include

## 4 | Other improvements

### > New attribute for Use Background shader

contributions from objects that surround them. For example, if the Final Gather Reflect value is set to 0.25, every fourth final gather point is precomputed on the object hit by the reflection ray.

## New attribute for Use Background shader

The Use Background shader now includes a new mental ray attribute called Occlusion Mask. Use the Occlusion Mask attribute to see the soft shadow, produced by final gathering, of an object on a surface that uses the Use Background shader. These soft shadows are also captured in the alpha channel of the image for compositing purposes.

You can set values for Occlusion Mask between 0 (no shadows) to 1 (full shadows), or map the attribute.

## New image output formats

When rendering with mental ray for Maya, you can now save rendered images in PNG (.png) and openEXR (.exr) formats. In the Render Global Settings window, Common tab, in the Image File Output section, select PNG (.png) or openEXR (.exr) from the Image Format drop-down list.

## IPR (Interactive Photorealistic Rendering) improvements

We've improve IPR in the following ways:

- mental ray for Maya's IPR now supports transformation (orientation, moving, and scaling) of lights, and cameras.
- You can now tweak attributes in the shader glow node to see immediate updates in IPR. The scene is not re-rendered, only post-process updates occur.

For more information on IPR, see "*Interactive Photorealistic Rendering (IPR)*" in the Maya Help.

## 4 | Other improvements

### > Support for renaming per particle fields

## Dynamics

### Support for renaming per particle fields

There is now improved renaming support for per particle fields. Specifically, the per particle field attribute automatically updates if the attribute is renamed and the per particle field attribute now works with file referencing.

### Specifying which particle systems to cache

We've added an option to Solvers > Create Particle Disk Cache >  to allow you to specify to cache either *Selected* or *All* particle systems. If *Selected* is specified, then only visible (non-intermediate) particle systems are cached.

## Painting

### Artisan improvements

When painting polygons with Artisan tools, there is now added support for Show > Isolate Select and Shading > Backface Culling. In previous versions of Maya, the hidden areas in both these cases used to get painted. Now the hidden areas are not affected by painting with Artisan Tools.

### 3D Paint Tool improvements

We have added support for Show > Isolate Select and Shading > Backface Culling. In previous versions of Maya, the hidden areas in both these cases would be painted. Now the hidden areas are not affected by painting with the 3D Paint Tool.

## **Maya Unlimited**

### **Fluid Effects improvements**

#### **outFoam attribute**

The outFoam attribute now works on the Ocean Shader node. This allows mapping of foam for various uses. outFoam is the output foam density value computed by the texture. Use this value to control the density of a layered foam shader. If applied to transparency, the value should be first inverted. (The outFoam attribute is not in the Ocean Shader node's Attribute Editor, but can be connected to using the Connection Editor.)

#### **New Solver Quality attribute**

We've added a new Solver Quality attribute to the fluid shape node (Dynamic Simulation section). Increasing the Solver Quality value increases the number of steps used internally by the Navier-Stokes solver, which may increase the accuracy of the simulation and certainly increases the time required to run. Depending on your fluids material properties, you may get satisfactory results lowering this from the default value of 20.

### **Fur improvements**

#### **Fur equalizer maps improvements**

We've redesigned the handling of equalizer maps, and corrected some known issues:

- We've added the ability to specify the frame number at which the equalizer map is generated.
- We removed the browse button of ReadEqualMap Path; the Custom Equalizer workflow is now used to read maps generated elsewhere. This ensures that the maps are always found correctly.

## 4 | Other improvements

### > Fur improvements

#### Fur equalizer map options

In the Fur Globals window there are now only three options for equalizer maps:

- No Equalizer Maps
- Default Equalizer Maps
- Custom Equalizer Maps

A default equalizer map is created at the frame specified in *Use Frame*. It is used throughout a sequence, even if the fur is rendered in sections or over a render farm. If Default Equalizer Maps is selected, at render time Maya searches for maps with the correct name and frame extension defined in *Use Frame*. If no default equalizer maps exist, then they are created; if they exist, the time stamp is checked against the scene time stamp. If the scene is newer, the maps are rewritten.

The new Default Equalizer Map option only ever reads a default equalizer map, and not one generated somewhere else. If you want to use a map that has been generated or edited elsewhere, you should select the Custom Equalizer Map option.

The behavior of the Custom Equalizer Map option has not changed, except that the *Use Frame* number is used to initially generate the maps. You should use custom equalizer maps in the following situations:

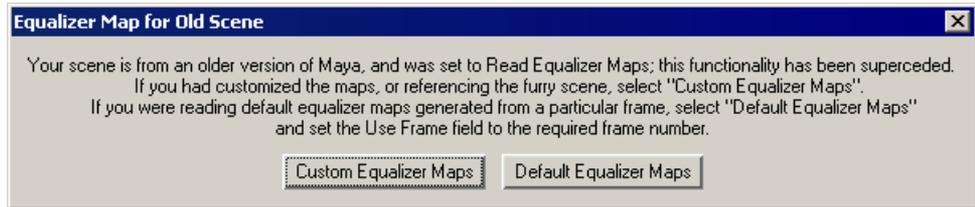
- if you want to edit the maps in any way, such as painting them in the Paint Fur tool or in an external package;
- or if you are referencing the furry model from another project and want to use the maps generated in the original project.

#### Support for fur scene files from Maya 6.0 and earlier

When you open an old scene with read equal maps, you are prompted with the following dialog box:

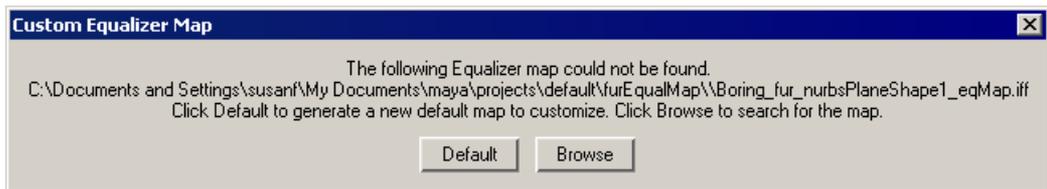
## 4 | Other improvements

### > Hair improvements



If you select Default Equalizer Maps, Fur Globals are set to default equalizer. If you select Custom Equalizer Maps and the map is found, old *read equal* maps will be converted to custom equalizer maps.

If the map is not found, you are prompted with the following dialog box:



Select Default if you want to generate a default equalizer map. Otherwise, select Browse to look for your own custom equalizer map.

## Hair improvements

- 2D Motion blur now works on hair when shadow casting is turned on. Previously it only worked with hair when Cast Shadows was turned off.
- 2D Motion blur now works correctly for hair when the number of hairSystem Sub Segments is greater than one. Previously a Sub Segments value greater than one resulted in overly long streaks at the hair tips.
- You can now use Motion Blur on self-shadowed hair.

## 4 | Other improvements

### > Cloth improvements

## Cloth improvements

- You can now visualize and animate the Cloth collision depth and offset. This allows users to better control the collision parameters, especially while painting. For more information, see “Visualizing Cloth collision object attributes” on page 137. In addition, you can now paint Collision Offset map values using the Paint Collision Properties Tool. See “Painting Cloth collision attributes” on page 138. For more information, see Cloth > Paint Collision Properties Tool in the Maya Help.
- We’ve added a new Cloth Collision Debug feature to the Cloth solver (cpSolver node) to allow you to diagnose problems with collision objects in complex production environments. For more information, see “Debugging Cloth collisions” on page 140.
- You can now edit membership of Cloth constraints using the new Constraints > Edit Constraint Membership Tool. Editing of membership is now possible, whereas in previous versions of Maya, you had to delete the constraint and recreate it again with the new component set. For more information, see “Editing Cloth constraints membership” on page 142.
- You can now use expressions to query and set vertices during simulation. See “Using expressions with Cloth” on page 142. This allows direct manipulation of a vertex and can be used to fix pinching problems.
- You can now animate the Cloth solver's Time Step Size (cpSolver node, Solver Attributes section). The timing control for cloth is now independent from Maya’s playback time. This allows the cloth to simulate faster or slower and still keep the same cloth property. As well, this allows you to control the step size over the course of a simulation; many steps need only be used where the simulation is particularly complex.
- We’ve added a new Time Scale attribute to the Cloth solver (cpSolver node, Solver Attributes section). The Time Scale attribute determines the relative time Cloth is running with

## 4 | Other improvements

### > Cloth improvements

respect to the Dependency Graph time. For example, if you want the cloth to run through the simulation twice as fast, Time Scale should be set to 2. The longer the Cloth's simulation time is, the more force has an effect on the Cloth. For example, if the force is gravity, the Cloth will appear to be heavier as the Time Scale increases. The default value is 1, which results in the same behavior as in previous versions of Maya. To reduce the solving time, you can animate the Time Step Size accordingly.

- You now have the option to allow animated collision geometry to block field constraints. This automates the occlusion detection for field constraints. Previously, users could only get this by applying the constraints to components. This became challenging if the collision object or the field was animated and the occlusion relationship changed over time.
- Simulation > Update Cloth State and the cpSolver command now automatically update the velocity based on the position from the previous frame and the new position from the current frame.

### Visualizing Cloth collision object attributes

You can visualize and animate a cloth object's collision depth and offset attributes. This helps to better control the collision parameters, especially when painting.

#### **To show the collision object's offset or depth mesh**

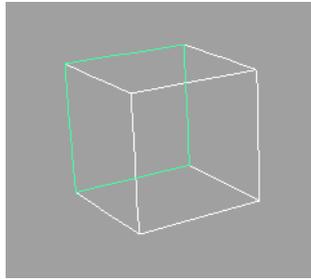
- 1** Select the collision object whose collision attributes you want to visualize.
- 2** In the Cloth menu set, select Cloth > Create Collision Offset/Depth Mesh.

A mesh appears around the collision object representing the Collision Offset value. A Collision Display drop-down list is added to the collision object attributes. Use this drop-down list to control which collision attribute is displayed: Depth, Offset, Depth

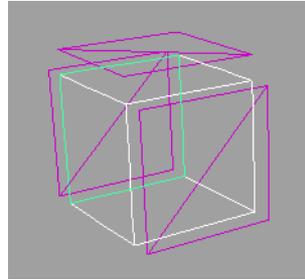
## 4 | Other improvements

### > Cloth improvements

and Offset, or None. For more information about collision attributes, see *“Adjust the collision object”* in the *Cloth* guide in the Maya Help.



Collision object without mesh



Collision object with mesh

### **To hide the collision object’s offset or depth mesh**

- 1 Select the collision object whose collision mesh you want to hide.
- 2 Select Cloth > Remove Collision Offset/Depth Mesh.

The mesh is removed from the collision object.

### **Painting Cloth collision attributes**

In Maya 6.0, we introduced the Cloth > Paint Collision Properties Tool feature, which can be used to paint the Collision Priority and Collision Depth map values for a collision object. In Maya 6.5 you can also paint Collision Offset map values for a collision object. To make it easier to see the effects of your painting, you can also use the new Cloth > Create Collision Offset/Depth Mesh menu item. The following are a couple of suggested workflows using these new features:

- “To paint Cloth collision attributes” on page 139
- “To tweak a Cloth collision simulation” on page 139

### **To paint Cloth collision attributes**

- 1** Select the Cloth collision object.
- 2** In the Cloth menu set, select Cloth > Create Collision Offset/Depth Mesh.  
A mesh appears around the collision object representing the Collision Offset value. To change the mesh display to Collision Depth, go to the Attribute Editor of the object's shape node and in the Extra Attributes section, select *depth* from the Display Collision drop-down list.
- 3** Select Cloth > Paint Collision Properties Tool > .  
The Paint Cloth Collision Properties Tool settings window appears.
- 4** Select the attribute you want to paint from the Paint Attribute drop-down list and then select a Value to paint.

#### **Tips**

Paint in Smooth Shaded display mode.

If painting Collision Depth, the mesh appears on the inside of the collision object. To make it easier to paint, go to the Attribute Editor of the collision object's shape node, in the Extra Attributes section and adjust the overall Collision Depth value, for example, to -3. Once you've finished painting, reset the Collision Depth value.

- 5** Paint the collision object and see the effects on the Offset/Depth mesh.

### **To tweak a Cloth collision simulation**

If a Cloth collision simulation is incorrect and vertices are sticking inside the collision object, you can use the following workflow to adjust the simulation results.

## 4 | Other improvements

### > Cloth improvements

- 1 Play the Cloth simulation.
- 2 Follow the steps in "To paint Cloth collision attributes" on page 139 to paint Collision Depth or Collision Offset values in the problem areas.
- 3 To clear the Cloth cache, select Simulation > Delete Cache.
- 4 Replay the Cloth simulation.
- 5 Repeat steps 2 to 4 until the Cloth collision simulation is corrected.

### Debugging Cloth collisions

Complex Cloth simulations that generate collision objects through the use of deformers or custom nodes can cause problems with the Cloth simulation when the object moves through the scene in an erratic or jumpy manner. If erratic motion occurs at the subframe level, you previously had no way to visualize and diagnose the problem. Now you can use the Cloth Collision Debug feature to troubleshoot. See "To debug a Cloth collision simulation" on page 141.

When you turn on Cloth Collision Debug and run the Cloth simulation, each collision object that was selected for debugging is displayed with a mesh. All of the generated meshes are parented under a common transform node called `CollisionObjectDebug_solvrname`. Under this transform node, transforms are created for each subframe being evaluated. These nodes are named `frame_frame_subframe`. Under these transforms, the actual meshes are created and named after the original collision object that they are based on.

The Frame Samples attribute in the cpSolver node determines how many meshes are created for each frame of the simulation.

## 4 | Other improvements

### > Cloth improvements

#### Note

When using the Cloth Collision Debug feature, groups are created each time the Cloth solver samples the shape of a collision object. This may result in additional groups being created that are slightly beyond the playback range. If the collision geometry is not animated beyond the last frame of the playback, these additional groups will be in the same position as the last frame. This is expected behavior. However as a workaround you can delete the groups you don't want.

Large amounts of meshes may be created using this feature. Collision Object Debug Enable is keyable and you are encouraged to only turn it on for problematic areas.

Once the simulation is run and the meshes are generated, they are visible, which may be confusing. To assist with visualizing all the meshes, there are new control buttons in the cpSolver node:

#### Show All, Hide All

Shows or hides all meshes.

#### First, Last, Previous, and Next

Use any of these to view the appropriate group of meshes and hide the previously viewed group (if any). First select Hide All (of the meshes) and then use these buttons to step through the subframes.

#### To debug a Cloth collision simulation

- 1 Open the Cloth scene.
- 2 Select the Cloth collision objects that seem to be causing the problem and then, in the Cloth menu set, select Cloth > Collision Object Debug > Add Selected Objects.

## 4 | Other improvements

### > Cloth improvements

- 3** In the cpSolver node, go to the Collision Object Debug section and turn on the Collision Obj Debug Enable attribute.
- 4** Do one of the following:
  - Select Simulation > Start Local Simulation to run the simulation.
  - Use the Collision Object Debug controls that are available in the cpSolver node to visualize the results.

The selected collision objects will be echoed as the solver evaluates them during simulation.

## Editing Cloth constraints membership

### To edit the set membership of a Cloth constraint

- 1** Select the Cloth constraint (in Outliner).
- 2** In the Cloth menu set, select Constraints > Edit Cloth Constraints Membership.
- 3** Do the following:
  - Shift+click to add items (for example, vertices, points).
  - Ctrl+click to remove items (for example, vertices, points).

## Using expressions with Cloth

### To use expressions with Cloth

- 1** In the Cloth cpSolver node, go to the Extra Attributes section and right-click the new Post Solver Callback attribute, then select Create New Expression from the pop-up menu.
- 2** Set the Evaluation option to be After Cloth.

## 4 | Other improvements

### > Cloth improvements

The other settings would force Maya to call this expression at times when the solver's internal state may not be valid, which might cause problems. If you're not getting the expected results, for example, the cloth starts bouncing around, it's likely because this option is not set correctly.

#### 3 Now you're ready to write your expression.

We've added two new commands, `cpGetSolverAttr` and `cpSetSolverAttr`. You can use the `Get` function to retrieve information for a single vertex, or for every vertex at once. Currently you can specify one of three data types: position, velocity, or stress. For example:

```
float $pos[] = `cpGetSolverAttr -attribute "position" "cloth1"`;  
float $v5[] = `cpGetSolverAttr -a "velocity" -index 5 "cloth1"`;
```

You can operate on the values returned from the `Get` functions in any way you like. To return them to the solver you must return information for each vertex individually, and the index and new values must be supplied. For example:

```
cpSetSolverAttr -a "velocity" 5 newVx newVy newVz
```

The `Set` function currently supports "position" and "velocity".

#### 4 The final step is to connect the new expression to the `clothSolver` node. Do this by adding the line `postSolverCallback = true` to the expression.

For example, the following expression clamps a piece of cloth to -3 along the Y-axis.

```
// Get the current positions. These are directly from the solver float  
$pos[] = `cpGetSolverAttr -a "position" "cloth1"`;  
int $numPts = size($pos)/3;  
// Check each point  
for ($i=0; $i<$numPts ; $i++ )  
{  
    int $idx = $i*3;  
    if ( $pos[$idx+1] < -3 )  
    {  
        // Set the new value into the solver
```

## 4 | Other improvements

### > Maya browser can get data from Maya

```
        cpSetSolverAttr -a "position" $i $pos[$idx] -3 $pos[$idx+2]
"cloth1";
    }
}
// Trigger to ensure that this expression is called immediately after the
// solver has completed
// its calculations for the current frame
postSolverCallback = true;
```

## Maya Web browser

### Maya browser can get data from Maya

We have implemented a feature that allows you to obtain data from Maya using scripting languages such as JavaScript in the Maya Web browser. The following example shows how this might be done.

The current IDL interface is:

```
interface nsIMEL : nsISupports
{
    boolean          ExecuteCommand(in string command, in boolean undoable,
in boolean display);
    // Return type of the last command executed through
    // ExecuteCommand.
    // Value:        Type:
    // 1             int
    // 2             int array
    // 3             double
    // 4             double array
    // 5             string
    // 6             string array
    //
    long             GetResultType();
    // Basic return types.
    //
    string          GetStringResult();
    long            GetIntResult();
    double          GetDoubleResult();
}
```

What's New in Maya 6.5

## 4 | Other improvements

### > Maya browser can get data from Maya

```
// Array return types.
//
unsigned long   GetArrayResultSize();
string         GetStringArrayIndexResult (in unsigned long index);
long           GetIntArrayIndexResult   (in unsigned long index);
double         GetDoubleArrayIndexResult (in unsigned long index);
void           GetStringArrayResult (out PRUint32 count, [retval, array,
size_is(count)] out string result);
void           GetIntArrayResult   (out PRUint32 count, [retval, array,
size_is(count)] out long result);
void           GetDoubleArrayResult (out PRUint32 count, [retval, array,
size_is(count)] out double result);
};
```

The following example demonstrates the use of the Web browser getting data from Maya.

```
Example: var imel;
// this line is only required if this is a web page
netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
imel = Components.classes["@alias.com/CommandEngine/MEL;1"].getService();
imel = imel.QueryInterface(Components.interfaces.nsIMEL);

// create a sphere, make the command undoable, and show results in the
script editor
imel.ExecuteCommand("sphere", 1, 1);

// The sphere command returns a string array.
var array;
var count = {};
array = imel.GetStringArrayResult(count);

var arrayLen = count.value;

for (index = 0; index < arrayLen; index++ )
{
// can do something with the results here, using array[index]
}
```

## 4 | Other improvements

### > FCheck improvements on Mac OS X

## Utilities

### FCheck improvements on Mac OS X

The following improvements have been made to FCheck on Mac OS X:

- FCheck supports Z-depth viewing
- FCheck is available through Terminal. The FCheck command-line utility has different options available on Mac OS X than on Windows and Linux. You can get complete details on all the FCheck options by typing `FCheck -h`.

### Additional utilities on Mac OS X

The following utilities are now available for Mac OS X through Terminal:

- `imgcvt`
- `interlace`
- `blur2d`
- `xpm2bmp`

For more details on these utilities, see the Maya Help.

## Environment Variables

### Improvements to IMF

IMF can now load plug-ins from multiple directories as opposed to loading from a default directory. You can set an environment variable `IMF_PLUG_IN_PATH` to use this functionality.

## MEL

### New MEL commands and flags

The following are new MEL commands and flags for Maya 6.5.

#### General

- The `getAttr` command now works on attributes of type `pointArray`.
- You can now use the `setAttr` command to set string attribute names that begin with a dash (-). For example:  

```
setAttr $nodeAttr -type "string" "-text"
```
- There is a new `-dag` flag for the `cycleCheck` command. Use the `-dag` flag to enable detection of cycles involving the DAG.
- There is a new `floatSlider2` command, which is a two-handled slider widget. The two handles cannot pass one another; in other words handle 1 will always have a value less than or equal to handle 2.
- The `nodeType` command now has a `-api` flag which allows you to determine the API type of a node.
- There is a new `mouse` command for system devices. It has two flags:

```
enableScrollWheel
```

Enable or disable scroll wheel support.

```
scrollWheelStatus
```

Returns current status of scroll wheel support.

- There is a new `-imageBaseColor` flag for the `textureWindow` command. It accepts 3 float inputs representing the Red, Green and Blue components to use as the base color for the image used as the background on the UV Texture Editor.

## 4 | Other improvements

### > New MEL commands and flags

The following example will give the UV Texture Editor background a red tint:

```
textureWindow -e -imageBaseColor 1.0 0.0 0.0  
polyTexturePlacementPanel1;
```

## File Referencing and Namespaces

- There is a new `-moveNamespace` flag for the `namespace` command. Use it to move the contents of one namespace into another.
- You can now use the reference command as follows to return objects in a reference file:

```
reference -rfn -q -node
```

- There is a new `-usingNamespaces` query flag for the `file` command. Use it to determine whether a reference is using a namespace or a renaming prefix to resolve name clashes.
- The `-namespace` flag on the `file` command can now be edited. Use it to change the namespace that a reference will be loaded into. This functionality can also be accessed through a new Edit Field in the Reference Editor.
- The `-cleanReference` flag for the `file` command has been expanded to allow the removal of edits from a reference node by their specific type. You can use it to remove `setattr`, `addattr`, `deleteattr`, `disconnectattr`, and `parent` modifications from the reference node.
- We've added a new `-saveReference` flag for the `file` command. This can be used to save internal edits and connections directly to the referenced file instead of saving it in the reference node of the parent scene.

## Modeling

- There is a new `polyMirrorFace` command that does mirroring of polygon objects.

## 4 | Other improvements

### > New MEL commands and flags

- There is a new `-sma` flag for the `polyExtrudeFacet` and `polyExtrudeEdge` commands. Use the `-sma` flag to specify the smoothing angle used for the edges of extruded polygonal geometry.
- There is a new `-sma` flag for the `polySplit` command. Use it to specify the smoothing angle for the edges of split polygonal geometry.

## Animation

- There is a new `-freezeJoints` flag for the `ikHandle` command. This flag forces the curve, specified by `-curve` option, to align itself along the existing joint chain. When false, or unspecified, the joints will be moved to positions along the specified curve.
- There is a new `-rawFrameNumbers` flag for the `playblast` command. Playblast typically numbers its frames sequentially, starting at zero. This flag will override the default action and frames will be numbered using the actual frames specified by the `-frame` or `-startFrame/-endFrame` flags.
- There is a new flag `-removeUnusedInfluence (-rui)` flag for the `skinCluster` command. If this flag is set to true then the transform or joint whose weights are all zero (they have no effect) will not be bound to the geometry. Having this option set will help speed up the playback of animation.
- There is a new `-smooth` flag for the `copySkinWeights` command. When this flag is used, the weights are smoothly interpolated between the closest vertices instead of assigned from the single closest vertex.

## Paint Effects

- There is a new `scaleBrushBrightness` command you can use to scale how bright a brush appears. There is no single parameter on a brush for overall brightness. Adjusting the `scaleBrushBrightness` can require editing several color, diffuse

## 4 | Other improvements

### > API and Devkit improvements

and specular values. This command provides a simple way of adjusting all the parameters to uniformly scale the brightness. You can adjust several brushes in the scene or just the template brush. There are two flags:

<code>scaleFactor</code>	This is a multiplier, so a value of 0.5 makes the brush half as bright.
<code>useTemplate</code>	If set to true (or non-zero) the scale is applied to the template brush, otherwise it is applied to all selected brushes.

### Cloth

- There is a new `cpSetEdit` command to edit constraint vertex membership lists with the Set Editor.
- There are new `cpGetSolverAttr` and `cpSetSolverAttr` commands for getting values from and setting values for the Cloth solver.

### PhotoShop

- There is a new `-uvSnapPositionTop` flag for the `psdTextureFile` command. This flag specifies the position of the UV snapshot image layer in the PSD file. If the flag is set to "True" this layer is positioned at the top, and if set to "False" the layer is positioned at the bottom next to the background layer in the PSD file.

## API and Devkit

### API and Devkit improvements

This release contains many improvements to the Maya API. New classes have been added to the API along with improvements to existing classes. Improvements include making some API operations faster. In addition, we have written new examples and have added them to the devkit.

## 4 | Other improvements

### > API and Devkit improvements

- "Performance improvements" on page 151
- "New classes and header files" on page 151
- "Important changes to existing classes" on page 153
- "Examples of new plug-in functionality" on page 177
- "Other significant changes" on page 178

### Performance improvements

See "API" on page 61 in the Performance Improvements chapter.

### New classes and header files

#### New MFnKeyframeDelta classes

```
MFnKeyframeDelta  
MFnKeyframeDeltaAddRemove  
MFnKeyframeDeltaBlockAddRemove  
MFnKeyframeDeltaBreakdown  
MFnKeyframeDeltaInfType  
MFnKeyframeDeltaMove  
MFnKeyframeDeltaScale  
MFnKeyframeDeltaTangent  
MFnKeyframeDeltaWeighted
```

These new classes provide access to atomic changes to keyframes as a result of animation curve edit operations. These function sets work in conjunction with the new methods added to the existing MAnimMessage class.

#### New Paint Effects classes

```
MFnPfxGeometry.h  
MRenderLineArray.h  
MRenderLine.h
```

These new classes let you access Paint Effects line information. Please see the related example "pfxInfoCmd" on page 177.

## 4 | Other improvements

### > API and Devkit improvements

#### **MFnAnisotropyShader**

This class facilitates the creation and manipulation of dependency graph nodes representing anisotropy shaders.

#### **MFnLayeredShader**

This class facilitates the creation and manipulation of dependency graph nodes representing layered shaders.

#### **MFnPhongEShader**

This class facilitates the creation and manipulation of dependency graph nodes representing phongE shaders.

#### **MIOStreamFwd.h**

This header file, in addition to the `iosfwd` header file, lets you manage the inclusion of IO stream system headers.

#### **MIteratorType**

This class is used to configure the type of iteration required. An instance of this object is passed into the newly added constructor and reset methods of the iterator classes.

This class supports the use of multiple `MFn::Type` IDs to filter through the iteration. This functionality can improve the performance of an API iteration since multiple `MFn::Type` IDs can be iterated over in one pass.

#### **MPxComponentShape**

This class is derived from `MPxSurfaceShape` and it provides a high level interface for creating surface shapes with components.

#### **MTimer**

This class lets you access to the functionality of the MEL `timerX` command.

## Important changes to existing classes

### M3dView

This class now has better support for view buffers.

```
enum DepthBufferFormat
{
    kDepth_8 = 0,
    kDepth_Float
};
```

The new DepthBufferFormat enum is used for querying the types of depth buffer formats that are available for reading.

```
MStatus readColorBuffer( MImage &image );
```

This method reads the RGB values from the frame buffer for a given view. The buffer is read in a RGBA pixel format, where each channel is one byte in size.

```
MStatus writeColorBuffer( const MImage &image, signed short x = 0, signed
short y = 0) const;
```

This method overwrites the RGB values for the frame buffer for a given view. The expected input is a block of RGBA, where each channel is one byte in size.

```
MStatus readDepthMap(unsigned short x, unsigned short y, unsigned int width,
unsigned int height, unsigned char* bufferPtr, DepthBufferFormat
depthMapPrecision);
```

This method reads the depth values from the frame buffer for a given view. The buffer is read into a block of data that is defined as an argument. The data block size must be large enough to accommodate (view width \* view height \* depth map precision) bytes of data.

```
MStatus readBufferTo2dTexture(unsigned short x, unsigned short y, unsigned
int width, unsigned int height);
```

This method reads the depth values from the frame buffer for a given view into a current, predefined OpenGL 2d texture.

## 4 | Other improvements

### > API and Devkit improvements

**Note** The texture should be created and bound before making this call.

#### MAnimMessage

This class now has better support for animation keyframe messages.

```
static MCallbackId addAnimKeyframeEditedCallback (  
    void (*func)(MObjectArray &editedKeys,  
    void *clientData),  
    void * clientData = NULL,  
    MStatus * ReturnStatus = NULL );
```

This method registers a callback that is called whenever groups of keys are modified. The callback is invoked once per change for either single or groups of keyframes.

```
static MCallbackId addAnimKeyframeEditedCallback (  
    MObject &animNode,  
    void (*func)(MObject &animNode,  
    MObjectArray &editedKeys,  
    void *clientData),  
    void *clientData = NULL,  
    MStatus * ReturnStatus = NULL );
```

This method registers a callback that is called whenever groups of keys are modified. The callback is invoked once per change for single or groups of keyframes on the specified animation curve node.

```
static void flushAnimKeyframeEditedCallbacks();
```

Animation keyframe edited callbacks are queued such that they are only issued on idle events. However, there may be a time when you want to issue the callback at a specific time. This method provides this functionality. This method will flush all animation keyframe edited callbacks and force them to issue their callbacks with the data they contain.

## 4 | Other improvements

### > API and Devkit improvements

#### MDataHandle

We have added support for getting the MTypeId information of a data handle. In some cases, data handles accept more than one type ID which requires an array for the results.

```
MStatus acceptedTypeIds( MUIntArray &) const;
```

This method returns an unsigned integer array of numbers which can be used to create a MTypeId of the accepted data type. This is useful when working with MPxDData objects.

#### MFileIO

There has been some substantial changes to this class in order to support the file referencing changes that have been made in Maya.

```
static MStatus import( const MString& fileName, const  
char* type = NULL, bool preserveReferences = false,  
const char* nameSpace = NULL);
```

```
static MStatus exportSelected( const MString& fileName,  
const char* type = NULL, bool preserveReferences =  
false);
```

```
static MStatus exportAll( const MString& fileName, const  
char* type = NULL, bool preserveReferences = false);
```

The three existing methods above now have a new default parameter for preserving references. In addition, the import() method has a new default parameter for specifying the namespace to be used.

We now have support for deferring reference loading:

```
static MStatus reference( const MString& fileName, bool  
deferReference = false );
```

This existing method has been modified to include a default parameter that controls the deferring of references.

## 4 | Other improvements

### > API and Devkit improvements

Additionally, we have provided many new methods for handling file referencing:

```
static MStatus exportAnimFromReference ( const MString&
fileName, const MString& refNodeName, const char* type =
NULL);
```

This method exports the animation from the specified reference node, to the specified file.

```
static MStatus exportSelectedAnimFromReference ( const
MString& fileName, const char* type = NULL);
```

This method exports the animation from a selected reference node, to the specified file.

```
static MStatus exportSelectedAnimFromReference ( const
MString& fileName, const MString& refNodeName, const
char* type = NULL);
```

This method exports the selected animation from the specified reference node, to the specified file.

```
static MStatus exportAnim ( const MString& fileName,
const char* type = NULL);
```

This method exports animation to a specified file.

```
static MStatus exportSelectedAnim ( const MString&
fileName, const char* type = NULL);
```

This method exports selected animation objects to the specified file.

```
static MStatus exportAsReference ( const MString&
fileName, const char* type = NULL);
```

This method exports selected objects to a reference file.

```
static MString getLastTempFile(MStatus * ReturnStatus =
NULL);
```

This method returns the last temporary file used in the file save operation.

## 4 | Other improvements

### > API and Devkit improvements

```
static bool getErrorStatus(MStatus * ReturnStatus =  
NULL);
```

This method returns the error status of the last file read.

```
static MStatus resetError();
```

This method resets the current file error state.

```
static MString loadReference(MString& referenceFileName,  
MStatus *returnStatus = NULL);
```

This method loads the reference file specified.

```
static MString loadReferenceByNode(MString&  
referenceNodeName, MStatus *returnStatus = NULL);
```

This method loads the reference, referenced by the given reference node name.

```
static MString loadReferenceByNode(MObject&  
referenceNode, MStatus *returnStatus = NULL);
```

This method loads the reference by the reference node specified.

```
static MString unloadReference(MString&  
referenceFileName, MStatus *returnStatus = NULL);
```

This method unloads the reference file specified by the filename.

```
static MString unloadReferenceByNode(MObject&  
referenceNode, MStatus *returnStatus = NULL);
```

This method unloads a reference by the specified reference node.

```
static MString unloadReferenceByNode(MString&  
referenceNodeName, MStatus *returnStatus = NULL);
```

This method unloads a reference node specified by an MObject.

```
static MString getReferenceFileByNode(MObject&  
referenceNode, MStatus *returnStatus = NULL);
```

This method returns the reference a node belongs to.

## 4 | Other improvements

### > API and Devkit improvements

#### MFn

```
kKeyframeDelta  
kKeyframeDeltaMove  
kKeyframeDeltaScale  
kKeyframeDeltaAddRemove  
kKeyframeDeltaBlockAddRemove  
kKeyframeDeltaInfType  
kKeyframeDeltaTangent  
kKeyframeDeltaWeighted  
kKeyframeDeltaBreakdown
```

These classes now have new enum types for working with function sets.

```
MFn::Type nodeType;
```

The MFn class now also contains a variable that you can use to store a MFn::Type.

**Note** An unused #if defined block has been removed from the MFn.h header file.

#### MFnAttribute

This class now has better support for controlling the display of attributes in the Channel Box. In addition, we now have better support for worldspace attributes of proxy DAG nodes.

```
bool isChannelBoxFlagSet( MStatus* ReturnStatus=NULL )  
const;
```

This boolean returns *true* if this attribute has its Channel Box flag set.

```
MStatus setChannelBox( bool state );
```

This method sets whether or not this attribute appears in the Channel Box when the node is selected.

```
MStatus setWorldSpace( bool state );
```

## 4 | Other improvements

### > API and Devkit improvements

This method sets the *workspace* flag on the attribute. If this flag is set, the attribute will be computed every time the node's workspace information is changed as long as there is another attribute requesting its value.

```
MStatus setAffectsWorldSpace( bool state );
```

This method sets the *affects workspace* flag on the attribute. If set, any modifications to the attribute will cause workspace to be computed as long as there is another attribute requesting workspace.

```
bool isWorldSpace( MStatus* ReturnStatus=NULL ) const;
```

This method returns true if the attribute has its workspace flag set.

```
bool isAffectsWorldSpace( MStatus* ReturnStatus=NULL ) const;
```

This method returns true if the attribute has its *affects workspace* attribute flag set.

### MFnClip

We have added two new methods to this class for working with tracks and channel settings:

```
MStatus setTrack(int index, MDGModifier* mod = NULL);
```

This method specifies the one-based track number for the clip.

```
MStatus setAbsoluteChannelSettings(const MIntArray& absoluteChannels, MDGModifier* mod = NULL);
```

This method sets which channels of the clip are absolute and which are relative.

### MFnDependencyNode

A new attribute utility method is available in this class.

```
bool hasAttribute(const MString& name, MStatus* ReturnStatus=NULL) const;
```

## 4 | Other improvements

### > API and Devkit improvements

This class returns *true* if the specified node already has an attribute with the given name.

#### MFnMesh

We have added more UV functionality to this class. In addition, we have added new functionality for finding intersections faster and tangent and binormal information.

```
MStatus getFaceUVSetNames( int polygonId, MStringArray
&setNames) const;
```

This method returns the UV set names that are mapped to the polygon specified by polygon ID.

The methods which improve mesh intersection speed are described next.

```
static MMeshIsectAccelParams uniformGridParams( int xDiv,
int yDiv, int zDiv );
```

Creates a MMeshIsectAccelParams configuration object to pass to the methods MFnMesh::closestIntersection(), MFnMesh::anyIntersection(), or MFnMesh::allIntersections().

```
static MMeshIsectAccelParams autoUniformGridParams();
```

Creates a MMeshIsectAccelParams configuration object to pass to the methods MFnMesh::closestIntersection(), MFnMesh::anyIntersection(), or MFnMesh::allIntersections().

```
bool closestIntersection( const MFloatPoint& raySource,
const MFloatVector& rayDirection, const MIntArray*
faceIds, const MIntArray* triIds, bool idsSorted,
MSpace::Space space, float maxParam, bool
testBothDirections, MMeshIsectAccelParams *accelerator,
MFloatPoint& hitPoint, float* hitRayParam, int* hitFace,
int* hitTriangle, float*hitBary1, float*hitBary2, float
tolerance = 1e-6, MStatus*ReturnStatus = NULL );
```

Finds the closest intersection of a ray starting at raySource and travelling in rayDirection with the mesh.

## 4 | Other improvements

### > API and Devkit improvements

```
bool anyIntersection( const MFloatPoint& raySource, const
MFloatVector& rayDirection, const MIntArray* faceIds,
const MIntArray* triIds, bool idsSorted, MSpace::Space
space, float maxParam, bool testBothDirections,
MMeshIsectAccelParams *accelerator, MFloatPoint&
hitPoint, float* hitRayParam, int* hitFace, int*
hitTriangle, float*hitBary1, float*hitBary2, float
tolerance = 1e-6, MStatus*ReturnStatus = NULL );
```

Finds any intersection of a ray starting at raySource and travelling in rayDirection with the mesh.

```
bool allIntersections( const MFloatPoint&raySource, const
MFloatVector& rayDirection, const MIntArray* faceIds,
const MIntArray* triIds, bool idsSorted, MSpace::Space
space, float maxParam, bool testBothDirections,
MMeshIsectAccelParams *accelerator, bool sortHits,
MFloatPointArray& hitPoints, MFloatArray* hitRayParams,
MIntArray* hitFaces, MIntArray* hitTriangles,
MFloatArray* hitBary1, MFloatArray* hitBary2, float
tolerance = 1e-6, MStatus* ReturnStatus = NULL );
```

Finds all intersections of a ray starting at raySource and traveling in rayDirection with the mesh.

```
MStatus sortIntersectionFaceTriIds(MIntArray* faceIds,
MIntArray* triIds );
```

Convenient routine for sorting faceIds or face/triangle ids before passing them into the closestIntersection(), allIntersections(), or anyIntersection() methods.

```
MStatus freeCachedIntersectionAccelerator();
```

If the mesh has a cached intersection accelerator structure, then this routine forces it to be deleted.

```
MString cachedIntersectionAcceleratorInfo( MStatus
*ReturnStatus = NULL );
```

Retrieves a string that describes the intersection acceleration structure for this object, if any.

## 4 | Other improvements

### > API and Devkit improvements

```
static MString globalIntersectionAcceleratorsInfo();
```

Retrieves a string that describes the systemwide resource usage for cached mesh intersection accelerators.

```
static void clearGlobalIntersectionAcceleratorInfo();
```

Clears the "total count", "total build time", and "peak memory" fields from the information string returned by `globalIntersectionAcceleratorsInfo()`.

```
MStatus getPolygonTriangleVertices( int polygonId, int  
triangleId, int triangleVertices[3] ) const;
```

This method retrieves the object-relative (mesh-relative/global) vertex indices for the specified triangle in the specified polygon. The indices refer to the elements in the array returned by the 'getPoints' method.

```
MStatus getFaceVertexTangent( int faceIndex, int vertexIndex, MVector &  
normal, MSpace::Space space = MSpace::kObject, const MString * uvSet = NULL  
) const;
```

This method returns the tangent of the vertex described by `faceIndex` and `vertexIndex`.

```
MStatus getFaceVertexTangents( int faceIndex, MFloatVectorArray & normals,  
MSpace::Space space = MSpace::kObject, const MString * uvSet = NULL ) const;
```

This method returns the tangents of all vertices of the face specified by `faceIndex`.

```
MStatus getBinormals( MFloatVectorArray& normals, MSpace::Space space =  
MSpace::kObject, const MString * uvSet = NULL ) const;
```

This method returns the binormals for all face vertices.

```
MStatus getFaceVertexBinormal( int faceIndex, int vertexIndex, MVector &  
normal, MSpace::Space space = MSpace::kObject, const MString * uvSet = NULL)  
const;
```

This method returns the binormal information of the face vertex specified by `faceIndex` and `vertexIndex`.

## 4 | Other improvements

### > API and Devkit improvements

```
MStatus getFaceVertexBinormals( int faceIndex, MFloatVectorArray & normals,
MSpace::Space space = MSpace::kObject, const MString * uvSet = NULL) const;
```

This method returns the per vertex per face binormals for the face specified by faceIndex.

#### MFnParticleSystem

This class now lets you create and manipulate particle systems.

The RenderType enum now contains new items that describe the exact kHardware rendering type. The new enum items are:

```
kMultiPoint
kMultiStreak
kNumeric
kPoints
kSpheres
kSprites
kStreak
```

**Note** kHardware has been removed from the enum since the specific hardware render type is now used.

```
MObject create( MStatus* status = NULL );
```

This method creates a new particle shape with a transform.

```
MObject create( MObject parent, MStatus* status = NULL );
```

This method creates a new particle shape with the specified parent.

```
MStatus emit( const MPoint& position );
```

This method adds a new particle at the specified position to the particle system.

```
MStatus emit( const MPointArray& positionArray );
```

This method adds an array of particles specified by the position array to the particle system.

## 4 | Other improvements

### > API and Devkit improvements

```
MStatus emit( const MPoint& position, const MVector&
velocity );
```

This method adds a new particle at the specified position and velocity to the particle system.

```
MStatus emit( const MPointArray& positionArray, const
MVectorArray& velocityArray );
```

This method adds an array of particles with velocity to the particle system.

```
MStatus saveInitialState() const;
```

This method resets the particle's current state as the initial state.

```
void velocity( MVectorArray& ) const;
```

This method returns the particle velocity array.

```
void acceleration( MVectorArray& ) const;
```

This method returns the particle acceleration array.

```
void mass( MDoubleArray& ) const;
```

This method returns the particle mass array.

```
unsigned int getPerParticleAttribute( MString &attrName, MIntArray&,
MStatus* status = NULL ) const;
```

This method returns the per particle integer attribute of the given name.

```
unsigned int getPerParticleAttribute( MString &attrName, MVectorArray&,
MStatus* status = NULL ) const;
```

This method returns the per particle vector attribute of the given name.

```
unsigned int getPerParticleAttribute( MString &attrName, MDoubleArray&,
MStatus* status = NULL ) const;
```

This method returns the per particle double attribute of the given name.

## 4 | Other improvements

### > API and Devkit improvements

```
void setPerParticleAttribute( MString &attrName, MVectorArray&, MStatus*  
status = NULL ) const;
```

This method sets the per particle vector attribute of the specified name with the specified values.

```
void setPerParticleAttribute( MString &attrName, MDoubleArray&, MStatus*  
status = NULL ) const;
```

This method sets the per particle double attribute of the specified name with the specified values.

```
bool isPerParticleIntAttribute( const MString &attrName, MStatus* status =  
NULL ) const;
```

This method checks if the input attribute is a per particle integer attribute.

```
bool isPerParticleDoubleAttribute( const MString &attrName, MStatus* status  
= NULL ) const;
```

This method checks if the input attribute is a per particle double attribute.

```
bool isPerParticleVectorAttribute( const MString &attrName, MStatus* status  
= NULL ) const;
```

This method checks if the input attribute is a per particle vector attribute.

```
bool isDeformedParticleShape( MStatus* status = NULL )  
const;
```

This method returns true if this particle shape is deformed.

```
MObject deformedParticleShape( MStatus* status = NULL )  
const;
```

This method returns the deformed particleShape.

```
MObject originalParticleShape( MStatus* status = NULL )  
const;
```

This method returns the original particleShape.

## 4 | Other improvements

### > API and Devkit improvements

#### MFnSkinCluster

This class now has enhanced support for getting and setting the weights of skin clusters.

```
MStatus getWeights(const MDagPath& path,  
                  const MObject& components,  
                  MDoubleArray& weights,  
                  unsigned& influenceCount) const;
```

This method gets the skinCluster weights for all influenceObjects for the specified components of the object whose dagPath is specified.

```
MStatus setWeights(const MDagPath& path,  
                  const MObject& components,  
                  unsigned jointIndex,  
                  double value,  
                  bool normalize = true,  
                  MDoubleArray *oldValues = NULL);
```

This method sets the skinCluster weight for the influence object that affects the specified components of the object whose dagPath is specified.

```
MStatus setWeights(const MDagPath& path,  
                  const MObject& components,  
                  MIntArray& influenceIndices,  
                  MDoubleArray& values,  
                  bool normalize = true,  
                  MDoubleArray *oldValues = NULL);
```

This method sets the skinCluster weights on the specified components of the object whose dagPath is specified for the influence objects corresponding to the specified influence indices.

```
MStatus getPointsAffectedByInfluence(const MDagPath&  
path,  
                                     MSelectionList& result,  
                                     MDoubleArray& weights);
```

This method lets you query the non-zero weights for a particular influence object.

## 4 | Other improvements

### > API and Devkit improvements

#### MGlobal

This class now has enhanced support for selection lists and the display of CVs. Also, there is a new method for executing MEL in a thread-safe manner.

**Note** The `MMayaState` enum no longer contains the `kBatchRender` item. Use `kBatch` instead.

```
static MStatus setHiliteList( MSelectionList & src );
```

This method sets the current hilite list. The current selection list is unchanged.

```
static MStatus setDisplayCVs( MSelectionList&, bool );
```

This method controls the draw of control points in the specified selection list. The selection items on the given list are marked for drawing. This overrides Maya's current draw list and, for example, lets you draw the control points without being in vertex selection mode.

```
static MStatus executeCommandOnIdle const MString&  
command, bool displayEnabled = false);
```

This method sets a MEL command to execute on the next idle event. This is a thread safe way to execute commands. However, there is no way to access the command results. With the `displayEnabled` boolean, you can set whether or not the command is displayed in the Script Editor. By default, `displayEnabled = false`.

#### MImage

This class now has enhanced image querying and manipulation operations.

```
MStatus getSize(unsigned int &width, unsigned int  
&height) const;
```

This existing method has now been made `const`.

## 4 | Other improvements

### > API and Devkit improvements

```
unsigned char* pixels() const;
```

This method returns a pointer to the first pixel of the uncompressed pixels array. This array is tightly packed, of size (*width \* height \* depth*) bytes. Currently, pixels are always stored in a RGBA (depth=4 bytes) pixel format.

```
void setPixels(unsigned char* pixels, unsigned int width,  
unsigned int height);
```

This method sets the pointer to the first pixel of the uncompressed pixels array. This array is tightly packed, of size (*width \* height \* depth*) bytes. Currently, pixels are always stored in a RGBA (depth=4 bytes) pixel format.

### MltDag

This is a new constructor with reset methods that accept an object that defines the iteration type. With this constructor, you can now specify multiple MFn types to filter an iteration using the MIteratorType class.

```
MltDag( MIteratorType& dagInfoObject,  
        TraversalType = kDepthFirst,  
        Status* ReturnStatus = NULL);
```

This method initializes the root of iteration at the world node, and sets the type of the traversal and the filter list.

```
MStatus reset(MIteratorType& dagInfoObject,  
              const MObject* object,  
              const MDagPath* path,  
              TraversalType = kDepthFirst);
```

This method resets the interator's root, type, and filter list.

### MltDependencyGraph

This is a new constructor with reset methods that accept an object that defines the iteration type. You can now specify multiple MFn types to filter the iteration using the MIteratorType class.

## 4 | Other improvements

### > API and Devkit improvements

```
MItDependencyGraph( MObject& rootNode,  
    MPlug& rootPlug,  
    MIteratorType& infoObject,  
    Direction direction = kDownstream,  
    Traversal traversal = kDepthFirst,  
    Level level = kNodeLevel,  
    MStatus* ReturnStatus = NULL );
```

This method constructs a DG Iterator with a root node or a root plug.

```
MStatus resetTo( MObject* rootNode,  
    MPlug* rootPlug,  
    MIteratorType& infoObject,  
    Direction direction = kDownstream,  
    Traversal traversal = kDepthFirst,  
    Level level = kPlugLevel );
```

This method clears the iterator data and reinitializes the iterator.

#### MItDependencyNodes

This is a new constructor with reset methods that accept an object that defines the iteration type. You can now specify multiple MFn types to filter the iteration using the MIteratorType class.

```
MItDependencyNodes( MIteratorType& infoObject,  
    MStatus * ReturnStatus = NULL );
```

This method initializes the iterator and the filter list.

```
MStatus reset(MIteratorType& infoObject);
```

This method resets the iterator and reinitializes the filter list. The filter list is specified by the MIteratorType object.

#### MItMeshPolygon

We have added more UV support to this class along with a new set points method.

```
MStatus getUVSetNames( MStringArray &setNames) const;
```

## 4 | Other improvements

### > API and Devkit improvements

This method returns the names of the UV sets that are mapped to the current mesh polygon that the iterator is pointing to.

```
MStatus setPoints( MPointArray & pointArray,  
MSpace::Space space = MSpace::kObject );
```

This method lets you set the positions of the current mesh polygon points using an array.

#### MItMeshFaceVertex

This class has been improved and now supports iterating over face vertices. A face vertex component can now be passed into the constructor of the iterator. As a result, you can now retrieve the face vertex with the following new method. Additionally, you can now access the tangent and binormal information of the component.

```
MObject faceVertex(MStatus * ReturnStatus = NULL) const;
```

This method returns the double index component of the vertex and face.

```
MVector getTangent( MSpace::Space space = MSpace::kObject, const MString *  
uvSet = NULL, MStatus * ReturnStatus = NULL);
```

This method returns the tangent information of the current mesh face vertex.

```
MVector getBinormal( MSpace::Space space = MSpace::kObject, const MString *  
uvSet = NULL, MStatus * ReturnStatus = NULL);
```

This method returns the binormal information of the current mesh face vertex.

#### MMatrix

This class now has support for finding out if a matrix is singular.

```
bool isSingular() const;
```

This method returns *true* if the given matrix is singular.

## 4 | Other improvements

### > API and Devkit improvements

#### MNodeMessage

This new dirty node callback provides information on which plug was made dirty.

```
static MCallbackId addNodeDirtyCallback(  
    MObject& node,  
    void (*func)( MObject& node,  
    MPlug& plug,  
    void* clientData ),  
    void * clientData = NULL,  
    MStatus * ReturnStatus = NULL );
```

This method registers a callback for node dirty messages. This callback provides the plug on the node that was dirtied. This method only provides dirty information on input plugs.

#### MPlug

This class now has support for controlling the display of plugs in the Channel Box. Also, you can now find out if a plug is dynamic, as well as construct and destruct data handles for the plug.

```
bool isChannelBoxFlagSet( MStatus* ReturnStatus = NULL )  
const;
```

This method returns *true* if this plug or its attribute has its Channel Box flag set. Attributes appear in the Channel Box if their Channel Box flag is set or if they are keyable.

```
MStatus setChannelBox( bool channelBox );
```

This method sets whether this plug is displayed in the Channel Box. This overrides the default display of a plug set with `MFnAttribute::setChannelBox`. Keyable attributes are always shown in the Channel Box, so this flag is ignored on keyable plugs.

```
bool isDynamic( MStatus* ReturnStatus = NULL ) const;
```

This method determines whether or not the attribute's type is dynamic.

```
MDataHandle constructHandle(MDataBlock&) const;
```

## 4 | Other improvements

### > API and Devkit improvements

This method returns a data handle constructed for this plug.

```
void destructHandle(MDataHandle&) const;
```

This method destructs the data handle for this plug.

#### MPxCommand

This class now has support for including the line number of a proxy command warning or error message.

```
static void displayWarning( const MString & theWarning,  
bool wantDisplayLineNumber );
```

```
static void displayError( const MString & theError, bool  
wantDisplayLineNumber );
```

The following overloaded methods include a parameter for requesting the generation of line number information when the warning or error is displayed.

**Note** The option to display line numbers in the MEL scripting window must also be on for line number information to be generated.

#### MPxFileTranslator

This class now has enhanced support for referenced models and now provides information on Maya's current file access mode.

The FileAccessMode enum now contains a new item called kReferenceAccessMode for controlling the reference of data in the current scene.

```
virtual bool haveReferenceMethod () const;
```

This routine is called by Maya while it is executing in the MPxFileTranslator constructor. Maya uses this entry point to query the translator and determine if it provides a reference method. This

## 4 | Other improvements

### > API and Devkit improvements

virtual method must be overloaded in derived classes if a writer method exists, as the default version defined in `MPxFileTranslator` always returns false.

```
static MPxFileTranslator::FileAccessMode fileAccessMode();
```

This method returns Maya's current fileAccess mode.

### MPxHwShaderNode

This class now provides more information on the object being drawn in a hardware shader node.

```
enum DirtyMask {
    kDirtyNone          = 0x0000,
    kDirtyVertexArray  = 0x0001,
    kDirtyNormalArray  = 0x0002,
    kDirtyColorArrays  = 0x0004,
    kDirtyTexCoordArrays = 0x0008,
    kDirtyAll          = 0x000f
};
```

This is the new `DirtyMask` enum that is used with the `dirtyMask()` method to find out which geometry items have changed since the last plug-in draw call.

```
const MDagPath & currentPath() const;
```

This method returns a reference to the current path for which the shader is invoked. The path is only valid before a call to any of these attribute specifying routines:

```
normalsPerVertex()
colorsPerVertex()
texCoordsPerVertex()
getTexCoordSetNames()
hasTransparency()
provideVertexIDs()
```

The path is not guaranteed to be valid at any other time.

```
unsigned int dirtyMask() const;
```

## 4 | Other improvements

### > API and Devkit improvements

This method returns a “dirty” mask that indicates which geometry items have changed from the last invocation of the plug-in to draw. The mask is valid *only* at the time that geometry() or glGeometry() is called.

```
virtual MStatus getAvailableImages( const MString&
uvSetName, MStringArray& imageNames);
```

Override this method to specify the list of images that are associated with the given UV set in this shader. This method is used to determine which texture images are available in the UV Texture Editor.

```
virtual MStatus renderImage( const MString& imageName,
const float region[2][2], int& imageWidth, int&
imageHeight);
```

Override this method to draw an image of this material. This method allows a shader to override a render. Maya only uses this method if getAvailableImages returns at least one image name. The imageWidth and imageHeight parameters should be populated with the native resolution of the input image to allow pixel snapping or other resolution dependent operations.

### MRampAttribute

This class now has enhanced support for getting the color or value of a ramp position.

```
void getColorAtPosition (float position, MColor& color,
MStatus * returnStatus);
```

This method gets the color at a position. If this ramp is a curve ramp, then this method returns a failure. If this ramp is empty, then this method returns a failure. If the position is out of range, either the start or end value is returned.

```
void getValueAtPosition (float position, float& value,
MStatus * returnStatus);
```

## 4 | Other improvements

### > API and Devkit improvements

This method gets the value at position. If this ramp is a color ramp, then this method returns a failure. If this ramp is empty, then this method returns a failure. If the position is out of range, either the start or end value is returned.

### MSceneMessage

This class now supports the before and after unload reference messages. Also, you can now cancel File New, Open, and Save operations using this class.

Called before a File > UnloadReference operation

kBeforeUnloadReference,

Called after a File > UnloadReference operation

kAfterUnloadReference,

These new Message enum items let you listen to before and after unload reference messages.

Called prior to File > New operation, allows user to cancel action

kBeforeNewCheck,

Called prior to File > Open operation, allows user to cancel action

kBeforeOpenCheck,

Called prior to File > Save operation, allows user to cancel action

kBeforeSaveCheck

These new Messages for File New, Open, and Save let you cancel the operation.

```
static MCallbackId addCallback( Message, void (*func)( bool* retCode, void* clientData),
```

```
void * clientData = NULL,
```

```
MStatus * ReturnStatus= NULL );
```

This method is used in conjunction with the new Message items for File New, Open, and Save which let you cancel the operation. The callback function contains a new retCode parameter. A retCode set to *true* continues the file operation as normal. A retCode of *false* aborts the operation.

## 4 | Other improvements

### > API and Devkit improvements

**Note** The Message parameter must be one of kBeforeNewCheck, kBeforeOpenCheck or KBeforeSaveCheck.

### MString

This class now has enhanced support for string operations.

```
MString& operator += ( unsigned int other );  
MString& operator += ( float other );
```

These are new MString concatenation operators.

### MTypes.h

The MAYA\_API\_VERSION number has been updated to 650.

### MUiMessage

This class now has enhanced support for view messages.

```
static MCallbackId add3dViewDestroyMsgCallback(  
    const MString& panelName,  
    void (*func) MString &panelName, void* clientData),  
    void * clientData = NULL,  
    MStatus * ReturnStatus = NULL );
```

This method registers a callback for when a specific 3D view is destroyed. The callback is called before the destruction of the view. The callback function is passed any client data that was provided when the callback was registered.

```
static MCallbackId add3dViewPreRenderMsgCallback(  
    const MString& panelName,  
    void (*func)( MString &panelName, void*  
clientData),  
    void * clientData = NULL,  
    MStatus * ReturnStatus = NULL );
```

## 4 | Other improvements

### > API and Devkit improvements

This method registers a callback when a particular 3D view is about to render its contents. It is called before the scene is drawn, but after the background has been drawn. The callback function is passed any client data that was provided when the callback was registered.

```
static MCallbackId add3dViewPostRenderMsgCallback(  
    const MString& panelName,  
    void (*func)( MString &panelName, void*  
clientData),  
    void * clientData = NULL,  
    MStatus * ReturnStatus = NULL );
```

This method registers a callback when the 3D view is about to display its rendered contents to the Render View. It is called for every refresh of the view, after the scene is drawn, but before any 2D adornments are drawn. The callback function is passed any client data that was provided when the callback was registered.

## Examples of new plug-in functionality

### apiSimpleShape

This is a new example that demonstrates the usage of the MPxComponentShape class for creating surface shapes with components.

### intersectCmd

This new example demonstrates the usage of the new acceleration intersection methods available in the MFnMesh class.

### pfxInfoCmd

This is a new example that demonstrates the usage of the new Paint Effects API classes.

### viewCallbackTest

This is a new example that demonstrates the new pre/post render and view destroyed callbacks of the MUiMessage class.

## 4 | Other improvements

### > API and Devkit improvements

#### weightListNode

This is a new example that demonstrates the reading and writing of a *multi of multi* attribute. An example of a multi of multi attribute is the `weightList` attribute of a deformer.

#### Other significant changes

- Maya now has a `-noAutoLoadPlugins` option.
- The `nodeType` MEL command now has an `-api` flag for finding out the associated `MFn::Type` of a node.
- Maya now searches the `$MAYA_LOCATION/modules` directory for module files which releases the restriction that require plugins be placed on local disks even if Maya is installed on a network.
- The behavior of the `MPlug::getValue( MString& val, MDGContext & ctx )` method has been altered to not fail if the default value of the attribute is `NULL`. Instead, this method will succeed and return an empty `MString` in this situation.
- The method `MStatus MFnDependencyNode::addAttribute( const MObject& attr, MFnDependencyNode::MAttrClass type )` will always fail if the type is `kLocalDynamicAttr`. This option is not fully supported in Maya so we have made the API fail under this situation.
- We have been reducing the “bloat” created by systems files such as `iostreams`. In some cases, we now use forward declarations for streams in the API headers. This will cause some existing code not to compile but can be worked around by including `MIOStream.h`.

#### Compiler requirements

- Red Hat 9 and `gcc 3.3.4` are required for developing on Linux.

## 4 | Other improvements

### > API and Devkit improvements

#### Note

Our example Makefiles now contain the `-pthread` compile and link option for building plug-ins and standalone API applications.

- Mac OS X 10.3.4 and gcc 3.3 (build 1666) or Xcode 1.5 are required for developing on this platform.

**4 | Other improvements**  
**> API and Devkit improvements**

# 5

# Performance tips

## Getting the most out of Maya

Use the following suggestions to maximize Maya's performance.

- "Best practices for performance" on page 181
- "Optimize your scene for playback" on page 188

## Best practices for performance

### Interactive drawing

- Turn on Shading > Interactive Shading to improve performance when moving the camera.
- Save the file in the simplest display mode to make load time faster. Avoid saving files in textured display mode.
- Use default display options for polygons.
- Turn off display of all non-essential UI that updates when the scene view updates, for example: Heads Up Display, Time Slider, Range Slider, Hypershade, Attribute Editor, UV Texture Editor.
- Turn off the display of all non-essential scene elements, ranging from the Grid to drawing skeletons.
- Work with a layout that only includes a single modeling panel.
- Partition the scene into logical grid areas that don't overlap. Use display layers to turn on or off partitions as required. If you need to set up layers for visibility, splitting up into non-overlapping rectangular grid sectors should help. Maya performs front, back, and side camera frustum culling per object. If even one triangle shows up, Maya draws everything.
- Use the new hardware Mipmap filtering options, which are in the Hardware Texturing section of the shader node, Texture Filter drop-down list.

## 5 | Performance tips

### > Best practices for performance

- Use the new Use default material display mode if applicable.
- Don't use extra Shading menu settings like Wireframe on Shaded if you don't need them.
- If possible, use Show > Isolate Select to limit what is displayed and refreshed in the scene.
- Objects with less than a few hundred triangles, and especially those with only a few triangles will have a high performance overhead. If possible, merge the objects together into one. Maya performs much better with less objects that each have thousands of triangles versus many objects with only a few triangles.
- Use instancing when possible. This includes instancing geometry, materials and textures. If using file texture nodes, it is better to have one file texture node that is instanced versus many copies of the node. This affects memory used when the file is loaded as well as on disk.
- Make sure to set your video card settings to Maya settings and disable vertical sync (sync).
  - The default material display option can be used to see the difference between using 1 shader for all objects versus  $n$  shaders.
  - If surfaces are partitioned into many layers, this may slow down shaded mode display. Attempt to use fewer partitions if possible, and if used for visibility to partition the surfaces into grid sections. This helps with visibility culling.
- Attempt to build surfaces that do not intersect each other in terms of their bounds (bounding boxes). Visibility testing performs worse in these cases.
- Sorting of the dag hierarchy by display attribute types may help. This can be done by reordering dag objects in Outliner.
- Ordering your dag hierarchy may help:
  - all surfaces first, then all non-surfaces

## 5 | Performance tips

### > Best practices for performance

- by visibility within a given region (perhaps by layer if layers are used for visibility partitioning)
- by depth to viewer (if feasible)
- material and lighting attributes: for example, all lamberts, then all blinns, then all shaders which don't use lighting, then all those that do, and finally all shaders which don't have transparency, then all those that do
- Use a minimal shader for an object. For example use a surface shader when you don't require lighting, or you'll be overriding lighting elsewhere, for example, with color per vertex.
- Remove duplicate shading networks.

## Hardware Shaders

These are coding tips for hardware shader plug-in writers.

- Do not push and pop all GL attributes if not required. This is more expensive than pushing specific attributes.
- If you're only using hardware shaders in the scene view, we recommend that you use the older API: `geometry()`, `bind()` and `unbind()`. If you also want to batch render your hardware shader, we recommend you use the newer API: `glGeometry`, `glBind` and `glUnbind`.
- We recommend you cache all node attributes as internal. The example plug-ins `hwColorPerVertex`, `hwPhongShader` and the `cgFxShader` all do this. Non-cached values are very expensive to evaluate and can double the draw time.
- Avoid using complex attributes (structures and arrays).
- If the output color on the shader is not important, do not make any attributes affect it. This causes additional computation as marking one attribute dirty causes a recomputation to derive the dependent one.

## 5 | Performance tips

### > Best practices for performance

- The `compute()` method can be left empty if not required. The simpler the method the better. You will not see Hypershade swatches if you do not compute the output color.
- Do not post-normalize tangents in the plug-in. This is done for you.
- Data sent via the geometry calls are read-only and are cached internally. Do not modify the values.
- Use simple data structures (versus Maya API objects) for simple data. For example, use `float3` versus `MFloatVector`. There is a performance overhead due to the interfaces used in OpenMaya.
- Take advantage of methods available on `M3dView`. OpenGL state is cached internally and can give better performance than using `beginGL()`, `endGL()` and direct calls to OpenGL.
- Use `glDrawRangeElements` to draw with. This is the API recommended by the card vendor for drawing with the geometry arrays passed to the hardware shader.
- Make sure to set the `hasTransparency()` return value appropriately. Setting it to true causes Maya to draw an object twice (once culling front faces, once culling back).
- Take into account the current display state in `M3dView`. For example, don't disable lighting if the display mode is Use No Lights.
- When transparency is enabled, framebuffer blending is already enabled. You do not need to enable it again.
- Take advantage of the new method: `MPxHwShaderNode::currentPath()` to send the appropriate information for attribute query methods (for example, `getTexCoordSetNames()`).
- Test which parameters on color/alpha and depth mask are enabled for interactive and especially for hardware rendering. They can give hints as to how to draw a simpler version of the geometry.

## 5 | Performance tips

### > Best practices for performance

- For the Hardware renderer, the plug-in can be called multiple times. The general sequence is usually: depth pass, [lighting pass[es]], color pass, [shadow map pass], [alpha pass], [depth pass]. Items marked with braces “[ ” “ ] ” are optional and dependent upon the number of lights in the scene, whether shadows are enabled for those lights, and whether alpha and depth output images are specified in the Render Globals.
- Use the new Ignore Hardware Shader option which is available per object (NURBS or polygonal surfaces). This is available in the Attribute Editor in the Object Display section and as a new option in the Display > Object Display menu. Use Ignore Hardware Shader to make the hardware shader not show up on the object, or use Use Hardware Shader to restore the default state of showing the shader. This allows users of the plug-in to selectively disable shader display for performance reasons.

## Memory

- Unlimited undo queue takes more memory than a limited undo queue. The default Undo queue is set to 50 in the Preferences window.
- Take advantage of large address awareness on Windows XP up to 3 GB.
- Use instancing when possible. This includes geometry, materials, textures, lights, and so on.

## Animation

- See also “Optimize your scene for playback” on page 188.
- See also “New attribute for Hardware Shader Plug-in” on page 102.
- Use Skin > Edit Smooth Skin to limit max influences, remove unused influences and prune small influences.

## 5 | Performance tips

### > Best practices for performance

- Use Deform > Prune Membership to remove components that aren't affected by the deformer.

## Fur

- Lowering the Fur Accuracy value for the Fur Feedback hairs significantly increases interactive draw speed, but makes it less easy to preview Scraggle, Curl or Clumping.
- Since Shadow Maps are expensive to render, don't use more than you need. Autos shading is free and can provide an acceptable alternative for some lights when using the Maya Software renderer.

## Miscellaneous

- On Windows, fragmented hard drives can cause serious performance issues. We suggest you defragment regularly and thoroughly with a dedicated application. Standard tools offered by the operating system are often not sufficient to gain performance benefits.
- Take advantage of file referencing. For more information, see "About file referencing improvements" on page 15.
- Become very familiar with Windows > Settings/Preferences > Performance Settings.
- Use File > Optimize Scene Size >  to remove unused scene data.

## Modeling

- Use the Reuse Triangles attribute on polygons.
- Use Poly Reduce to simplify complex geometry. Polygonal models that don't have the following will display faster: unshared normals, unshared texture coordinates, unmapped faces, and faces that are not triangles (not triangulated).

## Dynamics

- Use the Stand In attribute on rigid body geometry.
- When starting or setting up a simulation, start with less geometry on particles.

## Cloth

- Turn off Cloth collisions for the initial setup and testing of a simulation.
- Animate settings like Frame Samples and Time Step Size to improve solve time.
- Solve in batch mode.

## Rendering

- Use Render Diagnostics for hints on improving performance. See *"Maya render diagnostics"* in the Maya Help for more information.
- Remove duplicate shading networks.
- See also *"Improved on demand translation"* on page 128.

## Artisan

- Increase Stamp Spacing if possible (Stroke section of Artisan Tool settings editor).
- Use Screen Projection if possible (Stroke section of Artisan Tool settings editor).

## Paint Effects

- Lower the Display Quality in the Paint Effects Tool settings editor.

## 5 | Performance tips

### > Optimize your scene for playback

## Optimize your scene for playback

This section describes in detail various ways you can increase the interactive and playback speeds of your Maya scene.

### Display and UI Elements

**Tip** Hiding any of the components (selection handles, locators, entire objects and so on) of your Maya scene will increase the scene's interactive speed. To hide components of your scene, use the options in the Display > Hide menu.

#### **Close any unnecessary windows**

- Close all Maya editors and windows you are not currently working with, especially the Graph Editor and Dope Sheet (which update during playback).
- Close as many windows from other applications as you can.
- Close UNIX shells, DOS windows, or Terminal windows.

#### **Hide UI Elements**

Select Display > UI Elements > Hide UI Elements.

This hides all panes in the Maya main window except for the scene views.

## 5 | Performance tips

### > Optimize your scene for playback

#### **Use alternate shading modes for your view**

In the Panel menu, select alternate shading modes from the Shading > Interactive Shading menu.

The performance of different shading modes from best to worst is:

- 1 Bounding Box
- 2 Points
- 3 Wireframe
- 4 Normal

#### **Use default lighting**

In the Panel menu, turn on Lighting > Use Default Lighting.

#### **Turn off hardware texturing and back face culling**

In the Panel menu bar, do the following:

- Turn off Shading > Hardware Texturing.
- Turn off Shading > Back Face Culling.

#### **Turn on fast interaction**

Turn on Display > Fast Interaction.

In this mode, objects are drawn at a resolution that is based on their screen size. In other words, the further an object is from the camera, the lower its drawn resolution.

## 5 | Performance tips

### > Optimize your scene for playback

#### **Put non-moving objects in Template mode**

Templated objects are not redrawn during playback. Also, templated objects will always be drawn in wire mode, even when the current view is shaded.

It is useful to put large, currently insignificant objects in the scene into template mode while working on other objects in the scene.

- 1** Select the non-moving object you want to template.
- 2** Select Display > Object Display > Template.

#### **Turn off grid display**

Select Display > Grid.

When the Grid menu item is off, the grid does not appear in the scene view.

#### **Turn off Time Slider ticks**

- 1** Select Window > Settings/Preferences > Preferences.

The Maya Preferences window appears.

- 2** In the Categories column, select Timeline.
- 3** In the Timeline section, set Key Ticks to None.

#### **Hide Time Slider**

Select Display > UI Elements > Time Slider.

When the Time Slider menu item is off, the Time Slider does not appear in Maya's main interface.

## 5 | Performance tips

### > Optimize your scene for playback

#### **Set Playback Update View to Active**

- 1 Select Window > Settings/Preferences > Preferences.  
The Maya Preferences window appears.
- 2 In the Categories column, select Timeline.
- 3 In the Playback section, set Update View to Active.  
Your scene's animation now plays back only in the active Maya scene view.

## Modeling

#### **Remove invalid geometry**

- 1 Select File > Optimize Scene Size > .  
The Optimize Scene Size Options window appears.
- 2 Turn on NURBS Surfaces + Curves, and then click the Optimize now button next to the option.

#### **Remove unused geometry**

- 1 Select File > Optimize Scene Size > .  
The Optimize Scene Size Options window appears.
- 2 Turn on NURBS curves and Locators, and then click the Optimize button.

#### **Reduce evaluations of NURBS objects**

Use the Display > NURBS Smoothness menu to reduce the number of evaluations along each span of a NURBS object.

Hull mode is the fastest to draw. The Rough, Medium, and Fine modes can be quickly accessed using the 1, 2, and 3 keyboard hotkeys.

## 5 | Performance tips

### > Optimize your scene for playback

#### **Precompute triangles for high resolution polygonal geometry**

- 1 Select the polygon mesh you want to precompute.
- 2 In the Attribute Editor, turn on Reuse Triangles in the Mesh Controls section of the *nShape* node. (Where *n* is the name of the poly mesh.)  
See "Precomputation of triangles for faster playback" on page 52.

## Skeletons

#### **Turn off joint display**

Select Display > Hide > Hide Kinematics > Joints.

## Skinning

#### **Remove unused skin influences**

- 1 Select your skinned object.
- 2 In the Animation menu set, select Skin > Edit Smooth Skin > Remove Unused Influences.

This disconnects all zero-weighted skin influences from your skinned geometry.

#### **Disable deformation of skin's user normals**

- 1 Select your smooth skinned object and open the Attribute Editor.
- 2 Select the skinCluster node's tab.
- 3 In the Smooth Skin Attributes section, turn off the Deform User Normals attribute.

## 5 | Performance tips

### > Optimize your scene for playback

#### **Display skinned geometry as wireframes**

In the Panel menu, select Shading > Wireframe.

This is useful when your skinned object is not triangle based.

## Deformers

#### **Decrease the wrap deformer's Max Distance**

The wrap deformer's playback speed and memory usage improves as you decrease its Max Distance.

We recommend that you use the lowest non-zero Max Distance value that still provides an acceptable wrap deformation on your geometry.

However, do not use a Max Distance value of zero, as it will disable the Max Distance option. Disabling this option reduces the playback speed of the wrap deformer and causes it to use more memory.

## 5 | Performance tips

### > Optimize your scene for playback

## Animation

### **Remove unused animation components**

**1** Select File > Optimize Scene Size > .  
The Optimize Scene Size Options window appears.

**2** Turn on:

- Animation Curves
- Animation Clips
- Poses
- Cached Data
- Deformers
- Expressions
- Constraints
- Pair Blends

**3** Click the Optimize button.

### **Delete all flat animation curves**

Select Edit > Delete by Type > Static Channels.

# Index

## Numerics

- 2D motion blur
  - Hair 135
- 3D Paint Tool
  - multiple textures 59

## A

- absolute
  - clip channel offset 82
  - Duplicate Absolute Settings 84
- Add Proxy 44
- Alias DirectConnect 65
- Alpha Blending 117
- Always Blend with Existing Connections 89
- ambient occlusion 111
- Ambient Occlusion shader 112
- animation
  - blend preferences 89
- animation detection 58
- anti-aliasing 119

## API

- compiler requirements 178
- examples of new plug-in functionality 177
- important changes to existing classes 153
- new classes and header files 151
- other significant changes 178
- performance improvements 151
- API example plug-ins
  - apiSimpleShape 177
  - pfxInfoCmd 177
  - viewCallbackTest 177
  - weightListNode 178
- apiSimpleShape 177
- approximation 110
- area light 128

## B

- Bake Alpha 116
- Bake Color 116
- bake-sets
  - ambient occlusion 111
- baking improvements 111
- Bevel 77
  - Merge Vertices 77
  - Smoothing Angle 78
  - Subdivide Ngons 78

## bind

- maintain settings after bind 97
- to non-joint objects 95
- Blackbody shader 112
- blend
  - Animation preferences 89
- Blend With All Except Constraints 90
- blur2d for Mac OS X 146
- Bone Radius Settings 99

## C

- CAD files
  - importing 65
- camera icon
  - scaling 69
- caustics
  - caustic scale 119
- Channel Box
  - how to make channels keyable/nonkeyable 87
  - keyable/nonkeyable 85
- Channel Control Editor
  - how to make channels keyable/nonkeyable 88
  - keyable/nonkeyable 86
- channel offsets
  - how to set 84

## Index

- channels
  - clip offsets 82
  - how to make keyable/  
nonkeyable 87
  - keyable/nonkeyable 85
- CIE D Illuminant shader 112
- Clamp Max 116
- Clamp Min 116
- classes, changed 153
  - M3dView 153
  - MAnimMessage 154
  - MFileIO 155
  - MFn 158
  - MFnAttribute 158
  - MFnClip 159
  - MFnDependencyNode 159
  - MFnMesh 160
  - MFnParticleSystem 163
  - MFnSkinCluster 166
  - MGlobal 167
  - MImage 167
  - MItMeshFaceVertex 170
  - MItMeshPolygon 169
  - MMatrix 170
  - MPlug 171
  - MPxCommand 172
  - MPxFileTranslator 172
  - MPxHwShaderNode 173
  - MString 176
  - MTypes.h 176
  - MUiMessage 176

- classes, new 151
  - MFnAnisotropyShader 152
  - MFnKeyframeDelta 151
  - MFnLayeredShader 152
  - MFnPhongEShader 152
  - MIteratorType 152
  - MPxComponentShape 152
  - Paint Effects 151
- clips
  - per channel offsets 82
- Closest In Hierarchy 96
- Cloth Collision Debug 136
- Cloth constraints 60
  - edit membership 142
- cluster deformers
  - playback improvements 53
- Collision Depth
  - visualize Cloth 136
- Collision Offset
  - Paint Collision Properties  
Tool 136
  - visualize Cloth 136
- collisions, debug
  - Cloth 140
- Color Blending 117
- Color Feedback
  - Artisan 58
- command line renderer 100
- compiler, API
  - requirements 178
- component editor
  - custom layouts 73
- constraints
  - Cloth 60
  - Lock Output 90

- constructors, new
  - MItDag 168
  - MItDependencyGraph 168
  - MItDependencyNodes 169
- copy smooth skin weights 92
  - how to 93
- Create Reference options 18
- Custom Entities 127
- custom layouts
  - component editor 73
- Custom Motion Offsets 124
- Custom Scene Text 129
  - Render Text 129
  - Root Group Text 129
- custom shelves 73

## D

- debug collisions
  - Cloth 140
- deformers
  - cluster playback 53
  - sculpt updates 53
  - soft modification tool 53, 56
  - wrap playback 52
- Diagnose Finalgather 126
- Diagnostics
  - Diagnose Finalgather 126
- Diffuse Bounces 126
- Dim Image 81
- dirty node callback, new
  - MNodeMessage 171
- displacement
  - approximation 109

- Displacement Overrides
  - Max Displace 126
- drawing
  - improved skeleton 52
- Duplicate Absolute Settings 84

**E**

- edges
  - soften extruded 74
  - soften split polygons 75
- Edit Constraint Membership
  - Tool
    - Cloth 136
- Edit Polygons
  - Bevel 77
  - Delete Faces 56
  - Delete Vertex 55
  - Extrude Edge 74
  - Extrude Face 74
  - Merge Vertices 55
  - Normals 56
  - Separate 56
  - Subdivide 55
- environment variable
  - expansion 70
- equalizer maps
  - Fur 133
- example plug-ins, API
  - apiSimpleShape 177
  - pfxInfoCmd 177
  - viewCallbackTest 177
  - weightListNode 178
- Export Tessellated
  - Polygons 121
- expressions
  - Cloth 136, 142
- extrude
  - soften edges/faces 74

**F**

- faces
  - soften extruded 74
- FCheck
  - Mac OS X command 146
- field constraints
  - Cloth 137
- file browse
  - environment variable
    - expansion 70
- file command 17
- file referencing
  - API update 17
  - Create Reference options 18
  - excluding attributes when
    - locking 18, 21
  - grouping improvements 18
  - locking and unlocking 18
  - MEL update 17
  - multilevel references 18
  - namespace editing 23
  - nested references 18
  - node architecture
    - updates 16
  - Preload Reference Editor 27
  - proxy referencing 27
  - proxy tags 29
  - Save Reference Edits 25
  - selective preload 17
- file texture 102

- file texture filter 129
- Final Gather 106
- Final Gather Cast 117
- Final Gather Receive 118
- Final Gather Reflect 130
- Final Gather settings 125
- framebuffer 130
- Freeze 125
- Fur
  - equalizer maps 133

**G**

- global illumination
  - global illum scale 119

**H**

- Hair
  - 2D motion blur 135
- hardware shader plug-in 102
- header files
  - MIOStreamFwd.h 152
- header files, new 151
- how to
  - copy smooth skin
    - weights 93
  - make channels keyable/
    - nonkeyable 87
- Hypershade 99
  - swatches 57

**I**

- IBL 110

## Index

image output formats 131  
image-based lighting  
  Render Stats 110  
  see IBL 110  
imgcvt for Mac OS X 146  
importing  
  CAD files 65  
increasing virtual memory 74  
influences  
  add at any pose 97  
  prevent attachment of zero-weighted 96  
instancing 57  
Interactive Photorealistic  
  Rendering  
  see IPR 131  
interlace for Mac OS X 146  
intersection  
  new MFnMesh methods 62  
IPR 131

**J**

joints  
  automatic radii 98

**K**

keyable  
  Channel Control Editor 86  
keyable/nonkeyable  
  channel states 85  
  set from Channel Box 85  
  set from Channel Control  
  Editor 86

## L

layer editor 68  
light icon  
  scaling 69  
Light Only 111  
locator  
  scaling 69  
Lock Output 90  
locking and unlocking file  
  references 18

## M

M3dView 153  
Maintain Max Influences 97  
MAnimMessage 154  
master machine 104  
Max Alpha 116  
Max Displace 126  
Max Object Samples 120  
Max Sample Level 118  
Maya browser  
  data from Maya 144  
Maya hardware rendering 102  
Maya vector rendering 102  
  resolution 102  
MEL commands  
  file 17  
  namespace 17  
  reference 17

mental ray for Maya  
  area light 128  
  file texture filter 129  
  framebuffer 130  
  image output formats 131  
  quality presets 118  
  region rendering 127  
  rendering 103  
  satellite rendering 103  
mental ray for Maya  
  rendering 57  
Merge Vertices 77  
methods, new  
  MFnMesh 62  
MFileIO 155  
MFn 158  
  iteration over several  
  types 62  
MFnAnisotropyShader 152  
MFnAttribute 158  
MFnClip 159  
MFnDependencyNode 159  
MFnKeyframeDelta 151  
MFnLayeredShader 152  
MFnMesh 160  
  faster mesh creation 61  
  new methods 62  
MFnParticleSystem 163  
MFnPhongEShader 152  
MFnSkinCluster 166  
MGlobal 167  
MImage 167  
Min Alpha 116  
Min Max Object Sample  
  Limits 118

Min Object Samples 120  
 Min Sample Level 118  
 MIOStreamFwd.h 152  
 MItDag 168  
 MItDependencyGraph 168  
 MItDependencyNodes 169  
 MIteratorType 152  
 MItMeshFaceVertex 170  
 MItMeshPolygon 169  
 MMatrix 170  
 MNodeMessage 171  
 modeling 54  
 Motion Back Offset 125  
 MPlug 171  
 MPxCommand 172  
 MPxComponentShape 152  
 MPxFileTranslator 172  
 MPxHwShaderNode 173  
 MRampAttribute  
     classes, changed  
     MRampAttribute 174  
 MSceneMessage  
     classes, changed  
     MSceneMessage 175  
 MString 176  
 MTimer  
     classes, new  
     MTimer 152  
 MTypes.h 176  
 MUiMessage 176

## N

namespace command 17

namespace editing 23  
 Never Blend with Existing  
     Connections 90  
 non-joint objects  
     binding 95  
 nonkeyable  
     Channel Control Editor 86  
 normal mapping 57

## O

Object Hierarchy 95  
 object instancing 57  
 object samples 119  
 object-specific render  
     attributes 117  
 Occlusion Mask 131  
 offset  
     clip channels 82  
 on demand translation 128  
 openEXR 131  
 Optimize Non-animated  
     Display Visibility 121  
 Optimize Raytrace  
     Shadows 122  
 options for proxy  
     referencing 42  
 Other Speed 91  
 outFoam attribute  
     Fluid Effects 133

## P

Paint Collision Properties Tool  
     Collision Offset 136

Paint Effects classes 151  
 Pass Custom Alpha  
     Channel 127  
 Pass Custom Depth  
     Channel 127  
 Pass Custom Label  
     Channel 127  
 pfxInfoCmd 177  
 photonvol flag 129  
 playback  
     cluster deformations 53  
     custom speed 91  
     faster skinned geometry 51  
     wrap deformations 52  
 plug-in manager  
     improved update 61  
 plug-ins, example  
     apiSimpleShape 177  
     pfxInfoCmd 177  
     viewCallbackTest 177  
     weightListNode 178  
 PNG 131  
 polygons  
     Cleanup 55  
     Combine 55  
     editing 54  
     loading large data sets 48  
     Mirror Geometry 55  
     Reduce 54  
     soften extruded edges/  
         faces 74  
     soften split 75  
 port number  
     satellite rendering 105  
 precomputation of triangles 52

## Index

- preferences
  - Animation Blending 89
  - maximum hardware texture resolution 57
- Preload Reference Editor 27
- presets 118
- Preview Convert Tiles 122
- Preview Tonemap Tiles 122
- proxy referencing
  - adding a proxy
    - reference 33, 36, 44
  - applications 28
  - description 27
  - options 33, 42
  - Proxy Tag options 31, 37
  - proxy tags 29
  - Reference Editor 29
  - Reload Proxy As 45
  - reloading multiple proxy references 39
  - Remove Proxy 44
  - removing a proxy
    - reference 38
  - workflow example 35
- proxy tags 29
  - Proxy Tag Options 40
- Prune Invisible Parts 121
- Prune Objects Without Material 121

## R

- Radius 99
- Rapid Scanline 108
- rebuild
  - merge 124

- reference command 17
- region rendering 127
- relative
  - clip channel offset 82
- Reload Proxy As 39, 45
- Remove Proxy 44
- Remove Unused Influences 96
- Render command 100
- render region 127
- Render Shaders With Filtering 122
- Render Stats 115
- rendering 57
- Reuse Triangles 52

## S

- sampling
  - object samples 119
- satellite rendering
  - master 104
  - master machine setup 104
  - mental ray for Maya 103
  - port number 105
  - slave 104
- Save Reference Edits 25
- Scale 125
- scaling
  - camera icon 69
  - light icon 69
- scrolling wheels 69
- sculpt tool
  - faster update 53
- sculpting
  - Artisan 58
- Selected Images 80
- Separate Shadow Bsp 120
- shader library 108
- shaders
  - library 108
  - Use Background 131
- Shading Quality 108
- Shadow Map Bias 124
- shadow maps
  - detail 123
- shadow maps rebuild
  - merge 124
- shared shelves 73
- sharpen
  - extruded polygon edges/faces 75
  - split polygons 75
- shelves
  - custom 73
  - shared 73
- skeletons
  - faster draw 52
- skin
  - add influences at any pose 97
  - bind to non-joints 95
  - faster playback 51
  - maintain smooth bind settings 97
  - prevent zero-weighted influences 96
- slave machine 104
- Smooth Polygon
  - Derivatives 121
- smooth skin copy weights 92

- Smoothing Angle 75, 78
- Soft Modification Tool
  - Use Distance Cache 53, 56
- soften
  - extruded polygon edges/  
faces 75
  - split polygons 75
- Solver Quality
  - Fluid Effects 133
- speed
  - custom playback 91
- Static Object Offset 125
- STEP 65
- Subdiv Details 67
- Subdivide Ngons 78
- subdivision surfaces
  - component display 50
  - display performance 56
  - improved
    - approximation 110
  - Subdiv Details 67

## T

- tagging proxy references 31
- Texture bake-set 115
  - ambient occlusion 111
  - Final Gather Reflect 130
- Time Scale
  - Cloth solver 136
- Time Step Size
  - Cloth solver 136
- Tonemap Scale 122

- Transmit Transparency 117
- Trax
  - Duplicate Absolute  
Settings 84
- triangles
  - precomputation 52

## U

- update
  - sculpt deformations 53
- Update Cloth State 137
- Use Background shader 131
- Use default material 50
- Use Distance Cache 53, 56
- Use Smooth Skin Weight  
Copy 92
- UV Texture Editor
  - Dim Image 81
  - performance  
improvements 54
  - Selected Images 80

## V

- Vertex bake-set 116
  - alpha blending 117
  - ambient occlusion 111
  - bake alpha 116
  - bake color 116
  - clamp max 116
  - clamp min 116
  - color blending 117
  - max alpha 116
  - min alpha 116
- View (Radii in Pixel Size) 125
- viewCallbackTest 177
- virtual memory
  - increase 74
- Visibility Samples 108
- Visible In Transparency 117
- Visualize Cloth collision
  - attributes 137
- Volume Samples 120

## W

- weightListNode 178
- weights
  - smooth skin copy 92
- wheel mouse 69
- wrap deformers
  - playback improvements 52

## X

- xpm2bmp for Mac OS X 146

## **Index**

What's New in Maya 6.5  
202