



Part I

Fundamentals of Blender 3D

- ◆ Chapter 1: Working in Blender
- ◆ Chapter 2: Working with Textures and Materials
- ◆ Chapter 3: Sculpting and Retopo Workflow
- ◆ Chapter 4: Rendering and Render Engines



Chapter 1

Working in Blender

Blender incorporates a dizzying amount of functionality in a single application, and learning to use all the tools as efficiently as possible is a daunting proposition. Even after the initial shock that every beginner feels upon seeing the properties window, experienced users often still sense that there is a great deal of potential that they have not fully tapped into. Indeed, many Blender users use only a small fraction of its capabilities for controlling their work environments. These capabilities include options available in the User Preferences window and a variety of lesser-known techniques and workflow shortcuts. Furthermore, by gaining insight into the design principles behind the Blender interface, you can prepare for the ways that new changes in the code base will help to enhance the power, flexibility, and accessibility of the interface in the future.

In this chapter, you will learn to

- ◆ Understand the difference between datablocks and objects
- ◆ Set the options available to you in the User Preferences window
- ◆ Use lesser-known methods for selecting, grouping, and organizing 3D elements to speed up your workflow

Data and Data Access

As I wrote in the introduction, this is a book for people who want to push the envelope of their Blender abilities—people who know how to use Blender but want to know more. Likewise, this is a chapter for people who know Blender’s interface and workflow but want to know it better, to understand it more deeply, and to learn to use it faster and more efficiently—in short, to master it.

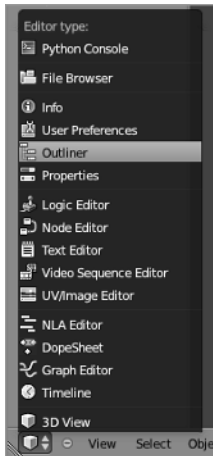
This chapter is intended to help you get beyond simply knowing how things are done in Blender and to truly explore the way *you* do things in Blender. In this chapter, you’ll learn about the preferences you can set to take control of your own working environment. You’ll learn about workflow tricks and techniques to give you more options for how to get from A to B in your Blender work. This chapter is intended to give you the knowledge and the confidence to start telling Blender how you want things done.

Understanding how Blender handles data internally is an important part of mastering Blender. Blender handles data in a unique way, and if you’re coming from other 3D packages, you may find it somewhat foreign. But once you understand it, it is fairly straightforward and enables Blender to deal with very complex data operations while maintaining consistency in the way you access and manipulate the data, whether directly through the Blender interface, through Python scripts, or through the proxy and linking system. The good news is that post 2.5 versions of Blender have much more informative interface tools for examining data. One of the most important is the Outliner.

Using the Outliner

The Outliner is Blender’s primary GUI-based data inspection tool (there are other important tools for inspecting data, but most of these are geared toward Python scripting uses. You’ll read about them in Chapter 13, “Python Scripting for Blender.”) In Blender’s default window configuration, an Outliner window is open in the upper-right corner. If you want to open a new one, simply choose Outliner from the Editor Type menu, as shown in Figure 1.1.

FIGURE 1.1
Opening an Outliner window



The default scene appears in the Outliner as shown in Figure 1.2. You can expand the items by clicking the plus icon to their left. The default view of the Outliner is All Scenes. In this view the scenes are listed and can be expanded to show their dependent data. (In the default case, shown here, there is only one scene, called Scene.)

FIGURE 1.2
The default scene in the Outliner



3D objects are branches off the main scene, and clicking the name of an object selects it. Datablocks used by the 3D objects are branches off those 3D objects. You can see in the figure that the Cube 3D object has a subordinate datablock called Cube representing its Mesh datablock. Below that is the Material datablock, and below that is a Texture datablock.

To the right of each 3D object menu item are three icons that can be clicked to toggle visibility, selectability, and renderability, respectively. If you click the eye icon for the Cube object, for example, the Cube will become invisible in the 3D Viewport. If you click the arrow icon, the Cube will become impossible to select. If you click the camera icon, it will not render.



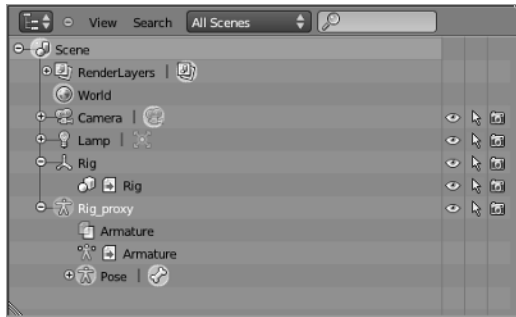
Library linked and proxied objects are also displayed in the Outliner scene display. In Figure 1.3, the Rig object is a linked object, indicated by the  icon, and the Rig_proxy object is a proxied object, indicated by the  icon.

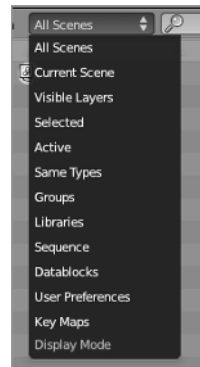
FIGURE 1.3
A proxied armature
in the Outliner



By right-clicking these Outliner items, you can bring up a menu of Outliner operations appropriate for the data type you clicked.

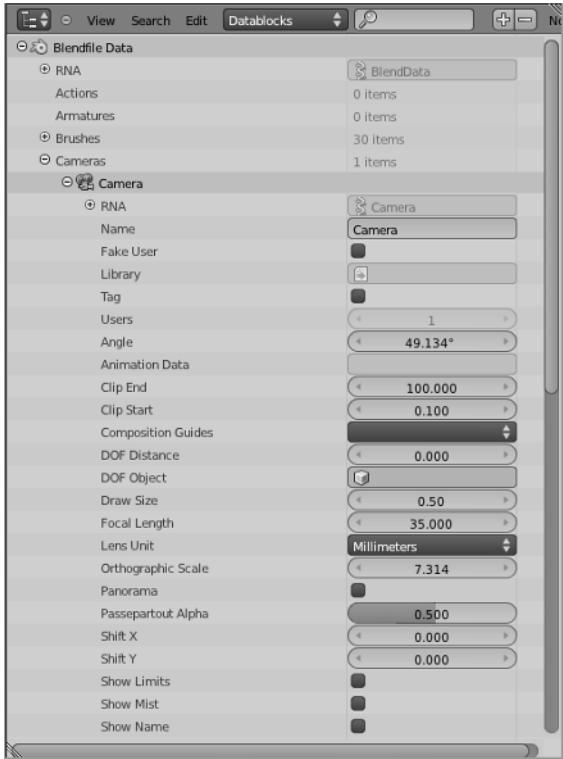
The Outliner can be set to show other types of data as well. The Outliner Display Mode menu is shown in Figure 1.4. These options enable you to display a variety of configurations; mostly these are self-explanatory. You can display active objects, selected objects, or all objects of the same type or group as selected objects. You can also view data other than 3D data, such as sequence data, user preference data, key map data, and library data.

FIGURE 1.4
The Outliner
Display Mode menu



For an even deeper look at the data, you can choose the Datablocks display mode, which enables you to inspect and all the properties of any datablock. In Figure 1.5, you can see the partial listing of the properties for the default Camera datablock as it appears in the Datablocks view of the Outliner. As you can see here, all of the information that you would ordinarily access through various other means via Blender's user interface is accessible as a concise listing of properties.

FIGURE 1.5
Viewing datablocks
in the Outliner



DATABLOCK “USERS” AND “FAKE USERS”

Datablocks hold data that is “used” by objects or other datablocks. These objects or other datablocks are called the datablock’s users. For example, if two 3D Mesh objects share the same Mesh datablock (which can be accomplished, for example, by doing a linked duplicate operation on a Mesh object with Alt+D), then that Mesh datablock has two users. If a material is used by three different objects, then that Material datablock will have three users. If an image is used in two different textures, then the Image datablock will have two users.

To see this in action, duplicate the default cube by pressing Shift+D to create two identical cubes, as shown in Figure 1.6.

Check the Material properties for one of the cubes. The material will be shared between the two objects and therefore will have two users. The count of users (2) appears to the right of the material’s name in the Material drop-down menu, as shown in Figure 1.7.

If you click the 2, a copy will be made of the material that is local to the current object. This creates a new material (automatically named Material.001), which is not shared between multiple users, so it no longer has a user count to the right of the material name, as shown in Figure 1.8.

FIGURE 1.6
Duplicating
the cube

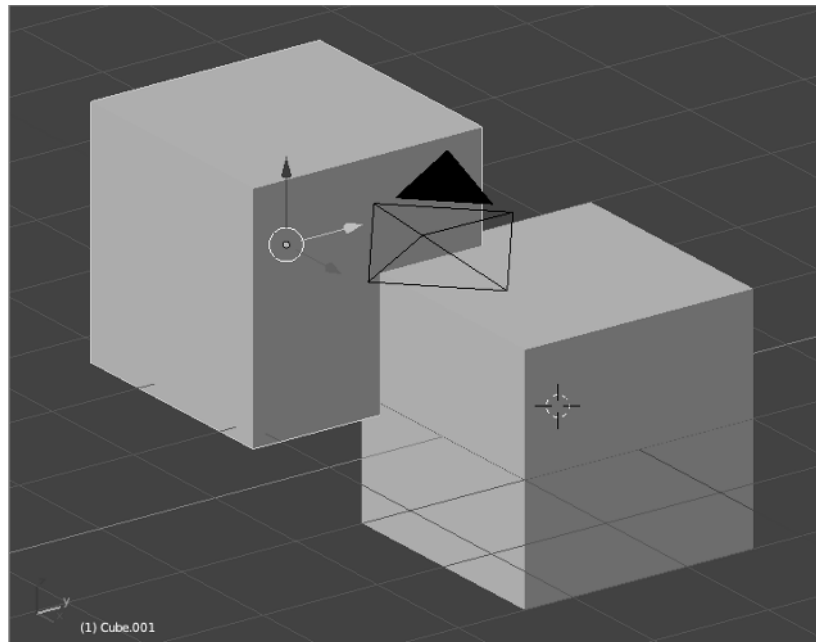


FIGURE 1.7
The Material
properties

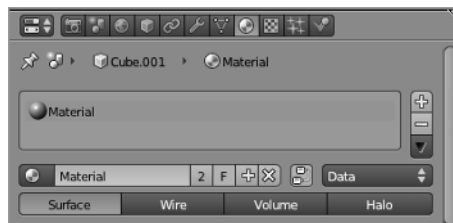
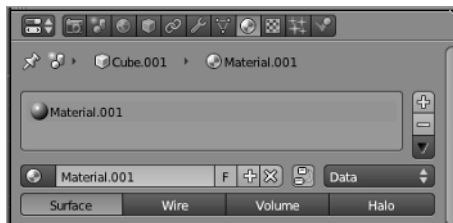


FIGURE 1.8
Creating a copy
of the Material
datablock



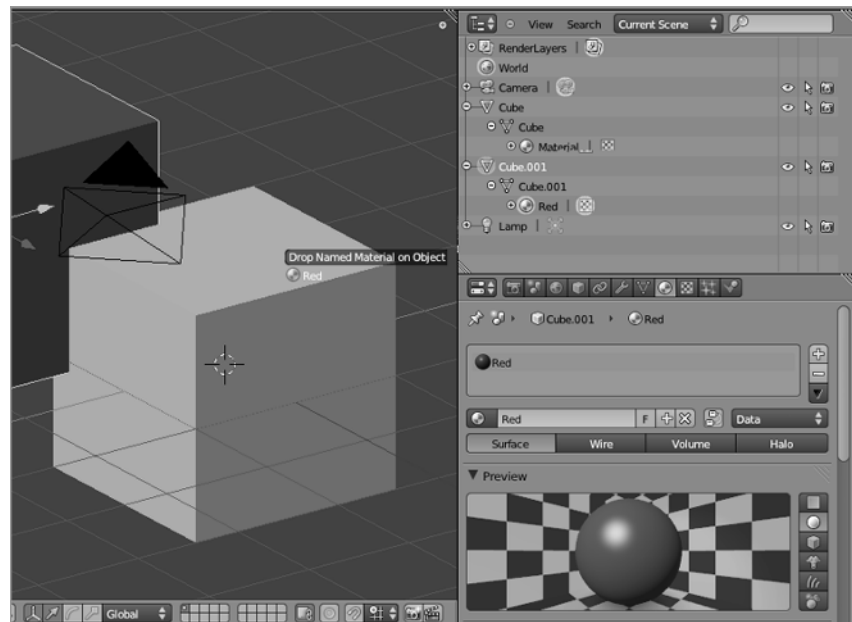
To distinguish the two materials more easily, make the new material red colored and rename it **Red**. Now, in the Outliner you should see the objects and materials displayed as shown in Figure 1.9.

FIGURE 1.9
Objects and materials in the Outliner



You can drag and drop materials directly from the Outliner onto objects in the 3D Viewport. Try dragging the Red material from the Outliner to the gray cube in the viewport, as shown in Figure 1.10. The gray cube will become red, and the two cubes will once again share a material (this time it will be the material Red).

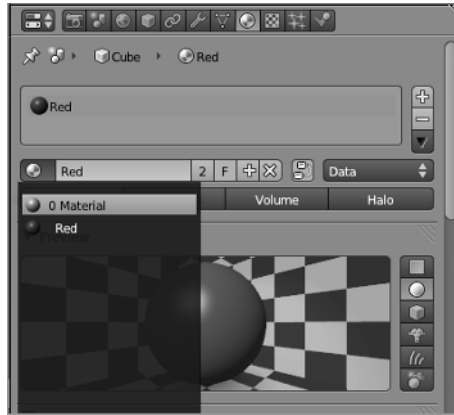
FIGURE 1.10
Dragging and dropping a material from the Outliner



If you now look in the Material drop-down menu on the Materials properties tab, you will see that the original material, Material, has a zero written to the left of the name in the menu, as shown in Figure 1.11. This indicates that Material has no users.

FIGURE 1.11

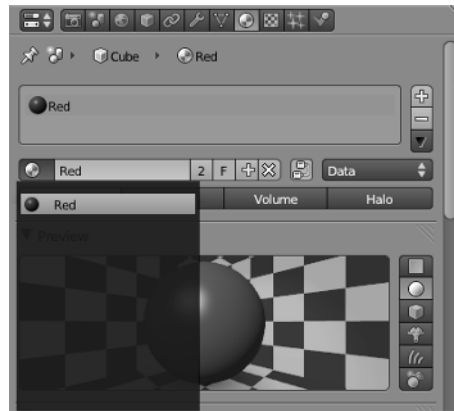
The menu entry
for a datablock with
no users



Datablocks with no users do not persist when you close and reopen Blender. If you save the file now and reopen the same file, then look again at the same drop-down menu, it will appear as shown in Figure 1.12. The Material entry will be gone. This is a form of Blender's garbage management, which keeps .blend files from becoming bloated with unused data. Because Blender does not explicitly delete a datablock, it can be frustrating to track down and delete every user. However, you can delete a datablock from *all* its users by holding Shift while clicking the X to remove it, thus guaranteeing its deletion when you save and reload.

FIGURE 1.12

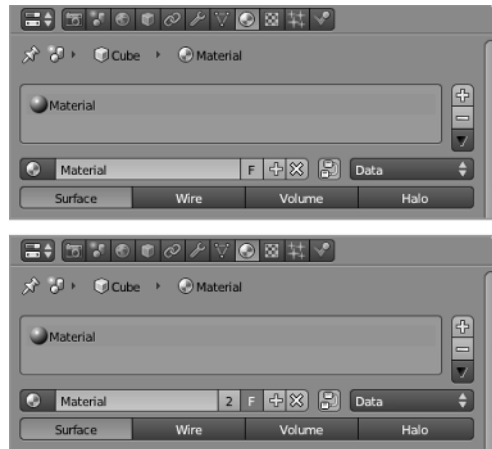
The menu
after closing and
reopening Blender



For a variety of possible reasons, you might want to have an unused datablock persist when you close and reopen a Blend file. This can be done by adding a *fake user* to the datablock. To add a fake user, click the F button to the right of the datablock drop-down menu. As you can see in Figure 1.13, a material datablock with only one user gains a second user when the F key is pressed.

FIGURE 1.13

Adding a fake user to a material datablock



The presence of a fake user appears as a check box in the Outliner datablock view (Figure 1.14). Unchecking this box in the Outliner has the same effect as clicking the F button to toggle off the fake user.

Understanding these basics about how datablocks in Blender operate, and how they can be accessed and inspected using the Outliner, will be helpful in all of your Blender work. In the next section, we'll turn to more superficial aspects of controlling how Blender works for you, specifically the user preferences that control the interface and environment.

DNA AND RNA

Blender uses a unique internal format called *DNA* to store and reference 3D assets. The name is an analogy to the biological term, with the implication that DNA is a highly compact encoding of all the information necessary to re-create the contents of what Blender users know as a *.blend* file: scenes, objects, and all associated datablocks. DNA is a binary format, which makes it very fast to load and save. For example, the same data represented in XML may be several orders of magnitude slower to load and save, particularly in the case of large files with many scene elements. This is the main reason why *.blend* files are so flexible and can be used to store large and complex scenes and even collections of scenes.

RNA is a current development that comprises an important behind-the-scenes component of the 2.5 changes. It is also loosely analogous to the biological meaning of the term. RNA will serve as a wrapper or low-level interface for accessing and setting values in DNA. In practice, RNA will be used to automatically generate interface elements and the Python API, making it easier to keep them up to date and consistent. The enhanced access that RNA enables also has the effect of finally realizing the long-held dream of having nearly everything in Blender capable of being animatable.

FIGURE 1.14
Material proper-
ties in the Outliner
datablock view



Exploring the User Preferences

When you think about options and customization for any software, the first thing that usually comes to mind is the set of user preferences available. Like most applications, Blender has a variety of user preferences that you can adjust. The User Preferences window is a window type like any other, but it can be accessed in a separate pop-out system window by pressing Ctrl+Alt+U. When you do this, it appears as shown in Figure 1.15.

FIGURE 1.15
The User Preferences pop-out window



Seven buttons are located along the top of the User Preferences window. Each of these buttons displays a different subcontext of User Preferences.

INTERFACE

The first subcontext of the User Preferences buttons area is the Interface subcontext, shown in Figure 1.16.

FIGURE 1.16
Interface user preferences



The Display options are buttons that control how information is displayed throughout the interface or in the 3D viewport. Those buttons are as follows:

- Tooltips** Enables and disables the display of tooltips when the mouse is over interface elements.
- Show Python Tooltips** Enables and disables the display of Python operator information with the tooltips when the mouse is over interface elements.

Object Info Displays the name of the active object in the lower-left corner of the 3D viewport.

Large Cursors Enables alternate mouse cursors if they are installed in your system.

View Name Displays the name of the view (Front, Back, Top, Bottom, Right, Left, Orthogonal, or Perspective) in the upper-left corner of the 3D viewport.

Playback FPS Displays the number of frames per second in the upper-left corner of the 3D viewport when the animation is playing.

Global Scene Causes the active scene to hold constant over various screens. If this option is enabled and the scene is changed in any one screen, all the screens will change scenes. If this option is disabled, a screen will continue to display the scene it last displayed, even if the scene is changed in another screen. You can set the size of Object Origin and the size and display of the Mini Axis here as well.

The next column of buttons and fields includes controls for view manipulation. The options available here are as follows:

Auto Depth Causes the rotation and zoom of the 3D space to pivot around the point directly under the mouse. This option automatically calculates the depth of the nearest object under the mouse as the pivot point.

Zoom To Mouse Position Enables you to control exactly what point in the space your view zooms in on when you rotate the mouse wheel, by positioning the mouse where you want to zoom.

Rotate Around Selection Causes the view to rotate around the median point between selected elements.

Global Pivot Causes the selected pivot to be fixed over all 3D viewport windows. If this option is not selected, each 3D viewport can use a different pivot.

Camera Parent Lock Causes objects above a camera in a parent hierarchy to be affected when the Lock Camera To View feature is activated. This can be useful when the camera is being controlled by camera rigs, because it enables the entire camera rig to be controlled by camera locking.

Auto Perspective Causes the view to enter Perspective view whenever it is rotated out of Front, Side, or Top view and to enter Orthogonal view when it enters those views by means of hot keys on the number pad.

Smooth View Sets a time interval in milliseconds for an animated transition between number-pad views.

Rotation Angle Sets the degree of rotation used by the 2, 4, 6, and 8 keys on the number pad to rotate the view incrementally.

Minimum Grid Spacing Determines the density of grids in terms of pixel spacing between grid lines for 2D viewports such as the Timeline, the Graph Editor, and the Dope Sheet.

TimeCode Sty Gives options for the writing style for time codes.

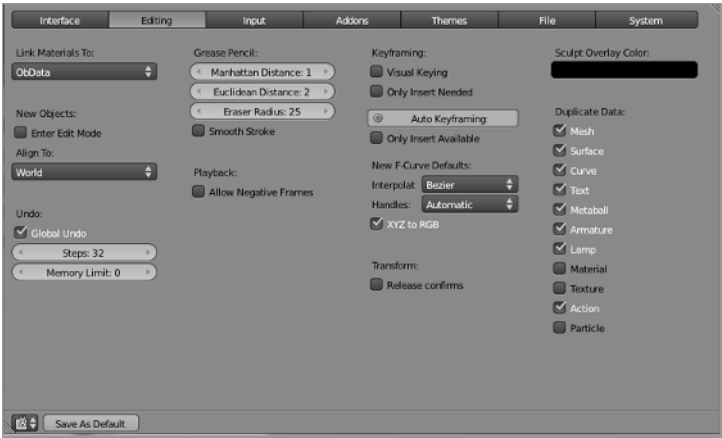
The options available in the third column of the Interface area are as follows:

- Manipulator** Toggles the 3D manipulator on and off. The Size, Handle Size, and Hotspot values determine how big the manipulator is, how big its handles are, and how big the area is where it is sensitive to being selected.
- Open On Mouse Over** Enables menus to open automatically when the mouse is held over them, without clicking. The numerical values for this option determine how long the mouse must be held over the main menu or submenus before the menus open.
- Show Splash** Determines whether the Blender splash screen is displayed on startup.

EDITING

The Editing user preferences context is shown in Figure 1.17.

FIGURE 1.17
The Editing user preferences



The options in the first column of this window are as follows:

- Link Materials To** Controls whether materials are linked to an object itself or the object's mesh datablock by default.
- New Objects** Enables you to choose whether to switch to Edit mode automatically upon object creation and whether newly created objects should be aligned to the view or placed at the 3D space origin with default orientation.
- Undo** Enables you to set the number of levels of Undo, the amount of memory devoted to Undo, and whether Global Undo is used. Global Undo requires more memory than regular Undo; however, regular Undo is limited in that you cannot undo edits made in Edit mode incrementally after leaving and reentering Edit mode. Global Undo enables you to do this.

In the second column, the options are as follows:

- Grease Pencil** Sets distance values that control how far you need to move the mouse in order to draw a segment of grease pencil line. The higher the distance values, the blockier the segments will be for the grease pencil. Default eraser radius can also be adjusted here. The Smooth Stroke option smooths grease pencil lines.

Playback: Allow Negative Frames Enables negatively numbered frames to be used in playback and on the timeline.

In the third column, the options are as follows:

Keyframing: Visual Keying Enables visual keyframing by default (you can keyframe visual keyframes ordinarily by choosing the Visual Loc/Rot/Scale options in the Insert Keyframe menu). Visual keyframing sets keyframes for object locations after constraints have been calculated. For example, if an object is parented to another object, and the parent object is offset, keying the visual location will key the parented object's position as it appears in the viewport, including the offset of the parent object.

Only Insert Needed Enables keyframing of necessary F-Curve values only. Values that have not changed from frame to frame will not be keyed.

Auto Keyframing Enables automatic keyframing as a default setting for new scenes (Auto Keyframing can otherwise be activated by clicking the red circle icon in the timeline header). With this activated, keyframes will be set for any object or bone that is transformed. Moving or rotating an object or bone will set a keyframe at the current frame.

Only Insert Available Restricts automatic keyframing to only inserting keyframes on F-Curves that have been previously created by manual keyframing.

New F-Curve Defaults Sets curve types and handle settings for newly created F-Curves. The XYZ To RGB option sets color coding for new F-Curves.

Transform: Release Confirms Sets transformations to be automatically confirmed when you release the mouse, in cases when transforming by dragging the manipulator.

Finally, the fourth column of the Editing preferences window contains the following settings:

Sculpt Overlay Color Sets the color used in the viewport overlay for the Texture brush in the Sculpt tools.

Duplicate Data Enables you to choose which datablocks are duplicated when you duplicate an object with Shift+D. Datablocks that are not duplicated will be shared by the new object (you can do this on a case-by-case basis by using Alt+D, in which case no datablocks are duplicated; all datablocks are shared by the new object).

RECOMMENDATIONS FOR EDITING OPTIONS

The best options for editing depend on the kind of work you do. If you typically find yourself going straight into modeling when you add a new object, you will save a step by setting the default to Switch To Edit Mode upon adding a new option. If you do a lot of animation and you are comfortable and confident working with lpos, enabling Auto-Keyframing may speed up your workflow. For beginning animators, I think it's better to set your keyframes deliberately by hand until you are sure you have the hang of it. For Keyframing, the Only Insert Needed option is useful to keep unnecessary keyframes from being set. For the Duplicate Data settings, if you find that you rarely want a duplicated object to share an lpo curve with the original object, you may want to select lpo in addition to the currently set defaults.

INPUT

The Input user preferences context is shown in Figure 1.18.

FIGURE 1.18
The Input user preferences



From here, you can control the following options:

Presets Enables you to choose from available sets of interface presets. The options currently available are Blender and Maya settings.

Emulate 3 Button Mouse Enables Alt+RMB to emulate the behavior of the middle mouse button (MMB).

Continuous Grab Enables the mouse movement to still be registered for certain purposes even when you move the mouse beyond the edge of the Blender window you are working with.

Select With: Left/Right Causes the left mouse button (LMB) or the right mouse button (RMB) to be used for selecting. Certain functions (such as Emulate 3 Button Mouse) are only available with RMB selection.

Double Click: Speed Controls how quickly you need to double-click the mouse for Blender to interpret it as a double-click event.

Emulate Numpad Enables you to use the number keys on the main keyboard instead of the number keys on the number pad. This is particularly useful if you are working on a laptop that doesn't have a separate number pad.

Orbit Style: Turntable/Trackball Selects between the Turntable and Trackball styles of rotating the view. Turntable causes the entire view to rotate strictly around the three spatial axes, resulting in a more constrained rotation than the Trackball option. Trackball style causes the entire view to rotate freely in all directions, analogously to the motion of a trackball. Blender's traditional default is Trackball.

Zoom Style: Continue/Dolly/Zoom Selects the way the scene zooms in and out. The Continue option causes the view zoom to continue forward or backward as long as the

left mouse button is held down and the mouse is moved above or below the center of the viewport. The distance of the mouse from the horizontal center of the viewport determines the speed with which the zoom moves forward or backward. The Dolly option causes the zoom to move forward when the mouse movement is downward and to move backward when the mouse movement is upward, by default. The Scale option causes the zoom to move forward when the mouse is pulled away from the center point of the viewport and to move backward when the mouse is pushed toward the center point of the viewport.

Vertical/Horizontal Causes the view to zoom in and out depending on the vertical or horizontal movement of the mouse. In Horizontal mode, the default direction is to zoom out with rightward mouse movement and to zoom in with leftward mouse movement.

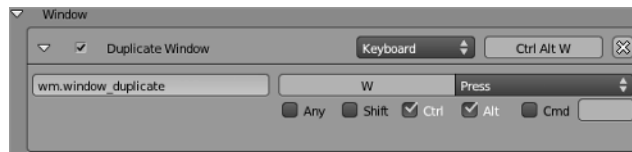
Invert Zoom Direction Causes the view to zoom forward when the mouse is moved upward and to pull away when the mouse is moved downward across the 3D view (as opposed to the default behavior, which is the reverse of this). In Horizontal mode, rightward movement zooms in and leftward zooms out.

Invert Wheel Zoom Direction Causes the view to zoom forward when the mouse wheel is rolled upward and to pull away when the mouse wheel is rolled downward (as opposed to the default behavior, which is the reverse of this).

NDOF Device: NDOF Sensitivity Controls the sensitivity of N-Degrees of Freedom devices such as 3D mouse devices.

The second column, which occupies the remainder of the window, enables you to control your hot key configurations precisely. At the top of the area is a drop-down menu enabling you to select Blender or Maya default hot key configurations. Clicking the white triangles to the left of the setting entries reveals hot key controls. An example of a detailed hot key setting is shown in Figure 1.19.

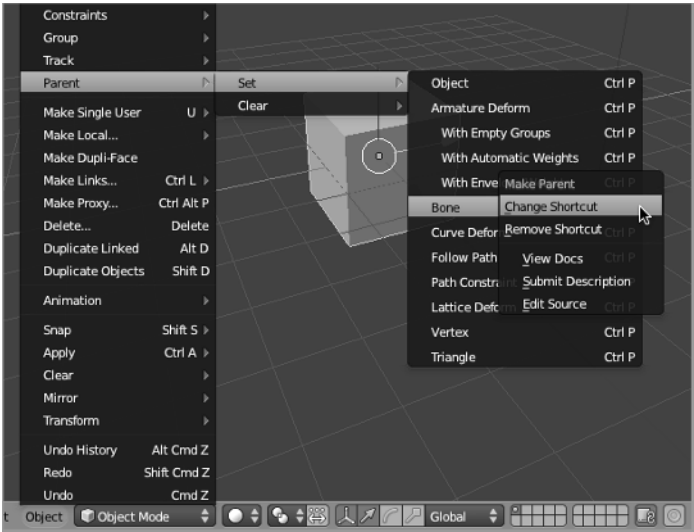
FIGURE 1.19
A hot key setting



Ctrl+Alt+W is the default hot key for opening a duplicate of the entire Blender window in a separate operating system window (this is useful for multi-monitor setups, for example). Figure 1.19 shows how this is set up. The drop-down at the top of the settings area indicates that the control is a Keyboard control. The RNA operator is written in the operator field, and W is entered in the key field. The Press option indicates that the operator is responsive to pressing the key. Modifier keys are indicated as check boxes below. All fields can be edited for complete control over your hot keys for any operators. Hot keys can also be added for operators that do not have hot keys by default.

Additionally, if you find yourself using a particular menu command, you can right-click any command (such as Mesh > Edges > Edge Slide) to add or change its shortcut. In Figure 1.20, you can see this method being used to change the hot key for setting a bone parent.

FIGURE 1.20
Right-clicking a menu to change the shortcut for setting a bone parent



ADDONS

The Addons user preference window is shown in Figure 1.21. Blender addons enable functionality that is not present by default in Blender. You can search for and select addons by area of functionality. Throughout this book there will be examples of addons in action.

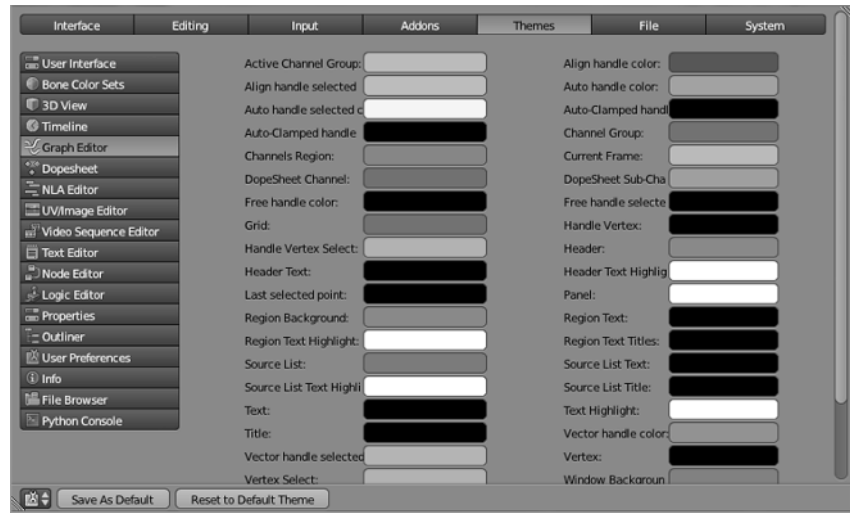
FIGURE 1.21
Addons



THEMES

The Themes context, shown in Figure 1.22, enables you to create and select themes with various options for the coloring and display of interface elements.

FIGURE 1.22
The Themes context



There are too many options to set in the Themes area to describe each one individually here, but they are mostly self-explanatory. You can change the color of almost every element in Blender, and in some cases such as drop-down menus and pop-up panels, you can change the alpha value as well.

Numerous Blender themes are available online for download. A quick Google search on *Blender themes* will give you the links for several good theme repositories. The themes may be downloadable in the form of a .blend file or in the form of a Python script. In the latter case, simply open the script in a Blender text editor window and execute it with Alt+P.

THEME RECOMMENDATIONS

Themes are a matter of taste; however, there's a reason why the two themes included in the default installation are largely gray and muted. Bright, lively colored themes can distract you from what you are working on and can lead to eyestrain. You should have enough contrast between elements to see them clearly, but large areas of white or very bright colors can quickly tire your eyes. Other theme options worth noting are those in the 3D View menu list. If you are planning to use Blender for instructing others or for making tutorials, you can change the size at which vertices and face dots are displayed.

FILE

The File user preferences, shown in Figure 1.23, are self-explanatory. The File Paths preferences enable you to define defaults for what the Blender file browser will open first when you import or save various types of assets. The default is //, which is Blender notation for the present working directory—that is, the directory you opened Blender from. For example, if you are opening Blender from the Windows Start menu, this will be your Blender installation directory. If you are opening from a file, this will be the directory that the file is in. The Relative Paths check box causes the file paths to be read as relative to the present working directory.

FIGURE 1.23
The File user
preferences



Preferences for saving and loading can also be found in this context. Here, you can set your preferences for how the Auto Save and backup features work. The Save Versions number enables you to select how many previously saved versions you want to keep backed up. In the past, you may have noticed the `filename.blend1` files in the directory alongside `filename.blend` files. These are the default single-version backup files, and they represent the contents of the previously saved session. If you select a value greater than 1 (and apply it with Ctrl+U), the correspondingly numbered backup versions will appear in your directory.

The Auto Save Temporary Files option enables numbered, automatically saved files to be saved to your temporary directory (the default is `/tmp`, so ensure that this directory exists on your system or else change the directory to wherever you want the files saved). The Timer value is how often these files are saved.

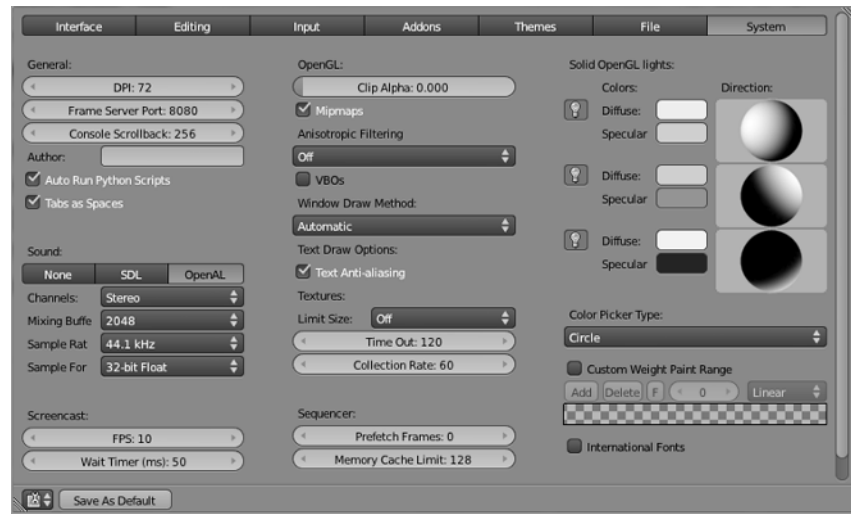
Setting up a directory for your Blender files is a good way to stay organized. In your Documents folder or similar directory, create a folder called **Blender**, including subfolders called **Fonts**, **Textures**, **Render**, and **Sounds**. Setting these folders to your default pathways will keep your files organized, and when working with others a unified directory structure helps sync the project.

SYSTEM

The System user preferences context, shown in Figure 1.24, enables you to control a variety of display-related and miscellaneous values.

Auto Run Python Scripts, when enabled, will allow Python scripts to be run automatically from within a `.blend` file. This is convenient in some cases, but it is not recommended if you're not sure of the source of your `.blend` files.

FIGURE 1.24
The System user
preferences



The Sound buttons enable you to select sound libraries and set the amount of memory to devote to audio mixing.

The buttons and fields in the middle column control a variety of specific values that you can adjust to improve performance if you are having memory problems or you are experiencing slowdowns in your 3D viewport. Disabling Mipmaps can speed up the OpenGL drawing in your viewport at the expense of some image quality.

When dealing with alpha textures in an Open GL drawing, Blender creates an imperfect draw of assorted objects. A plane with an alpha gradient will show objects below it as if it were transparent. However, objects behind it will be “clipped” by the extent of the mesh rather than the extent of the alpha texture. When Clip Alpha is set to .5, Blender will choose to clip by the texture whenever the alpha is less than 50%, but it also won’t show the alpha gradient beyond that. Setting Clip Alpha to .995 will almost completely reduce the alpha texture whenever a pixel has any transparency, which is useful in certain modeling situations, and setting it all the way to 1 is the same as setting it to 0. For accurate alpha judgments the default of 0 is preferred, but for modeling with lots of alpha textures .5 is recommended.

Three OpenGL lights can be used to illuminate objects in the Solid Draw mode. By default, two of these lights are activated. The first is a key light from the left, and the second is a dimmer fill light from the right. A third light is also available, which by default is set to provide high-lights from the lower right. You can enable or disable each of these lights, adjust their colors, or change their angles by clicking and dragging directly on the preview spheres for the lights.

The color-band widget under Color Range For Weight Paint Range check box enables you to override the default blue-to-red coloring range for weight painting and to define your own range when the box is checked.

Clicking the check box for International Fonts enables non-English interface options. There has been a significant push in internationalization recently.

Other Options

Many other options are available throughout the Blender interface, and it is worthwhile to make a note of the ones that you often find yourself adjusting and to use Ctrl+U to set them as you prefer them once and for all. The Occlude Background Geometry option in the 3D view header is a common option to activate. This makes unseen vertices and faces unselectable when not in Wireframe Draw mode, creating a sharper distinction between selection behavior in Wireframe and Solid Draw modes. If you usually rotate, grab, and scale using the R, G, and S keys, you may want to disable the manipulators, also in the 3D view header. Render settings such as the output format and compression quality are also common places where you might want to customize your defaults.

SAVING THE CHANGES

After you have set all the options the way you want them, don't forget to set the current setup as your default setup by using Ctrl+U. Remember, Ctrl+U saves the exact state of Blender at the moment you press it, so be sure you've put everything in place exactly the way you want to see it when you open Blender. Objects, materials, animations, and any other data in the `.blend` file will also be saved.

The resulting settings are stored in the `.B.blend` file in your `.blender` directory. To use these settings with another Blender installation, you can simply copy that file into the `.blender` directory of the Blender installation you want to use.

To save the current theme in the form of a Python script, select the File menu and choose Export > Save Current Theme. The resulting script can then be executed in another instance of Blender to import the theme.

Improving Your Workflow

Setting and saving your user preferences is the first step in optimizing your workflow. This section presents a variety of miscellaneous tips and tricks that you may find helpful for increasing your efficiency and improving your experience working with Blender.

Viewport Hot Keys and Properties

The 3D viewport has a number of hot keys and properties associated with it that enable you to view your work.

Common Number Key Shortcuts You are no doubt familiar with the most commonly used number pad shortcuts for Front view (number pad 1), Side view (number pad 3), and Top view (number pad 7). Pressing these keys with Ctrl will show you the reverse view; Ctrl+number pad 1 yields the rear view of the object, and so on. Number pad 5 toggles Orthogonal and Perspective views; and 2, 4, 6, and 8 rotate the 3D space by the amount determined in the Rotation Angle field in the Interface user preferences window.

Decimal (.) Key The decimal (.) key on the number pad centers the selected object in the 3D viewport. Related keys on the main keyboard include the C key, the Home key, and the Shift+C key combination.

C Key The C key shifts the view so that the 3D cursor is centered.

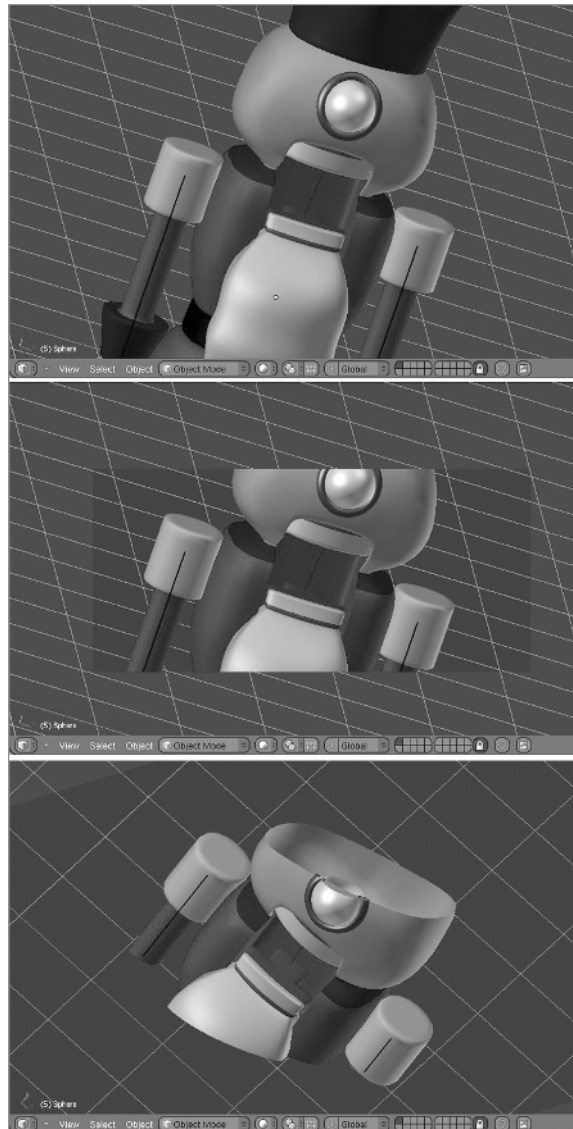
Home Key The Home key displays and centers the median point of all the objects in the scene.

Shift+C The Shift+C key combination does the same thing as the Home key with the addition of placing the 3D cursor at the zero point of the 3D space.

Slash Key (/) The slash key (/) on the number pad changes the display to show only the selected objects. Pressing the same key again toggles back into full scene display mode.

Alt+B On the main keyboard, the Alt+B key combination enables you to select even smaller portions of the 3D view for display. Pressing Alt+B and dragging the box to select an area results in clipping the display of everything outside that box selection, as shown in Figure 1.25. The resulting displayed selection can be viewed from all angles.

FIGURE 1.25
Clipping the view
with Alt+B



3D View Properties The 3D View Properties panel can be accessed by pressing N when your cursor is over the 3D Viewport. View settings on this panel are shown in Figure 1.26, and Display settings are shown in Figure 1.27. In these panels you can control the display and qualities of the background grid; the x-, y-, and z-axes; and the relationship lines (dotted lines between parents and their child objects).

FIGURE 1.26
View settings in
the 3D View
Properties panel

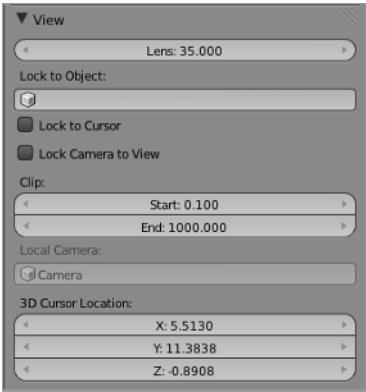
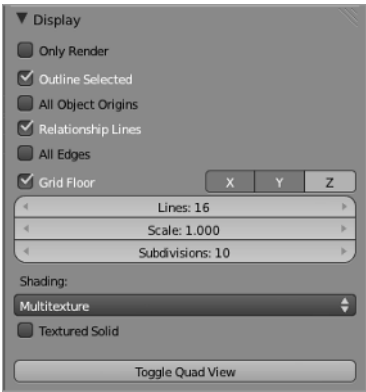


FIGURE 1.27
Display settings
in the 3D View
Properties panel



Solid Tex You can toggle the drawing of textures in Solid Draw mode with the Solid Tex button and toggle between displaying all object centers or only the selected object's center. You can toggle the drawing of an outline around the selected object. You can change the angle of the view lens, adjust the point past which the view is clipped, and place the 3D cursor by entering coordinates by hand. If you are modeling micro-details that require you to zoom in close on a model, you should set the view clip start to .001, and if you are navigating a huge, sprawling scene, you should change the clip end to 10,000 or more.

View Locking View Locking enables you to enter an object name (and in the case of an armature, a bone name) and force the view to follow the movement of that object, holding the object in the center of the view. This can be useful when you're animating detail on moving objects, such as when you're animating the movement of fingers on a moving hand.

Lock Camera To View Lock Camera To View makes your camera follow regular view navigation while in Camera View (numpad 0). This makes for easy positioning of camera angles, but make sure to turn it off once you have the perfect rendering view.

Grouping and Selection

Objects can be grouped by selecting the object and choosing a group from the Add To Group drop-down menu in the Object And Links panel of the Object buttons area. Objects that share a group can be appended into other .blend files in one step by appending the group. When lamps are grouped, it is possible to restrict a material's lighting to lamps from the group by entering the group name in the GR field in the material's Shaders tab.

Note in the Outliner that groups do *not* create hierarchy. The group does not represent a parent/child relationship, only a shared attribute. You can change the Outliner view to Group to sort objects by their groups.

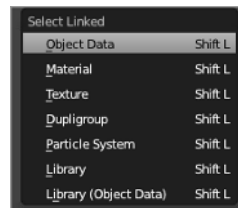
Groups are one of many criteria by which you can select objects. You can select variously grouped objects by selecting a single object and pressing Shift+G to open the menu shown in Figure 1.28. You can select other objects based on their relationship with the first selected object.

FIGURE 1.28
The Select
Grouped menu



You can also select objects based on linked data, by pressing Shift+L to open the menu shown in Figure 1.29 and selecting the linked datablock upon which to base the selection.

FIGURE 1.29
The Select
Linked menu



Using the Select menu in the 3D viewport in Object mode, you can directly select objects by type or by layer. It is also possible to select a random collection of objects and to invert the current selection.

BORDER, CIRCLE, AND LASSO SELECTION

Pressing the B key once initiates the Border selection state, where you can drag your mouse to select whatever falls within the rectangular area you define. Holding down the Alt key while doing this will deselect whatever falls within that area. Pressing the B key twice will enable the Circle selection state, where you can drag your mouse to select all that falls within a circular area following the mouse. Likewise, holding down the Alt key while doing this will deselect the elements.

Holding down the Ctrl key while dragging the left mouse button introduces the Lasso selection state, which enables you to define the area to be selected by moving your mouse around the area directly. This is a very fast selection method.

EDIT MODE SELECTION

Numerous little-known selection methods are available for meshes in Edit mode. The first option you have is whether to select by vertex, edge, or face by using the viewport header buttons shown in Figure 1.30 (both default and alternate icon sets). Vertex, Edge, and Face selection modes correspond with the buttons from left to right (the rightmost button occludes hidden geometry). You can choose more than one mode simultaneously by holding down the Shift key when you click these buttons. You can also switch between these three modes with Ctrl+Tab in Edit mode.

FIGURE 1.30

Buttons for choosing the selection mode



Selecting Edges, Loops, and Rings

Many selection options are available independently of the specific selection mode you are in. Selection options that deal specifically with edges can be found by pressing Ctrl+E in Edit mode. The Region To Loop selection option in that menu enables you to choose the edge outline (strictly speaking, *loop* here is a misnomer) of any selected region of faces, as shown in Figure 1.31 (this image is repeated for visual clarity in the color insert of this book). The reverse, selecting a region of faces based on a selected closed edge border around the region, is possible with the Loop To Region menu entry.

Other very useful selection options include loop and ring selection using Alt+RMB and Ctrl+Alt+RMB. By holding down the Alt key and right-clicking a single edge in Edit mode, you can select the entire edge loop that the edge belongs to. By using Ctrl+Alt+RMB, you select the perpendicular ring of faces that includes the edge you clicked on, as shown in Figure 1.32.

In Edge selection mode, the behavior is similar, except that the edge ring selected with Ctrl+Alt+RMB does not include faces, as shown in Figure 1.33.

FIGURE 1.31
Choosing a loop
from an area

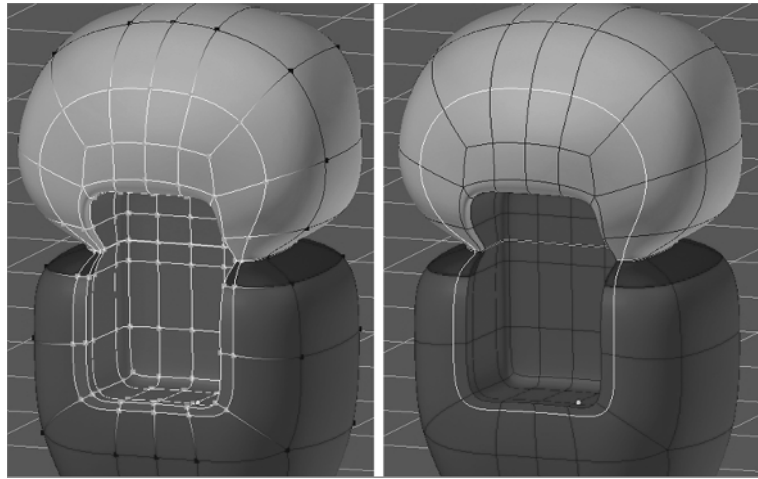


FIGURE 1.32
Edge loop and ring
selection in Vertex
selection mode

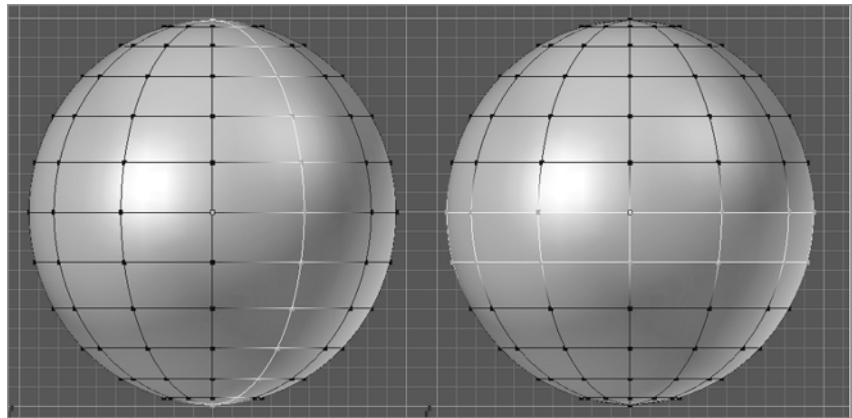
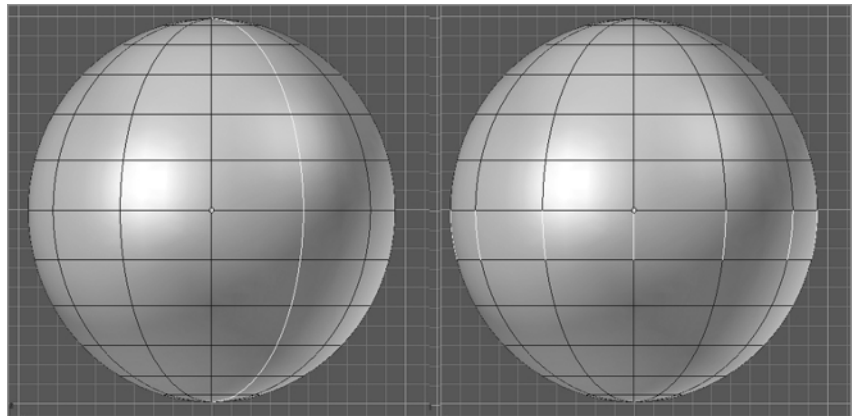


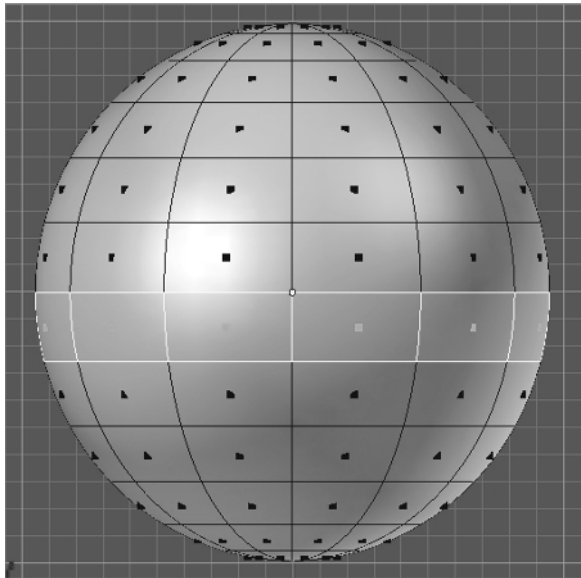
FIGURE 1.33
Edge loop and ring
selection in Edge
selection mode



In Face selection mode, there is no difference between the selections. Both hot keys select the same ring of faces, as shown in Figure 1.34. These figures are also included in the color insert of this book for visual clarity.

FIGURE 1.34

Loop selection in
Face selection mode



Another useful selection tool, Select Vertex Path, can be found in the Specials menu by pressing the W key when your cursor is over the 3D viewport. With exactly two vertices selected, this option will select the shortest edge path between the two vertices.

Selecting Similar Elements

Pressing Shift+G opens a menu that enables you to select all similar elements to the currently selected element, based on a variety of possible criteria.

In Vertex selection mode, pressing Shift+G enables you to select other vertices that share the same vertex normal direction as the currently selected vertices, vertices that are members of shared vertex groups with the currently selected vertices, or vertices that are used by the same number of faces.

In Edge selection mode, pressing Shift+G brings up a menu that enables you to select edges that are the same length, run in the same direction, or have the same number of face users as the selected edges. You can also select edges based on whether they are part of a seam or crease or based on their sharpness value. This is an excellent method for quickly selecting all seams on an object: Simply select a seam edge, and then use this selection method to select them all.

In Face selection mode, you can select faces that share the same area, share a material, share an image, have common normal directions, or are coplanar, meaning that the faces share their normal directions and are located on a single imaginary plane in the 3D space. Finally, the Perimeter option enables you to select regions of faces that have the same size perimeter or outline as the originally selected region.

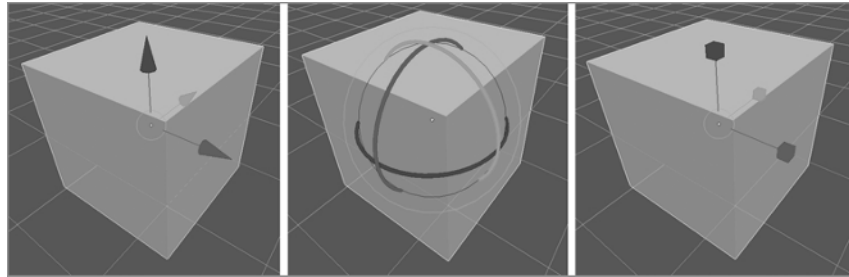
When a mesh has multiple materials assigned to different faces, the Shift+G menu enables you to select faces with the same material.

Object Manipulation

The most commonly used and taught methods of translating, rotating, and scaling 3D elements are the hot keys G, R, and S. These are the easiest to control, but using other methods can increase the speed and efficiency of your workflow in some cases.

It may come as a bit of a surprise, but the 3D manipulator widgets, shown in Figure 1.35 (and repeated in the color insert of this book), also require a little practice to get a feel for using them properly. These can be enabled individually or all at once using the manipulator buttons on the 3D viewport header (to select more than one, hold down the Shift key while choosing, just as in other contexts).

FIGURE 1.35
Translate, rotate,
and scale manipula-
tor widgets



The easiest way to use the manipulator widgets is to left-click the colored manipulator widgets (the arrows for translation, curves for rotation, and cube-shaped tips for scale) and drag. The transformation is finalized when you release the left mouse button. To abort the transformation, either press the Esc key or click the right mouse button *before releasing the left mouse button*.

Finally, you can scale or translate along two axes by holding down the Shift key and clicking the hot spot of the third axis. This is analogous to the way axes are constrained by hot key. Thus, to scale along the x- and y-axes, hold down Shift and click the z-axis manipulator hot spot. The hot spot is the arrow, box, or curve part of the manipulator for translate, scale, and rotate transformations, respectively.

THE DEVELOPING WORLD

As development on each Blender release intensifies, the #blendercoders IRC channel and the various development-related mailing lists are filled with developers communicating their ideas and intentions with each other. Every release requires a high degree of organization and coordination.

The smooth progress of Blender's development is all the more remarkable considering what a truly global project Blender is. According to the open source software resource Ohloh.net, Blender's regular committing developers are spread all over the globe—in Europe, North America, South America, Oceania, Asia, and Africa. The dream of a good non-English Blender interface is getting closer, but for the present, it's probably still best to stick with English.

Some of the stories of Blender development around the world serve as inspiring reminders of the power of open-source software. The work of Raúl Fernández Hernández (Farsthary) on true volumetrics for Blender is an excellent example. As a student living in Cuba, Raúl has had limited access to many of the resources that people in other parts of the world take for granted. Nevertheless, he identified a glaring need in Blender for true volumetric simulations and took advantage of the open code to study for himself how to implement his ideas in Blender. Although he lacked regular access to an Internet connection and was unable to access the Subversion code repository directly, participate in chats, or take part in regular communication with developers, he nevertheless succeeded in creating an impressive foundation for true volumetrics. He reported about his work sporadically in his blog, <http://farsthary.wordpress.com/>, including some amazing renders and animations of convincing flame and smoke effects. Although initially carried out with very little interaction with others, Raúl's work quickly began to get attention from the Blender user and developer community. After hurricane Gustav devastated his town, leaving him without electricity for a week, the community rallied to assist him, and two core Blender developers, Matt Ebb and Daniel Genrich, became more actively involved in helping him recode the volumetric simulation to be more consistent with existing Blender code. The project's completion put Blender among the best volumetric simulations on the market. Since then, Farsthary worked on an Unlimited Clay sculpting mode in Blender, which led him to a position on the team developing the sculpting software 3D Coat.

The Bottom Line

Understand the difference between datablocks and objects. Mesh datablocks and Mesh objects present a clear example of the distinction between 3D objects and their associated datablocks.

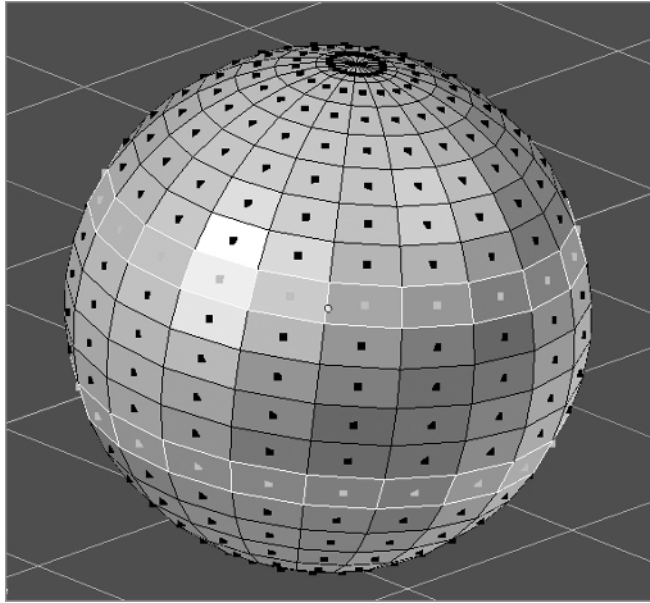
Master It One way to associate two different 3D objects with the same Mesh datablock is to create independent 3D objects and then select the same Mesh datablock from the Mesh drop-down menu. If you do this, you will wind up with an extra unused Mesh datablock that will disappear when Blender is saved and reopened. Find another way to accomplish the same thing, but without leaving any unused Mesh datablocks lying around and without needing to reopen Blender to clear it away.

Set the options available to you in the User Preferences window. A wide variety of often-overlooked options are available in the User Preferences window, including settings for views, controls, edit methods, and themes, among others.

Master It Create your own preferred default starting state and save it so that it will be active every time you start Blender.

Use lesser-known methods for selecting, grouping, and organizing 3D elements to speed up your workflow. There are numerous ways to select and group objects and 3D elements that can considerably increase your speed and efficiency when working.

Master It Use the selection methods described in this chapter to make the face selections shown in the following graphic.



You should be able to make this selection using a single (modified) mouse click followed by a single hot key combination. There are several ways to do this.

