



Chapter 1

Getting Started

In this chapter, you will be introduced to the concept of parametric 3D design and the general tools and interface of the Autodesk® Inventor® program. This chapter will focus on the concepts of parametric modeling and the workflow, tools, and interface elements in the Inventor software that are used to turn your ideas into a design.

In this chapter, you'll learn to

- ◆ Create parametric designs
- ◆ Get the “feel” of Inventor
- ◆ Use the Inventor graphical interface
- ◆ Work with Inventor file types
- ◆ Understand how project search paths work
- ◆ Set up library and Content Center paths
- ◆ Create and configure a project file
- ◆ Determine the best project type for you

Understanding Parametric Design

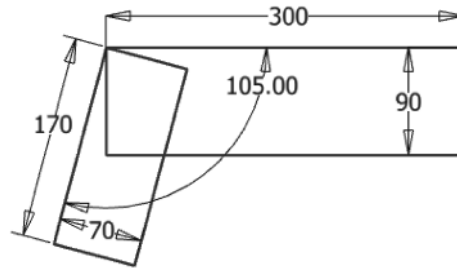
Autodesk Inventor is first and foremost 3D parametric modeling software. And although it has capabilities reaching far beyond the task of creating 3D models, it is important for you to understand the fundamentals of parametric 3D design. The term *parametric* refers to the use of design parameters to construct and control the 3D model you create. For instance, you might begin a design by creating a base sketch to define the profile of a part and then use dimensions as parameters to control the length and width of the sketch. The dimensional parameters allow you to construct the sketch with precise inputs.

Creating a Base Sketch

Well-constructed parts start with well-constructed sketches. Typically, the 3D model starts with a 2D sketch, which is assigned dimensions and *2D sketch constraints* to control the general size and shape. These dimensions and constraining geometries are the parameters, or input points, that you would then change to update or edit the sketch. For instance, Figure 1.1 shows a base sketch of a part being designed.

FIGURE 1.1

Creating a parametric model sketch



You can see four dimensions placed on the two rectangles defining the length and width of each along with a fifth dimension controlling the angle at which the two rectangles relate. These dimensions are parameters, and if you were to change one of them at any point during the design or revision of the part, the sketch would update and adjust to the change.

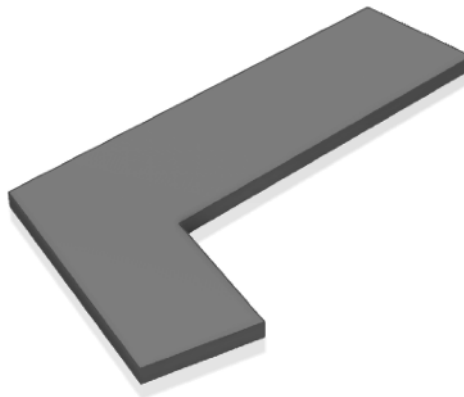
An important part of working with sketches is the concept of a fully constrained sketch. *Fully constrained* simply means that all the needed dimensions and sketch constraints have been applied to achieve a sketch that cannot be manipulated accidentally or as an unintentional consequence of an edit. For instance, if you were to sketch four lines to define a rectangle, you would expect two dimensions to be applied, defining the length and width. But you would also need to use 2D sketch constraints to constrain the lines so that they would stay perpendicular and equal to one another if one of the dimensions were to change. Without the sketch constraints, a dimensional edit to make the rectangle longer might result in a trapezoid or a parallelogram rather than the longer rectangle you anticipated. By fully constraining a sketch, you can anticipate the way in which it will update. Inventor helps you with this concept by automatically applying many sketch constraints and by reporting when a sketch is fully constrained. This will be covered in more detail in Chapter 3, “Exploring Sketch Techniques.”

Creating a Base Feature

Not only do you add 2D sketch parameters, but you also add parameters to control the 3D properties of parts. This is done by using the sketch to create a feature such as an extrusion to give a depth value to the sketch. The depth dimension is a parameter as well, and it can be updated at any time to adjust the part model as required. Figure 1.2 shows the sketch from Figure 1.1 after it has been given a depth using the Extrude tool.

FIGURE 1.2

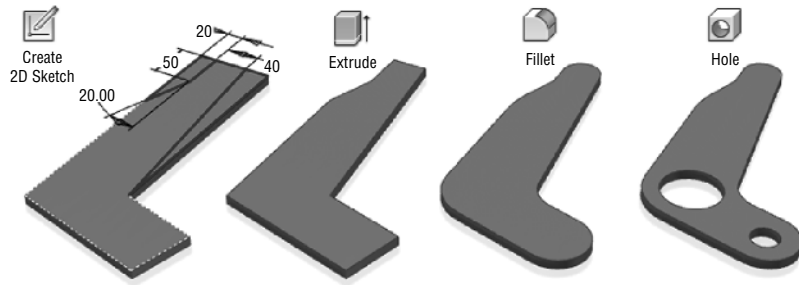
A basic part model created from the sketch



Adding More Features

Once the part is three-dimensional, more sketches can be added to any of the faces of the 3D shape, and those new sketches can be used to create some feature that further defines the form and function of the design. The model is then enhanced with more features, such as holes, fillets, and chamfers, until it is complete. Each added feature is controlled by still more parameters defined by you, the designer. If a change is required, you simply update the parameter and the model updates accordingly. This type of parametric design allows you to build robust and intelligent models quickly and update them even faster. Figure 1.3 illustrates the typical workflow of adding secondary features to a base feature to fully realize the part design, in this case a simple pivot link.

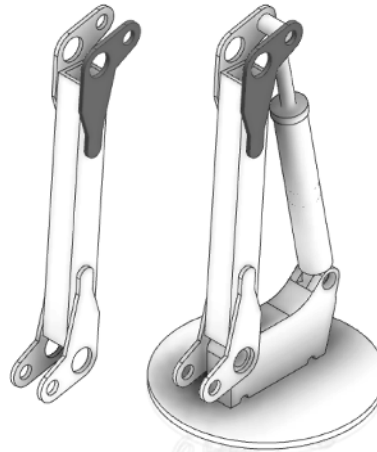
FIGURE 1.3
Adding features to complete the part model



Using the Part in an Assembly

Just as well-constructed parts start with well-constructed sketches, well-constructed assemblies start with well-constructed parts. Once the part model is built up from the features you create, you can use it in an assembly of other parts created in the same manner. You can copy the part to create multiple instances of the same part, and you can copy the part file to create variations of the original part. To assemble parts, you create geometric relationships called *assembly constraints* to define how the parts go together. The constraints are parameters that can be defined and revised by you at any time in the design process as well. Part models can be arranged into small assemblies and placed into larger assemblies to create a fully realized subassembly structure that matches the way your design will be built on the shop floor. Figure 1.4 shows the part model from the previous illustrations placed multiple times in a subassembly and then that subassembly placed in a top-level assembly.

FIGURE 1.4
A subassembly and an assembly model using the part model



Making Changes

Once parts are created, they are then used in assemblies, which also employ parameters to define the offsets and mating relationships between assembled parts. Designing with the use of parameters allows you to make edits quickly and lends itself to creating product configurations, where parameter values are changed to create variations of a basic design.

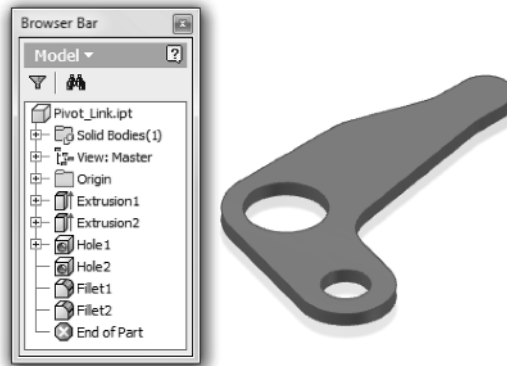
Of course, as with building anything, there are general rules and best practices to be learned and followed to prevent your work from “falling apart.” For instance, what if the pivot link used in the previous examples were to incur a design change that made one leg of the link longer? How would the holes be affected? Should they stay in the same place? Or should they stay at some defined distance from one end or the other?

Anticipating changes to the model is a large part of being successful with Inventor. Imagine, for instance, that a simple design change required that the pivot link become 50 millimeters longer on one leg. This should be a simple revision that requires you only to locate the dimension controlling that leg length and change the parameter value. Unfortunately, if you did not follow the best-practices guidelines when creating the part originally, the change in the length might displace the secondary features such as holes and material cuts and require you to stop and fix each of those as well. This is one of the most frustrating parts of learning Inventor for any new user who has not taken the time to learn or follow the known best practices of parametric modeling. Fortunately for you, within the pages of this book you will learn how to create models that are easy to update and do not “fall apart” during design changes.

Understanding History-Based Modeling and Dependencies

Inventor is often referred to as a history-based modeler, meaning that as you create sketches and turn them into features and then add more features and still more features, each addition is based on a previous feature, and so the model is said to have history. This history is recorded and tracked in the Model browser. The Model browser is a panel that displays onscreen and shows every feature you create during the design of your part. Figure 1.5 shows the Model browser for the pivot link file.

FIGURE 1.5
The Model browser
showing the feature
tree (history) of a
part named Pivot:
_Link.ipt



You can see that each feature is listed in the browser in the order in which it was created, forming a history tree. To create a part that handles changes predictably, you must create a solid foundation on which to build the rest of the model. In most cases, when you are designing a part model, you will start with a sketch, much like the one shown earlier in Figure 1.1. This base sketch will be your foundation, and therefore you must create it to be as stable as possible.

Each part, no matter what it is or what it looks like, has a set of origin geometry in the form of the origin planes, origin axes, and a single origin point. You can find these origin features by expanding the *Origin* folder in the Model browser. Figure 1.5 shows the *Origin* folder not expanded. If you expand the *Origin* folder in any part or assembly file, you will see the following items:

- ◆ YZ plane, the plane that runs infinitely in the Y and Z directions
- ◆ XZ plane, the plane that runs infinitely in the X and Z directions
- ◆ XY plane, the plane that runs infinitely in the X and Y directions
- ◆ X-axis, the axis running infinitely in the X direction
- ◆ Y-axis, the axis running infinitely in the Y direction
- ◆ Z-axis, the axis running infinitely in the Z direction
- ◆ Center point, the point found at zero in the X, Y, and Z directions

When creating the base sketch of a part file, you typically start on one of the origin planes. Because the origin plane cannot be edited, deleted, redefined, or upset in any manner, this base sketch is inherently stable, and as a result, the base feature you create from it is stable as well. If the second sketch of your part is created on a 3D face of the base feature, this sketch is dependent on the base sketch and is considered slightly less stable than the base sketch. This is because the base sketch could be edited, deleted, or redefined in a way that would upset the secondary sketch.

Understanding how dependencies are created when a sketch and features are based on one another will help you avoid creating a “house of cards” that will fall apart if the base is upset. Although you could base all your sketches and features on origin geometry to minimize dependencies, it is generally not practical to do so. It should be your goal, however, to keep the number of chained dependencies to a minimum. Assemblies work in much the same way, using the faces and edges of parts to constrain them together and as a result building dependencies between them. Just like part files, assembly files have origin planes, axes, and a center point that can be used to minimize chained dependencies, thereby creating a more stable model.

Taking a Closer Look at Sketch Dimensions

A large part of creating a stable sketch comes from understanding the way sketch dimensions work in Inventor. To do so, you might compare Inventor dimensions with standard dimensions in Autodesk® AutoCAD® software. When you create a design in AutoCAD, that design process is not much different from creating the same design on a paper drawing. But in AutoCAD, you can draw precise lines, arcs, circles, and other objects and place them precisely and with accurate dimensions reflecting your design in a way that you cannot do by hand. When a design requires modification, you erase, move, copy, stretch, and otherwise manipulate the existing geometry more quickly than you can by hand. But other than those gains in speed and accuracy,

the workflow is much the same as working with pencil and paper. In short, AutoCAD automates drafting tasks but does less to speed up and enhance the design process. By comparison, Inventor's sketch dimensions allow you to add design parameters and a bit of intelligence to your sketches.

DRIVEN DIMENSIONS

Standard dimensions in AutoCAD are called driven or reference dimensions. A *driven dimension* is controlled by the geometry, and it reflects the actual value of the geometry being referenced by the dimension. If you stretch a line, for example, the dimension attached to the line will update to the new value. If you think about it, the only reason for a dimension on a traditional AutoCAD drawing is to convey the value of a feature or part to the person who is going to build it. If you import that 2D file into computer-aided manufacturing (CAM) software, no dimensions are needed because the line work contains all the information about the part.

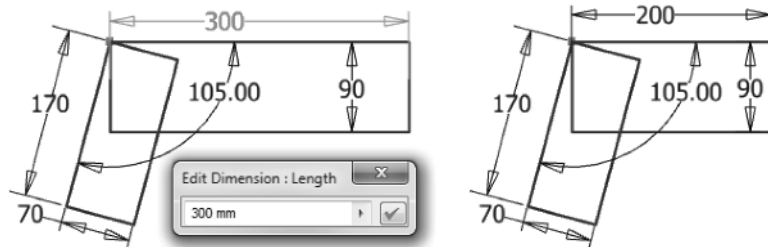
PARAMETRIC AUTOCAD

Starting with AutoCAD 2010, you can create 2D parametric dimensions and constraints much as you can in Inventor.

DRIVING DIMENSIONS

The workflow in Inventor sketching is substantially different from that in traditional AutoCAD, even beyond dimensions. In Inventor, you create sketches in 2D and then add geometric constraints such as Horizontal, Vertical, Parallel, and so on, to further define the sketch entities. Adding the geometric constraints allows line work to adjust in a predictable and desired manner and helps control the overall shape of the sketch. Once geometric constraints are in place, you add parametric *driving dimensions* to the sketch geometry. By changing the value of these driving dimensions, you change or drive the size of the sketch object. Because of this, the Inventor dimension is far more powerful than the standard AutoCAD dimension because it not only conveys the value of a feature or part but also serves as a design parameter, allowing you to change the dimension to update the design. This is done simply by double-clicking the dimension and typing in a new value. Figure 1.6 shows a dimension being edited in a sketch on the left and the result on the right.

FIGURE 1.6
Editing Inventor
sketch dimensions



Following Part Modeling Best Practices

A solid sketch is the foundation on which stable parts are built. Many new users do not understand the importance of having fully constrained sketches, and they find it highly frustrating to have a model fail when a simple change is made, all because a sketch was not properly constructed. You can avoid this frustration by following some basic best practices.

KEEP SKETCHES SIMPLE

The most effective way to create a healthy sketch is to keep it simple. The purpose of keeping your base sketch simple is to get it fully defined, leaving no part of it up for interpretation. Underdefined sketch entities (lines without defined lengths, circles without defined diameters, and so on) will most likely not update properly and will cause your sketches to distort and break when you try to update them. And because you often base the rest of your model on the initial sketch, your entire feature tree might incur errors, requiring you to stop and spend time rebuilding it again. Examine the sketch in Figure 1.1 and compare it to the finished shape shown earlier in Figure 1.5. As you can see, the simple sketch containing two rectangles dimensioned at an angle defines the basic shape and is much easier to sketch and fully constrain than the finished shape would be.

If the idea of simple sketches seems at first not to fit the type of design you do, understand that most designs will benefit from the simple-sketch philosophy. More important, if you start out employing simple sketches, you will more quickly master the sketch tools and then be ready to create more complex sketches when a design absolutely requires it.

CREATE SIMPLE FEATURES FROM SIMPLE SKETCHES

Another aspect of creating simple sketches is that it allows you to create simple features. Parametric, feature-based modeling relies on the creation of numerous simpler features within the model to achieve a complex design in the end. By creating a number of features within the model, you are able to independently change or modify a feature without rebuilding the entire model. An example of editing a feature would be changing a hole size. If you create a simple rectangular base feature first and then create a hole feature as a secondary feature to that base feature, you can make changes to both independently. By contrast, if you were to include a circle in your base sketch and use that to create the base feature with a circular profile pocket, your hole would no longer be as easily updated.

PATTERN AND MIRROR AT THE FEATURE LEVEL

Although there are mirror and pattern (array) tools in the sketch environment, it is generally best to create a single instance of the item in the sketch, then create a feature from it, and finally create a mirror or pattern feature from that feature. The logic behind this is based on the idea of keeping sketches simple in the anticipation of future edits. Should the mirror or pattern feature need to be updated, it is much easier to update it as a separate feature.

CREATE SKETCH-BASED FEATURES AND THEN PLACED FEATURES

Part features can be separated into two categories: sketch-based and placed. *Sketch-based features*, as you might guess, are created from sketches. *Placed features* are features such as fillets and chamfers that are placed by using model edges or faces and have no underlying sketch. Issues

arise when placed features are created too early in the development of the part because you may then be required to dimension to the placed feature, which creates a weak dependency. For instance, you might place rounded fillet features along the edges of a part. Then you could use the tangent fillet edges to define the placement of a hole. But then if you realize that machining capabilities require a beveled chamfer edge rather than a rounded filleted one and delete the fillet feature, the hole feature is sure to fail because the tangent fillet edges used to define the hole placement no longer exist. Keep this in mind as you create placed features such as fillets and chamfers and reserve placed features for the end stages of the part.

UNDERSTAND DEPENDENT AND INDEPENDENT FEATURES

Parametric model features are typically either dependent or independent of one another. A *dependent* feature depends on the existence or position of a previously created feature. If that previously created feature is deleted, the dependent feature will either be deleted as well or become an independent feature. As mentioned earlier, each part file contains default origin geometry defining the x-axis, y-axis, and z-axis of the part. These origin features are used to create the first sketch in every part by default. An *independent* feature is normally based on an origin feature or is referenced off the base feature.

For instance, to create the base feature for the pivot link, you would create a sketch on a default origin plane, such as the XY plane. Because the XY origin plane is included in every part file and cannot be changed, your base feature is stable and independent of any other features that may follow. To create a hole in the base feature, you would typically select the face of the base feature to sketch on. Doing so would make the hole feature dependent on the base feature. The hole feature is then inherently less stable than the base feature because it relies on the base feature to define its place in 3D space.

Although the specifics of how sketches, features, and parts are created will be covered in the chapters to come, remember these principles concerning part file best practices, and you will find Inventor (and any other parametric modeler) much more accommodating.

Following Assembly Modeling Best Practices

Once you've created part files, you will put them together to build an assembly. And when you do, you want to build it to be as stable as possible so that if you move, replace, or remove a part, the rest of the assembly will not fall apart. There are two entities to an assembly file: links to the component files (parts or subassemblies) it is made of and the geometric information about how those components fit together. Basic assemblies are not much more than that, and understanding those two concepts will go a long way toward building stable assemblies.

UNDERSTAND FILE LINKING AND RELATIONSHIPS

You can think of an assembly file as an empty container file to start. Once you place the first part in the assembly, the assembly file contains a link to the file for that part. When you place a second part and fit it to the first, the assembly then contains links to the two files and the information about how those parts go together in this particular assembly. If you decide to rename the first part file and do so using Windows Explorer, the assembly file will still look for the file by the old name. When this happens, you will be prompted with a file resolution dialog box asking you to locate the file. You can then browse and manually point the assembly to that file, and the assembly will record the new name in its internal link. If you decide to move the second part file

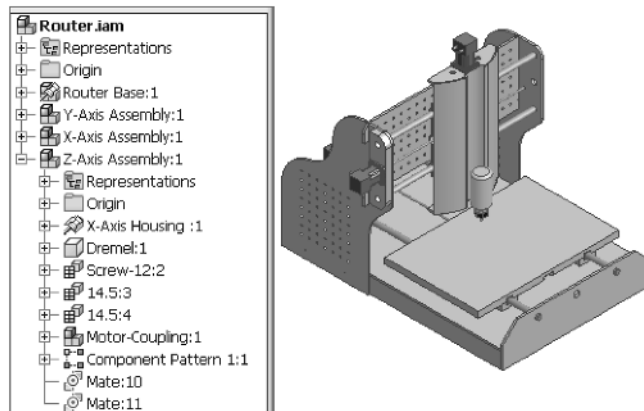
to a folder other than its original, the assembly file might again prompt you to find it manually, depending on the folder structure. It should be your goal to never need to resolve file links manually, and understanding this part of how assemblies work is the first step in reaching that goal. In the coming chapters you will learn how to set up Inventor properly so it can find your files.

ALWAYS MAINTAIN AT LEAST ONE GROUNDED COMPONENT

To understand how grounded parts help you build stable assemblies, you should first understand a little about the assembly Model browser. Figure 1.7 shows the Model browser for an assembly model of a small hobby-type CNC router.

FIGURE 1.7

The Model browser showing the model tree of an assembly named Router.iam



The Model browser shows an assembly named Router Base at the top and under it three other subassemblies named Y-Axis Assembly, X-Axis Assembly, and Z-Axis Assembly. The Z-Axis Assembly is expanded in the browser so you can see the parts it contains as well. You should note that the Router Base subassembly is shown in the browser with a pushpin button. This denotes that this subassembly is grounded, or pinned in place, and its coordinates cannot accidentally change. Keeping one grounded component in each assembly will allow you to fit other parts to it without it moving or rotating off the x-, y-, and z-coordinates.

Recall the old carnival game where you throw a ball at a pyramid stack of metal bottles. To win the game, you had to knock down all the bottles. However, if the bottle in the center on the bottom were nailed down, it would be impossible to win the game, and as a matter of physics, it would be difficult to knock down the bottles next to it. Having a grounded component in your assemblies, one that is “nailed down,” will likewise keep your assemblies from falling over as you build onto them. By default, the first component you place into an empty assembly file will automatically be grounded. You can unground it and ground another if need be, but you should always maintain at least one grounded component. You can also have more than one grounded component.

MAKE YOUR MODELS MIMIC THE MANUFACTURING PROCESS

The simplest advice that new users can receive on the subject of assemblies is to structure them as you would in real life. For example, if in the design you plan to assemble several parts into a

transmission and then drop that transmission into a housing, you should make the transmission a subassembly and insert it into the upper-level housing assembly. Alternatively, a new user might place all the parts into one big assembly, only to later realize that subassemblies are needed for the purpose of getting the bill of materials (BOM) organized. This can be accomplished by using the Demote assembly tool to create the subassemblies and then demote the parts from the top-level assembly to these new subassemblies. By making your models mimic the manufacturing process, you can also find possible flaws in your design, such as fasteners that cannot be accessed or areas where parts may interfere with each other during assembly.

In some instances, a model will be developed in the research and development (R&D) department and then handed to the manufacturing engineering (ME) department to be built. Although the people in R&D may enjoy the freedom of “dreaming up” anything they can think of, an effective R&D designer will always think about what can actually be built within the capabilities of the shop floor. Keep this in mind during the initial development cycle, and it will prevent those downstream from having to re-create much of your work. However, if restructuring the components into more or fewer subassemblies is required after the initial design, Inventor has demote and promote tools to assist with that. These tools will be covered in the chapters to come.

CONSTRAIN TO ORIGIN GEOMETRY

As mentioned earlier in this chapter, each part file has default origin geometry built in. You should build parts around the origin geometry whenever possible. For instance, a transmission has gears, bearings, seals, and so on, that are all concentric with the shaft. If you model all the parts so their x-axes will be aligned in the assembly, then you can use the x-axis of each part to constrain to in the assembly and it will be much more stable. However, if you constrain the parts by selecting model features, you run the risk of constraints failing once a revision to a part changes or removes the originally referenced geometry. To build a completely “bullet-proof” assembly, you could constrain the origin geometry of each part to the origin geometry of the assembly. In this way, no matter how the geometry of the parts changes, it will not cause issues with assembly constraints.

You will learn more about how to create assemblies, set up search paths to avoid manual file resolutions, and work with grounded components in the coming chapters, but you should remember these concepts and work to abide by them.

Understanding the “Feel” of Inventor

To the new user, the ever-changing Inventor interface may seem a bit disorienting. Taking a few minutes to understand why menus and tools change from one context to another will go a long way in getting comfortable with the “feel” of Inventor and anticipating the way the user interface works. If you’ve used other applications with the Microsoft ribbon-style interface, you’re probably already familiar with much of this context-specific behavior.

Understanding the Intuitive Interface

The overall user interface of Inventor could be called *context intuitive*, meaning that menus change depending on the task and the environment. Inventor is organized by tools grouped onto tabs, offering only the tools needed for the appropriate task at hand. If you are sketching a

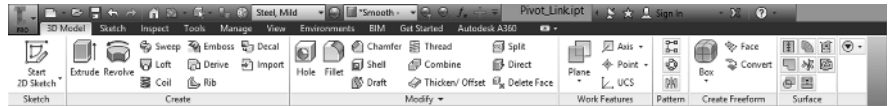
base feature, the tools you see are sketch tools. In Figure 1.8, the Sketch tab is active, and the displayed tools are the ones used to create and dimension sketches.

FIGURE 1.8
The Sketch
tab and
sketch tools



Upon the completion of a sketch, click the Finish Sketch button on the far right, and you will exit the sketch. The 3D Model tab then becomes active, and the Sketch tab is hidden. This allows you to see the tools that are appropriate for the immediate task, and only those tools, without having to hunt around for them among tools that you are not able to use at the current moment. If you create a new sketch or edit an existing one, the Sketch tab is immediately brought back. Figure 1.9 shows the active 3D Model tab.

FIGURE 1.9
The 3D Model
tab and model
tools



When you work with assemblies, the active tab changes to the Assemble tab (as shown in Figure 1.10), allowing you to place components, create new components, pattern them, copy them, and so on. When in the assembly environment, there are also a number of other tabs shown that you can manually switch to (by clicking them) at any time to use the tools they contain.

FIGURE 1.10
The Assemble
tab and assembly
tools



When you create a 2D drawing of parts or assemblies, you are automatically presented with tools needed to create views and annotation. By default, the Place Views tab is displayed because you need to create a view of a model before annotating it. However, you can manually switch to the Annotate tab by clicking it. Figure 1.11 shows the active Place Views tab and the inactive Annotate tab next to it.

FIGURE 1.11
The drawing tabs
and drawing
tools



As you can see, the collection of tabs (called the ribbon menu) changes intuitively with every task or environment you switch to. With this task-based user interface, there is no need to display every possible tool at once. In the next section, you will explore more of the user interface.

Using General Tools vs. Specific Commands

In this section you'll see how Inventor tools are set up, using AutoCAD tools as a comparison. If you've never used AutoCAD, you can still gain some insight from this section, although you may have to use your imagination concerning the references to AutoCAD. A key difference between AutoCAD and Inventor is that in AutoCAD many commands are very specific. For example, there are different dimension commands for lines, angles, and circles. In contrast, Inventor has one General Dimension tool that creates the appropriate dimension based on what you select.

For instance, in AutoCAD you might select the Horizontal dimension tool to place a dimension on a horizontal line, select the Diameter dimension tool to place a dimension on a hole, select the Radius dimension tool to place a dimension on a fillet, and so on. But in Inventor you select the General Dimension tool and select a horizontal line and you get a horizontal dimension; then, without exiting the General Dimension tool, you select a circle and you automatically get a diameter dimension. And of course to dimension a fillet, you continue with the General Dimension tool and you will automatically get a radius dimension.

DRAWING IN AUTOCAD BECOMES SKETCHING IN INVENTOR

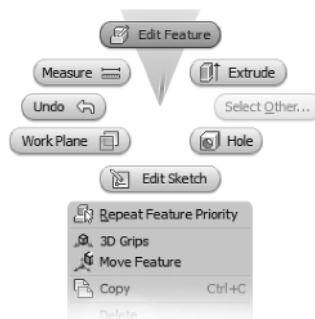
The fundamental difference between traditional AutoCAD and Inventor is that in AutoCAD you draw and in Inventor you sketch. This difference sounds subtle, but it is important. In AutoCAD, you likely construct lines precisely to specific dimensions to form the geometry required. In Inventor, you create lines and geometry that reflect the general form and function of the feature and then use constraints and dimensions to coax it into the desired shape. Expecting Inventor to work just like AutoCAD is probably the single biggest stumbling block that experienced AutoCAD users face when starting to use Inventor.

When in Doubt, Right-Click

Inventor is very right-click-driven, meaning that many of the options are context specific and can be accessed by right-clicking the object in question. For instance, if you want to edit a sketch, you right-click the sketch in the browser and choose Edit Sketch. The same is true of a feature. If you want to change a hole feature from a countersink to a counterbore, you right-click it in the browser and choose Edit Feature. You can also right-click many objects in the graphics window, with no need to locate them in the browser. Figure 1.12 shows a typical right-click context menu with the default marking menu option enabled.

FIGURE 1.12

A typical right-click marking menu



Also worth mentioning are the options in the context menus. For instance, if you are editing a part in an assembly and want to finish the edit and return to the assembly level, you could use the Return button on the Sketch tab menu, or you could just right-click (taking care not to click any sketch object) and choose Finish Edit from the context menu. Both options do the same thing.

TRADITIONAL RIGHT-CLICK MENUS VS. THE MARKING MENU

When enabled, marking menus replace the traditional right-click context menu. Since marking menus are customizable, this book references the traditional right-click menu, so specific references to items in the right-click menus may need to be interpreted if you choose to use the marking menus or have customized your marking menus. To enable or disable the marking menus, select Customize on the Tools tab of the ribbon and then select the Marking Menu tab. Then check or uncheck the Use Classic Context Menu check box.

Selections from the marking menu can be made in either menu mode or mark mode.

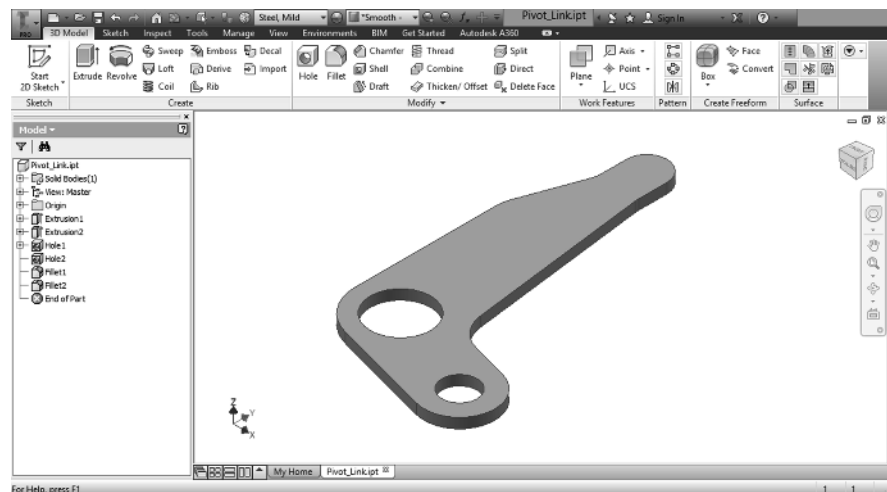
Menu Mode When you right-click in the graphics window, menu items surround the cursor. Simply click a menu item to select it.

Mark Mode When you press and hold the right mouse button and immediately move the cursor in the direction of a known menu item, a “mark” trail appears. Release the mouse button to select the menu item corresponding to the direction of cursor movement in the marking menu.

Using the Graphical Interface

The Inventor graphical interface might be different from what you are accustomed to in other general software applications and even different from other design software. In Figure 1.13, you see the entire Inventor window, which shows a part file open for editing.

FIGURE 1.13
The complete
Inventor
screen in part
modeling
mode



Inventor Title Bar

Starting at the upper left of the Inventor window, you'll see the Inventor button (look for the large *I* character), which has a drop-down panel similar to the File menu in previous versions. Next to the Inventor button, the title bar includes two toolbars:

- ◆ The Quick Access bar has frequently used tools.
- ◆ The Help toolbar provides access to help files and Autodesk websites.

You can customize the Quick Access bar for each file type by selecting and deselecting buttons from a list. You can access the list of available tools by clicking the drop-down arrow shown on the far right of Figure 1.14.

FIGURE 1.14
The Inventor button and Quick Access bar



Table 1.1 defines the default Quick Access bar buttons available in part modeling mode.

TABLE 1.1: Quick Access bar buttons

BUTTON	DEFINITION
	The New button launches the New File dialog box. The drop-down list allows you to create a new part, assembly, drawing, or presentation file using the standard templates.
	The Open button launches the Open dialog box. It displays a location defined in your active project.
	The Save button saves the file.
	The Undo button undoes the last action. The Undo list tracks changes for the current Inventor session, not just the current document. If you have two part files open, this button will undo changes that are made in both files. Undo will also close files if your undo sequence takes you back past the point of a file being opened or created.
	The Redo button restores a change that was removed with Undo. It will reopen a file that was closed with Undo.
	The Update button updates the files. For example, if you edit a part in an assembly, other parts might need to be updated because of the changes. This button is grayed out unless the file needs to be refreshed.

TABLE 1.1: Quick Access bar buttons (CONTINUED)

BUTTON	DEFINITION
	The Select button allows you to choose a filter for object selection.
	The Refine button can be used to speed up large Assembly performance by providing a coarser image (Disable Automatic Refinement must be turned on in Applications Options > Display).
	The Material button and drop-down box allow you to change the component material. Clicking the button displays the material browser. Selecting a material from the drop-down changes the component's material property.
	The Appearance button and drop-down box allow you to change the component color. Clicking the button displays the appearance browser. Selecting an appearance from the drop-down changes the component's color appearance.
	The Adjust button allows you to modify the color and texture appearance.
	The Clear button allows you to remove overrides to the component's color and texture appearances.
	The Parameter button is used to access the parameters table, where you can rename, change, and create equations in dimension and design parameters.
	The Design Doctor button launches a dialog box that helps you diagnose and repair issues with a file. The button is grayed out unless there is an issue.

Graphics Window Tools

Inventor has two sets of tools for manipulating the graphics window:

- ◆ The ViewCube allows you to change the view orientation.
- ◆ The Navigation bar has tools such as Zoom and Pan.

EXPLORING THE VIEWCUBE

The ViewCube, shown in Figure 1.15, is a 3D tool that allows you to rotate the view.

FIGURE 1.15
The ViewCube



Here are some viewing options:

- ◆ If you click a face, edge, or corner of the ViewCube, the view rotates so the selection is perpendicular to the screen.
- ◆ If you click and drag an edge, the view rotates around the parallel axis.
- ◆ If you click and drag a corner, you can rotate the model freely.
- ◆ If you click a face to have an orthogonal view, additional controls will display when your mouse pointer is near the cube.
- ◆ The four arrowheads pointed at the cube rotate the view to the next face.
- ◆ The arc arrows rotate the view by 90 degrees in the current plane.

If you click the Home button (it looks like a house), the view rotates to the default isometric view. Clicking the drop-down arrow or right-clicking the Home button reveals several options to change the default isometric view behavior. For instance, you can modify the home view to any view you like, and you can reset the front view in relation to your model so the named views of the cube match what you consider the front, top, right, and so on.

USING A WHEEL MOUSE AND 3D-INPUT DEVICE

Using a wheel mouse with Inventor is recommended. Scrolling the wheel will zoom in or out, while pressing the wheel will perform the Pan function. In Inventor, the wheel zoom is reversed from AutoCAD. You can change this setting by clicking Application Options on the Tools tab, selecting the Display tab, and selecting Reverse Direction in the 3D Navigation group.

Another useful tool for navigating in Inventor is a 3D-input device. A popular brand is the Space series made by 3Dconnexion. These devices are small “joysticks” or “pucks” that sit on your desk. The user grasps the puck and, by making very slight movements with the device, moves the model on the screen. Pulling, pushing, and twisting the puck allows you to zoom, pan, and orbit the model onscreen. Although you may find using one of these devices awkward at first, most users say they could never work as efficiently without one after just a few days of use.

A LOOK AT THE NAVIGATION BAR

Continuing with the interface tour, you’ll see the Navigation bar located on the right side of the graphics window. At the top of the bar is the steering wheel. Below the steering wheel are the other standard navigation tools: Pan, Zoom, Orbit, and Look At. Figure 1.16 shows the Navigation bar.

FIGURE 1.16
The Navigation bar



You can use the Navigation bar's steering wheel to zoom, pan, walk, and look around the graphics area. Also available is the ability to rewind through previous steering wheel actions. The steering wheel has more functionality than can be explored in this book. You should review the help topics for more information (click the steering wheel and then press F1).

The Ribbon Menu

The *ribbon menu* is composed of tabs and panels and is similar to the menu used in Microsoft Office products (starting with Office 2007). Each tab contains panels for a particular task, such as creating sketches, and each panel contains related buttons for the tools. As previously mentioned, the ribbon will change to the proper tab based on the current task (for example, sketching brings up the Sketch tab, which allows you to click the Line tool button), but you can select a different tab as needed.

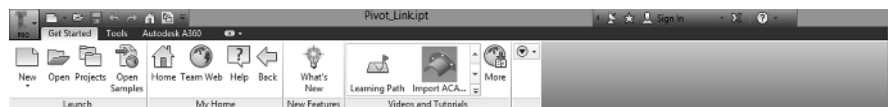
You can customize the ribbon by right-clicking it and choosing among the following:

- ◆ Turning off tool button text, reducing button size, or using a compact button layout
- ◆ Turning off panels that you don't use
- ◆ Adding frequently used commands to a tab
- ◆ Changing the ribbon appearance
- ◆ Undocking the ribbon so it becomes a floating tool palette
- ◆ Docking the ribbon on the left, right, or top of the Inventor window

THE GET STARTED TAB

On the Get Started tab of the ribbon, the tools on the Launch panel are used to access and create files. The rest of the buttons on the Get Started tab link to help topics. You can use the What's New button to read about the new features for the current release as well as the last few releases. The Videos and Tutorials tools contain built-in tutorials and a collection of learning resources. Figure 1.17 shows the Get Started tab and its tools.

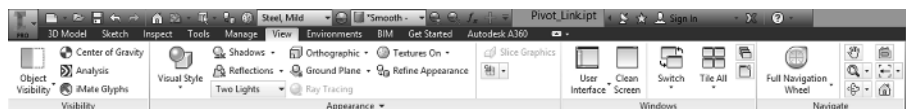
FIGURE 1.17
Tools on the Get
Started tab



THE VIEW TAB

The View tab, shown in Figure 1.18, has controls for object visibility and appearance, window control, and navigation. There are some variations in the buttons, depending on the environment, but most of the buttons are used in all the modeling environments.

FIGURE 1.18
The View tab



Visibility Panel

The Visibility panel has tools for controlling which objects are visible. When you click Object Visibility, a large list is displayed so you can control the visibility of the listed objects in your graphics window.

Appearance Panel

The Appearance panel has tools for controlling the way models are displayed. You can display the model in a number of visual styles, such as Realistic, Shaded, Shaded With Edges, Illustration, and many more.

VISUAL STYLES AND PERFORMANCE

Inventor includes a number of enhanced visual styles, shadows, and reflection options that might have a negative impact on your graphics performance if your workstation is a bit older. If you notice issues with your graphics display, you can go to the Tools tab, click the Application Options button, click the Hardware tab, and then select the Performance or Compatibility option.

Another important option on the Appearance panel is the View Camera Projection setting, which allows you to choose between orthographic and perspective views. Setting the perspective options to be current displays the model with a vanishing point, as it would be in the real world. With the Orthographic option, points of the model are projected along lines parallel to the screen.

Using a perspective view may be desirable when viewing the model in a 3D view, but it can be distracting when sketching on a flat face or viewing the model from a standard 2D orthographic view because you see what appear to be tapering faces and edges. However, you can get the best of both projections by setting the ViewCube to Perspective With Ortho Faces so that the model is displayed in orthographic mode when one of the standard orthographic faces is active and in perspective mode in any other view. To do this, simply right-click the ViewCube and you will see the option. Note that this setting is set for each document rather than for the application itself, so you will typically need to do this for each model.

Windows Panel

Most of the tools in the Windows panel are standard controls, such as for switching tiling windows. If you click User Interface, a list of items such as the ViewCube and the status bar are displayed. The Clean Screen button hides most of the UI elements. Only the title bar and a minimized ribbon bar are displayed. Although the Clean Screen setting certainly maximizes your screen real estate, it turns off one critical interface object, the browser pane. To use the Clean Screen function effectively, you must turn the browser pane back on. To do so, click the View tab, use the User Interface drop-down, and select the Browser option. You can click the View tab again and then click the Clean Screen button again to disable it and display the tools panels.

Navigate Panel

Also on the View tab is a Navigate panel. The tools in the Navigate panel are the same as those on the Navigation bar, as discussed earlier in the chapter. Many of these tools, such as Pan, Zoom, and Orbit, can be accessed by using the buttons on your mouse or by using the function

keys. For instance, spinning the wheel on a standard three-button mouse allows you to zoom in and out. Similarly, if you hold down the F4 button on the keyboard, you will see that the Orbit tool is active.

The Browser Pane/Model Browser

The *browser pane* (often called the Model browser) is a listing of everything that makes up an Inventor file. The part browser shows all the features, the assembly browser shows all the components, and the drawing browser shows the sheets with the views. Because Inventor files are similar to actual parts and assemblies, the browser plays an important role in navigating the files.

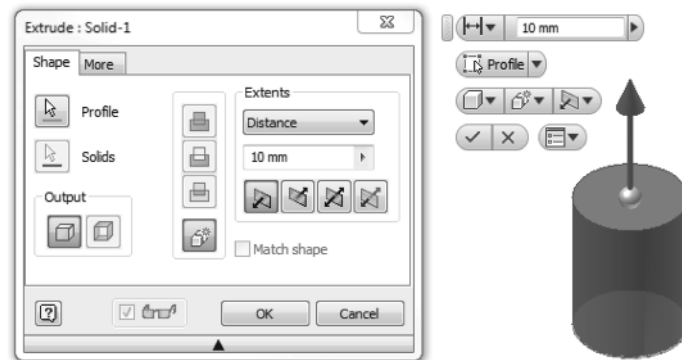
TURNING ON A MISSING MODEL BROWSER

Although it isn't common to need to turn the Model browser off, you can do so. More commonly, you may accidentally turn it off by clicking the X button on the right side of the browser title bar. To display it again, from the View tab click the User Interface button on the Windows panel. You'll most likely want to have all the items in this list selected.

Dialog Boxes and the In-Canvas Mini-Toolbars

As you use Inventor, you will notice that there are often two sets of input controls: the traditional dialog box controls and the in-canvas mini-toolbars. The inputs in the dialog box are the same as those found in the mini-toolbars, and therefore you can use either one to input information or change options. Changing an option in one updates it in the other. You can use the arrow at the bottom of the dialog box to expand or collapse it. Figure 1.19 shows both sets of controls as they appear for editing a simple extrusion.

FIGURE 1.19
Dialog box and mini-
toolbar controls

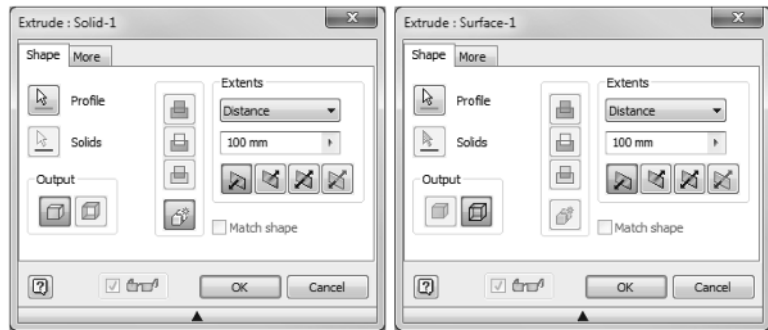


If you find the mini-toolbar controls become distracting by popping up in a position that is in the way of making selections onscreen, you can use the Mini Toolbar Options menu button (the button on the far right of the last row) to pin the mini-toolbar to a location onscreen of your choosing.

Task-Based Tools

You saw in the previous section that the tabs of the ribbon menus update based on the current environment. For instance, when in the sketch environment, the Sketch tab is active. The task-based nature of the available tools is common throughout Inventor. For example, many Inventor dialog boxes are also task-based. Instead of containing every control needed for every environment, most dialog boxes display only the controls necessary for the current task. Figure 1.20 shows two Extrude dialog boxes.

FIGURE 1.20
Solid and surface
Extrude dialog boxes



Because creating and editing a solid extrusion is different from creating and editing a surface extrusion, some options are simply grayed out and not available. You will notice this throughout Inventor because options are offered and suppressed depending on the task at hand.

Learning the File Types in Inventor

If you've used AutoCAD, you might be accustomed to having the DWG (.dwg) file format as your primary file format; in Microsoft Word you might use primarily just a DOC (.doc) file; and in Microsoft Excel, you might use the XLS (.xls) file type for most of the work you do. All three of these commonly used programs use a single primary file type throughout. Inventor, on the other hand, follows the structure common to most other 3D modelers in the engineering field today and uses different file types for different tasks.

The purpose of using multiple file types is so the data load is distributed into many different files instead of having all information in one file. For instance, you use an IPT (.ipt) file to create an Inventor part file, an IAM (.iam) file to assemble that part with other parts, and an IDW (.idw) file to make a detail drawing of the parts and the assembly.

Placing the data in multiple files permits quicker load times, promotes file integrity, and vastly improves performance across the board on large designs. For example, when you open an assembly made of 12 different part files, only the information concerning the file paths and the way the parts fit together in the assembly is loaded along with the information required to display the parts. It is only when you decide to edit a part that the information about all of that part's features is loaded. As you learned in the previous section, having different file types allows you to use environment-specific tools. So if you edit a part from within an assembly file, Inventor automatically presents the part-editing tools.

TURNING ON FILENAME EXTENSIONS

It's often helpful when working with Inventor files to be able to view the filename extensions. By default, Windows hides the extensions for known file types. To show filename extensions, follow these steps for Windows 7:

1. Open Folder Options by clicking the Start button, clicking Control Panel, and then clicking Folder Options. (If Folder Options is not available, change View By to Large Icons in the upper-right portion of the Control Panel.)
2. Select the View tab and uncheck the Hide Extensions For Known File Types option.

Another payoff of multiple file types is exemplified in the comparison between the way AutoCAD handles tasks related to model space/paper space and the way Inventor handles the same tasks. To put it simply, in Inventor the part and assembly files are the model (model space), and the drawing file is in effect paper space. Using multiple file types to handle the separate tasks required for modeling vs. detailing simplifies the interaction between both tasks, and as a result, the headaches of managing model space and paper space that exist in AutoCAD are eliminated in Inventor.

Table 1.2 describes the filename extensions for the file formats commonly used in Inventor.

TABLE 1.2: Common filename extensions in Inventor

EXTENSION	DESCRIPTION	USE
.ipj	Inventor project file	Used to manage file linking paths
.ipt	Inventor single-part file	Used to create individual parts
.iam	Inventor assembly file	Used to assemble parts
.ipn	Inventor presentation file	Used to create exploded views of assemblies
.idw	Inventor 2D detail drawing file	Used to detail part, assembly, and presentation files
.ide	Inventor iFeature file	Used to save features to the iFeature library for reuse in future designs
.dwg (Inventor)	Inventor 2D detail drawing file	Like IDW files, used to detail part, assembly, and presentation files
.dwg (AutoCAD)	AutoCAD nonassociative drawing file	Used to convert an Inventor drawing file to a standard AutoCAD file
.xls	Excel files that drive iParts, threads, and other data	Used to manage tabulated data linked or embedded in a part, assembly, or drawing file

Although this table may seem intimidating, once you become familiar with Inventor, having many different file types will be less of a concern. The benefit of using multiple file types to have fully associative, automatically updating designs is a cornerstone of most 3D parametric modelers. Performance and stability in the use of Inventor require good data management principles, including storing the saved files in an efficient and organized manner.

DWG FILE SIZE

Although the benefits of using an Inventor DWG instead of an IDW may be favorable, you should be aware that the extra abilities of the DWG file do come at the expense of file size. Inventor DWGs are typically two to three times larger than identical IDW files. If you create large assemblies, it is advisable to use the IDW template as opposed to the DWG to keep files manageable. The extent to which the DWG in Inventor is employed will largely be determined by the amount of collaboration required between Inventor and AutoCAD users.

What Is an Inventor Project?

Whether you use Inventor as a stand-alone user or as part of a design group, you should configure and use an Inventor project file to help Inventor resolve file links and keep your designs organized. You can think of project files in Inventor simply as configuration files that tell Inventor where to look for component files when working with assemblies and drawings. For instance, an Inventor assembly file is essentially an empty “bucket” into which parts (and subassemblies) are placed and assembled. Therefore, the assembly file contains only the file path references for the components it is composed of and the information about how those components are assembled. As a result, the location of referenced files is a key issue.

If, when an assembly is opened, referenced files cannot be found at the search path recorded in the assembly file, a manual file resolution process is activated. This happens most often when component files are renamed or moved outside the search path established in the project file.

A NOTE TO AUTODESK® INVENTOR LT™ USERS

If you are using the Autodesk Inventor LT software, you should be aware that it doesn’t use project files, and therefore this book’s instructions concerning Inventor project files do not apply.

Project Files and Search Paths

Project files are often referred to as IPJ files because .ipj is the extension for project filenames. You can create a project file anywhere it makes sense to do so, and Inventor will look at that location and lower in the directory structure for the files in your design. Take a moment and study the file structure shown in Figure 1.21.

FIGURE 1.21
A job-based folder structure

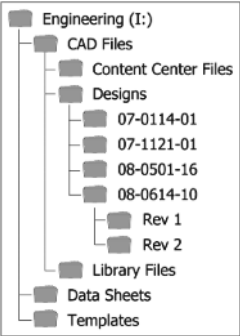


Figure 1.21 shows a typical job-based folder structure, where all files are located on the Engineering (I:) drive. Engineering contains three subfolders: CAD Files, Data Sheets, and Templates. In the CAD Files folder are three more folders: Content Center Files, Designs, and Library Files. The Designs folder contains a folder for each job (named using the job number) and subfolders containing revisions. So, where would you create an Inventor project file? There are two basic solutions: create multiple IPJ files for each new job or create a single all-encompassing IPJ file for the entire engineering drive. Which method you should use depends largely on the way your engineering department operates.

JOB-BASED IPJ SETUPS

You could create one project file for each of the four jobs. You would have a file named 07-0114-01.ipj in the 07-0114-01 folder, one named 07-1121-01.ipj in the 07-1121-01 folder, and so on. This strategy can work fine if you typically work on one project at a time and then “close” the project upon completion. In Inventor, you would simply switch to the specific project file that matches the job number for the job you are working on (for example, 07-0114-01), and because the IPJ file is stored in that folder, Inventor will search for design files only in its *workspace*. The workspace is defined as the folder containing the IPJ folder and everything below it. It may help to think of a workspace as a search cone, starting at the IPJ and spreading out from that point.

This job-based approach is fairly intuitive and is what people generally think of when they see the term *project file*: one IPJ file for each job/project. This is a common approach when a job has a long development cycle and designs are specific to that job.

But what happens if you wanted to use a part that was created for job 07-0114-01 in job 08-0614-10? You could place the part into the 08-0614-10 assembly, but the next time you opened that assembly, Inventor would not be able to find it because it exists outside the 08-0614-10 workspace. If you were to move a part file from the 07-0114-01 folder into the 08-0614-10 folder, Inventor would not find it while you were working on job 07-0114-01 because it would now be outside of its workspace. Likewise, if you moved the file up to the Designs folder, to the CAD Files folder, or to (almost) any location that is not next to or below the 07-0114-01.ipj file, Inventor would not find it as long as you are working with the 07-0114-01.ipj project file. If you copy the file to the 08-0614-10 folder, then you have two versions of it and it becomes difficult to track changes because you need to update both copies to keep everything up-to-date.

The solution would be to configure the IPJ file to include a *library*. When a folder is configured to be used as a library in an IPJ file, Inventor sees all the files in that folder (and its subfolders) as read-only. This protects commonly used files from being accidentally changed and upsetting all of the many designs in which they may be used.

To solve the issue of the commonly used part in this example, you could configure each of your IPJ files to use the folder named Library Files as a library of approved, read-only parts to be used across multiple jobs. Whenever you open an assembly, Inventor first looks in the library path for the parts and then looks at the workspace. So, to convert a part created as part of the 07-0114-01 job into a library part, you would follow these steps:

1. Copy the file to the library folder.
2. If the original file has a job-specific name, rename the copy according to a defined library nomenclature.
3. Open the assembly (or assemblies) that uses the original part.

4. Use the Replace Components tool to replace the original part with the library part in the assembly.
5. Save and close the assembly.
6. Delete the original part so that no duplicate is present.

SINGLE IPJ SETUPS

The multiple project file strategy described previously is often not the best approach for many design departments, because of various contributing factors. When this is the case, you should consider using a single project file. Using the same folder structure shown in Figure 1.21 as an example, you could use a single all-encompassing IPJ file and place it in the **Designs** folder. By doing so, you would be setting the workspace at that level. This configures the search paths in Inventor to look for files in the **Designs** folder and everything below it. Essentially, you have expanded the search cone by moving it up a level compared to the job-based setup. Now if you need to use a part that resides in the 07-0114-01 folder in the 08-0614-10 assembly, you can do so and Inventor will be able to find it, without requiring it to be in a library folder.

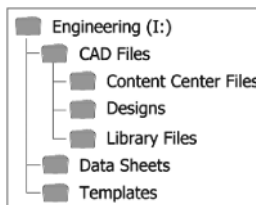
Of course, you can still use library folders when using the single IPJ file approach, and in fact it is generally recommended that you convert common parts to library parts when they are being used in many different designs. Because folders configured as libraries in the IPJ file are handled as read-only, this protects them from accidental modifications.

One major caveat to using just a single IPJ file is that in order to prevent the possibility of the wrong part being loaded in an assembly, it is important for every part located in the search path to have a unique name. If Inventor finds two files named `BasePlate01`, it will either use the first one it finds or stop and make you decide which one to use. In either case, you should consider a nomenclature that references the job number, date, or other unique identifier in the name.

ITEM-BASED SETUPS

If your company uses an item-based file management setup and tracks each part you create or purchase as an item, you are probably not concerned with job numbers as much as you are about part numbers. Most likely you will want to employ a single IPJ file setup as described previously and again place the IPJ file in the **Designs** folder. Additionally, your file structure may be a bit flatter and look like Figure 1.22, where the **Designs** folder has no subfolders.

FIGURE 1.22
An item-based folder structure



So if folders configured as libraries are configured as read-only in the IPJ file, how are controlled, purposeful revisions carried out on library files? The answer is to create an IPJ file configured to look at the library folder as a standard folder. For instance, you might create an IPJ file in the `Library Files` folder and assign it no library path. You would then switch to this IPJ file only when doing library maintenance. Because this IPJ has no library path called out, the files are not handled as read-only when Inventor is using this IPJ file. Often in a large engineering department only a couple of people have access to the library editor project file. When other team members see a need to change a library file, they would submit a change order, and the designated person (or people) would then make the change.

Content Center Files

In the previous figures you may have noticed a folder called `Content Center Files`. This is a special kind of library that stores component files generated by Inventor's Content Center tools. The path to this folder is specified in the IPJ file, much like a library file is.

It is important to understand what Content Center is and how it works. Content Center libraries are collections of table data containing the definitions used to create more than 800,000 standard parts and features. This database is managed by the *Desktop Content* settings or the *Autodesk Data Management Server (ADMS)*. Once you've installed the content libraries, you can use the content in your designs. To do this, choose a component from the database to place into your design, typically by using the *Place From Content Center* button in the assembly environment. It is at this point that the Content Center part file is created. Up to this point, the part existed only as a definition in the database table.

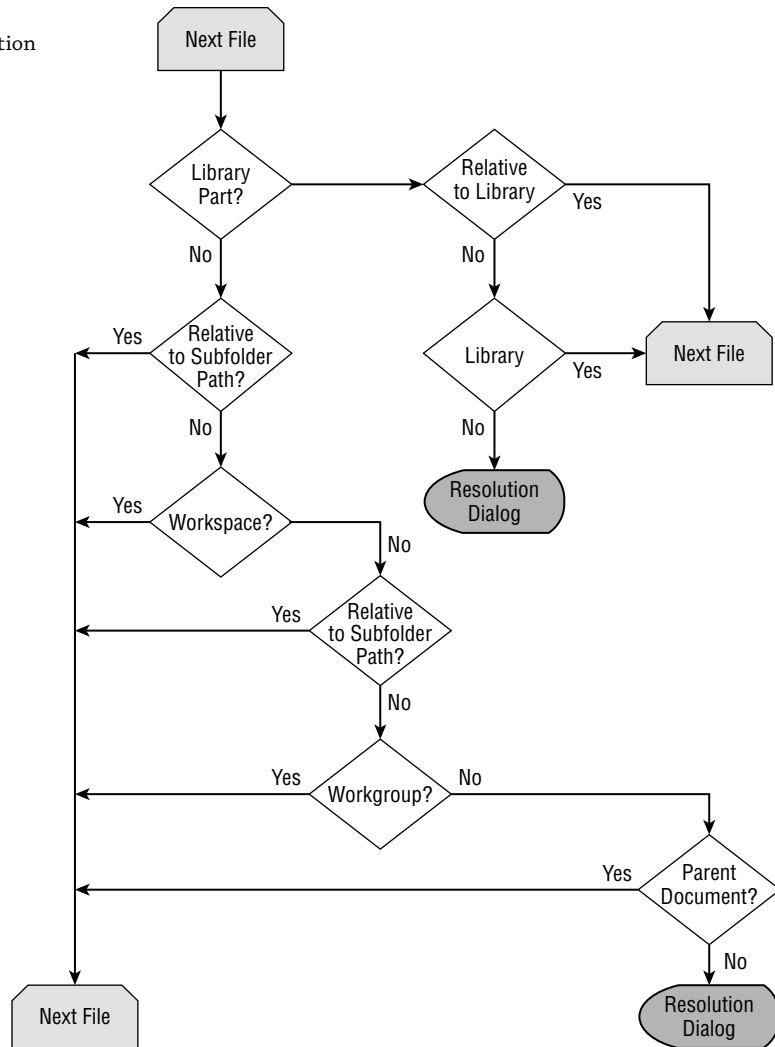
In your IPJ, you need to specify a Content Center file store location so that Inventor knows where to save the file and where to find it next time. The file store folder will include additional subfolders where Content Center files will be stored once they are used in your designs. These additional folders are created automatically as parts are created. The next time a part is specified from Content Center libraries, Inventor first searches Content Center file store directories and then creates the part from the database only if the part file does not already exist in the file store location. It is required that the Content Center file store location be outside the main project data path. From this discussion of libraries, you can see that high importance is placed on planning the correct part locations and workflow.

How Search Paths and Project Files Are Used

The IPJ files in Inventor are easy to create and use, provided you understand how Inventor uses them. An Inventor project file is a configuration file that lists the locations and functions of each search path. Inventor uses these definitions to resolve file links and locate the files needed for the parts and assemblies on which you want to work. Figure 1.24 shows how Inventor loads assemblies and parts inside an assembly file.

When opening an assembly file, Inventor finds files by searching for the first file to be located within the assembly file. Inventor first looks in the library folders for that file. Next, Inventor searches in the local workspace for the file. When a file is not found in any of the referenced folders, Inventor launches a manual file resolution dialog box offering you the opportunity to browse and point to the file manually.

FIGURE 1.24
Inventor file resolution
protocol



Exploring Project File Types

As mentioned previously, file management in Inventor is handled through the use of a project file (*.ipj). A *project file* is a configuration file set up and used to control how Inventor creates and resolves file links, where you edit files, how many old versions of the files to keep, and how Content Center files are stored and used. In the early days of Inventor, Autodesk offered two basic project types: single-user projects and multiuser projects. At this point, the Autodesk Vault project has replaced the earlier multiuser project types.

Unless you are planning to use Vault, the default project type is the single-user project. The term *single-user* could be considered a misnomer because this project type is widely used by one-person shops and multiseat design departments alike. Single-user does not mean that only one user can access the files in the project, as it might suggest; instead, it refers to the fact that there are no means of preventing files from being accessed for editing while another user is already editing the file. This can create a last-person-to-save-wins situation if you aren't careful.

SINGLE-USER PROJECTS AMONG MULTIPLE USERS

What happens when two users access the same file in a single-user project? Typically this is first noticed when one of the two tries to save the file. Inventor notifies the person trying to save that they are not working with the most current version and gives the other user's name (depending on the network setup) so the first user knows what is going on. Inventor instructs the user that they must save the file using a different name to prevent losing the changes made.

Typically, at this point a conversation takes place to determine how to proceed. If it is decided that the first person is the one who needs to save changes, then the file this person was working on is saved using another name, the original file is deleted (or renamed as a reserve), and the other file is renamed to replace the original. In this way, the changes that were made to the original file are preserved.

Although this may seem like a terrible hassle, there are many design departments that use single-user projects in a team setting effectively and only rarely run into this situation. More than likely you already have an idea of how often you and your colleagues handle the same files at the same time. But if you try to use single-user projects and find this situation happens fairly often, you should consider a true multiuser project.

Many design departments use single-user projects effectively in collaborative environments because of workflows that lend themselves to this type of project; others make it work by simply maintaining good communication among the design team. For collaborative environments that require some safeguard against situations in which users could potentially save over one another's work, using a *multiuser* project (Vault project) is recommended.

Autodesk Vault is a data management application that, as the name implies, locks down files for their protection. Once a file is in Vault, it must be checked out by a user to be edited. Vault typically resides on a file server where the entire design team can access it. When the file is checked out of the Vault server, it is placed on the user's local machine for editing. The next user who comes along and attempts to access that file can access only a read-only version. Once the first user has finished editing, the file is checked back into Vault and automatically versioned.

It is also important to note that Inventor installs with a *default* project setup. The default project is typically not used for production work because it is not fully configurable and will almost always lead to file resolution issues because it has no defined search path.

CREATING A GOOD DATA-MANAGEMENT PLAN

A good data management plan is the key to using Inventor projects successfully. Using Vault will not resolve a poor project file or data management strategy.

One part of a successful Inventor deployment is the hardware network on which the software will run. It is important that the engineering group has buy-in by the IT group. You will need to discuss several issues with this group, including hardware for servers and workstations, the network setup (100BaseT or Gigabit), mapped network drives, and user permissions. A good server can be the difference between success and failure in your rollout.

Although you do need to think about your file structure, don't obsess over it. Most likely you will end up changing the structure at least a few times before you settle on a final structure. Keep an open mind, and realize that if you have five people in a room discussing file structures, you'll end up with five different ideas. Again, involve IT in your discussions.

Finally, you should designate one person in engineering to be the engineering administrator. This person needs to have administrative privileges on the engineering server or network share. IT may resist, but you have to keep pushing. This is important because you will need the ability to easily create, delete, and move files and folders without having to submit a help-desk ticket. Nothing will slow down a design process faster than having to wait for IT to make a simple change. Explain this need to your IT administrator, and most likely they will understand.

Creating a Project File

Whether you choose to use a single-user or multiuser project file, it is important that you pick one. Without a project file to configure Inventor's search path, you are likely to create a file structure that will be problematic as you create more and more files in Inventor.



Creating Single-User Projects

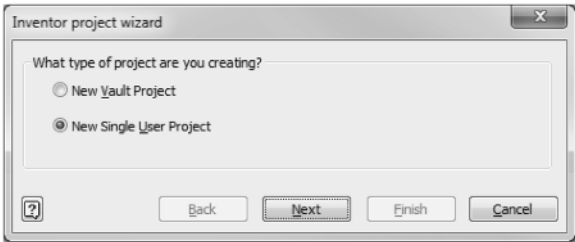
Probably the best way to learn about projects is to create a test single-user project. With single-user projects, you can open, edit, and save files without checking the files in or out. In the following sections, you will investigate the single-user file project mode. Once you gain an understanding of single-user projects, you will be ready to investigate the other project file types. To create a test project, you will use the Inventor Project Wizard.

THE INVENTOR PROJECT WIZARD

To get the most out of this exercise, open your version of Inventor, ensure that you have closed all the open files, and then access the Inventor Project Wizard by clicking the Projects button on the Get Started tab. Then follow these steps:

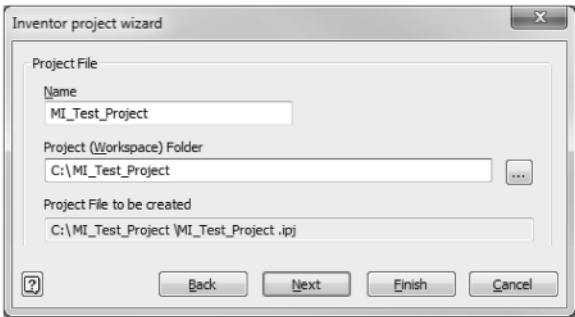
1. In the Projects dialog box, click the New button at the bottom.
2. Select New Single User Project, as shown in Figure 1.25, and then click Next.

FIGURE 1.25
Creating a single-user project



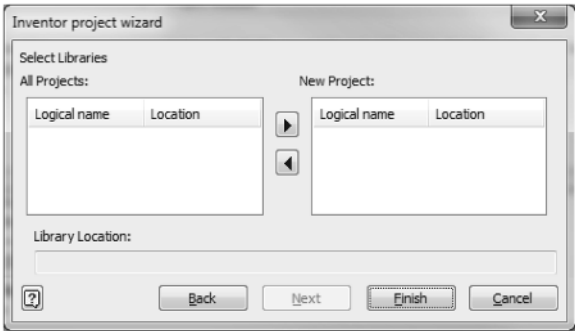
- 3. Enter **MI_Test_Project** in the Name field.
 - 4. Enter **C:\ MI_Test_Project** in the Project (Workspace) Folder field.
- Figure 1.26 shows a Project File page specifying the project.

FIGURE 1.26
The Project File page filled in



- 5. Click Next to advance to the next page of the wizard.
- 6. If you already created a folder for your library files and used those libraries in a previous project, their locations will appear on the Select Libraries page, shown in Figure 1.27. When creating a new project, you can choose to include some, all, or none of the defined library locations. Click the Finish button to include no libraries at this point.

FIGURE 1.27
Select Libraries page



- 7. Click OK in the message box informing you that the project path entered does not yet exist.

SWITCHING AND EDITING PROJECTS

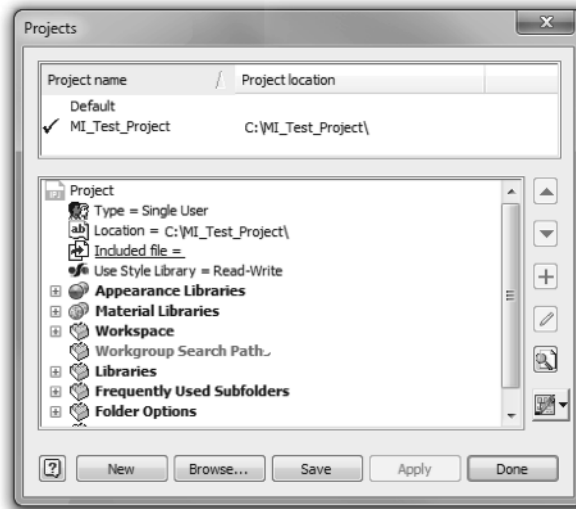
Only one project can be active at a time. To switch projects, you must first close all files that are open in Inventor. You cannot edit the file paths of the active project, but you can edit the file paths of items such as Content Center libraries that are included. You can edit anything in an inactive project.

Certification Objective

THE PROJECTS DIALOG BOX

Now that you have created your sample project file, you'll explore the options and settings available for your new project. To activate and use your new project, highlight the new project and click Apply. You can also activate or select a new project link by double-clicking the project link. Notice the check mark next to the project name `MI_Test_Project`, indicating that the project is now active, as shown in Figure 1.28.

FIGURE 1.28
Projects dialog box



In the lower pane, you can view and access parameter settings for the following:

- ◆ Project type
- ◆ Project location
- ◆ Optional included project file
- ◆ Appearance libraries path
- ◆ Material libraries path
- ◆ Workspace path
- ◆ Optional workgroup paths

- ◆ Libraries you want to use
- ◆ Frequently used subfolders
- ◆ Folder options
- ◆ Other project options

Right-click a parameter group to view the settings available within that group. In the Project group, you can change the project type, view the project location, and include other project files. Project types were discussed earlier in this chapter. The project location is a read-only parameter. Included files deserve some additional discussion because the Included File parameter allows you to apply a master project to your current project; this setting, as well as the other project settings, is discussed in the coming pages.

Included Files

Although it's not required, you can include an existing project in the configuration of the current project by right-clicking Included File. The properties and settings in the project file that you attach override the settings in the current project file. This is useful for restricting and controlling a user's ability to change the project file. Also, if you frequently create new project files, you might consider creating a master project file that contains library locations and other settings you commonly use and then including the master project file in each new project file.

Appearance Libraries

An appearance library is a collection of appearances (for instance, colors) that either are installed with Autodesk Inventor or are created by the user. User-created libraries can contain appearances and materials in the same appearance library. The Autodesk Appearance Library contains only appearances. The active library is shown in bold text.

Material Libraries

A material library is a collection of material definitions (for instance, physical materials such as Stainless Steel) that either are installed with Autodesk Inventor or are created by the user. User-created libraries can contain materials and appearances in the same material library. The Autodesk Material Library contains only materials. The Inventor Material Library provides a basic set of manufacturing-related materials and appearances. The libraries that are installed with Autodesk products are read-only libraries. Custom libraries can be set to read-only or made writable using the file's read-only attribute. The active library is shown in bold text.

Workspace

For single-user projects, the workspace is defined by the location of the project file (*.ipj). For Vault projects, the workspace is defined on the workstation and is configured to match the Working Folder setting in the Vault settings. The workspace is the folder that files are copied to when they are checked out. The Workspace folder may include any number of subfolders as required for your file management needs.

Workgroup

The workgroup search path specifies a location outside the current project file paths where Inventor can search for existing files that are not included in a library. A workgroup is specified when the project is created. Each single-user project should have no more than one workgroup. Using a workgroup path is not required and is not a common configuration to make.

Style Library

Inventor uses styles to specify dimensions, text, colors, materials, and other properties. This is similar to styles used in AutoCAD. Inventor stores styles locally within the templates or in an external style library that may be used with any project file.

The Use Style Library function in projects specifies whether the project uses only read-only or read-write style libraries. The read-only style library is recommended for projects that have multiple users. With multiple users, changing or editing the style library on the fly can cause downstream problems. To change the Use Style Library parameter, right-click and select the new setting.

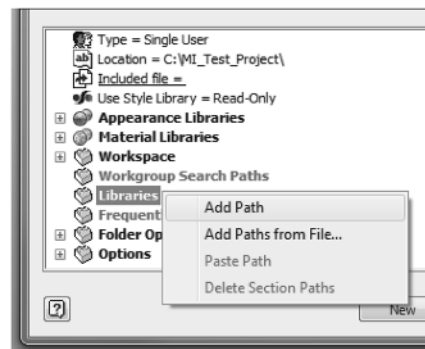
Remember that for your projects, you can right-click to select another option when it is appropriate. Click Read-Only (the default) if you want to access style libraries and local styles without enabling style-editing capabilities. Click Read-Write if you want to be able to coordinate style by saving them into the style library.

Library Options

Next on the list are libraries. Library folders are usually located outside the project file path. They may be located anywhere on your system or on your server. If you are sharing library files, you should place them on your server in a commonly accessed location. Inventor considers the contents of directories specified as libraries to be read-only.

In your newly created project file, you have not added any library folders. If at any time you want to add library folders, you can do so by right-clicking Libraries and choosing Add Path, Add Paths From File, or Paste Path, as shown in Figure 1.29.

FIGURE 1.29
Adding library paths by right-clicking



You can manually add a path, either by browsing or by typing a new file location. Be sure to give the library a descriptive name that identifies the contents of the file location. Add Paths From File permits you to extract library paths from another project file. Paste Path allows you to copy and paste. Once you have specified library paths, the Delete Section Paths option becomes

available, and you can remove paths not needed by the project. Deleting unused library paths reduces search and resolution time.

Shortcuts to Frequently Used Files

Frequently used subfolders are similar to the bookmarks you can set in Internet Explorer. The subfolders must already be nested within the current project workspace, workgroup, or library. Adding frequently used subfolders to your project provides navigation links in your open, save, and place dialog boxes so you can quickly navigate to those locations. In the Mastering Inventor project that you will use throughout the rest of this book, the chapter folders have been added as frequently used subfolders.

Folder Options

The Folder Options setting allows your project to access other file locations that are specified on the Files tab of the Application Options dialog box. Keep in mind that you may have to close and reopen Inventor in order to reinitialize the optional project file locations. You can use this option to specify different default locations for templates, design data (styles, and so on), and Content Center files. When the locations are set to the defaults, the location/storage of the files is specified on the Files tab of the Application Options dialog box. Right-click any of the option entries to change the storage and access location. You can find more information about Content Center in Chapter 7, “Reusing Parts and Features.”

Project Options

Expand the Options heading to show the global defaults for the selected project. The Options settings in a project determine file management functions; right-click an option to edit it.

Versioning and Backup

Use the Options settings to determine how many old versions or backup copies of each file to save. The Old Versions To Keep On Save option specifies the number of versions to store in the Old Versions folder for each file saved. The first time a file is saved in a project, an Old Versions folder for that file is created. When the file is saved, the prior version is moved automatically to the file's Old Versions folder. After the number of old versions reaches the maximum in the setting, the oldest version is deleted when a newer version is moved into the folder.

If you are familiar with AutoCAD, you may expect Inventor versioning to be similar to AutoCAD's backup scheme. AutoCAD creates a *.bak file saved in the same folder as the design. Inventor saves the backup files in a separate Old Versions folder. All versions located in the Old Versions folder have the same name and extension, except that a number is appended after the name. In the project options, the default setting of 1 creates one backup file in the Old Versions folder. If you are working with a complex model, you might decide to specify additional backup versions by changing this setting to a higher number; however, remember that with each additional backup version you are creating additional files (and using additional space) on your hard drive. Setting Old Versions to -1 will cause Inventor to save all backup files.

RESTORING AN OLD VERSION

Occasionally a file may become corrupt, or you may have accidentally saved a design change that you did not intend to. In these cases, you can browse to the **Old Versions** folder and open the versioned file. Upon doing so you will be presented with these three options:

- ◆ Open Old Version (Save Not Allowed)
- ◆ Restore Old Version To Current Version
- ◆ Open Current Version

Filenaming Conventions

The **Using Unique File Names** setting in the options will allow Inventor to check for unique part names for all files in the project, including subfolders. Libraries are excluded in this option. Proper design workflow demands that each unique part have a unique filename. When a part is reused, you should ensure that any revision to it is acceptable to all designs in which it is used. If the revised part cannot be used in all the designs, you should use a new part name because you have now created an additional unique part.

Setting the **Using Unique File Names** option to **Yes** will cause Inventor to search the entire project workspace to compare filenames but does not prevent users from creating duplicate filenames. Having this option set to **Yes** can cause issues when a large number of files are present, particularly when those files are organized with a large number of subfolders. If the **Using Unique File Names** option is set to **No**, Inventor will not search to compare filenames.

If you do not have a part-numbering scheme already, take the time to implement one to make working with your Inventor files easier. Keep in mind that the most effective numbering schemes are often the simplest. Many an engineering department has eventually run into unanticipated limitations when using a numbering scheme that is too specific or when attempting to include too much information in some built-in code.



Real World Scenario

THE USING UNIQUE FILE NAMES SETTING AND SLOW SEARCH TIMES

The **Using Unique File Names** option for your project is set to **Yes**. You have an engineering directory with 8,500 folders and subfolders containing thousands of files. You have a drawing file that references a part file named `12-865.ipt`, but you've renamed the part file to `22-865.ipt` using Windows Explorer.

Because the internal link in the drawing file is still looking for `12-865.ipt`, when you open the drawing file that references the renamed part file, Inventor will search through all 8,500 folders and subfolders looking for the missing file. When it doesn't find the file, it will finally present the **File Resolution** dialog box, allowing you to point the drawing file to `22-865.ipt`.

continues

continued

Because of the number of folders and files, this search may take several minutes. By contrast, if you had the same scenario but your project file option Using Unique File Names was set to No, Inventor would not search for the missing file. Instead, it would immediately present you with the File Resolution dialog box and allow you to point the drawing to 22-865 .ipt. For this reason, you should set this option to No when large file collections are present. However, even if it is set to No, you should employ a unique filenames scheme for all your files.

Keep in mind that you can toggle this option to Yes if you find yourself needing to search the project for unique filenames.

Here are a few suggestions that may help you in determining a numbering scheme that will work well for you:

Sequential Project-Based: 0910-00001 Here, the first four digits correspond to the project number, and the last five are sequenced, starting at 00001. This numbering system works well when parts are not often used across different projects. Common parts might be created under a “common part project” prefix such as 7777.

Generic Date-Based: 09-0707-01 Here, the first six numbers are assigned by using the current date when the part number is created. In this example, the part number was created on July 7, 2009. The last two digits are sequenced starting with 01. This is a highly flexible numbering system that allows 100 part numbers to be created per day. If more than 100 are needed, then backdating using an unused date can be done. The date itself holds no real significance, other than helping to ensure the unique part numbers.

Sequential Product-Based: NG-00001 Here the first two letters reflect a specific product line, such as the Next Generation (NG) line, and the last five are sequenced starting at 00001. This numbering system works well when products lines are engineered and maintained separately.

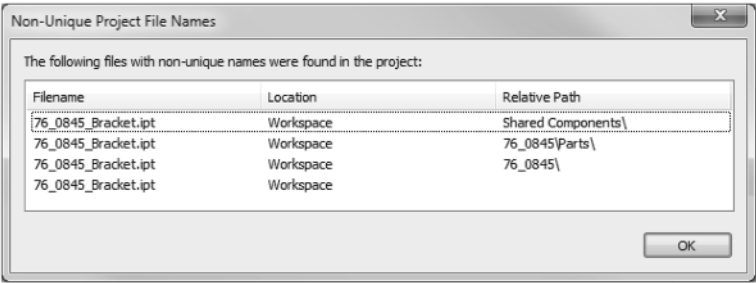
Once you’ve decided on a part numbering scheme, you will want to come up with a central part number log file or tracking system to be used in assigning numbers to ensure that there are no duplicates. If you have a resource planning system (commonly referred to as a manufacturing resource planning [MRP] or enterprise resource planning [ERP] system), you likely have the ability to manage part numbers using that software. If not, you can use a simple XLS spreadsheet file to assist with this task.

THE PROJECTS DIALOG BOX’S TOOL PANEL

The buttons along the right side of the lower pane of the Projects dialog box provide access to tools that allow you to add, edit, and reorder project parameter settings and paths; check for duplicate filenames; and configure Content Center libraries used for the active project.

Use the magnifying glass button located on the lower-right side of the Projects dialog box to check your project paths for duplicate filenames. Figure 1.30 shows the result of searching a project where someone has not been careful to ensure that duplicate filenames are not used.

FIGURE 1.30
The Non-Unique Project File
Names dialog box



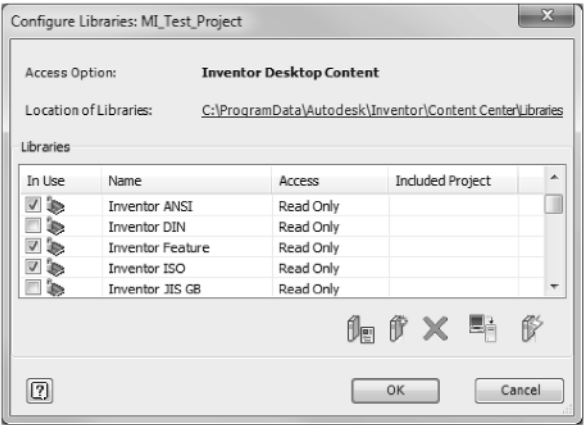
WHY RELATIVE PATHS?

An Inventor assembly file records relative paths when it links a subassembly or single parts to itself. The use of relative paths in assembly files allows the relocation of an assembly and its associated parts and subassemblies to other locations on servers or drives without requiring the resolution of a new location. Relative paths, however, introduce the danger of the assembly locating only the first of two parts that happen to have the same name. For instance, if you’ve saved files named Part1 in two different file folders, Inventor might resolve the link to the first one it finds and then stop searching. To prevent the possibility of the wrong part being loaded in an assembly, ensure that every part located in the search path has a unique name.

The Projects dialog box supports the configuration of one or more Content Center libraries. Content Center provides multiple database libraries that can be used in assemblies or by the design accelerator.

If you elected to install Content Center libraries while installing Inventor, you must configure Content Center libraries in the project before you can access them. Click the Content Center button in the lower-right corner of the project editing dialog box. Then select Content Center library or libraries you want to use and click OK. Figure 1.31 shows the Configure Libraries dialog box.

FIGURE 1.31
Configuring Content Center



Select the Content Center libraries you think you'll use. Installing all the Content Center libraries may slow your system down significantly when you are accessing Content Center because Inventor will need to index each library upon initialization.

When you finish editing the project file, click Save and then make sure your desired project file is active before clicking Done to exit the Projects dialog box.

Creating Multiuser Projects

Working as a team can increase productivity many times over. In a collaborative design environment, several users may be working on a project at the same time. When you use Vault data management software, you are prevented from working on the original version of a file located in Vault. Each user creates a local Vault project file that specifies a personal workspace located on the local drive and includes search paths to one or more master projects.

WHAT IS AUTODESK VAULT AND WHO NEEDS IT?

Autodesk Vault is a file management tool integrated with Inventor. Essentially, Vault allows you to check in and check out files in a collaborative workgroup so users do not accidentally save over one another's work. Not every design department needs to use Vault, and in fact many find that it is not a good solution for their particular needs. However, you should investigate Vault to see what it has to offer, even if you already have a data management system in place.

To edit a "Vaulted" file, the user must check the file out of Vault. The process of checking the file out copies the file to the local workspace. When someone checks out a file for editing, the original stored in Vault is flagged as "checked out" to that particular user. Other users can view the checked-out files in read-only mode, but they can't edit it.

The user who checked out the file can edit and save the file in their local workspace without checking the file back into Vault. When they save the file, they will be prompted to choose whether they want to check the file back into Vault. If they choose to check the file into Vault, the file will be saved into Vault and is then available for editing by a different user. Optionally, they may save the file into Vault but keep it checked out to their local workspace, allowing other users to view the updated file without being able to edit it.

PROJECT SHORTCUTS

If you right-click a project in the Project dialog box, you can choose Delete to remove it from the list. But if you browse to the file location, you'll see that the IPJ file is still there. What is going on?

When you create a new project file or point Inventor to an existing project file, Inventor will create a shortcut to that file. When you choose Delete and remove the project from the list, you are not actually deleting the IPJ file but instead deleting the shortcut. When you choose Browse and locate the IPJ file to add back to the list, the shortcut is re-created.

You can set the path to the shortcuts by accessing the Tools tab of the ribbon and choosing Application Options. In the Application Options dialog box, click the Files tab and set the project's folder path.

Collaborative design project files are created using the Inventor Project Wizard in much the same manner as a single-user project file is created. The file resolution process within a collaborative project file functions in the same way.

With Vault installed on your server or your own system, verify that Vault is correctly installed and that you can open and create a new Vault file store using the ADMS console. The new Vault file store must be accessible on your local system from Vault Explorer. If Vault functions correctly, you are now ready to create a Vault project file. As with a single-user project, use the Inventor Project Wizard to name the project, specify the workspace, assign libraries for use with the project, and configure project parameters.

Again, as in other project file types, you will need to edit the default settings in your project file and optionally configure your Content Center for use.

IMPLEMENTING AUTODESK VULT

Once the Vault software is installed on your system, you can find the Autodesk Vault Implementation Guide in PDF in the Help folder of the install directory. Or you can search the Internet for *autodesk+vault+implementation+guide*. This detailed guide provides you with information on Vault fundamentals and installation as well as information on configuring and maintaining Autodesk Vault for your data management needs.

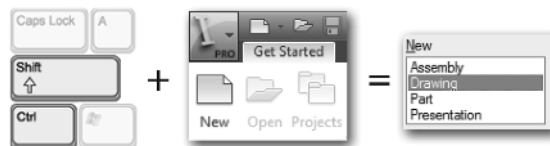
Understanding Inventor Templates

You can create template files in Inventor by opening an Inventor file, making the desired edits to the file, clicking the Inventor button, choosing Save Copy As, and then choosing Save Copy As Template. It is typically recommended that you set the Template directory by configuring your project file using the Folder Options node and setting the Template path to a network location. Often when creating a template, you might set up the following items in the file before saving:

- ◆ Custom iProperties
- ◆ User parameters
- ◆ Cached styles
- ◆ Specific document settings such as units (by clicking the Document Settings button on the Tools tab)

When creating any template file, it is a good idea to work with a file that is as “clean” and uncluttered with extra styles as possible. The best way to do this is to hold down the **Ctrl** and **Shift** keys and click the New button; then select the type of file you want to create from the list, as shown in Figure 1.32.

FIGURE 1.32
Creating a new clean file



Working with Styles, Style Libraries, and Company Standards

Your Inventor files will use a number of *styles* to allow you to maintain consistencies and to save you from needing to define common setups over and over. In part files, the style collections include lighting styles, color styles, and material styles. Color and lighting styles allow you to change the appearance of your model, and material styles allow you to change its physical properties. In drawing files, style collections include such things as dimension styles, balloon styles, parts list styles, and much more. These drawing style collections are organized in *standards*, which are used to manage the standard company styles used within your organization. You can even set up multiple standards, allowing you to change from one set of styles to another quickly and consistently.

The Bottom Line

Create parametric designs. The power of parameter-based design comes from the quick and easy edits when changing a parameter value drives a change through the design. To make changes easily, though, you need to follow certain general rules so that the changes update predictably.

Master It You want to create a model of a base plate, a rectangular-shaped part with a series of holes and rectangular cutouts. What would your initial sketch look like in Inventor?

Get the “feel” of Inventor. The interface contains many elements that change and update to give you the tools you need to perform the task at hand. Getting comfortable with these automatic changes and learning to anticipate them will help you get the “feel” of Inventor.

Master It You create an extrude feature using the Extrude button, but you cannot seem to find an Edit Extrude button. How can you edit the extruded feature to change the height?

Use the Inventor graphical interface. Inventor 2016 uses the ribbon interface introduced in Inventor 2010. Tools are grouped, which makes finding them intuitive once you become familiar with the basic layout.

Master It You are trying to draw a line on the face of a part, but you seem to have lost the Sketch tab in the ribbon. How do you get it back?

Work with Inventor file types. Inventor supports many different file types in its native environment, separating tasks and files to improve performance and increase stability.

Master It You have trouble keeping the various file types straight because all the file icons look rather similar to you. Is there a way you can see which file is what type?

Understand how project search paths work. Knowing how Inventor resolves file paths when it opens linked files, such as assembly files and drawings, goes a long way toward helping prevent broken links and repairing links that do get broken.

Master It What type of file does Inventor use to point the assembly file to the parts that it contains?

Set up library and Content Center paths. Library and Content Center paths are read-only library configurations set up in the project file.

Master It When you set up a library or Content Center path to a folder that does not exist, what happens?

Create and configure a project file. Project files are a key component of working successfully in Inventor, but for many people, this is a one-time setup. Once the project is created, for the most part you just use it as is.

Master It After creating a project file initially, you want to make one or more changes to the configuration, but you can't seem to do so. What could be the problem?

Determine the best project type for you. Although the Autodesk solution to a multiuser environment is Autodesk Vault, many people may not be able to use Vault. For instance, if you use another CAD application that links files like Inventor, Vault will likely not know how to manage the internal links for those files.

Master It Because you generally do not work concurrently on the same files as your co-workers, you think it might be best to set up a single-user project for now while you continue to investigate the Vault solution, but you are not sure if that will work. Can single-user projects be used in this manner?

