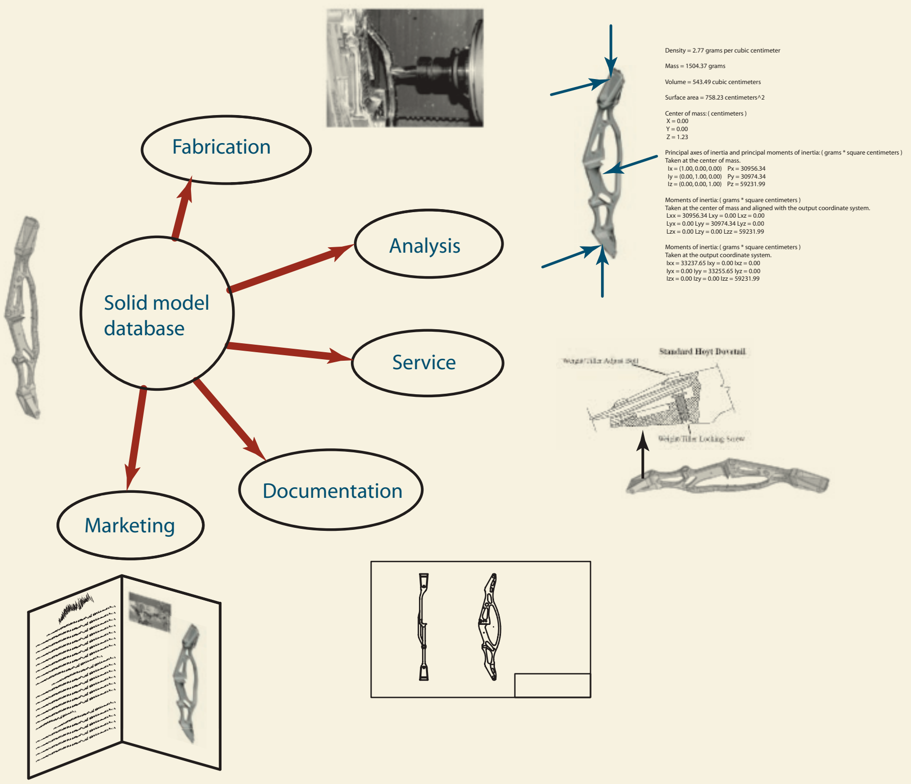
5.01 INTRODUCTION

Solid modeling is a computer-based simulation that produces a visual display of an object as if it existed in three dimensions. **Solid models** aid in forming a foundation for the product development process by providing an accurate description of a product's geometry and are used in many phases of the design process and life cycle of the product. This chapter will focus on methods for creating robust solid models of mechanical parts; however, these methods can be applied to other domains as well.

Solid models are created with specialized software that generates files for individual as well as assembled parts. These models are then used in a variety of applications throughout the design and manufacturing processes, as shown schematically in Figure 5.01. During the product concept stage, solid models are used to visualize the design. As the product is refined, engineers use solid models to determine physical properties such as the strength of the parts, to study how mechanisms move, and to evaluate how various parts fit together. Manufacturing engineers use solid models to create manufacturing process plans and any special tools or machines needed to fabricate or assemble parts. Solid models also can be used to generate formal engineering drawings to document the design and communicate details of the design to others. People responsible for the product life cycle may depend on solid models to help create images for service manuals and disposal documentation. Even sales and marketing functions use graphics generated from solid models for business presentations and advertising. Thus, it is very important not only to learn how to create solid models but also to understand how others will use the models. Solid models must be built with sound modeling practices if they are to be useful in downstream applications. In this chapter, you will learn how to create robust solid models that not only look like the real thing but also support the entire product life cycle. You also will learn about the history of CAD tools and the importance of solid modeling as part of an engineering design graphics system.

FIGURE 5.01. Uses for a solid model database.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

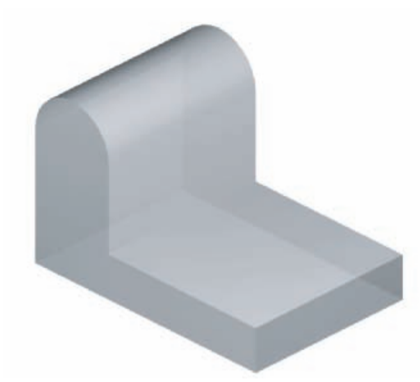


FIGURE 5.09. A surface model with semitransparent surfaces to reveal detail.

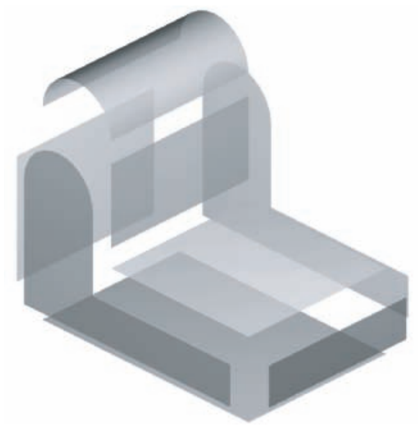


FIGURE 5.10. A surface model exploded to show individual surfaces.

Surface models evolved from wireframe models by mathematically describing and then displaying surfaces between the edges of the wireframe model. Thus, a surface model is a collection of the individual surfaces of the object. This modeling method is called **boundary representation**, or **b-rep**, because the surfaces "bound" the shape. The bounding entities of a simple part created using boundary representation are shown in Figure 5.10. The bounding entities can be planes, cylinders, and other surfaces in three dimensions. These surfaces are in turn bounded by simpler curve entities such as lines and arcs.

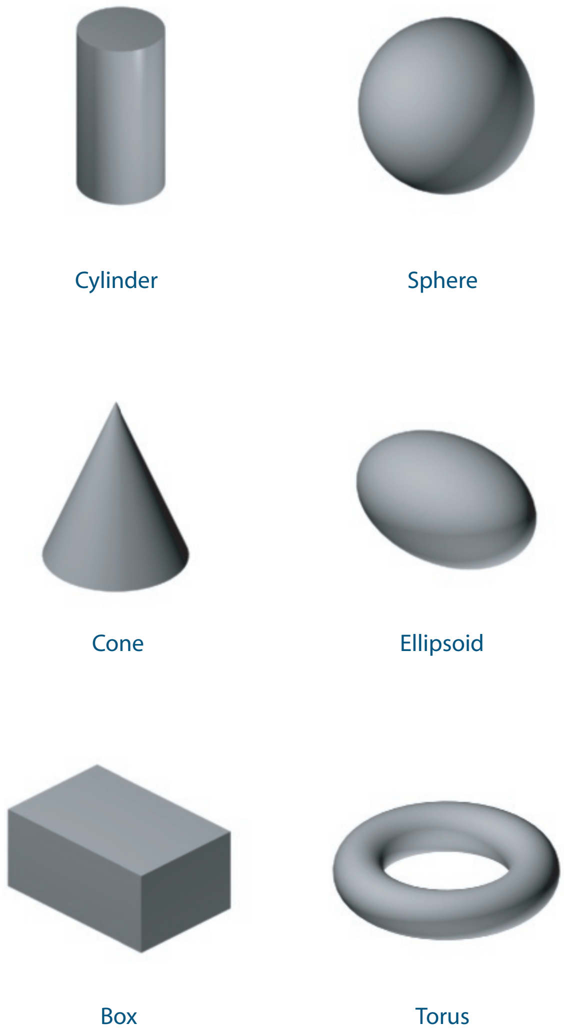
The use of surface models eliminates most of the problems with visual ambiguity encountered with wireframe models.

5.02.04 Solid Modeling

Solid models are visually similar to surface models, so it is sometimes difficult to distinguish between them. With a solid model, however, the software can distinguish between the inside and outside of a part and the objects can have thickness. Thus, the information stored in the 3-D database is sufficient to distinguish between an empty shoe box and a brick. The software also easily computes information such as the object's volume, mass, center of mass, and other inertial properties. Early solid models, developed in the late 1980s, were made using a technique known as **constructive solid geometry (CSG)**. CSG models are composed of standard building blocks in the form of simple solids such as rectangular prisms (bricks), cylinders, and spheres, called **primitives**. The shapes are easy to define using a small number of dimensions. Figure 5.11 shows some of these basic solids. To create more complex solids, the primitives are assembled using Boolean operations such as addition (union), subtraction (difference), and interference. Examples of these operations are shown in Figure 5.12.

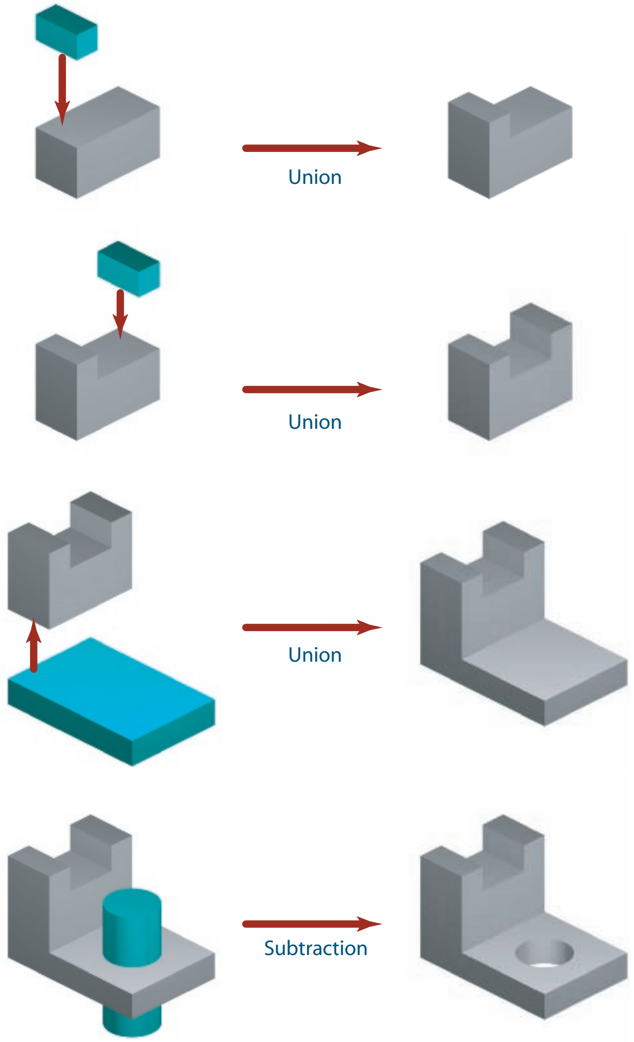
Surface and CSG models were very powerful tools for design, but their early versions were rather cumbersome to use. As computational resources improved, so did the capabilities of modeling software. Increasingly more sophisticated modeling methods, such as creating a solid model by moving or rotating a closed 2-D outline on a path through space, as shown in Figure 5.13, were developed. Further developments included software tools for taking many individual solid model parts and simulating their assembly into a larger structure, as explained in Chapter 1, and for easily creating formal engineering drawings for parts and assemblies from their solid models.

A more accurate and efficient modeling tool called **feature-based solid modeling** was developed in the mid-1990s. This modeling method permitted engineers and



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

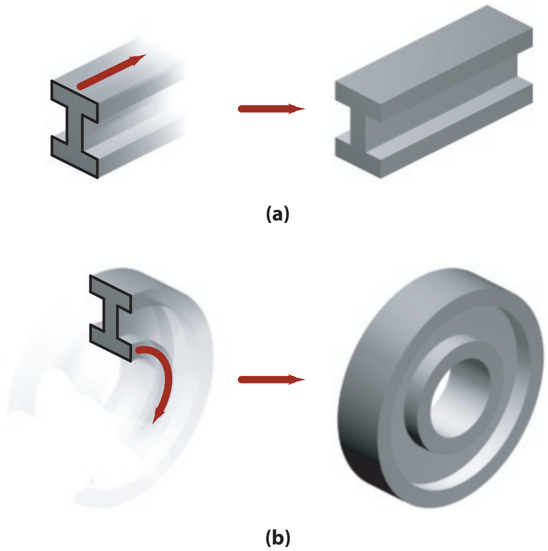
FIGURE 5.11. Some 3-D primitives used in solid modeling.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.12. Steps in using solid primitives to build a more complicated solid model using Boolean operations.

FIGURE 5.13. Solids created by (a) moving and (b) revolving a 2-D outline through space.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

designers to create a more complex part model quickly by adding common features to the basic model. Features are 3-D geometric entities that exist to serve some function. One common and easily recognizable feature is a hole. Holes in a part exist to serve some function, whether it is to accommodate a shaft or to make the part lighter. Other features, such as bosses, fillets, and chamfers, will be defined later in this chapter.

Parametric solid modeling is a form of feature-based modeling that allows the designer to change the dimensions of a part or an assembly quickly and easily. Since parametric feature-based solid modeling is currently considered the most powerful 3-D CAD tool for engineers and designers, the remainder of this chapter will be devoted to this modeling method.

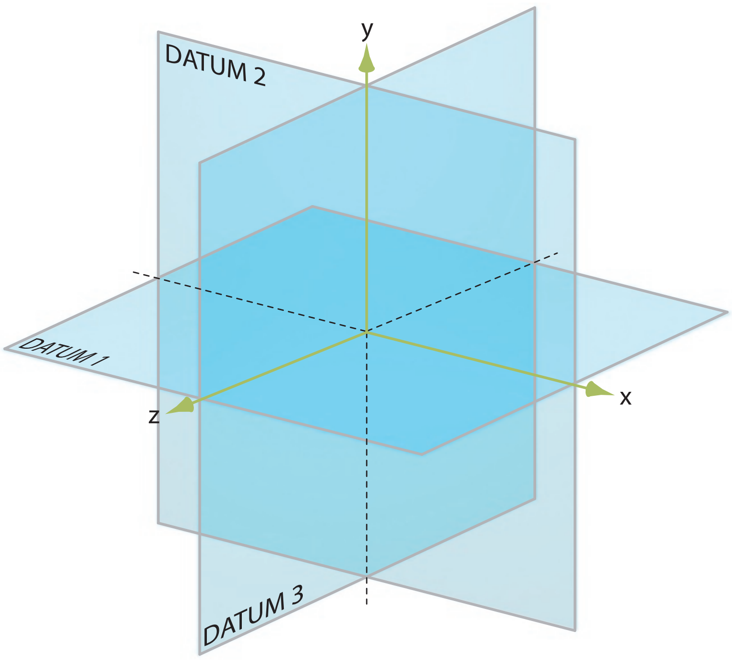
5.03 A Parametric Solid Model

So how does one go about creating a parametric feature-based solid model? In this section, a very simple model will be created to demonstrate basic concepts. More detail and sophistication will be presented in subsequent sections of this chapter. The tools that you need to create a parametric model are solid modeling software and a computer that is powerful enough to run the software. As you create the model, the software will display an image of the object which can be turned and viewed from any direction as if it actually existed in three dimensions.

Using the mouse and keyboard, you will interact with the software through a **graphical user interface (GUI)** on the computer's display device (i.e., the computer monitor). The GUI gives you access to various tools for creating and editing your models. GUIs differ slightly in different solid modeling software. However, most of the packages share some common approaches. When creating a new model (i.e., with nothing yet existing), you will probably be presented with a display of 3-D Cartesian coordinate x-, y-, and z-axes and the three **primary modeling planes**, which are sometimes called the **principal viewing planes** or **datum planes**. These planes help you visualize the xy-, yz-, and xz-planes and are usually displayed from a viewing direction from which all three planes can be seen, as shown in Figure 5.14.

Nearly all solid modelers use 2-D **sketches** as a basis for creating solid features. Sketches are made on one of the planes of the model with a 2-D sketching editor

FIGURE 5.14. The primary modeling planes for solid modeling.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

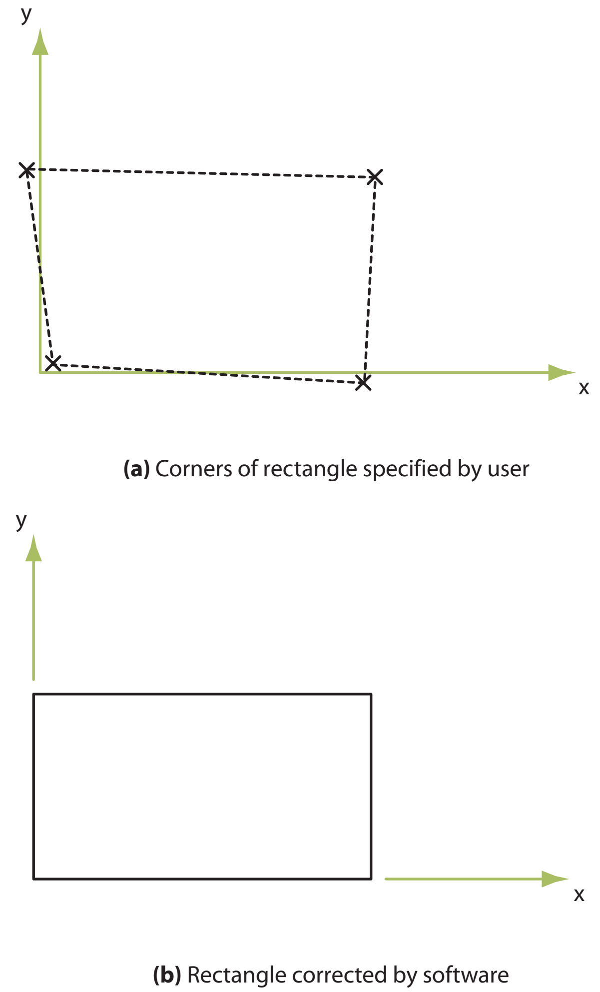
similar to a drawing editor found on most 2-D CAD drafting software. When you begin a new model, you often make a sketch on one of the basic modeling planes. When the sketching plane is chosen, some modelers will reorient the view so you are looking straight at the 2-D sketching plane. You can then begin sketching.

Line segments are usually inserted using mouse clicks, as shown in Figure 5.15a. A sketch is initially created without much attention being paid to precise dimensions and exact orientations of the different segments. For convenience, the **sketching editor** in most solid models automatically corrects sloppy sketches by making assumptions about the intended geometry. For example, if a line segment is sketched almost vertically or almost horizontally, the sketching editor will force the line into a vertical or horizontal orientation. Figure 5.15a shows a sketch of a rectangle created by clicking the four corners, or vertices; Figure 5.15b shows the cleaned-up sketch after the sketching editor corrects the user input and reorients the line segments.

5.03.01 Valid Profiles

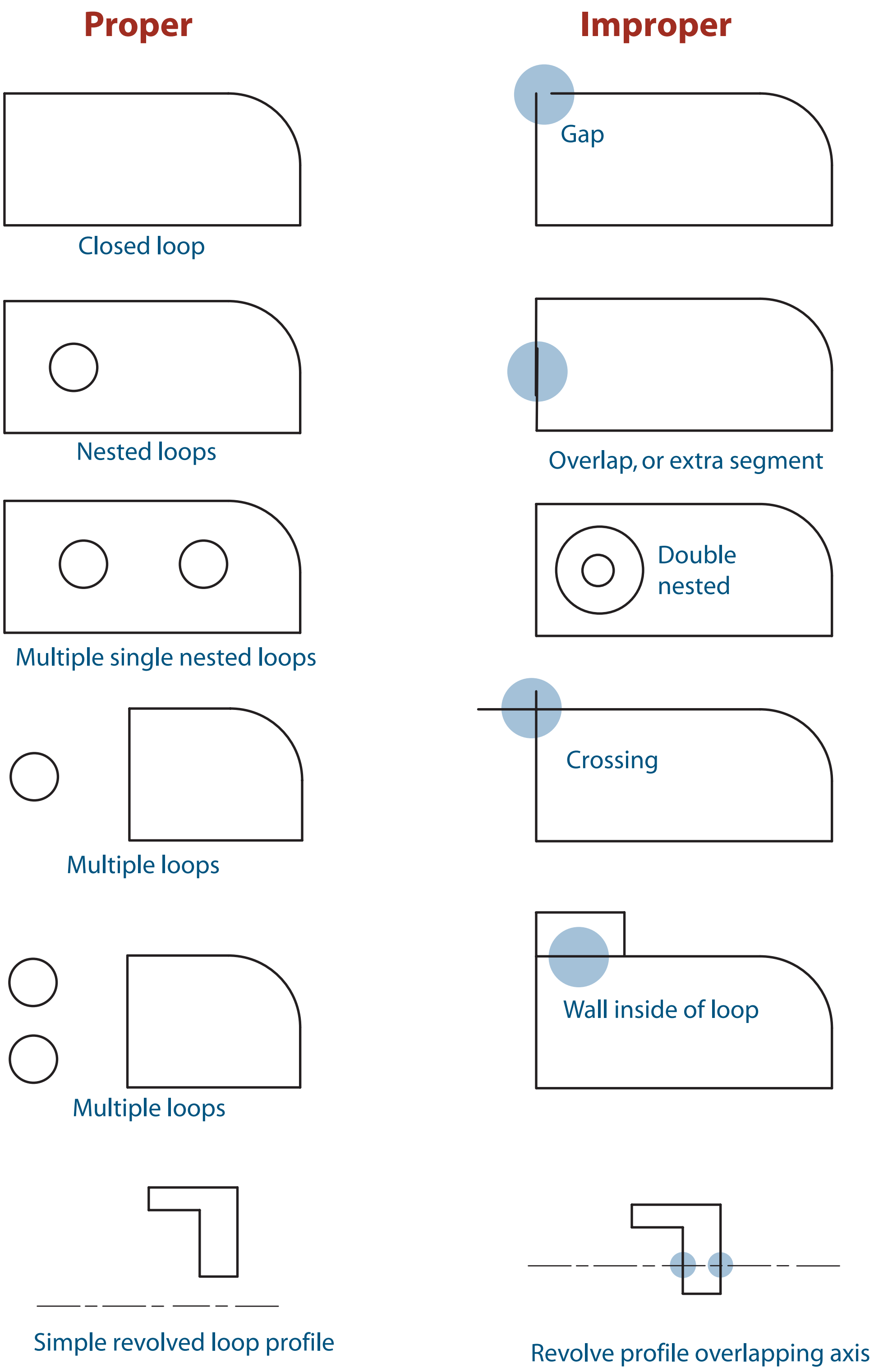
Before a solid feature can be created by extrusion or rotation, the final profile of the shape must be a closed loop. Extra line segments, gaps between the line segments, or overlapping lines create problems because the software cannot determine the boundaries of the solid in the model. Samples of proper and improper profiles are shown in Figure 5.16.

FIGURE 5.15. 2-D sketching.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.16. Examples of proper and improper profiles.

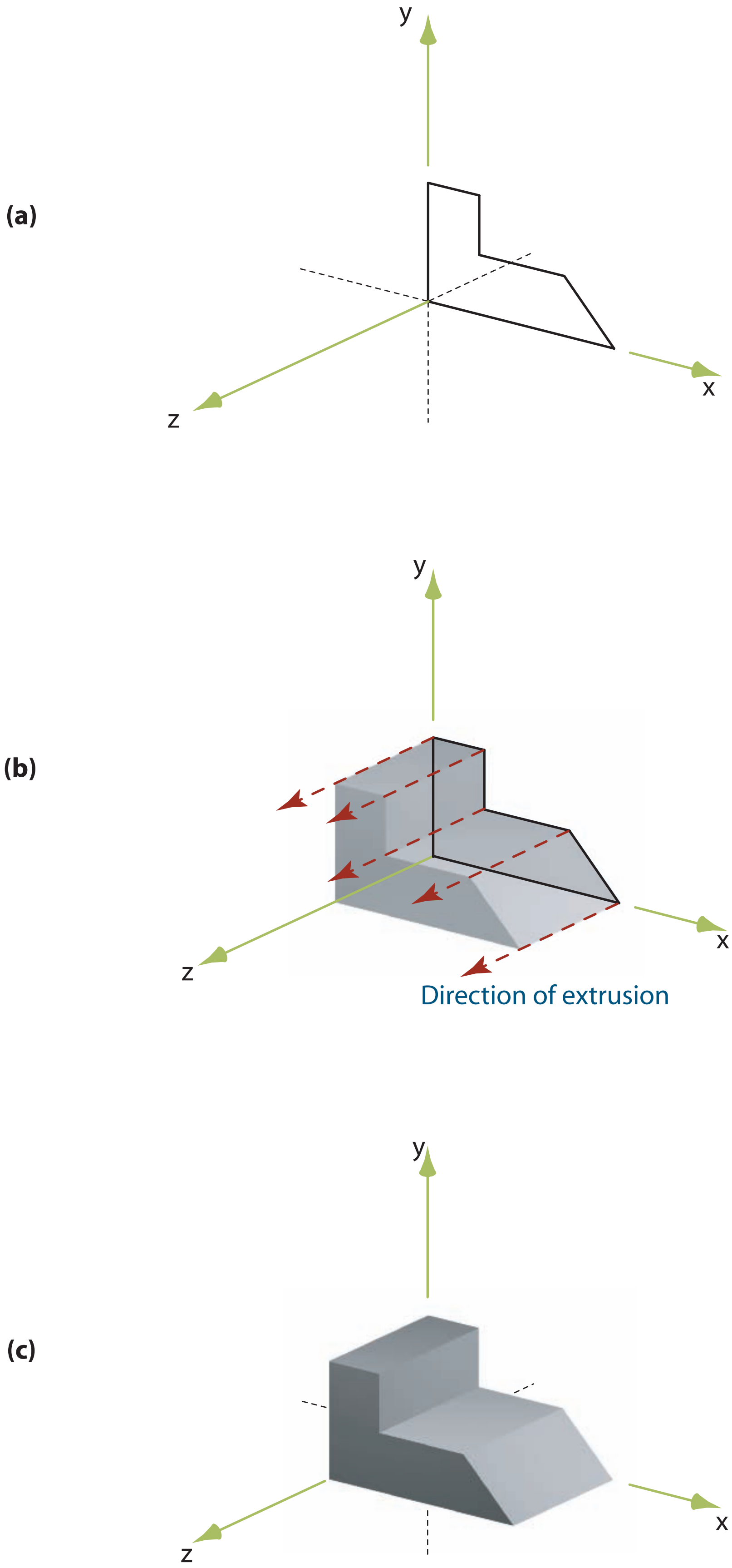


© 2009 Cengage Learning®. Courtesy D. K. Lieu.

5.03.02 Creation of the Solid

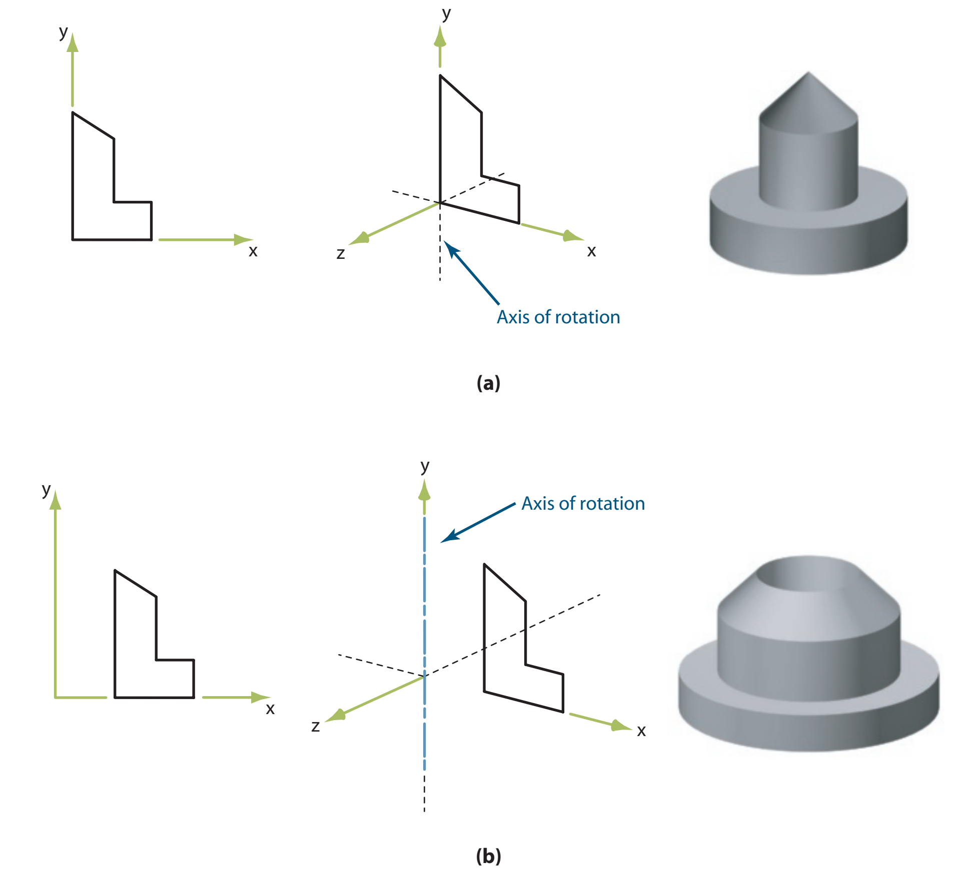
A completed sketch that is used to create a solid is called a **profile**. A simple solid model can be created from the profile by a process known as **extrusion**, as shown in Figure 5.17. Imagine the profile curve being pulled straight out of the sketching plane. The solid that is formed is bound by the surfaces swept out in space by the profile

FIGURE 5.17. A solid created by extrusion of a 2-D profile.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.18. A solid created by rotation of a 2-D profile, with the axis on the profile in (a) and with the axis off the profile in (b).



as it is pulled along the path. Both the geometry of the profile and the length of the extrusion must be specified to define the model fully.

A different model can be created from the profile by a process called revolution. To create a **revolved solid**, a profile curve is rotated about an axis. The process is similar to creating a clay vase or bowl on a potter's wheel. The profile of a revolved part is also planar, and the axis of revolution lies in the profile plane (sketching plane). One edge of the sketch may lie along the axis of revolution, as shown in Figure 5.18a; or the sketch may be offset from the axis, as shown in Figure 5.18b. It is important to make sure that the profile does not cross over the axis of revolution. This would create a self-intersecting model (i.e., a solid created inside another solid), which most solid modeling software interpret to be a geometric error. The geometry of the profile and the angle of rotation must be specified to define the model fully. The models shown in Figure 5.18 are revolved through a full 360 degrees.

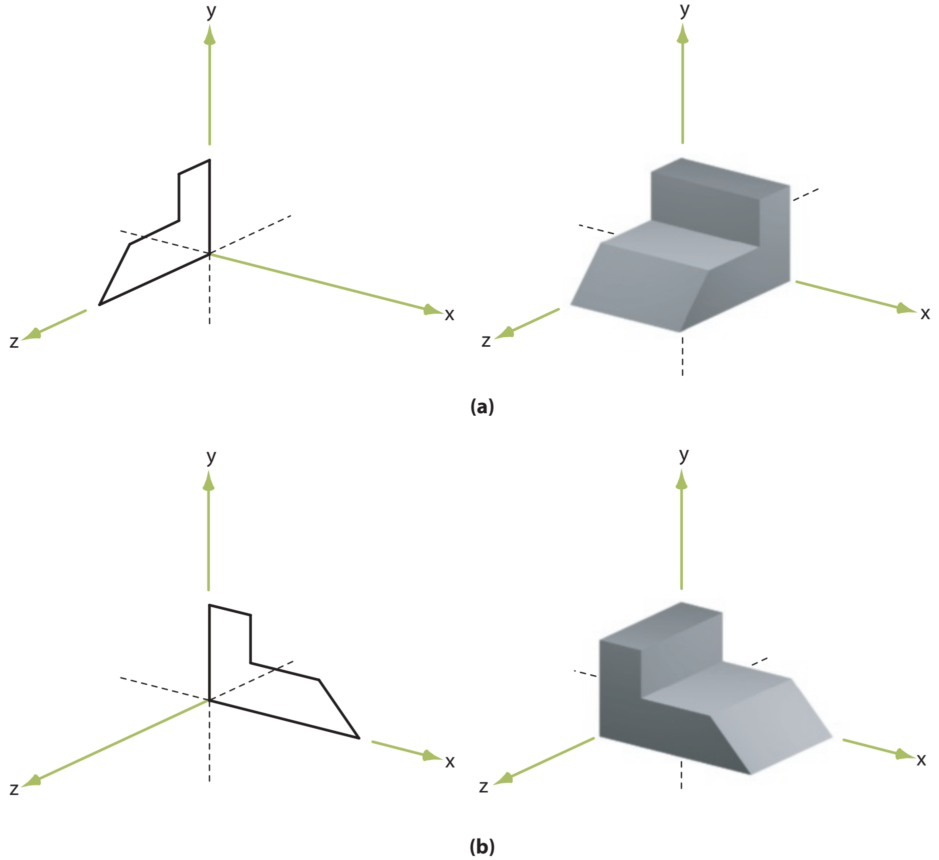
5.04 Making It Precise

Before a part can be submitted for analysis or fabrication, the sizes and locations of all of its features must be completely specified. To see how this is done, let's back up a few steps in our discussion of the creation of the model.

5.04.01 Orientation of the Sketch

Before you begin to create the first extrusion or revolution, you must decide where to place the part in the space relative to the xyz-coordinate system. With the model shown in Figure 5.17, the initial sketch was placed on one of the basic modeling

FIGURE 5.19. The same profile made in different sketching planes produces the same object but in different orientations. In (a), the profile is made in the yz-plane; and in (b), the profile is made in the xy-plane.



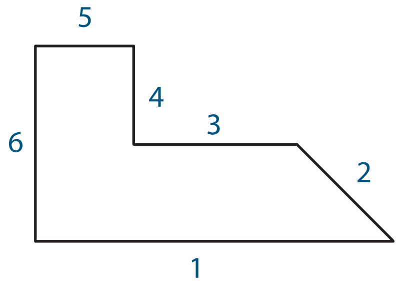
© 2009 Cengage Learning®. Courtesy D. K. Lieu.

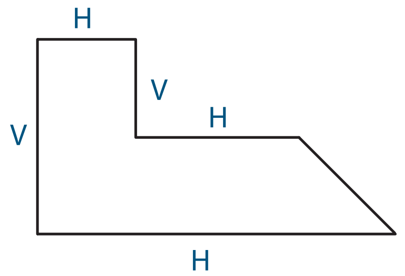
planes. If the sketch was placed on one of the other basic modeling planes instead, the model would have the same geometry but with a different orientation in space, as shown in Figure 5.19.

5.04.02 Geometric Constraints

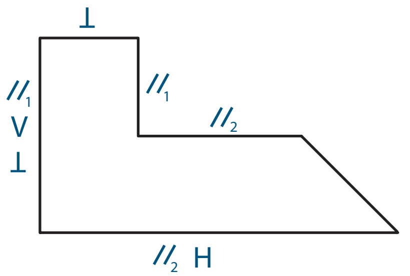
Formally, **constraints** are the geometric relationships, dimensions, or equations that control the size, shape, and/or orientation of entities in the profile sketch and include the assumptions that the CAD sketcher makes about your sloppy sketching. Constraints that define the size of features will be discussed in the following section. The previous section provided a few examples of **geometric constraints** that were applied to a simple sketch: lines that were drawn as nearly horizontal were assumed to be horizontal, and lines that were drawn as nearly vertical were assumed to be vertical. Those assumptions reduce the number of coordinates needed to specify the location of the endpoints. Some solid modelers require you to constrain the profile fully and specify the sizes and locations of all of its elements before allowing the creation of a solid feature; others allow more free-form sketching. Geometric constraints may be either implicitly defined (hidden from the designer) or explicitly displayed so you can modify them. A set of geometric constraints is not unique, as demonstrated in Figure 5.20. In this example, a set of geometric constraints that restricts some lines to being horizontal or vertical is equivalent to another set of constraints that restricts some lines to being either parallel or perpendicular to each other.

FIGURE 5.20. The line segments in a profile are numbered in (a). The implied geometric constraints for each segment are shown in (b), and an equivalent set of applied constraints is shown in (c). A letter or symbol beside a segment signifies the type of geometric constraint applied to it.





Segment	Constraint
1	Horizontal
3	Horizontal
4	Vertical
5	Horizontal
6	Vertical



Segment	Constraint
1	Horizontal, parallel to 3
3	Parallel to 1
4	Parallel to 6
5	Perpendicular to 6
6	Vertical, parallel to 4, perpendicular to 5

Geometric constraints specify relationships between points, lines, circles, arcs, or other planar curves. The following is a list of typical geometric constraints. The results of applying the constraints are shown graphically in Figure 5.21.

- Coincident—forces two points to coincide
- Concentric—makes the centers of arcs or circles coincident
- Point on Line—forces a point to lie on a line
- Horizontal/Vertical—forces a line to be horizontal/vertical
- Tangent—makes a line, a circle, or an arc tangent to another curve
- Colinear—forces a line to be colinear to another line
- Parallel—forces a line to be parallel to another line
- Perpendicular—forces a line to be perpendicular to another line
- Symmetric—makes two points symmetric across a centerline

The sketching editors in most solid modeling software are usually configured to try to interpret the user's sketching intent such that certain constraints are created automatically. In addition to adjusting nearly horizontal or vertical lines into true horizontal or vertical lines, if two lines are nearly perpendicular or parallel or an arc and a line are nearly tangent at the common endpoint, the sketching editor will impose the assumed geometric relationship. These automatically applied geometric constraints can be changed at a later time if desired.

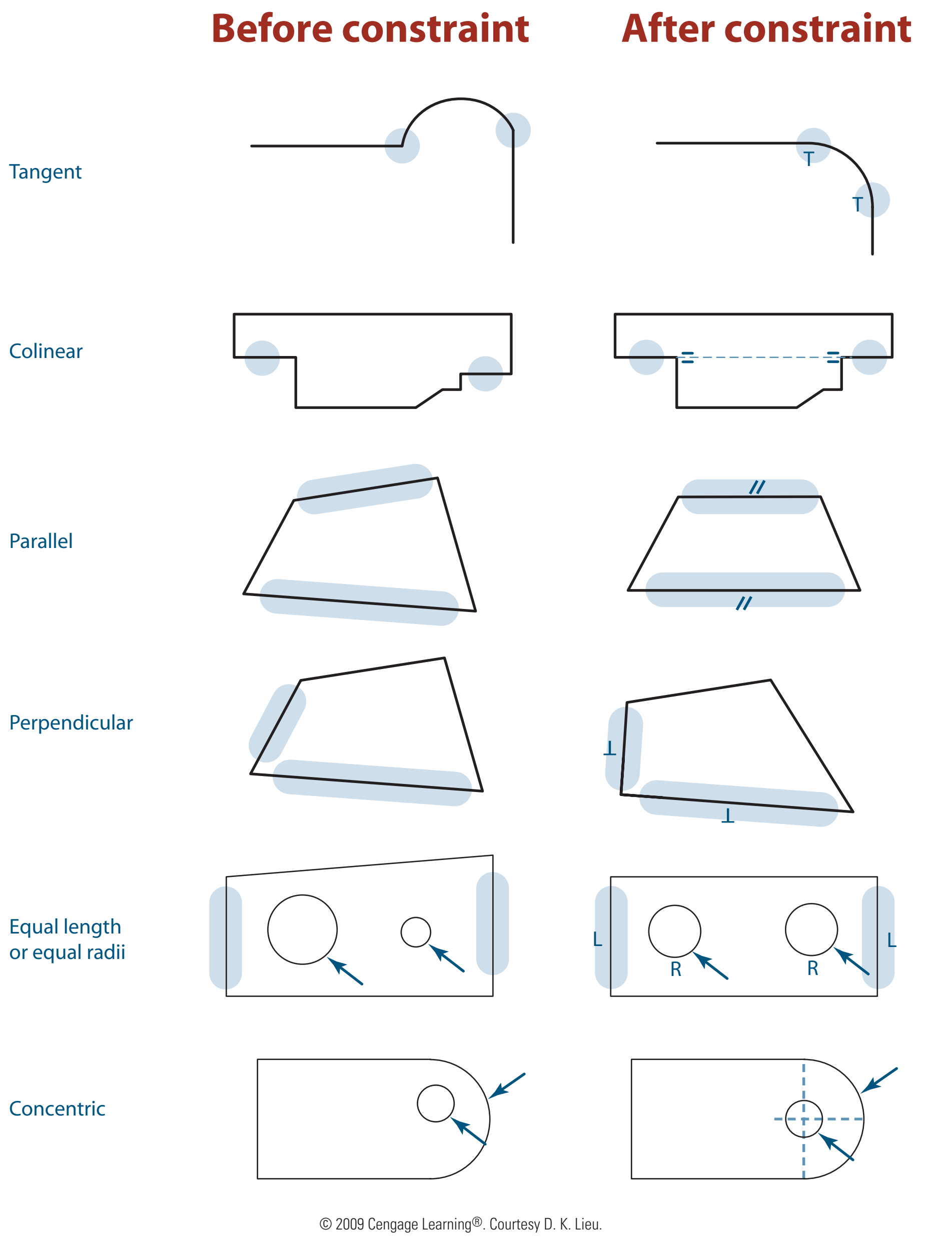


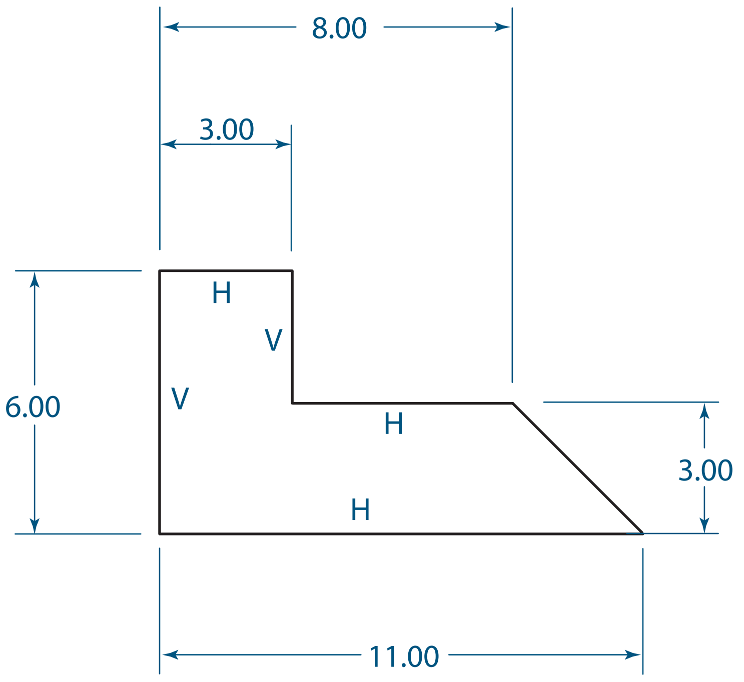
FIGURE 5.21. Geometric constraints commonly found in sketching editors.

5.04.03 Dimensional Constraints

Each of the 2-D entities in the profile must have size and position. **Dimensional constraints** are the measurements used to control the size and position of entities in your sketch. Dimensional constraints are expressed in units of length, such as millimeters, meters, inches, or feet. For example, look at the profile in Figure 5.22, which shows dimensional constraints that define its size. If you, the designer, do not fully specify all of the necessary information, the software will default to some value that you may not want. It is better if you control the model, rather than have the software assign assumed parameters and conditions to the model.

Dimensional constraints can be created interactively while you are sketching, but also automatically as a result of a feature operation, an extrusion, or a revolution. There are three principal types of dimensional constraints:

- Linear dimensional constraints define the distance between two points, the length of a line segment, or the distance between a point and a line. Linear dimensions can be measured horizontally or vertically or aligned with the distance being measured.



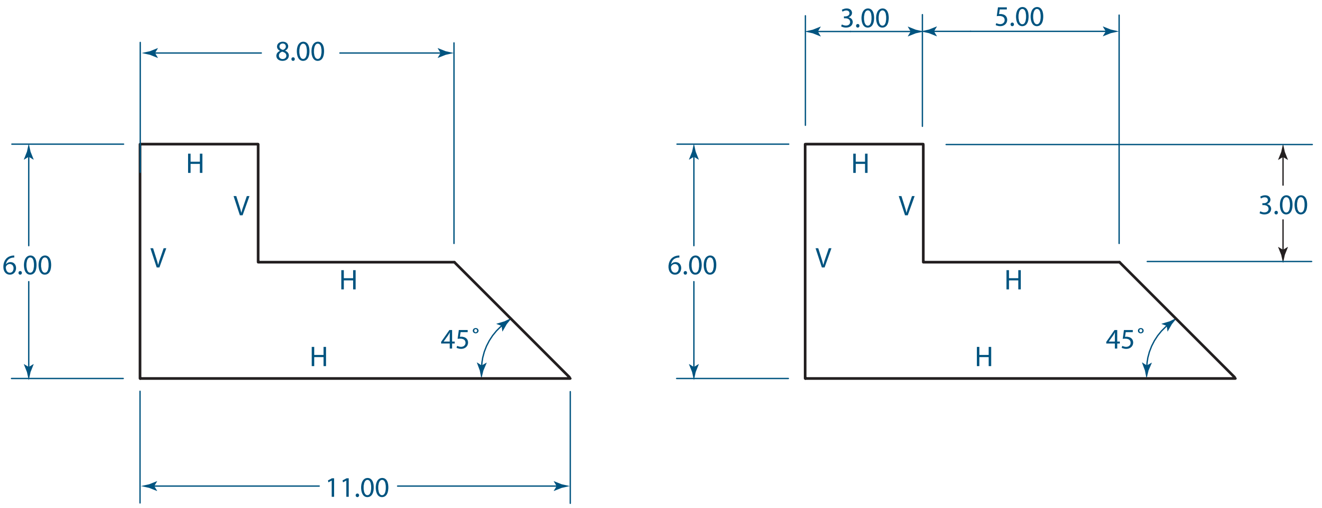
© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.22. A profile fully constrained with geometric and dimensional constraints.

- Radial and diametral dimensional constraints specify the radius or diameter of an arc or a circle.
- Angular dimensional constraints measure the angle between two lines. The lines do not need to intersect, but they cannot be parallel.

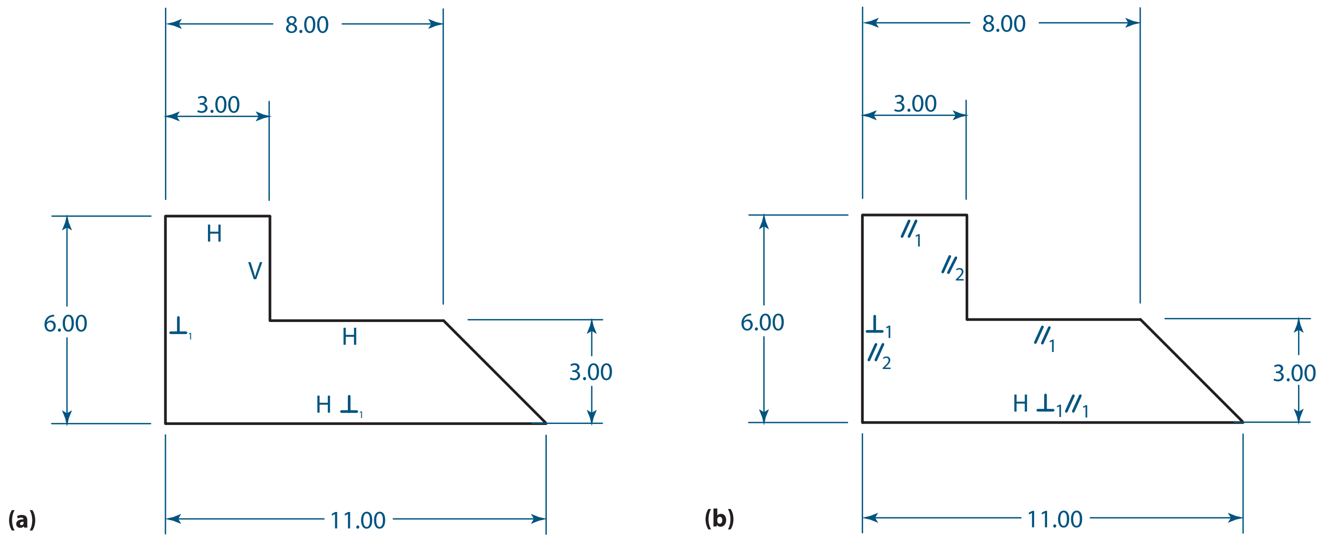
5.04.04 Uniqueness of Constraints

A set of dimensional constraints is not unique. It is possible to apply a different set of dimensional constraints on a profile to produce exactly the same geometry, as shown in Figure 5.23.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.23. Two different sets of dimensional constraints that can be used to define the same geometry.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.24. Two different sets of geometric constraints that define the same geometry.

Combinations of dimensional and geometric constraints also are not unique. It is possible to have different combinations of geometric constraints and dimensional constraints define exactly the same geometry, as shown in Figure 5.24.

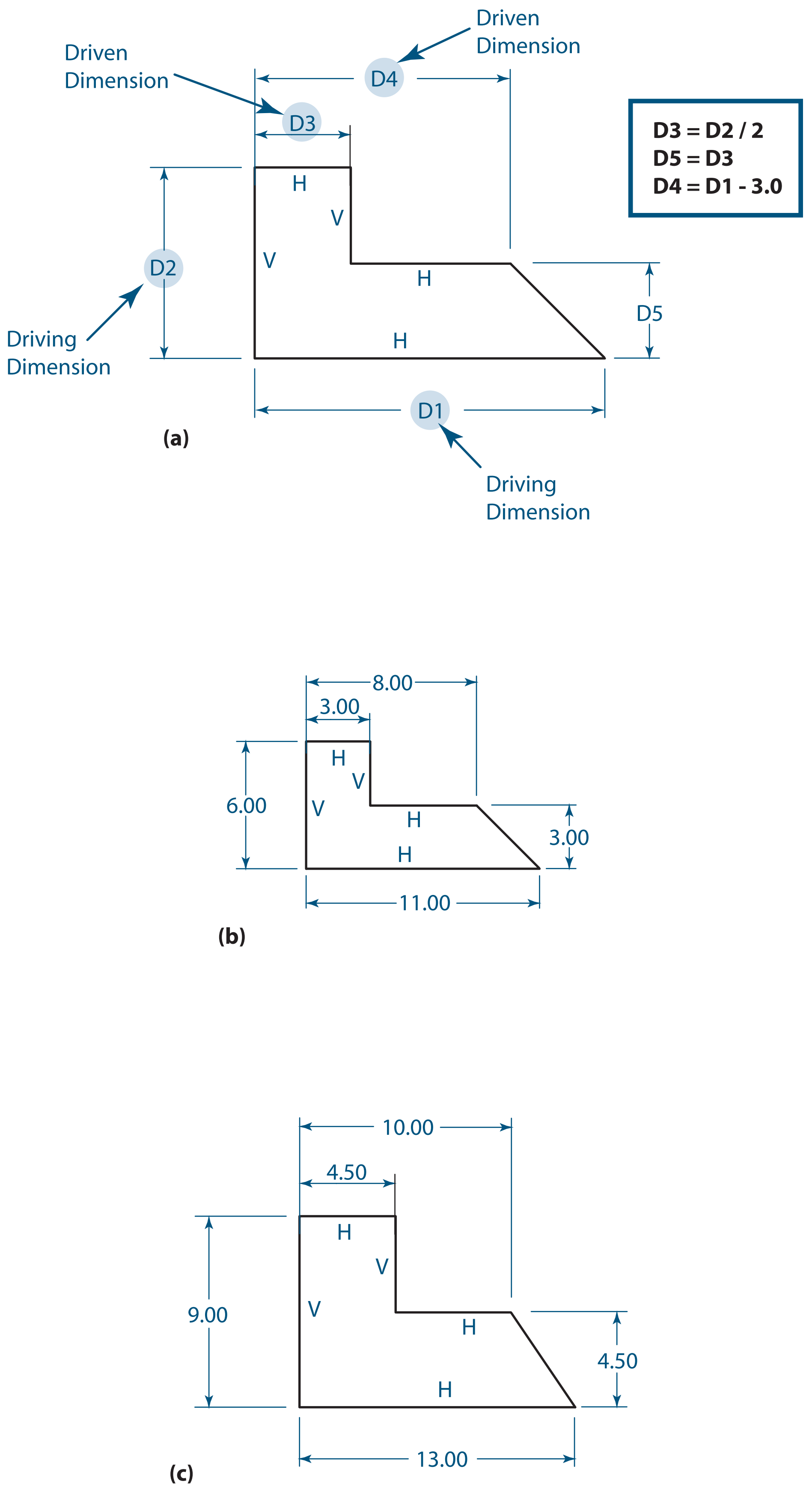
The natural question then becomes, which set of constraints is correct or preferred? The answer depends on what the function of the part and the design intent is or how the designer wants to be able to change the model. You also should consider how the solid model will be used for analysis, manufacturing, and documentation when applying sets of constraints. One of the greatest advantages of a parametric solid model is that the model can be changed easily as the design changes. However, the constraints limit the ways in which the model can be changed.

5.04.05 Associative and Algebraic Constraints

Associative constraints, sometimes called **algebraic constraints**, can be used to relate one dimensional constraint to another. The dimensional constraints on a profile are expressed in terms of variables. Each dimensional constraint is identifiable by a unique variable name, as shown in Figure 5.25. Algebraic constraints can be used to control the values of selected variables as the result of algebraic expressions. Algebraic expressions consist of constants and variables related to each other through the use of arithmetic functions (+, −, ×, absolute value, exponent, logarithm, power, square root, and sometimes minimum and maximum); trigonometric functions; and conditional expressions (if, else, or when) including inequalities comparisons (if A > B then …).

There are two different methods for solving sets of algebraic constraint equations. Software that uses **variational techniques** solves the equations simultaneously. A compatible solution for all of the variables can be calculated when there are a sufficient number of equations. In a system using **parametric techniques**, the equations are usually solved in sequential order. The equations will have only one unknown variable. All other variables in the algebraic expression must be known for the value of the unknown variable to be calculated, which is called the dependent or **driven dimension**. The known variables are called the **driving dimensions**. As shown in Figure 5.25, when the value of a driving dimensional constraint is changed, the value of its driven dimensional constraints are automatically changed, too.

FIGURE 5.25. Dimensional constraints are shown in terms of variables and a set of algebraic constraints in (a). Dimensions D3, D4, and D5 are automatically specified by specifying dimensions D1 and D2 in (b). Dimensions D3, D4, and D5 change automatically when D1 and D2 are changed in (c).

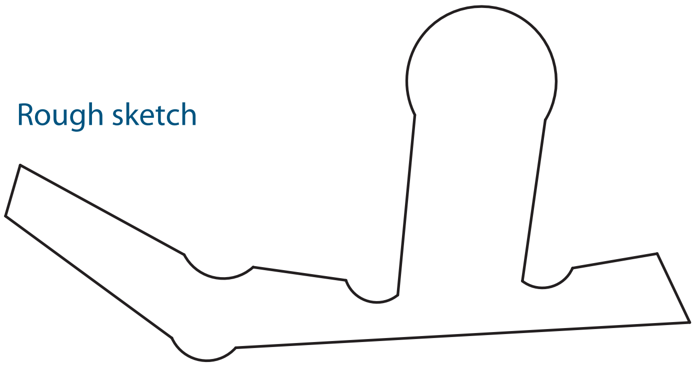


© 2009 Cengage Learning®. Courtesy D. K. Lieu.

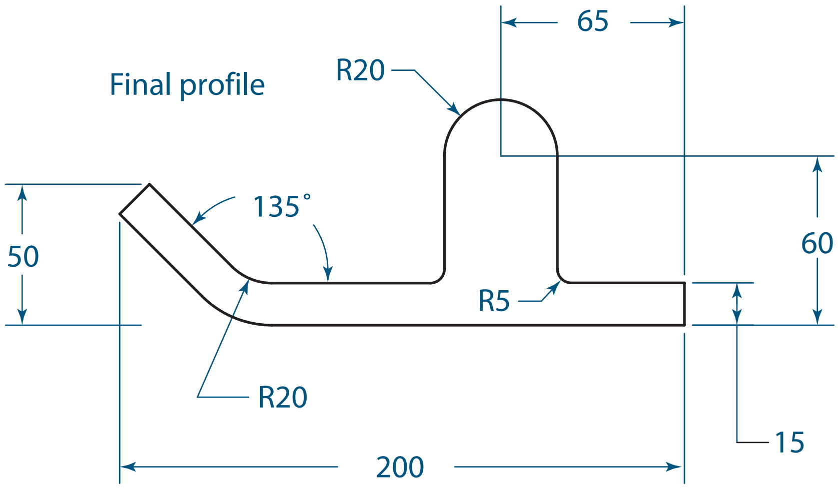
This constraint strategy demonstrates how to make your parts more robust. Through this simple example, you can see the importance of fully understanding the behavior of your model and the effects of your selection of dimensions and constraints. Your choices for geometric, dimensional, and algebraic constraints are not unique; but the decisions you make in selecting a set of constraints will have a big impact on the behavior of your model if you make changes to it. You must choose a modeling strategy that will reflect your design intent.

As an exercise for developing skill in the application of constraints, consider the rough sketch and the finished profile shown in Figure 5.28. For the profile to be fully constrained using only the dimensional constraints shown, certain geometric constraints are needed. Segment 1, for example, needs to be horizontal and tangent to Segment 2. Segment 2 needs to be tangent to Segment 1 as well as to Segment 3. Segment 3 needs to be tangent to Segment 2, perpendicular to Segment 4, and parallel to Segment 5. Segment 4 needs to be perpendicular to Segment 3 and equal in length to Segment 14. These constraints and the required geometric constraints on the remaining segments are shown in Figure 5.29. Keep in mind that a set of geometric constraints may not be unique. Can you specify another set of geometric constraints for this example that would create the same profile with the same dimensional constraints?

FIGURE 5.28. Geometric constraints need to be applied to the rough sketch (a) to produce the desired, fully constrained profile (b).



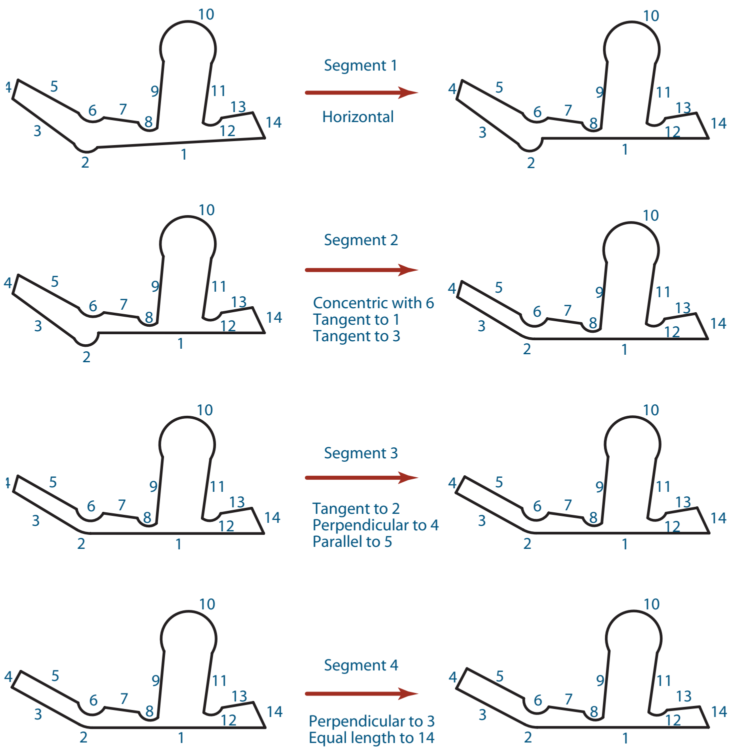
(a)



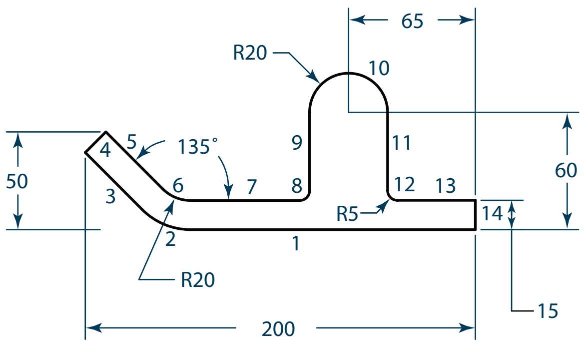
(b)

© 2009 Cengage Learning®. Courtesy D. K. Lieu.

Figure 5.29. Applying geometric constraints to the first four segments of the sketch in Figure 5.28a to produce the finished profile in Figure 5.28b.



(a)



Segment	Constraint
1	Horizontal, Tangent to 2
2	Concentric with 6, Tangent to 1, Tangent to 3
3	Perpendicular to 4, Parallel to 5, Tangent to 2
4	Equal Length to 14, Perpendicular to 3
5	Parallel to 3, Tangent to 6
6	Concentric with 2, Tangent to 5, Tangent to 7
7	Horizontal, Tangent to 6, Tangent to 8
8	Tangent to 7, Tangent to 9
9	Vertical, Tangent to 8, Tangent to 10
10	Tangent to 9, Tangent to 11
11	Vertical, Tangent to 10, Tangent to 12
12	Tangent to 11, Tangent to 13
13	Horizontal, Tangent to 12
14	Vertical, Equal Length to 4

(b)

5.06 More Complexity Using Constructive Solids

You have seen how to create solid models by sketching a 2-D profile on one of the basic modeling planes and then using a single extrusion or a single rotation to create a 3-D model. Adding material to or removing material from the original model can create a more complex model. When material is added, a **protrusion** feature is created. When material is removed, a **cut** feature is created. Both protrusions and cuts begin with sketched profiles that are then extruded or revolved to form solid shapes that are added to or removed from the existing body of the model. For an extruded feature, the profile lies in the sketch plane and is extruded in a direction perpendicular to the sketching plane. For a revolved feature, the profile and the axis of revolution must be coplanar so both will lie on the sketch plane.

When protrusions or cuts are made on an existing model, sketches and profiles are no longer restricted to be located on one of the basic modeling planes. Instead, any planar surface on the model can be selected and used as a **sketching plane** on which sketches and profiles can be created. Once a sketching plane has been selected, any 2-D element that is created will be forced to lie on that plane. After a sketching plane is selected, the model can be reoriented to look directly into the sketching plane. Although you can sketch when not looking directly into the sketching plane, you need to be very careful when viewing from a different orientation. Edges of your sketch may not be shown in their true shape, and angles may appear distorted. Most people find it easier to create 2-D profiles when they are looking directly into the sketching plane, just as it is easier for someone to draw straight lines and angles with correct measurements when the paper is oriented straight in front of them.

Examples of profiles on various sketching planes on a model and resulting extruded protrusions are shown in Figure 5.30; examples of extruded cuts are shown in Figure 5.31. Examples of revolved protrusions are shown in Figure 5.32, and examples of revolved cuts are shown in Figure 5.33.

As with the first extrusion or revolution that created the main body of the model, the profiles for the added protrusions or cuts must be fully defined by geometric, dimensional, and algebraic constraints before they can be extruded or revolved. A common geometric constraint for protrusions or cut features is to make one or more edges or vertices of the new profile coincident with edges of the surface used as the sketching plane. In Figure 5.30a, notice that one surface of the original object has been selected as a sketching plane and a rectangular profile has been sketched on the selected plane. The top and bottom edges of the sketched profile are coincident with edges of the sketching surface. The direction of extrusion is, by default, perpendicular to the selected sketching plane.

The length of the extrusion or angle of rotation also must be specified. There are several options for defining the length of the extrusion, as shown in Figures 5.30 and 5.31. The simplest is to specify a **blind extrusion**. A blind extrusion is one that is made to a specified length in the selected direction, analogous to specifying a dimensional constraint, as shown in Figure 5.30b. If your extrusion is the first feature used to create your initial model, it will be a blind extrusion. For a cut such as a hole, a blind extrusion creates a hole of a specified depth, as shown in Figure 5.31b.

Another way to determine the length of the extrusion is to use existing geometry. One option for specifying an extrusion length is to **extrude to the next surface**. With this option, the extrusion begins at the profile and the protrusion or cut stops when it intersects the next surface encountered, as shown in Figure 5.30c and Figure 5.31c. Another option is to **extrude to a selected surface**, where the protrusion or cut begins at the profile and stops when it intersects a selected surface, which may not necessarily be the first one encountered. See Figure 5.30d and Figure 5.31d. For extruded cuts, there is an option to **extrude through all**. This option creates a cut or protrusion that starts at the profile and extends in the selected direction through all solid features, as shown in Figure 5.31e. A **double-sided extrusion** permits

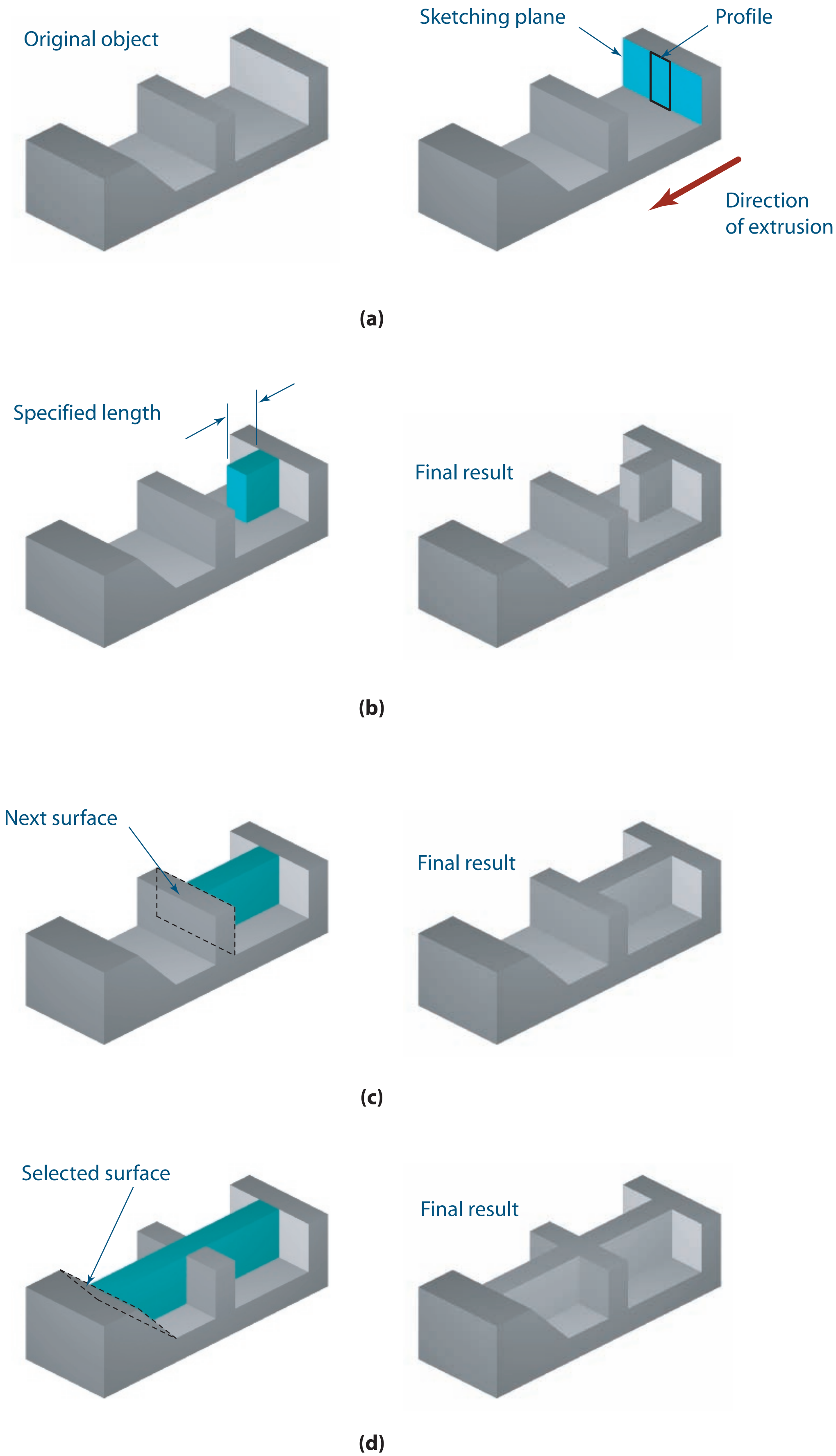


FIGURE 5.30. Different ways to terminate an extruded protrusion from the profile in the sketching plane in (a). Blind extrusion in (b), extrude to next surface in (c), extrude to a selected surface in (d).

© 2009 Cengage Learning®. Courtesy D. K. Lieu.

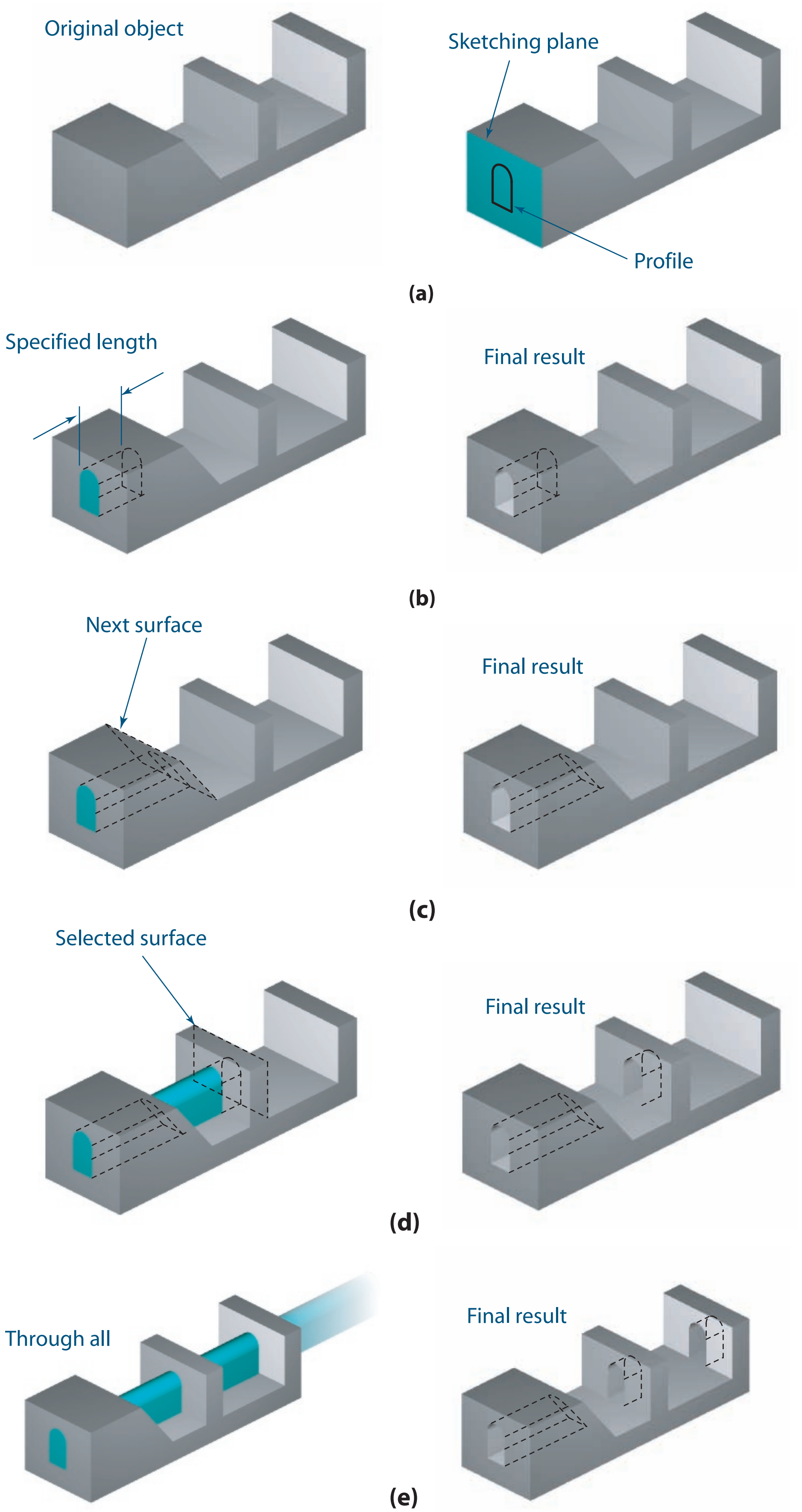
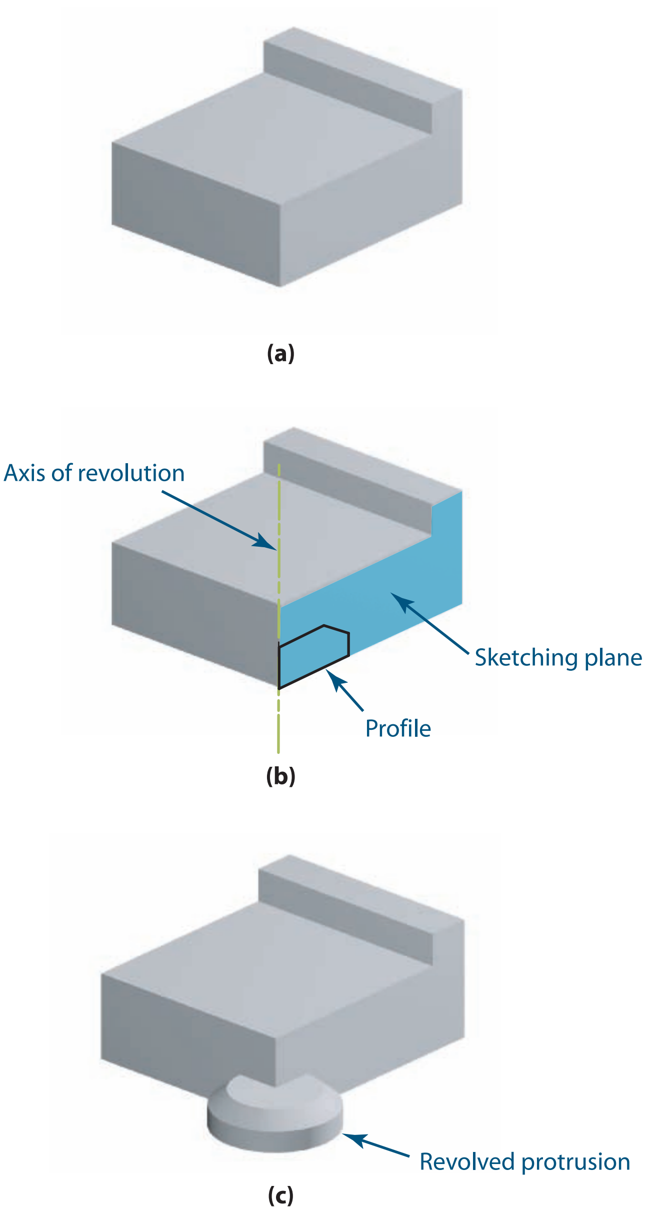


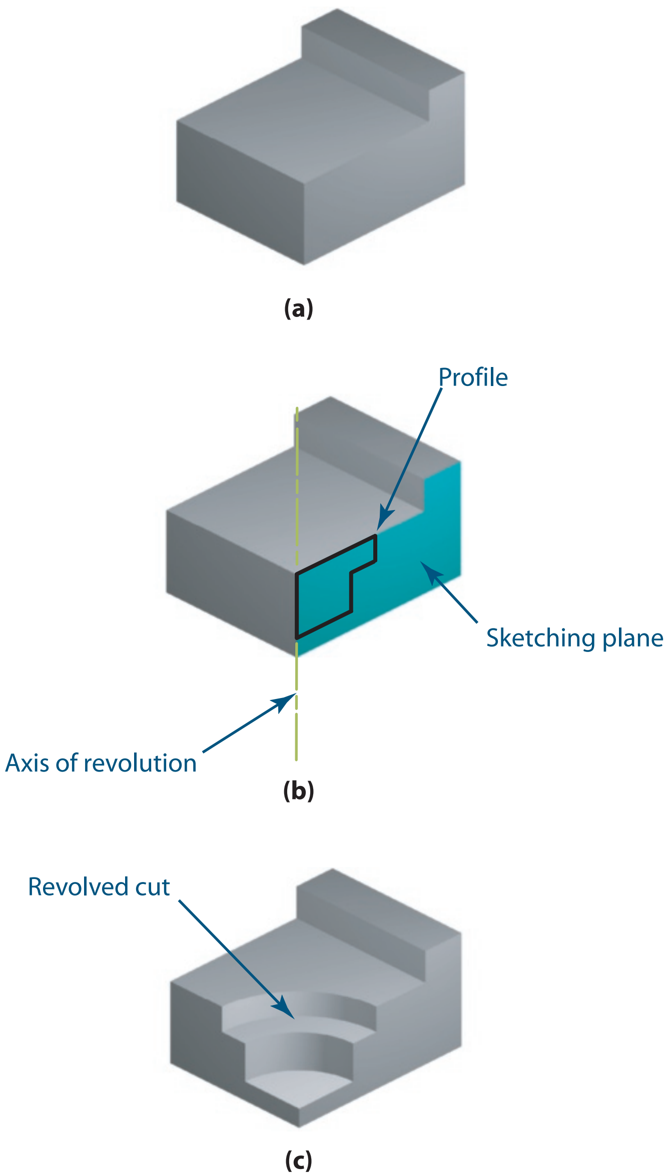
FIGURE 5.31. Different ways to terminate an extruded cut from the profile in the sketching plane in (a). Blind cut in (b); cut to next surface in (c). Cutting to a selected surface (d) and cutting through all (e).

© 2009 Cengage Learning®. Courtesy D. K. Lieu.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.32. The addition of a revolved protrusion to an existing base in (a) by using one of its surfaces as a sketching plane to create a centerline and profile in (b) and revolving it to produce the final result in (c).



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.33. The addition of a revolved cut to an existing base in (a) by using one of its surfaces as a sketching plane to create a centerline and profile in (b) and revolving it to produce the final result in (c).

the protrusion or cut to extend in both directions from a profile. The method of termination in each direction can then be specified independently. Other methods of terminating the extrusion length may be available depending on the specific solid modeling software used. You also can specify the angle of rotation of a revolved protrusion or cut in a similar manner by using a specified angle (blind revolution) or by revolving up to next or selected surfaces.

5.07 Breaking It Down into Features

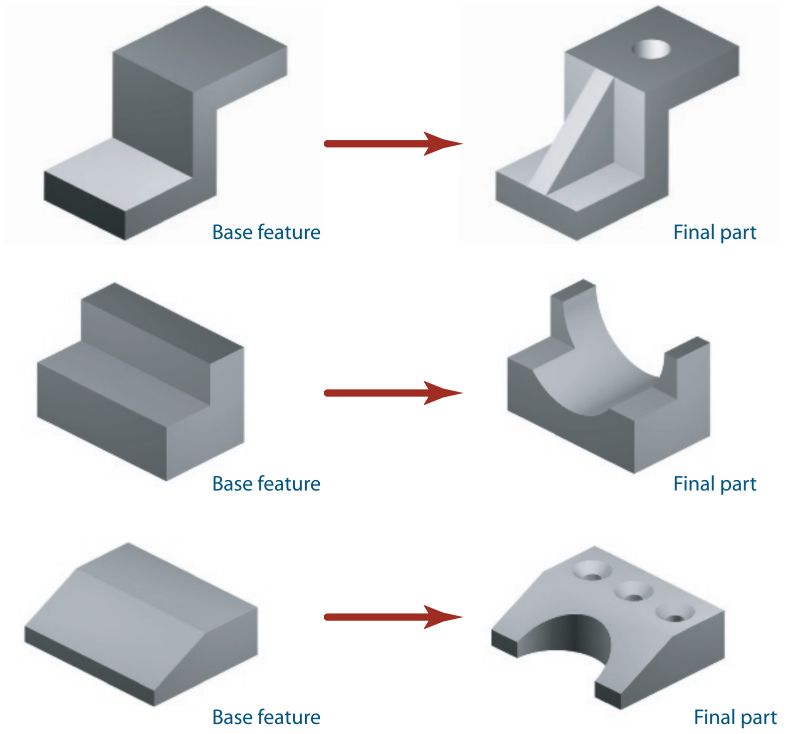
When you build a solid model, you need to decide how to create the various shapes that compose the part. Very few parts can be modeled as a single extrusion or revolution. The various protrusions and cuts on the main body of a model are called **features**. What are features? If you consider your face, you might say that its features are your eyes, nose, lips, and cheeks. It is not much different on a manufactured part; a feature can be any combination of geometric shapes that make up the part and are distinctive in shape, size, or location. Features are characteristic elements of a particular object, things that stand out or make the object unique. Features often have characteristic geometric shapes and specific functions. A simple hole, for example, is a cylindrical cut that is often used as a receptacle for a fastener such as a bolt or screw. A manufactured part may have many different types of features. Since these features are the foundation of contemporary solid modeling systems, you must be able to recognize them.

Engineered parts also have features that are composed of repeated combinations of shapes. Most feature-based modelers have a collection of standard built-in features and may also allow you to define your own features. This can be handy when your products are designed with a particular shape that varies in size, such as gear teeth, airfoils, or turbine blades. The challenge for designers is to identify part features and build solid models that reflect the function of the part and design intent.

5.07.01 The Base Feature

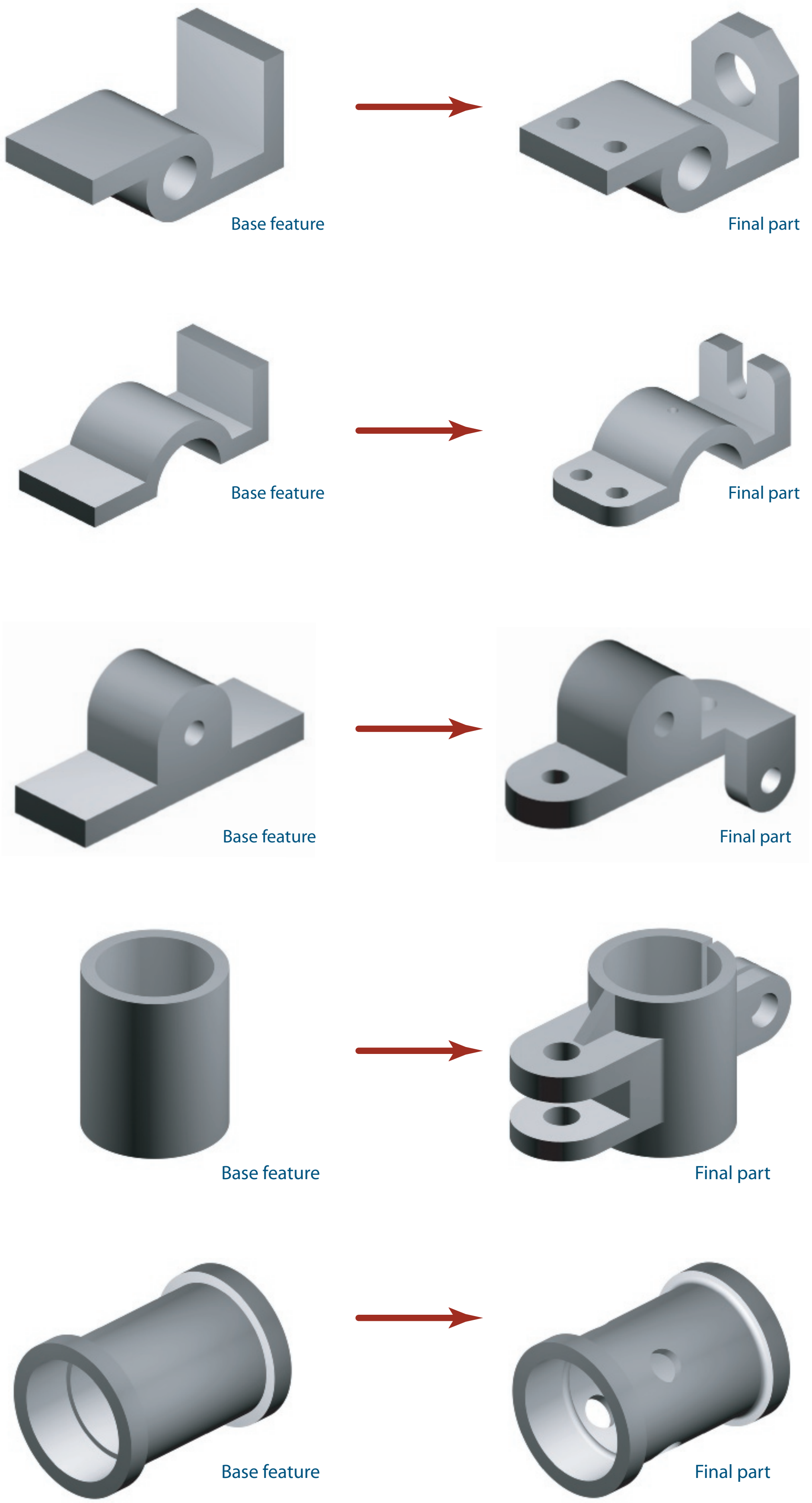
All of your parts will be created from a collection of features, but you need to start your model with a basic shape that represents the general shape of the object. Your first step should be to study the part and identify the shape that you will use as the **base feature**. The base feature should be something that describes the overall shape of the part or something that gives you the greatest amount of functional detail that can be created with a single extrusion or rotation. Figure 5.34 shows several parts with the base features used to create the solid models.

FIGURE 5.34. Parts and their base features.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.34. (CONTINUED)
Parts and their base features.



After the base feature is created, you can modify the shape by adding or subtracting material to it to create form features. A **form feature** is a recognizable region or area on the part geometry that may have a specific function and/or method of manufacture. The geometric components or shapes within the feature usually have some geometric relationships or constraints. Different CAD systems use various names for these features, but you should become familiar with some of the common terms. The following section discusses common feature types.

5.07.02 Chamfers, Rounds, and Fillets

Unless otherwise specified, adjoining surfaces on a virtual part can intersect to form sharp corners and edges, but real parts often have smooth transitions along the edges of these surfaces. The most common edge transitions are **rounds**, **fillets**, and **chamfers**. A round is a smooth radius transition of the external edge created by two intersecting surfaces. A fillet is a smooth transition of the internal edge created by two intersecting surfaces. Geometrically, the rounds and fillets are tangent to both intersecting surfaces. Examples of rounds and fillets are shown in Figure 5.35. Fillets and rounds are specified by the size of their radii and the edge(s) that are rounded.

Chamfers also provide a transition between two intersecting surfaces, but the transition is an angled cut instead of a radius. Examples of chamfers are shown in Figure 5.36. Chamfers can be specified by the distance along each intersecting surface to the original edge or by the distance along one of the original surfaces and the angle made with that surface.

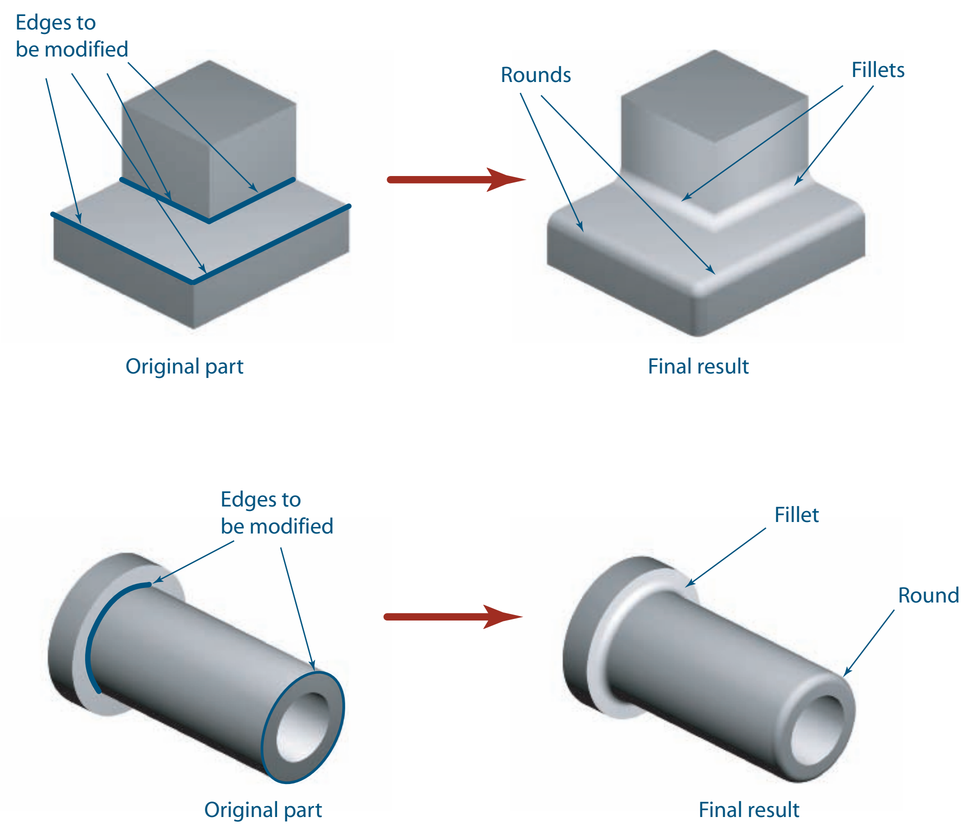


FIGURE 5.35. Examples of rounds and fillets applied to the edges of a part.

© 2009 Cengage Learning®. Courtesy D. K. Lieu.

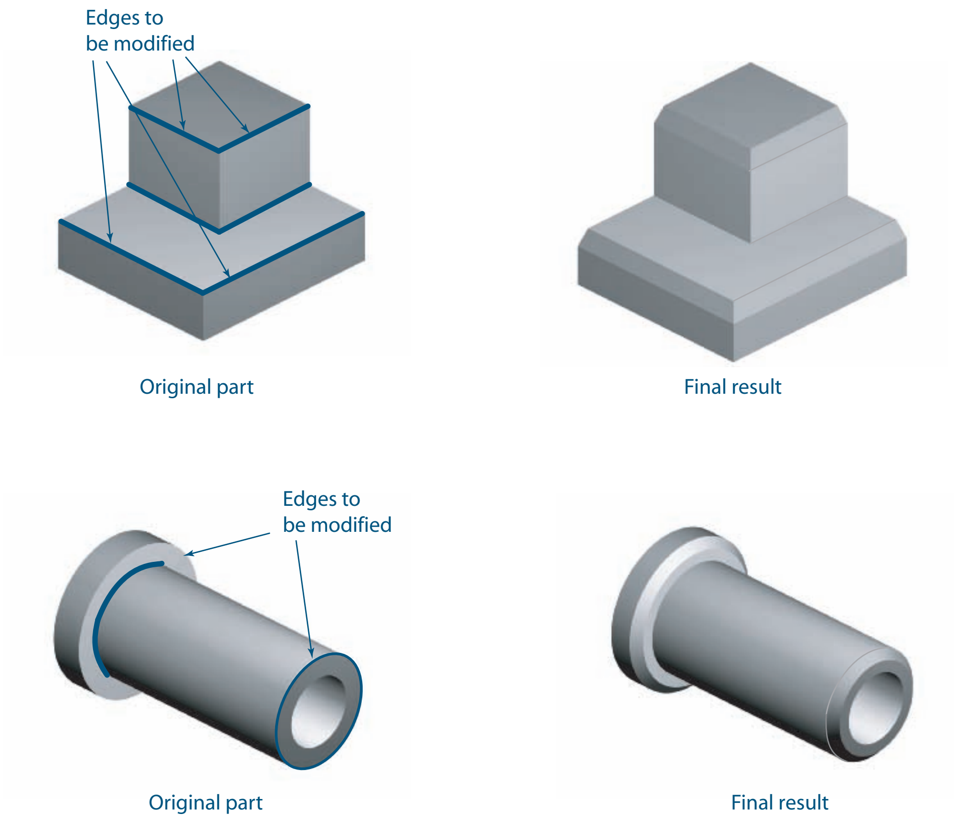


FIGURE 5.36. Examples of chamfering applied to the edges of a part.

© 2009 Cengage Learning®. Courtesy D. K. Lieu.

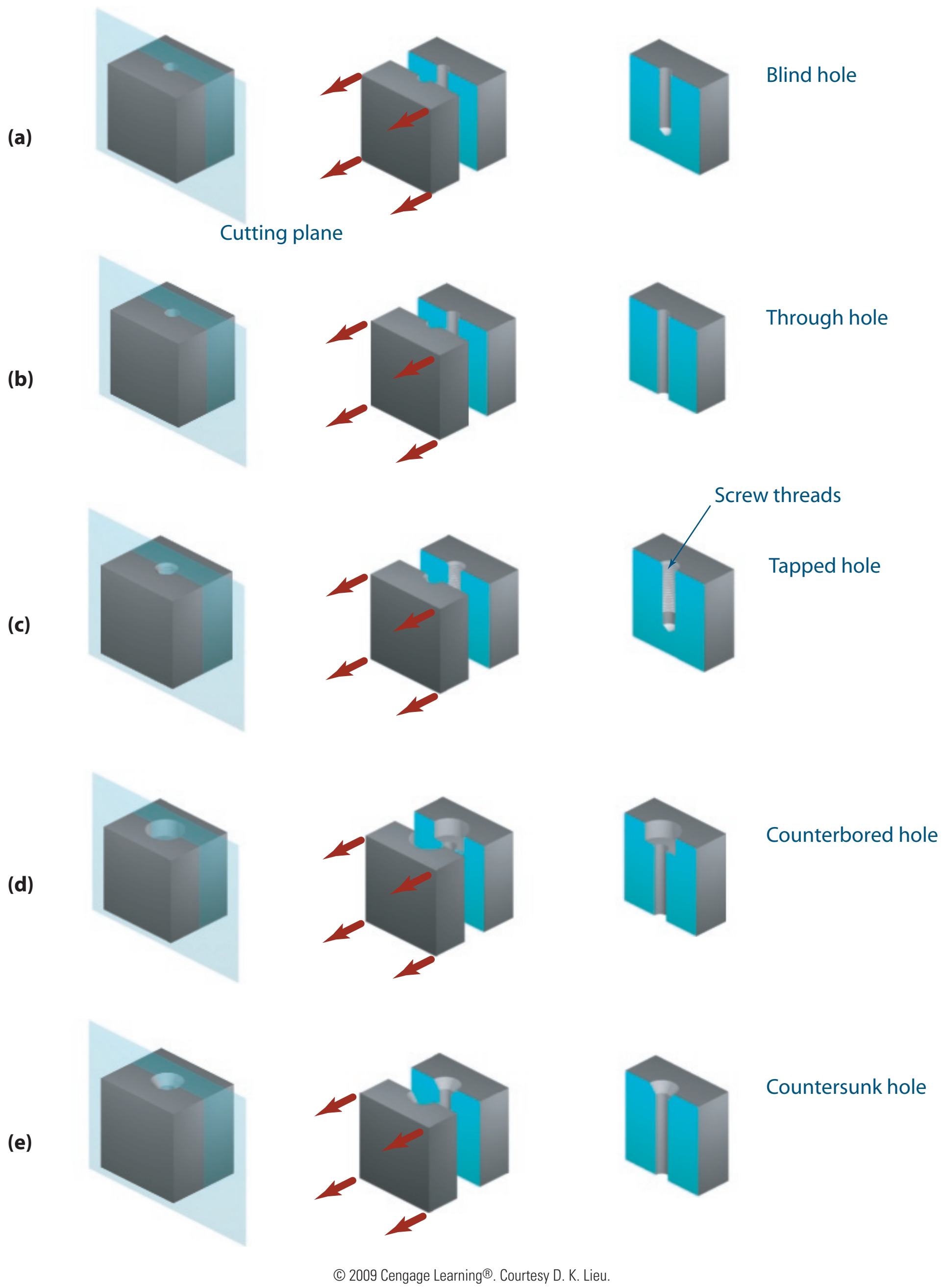
Functionally, on an inside edge, a fillet may be necessary to facilitate fabrication or to reduce stresses at the corner so the part does not break as easily. On an outside edge, rounds and chamfers are usually used to eliminate sharp edges that can be easily damaged or that can cause injury or damage when the part is handled. Rounds, fillets, and chamfers are generally small when compared to the overall size of the associated base or parent feature.

5.07.03 Holes

Holes are ubiquitous in nearly all manufactured parts and, therefore, can be inserted into a model as features by most solid modeling software. Holes are often used with bolts or screws to fasten parts together. Many different types of holes can be used with specific fasteners or can be created using different manufacturing processes. Some special types include holes that are blind, through, tapped, counterbored, or countersunk, as shown in Figure 5.37. Each type of hole has a particular geometry to suit a specific function. You should study the hole types so you recognize them when you model your parts.

Many solid modeling software packages include standard or built-in features to help you with your modeling task. When you use a standard hole feature, the solid modeling software makes certain assumptions about the geometry of a hole so you do not need to specify all of the dimensions and constraints that make up the feature. A countersunk hole, for example, can be made as a revolved cut. What do you need to do to create this feature? You begin by selecting a sketching plane, then create and constrain the sketch and revolve the sketch about a specified axis. Many things can go wrong if you are not careful. There might not be a plane on which to sketch. Your sketch might not have the proper shape, or the axis might not be perpendicular to the desired surface.

FIGURE 5.37. Cross sections of various types of holes to reveal their geometry.



However, a countersunk hole feature can often be created from a standard feature by selecting the location of the axis of the hole on the desired surface, the diameter of the hole, the diameter and angle of the countersink, and the depth of the hole. The shape of the profile, axis of revolution, and angle of revolution are included automatically in the feature definition. No sketching plane is needed. Figure 5.38 illustrates the use of a cut feature compared to the use of a built-in hole feature to create a countersunk hole. In most cases, it is more desirable to create a hole using the built-in hole feature instead of a general purpose cut feature. Besides being a more natural way to place a hole in a model, you avoid potential errors in creating the desired geometry. Furthermore, using a general cut feature does not incorporate the specific geometry and function of a "hole" in the knowledge base of the model, which may be useful in downstream applications such as process planning for manufacturing the hole.

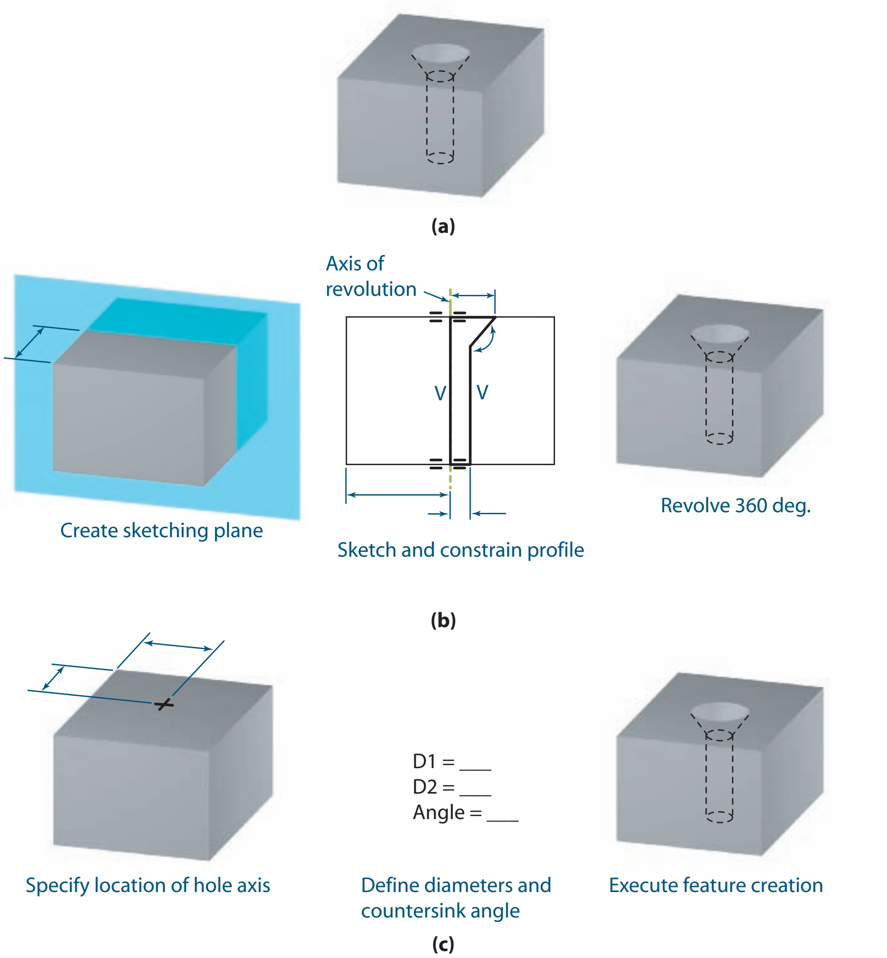


FIGURE 5.38. The countersunk hole shown in (a) can be created by using a general revolved cut, as shown in (b), or by specifying the hole as a built-in feature, as shown in (c).

© 2009 Cengage Learning®. Courtesy D. K. Lieu.

5.07.04 Shells

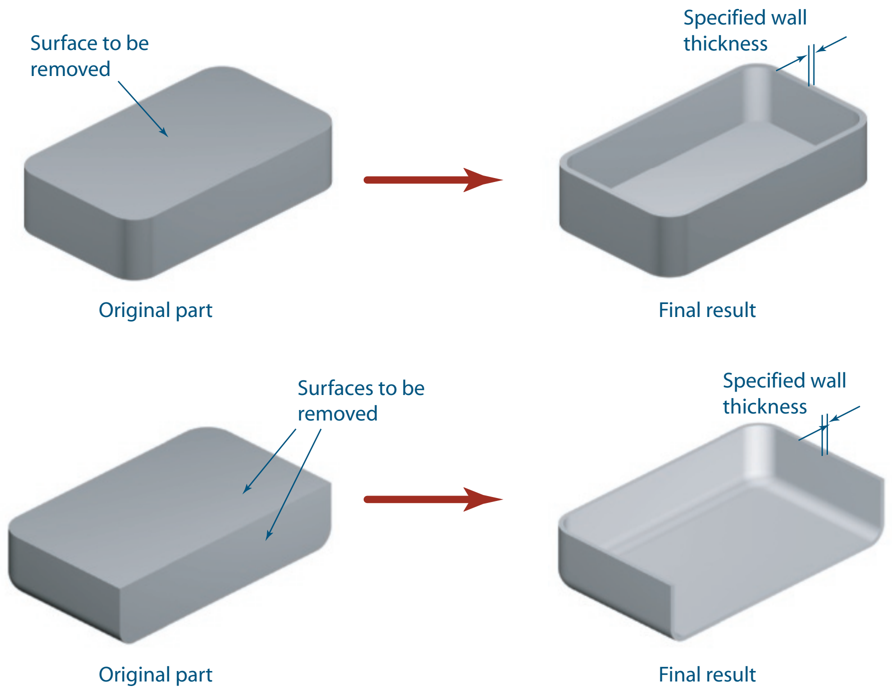
The process of creating a shell, or **shelling**, removes most of the interior volume of a solid model, leaving a relatively thin wall of material that closely conforms to the outer surfaces of the original model. Shelled objects are often used to make cases and containers. For example, a soda bottle is a shell, as are cases for electronic products such as cell phones and video displays. The walls of a shell are generally of constant thickness, and at least one of the surfaces of the original object is removed so the interior of the shell is accessible. Figure 5.39 shows examples of a model that has been shelled.

Shelling is sometimes considered an operation rather than a feature. It is usually performed on the entire model, including all of its features, by selecting the surfaces to be removed and the thickness of the shell wall. Any feature not to be shelled should be added to the model after the shelling operation is complete. The order of feature creation and shelling operations may have a dramatic effect on the shape of the part, as will be shown later in this chapter.

5.07.05 Ribs and Webs

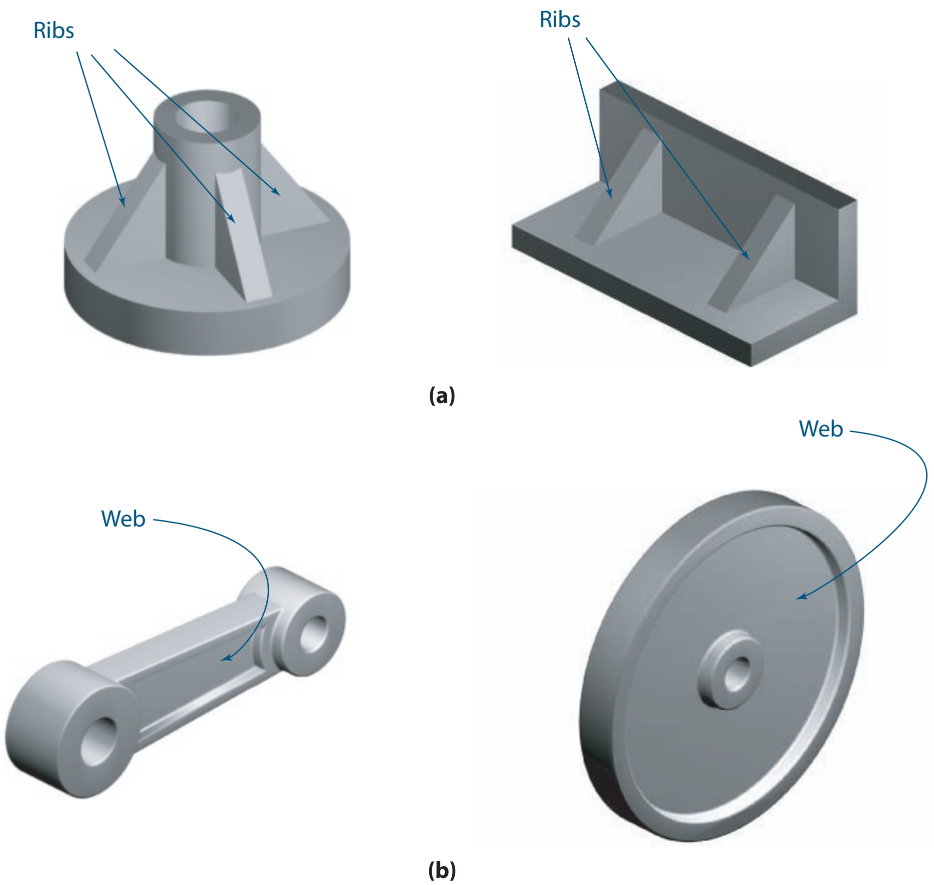
Ribs are small, thin, protrusions of constant thickness that extend predominantly from the surface of a part. Ribs are typically added to provide support or to stiffen a part. Sometimes they are added to improve a part's heat transferability. **Webs** are areas of thin material that connect two or more heavier areas on the part. Examples of ribs and webs are shown in Figure 5.40. These features are usually specified by their flat geometry, thickness, and location.

FIGURE 5.39. Examples of shelling.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.40. Ribs (a) added to parts to reinforce them, and webs (b) connect thicker sections on parts.



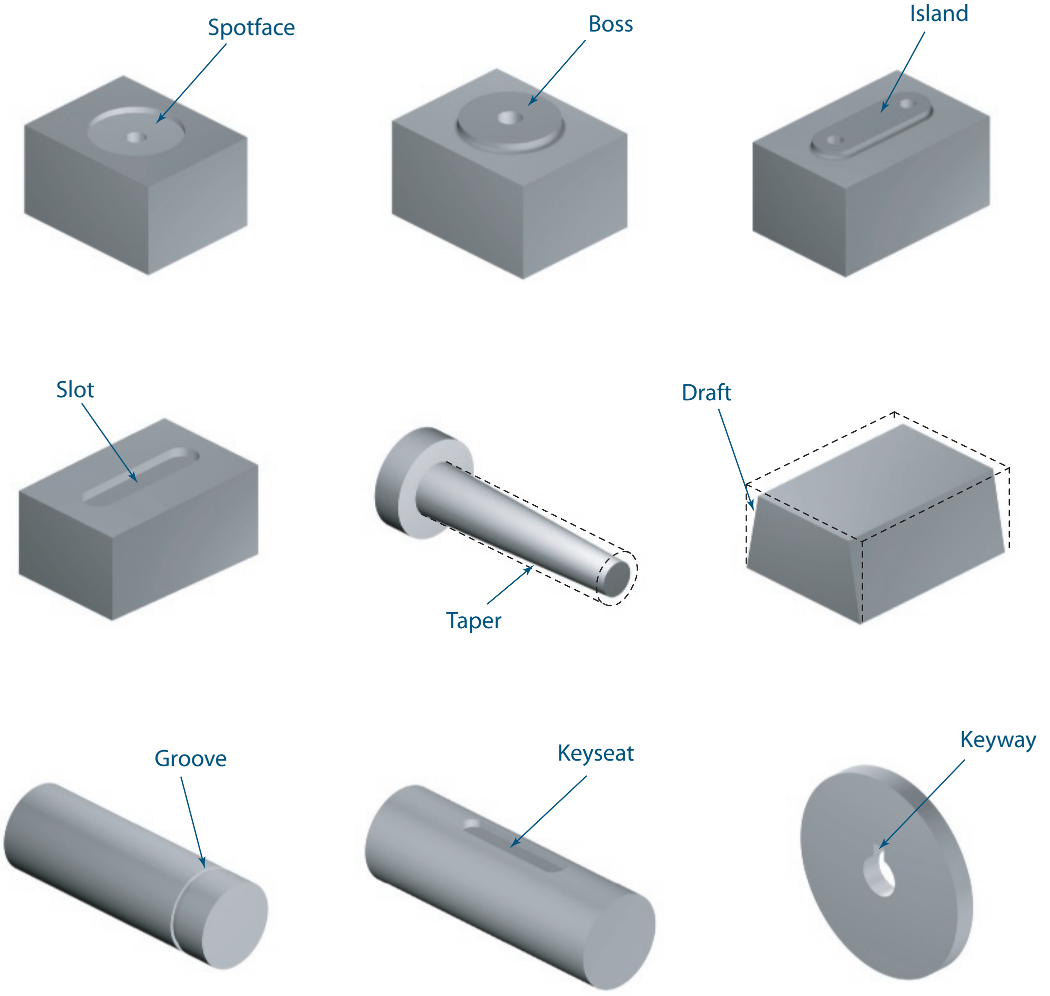
© 2009 Cengage Learning®. Courtesy D. K. Lieu.

5.07.06 Other Feature Types

The features that follow (and that are shown in Figure 5.41) are less commonly found in solid modelers. When available, they should be used as needed. When such features are not available in the solid modeler, the geometric shapes can still be created from sketched profiles as protrusions or cuts. Note that special feature types usually imply a particular shape, function, manufacturing process, or other feature attribute.

- Boss—a slightly raised circular area, usually used to provide a small, flat, clean surface
- Draft—a slight angle in the otherwise straight walls of a part, usually used to facilitate its removal from a mold
- Groove—a long, shallow cut or annulus
- Island—an elongated or irregularly shaped raised area, usually used to provide a flat, clean surface
- Keyseat—an axially oriented slot of finite length on the outside of a shaft
- Keyway—an axially oriented slot that extends the entire length of a hole
- Slot—a straight, long cut with deep vertical walls
- Spot face—a shallow circular depression that has been cut, usually used to provide a small, flat, clean surface
- Taper—a slight angle in the otherwise cylindrical walls of a part, usually used to facilitate its insertion or removal into another part

FIGURE 5.41. Various features with specific functions that may be added to a solid model.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

5.07.07 Cosmetic Features

Parts can be modified by altering their surface characteristics. These characteristics are called **cosmetic features** because they generally modify the appearance of the surface but do not alter the size or shape of the object, just like lipstick or hair coloring. Cosmetic features are necessary to the function of the part and may be included in the model so they can be used in later applications, such as fabrication. Some common cosmetic features include threads and knurls. Since the geometric changes are small and detailed, the cosmetic features usually are not modeled in their exact geometric form in the database of the object, but are included as notes or with a simplified geometric representation. You will learn more about simplified representations on drawings in later chapters. Some cosmetic features are shown in Figure 5.42.

5.07.08 An Understanding of Features and Functions

As a design engineer, you need to become familiar with the different types of features on various parts. Doing so will help you communicate with other engineers as well as imbed more of a part's engineering function into your models. For example, if you look at Figure 5.43, you will notice a rectangular cut on the edge of the hole. This cut is a geometric feature called a keyway. Why is it there? What purpose does it serve? In Figure 5.44, the gear is mounted to a shaft, which also has a rectangular cut.

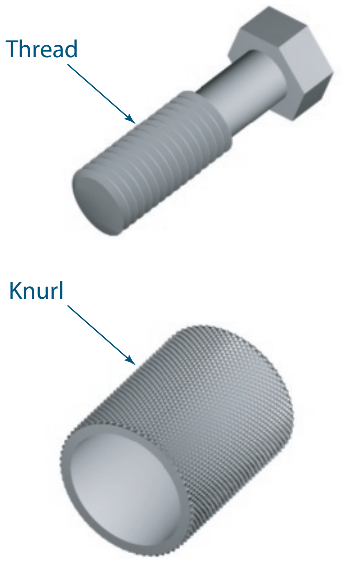


FIGURE 5.42. Some cosmetic features.

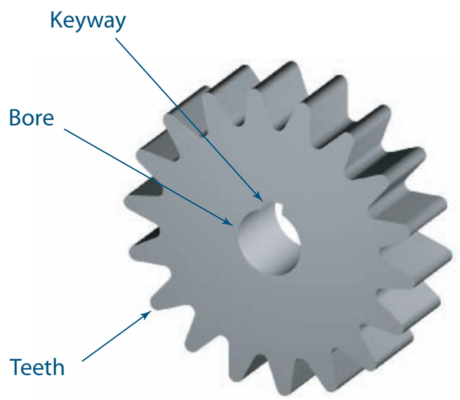


FIGURE 5.43. A gear with teeth, a bore, and a keyway as functional features.

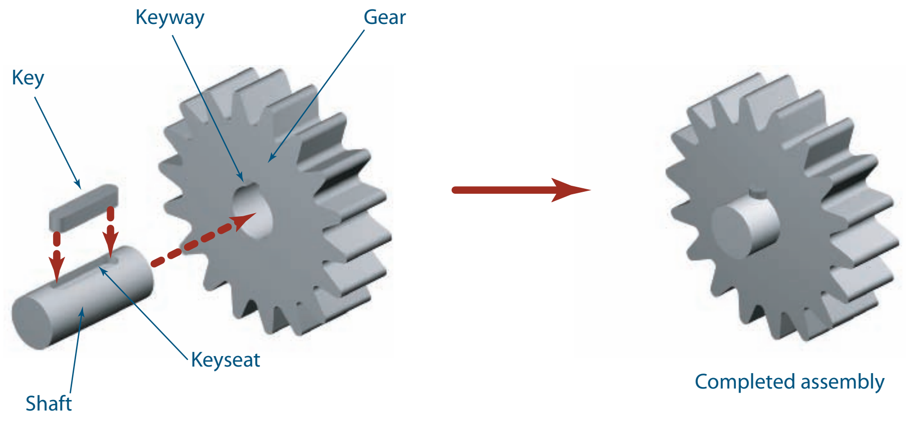


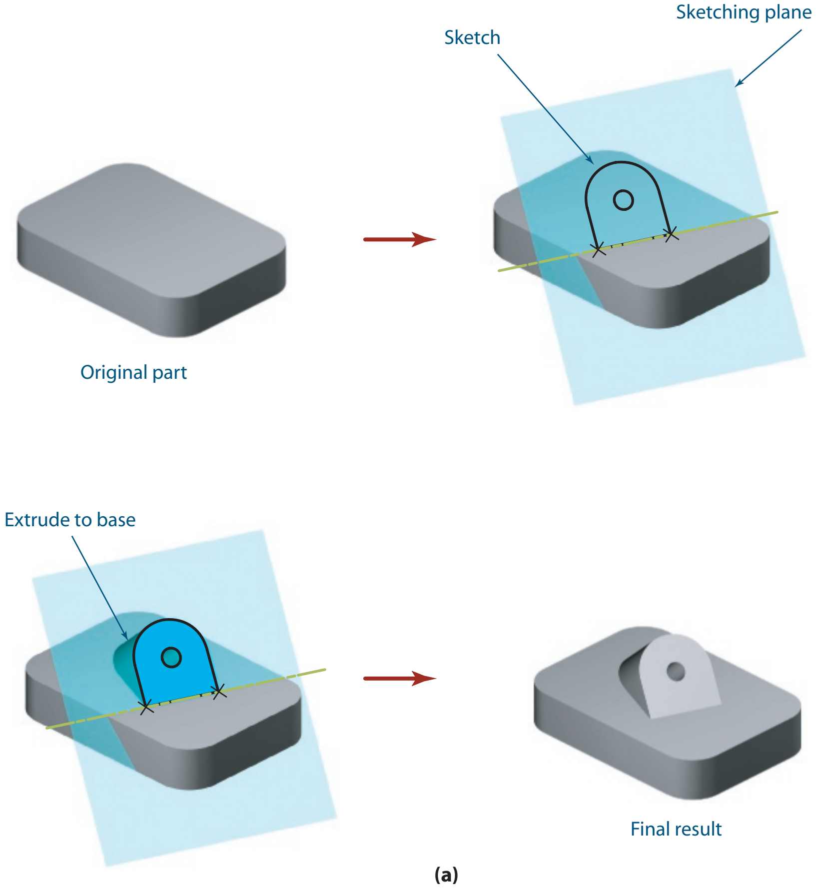
FIGURE 5.44. A gear and shaft assembly. The key functions to transmit torque. The keyseat receives the key in the shaft, and the keyway receives the key in the gear.

A small part called a key is used to line up the shaft and the gear and transmits torque from the shaft to the gear. If you were to create a feature-based solid model of the gear, you could identify the rectangular cut as a keyway feature. If the model parts were to be assembled with assembly modeling software (which is explained in detail in a subsequent chapter), the computer and software would recognize the models as mating parts and orient the gear, key, and shaft automatically.

5.08 More Ways to Create Sophisticated Geometry

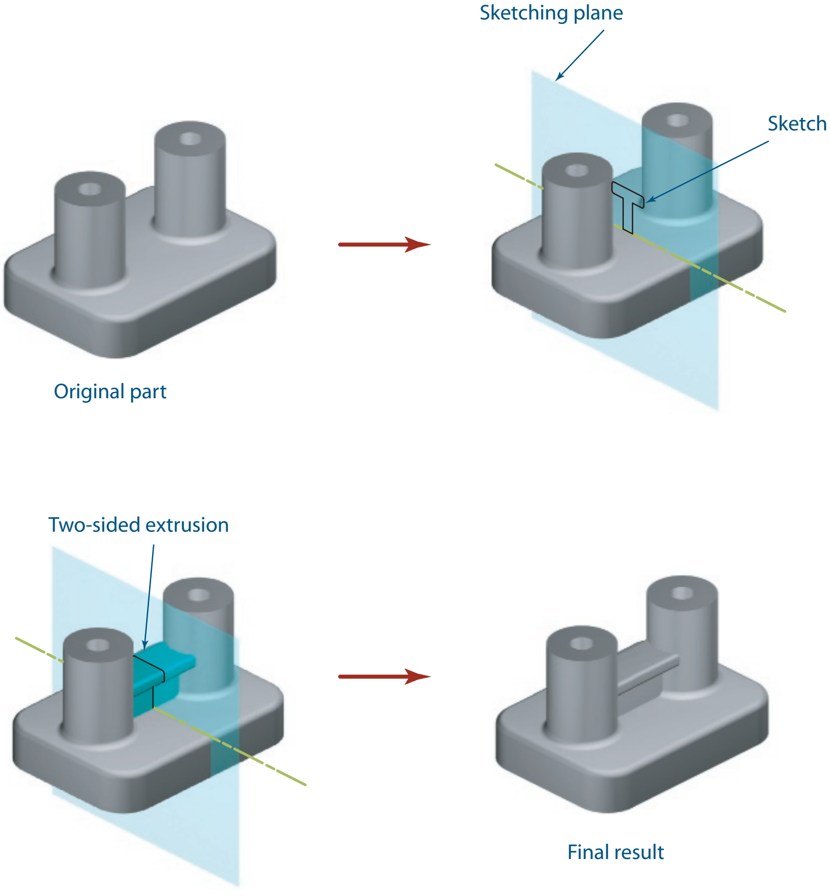
Creating protrusions and cuts by extending the sketch profiles made on either the basic modeling plane or one of the existing surfaces of the model results in a wide variety of possible models. Even more sophisticated models, however, can be created by using reference geometries called datums, which can be added to the model, displayed, and used to create features. Generally, solid modelers offer at least three types of **datum geometries** that can be placed into a model: datum points, datum axes, and datum planes. These datum geometries do not actually exist on the real part (i.e., they cannot be seen or felt) but are used to help locate and define features. Consider, for example, the part shown in Figure 5.45a. The angled protrusion with the hole would be easy to create if an angled sketching plane could be defined as shown. The extrusion could be made to extend from the sketching plane to the surface of the base feature. This feature would be more difficult, although not impossible, to define using extruded protrusions and cuts that extended only from the basic modeling planes or one of the surfaces on the existing model. In Figure 5.45b, the uniquely shaped web would be easy to create if a sketching plane could be placed between the connected features as shown. An extruded protrusion could extend from both sides of the sketching plane to the surfaces of the connected features.

FIGURE 5.45. Using a sketching plane and profile, which are not on an existing surface of the object, to create a protrusion feature (a).



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.45. (CONTINUED) Using a sketching plane and profile, which are not on an existing surface of the object, to create a web feature (b).



(b)

© 2009 Cengage Learning®. Courtesy D. K. Lieu.

The next few sections will describe methods in which the three different types of datums can be defined geometrically and how the datums can be used to create a variety of new types of features. Depending on the specific solid modeling software being used, some of the methods described here for datum definition may or may not be available.

5.08.01 Defining Datum Points

Following are some of the different ways a datum point can be defined and created. The definitions are shown graphically in Figure 5.46.

- At a vertex
- On a planar surface at specified perpendicular distances from two edges
- At the intersection of a line or an axis and a surface that does not contain the line

5.08.02 Defining Datum Axes

Following are some of the different ways a datum axis can be defined and created. The definitions are shown graphically in Figure 5.47.

- Between two points (or vertices)
- Along a linear edge
- At the intersection of two planar surfaces

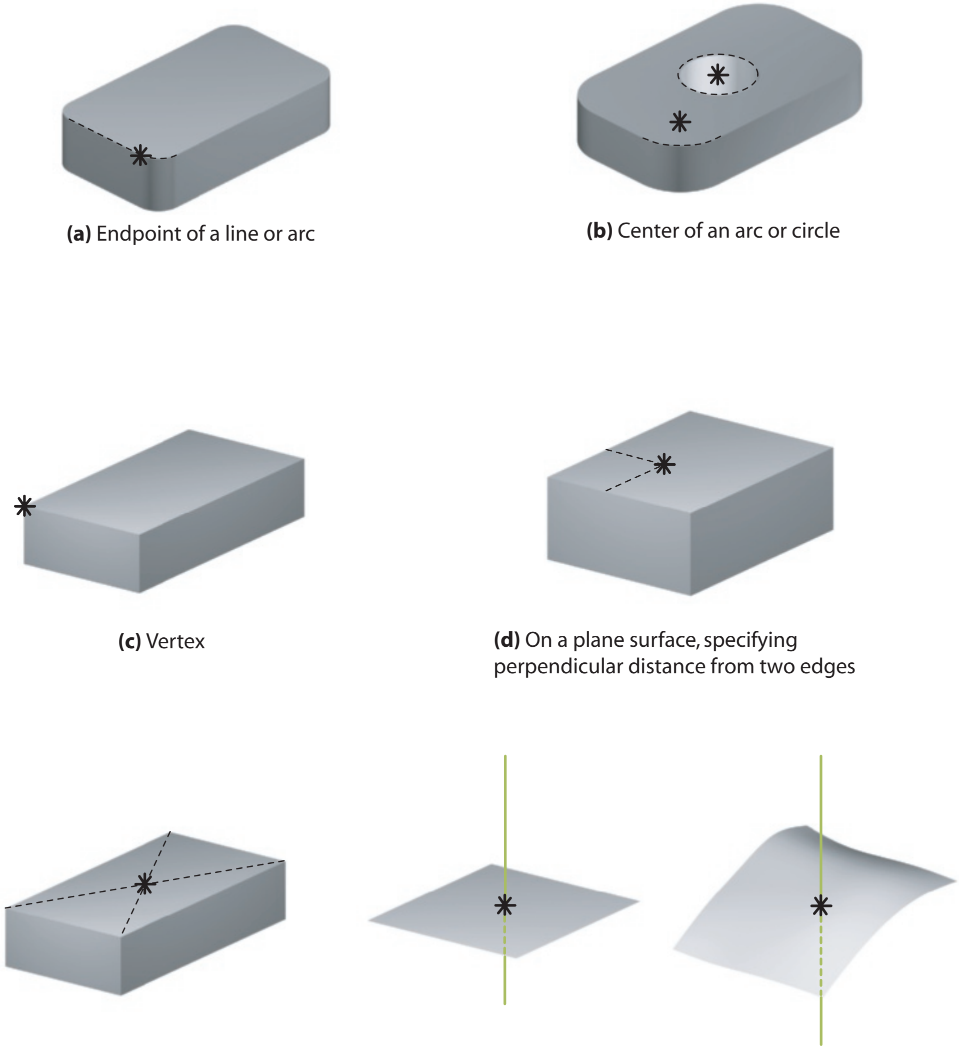


FIGURE 5.46. Various ways to define a datum point.

(a) Endpoint of a line or arc

(b) Center of an arc or circle

(c) Vertex

(d) On a plane surface, specifying perpendicular distance from two edges

(e) Intersection of two lines

(f) Intersection of a line or axis with a plane or surface

© 2009 Cengage Learning®. Courtesy D. K. Lieu.

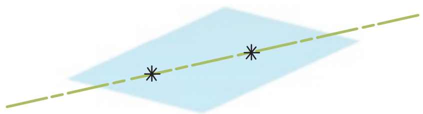
- At the intersection of a cylinder and a plane through its axis
- Along the centerline of a cylinder or cylindrical surface

5.08.03 Defining Datum Planes

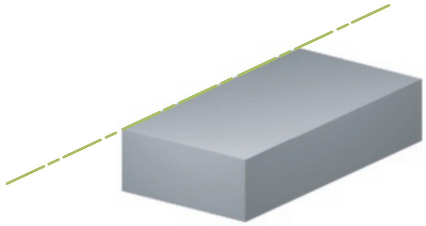
Following are some of the different ways a datum plane can be defined and created. The definitions are shown graphically in Figure 5.48.

- Through three noncolinear points
- Through two intersecting lines
- Through a line and a noncolinear point
- Offset from an existing flat surface at a specified distance
- Through an edge or axis on a flat surface at an angle from that surface
- Tangent to a surface at a point on that surface
- Perpendicular to a flat surface and through a line parallel to that surface
- Perpendicular to a flat or cylindrical surface through a line on that surface
- Tangent to a cylindrical surface at a line on that surface

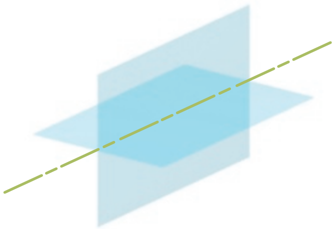
FIGURE 5.47. Various ways to define a datum axis.



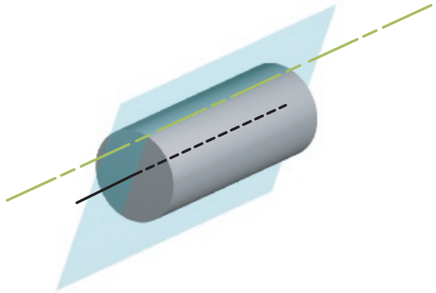
(a) Two points



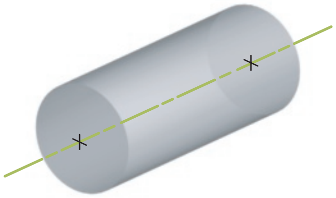
(b) Along an edge



(c) At the intersection of two planar surfaces

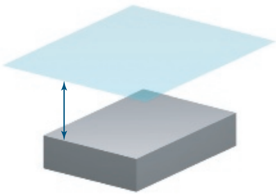


(d) Intersection of a cylinder and a plane through its axis

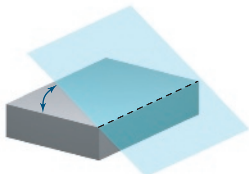


(e) At the centerline of a cylinder or cylindrical surface

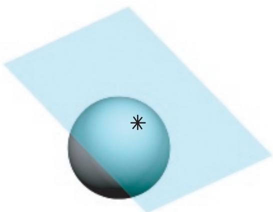
FIGURE 5.48. Various ways to define a datum plane. More ways to define a datum plane.



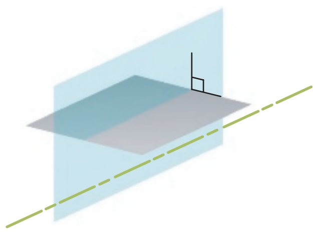
(a) Offset from existing flat surface, and a distance



(b) A flat surface, through an edge or axis on that surface, and an angle from that surface

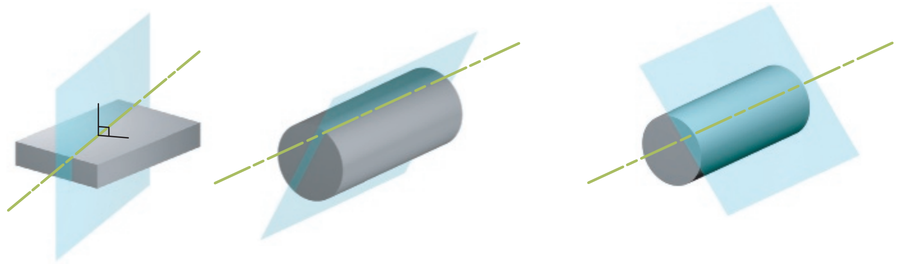


(c) Tangent to a surface at a point on that surface



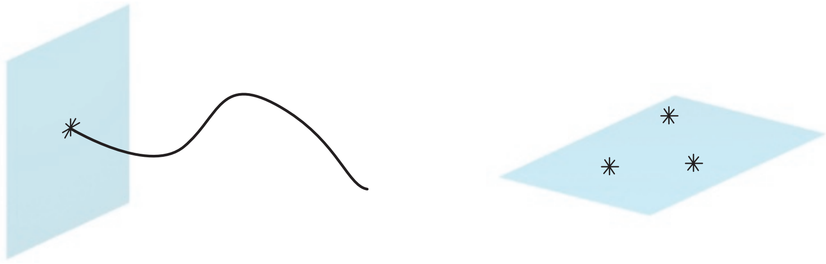
(d) Perpendicular to a flat surface, and through a line parallel to that surface

FIGURE 5.48. (CONTINUED) Various ways to define a datum plane. More ways to define a datum plane.



(e) Perpendicular to a flat or cylindrical surface and through a line on that surface

(f) Tangent to a cylindrical surface at a line on that surface



(g) Perpendicular to a curve at a point on that curve

(h) Three points

(i) Two intersecting lines

(j) A line and a point

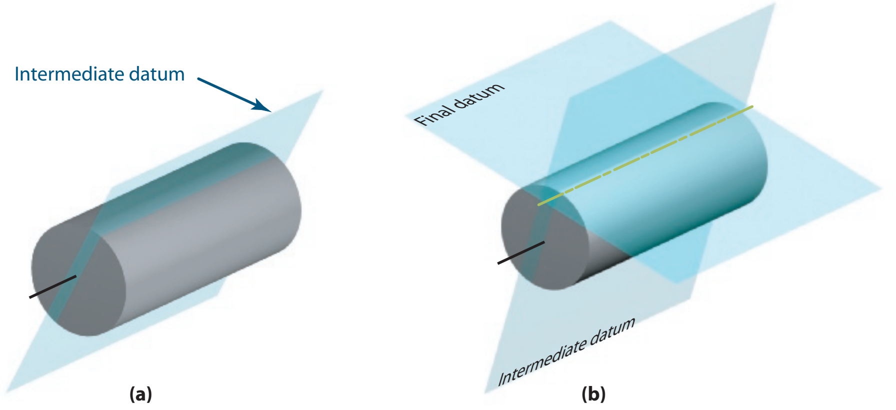
© 2009 Cengage Learning®. Courtesy D. K. Lieu.

5.08.04 Chaining Datums

Series of simply defined datums are often used for creating more complex datums. In the example shown in Figure 5.45a, the angled protrusion was created in this manner: On the top surface of the base extrusion, two datum points were created by defining each of their locations from the edges of the base extrusion. A datum axis was then created using the two datum points as the endpoints of the axis. Finally, the desired datum plane was defined using the top surface of the base extrusion, using the datum axis created in that plane, and specifying the angle that the new datum plane makes with the top surface.

Another example is shown in Figure 5.49, where a datum plane is created to be tangent to the surface of a cylindrical extrusion. An intermediate datum plane is defined by one of the basic planes; the axis of the cylinder, which lies on that basic plane; and the angle the intermediate datum plane makes with the basic plane. The final datum plane is then created to be tangent to the surface of the cylindrical extrusion at its intersection with the intermediate datum plane. A datum plane tangent to the surface of a cylinder is commonly used to create cuts that extend radially into a cylindrical surface, such as holes or slots, and protrusions that extend radially from the cylindrical surface, such as spokes or vanes. Note that with protrusion from a tangent datum plane, the extrusion must be specified to extend in both directions from that datum; otherwise, there will be a gap between the extrusion and the curved surface.

FIGURE 5.49. To create a datum plane that is tangent to a cylindrical surface at a specific location, an intermediate datum plane, shown in (a), can be created through the centerline of the cylinder. The intersection of the intermediate datum with the cylinder creates a datums axis that is used to locate the final datum plane.



5.08.05 Using Arrays (Rectangular and Circular)

One method of creating multiple identical copies of a feature in a model is to create a **feature array**, which is sometimes called a **feature pattern**. A feature array takes one feature, called the **master feature**, and places copies of it on the model at a specified spacing. The copied features are identical to the master feature, and changing the geometry of the master at a later time also changes the geometry of the copies at that time. Including features in this manner can save time and effort in creating the entire model, especially when the features are rather complex. An example of a model with a rectangular array of features is shown in Figure 5.50. An array of rectangular cuts is shown in Figure 5.51. As shown, rectangular arrays can generate copied features

FIGURE 5.50. A rectangular array of protrusions created from a master feature.

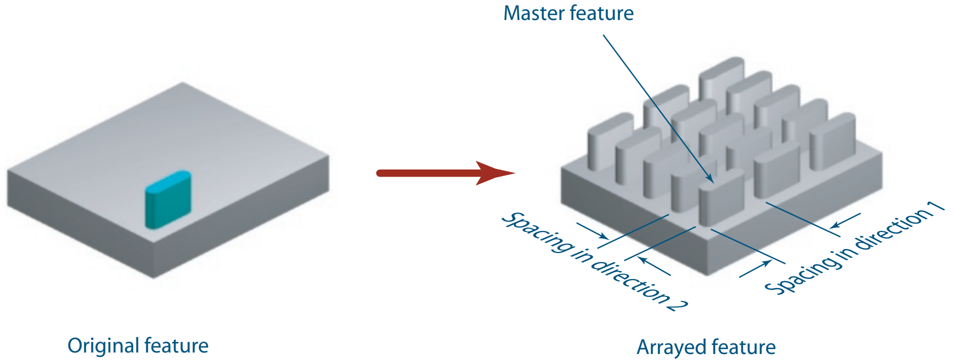
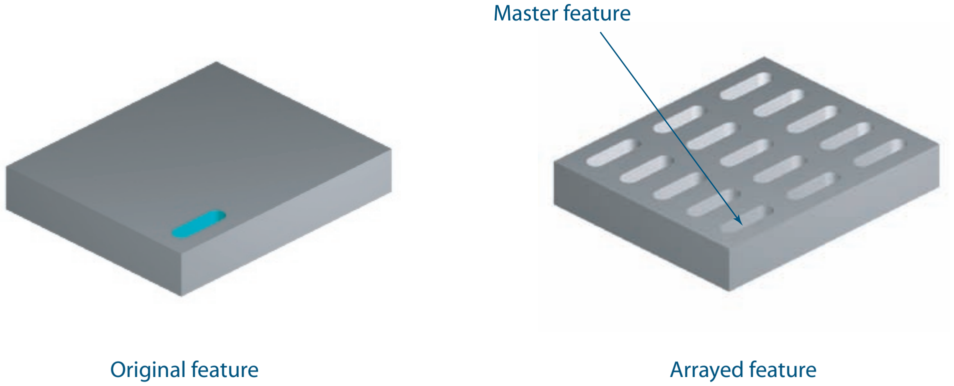


FIGURE 5.51. A rectangular array of cuts created from a master feature.



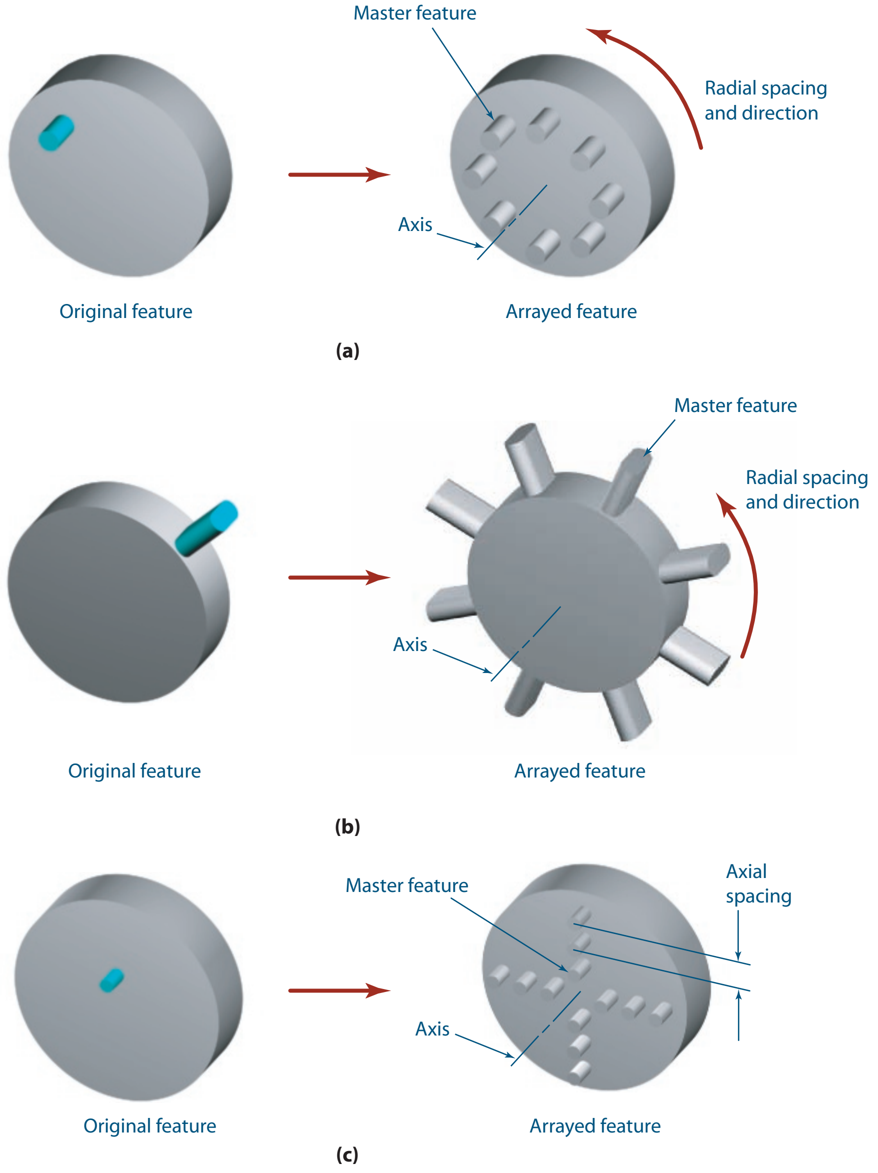
in two directions. These directions must be specified, as well as the spacing of the copied features in each direction. Finally, the number of copies in each direction must be specified. Care must be taken to ensure that there is enough room on the model to accommodate all of the copied features.

Examples of models with radial arrays of protrusions are shown in Figure 5.52. Radial arrays can extend radially or axially. For radial arrays, in addition to the master feature being selected, the axis of revolution for the array must be selected. If such an axis does not already exist on the model, one must be created from an added datum axis. The number of copies, the direction of the array, and the radial and axial spacing of the copies must be specified.

5.08.06 Using Mirrored Features

Another method of creating a feature, when applicable, is to create its mirrored image. To create a **mirrored feature**, you must first identify a mirror plane. You can use an existing plane or define a new datum plane to use as the mirror plane, as shown in

FIGURE 5.52. A circular array of protrusions created from master features in the axial direction (a) and (b), both in the axial and radial direction (c).



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

Figure 5.53. A mirrored duplicate of the master feature can then be created on the model on the opposite side of the mirror plane. Mirrored features can be cuts or protrusions; however, keep in mind that the copied feature will be a mirror image of the master, not an identical copy. Changing the master feature at a later time also will change the mirrored feature correspondingly. As with arrayed features, using mirrored features can save a great deal of time in model creation, especially when the mirrored feature is complex.

5.08.07 Using Blends

Not all models can be created using just extruded or revolved features. One complex feature is a **blend**. Figure 5.54 shows models with blended surfaces. A blend requires at least two profile sketches, and the model is formed by a smooth transition between these profiles. The profiles can be sketched on the basic modeling planes, on surfaces of an existing model, or on datum planes. In the simplest blends, the profiles are sketched on parallel planes. Many software packages require the number of vertices on each of the sketched profiles to be equal. If your profiles do not have the same number of vertices, you will have to divide one or more of the entities to create additional vertices. In some sketching editors, circles include four vertices by default. The vertices in all profiles are usually numbered sequentially, and the software usually tries to match the vertices to create an edge between vertices with the same number, as shown in Figures 5.54a and 5.54b. Rotating the profiles or redefining the vertex numbering can control twisting of the blended transition, as shown in Figure 5.54c. Further control on the model transition usually can be performed by specifying the slope of the transition at each vertex for each shape.

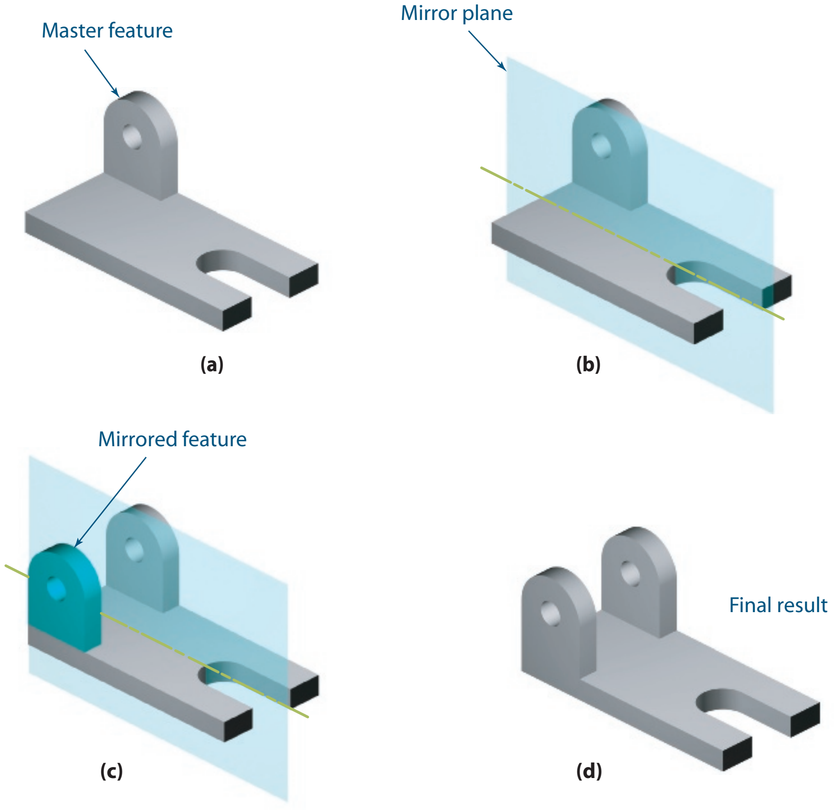


FIGURE 5.53. Creation of a mirrored feature. The master feature in (a) is mirrored by creating a datum plane as a mirror plane (b). A mirror image of the feature is produced on the opposite side of the datum plane in (c), and the final result is shown in (d).

© 2009 Cengage Learning®. Courtesy D. K. Lieu.

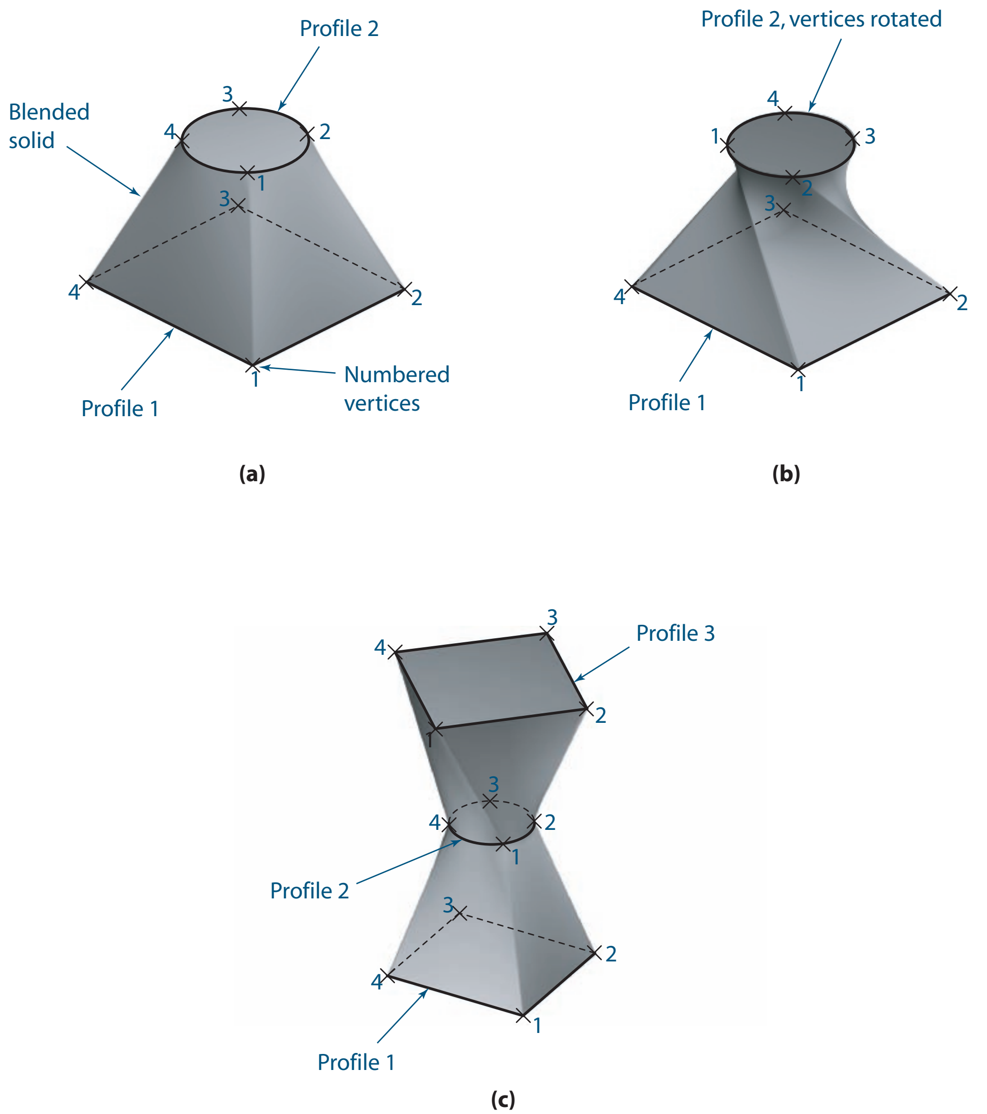


FIGURE 5.54. Blended solids created with two profiles in (a), with the same profiles but rotated vertices in (b), and with three profiles in (c).

5.08.08 Sweeps

Swept features, as with simply extruded or revolved features, are created with a single profile. The difference is that a swept feature does not need to follow a linear or circular path, but can follow a specified curve called a **path** or **trajectory**. The profile is created at an endpoint of the path on a sketching plane that is perpendicular to the path at that endpoint. In sweeping out a solid volume, the profile is imagined to travel along the path. Usually, the profile is constrained to remain perpendicular to the path. A good example of a swept solid is a garden hose. The cross section or profile is a simple circle, but the path can be curved. Figure 5.55a shows the path and profile of a swept feature where the path is open. Figure 5.55b shows a swept feature where the path is closed. Care must be taken in defining the profile and path of a swept solid. Just as you cannot bend a garden hose around a sharp corner without creating a kink, if the path of your sweep contains a sharp corner or a small radius, the feature may fail by trying to create a self-intersecting solid. A special case of a swept solid is a coil

FIGURE 5.55. Features created by sweeps. The sketching plane is perpendicular to the path. The path in (a) is open, and the path in (b) is closed.

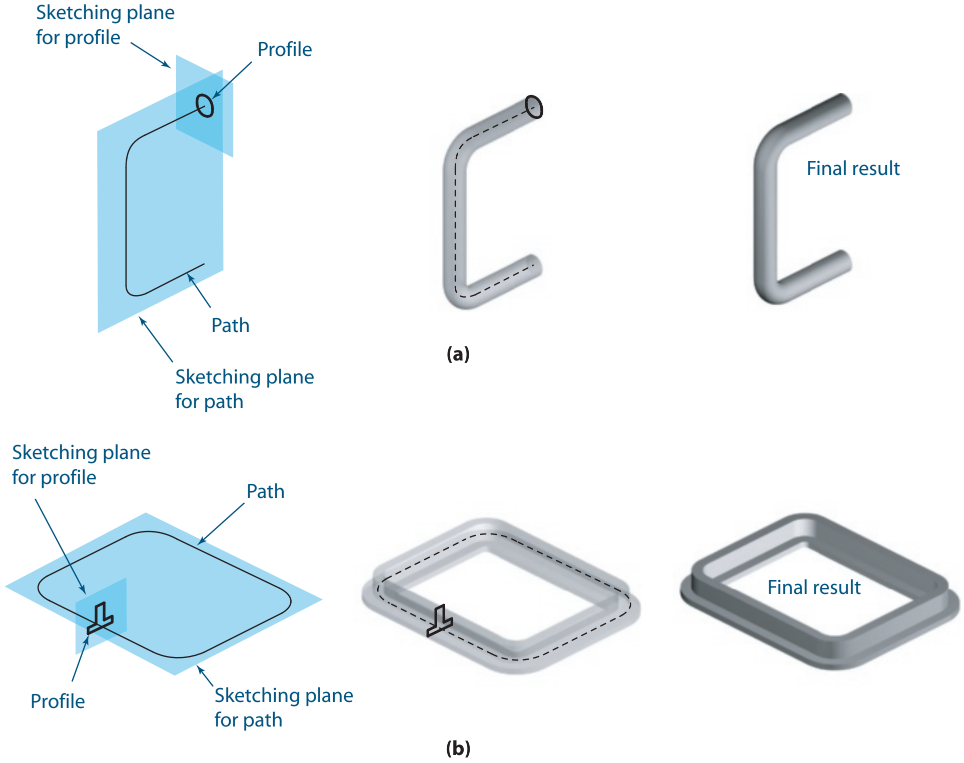
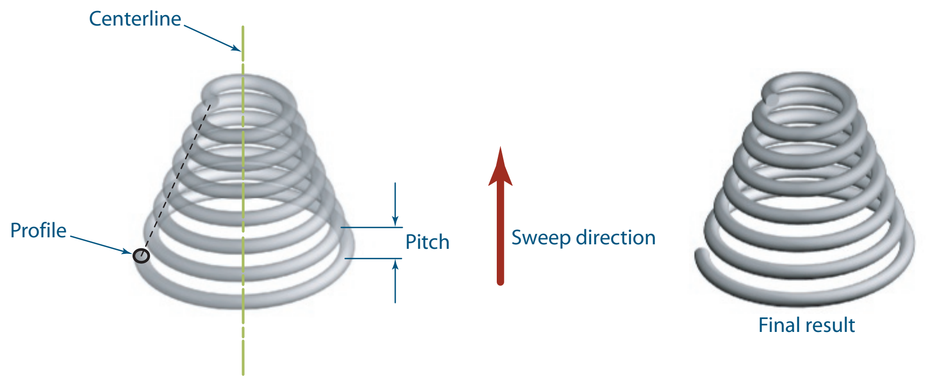


FIGURE 5.56. A tapered spring created by sweeping a circular profile on a helical path.



spring. In this case, the path is a helix, as shown in Figure 5.56. Many solid modelers include a helical sweep as a special feature so you do not have to sketch the helix. In this case, you sketch the profile and specify an axis on the sketching plane. The helix is specified by a pitch dimension, which is the distance between coils, and the direction of the sweep. To avoid self-intersection, the pitch must be larger than the maximum size of the profile in the sweep direction.

5.09 The Model Tree

An extremely useful editing tool included in most solid modeling software is the **model tree**, sometimes called the **feature tree**, **design tree**, or **history tree**. The model tree lists all of the features of a solid model in the order in which they were created, providing a "history" of the sequence of feature creation. Further, any feature in the model tree can be selected individually to allow the designer to edit the feature.

An example of a model tree and its associated solid model are shown in Figure 5.57. Usually, new features are added at the bottom of the model tree. Some software allows the designer to "roll back" the model and insert new features in the middle of the tree. In this case, the model reverts to its appearance just before the insertion point, so any inserted feature cannot have its geometry or location based on features that will be created after it.

The order in which features are created may have a profound effect on the results. In the previous example, a shell feature, which has the effect of hollowing out a part, was performed with the top surface of the part removed from the feature. A hole was then added to the model after the shelling operation. If the hole was added to the block before the shelling operation, the result would be different, as shown in Figure 5.58, because the surface around the hole through the block would have been considered a part of the shell. In most solid modeling software, removing the feature from one location in the model tree and inserting it in a new location changes the order of creation of the feature.

FIGURE 5.57. A typical model tree showing the features of a model in the order in which they were created.

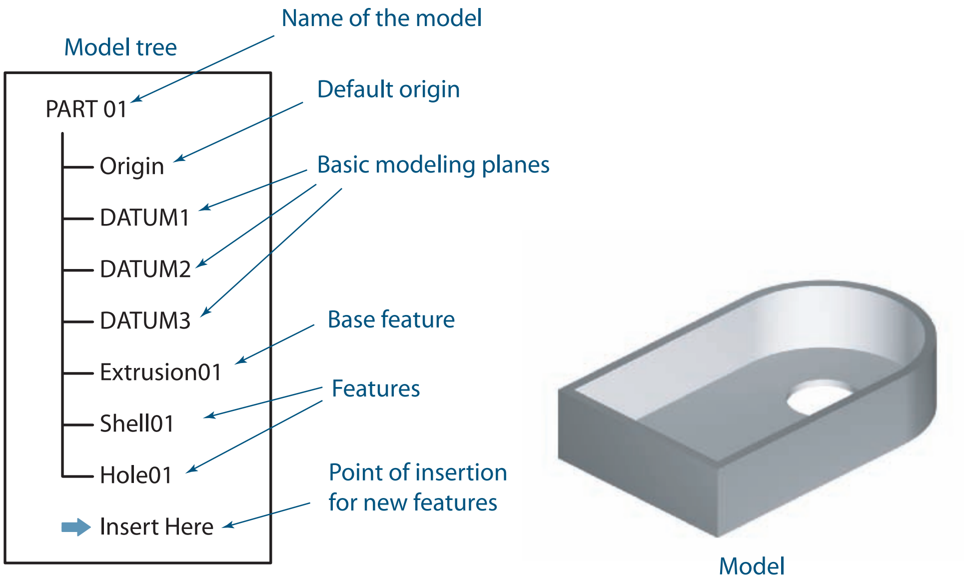
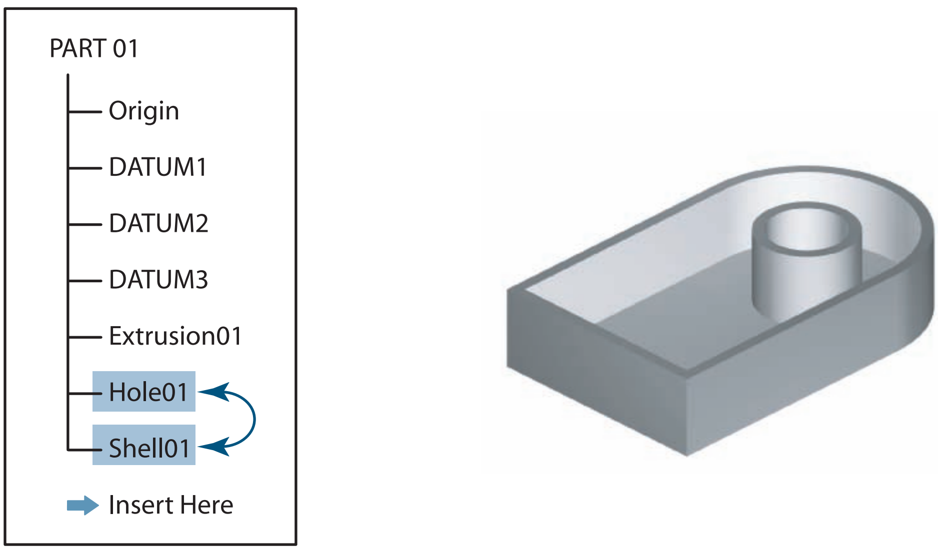


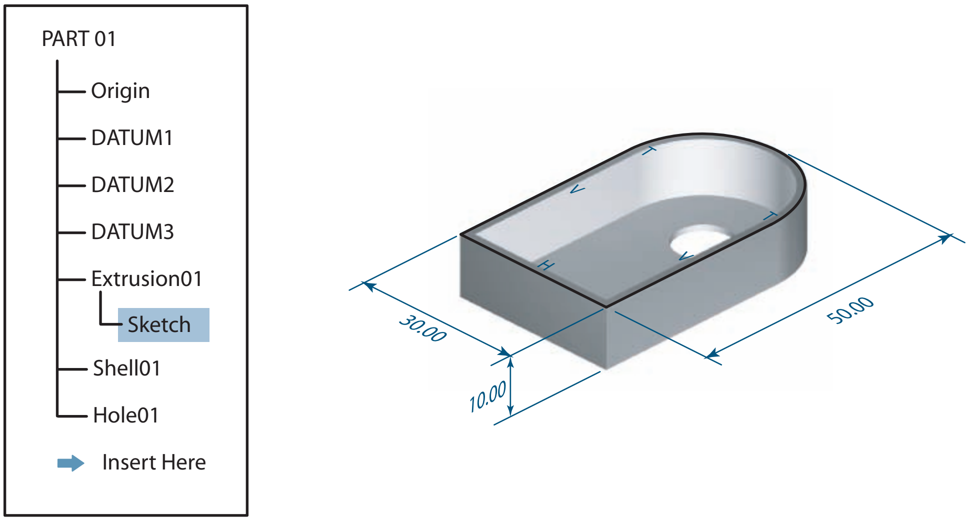
FIGURE 5.58. The result of reversing the order of creating the hole and shell features.



The model tree also provides access to the editing of features. Each feature item on the model tree can be expanded. The base extrusion in the previous example is composed of a fully constrained rectangular sketch profile that has been extruded to a specified length. The feature can be expanded in the model tree, as shown in Figure 5.59, to give access to the profile so it can be selected for editing. The sketch can then be edited by restarting the sketching editor. The dimensional constraints can be changed by selecting and editing their numerical values. Access to the sketching editor and feature parameters may vary with different software, and changes made through the model tree may be one of several different ways to modify your model.

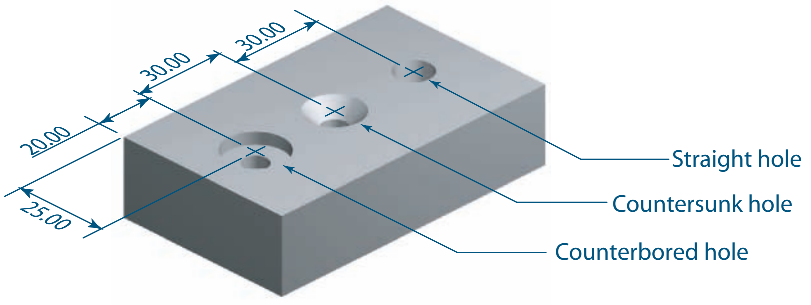
In many models, certain features are dependent upon the existence of other features. For example, consider the features shown in the model in Figure 5.60. The location of the counterbored hole is measured from the edges of the rectangular base. However, the location of the countersunk hole is measured from the location of the counterbored hole and the location of the straight hole is measured from the location of the countersunk hole. Imagine what would happen to the straight hole if the countersunk hole were deleted. There would be no reference for placing the straight hole; therefore, it could not be created. Similarly, if the counterbored hole was deleted, neither the countersunk hole nor the straight hole could be created. This relationship is often referred to as a parent-child relationship. The straight hole is considered the **child feature** of the countersunk hole, and the countersunk hole is considered the child of the counterbored hole. The counterbored hole is considered the **parent feature** of the countersunk hole, and the countersunk hole is considered the parent of the straight hole. Just as you would not be reading this text if your parents did not

FIGURE 5.59. Use of the model tree to access and edit the sketch used to create the base feature (Extrusion01).

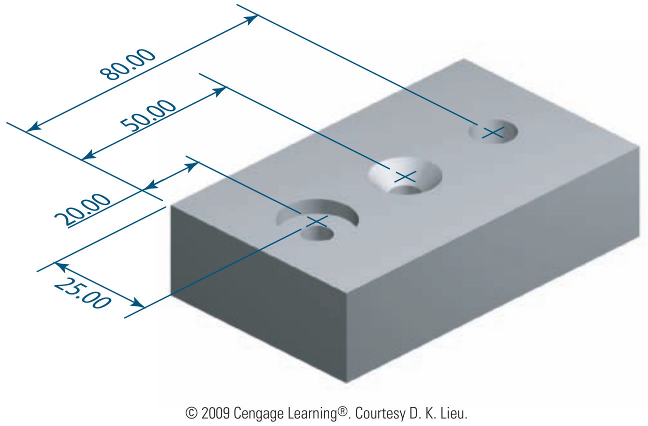


© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.60. The holes in the model show parent-child dependencies. The existence of the straight hole depends on the existence of the countersunk hole, which depends on the existence of the counterbored hole. Elimination of a parent also eliminates its child.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.



© 2009 Cengage Learning®. Courtesy D. K. Lieu.

FIGURE 5.61. This model demonstrates horizontal modeling. Each hole has no parent-child dependencies except to the base feature.

exist, neither can features in a solid model exist without their parent (or grandparent) features. On the model tree, if you try to delete a particular feature, its progeny also will be deleted. However, different software behaves differently; while some software provides specific warnings about the deletion of features, other software does not.

Understanding parent-child relationships in solid models is important if your model needs to be flexible and robust. As a designer, you undoubtedly will want to change the model at some time. You might need to add or delete features to accommodate a new function for the part or reuse the model as the basis of a new design. If you minimize the number of dependencies in the feature tree (like a family tree), it will be easier to make changes to your model. When it is likely that some features will be deleted or suppressed in a future modification of the part, those features should not be used as parents for other features that must remain present. The most extreme example of this strategy is called **horizontal modeling**, where the feature tree is completely flat; that is, there are no parent features except the base feature. This type of modeling strategy was patented by Delphi and has been used successfully by many companies. In Figure 5.61, the locations of three holes have been redefined so they are measured from the edge of the rectangular base instead of relative to one another. The base then becomes the parent to all three holes, and deleting any one of the holes does not affect the others.

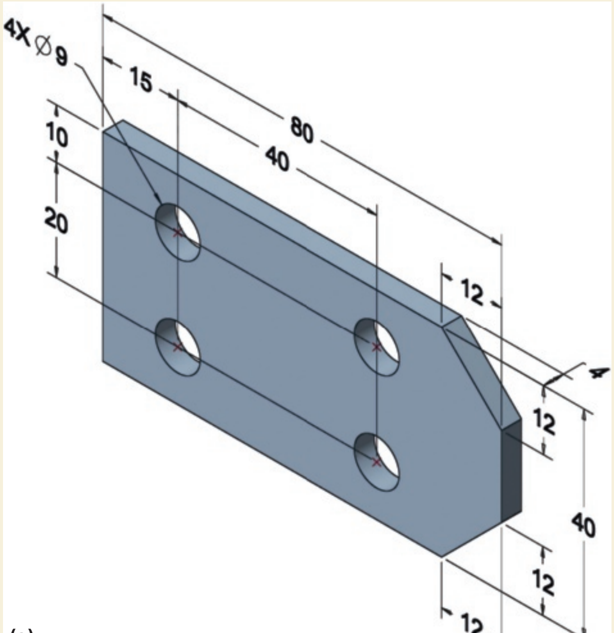
5.10 Families of Parts

Groups of engineered parts often have very similar geometry. An everyday example is bolts and screws. A group of bolts may have the same head and thread geometries, but differ in their available length. Another example is the family of support brackets shown in Figure 5.62. Each bracket has a rough L-shaped base feature, holes, and a support rib (except for Version 3). Only the size and number of holes are different for each version.

When a group of parts is similar, it is possible to represent the entire group with a **family model**, with different versions of that model selected to specify particular parts. Such a model includes a **master model**, which has all of the features that are in any of the members of the group, and a **design table**, which lists all of the versions of that model and the dimensional constraints or features that may change in any of its versions. The attributes that may change are sometimes called **parameters**. The first step in building a family model is to identify all of the features and parameters that can be varied in the members of the family. In addition to a numerical value, every dimensional constraint in a model has a unique **dimension name**, which can be shown by selecting the appropriate display option. In Figure 5.63, all of the dimensional constraints have been changed to show their dimension names and the features have been identified by the feature names that appear on the design tree.

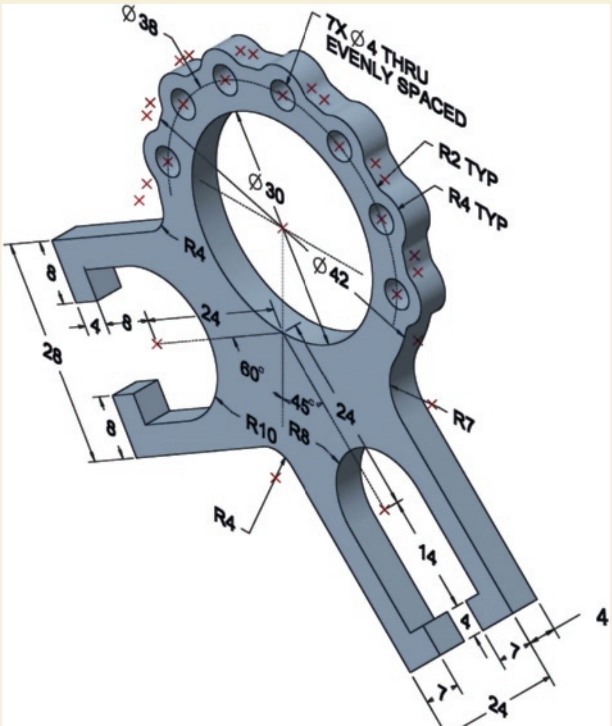
5.15 PROBLEMS

1. Using a single extrusion, create each the following object with a single closed-loop profile using the 2-D drawing capabilities of your solid modeling software. Define the geometry and sizes precisely as shown, using the necessary geometric constraints. Do not over- or underconstrain the profiles.



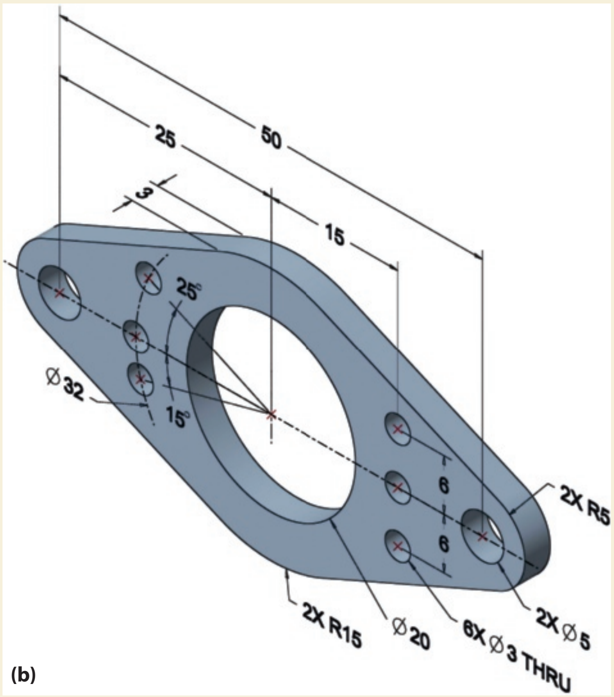
(a)

© 2017 Cengage Learning®. Courtesy of D. K. Lieu.



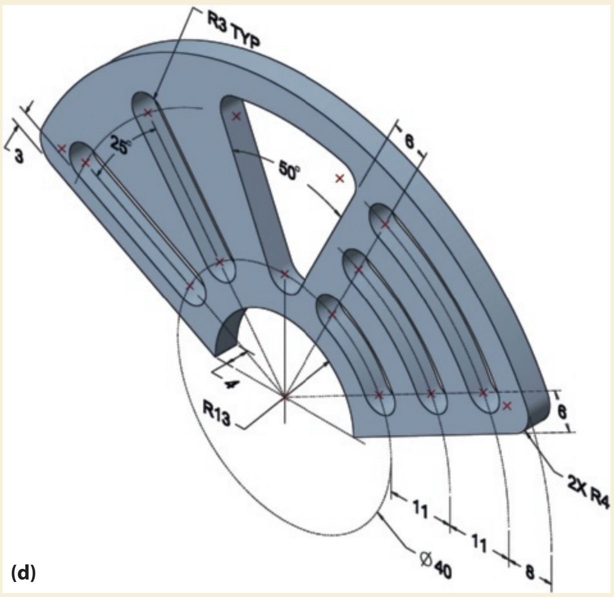
(c)

© 2017 Cengage Learning®. Courtesy of D. K. Lieu.



(b)

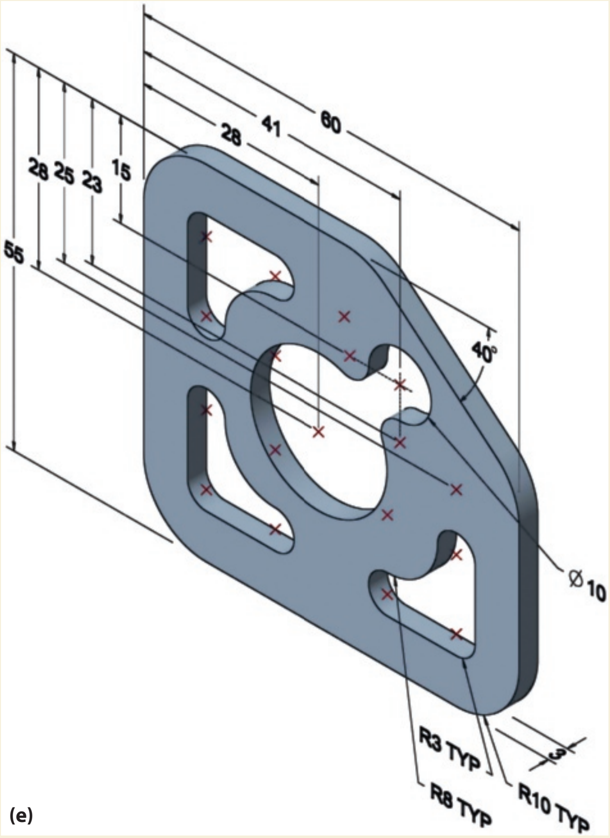
© 2017 Cengage Learning®. Courtesy of D. K. Lieu.



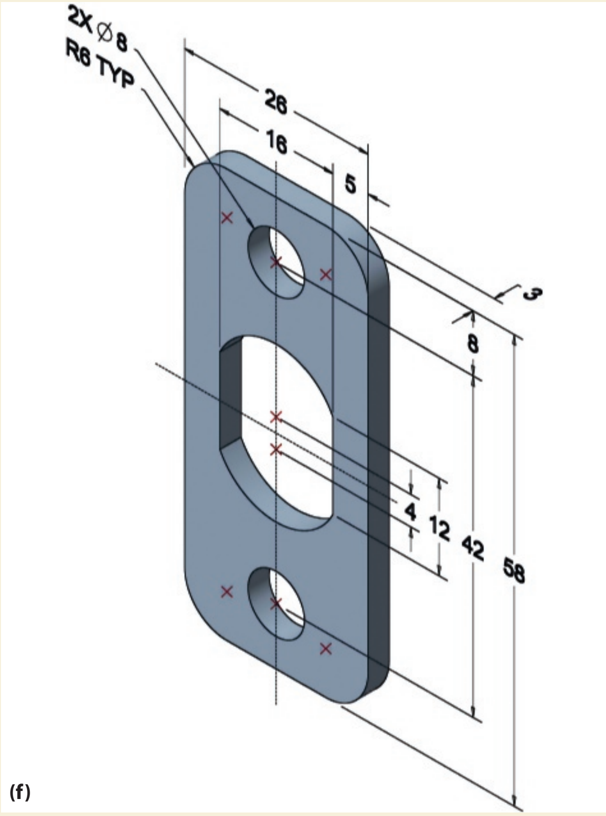
(d)

© 2017 Cengage Learning®. Courtesy of D. K. Lieu.

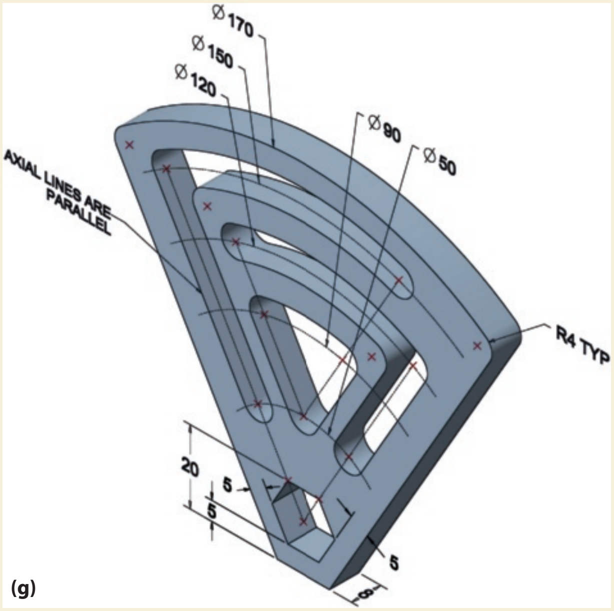
5.15 PROBLEMS (CONTINUED)



© 2017 Cengage Learning®. Courtesy of D. K. Lieu.



© 2017 Cengage Learning®. Courtesy of D. K. Lieu.

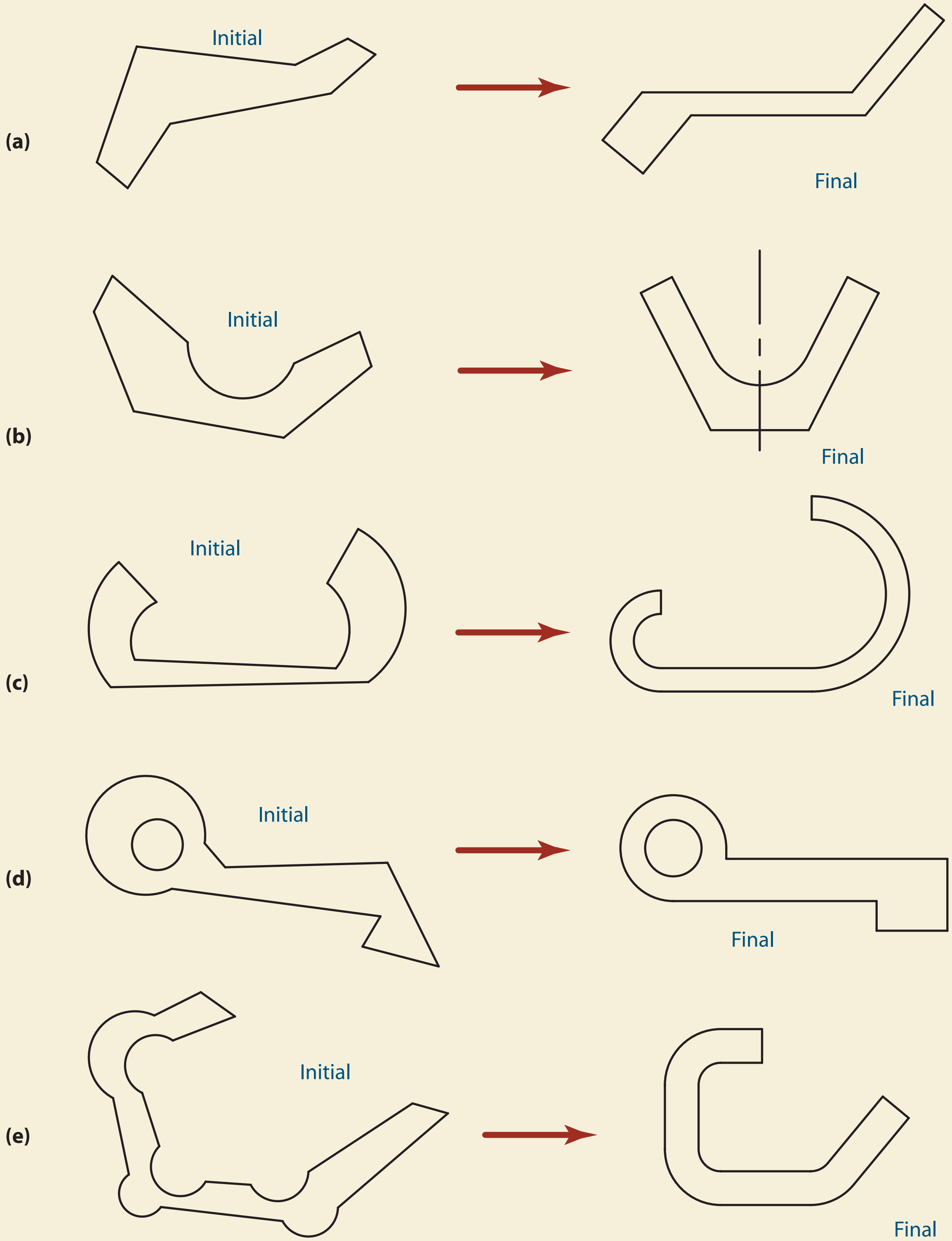


© 2017 Cengage Learning®. Courtesy of D. K. Lieu.

FIGURE P5.1.

5.15 PROBLEMS (CONTINUED)

2. Study the following closed-loop profiles for which geometric constraints have not been added. Number each segment of the profiles and specify the necessary geometric constraints on each segment to create the final profile. Do not over- or underconstrain the profiles.

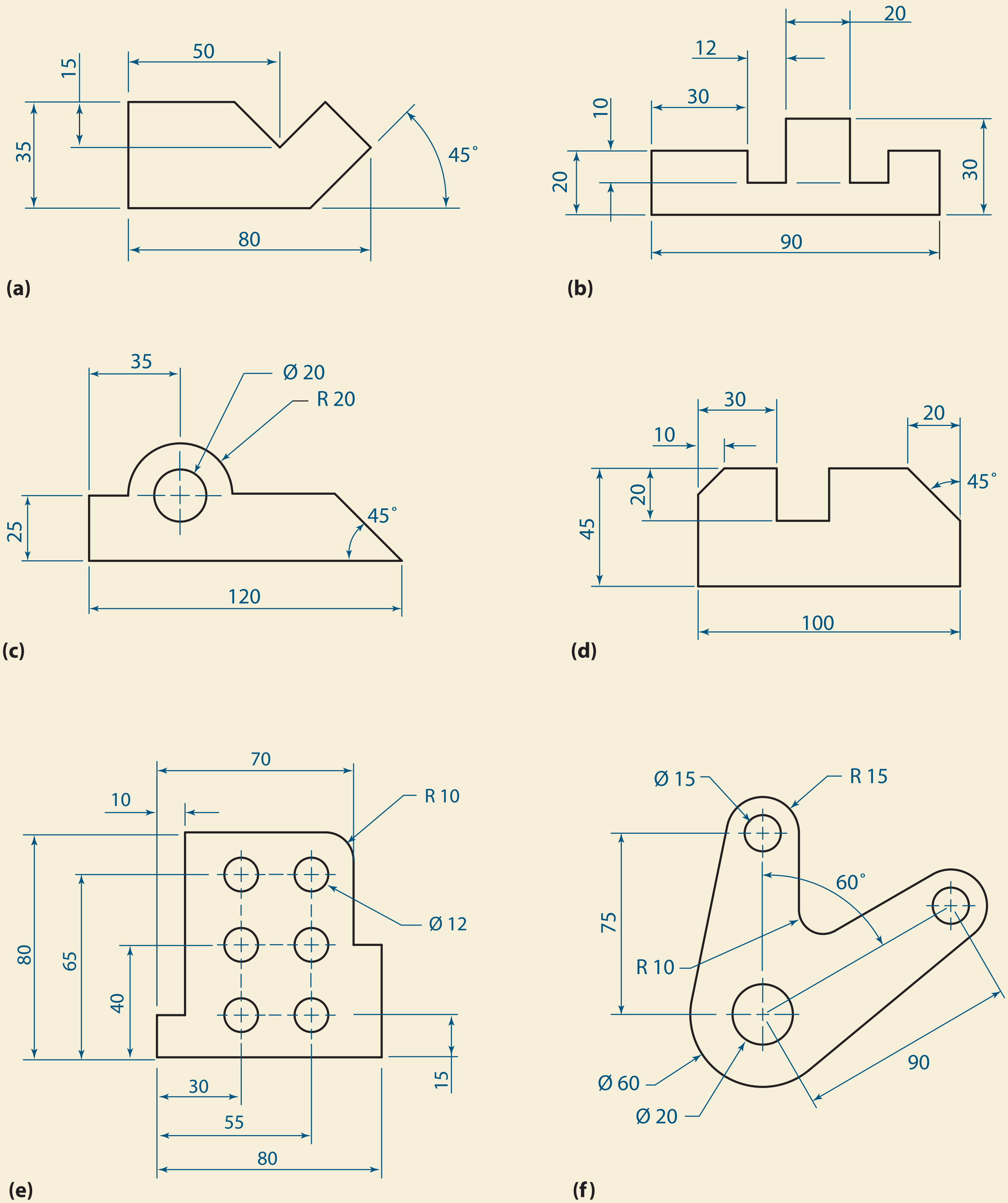


© 2009 Cengage Learning®. Courtesy of D. K. Lieu.

FIGURE P5.2.

5.15 PROBLEMS (CONTINUED)

3. At first glance, these profiles may appear to be missing key dimensions. However, they are fully constrained by the addition of geometric constraints. Number each segment of the profiles. What were the geometric constraints used for each segment?

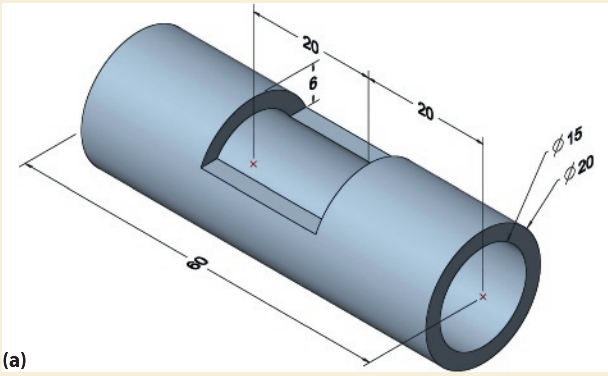


© 2009 Cengage Learning®. Courtesy of D. K. Lieu.

FIGURE P5.3.

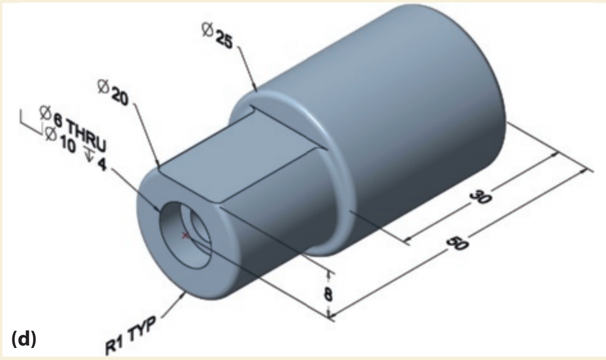
5.15 PROBLEMS (CONTINUED)

4. Create solid models of the following parts in your CAD system. Identify what you consider to be the base geometry for each part. Are any (child) features dependent upon the existence of other (parent) features? If so, specify the hierarchy.



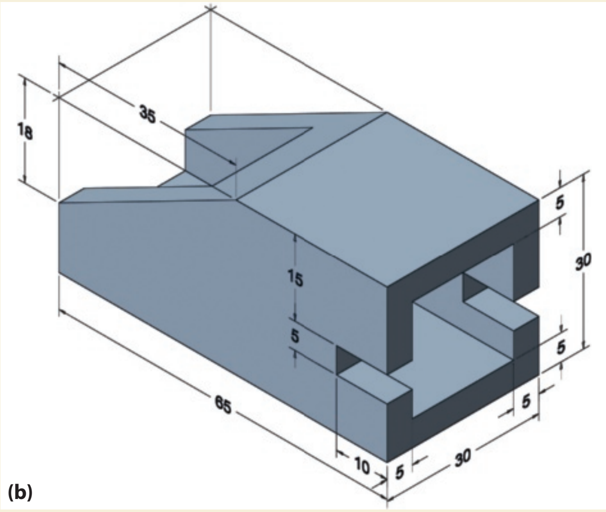
(a)

© 2017 Cengage Learning®. Courtesy of D. K. Lieu.



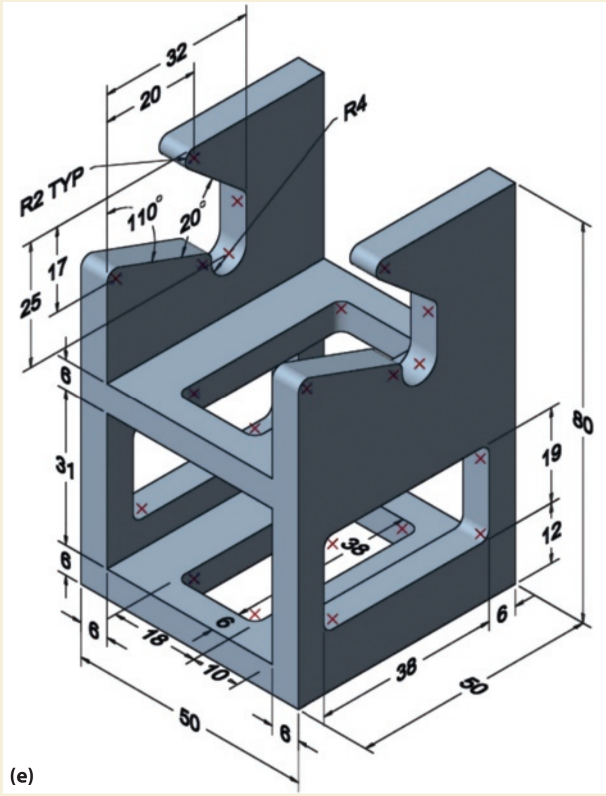
(d)

© 2017 Cengage Learning®. Courtesy of D. K. Lieu.



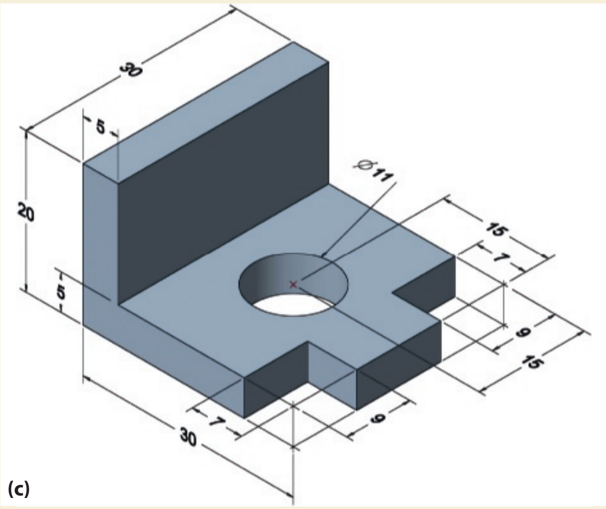
(b)

© 2017 Cengage Learning®. Courtesy of D. K. Lieu.



(e)

© 2017 Cengage Learning®. Courtesy of D. K. Lieu.



(c)

© 2017 Cengage Learning®. Courtesy of D. K. Lieu.