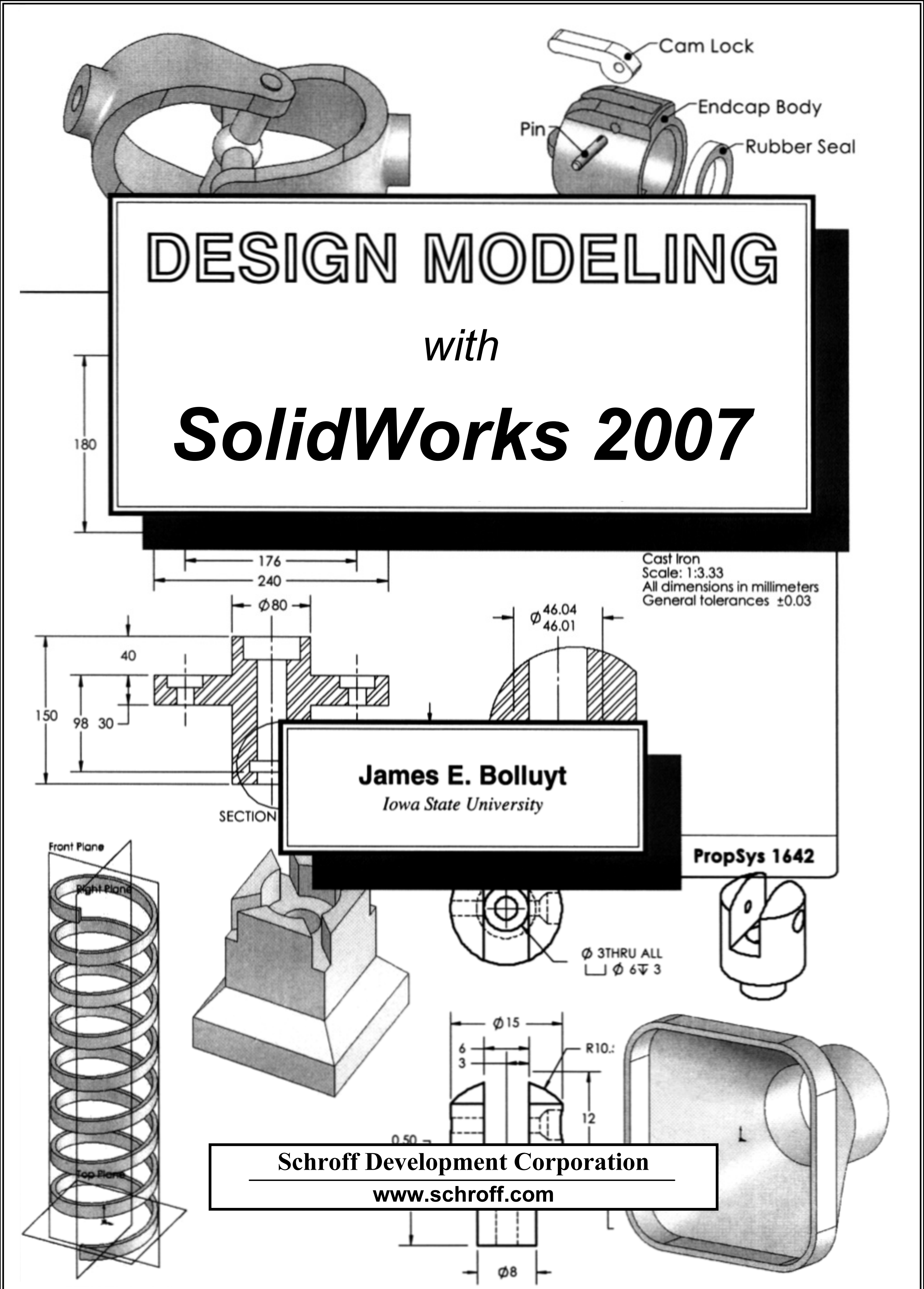
DESIGN MODELING

with

SolidWorks 2007



James E. Bolluyt
Iowa State University

Schroff Development Corporation
www.schroff.com

Parametric Modeling Tools I

3

3.0 INTRODUCTION

Many current CAD packages allow us to create “models” using any one of several representation schemes. For example, even if the software is a 3-D modeler, we might decide to ignore the 3rd dimension and use it to “draw” one or more views of an object on the screen as if we were drawing on paper. This would result in a “2-D wireframe” representation. Choices of 3-D representation schemes might include wireframe, surface, or solid (see Section 1.1). *SolidWorks* allow us to create "models" using any one of these representation schemes. The representation scheme used determines the kinds of operations we can do on the model and the kinds of information that can be associated with the model.

Of the modeling schemes described in Section 1.2, a solid modeling scheme is the most complete and useful. A solid representation of an object is a complete and unambiguous mathematical representation of the geometry of the object. In most current solid modelers, the "material" can be specified by name and associated material properties, including material appearance characteristics. Such a complete computerized representation of an object makes possible many kinds of analyses based on the computer database as well as photorealistic display and hardcopy.

3.1 FEATURE-BASED MODELING

In this chapter, we will begin to use what is known as ***parametric solid modeling***, the approach to the creation of 3-D solid models for which *SolidWorks* was primarily designed (see Section 1.3 for a brief description of parametric modeling). Many current parametric modelers are also known as *feature-based modelers*. ***Feature-based modeling*** is a variation or extension of solid modeling in which various combinations of sweeping and constructive solid geometry operations define typical "features" of solid objects. The primary goal is to provide operations that produce a wide variety of ***geometrical features*** (extrusions, pins, slots, holes, etc.) that can be identified by common ***feature terminology*** (e.g., a "hole" command to create holes) without unduly limiting modeling versatility. In such a modeler, like *SW*, the sweeping and Boolean operations described in Chapter 1 are applied as needed, but we do not specifically request many such operations. These operations are incorporated into feature producing sequences and are therefore often done *transparently* from our point of view. Though *SW* will allow us to "draw" models and create 3-D wireframe, surface, and solid models using various techniques and representation schemes, we will only be taking full advantage of the power and sophistication of *SW* if we create models as ***parametrically defined solid models***. We will continue to use such things as viewing controls, construction aids, snap functions, etc., to make the modeling process easier.

3.2 MODELING IN *SOLIDWORKS* – A FIRST EXAMPLE

To begin to illustrate basic parametric modeling concepts as applied in *SolidWorks*, we will model the retainer shown in Figure 3-1. The procedure we will use to create the model is only one of many that might be employed. When we begin a new model in *SW*, the software loads a simple "start-up" or "template" model file that includes such things as a global origin and XYZ coordinate system, a set of 3 mutually perpendicular reference (or construction) planes, and default values/settings for such things as units, grid spacing, snap, and initial viewport size already defined. We can change or add to most of these start-up features or parameters any time we so desire.

We can also create custom template files to accommodate various units, orientations, model sizes, etc. Such files can be saved either as normal "model" (.SLDPRT extension) files or as "template" (.PRTDOT extension) files. We will look at such starting model files in more detail in Chapter 4. For this first parametric modeling example, we will use a custom template file for parts that comes with this text called DMwSWmm1 (.PRTDOT). "Part" implies a single component that is made as or from a single piece of a given material and is one component of a larger assembly or subassembly. To make the modeling process easier, we will freely use such things as view, display, and rotate functions, sometimes in the middle of other operations. We begin with

File ♦ New ♦ <find and select the part template file DMwSWmm1>

The screen image of the template file should look like that in Figure 3-2. The only "features" visible are the default set of 3 mutually perpendicular reference planes (Front, Top, and Right) and the global origin and coordinate system icon (shorter arrow in +X direction, longer arrow in

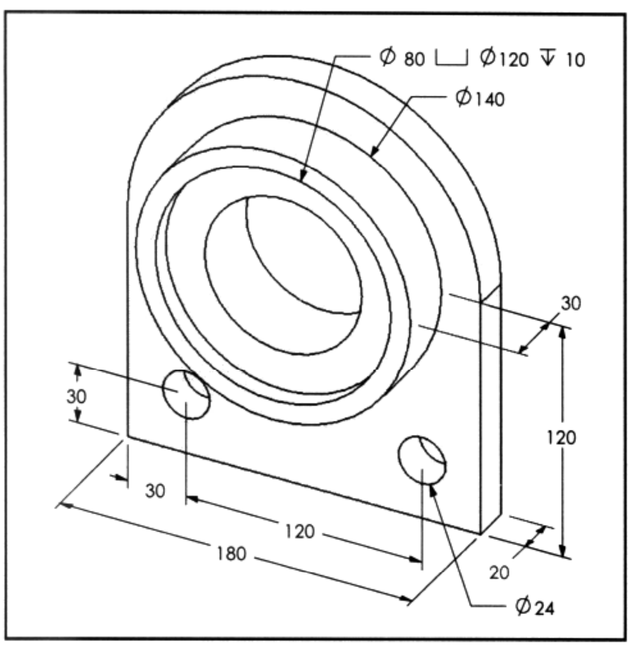


Figure 3-1. *The front-right-top trimetric pictorial of a retainer. Units are millimeters.*

+Y direction, triad "point" at the origin).

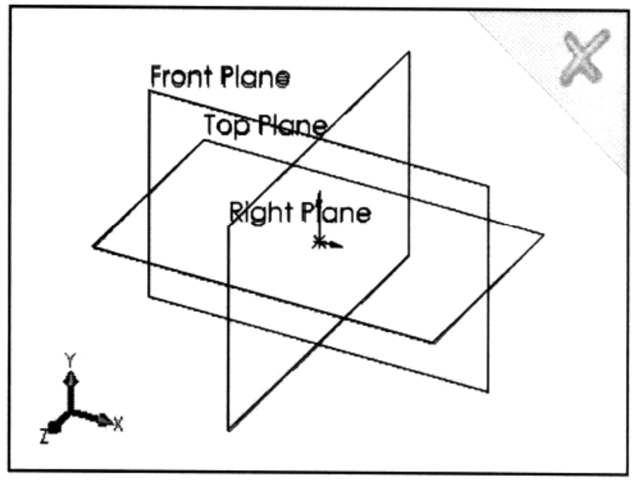


Figure 3-2. *SW prompts us to select one of the three default reference planes for the sketch.*

Beginning the Base Feature

We are now ready to begin to create model geometry. In a parametric modeler like *SW*, it is usually good practice to create the major or overall volume or *base feature* of the part first. The base feature is generally created by applying a sweeping operation of some kind (e.g., extrude or revolve) to a planar profile or "sketched section" (see Section 1.2 and Figures 1-16 and 1-17). For the retainer, we will use a linear sweep or "extrude" operation to create the first major volume or base feature.

The first thing we should consider in creating the base feature is how we want the part oriented in global space. For many parts, the orientation is arbitrary; for some, there is one orientation that is more logical than others because of the nature of the assembly to which the part belongs. Assuming the orientation shown in Figure 3-1 (front-right-top pictorial) is the orientation we want to use, the characteristic shapes of the part features would be seen primarily in the front view. Therefore, we will use the Front plane for the sketch of the base feature. We begin the sketch with

<select (click on any edge of) Front reference plane as sketch plane> ♦ **Insert** ♦ **Sketch**

As soon as we begin a new sketch, a corresponding open sketch listing appears in the Feature Manager window (see Figure 3-3).

This sequence also results in an automatic view rotation such that we are looking perpendicular to the selected sketch plane (in this case, the "front" view). The big advantage of such a view (perpendicular to the sketch plane) is that we see the sketch profile in true shape. Two disadvantages are that we no longer see the third dimension of the model and selecting model items (nodes, edges, etc.) becomes problematic because of the frequent coincidence of such things in a "2-D" image. For these reasons, *SW* does not automatically perform this rotation for subsequent feature sketches, but leaves such view controls entirely at the discretion of the user. Because this author prefers, in general, to work in pictorial views, even for the base feature sketch, we will move back to the default trimetric view (optional) with

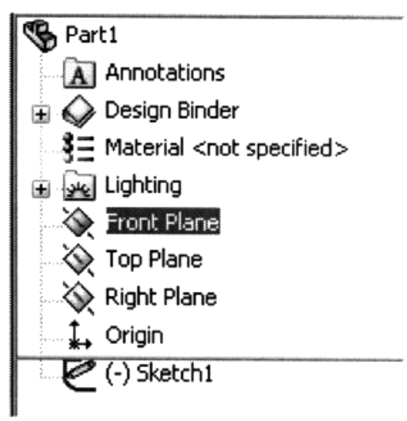


Figure 3-3. *An open sketch listing ((-) Sketch#) appears in the Feature Manager window as soon as we begin the sketch for a new feature.*

View ♦ **Modify** ♦ **Orientation** ♦ <double click on Trimetric (see Figure 3-4)> (this is optional - for any modeling

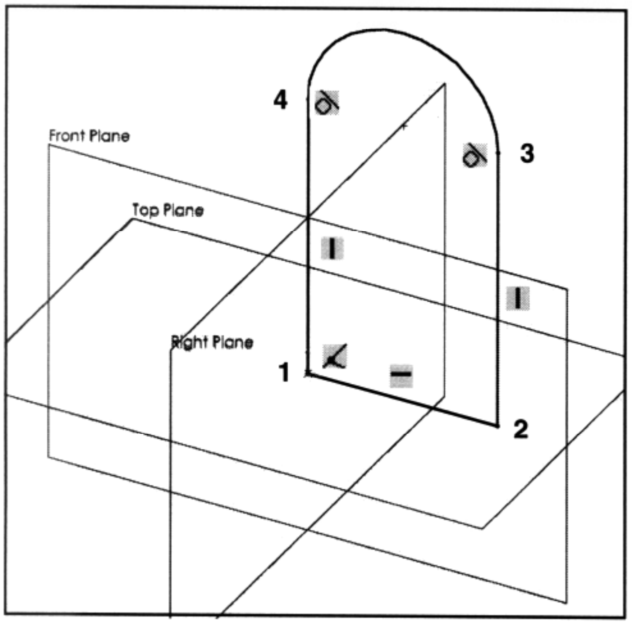


Figure 3-4. *The first "sketch" profile on the Front Plane (as sketch plane), shown in the default trimetric view, consists of 3 lines and an arc (tangent at each end to a vertical line).*

operation, you should work in the view(s) in which you are the most comfortable and confident)

Now that we have identified the sketch plane and obtained the view we prefer, we are ready to create the sketch geometry using the sketching/drawing tools available under the *SW* **Tools ♦ Sketch Entities** menu (or from the tool bar icons activated for sketch mode). We will create the given profile with 3 **Line**s and a **Tangent Arc** as suggested in Figure 3-4.

In general, profiles for sweeps (including extrudes, revolves, and general sweeps) must be closed loops. As long as we connect the sketch entities end to end, *SW* will "see" them as connected to form such a closed loop. To illustrate a primary characteristic of parametric modeling, we will "sketch" this initial profile with dimensions that are not very close to the dimensions shown in Figure 3-1. The reasons for doing that are the default initial screen size (which we could change), but more importantly, to illustrate from the start that in parametric modeling, ***the basic shape or topology of the initial sketch or model is important, but the exact geometry and dimensions of the initial sketched shape(s) are not***. That's not to say that it isn't a good idea to be reasonably close to final proportions and dimensions if practical, but in general, it is not a requirement as it would be with other modeling techniques. That's because *the values to be associated with all dimensions are* ***parameters*** ("constants that can vary" – see Tutorial 1, Section 1.3). As long as new values do not require changes in the basic topology, we can change dimensional values at any time, including after the entire model is "finished." This is one of the primary advantages of parametric modeling - we can easily make changes in the design geometry very near the end of the design process. We will begin to explore how such model "editing" is implemented in *SW* in Section 4.3.

Creating the 2-D Profile or Sketch

We will create the first "sketch" profile with

Tools ♦ Sketch Entities ♦ Line ♦ <move the cursor to the global origin and when the origin point highlights, select it for the starting point (point 1 in Figure 3-4) ♦ <move the cursor to the right some distance and click to identify a

point 2> ♦ <move the pointer up some distance and click to locate point 3) ♦...

Though we need not start at the origin, it is usually convenient to do so. So we will use the **Line** tool to sketch lines from points 1 to 2 and then 3 (Figure 4-2). If we move the pointer anywhere close to directly right after defining point 1, *SW* displays the distance and a 180° note near the cursor to indicate it will assume we want the line to be parallel to the global +X axis (the distance is not critical). In other words, it will ***constrain*** the line 1-2 to "horizontal" (parallel to the current X axis), which is what we want in this case. X and Y displacements are also dynamically displayed in the Status Area (bottom right corner of screen). Once point 2 is defined, *SW* displays horizontal and vertical alignment lines through point 2 and highlights the "current segment" if we move more or less straight up (or down or right or left) from point 2. Once point 3 is defined, we are ready for the **Tangent Arc**. We will continue the sketch creation with

... **Tools** ♦ **Sketch Entities** ♦ **Tangent Arc** ♦ <pick on point 3 and then move the pointer in the approximate shape of the arc we want and when alignment lines appear both horizontally from point 3 and vertically from point 1, click for point 4> ♦ **Tools** ♦ **Sketch Entities** ♦ **Line** ♦ <pick on point 4 and then on point 1> ♦ ...

The path we follow from point 3 to point 4 determines what tangent direction *SW* assumes we want. If the "wrong" arc begins to appear, move back toward point 3 and circle around a little until the correct version appears. Though we do not have to identify point 4 at the intersection of the two alignment lines, that is what we want, so doing so saves us the time of having to specify that information (i.e., those "constraints") later.

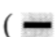
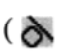
The profile is created in the ***current sketch plane***, in this case, the XY (Z = 0) or Front plane of the *SW* global space. The extruding will automatically be done perpendicular to the plane of the sketched profile. We can do the sketching in a pictorial view as suggested by Figure 3-4 or a "2-D" view (i.e., one perpendicular to the plane of the sketch), the choice is ours.

Mistakes in input can usually be corrected in one of several ways. The **<Esc>** key will abort the current command or action. The **Edit** ♦ **Undo...** (or Undo icon) will, in general, undo the last action taken. Individual sketch entities (geometry and dimensions and constraints) can be deleted by selecting the item/ icon on the screen and pressing the **<Delete>** key.

Constraining the Sketch Geometry

Once we have created the characteristic shape of the sketch profile, we usually want to "completely constrain" or "Fully Define" the geometry! Some constraints may be assumed by the software. For example, lines created close to vertical (i.e., close to parallel to the current Y-axis) will be made exactly vertical, lines that are close to parallel will be made exactly so, etc. – see on-line HELP for more information. But constraints that we want that *SW* does not initially assume we must specify.

There are two general categories of constraints: 1) *geometrical constraints* and 2) *dimensional constraints*. ***Geometrical constraints*** are those that define relationships between two geometrical entities such as tangency between an arc and a line. ***Dimensional constraints*** are, as the name implies, those that specify values associated with size or location. ***In general it is best to make sure that all the geometrical constraints we want are applied before we add dimensional constraints***. Though we could use auto dimensioning (see online Help), we will add most, if not all of the dimensions to text examples "by hand" to ensure that we get the dimensioning scheme we want.

As we create the sketch, *SW* automatically checks to see that we have a valid profile and also whether or not the geometry is completely constrained or ***Fully Defined***. For the current sketch, the status area displays ***Under Defined***, meaning the current geometry is not completely and unambiguously defined, as "understood" by the software. In other words, we probably want to provide additional constraints (geometrical and/or dimensional) to make sure we get the final product we want. Since it is generally best to make sure all the geometrical constraints we want are part of the profile first, we will check those first by simply looking at the sketch image with **View ♦ Sketch Relations**. *SW* uses icons to indicate what geometrical constraints it assumed based on how we created the sketch. For the sketch in Figure 3-4, the icons indicate that *SW* has assumed we want vertex 1 at the origin (coincident icon ), edge 1-2 horizontal ( icon) edges 2-3 and 4-1 vertical ( icon), and arc 3-4 tangent to lines 2-3 and, perhaps, 4-1 ( icon). If the second tangency has not been picked up by the software, we can add it with **Tools ♦ Relations♦ Add...** and then select arc 3-4 and line 4-1 (either order) and then select the **Tangent** relation from the Add Relations list displayed. For this profile, the relations shown in Figure 3-4 are all of the geometrical relationships we need, so we only need add the required dimensional constraints to Fully Define the profile.

SW indicates the constrained status of entities (lines, arcs, etc.) in a sketch using a color code. The code is:

Black: fully defined
Blue: under defined
Red: over defined (e.g., redundant relation or dimension)
Yellow: invalid (e.g., self-intersecting curve)
Brown: dangling (not part of a closed loop)

In the current sketch, two edges are black and two edges are blue, suggesting that at least two additional constraints are required. Because of the arc tangencies and vertical constraint on edges 2-3 and 4-1, if the length of edge 2-3 is added, then the length of 4-1 is fixed (or vice versa). Similarly, if the radius of arc is added, then the length of edge 1-2 is fixed (or vice versa).

So, in this example, we need add only 2 dimensional constraints, one vertical and one horizontal, to completely define the profile. There are several ways we might define both the vertical and horizontal dimensional parameters. For example, for the horizontal dimension we might specify the length of line 1-2, the distance between lines 1-4 and 2-3, or the diameter or radius of arc 3-4. For this first example, we will not worry about why we might choose one or the other of these options, but in real designs it may be an important consideration because of what it implies about how the shape is to be dimensioned for production, how it is to be made, or how the final shape is to be compared to the "ideal" for quality control purposes. For this first example, we will specify the length of line 1-2 (horizontal dimensional parameter) and the distance from line 1-2 to the center of arc 3-4 (vertical dimensional parameter) with

Tools ♦ Dimensions ♦ Smart ♦ <select edge 1-2 and then move pointer some distance below the edge and click to locate the dimension> ♦ <in the automatic pop-up, change the value to 180 (mm) in the dialog (see Figure 3-5)> ♦ <select edge 1-2, then select on the arc 3-4 (*SW* will assume we mean the center of the arc), and then move the pointer some distance to the left of edge 1-4 and click a third time to locate the dimension> ♦ <change the value to 120 in the pop-up dialog (see Figure 3-6)> ♦ <click on OK>

In the Modify dialog box, we can specify whatever values for the dimension *parameters* we want, regardless of what dimensions the original sketch had. The software displays the current value in the dialog and we can accept that value or enter a new one. Note how the profile changes proportions as the new values are entered, though the basic shape does not change because of the

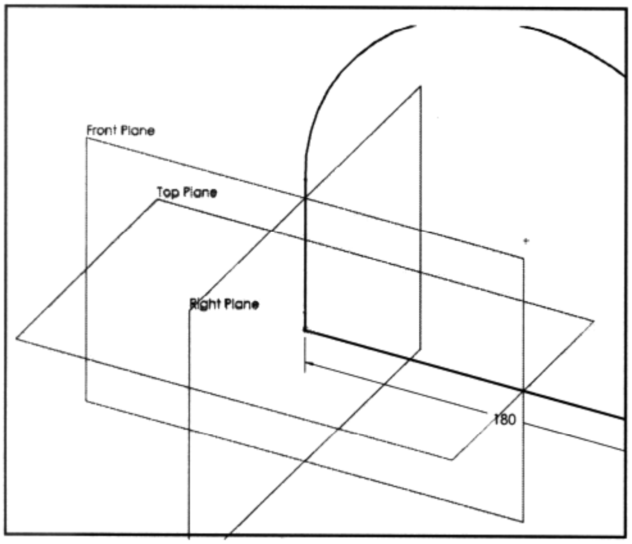


Figure 3-5. *Select the bottom edge with the first click, locate the dimension with the second, and then enter the desired dimension value in the pop-up.*

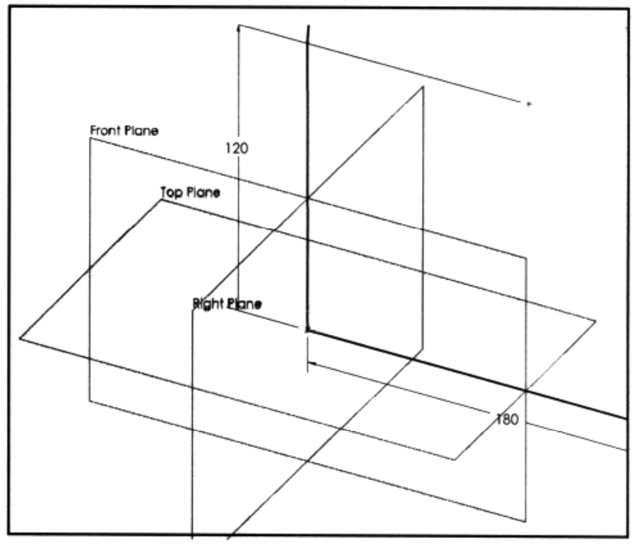


Figure 3-6. *Changing dimension values may result in some of the image disappearing off the screen - use zoom functions as approppriate.*

geometric constraints associated with the profile (compare Figures 3-4 and 3-5). For the second pick in each dimensioning set, the software analyzes where we have picked on the screen to determine if we seem to be picking to locate the dimension line ("off" the sketch) or to select a second sketch element to help define the desired dimension (such as the arc in the second dimensioning sequence above) and reacts accordingly. For arcs and circles, the default reference is the center (we can change the default for an individual arc or circle by using **RMB-click ♦ Properties...**).

Once the two dimensions are added, the message "Fully Defined" appears in the status area. That is, as the software "understands" it, we have now completely defined the shape and size of the base feature profile. However, this definition is parametric; i.e., it is complete, but not necessarily permanent, so we can make changes at any time. Before completing the feature with an extrusion, we will exit sketch mode and zoom the current model with

Insert ♦ Exit Sketch
View ♦ Modify ♦ Zoom to Fit

The (-) disappears from in front of the sketch to indicate it is Fully Defined and the Feature Manager bottom boundary moves up to include the sketch listing (i.e., the sketch would now me saved with the model whether or not we do the extrude).

Completing the Base Feature

Once this first profile sketch is completely defined or "constrained," we will have the software execute the extrusion (or linear sweep) operation with

> **Insert ♦ Boss/Base ♦ Extrude ♦** <if sketch is not still highlighted, click anywhere on it to select it for the extrude> ♦ <in the Extrusion dialog (see Figure 3-7), specify Blind, enter **20** for depth of extrusion, use the Flip button to switch from default +Z extrusion direction to -Z> ♦ check OK button

The **Extrude** selection brings up the Extrusion dialog box shown in Figure 3-7. The **Blind** option tells *SW* that we want to define the distance for the extrusion with an entered value rather than using some existing geometry as a limit. **Blind** (one direction, either +Z or -Z) and **Midplane** (equally in both +Z and -Z directions from the profile plane) are usually the logical options for a first extrusion since most of the other options require other geometry of which there is yet none. *SW* displays an arrow for the default direction (+Z) for the extrusion. Either +Z or -Z would work fine for the extrusion direction for this example. The **Flip** button allows us to make this change. The software always has a current default value for the extrusion Distance D1 (and Draft or taper angle) which we can accept or replace with a new value. *SW* also allows us to specify a combination of Direction 1 and Direction 2 distances, an extremely useful option occasionally, but unnecessarily complicated for this simple case. The base feature resulting from this first extrusion is shown in Figure 3-8.

Figure 3-7. *The direction (+Z or -Z) and depth of the extrusion are controlled in the Extrude dialog.*

Adding the First Detail Feature to the Base Feature

Once the base feature is created, we can begin to add "detail features." The first detail feature we will add is the cylindrical projection with a diameter of 140 (see Figure 3-1). We will again use an extrusion operation to add this feature. *SW* allows us to use any existing planar model surface or reference (construction) plane for such sketches, so we will simply use the front face of the first extrusion as the sketch plane and sketch a circular profile on that plane. We will begin creating the next feature with

> **Tools ♦ Sketch Entities ♦ Circle ♦** <move pointer somewhere inside edges of front face of first extrusion and, when the boundary highlights, click to select sketch plane> ♦ <move the cursor to an approximate center location and click to select (point 1 in Figure 3-9)> ♦ <move the cursor away from the center some distance and click to locate the circle (such as point 2 in Figure 3-9)> ♦ ...

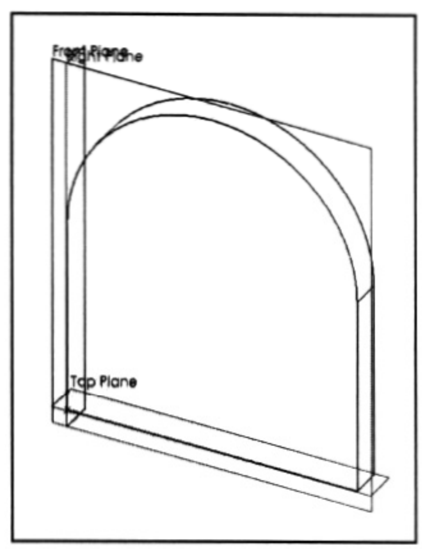


Figure 3-8. *The base feature completed using a "blind" extrusion in -Z to a depth of 20 (mm).*

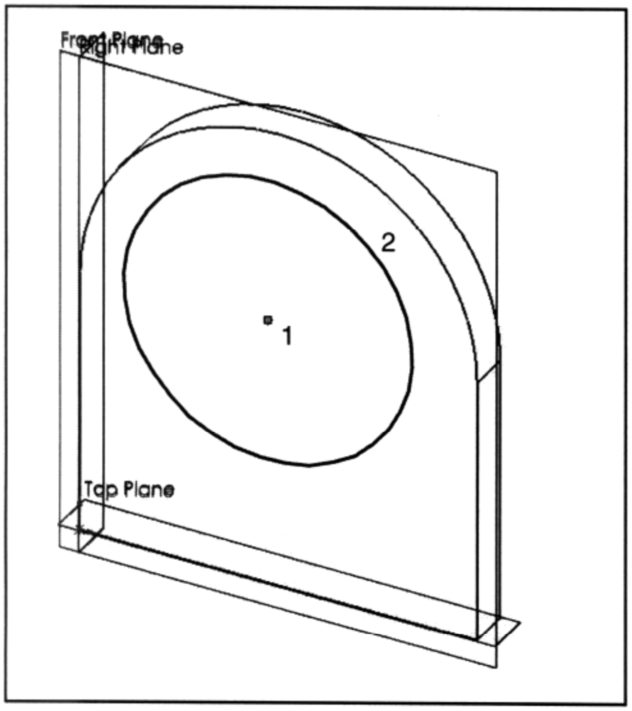


Figure 3-9. *As for the first sketch, the location and size of the second sketch need not be exact.*

For the base feature, we identified the sketch plane before selecting the sketch entity; for this feature, we selected the sketch entity before identifying the sketch plane. In general, the order does not matter and *SW* will prompt us for what it needs. We could even begin with **Insert ♦ Boss/Base ♦ Extrusion** and *SW* would prompt us for the basic things the software requires. For a few operations, however, some orders can confuse the software or cause undesirable results. So as with any sophisticated software, if one order does not seem to work, don't be afraid to try another (even, in some cases, if HELP says otherwise). But also, **SAVE frequently!**

For this second profile, we will center the circle near the position we ultimately want and make the size close to what we want – it is to our advantage to make the location and dimensions of initial sketches reasonably close to the desired final geometry. But it is rarely worth taking the time to do calculations or change **SNAP** and/or **GRID** settings to obtain refined locations or size dimensions for sketches in a parametric modeler.

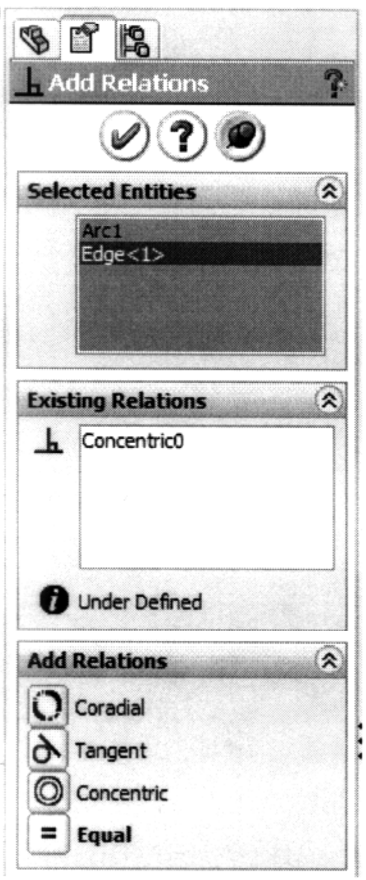


Figure 3-10. *The Concentric relation constrains the circle in both the x and y directions.*

For this circle profile, we next want to add the geometrical constraint of Concentric (to the arc of the first extrusion) and one dimensional constraint (diameter) to fully constrain the sketch. The two sequences, one to locate, the second to size the circle, are

Tools ♦ Relations ♦ Add ♦ <in the Relations dialog (see Figure 3-10), Arc1 (the circle) is listed by default because it's the only geometry available in the current sketch, so all we need do is select the (Edge<1>) arc for the second entity> ♦ <select **Concentric** from the list in the dialog>

Tools ♦ Dimensions ♦ Smart ♦ <select somewhere on the circular edge> ♦ <move the pointer to locate the dimension and click> ♦ <change the dimension value to 140 (mm) in the Dimension dialog> ♦ <if all looks as it should, click on the OK check to accept the (Fully Defined) sketch>

Icons (small concentric circles) appear near the arc and circle as soon as we assign the Concentric relation. Once the diameter dimension is added, the status area should again say ***Fully De-***

fined. To extrude this circular profile we will use

> **Insert** ♦ **Boss/Base** ♦ **Extrude** ♦ <in the Extrusion dialog, specify Blind, enter **30** for depth of extrusion, use the Flip button to switch the extrusion direction from -Z to +Z if necessary, make sure Merge result box is checked> ♦ check OK (green check) button>

As for the first extrusion, we want to use the blind termination option so we can specify the depth of the extrusion (D1 in the dialog) with a value (**30**). We want this second extrusion to be "added to" the first extrusion, hence the **Merge result** option (Boolean *union* operation). If the default extrusion direction is +Z (likely), we do not need to use the Flip arrow this time. The resulting solid is shown in Figure 3-12 (Shaded with Edges). To increase or decrease the display resolution (smoothness of curved edges and surfaces) of the image, we can use

> **Tools** ♦ **Options** ♦ **Performance** ♦ <click on Go to Image Quality button at bottom right of dialog> ♦ <in Image Quality dialog, slide the HLR/HLV higher (for higher screen resolution but also slower performance> ♦ <click on OK>

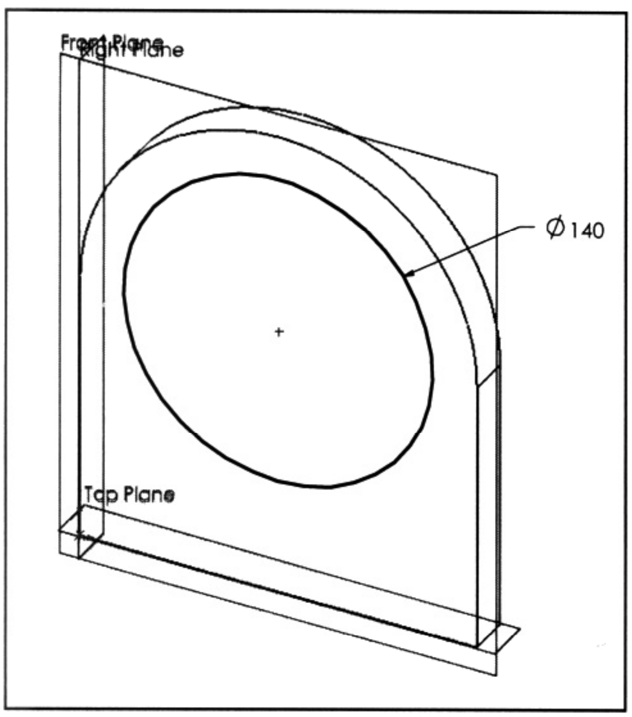


Figure 3-11. *The Concentric relation and one diameter dimension Fully Define the second sketch.*

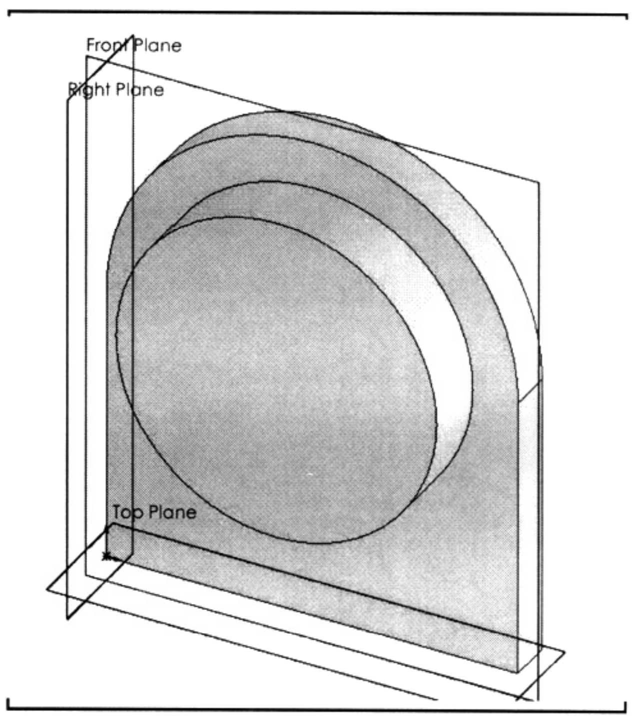


Figure 3-12. *The results after the second extrusion, merged (unioned) with the base feature extrusion.*

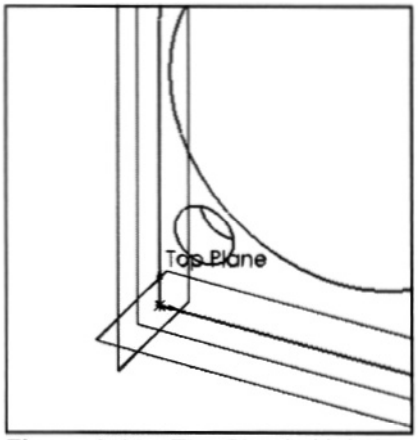


Figure 3-12. *The Simple Hole is innitially placed at the point we click at on the surface we click on.*

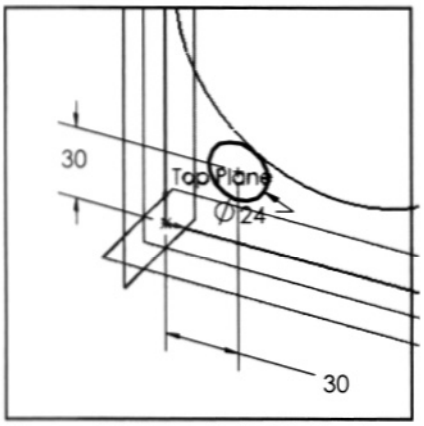


Figure 3-13. *To Fully Define the Hole sketch (centerpoint), we must add locating dimensions.*

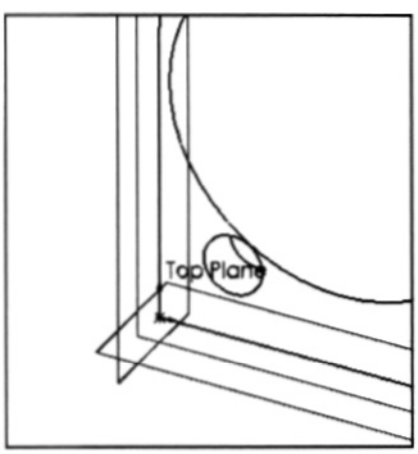


Figure 3-14. *The first Simple Hole completely located and sized.*

Adding the Hole Features to Complete the Model

Some kinds of detail features (for example, *counterbored holes* and *chamfers*) have more or less standard geometry or topology. For many such features, *SW* provides built-in procedures under the **Insert ♦ Features** menu. To complete this first model, we will create two "simple" holes (holes with constant diameter) and one ***counterbored hole*** (two diameters, larger diameter near the placement surface) using the *Hole Wizard.* The sequence to place the first simple hole (bottom left corner of the base feature) is

> **Insert ♦ Feature ♦ Hole ♦ Simple ♦** <select near the desired hole location on the front surface of the first extrusion> ♦ <in the Hole dialog, change depth to Through All (rather than Blind with a distance), change the diameter to **24**> ♦ <check OK button>

Though **Blind** could be used here too, in general **Through All** is a more logical choice if, in fact, we will always want the hole to go through no matter what we might make the **Depth** of the first extrusion. For the **Simple** Hole, *SW* places the hole on the surface we click "on" at the location on that surface we click "at" (see Figure 3-12). But *SW* (2007) does not assign locating dimensions to the hole "sketch" (a single point locating the hole), so we need to add those dimensions and enter appropriate values as we did for the previous sketches. The sequences are

> <click on the + in front of the Hole listing in the Feature Manager window to expand the Hole feature, then RMB click the Sketch# listing under Hole1> ♦ <in the pop-up, select **Edit Sketch**> (note that the Hole sketch listing has a (-) in front of it, meaning it is not yet Fully Defined)
>
> **Tools ♦ Dimensions ♦ Smart ♦** <select the front-left edge of first extrusion, then the circle defining the hole, then click to place the dimension; change the dimension value to 30 (see Figure 3-13)> ♦ <select the front-bottom edge of the first extrusion, then the circle, then click to place the dimension; change the dimension value to 30> ♦ <check OK button>
>
> **Edit ♦ Rebuild** (to get *SW* to update model using new parametric values)

The completed hole feature is shown in Figure 3-14.

We will create the second Simple Hole in a similar manner as an independent feature, though we could also use a copy operation. The copy option might have some advantages if, for example, we decided to change the size of the hole. If the second were a copy of the first, changing the diameter of the original would automatically change the diameter of the second. That kind of reasoning suggests the concept of ***design intent***. If we *intend* for the two holes to always be the same size, then it is logical to incorporate that *design intent* into the model database if such things are possible. In *SW* (and other high-end CAD packages), such things are possible. We will explore this concept in Chapter 5.

For this first example, however, we will create both holes as independent features except that we will dimension the second hole with respect to the first in the horizontal directions which is consistent with the dimensioning scheme shown in Figure 4-1. It is also likely to be a better way of insuring acceptable alignment between these two holes and the corresponding holes in any other parts (as opposed to, say, locating the second hole from the right front edge). The sequence is

Insert ♦ **Feature** ♦ **Hole** ♦ **Simple** ♦ <click near the desired location for the second hole on the front surface of the first extrusion> ♦ ... (complete as for the first hole)
<RMB click on the Hole2 listing in the Feature Manager window> ♦ <in the pop-up, select **Edit Sketch**>
Tools ♦ **Dimensions** ♦ **Smart** ♦ <select the first hole, then the new hole, then click to place the dimension; change the dimension value to 120 (see Figure 3-15)> ♦ <select the front-bottom edge of first extrusion, then the second hole, then click to place the dimension; change the dimension value to 30> ♦ <check OK button>
Edit ♦ **Rebuild** (to get *SW* to update model using new parametric values)

The resulting model is shown in Figure 3-16.

The last feature we need for this first model is a counterbored hole (see Figure 4-1). For a counterbored hole, we must specify the through-hole diameter (**80**), the counterbore diameter (**120**), and the counterbore depth (**10**). Though we could use two simple holes of different diameters and depths to create the counterbored hole, it is often more appropriate and somewhat simpler to use the Hole Wizard. The Hole Wizard dialog (Figure 3-17) allows us

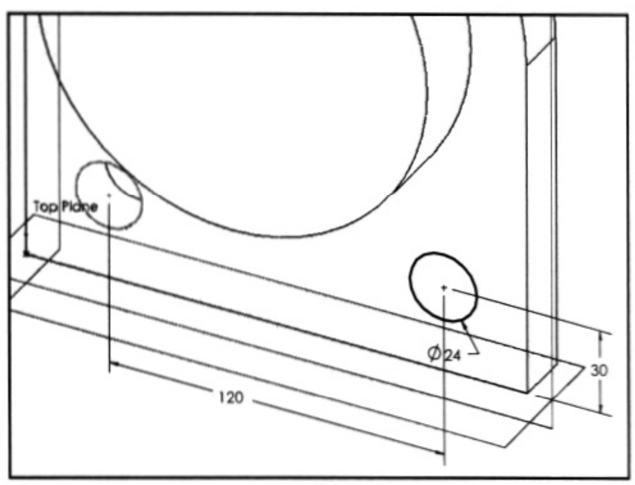


Figure 3-15. *The second hole is probably most logically located with respect to the first hole.*

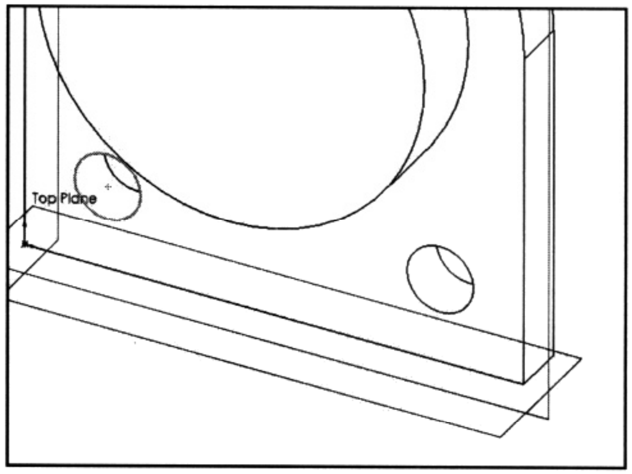


Figure 3-16. *The retainer model after the second hole has been located and sized.*

to select a standard counterbored hole or enter values for a custom version. The values given in Figure 4-1 are custom values. The logical locational specification for this hole is the Concentric relation. The easiest way to get *SW* to apply this relation is to briefly hold the pointer on the front circular edge of the second extrusion to both identify the placement surface and get *SW* to highlight the center of this surface and then click on this center to place the hole (we could also place the hole anywhere on the surface and then add the concentric relation with an Edit Sketch operation). With reference to Figures 3-17 and 3-18, the sequence to obtain the counterbored hole using the Hole Wizard is

> **Insert** ♦ **Feature** ♦ **Hole** ♦ **Wizard** ♦ <in the Hole Specification dialog (Figure 3-17): select **Ansi Metric** as the Standard; click on the Counterbore icon; for End Condition, select Through All; under Custom Sizing, click in the Through Hole Diameter box and type in **80**; in a similar fashion, change the C-Bore Diameter & Depth to **120** (diameter) and **10** (depth), respectively; click on the Positions tab near the top of the dialog, then hold the pointer on the circular edge of the second extrusion and, when the circle-center icon appears at the center of the front face, click on it; select the **Finish** button>

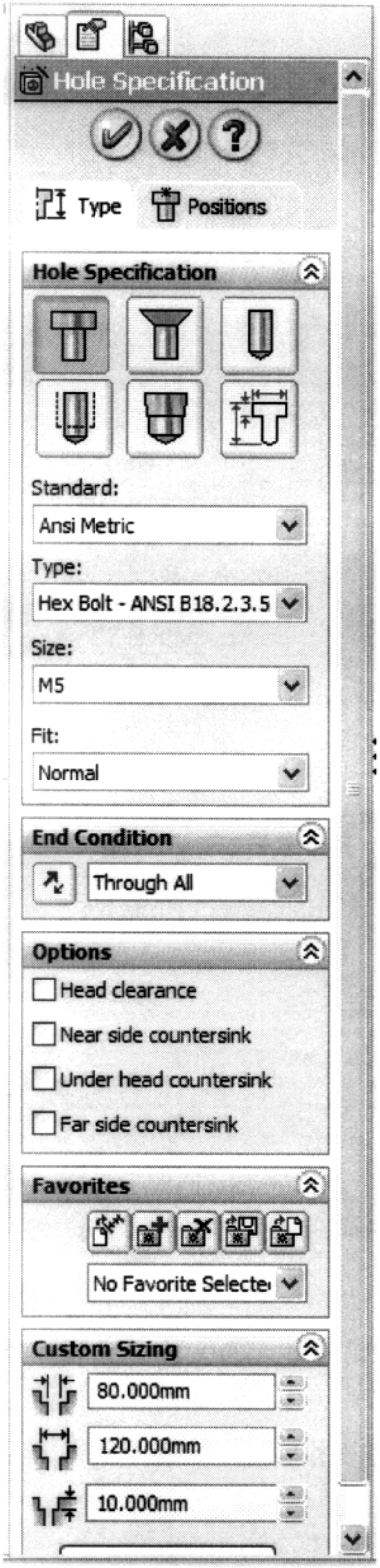


Figure 3-17. *The Hole Specification dialog includes parameter values for a wide variety of "standard" counterbored and countersunk holes.*

The completed initial model with the counterbored hole placed concentrically in the front face of the second extrusion should look like that in Figure 3-18. (To turn off the three basic Reference planes as suggested by the figure, hold the **<Ctrl>** key down and LMB click on the *Front Plane*, *Top Plane*, and *Right Plane* listings in the Feature Manager window, then RMB click on one of the three, and then in the pop-up, select **Hide**.)

Before exploring one of the biggest advantages of parametric modeling over traditional solid modeling techniques, we will save this initial version of our parametric model with

File ♦ **Save As** ♦ <move to an appropriate directory and enter "RETAINER" or other name of your choice and save as normal *SW* part model file, .SLDPRT extension>

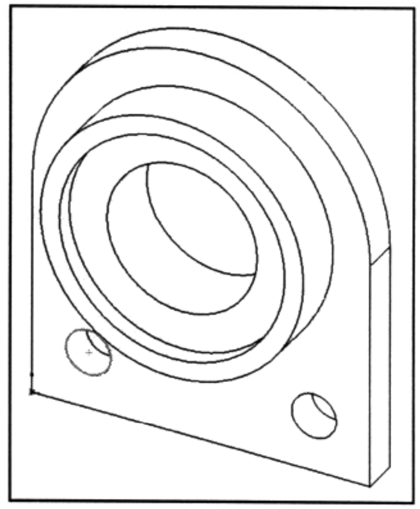


Figure 3-18. *The initial version of the first parametrically defined solid model completed.*

3.3 ADVANTAGES OF PARAMETRIC MODELING

It may seem like considerable time is required to input all the information required by *SW* (i.e., a parametric solid modeler) to define model geometry compared to more "straight forward" solid modeling techniques available in numerous other modeling packages . If design were a straight forward endeavor and we always knew all the dimensions, relationships, etc., that would make the best design right from the start, it would probably not make sense to use software that works like *SW*. The fact of the matter is, however, that ***design is by nature an iterative process***. We rarely and perhaps never know up front what the final best design will be. Analysis and redesign often continue right up until the final production/construction/fabrication process begins, perhaps beyond. Parametric modeling can offer tremendous benefits in design environments in which changes in model geometry are being made long after the initial design solution models were created. These benefits can more than make up for the "extra" time spent in the initial modeling stages.

One of the characteristics of parametric (or *variational*) modelers that provides a powerful tool in design refinement and redesign is the fact that the model geometry is driven by the geometrical and dimensional constraints associated with the model, and we can modify any or all of these constraints (within realistic limits) any time we want. To illustrate the nature of this concept, we will make a few "design changes" in the retainer we just created (be sure to **SAVE** the original model as a .SLDPRT file before proceeding). First we will change the width of the base feature from 180 to 240 (see Figures 3-19 and 3-20). The sequences are

<RMB click on the name of the base feature (Extrude1?) in the Feature Manager window (see Figure 3-21)> ♦ <select the **Edit Sketch** option in the Feature pop-up (Figure 3-22)> ♦ <LMB double click on the 180 value in the sketch

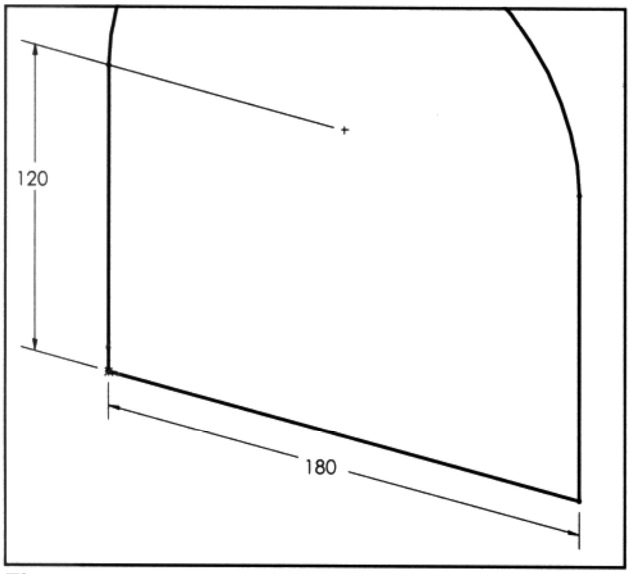


Figure 3-19. *In **Edit Sketch** mode, dimensional values are displayed for the selected feature.*

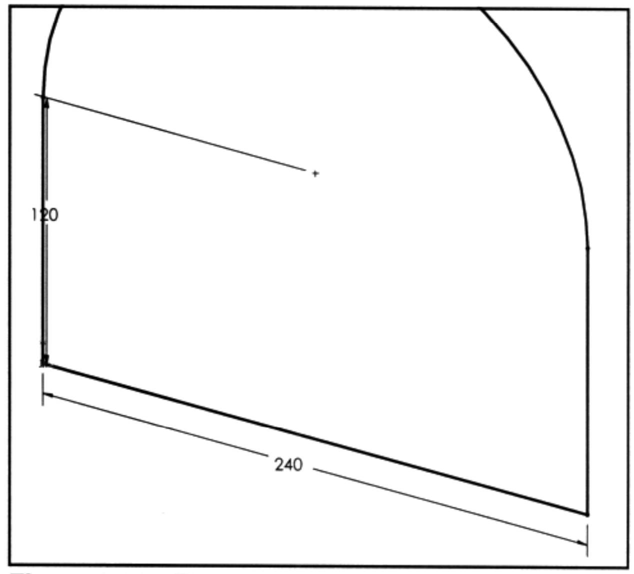


Figure 3-20. *The sketch profile display changes in reponse to new dimensional values specified.*

image (Figure 3-19)> ♦ <enter new value of **240** in the dialog (results shown in Figure 3-20)> ♦ <click on OK check in Dimension dialog>

Edit ♦ Rebuild (to get *SW* to check and regenerate model using new parametric values)

The **Rebuild** command makes the software recheck the results of any changes we have made to make sure we still have a valid model and then regenerates the model image. The results of this first change are shown in Figure 3-23.

For the changed base feature width, the two "drilled" holes are no longer symmetrically placed (though we could add a dimension equation to the model file so symmetry would be maintained automatically – see Chapter 5). But changes can easily be made in such detail features as well as the base feature, so we will move the right hole (as seen in Figure 3-23) and also change its diameter. To suggest that in making such changes, it may be possible to "stretch" the general rule that such model editing should not change the initial topology, we will use some new values that we would not likely want in a real design. The command sequences are

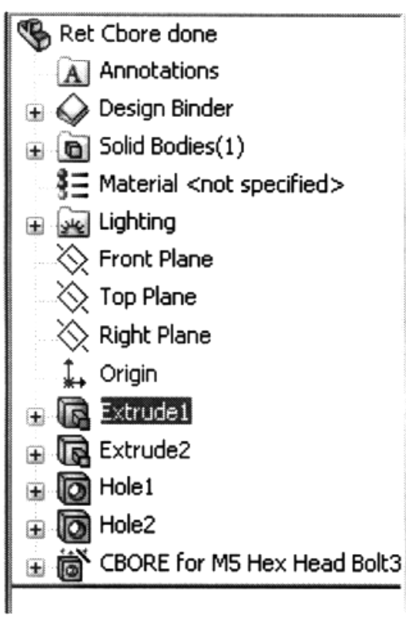


Figure 3-21. *To make changes to an existing feature, RMB click on listing in Feature Manager and select **Edit Sketch** (or **Edit Feature** for changes in sweeping parameters, etc.).*

<RMB click on the name Hole 2 in the Feature Manager window > ♦ <select the **Edit Sketch** option in the Feature pop-up> ♦ <LMB double click on the 120 value in the sketch image)> ♦ <enter new value of **210** in the dialog> ♦ <click on OK check in Dimension dialog>

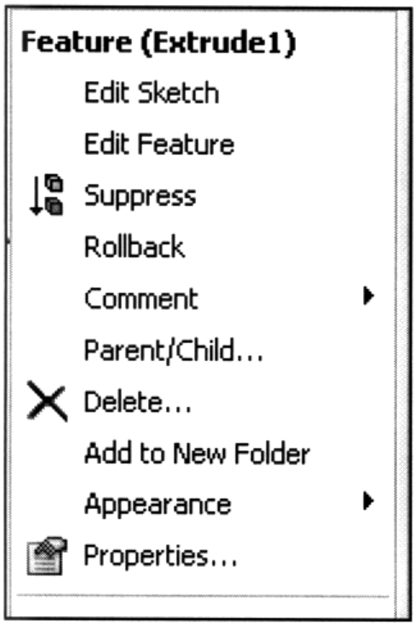


Figure 3-22. *Each "feature pop-up" gives un numerous options for adding information, making changes, deleting, etc.*

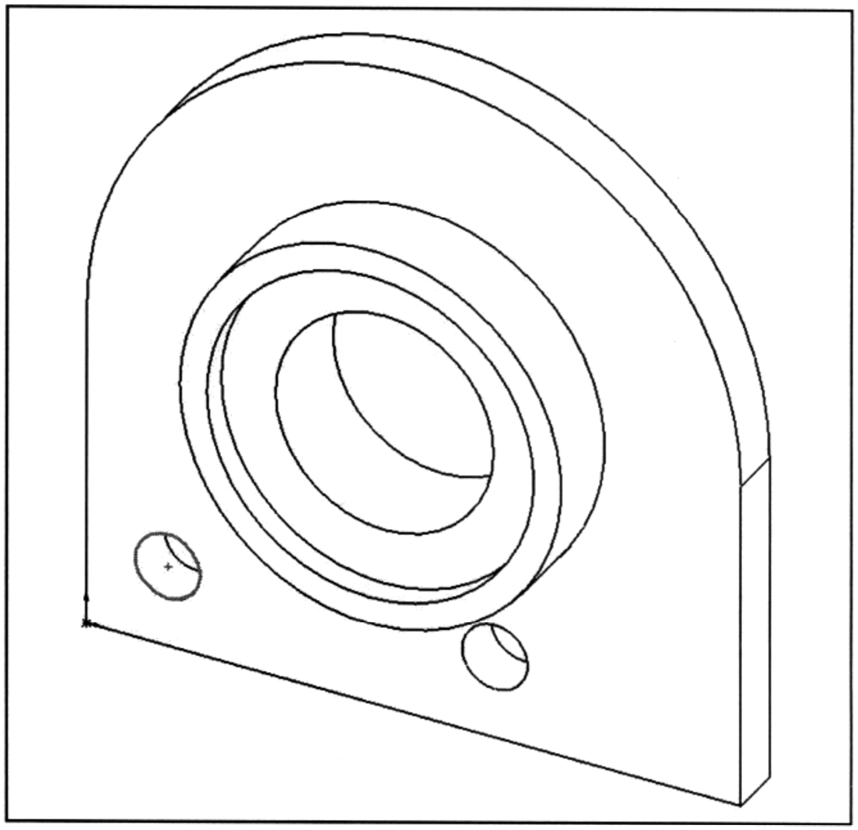


Figure 3-23. *As the model is currently defined, the distance between holes does not change with the base feature dimension change.*

Edit ♦ **Rebuild** (should now look like Figure 3-24)
<RMB click on the name Hole 2 in the Feature Manager window >
♦ <select the **Edit Feature** option in the Feature pop-up> ♦ <change the diameter value to 60)> ♦ <click on OK check in Hole 2 dialog>

The results of these two changes are shown in Figure 3-25.

These changes begin to suggest how easy and quick it is to make changes in a parametrically defined model if design changes so dictate. And because part models, assemblies, and drawings (documentation) are so closely bound together in *SolidWorks*, such changes are automatically propagated to all associated files. This makes parametric modeling a tremendous tool for an iterative process like engineering design.

3.4 PARAMETRIC MODELING - A SECOND EXAMPLE

The first parametric modeling example made use of linear sweeps (extrudes) and placed features (holes). A second type of sweep that is used frequently to create model features is a circular sweep or what *SW* calls ***Revolve***. To illustrate the use of circular sweeps, we will create the model of the plastic handle shown in Figure 3-

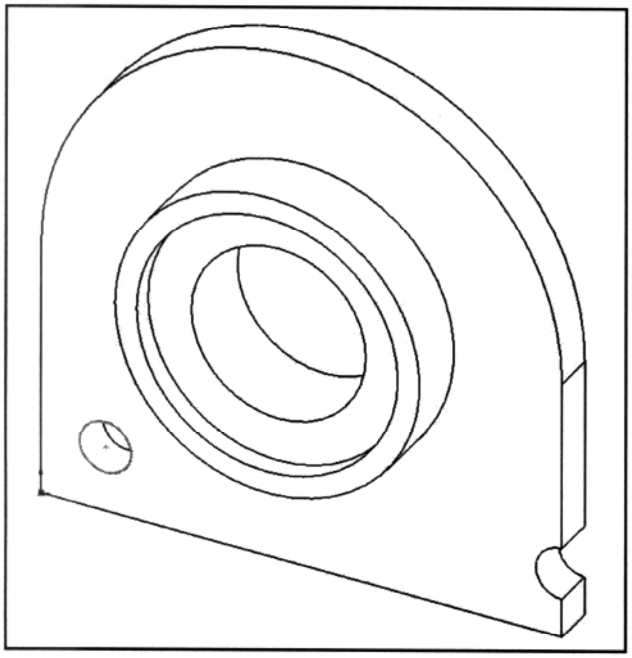


Figure 3-24. *The retainer model after changing the horizontal locating dimension for the right hole.*

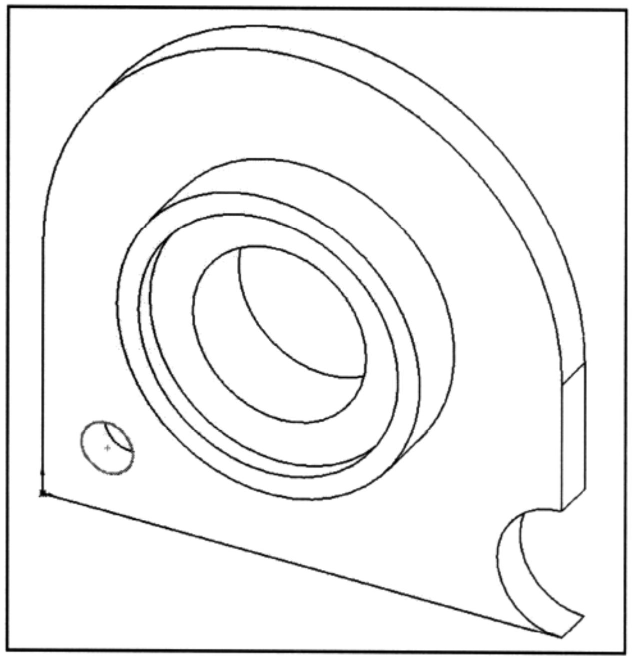


Figure 3-25. *The retainer model after changing the size of the right hole (to a questionable value?).*

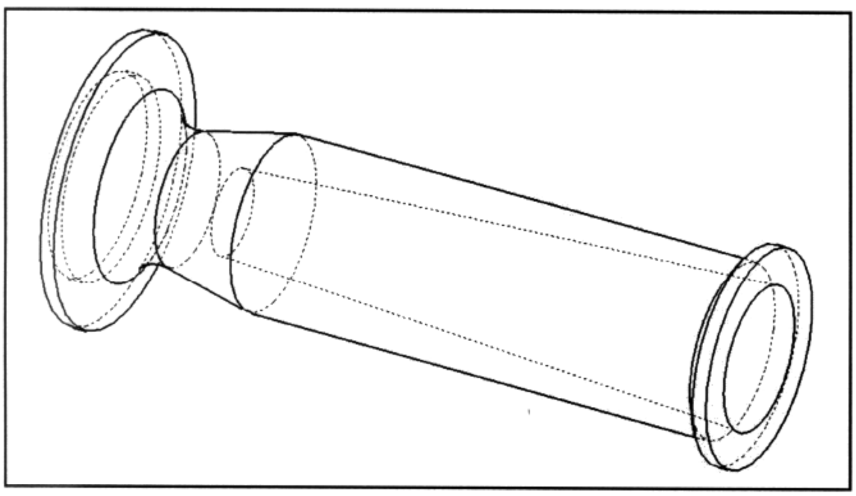


Figure 3-26 *Circular sweeps or "revolves" are an easy way to create shapes like this plastic handle in 3-D solid modelers like SolidWorks.*

26. To create this solid model, we will use two **Revolve**s, one to model the outer shape and one to "hollow out" the handle. Though we could create this model with one revolve (and a more complex profile), *it is usually better to use* ***a larger number of simpler shapes*** *(and Boolean operations) than fewer but more complex shapes to create models.*

Before we actually begin creating the part geometry, we will open the part template used in Section 3.2, change the unit system to IPS (inch-pound-second), and then save the file as a second part template for models with units of inches. The sequences are

File ♦ **Open** ♦ <change the "Files of type" specification to **Templates (.prtdot, .asmdot, drwdot)**> ♦ <locate and select DMwSWmm1 (.prtdot)>

Tools ♦ **Options** ♦ <under **Document Properties** tab, in the **Units** dialog, select the **IPS** system; in the Detailing dialog, set the trailing zeroes option to **Remove**; click on

OK>

File ♦ Save As ♦ <make sure the "Files of type" specification is still **Templates (.prtdot, .asmdot, drwdot)**> ♦ <save as DMwSWinches1 (.prtdot) in an appropriate directory (see Text Files section at front of text)>

Now to begin the model, we will first tell *SW* to add a reference axis along the intersection of the Top and Front reference planes. Though we could identify an edge of the first sketch profile as the axis for the revolve, we are using two revolves to create the basic handle geometry, so setting up a reference axis that can be used for both revolves is a good way to avoid any ambiguity. Once the axis is created, we will also turn off (Hide) the Top and Right reference planes since we will not use either of those for this model. To set up the axis and then turn off two of the three Reference planes, the sequences are

Insert ♦ Reference Geometry ♦ Axis ♦ <in the Axis dialog, select the **Two planes** option and then select the **Front** and **Top** reference planes (see Figures 3-27 and 3-28)>

<RMB click on the Top Plane listing in the Feature Manager window, then select **Hide** from the pop-up; repeat for the Right Plane (see Figure 3-28)>

Since we are going to use a **Revolve**, the first sketch should be a half-section profile of the overall shape of the handle (*SW* generally requires that profiles for Revolves be closed shapes). Figure 3-29 shows the relations (or geometric constraints) we want for this half profile and also the dimensions defining the half profile. With a little care and patience in sketching, we can get *SW* to "see" all of the relations we want in the sketch. For example, the portion of the sketching sequence used to get *SW* to "see" the two tangencies at either end of the arc was

<locate nodes 1, 2, ..., 7; with Line still active, select the **Tangent Arc** tool (thereby replacing Line as the active tool), then select node 7 again and then node 8; with the Tangent Arc tool still active, select the Line tool, then select node 8 again and locate point 9 along the dashed tangent construction line that appears automatically whenever the pointer is close to this tangent direction>

If the software does not "see" all of the relations suggested by Figure 3-29, we can simply **Add** them (or **Delete** those *SW*

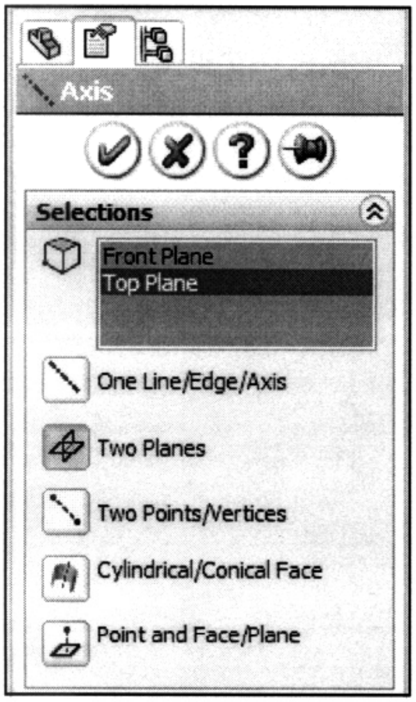


Figure 3-27 *The Front and Top Reference Planes are used with the Two Planes option to define the axis to be used for the circular sweeps or "revolves".*

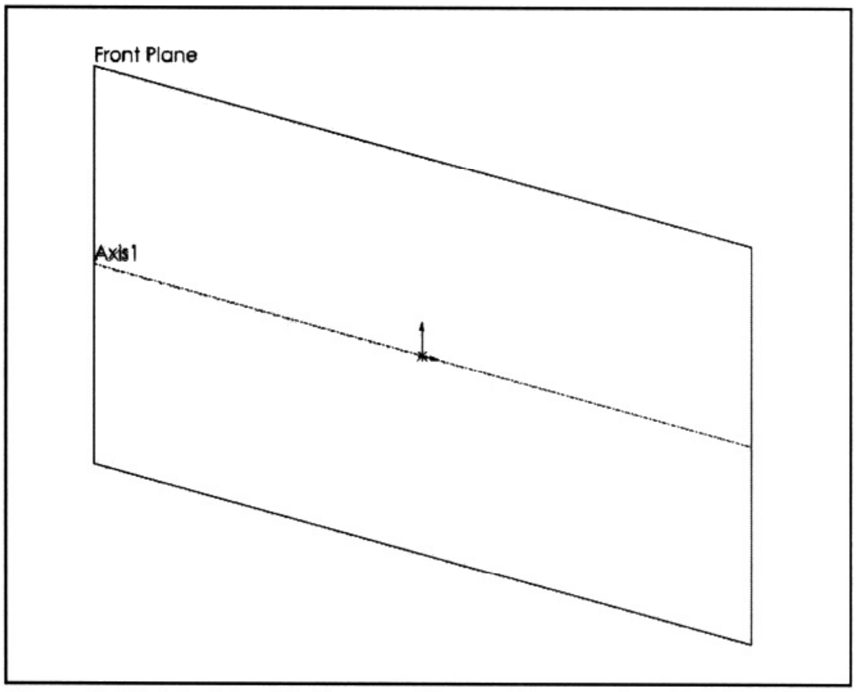


Figure 3-28 *The Front Reference Plane will be used as the sketch plane for both circular sweep (Revolve) profiles.*

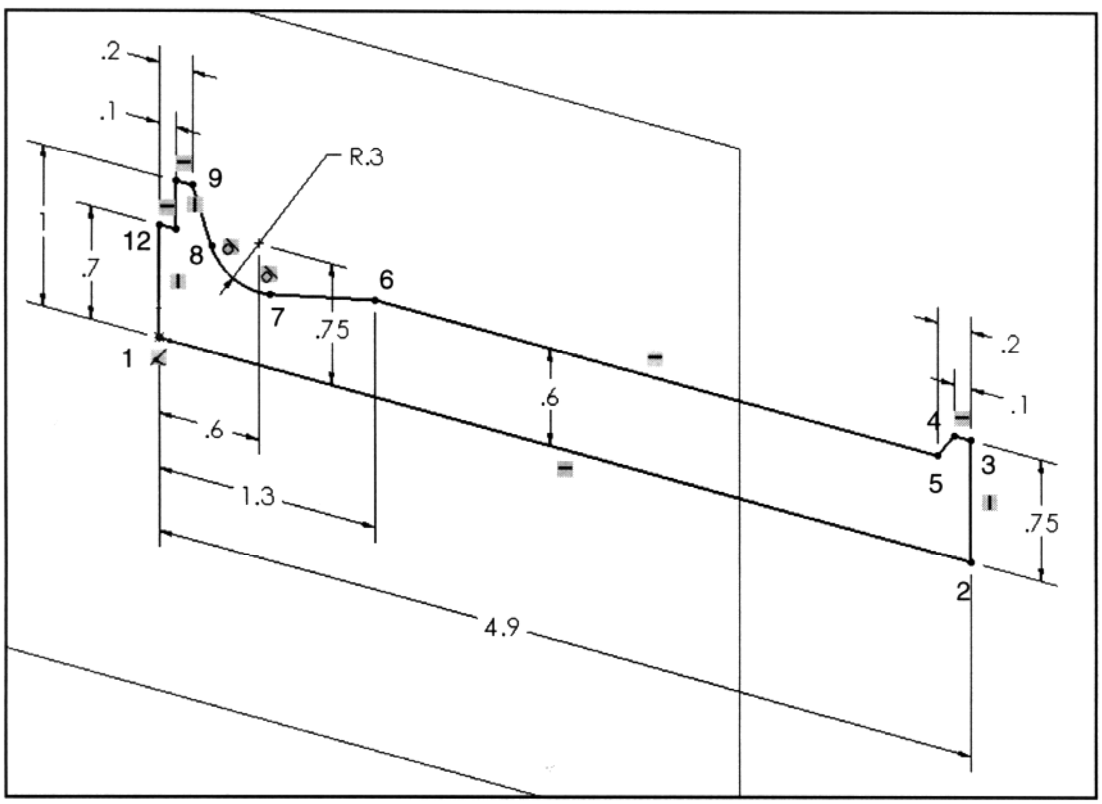


Figure 3-29 *With a little care, we can get SW to "see" that we want all but three of the lines to be either horizontal or vertical, the arc tangent to two of the sloping lines, and the bottom and left edges meet at the origin (points 10 and 11 not labeled in figure because of space).*

assumes that we don't really want, which happens occasionally). The sequences for sketching and constraining the profile are

<select **Front** reference plane as sketch plane> ♦ **Insert** ♦ **Sketch** ♦ <keep default Front view or switch back to Trimetric as suggested by Figure 3-29)

Tools ♦ **Sketch Entities ...** (starting with point 1 at the origin, use lines and a tangent arc to create a sketch profile like that in Figure 3-29)

Tools ♦ **Relations** ♦ **...** (**Add** any relations shown in Figure 3-29 that are not shown in your sketch; **Delete** any unwanted relations)

Tools ♦ **Dimensions** ♦ **Smart** ♦ **...** (add dimensions shown in Figure 3-29; try to add dimensions in an order that does not produce overly distorted shapes, especially intersecting edges; add dimensions (and relations) until status line says "Fully Defined"; click on green check to confirm sketch)

The sketch shown was created as a continuous path starting at the origin (node 1 in Figure 3-29) and working counterclockwise. If we want to interrupt the sketching process, we can double click to deactivate the current sketching tool (or unselect through the menu or tool bars) without selecting a new tool. We can also use **Edit ♦ Undo** (or the Undo icon, repeatedly if necessary) to undo or erase the previous sketch entity.

As illustrated in earlier examples, the dimensions and proportions of the initial sketch need not be exact, but making them close to the final dimensions intended makes it less likely that the constraining process will be confusing or difficult. As for the first modeling example, it makes sense to use the origin as the starting point (or "anchor" point) for the sketch (as shown by a "coincident" relation icon). This corner of the sketch will then remain fixed in model space; all other points are free to move with respect to this anchor point as additional constraints (relations and dimensions) are specified. Using the global origin for this anchor point (or adding a **Fix** relation to some other designated "anchor" point), making initial sketches about the right size and shape with respect to this point, and giving a little thought to the order in which dimensions are entered can avoid confusing, perhaps even troublesome, intermediate sketch shapes. ***It usually works best to work "outward" from this anchor point in adding dimensions.***