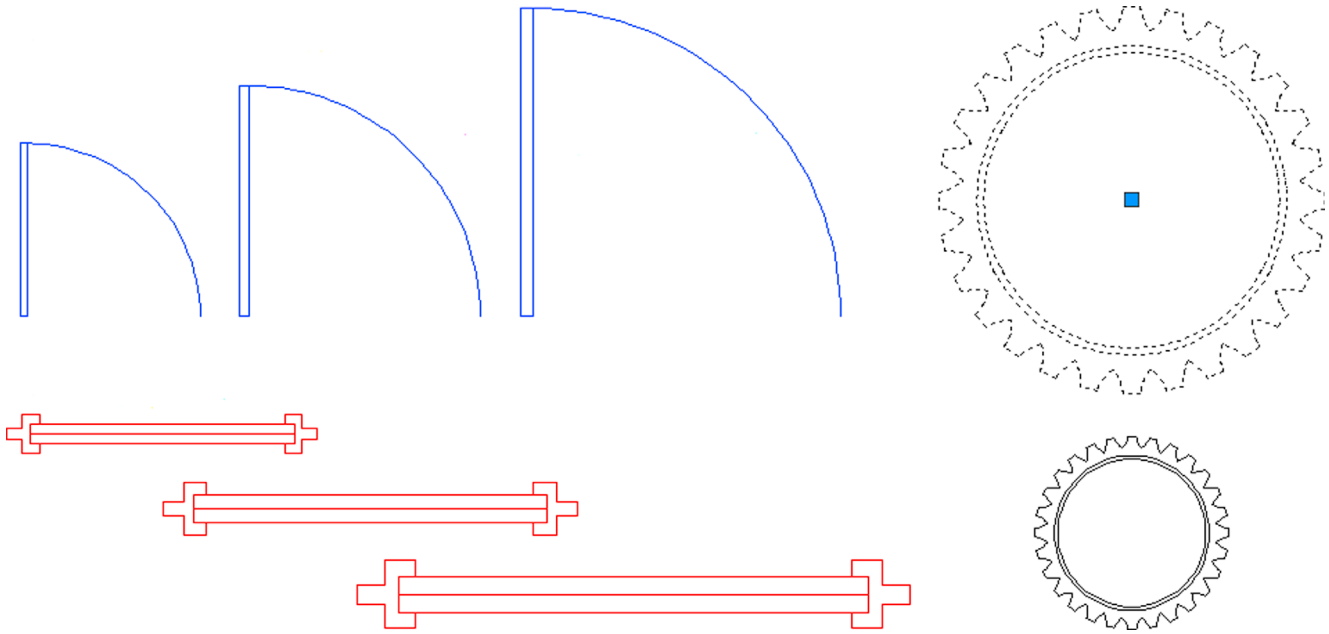
Chapter 7

Blocks, Wblocks, Dynamic Blocks, Groups, and Purge



Learning Objectives

In this chapter, we introduce the concept of blocks and discuss the following:

- Creating and working with blocks
- Retain, convert, and delete options
- Redefining blocks
- Inserting blocks
- Purge
- Creating and working with wblocks
- Creating and working with dynamic blocks
- Creating and working with groups

At the end of this chapter, you will be able to create blocks, wblocks, dynamic blocks, and groups as well as insert symbol libraries and purge your drawing.

Estimated time for completion of this chapter: 1–2 hours.

Up and Running with AutoCAD 2017: 2D and 3D Drawing and Modeling. DOI: http://dx.doi.org/10.1016/B978-0-12-811058-4.00007-4

7.1 INTRODUCTION TO BLOCKS

This chapter is about some very useful AutoCAD concepts that greatly simplify your workload. Blocks, wblocks, dynamic blocks, and groups are all members of the same family. They exist around the idea that objects can and should be grouped together, if possible, to allow for easier handling and storage for future reuse.

Let us formally define a block. It is a collection of objects that are grouped and held together under some identifying name. These objects can then be copied, moved, erased, and just about anything else, all as one unit. Think of blocks as electronic "glue" that binds the objects they contain to each other.

There are many benefits to this. If you have an office table with eight chairs around it, and you need to move them all around many times while optimizing a room layout, it is a lot easier to click on a block once than attempt to select all nine pieces using a Window or a Crossing or one at a time. Imagine an engineering example involving a simple gear with all its intricately designed teeth. Dozens if not hundreds of lines and arcs may make up the gear. It would be not just wise but mandatory to form them all into a single block.

There is however another, more compelling, reason for blocks. That is the concept of *reuse* and *symbol libraries*. Blocks can be saved and reused for future drawings that may need the same exact object. That way you never have to draw the same thing twice, an AutoCAD Golden Rule. A collection of these objects cataloged in some orderly fashion becomes a symbol library, an indispensable part of just about any designer's toolset. Likely candidates for these libraries include doors, windows, furniture, bolts, standard parts, fasteners, and really just about anything that can be used in more than one design or drawing.

Difference Between Blocks and Wblocks

Ideally, you are sold on the importance of blocks as we go through the specifics. The differences between blocks and wblocks (write blocks) are blurred; they are alike in many ways. The key difference is that blocks are *internal* to the file you are working on and wblocks are *external*. This means that blocks are saved inside whatever file you are currently working on. As such, their main use is for grouping things together and reuse in that particular AutoCAD drawing. While they can certainly be moved between files, via the clipboard Copy/Paste tool, this is not really what they are intended for.

Wblocks are intended for creating symbol libraries. Wblocks are saved externally, anywhere you want to put them, and become independent, stand-alone AutoCAD drawings, which you can open and work on if needed. Both types of blocks can, of course, be easily inserted back into drawings, and we cover that procedure in detail.

Creating a Block

To create a block you first need something from which to make a block. Once you draw something, only three things need to be done:

1. Select the object(s).
2. Give the block a name.
3. Select a base point.

Let us try this out. Draw a chair, as shown in Fig. 7.1. It is made up of three rectangles and three lines. Use accuracy tools. We need to now group these lines all together into one object.

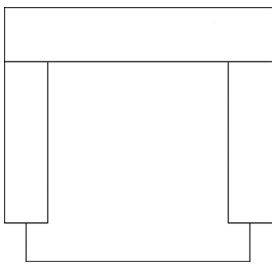
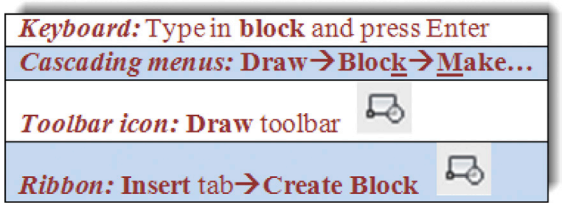


FIGURE 7.1 Basic chair symbol.



Start the block command via any of the preceding methods. The Block Definition dialog box appears, as seen in Fig. 7.2.

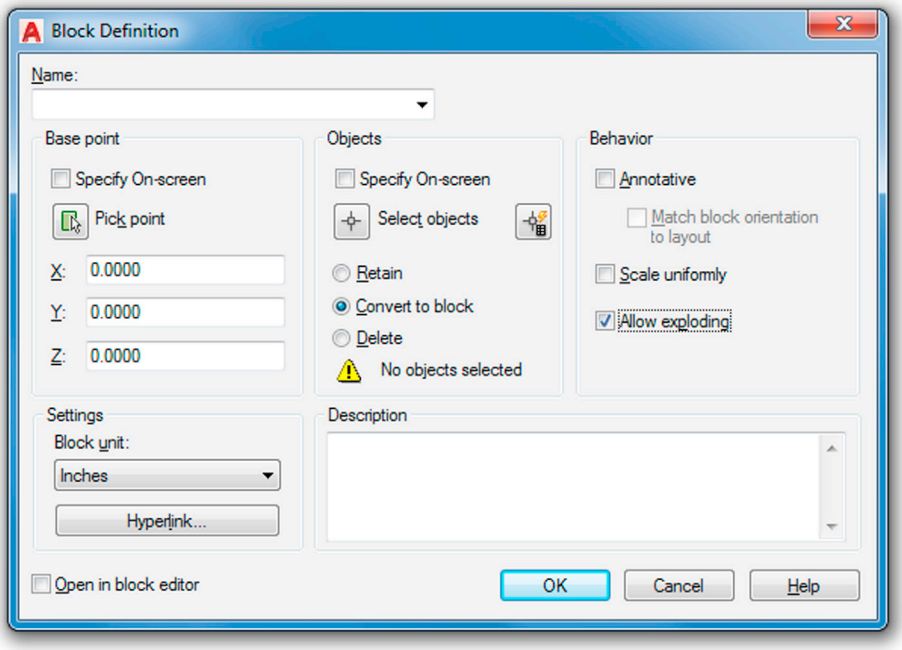


FIGURE 7.2 Block Definition dialog box.

Step 1. Press the Select objects button in the middle column. Select the entire chair and press Enter.
Step 2. Enter a name for this block in the top left field under Name.
Step 3. Pick a base point with the Pick point button at the upper left. Select a corner or midpoint somewhere on the object.

The significance of this becomes more apparent when we insert the object back into the file.

There are a few other items in the dialog box under Settings, Behavior, and Description, of course, but they are not usually needed (although feel free to enter a description). Annotative is covered in advanced chapters. The one setting of importance, however, is the Retain, Convert to block, and Delete choices under the Select objects button in the middle column. They are explained next and appear with almost similar wording in wblock as well.

Retain

This means that the block is created and stored in memory, but the original object (the chair you are looking at) remains in separate pieces, useful if you wish to make several blocks of the chair, one after the other, each slightly different yet based on the same original.

Convert to Block

This option is what you are likely to use the most. It creates the block and stores it in memory and also makes a block out of the original object. Use it when you are sure one block is all you need to make out of this one object and you have a need for it right away.

Delete

Here, the block is created and stored in memory, but the original is deleted from the drawing. This is useful when you are making a block for future use but have no need for it now.

The common theme with all three of these choices is that, whichever way you use it, the block is created and saved to memory. The only real choice is what to do with the original.

Finally, you are ready to click on OK. The Block Definition dialog box should look like Fig. 7.3. Notice the small preview window in the upper middle of the dialog box.

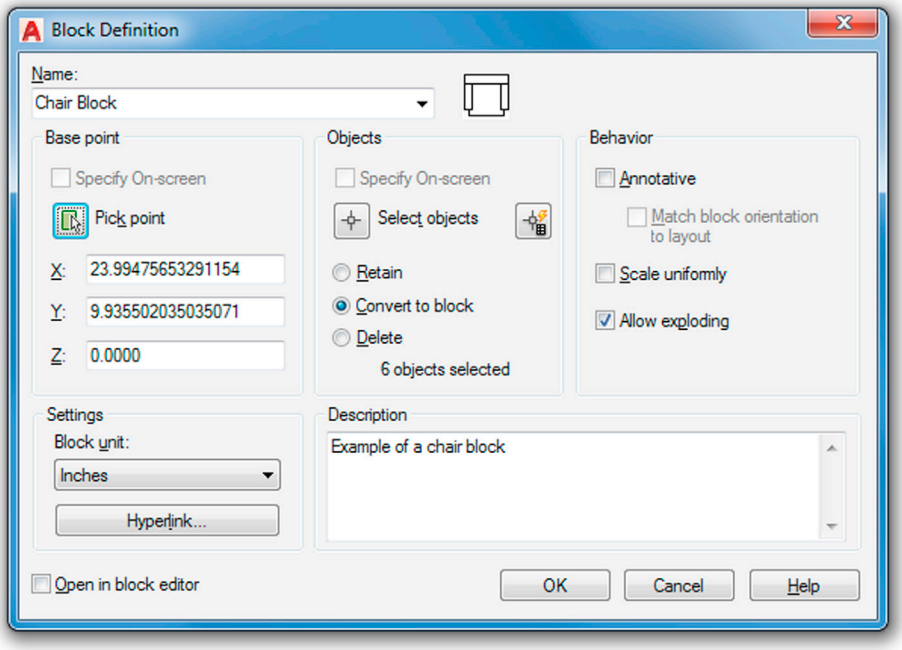


FIGURE 7.3 Completed block definition.

The chair is now a block, which can be proven by clicking on it once with the mouse. Notice that only one blue grip point appears (wherever you picked the base point; see Fig. 7.4). If you now erase it, the entire chair will disappear. Go ahead and erase it; we need to do this to introduce two new commands before we discuss wblocks.

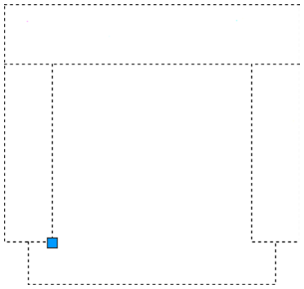
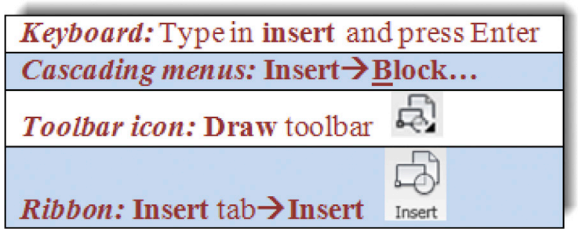


FIGURE 7.4 Chair as a block.

7.2 INSERT

As mentioned before, the process of making the block adds it into the memory of the file you have open. Then, you can insert it back into the drawing when needed.



Start up the insert command via any of the preceding methods and the Insert dialog box appears, as seen in Fig. 7.5. Notice the chair is cued up and ready to go, as it is the only block created so far. A large preview of the chair appears on the right, increased in size over AutoCAD 2016's Insert dialog box preview.

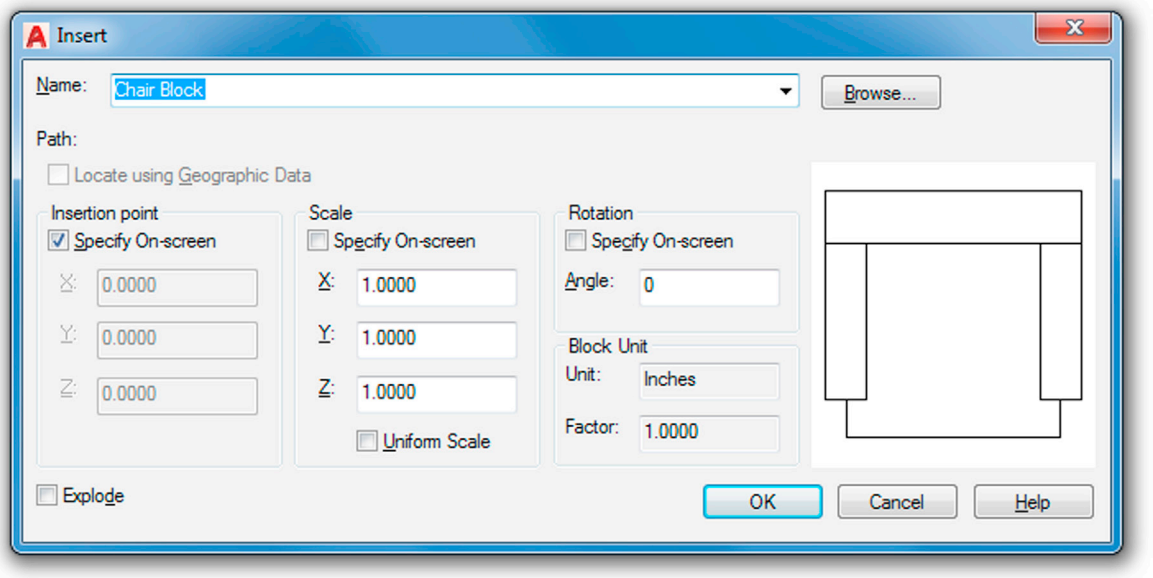


FIGURE 7.5 Insert dialog box.

Look over the various options, such as Insertion point, Scale, Rotation, and Block Unit, but at this point, you need not do anything extra. Go ahead and click on OK; the chair appears attached at the insertion point to the mouse. The point of attachment was specified when you made the block. Click anywhere on the screen and the block becomes part of the drawing. This process can be repeated as often as necessary as long as the block remains in memory. Note that you can always specify a precise point of insertion for the block by typing in the coordinates (e.g., `0,0,0`).

There is something important to remember concerning redefining blocks. If a block is saved under a certain name and another block is created and saved under the same name, the first block is redefined, a sort of overwriting process. Sometimes, this is bad, as you are destroying the old block. Sometimes, however, this is desirable, such as when you are looking to update the appearance of the block. When there are many blocks in the drawing under the same name, all are updated in this manner.

7.3 PURGE

Between introduction of the block and wblock concepts is a good time to take an intermission of sorts and discuss a command called *purge*. Purge is a very useful tool to "clean up" your drawing of unwanted and often unseen geometry and data. It is especially useful when it comes to blocks and wblocks, hence its inclusion here.

Conceptually, here is the idea. A complex AutoCAD drawing often grows in size and becomes weighed down, similar to an iceberg floating in the ocean. Much like the mountain of ice, mostly hidden under the surface, what you see on screen may be only a small fraction of what actually is present in the file.

The reason why becomes apparent when you think about the drafting process for a moment. If you create 50 layers for anticipated future use but end up needing only 40 of them, the other 10 are uselessly hanging out in your layer dialog box. Same thing with unused fonts or linetypes (remember loading *all* of them but using only a few). Layers, linetypes, and fonts really take up little space, but with blocks and wblocks, it is a whole different story.

Wblocks are miniature drawings all to themselves, and while many are just simple doors or window symbols, others can be quite large, such as entire furniture sets or appliances in an architectural layout. Worse yet, these blocks rarely come alone; they bring friends, lots of them. Complex multistory sprawling building layouts feature not dozens but *hundreds* of blocks. They all take up room in a file, ballooning it up to a rather huge size.

If all these blocks are needed, then fine; this is a necessary evil. But if not—say, a type of chair was deleted from the specs—they need to be removed. Erasing them all is easy enough, but are you done? Are they all gone for good? Unfortunately, no. They may not be visible, but they are still there weighing down your file, slowing down computer performance and just about everything else. This is where purge comes into play.

Purge gets rid of items that are not actually used in the drawing file and permanently deletes them. You can use the command to selectively purge items or purge everything all at once. The two main points to remember are

- Purge does not get rid of objects that are visible and present in the drawing—it is not an erase command.
- Purge is permanent (unless you immediately undo it), so be careful of what you purge. If you anticipate eventually needing something, do not purge it out.

Let us give the command a try.

Keyboard: Type in **purge** and press Enter
Cascading menus: **File→Drawing Utilities→Purge...**
Toolbar icon: **none**
Ribbon: **none**

First of all, erase the chair from your screen. Then, start up the purge command via one of the two methods shown in the command matrix, and the dialog box in Fig. 7.6 appears.

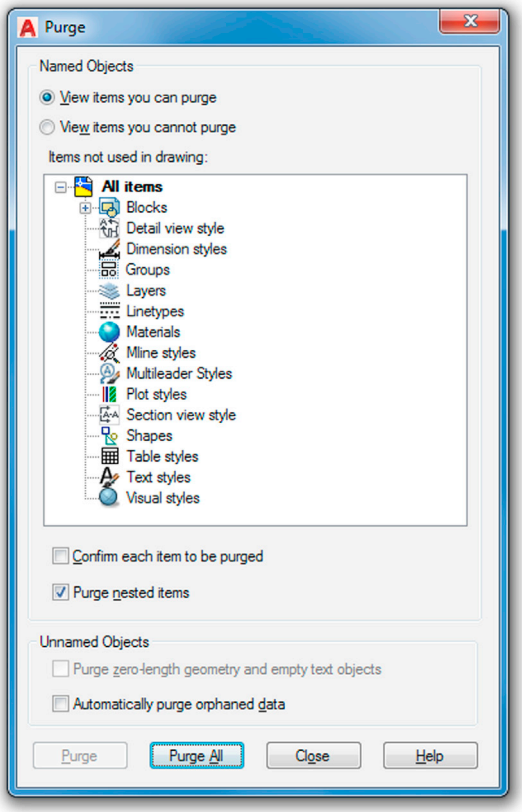


FIGURE 7.6 Purge dialog box.

As you can see from looking at the list of items in the dialog box, quite a few items are eligible for purging. Some were already mentioned, like Layers and Linetypes, while others you may not have heard of, like Materials and Shapes. Either way, what is important is whether or not the item has a plus sign next to it. If it does, you can click on the plus sign and expand the folder to see what specific items are in those general categories. In our case, you see the chair block under the Blocks category. You may or may not have a few other plus signs. If you do, then whatever may be there will just get purged along with the chair block.

The actual process of purging is easy. You can either select each item one at a time (usually done only when you want to selectively purge) or you can click on Purge All at the bottom. Be sure the Purge nested items box is checked and the one above it is not, as seen in Fig. 7.6. When all the items are purged, there are no more plus signs and the buttons gray out as well. Press Close after you are done and that is it.

Experienced users purge often and keep little useless junk in the file unless absolutely necessary. Purging is certainly something that should be done prior to saving the file for the day, emailing it, or archiving it for storage and record keeping. Be careful though, and do not get rid of needed items. If in doubt, do not purge the items.

7.4 WBLOCKS

Pretty much everything said about blocks applies to wblocks as well, and so this section is short, as you have seen this before. The central idea to keep in mind is that a wblock is external, and as such, you need to tell AutoCAD where to put the wblock as you make it. Let us try this out. Clear your screen and make sure all items from the previous exercise are purged. Now go ahead and draw another furniture object, perhaps a sofa this time.

The only way to start up this command is via typing in `wblock` and pressing Enter. Once you do that, the dialog box in Fig. 7.7 appears.

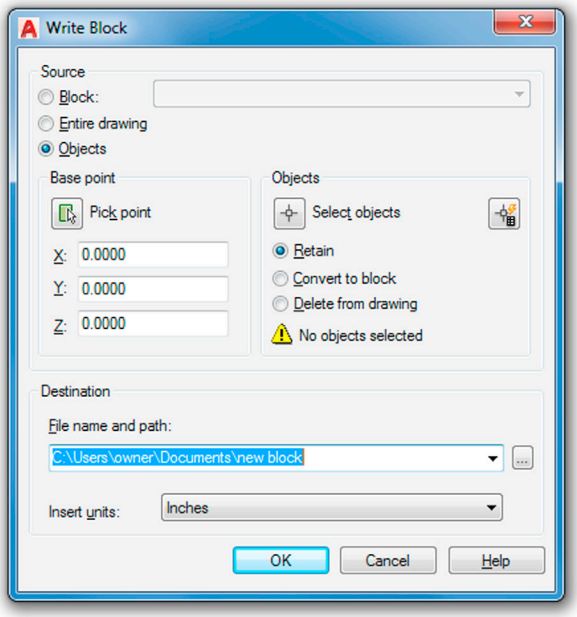


FIGURE 7.7 Write Block dialog box.

For the screen shot in this figure, a specific path has not been selected, but you can do that by clicking on the button with the three dots, …, to the right of File name and path, and browsing around until you find where you want the file to go. Be sure to also change the name of the file at the end of the string, so it does not just say new block. Then, select the sofa and pick one of the three options to Retain, Convert, or Delete the original. When done, press on OK and the wblock is created and dropped off where you indicated in the path.

Inserting Wblocks

Go ahead and insert your sofa back in using the insert command. The procedure is essentially the same, except that now you have to browse for the file. The Browse… button is just to the right of the Name: field. Look around for your block and when you find it, click on it and press OK. It inserts just as with the previous regular block command.

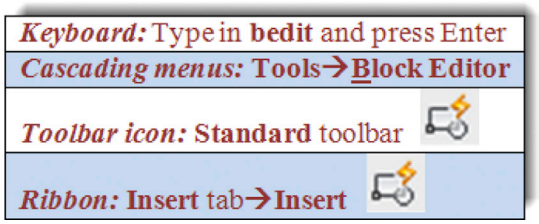
7.5 DYNAMIC BLOCKS

Dynamic blocks were introduced back in AutoCAD 2006. Like many new features, they are not a radically new concept. Rather, they are an evolution of the regular block and wblock. The idea here is to be able to modify the block in response to different design conditions. A simple example is of a door. If you insert a block of a 3′ door, rotated at 45°, and realize that you need a 3′-6″ door, rotated to a closed position, you can change that in place without redefining the block or inserting another one. You can even go a step further and, using the visibility option, change the block to something else; in other words, not just its orientation and size, but what it is completely, akin to having many blocks in one.

So what we are essentially doing here is adding a level of intelligence and automation to the standard block and wblock concept. We use this for scaling, stretching, mirroring, and otherwise modifying the block's sizing and orientation. We also use this intelligence to add menus to a block, and it allows you to select many different versions of the block as dictated by your design needs. AutoCAD provides options for almost every conceivable design intent, so let us take a look at this from the basics on up by first considering very simple parameter changes.

Actions and Parameters

To create a generic dynamic block, you need to modify a regular block by adding parameters and actions. Three things make up a dynamic block: the *geometry* itself, at least one *parameter*, and at least one *action* associated with that parameter. A parameter defines the features of the geometry, and the action defines the modification you would like to act on that parameter. The result is a block that has one or more special grip points, each defined with a certain modifying property (e.g., stretch, rotate, scale). The block can then be modified in place as needed. For a simple example, recreate or retrieve our existing chair block, and follow the procedure outlined next.



Regardless of which method you use to start the command (you can also just double-click on the block itself), you see the dialog box in Fig. 7.8. Select the block you want to work with, which in this case is the only one available.

After you select Chair Block, you are taken to the main editing screen, called a *Block Editor* (recognizable, unless changed via Options, by its medium gray background), shown in Fig. 7.9. This is where the settings are applied.

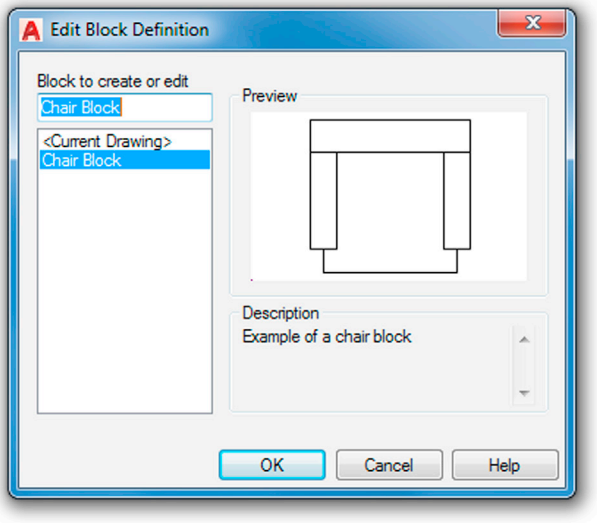


FIGURE 7.8 Edit Block Definition.

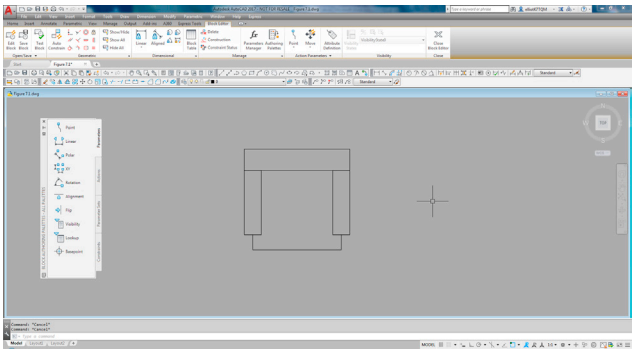


FIGURE 7.9 Dynamic Block Editor.

A Block Authoring palette also appears. Sets of parameters and actions are available from this palette, and the block can be set up to have any of those. When done, press Close Block Editor (top right of screen) to return to the regular drawing space. So what can we do here? Well, let us add two actions and parameters to the chair that give it some scaling and rotational abilities.

Step 1. Open the Parameter tab on the palette and pick the Linear parameter. That, after all, is the way scale works, by linearly scaling up an object. First, pick the upper left corner start point, then pick the upper right corner endpoint of the chair (like adding a horizontal dimension). The result is shown in Fig. 7.10. Note that a yellow exclamation point is seen as a reminder that no action has yet been added to this new parameter.

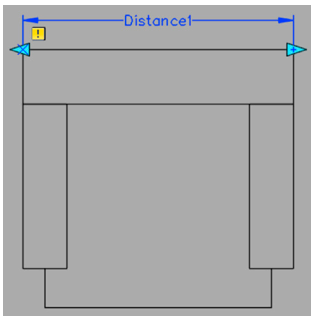


FIGURE 7.10 Dynamic Block, Step 1.

Step 2. While you are there, add a rotation by running through the required Base Point (lower left), Radius (any value), and Angle (30°) prompts. The result is shown in Fig. 7.11.

Step 3. Now, you can add actions associated with those parameters. Pick the Actions tab and pick Scale. Then, select the Parameter and the Objects (the chair), as prompted. Repeat the same steps for the rotation. The result is shown in Fig. 7.12. Notice the new (faded) scale and rotate symbols, and the disappearance of the exclamation point.

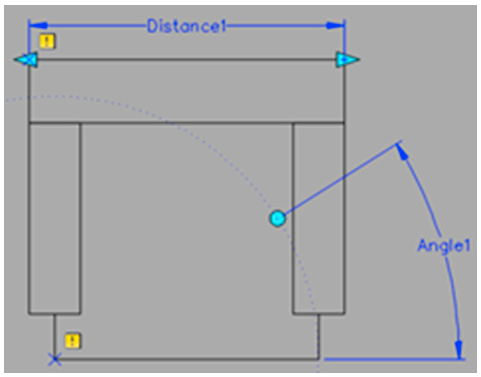


FIGURE 7.11 Dynamic Block, Step 2.

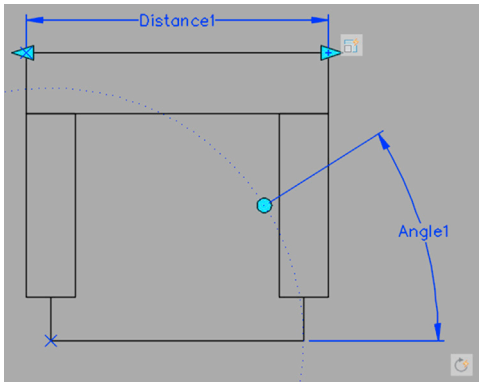


FIGURE 7.12 Dynamic Block, Step 3.

Close the Block Editor (button at upper right of the Ribbon) and you are prompted to save your work, as seen in Fig. 7.13.

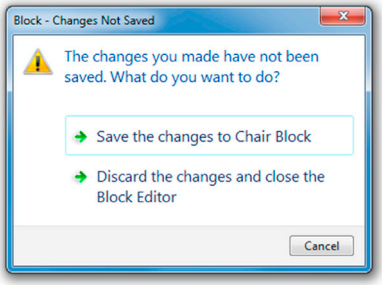


FIGURE 7.13 Save changes prompt.

Finally, you are back where you started. Click on the new block and notice the dynamic grips, as seen in Fig. 7.14. You can use these to change the size and rotation of the block, as seen in Fig. 7.15 with scaling.

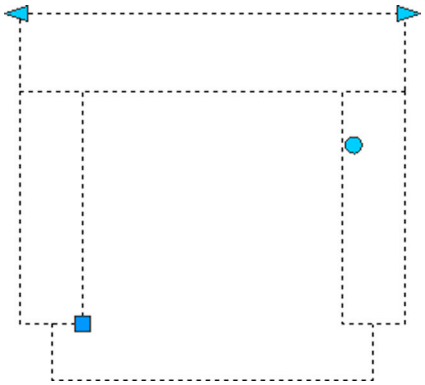


FIGURE 7.14 Dynamic Block grips.

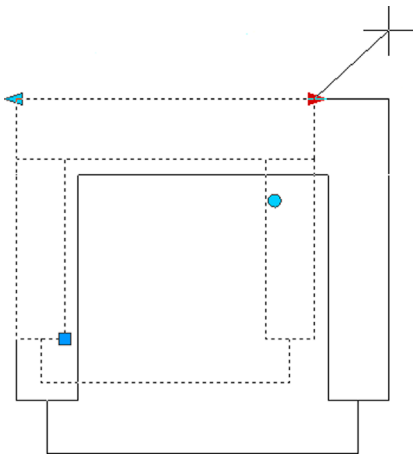


FIGURE 7.15 Dynamic Block modification.

This procedure features the absolute basics of creating a dynamic block. Let us try this again, with a door this time, and discuss a few more features. Create the door shown in Fig. 7.16, and call the block `Door_36`. Be sure to add in the door swing, the hatching, and the door jams. The dimensions are only for your reference.

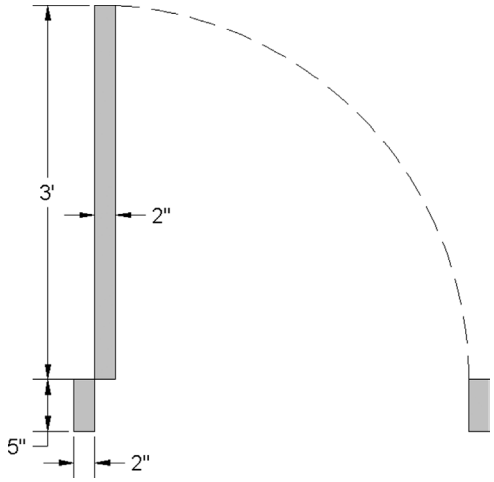


FIGURE 7.16 36″ Door.

Go into dynamic block edit mode and let us add the Flip action. This allows you to reverse the opening of the door after it has been inserted into place.

Step 1. Add a horizontal reference line across the bottom from the left side of the door to the far right end of the door swing.
Step 2. Add the Flip parameter using the midpoint of the reference line and tracing upward using Ortho. Position the `Flip state1` somewhere near the line. You can rename this text string to `Flip Left-Right` or something similarly descriptive if you wish, using the change properties palette.
Step 3. Add the flip action by selecting the icon, then the parameter, then all the objects. Fig. 7.17 shows what you should have on your screen. You can go ahead and erase the horizontal guide line at this point.

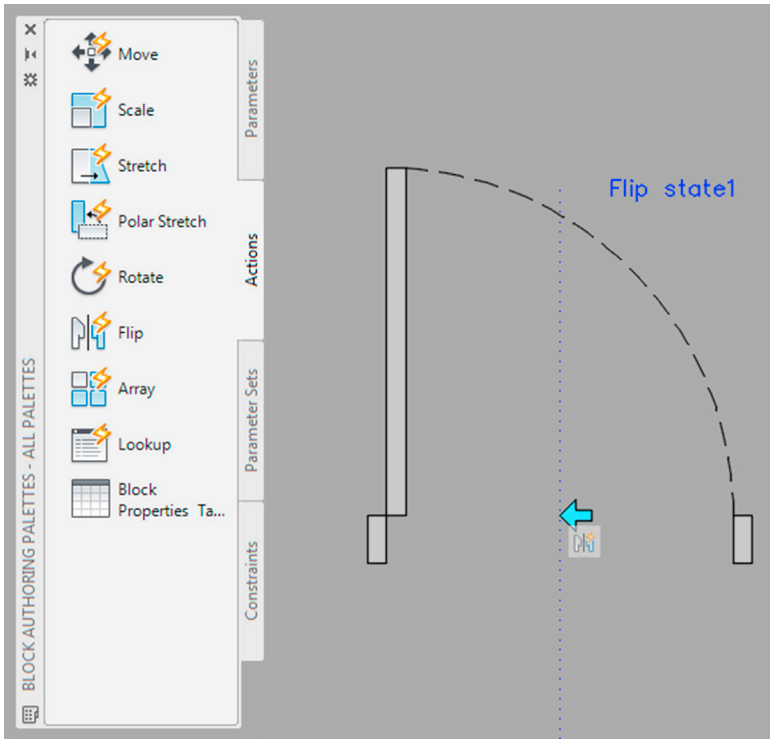


FIGURE 7.17 Flip parameter and action.

Step 4. As a final step, press the Test Block button at the upper left of the Ribbon. This takes you into an intermediate area (still in block edit mode) where you can test out what you have done. In our case here, click on the door and you see a blue arrow pointing to the left at the bottom. Click on that and the door's swing flips around (Fig. 7.18).

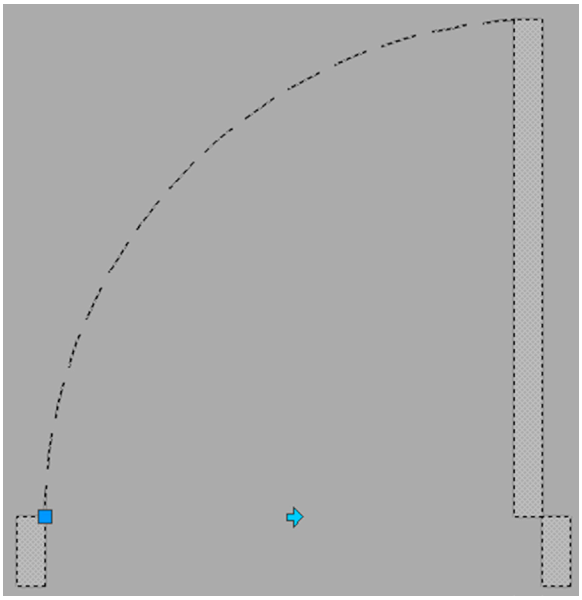


FIGURE 7.18 Flip executed.

You can also set up the flip so the door opens "downward" and not "inward," as it does now, of course. Your reference point would then be the midpoints of the two door jamb blocks. Try that as an exercise.

Another useful feature you can try is called *Value Set*. Add a linear parameter and action to the door (exactly like Distance 1 in Fig. 7.10). Then, while Distance 1 is selected, go to the Properties palette, Value Set category, and for the Dist type on the left side, select List on the right. Just below that for Dist value list, enter a few incremental values greater than `36″` (the current door size), such as `48″`, `60″`, and `72″`, into the Add Distance Value dialog box. After you complete this step and go to test the door by clicking on the scale grip, you see a series of tick marks appear and you can increase the sizing only incrementally, as opposed to arbitrarily. There is quite a bit more to dynamic blocks, of course, and you can combine many features to create new ones.

Visibility

The final item to cover with dynamic blocs is a very useful feature called *Visibility*. This option is a favorite among quite a few designers because it allows you to combine virtually unlimited variations of a block into one, then simply select what you want via a drop-down menu. It can take a while to set up the numerous blocks and set the visibility states, but the payoff in the end is significant.

Here is the basic idea. You make a block by combining a number of other blocks (or even nonblock geometry by itself) by putting them one on top of the other, usually with a common reference point. You then create named visibility states—with each state representing a version of the block you wish to see when you click that name. You then set all but the relevant block as invisible. The process gets repeated until you process all of the geometry in such a manner. The result is one large block that contains numerous sets of geometry, but only one set is visible at a time. To pick which is the visible one, you select from a drop-down menu, which appears when you click on the block. There, you see all of your visibility state names, as you click each one the relevant geometry appears and all others disappear. Let us give this a try with a very simple example.

Step 1. Create five shapes and make a block out of each—a square, a circle, a hexagon, a triangle, and an ellipse—as seen in Fig. 7.19.

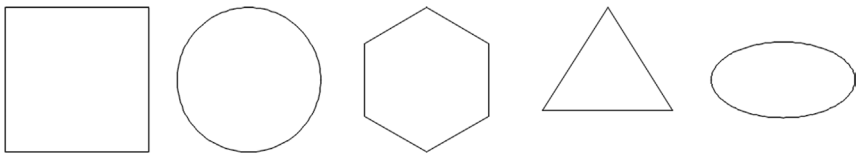


FIGURE 7.19 Basic block shapes.

Step 2. Put all the shapes on top of each other, centering them at the top. Make a block out of all of them calling it `Main_Block`. Then go into the dynamic block editing mode via `bedit` or just double clicking on `Main_Block`. You should see what is shown in Fig. 7.20.

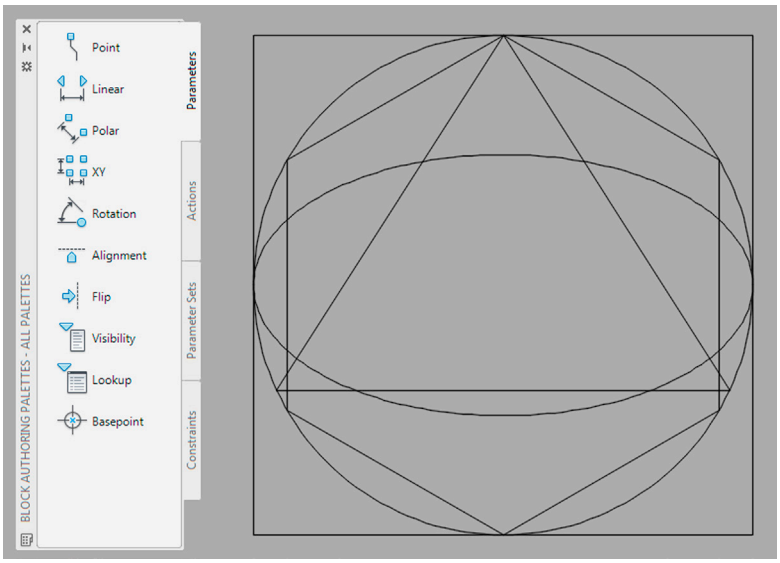


FIGURE 7.20 Combined shapes in `bedit`.

Step 3. Now add the visibility parameter (highlighted with a black rectangle in Fig. 7.21) by clicking on the Parameters tab. AutoCAD asks where to locate it. Select any convenient spot next to the shapes, as seen in Fig. 7.21.

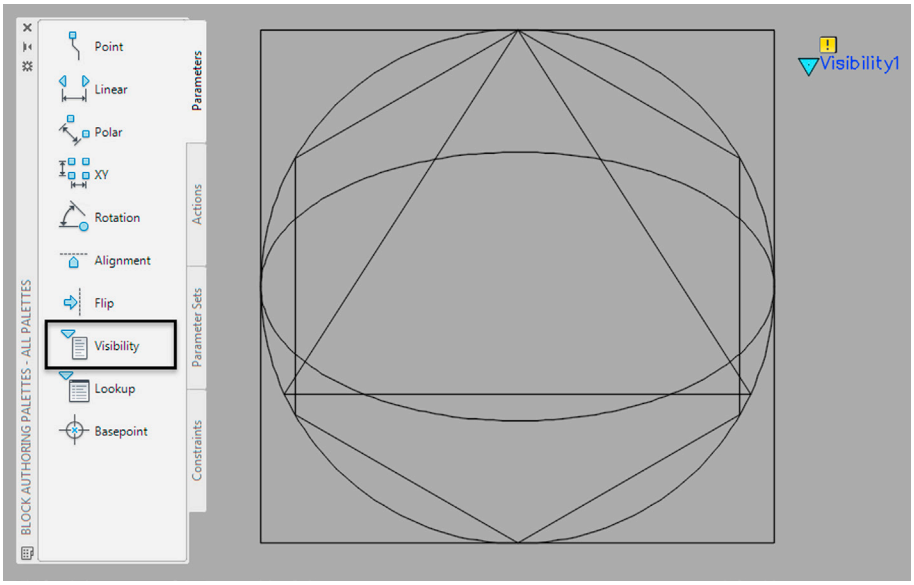


FIGURE 7.21 Visibility parameter added.

Step 4. Select the Visibility States button in the Ribbon's Block Editor tab, Visibility panel (Fig. 7.22). The Visibility States dialog box appears. Rename the existing state to `Square` and add the other four via the New… button (leaving any options as default in the dialog box). The result is seen in Fig. 7.23. Press OK.

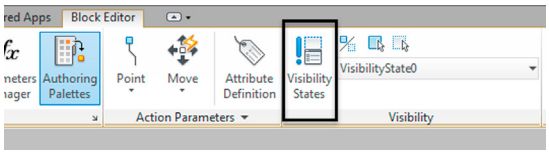


FIGURE 7.22 Visibility States button.

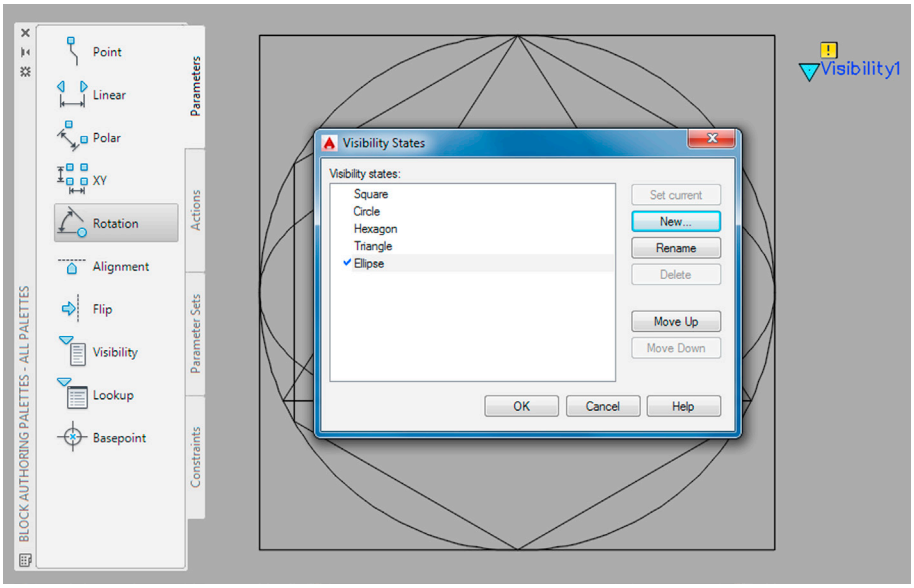


FIGURE 7.23 Visibility States completed.

Step 5. You now need to actually set the invisibility parameter of each visibility state. Go up to the Ribbon and find the drop-down menu just to the right of the Visibility States button (it can be seen in Fig. 7.22.). Drop it down and double-click on the `Square` state. It is now the current visibility state, and it is here that you need to make all the other shapes invisible. To do that, press the button just above the drop-down menu called *Make Invisible* (it looks like a faded square with an arrow and can also be seen in Fig. 7.22.). AutoCAD says `Select objects to hide:`. Go ahead and pick *all* of the shapes *except* for the square. The selected shapes all fade to a gray color, as seen in Fig. 7.24.

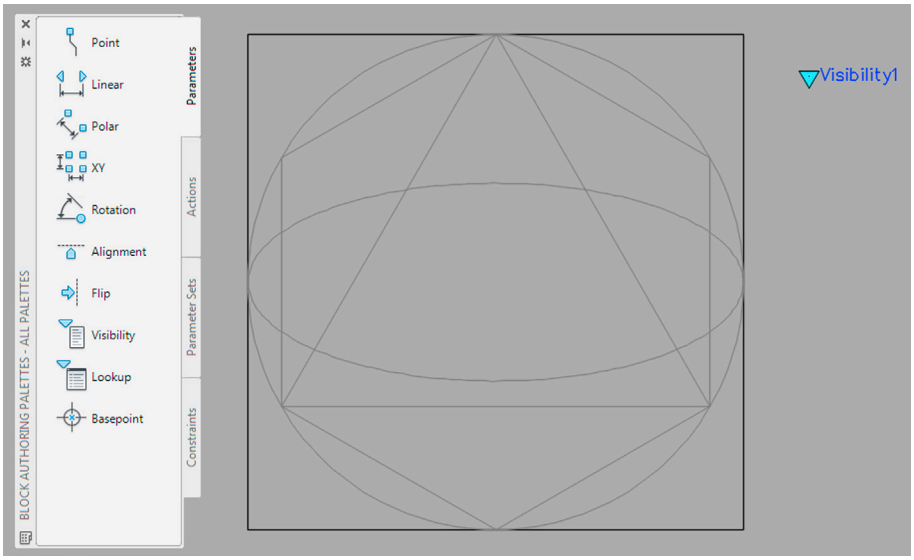


FIGURE 7.24 Square visibility state.

Step 6. Repeat this process for the other shapes by going again to the drop-down menu and double-clicking on the next shape on the list (the circle). All the shapes reappear, and now you have to use the Make Invisible button to select *all* of the shapes *except* the circle. This is why it was remarked that it may take a while if you have several dozen or more blocks embedded in one. The effort is well worth it, however.

Step 7. When you completed setting all of the visibility states, test them out via the Test Block button all the way on the left of the Ribbon's Block Editor tab. A new tab opens; go ahead and click on the block. Fig. 7.25 shows the first visibility state (square) with the drop-down menu revealing the rest of the choices. Click through each one to test each visibility state. When done, exit the test tab and save the dynamic block. It is now ready for use.

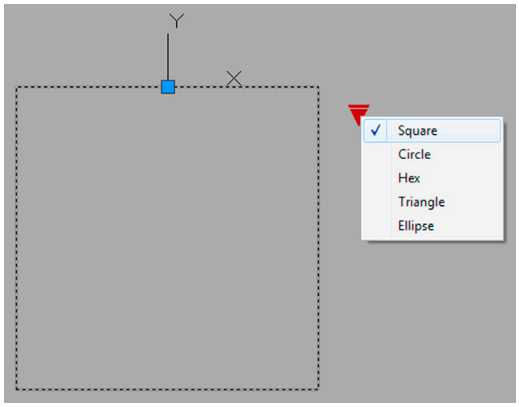


FIGURE 7.25 Visibility state menu.

This of course was a very basic example, but the process for more complex, realistic parts is identical. Keep them as blocks, stack them one on top of the other, create and set visibility states, save, and use. Here is a real-life example from the world of electrical (power) engineering. Fig. 7.26 is a top view of an actual relay switch commonly found at power substations.

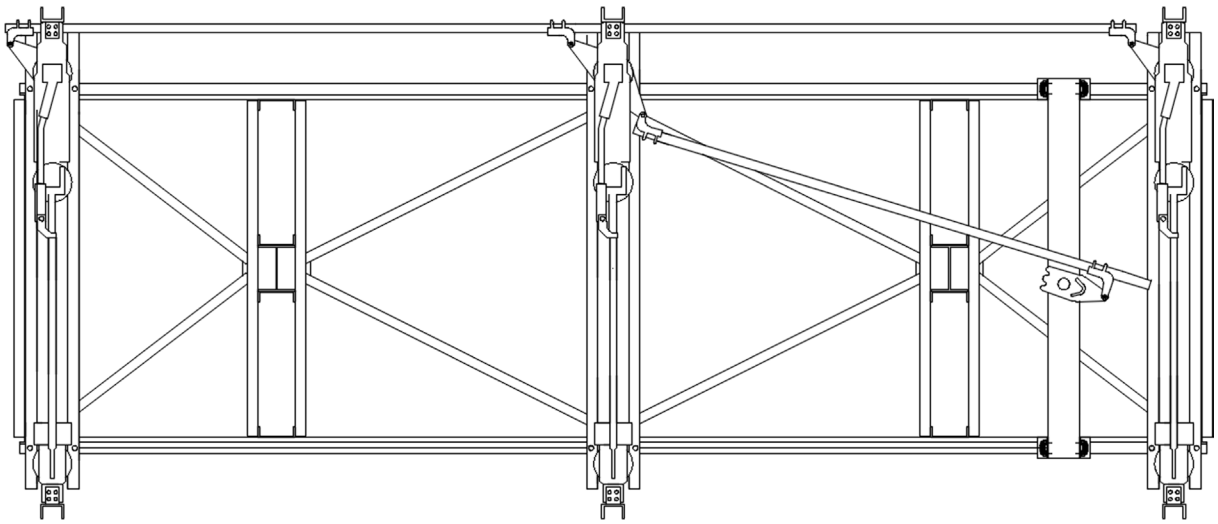


FIGURE 7.26 Relay switch.

As it happened in this case, a number of switches were at the disposal of the electrical engineer, all of them essentially the same in their basic function but very different in how they looked. It made sense to create one main block of them and just select what was needed for a particular job. That way the designer need not keep track of numerous blocks in a library. Such was the approach to drafting the switch in Fig. 7.26, as seen in the next image (Fig. 7.27).Notice the menu of available visibility state alternatives, with "Disconnect – Vertical Break SS20220 (B-42)" as the currently visible one.

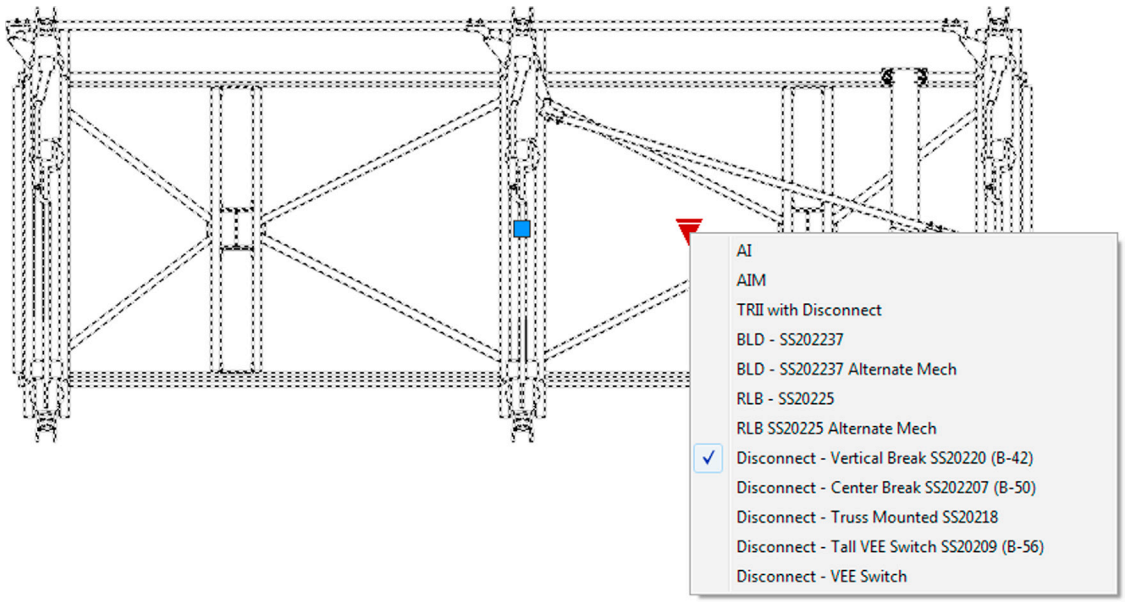


FIGURE 7.27 Available visibility states.

Figs. 7.28 and 7.29 show two other versions of the same switch. As you can see, they are complex blocks and quite different from one another but all sourced from the same file. Be sure to practice and get well-versed in setting up these types of dynamic blocks, visibility states are a very useful in actual design work!

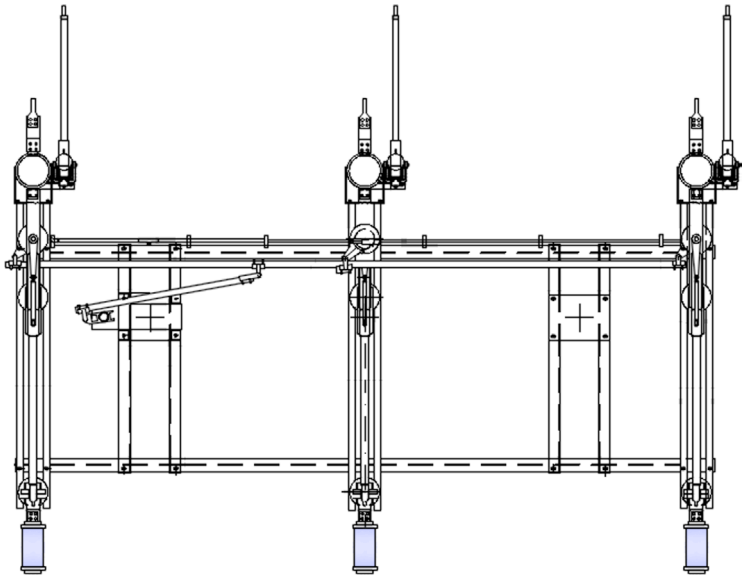


FIGURE 7.28 Alternate visibility state 1.

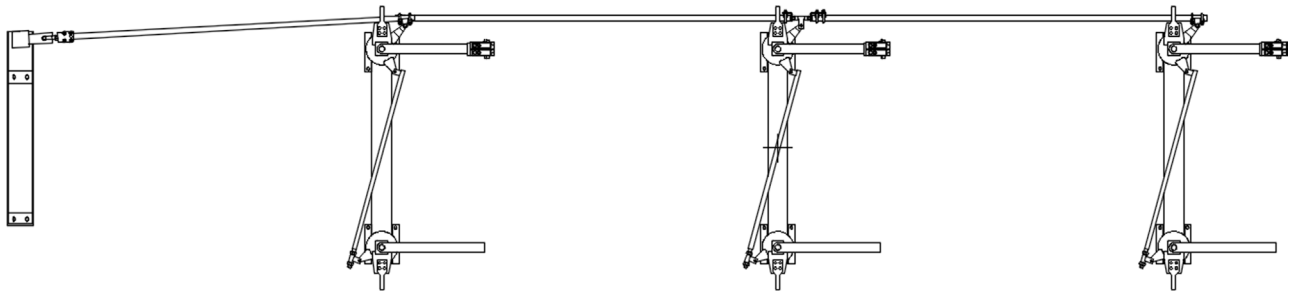


FIGURE 7.29 Alternate visibility state 2.

So, how do dynamic blocks fit into the big picture? Are they used as often as they could be? The results are mixed. Clearly, they are a big step forward for blocks and the built-in intelligence is very useful. However, one potential problem for long-time users is that, if they already have an extensive library of nondynamic blocks in use, there may not be any real incentive to rework them into dynamic blocks. This is an effort that sees more payoff if done from the beginning, but because dynamic blocks are relative latecomers (having appeared only in 2006), some may not want to upgrade their blocks and long-standing libraries if they have been in use and working fine for all these years. It is worth the effort however, as dynamic blocks, by allowing multiple variations on a single block, actually reduce the sizes and complexities of these libraries.

The bottom line for you as a student, however, is that it is a good skill to pick up, so try to explore further this interesting topic on your own. Chances are that, in the future, these will be the main types of blocks you will run into out there in the industry.

7.6 GROUPS

A group is a useful, but often overlooked, command from the block family. It is, perhaps, the easiest way to collect objects together and presents some advantages (but also some disadvantages) over the familiar block, as shown next.

Similar to a block, a group can

- Be given a name.
- Bind a collection of objects together.
- Be moved around, copied, and rotated.

Advantages of a group are these:

- A group can have objects added and subtracted from it.
- A group can have individual objects (and their properties) edited.

Disadvantage of a group are these:

- A group cannot be transferred between drawings (for "local" use only).
- An update to one group does not update other groups with the same name.

The group command went through some changes in the 2012 release of AutoCAD. You can now use the command line, Ribbon, or the Group toolbar (Fig. 7.30) to create and manage the groups, but the classic Object Grouping dialog box, which has been around forever, is still available as one of the options and is discussed later.

FIGURE 7.30 Group toolbar.

To try this command, first draw a random collection of objects, as seen in Fig. 7.31, then follow the steps shown.

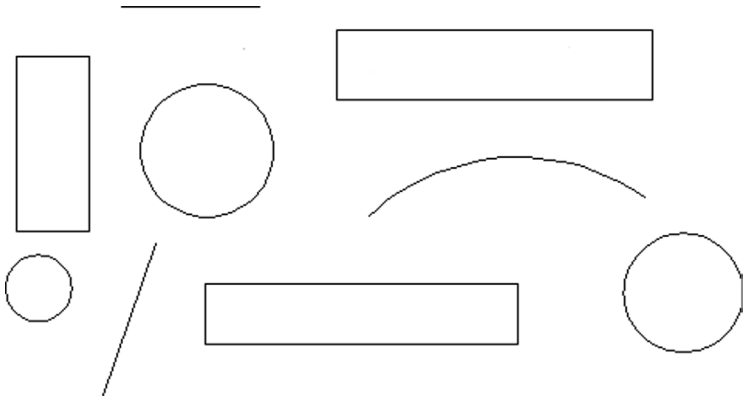


FIGURE 7.31 Random objects for grouping.

Step 1. Begin the group command via any of the preceding methods.
- AutoCAD says: `Select objects or [Name/Description]:`

Step 2. Press `n` for `Name`.
- AutoCAD says: `Enter a Group Name or [?]:`

Step 3. Type in a descriptive name: `SAMPLE_GROUP` in this case. Note that spaces in the name are not allowed.
- AutoCAD says: `Select objects or [Name/Description]:`

Step 4. Select all the objects.
- AutoCAD says: `Group "SAMPLE_GROUP" has been created.`

The group is now completed, and you can see results by clicking on any object (Fig. 7.32). Note how there is a rectangle around the objects and a single grip in the middle. You can now manipulate all the objects together as one unit.

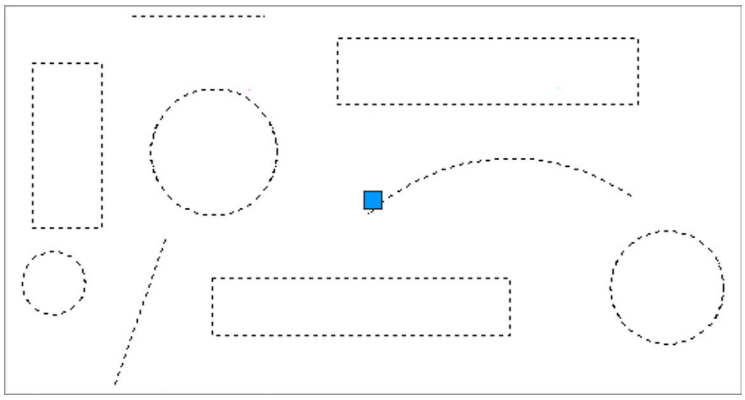


FIGURE 7.32 Completed group.

Let us take a look now at some of the features of a group and some additional functionality. One property you can take advantage of right away, and the main reason to use a group, is the ability to alter the properties of the objects in the group (or the shapes of the objects themselves). Double-click on any of the shapes and a palette called *QuickProperties* appears, as seen in Fig. 7.33.

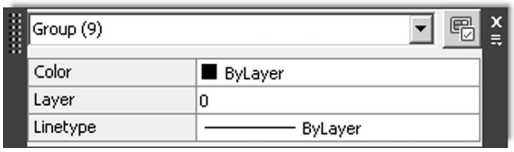
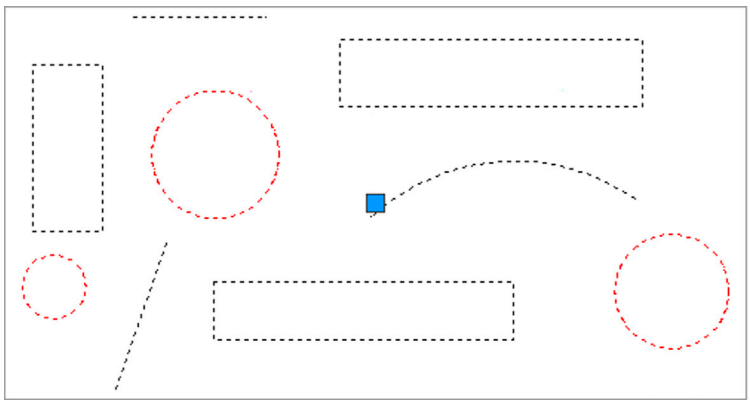


FIGURE 7.33 Group QuickProperties.

Drop down the top menu and you see a breakdown of the group, object by object. There are three rectangles (polyline), two lines, one arc, and three circles, for a total of nine objects. Select Circle (3) in the menu and the palette shows only the properties of the circles. Go ahead and change the color to red and observe the result (Fig. 7.34).



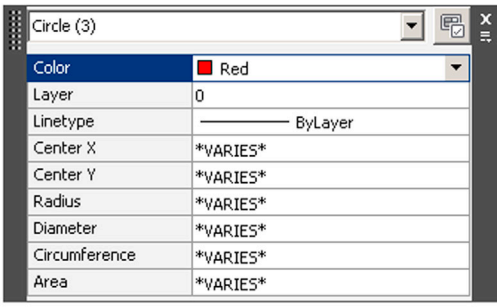


FIGURE 7.34 Color change to circles in group.

You can also work with individual objects in the group by clicking on the very last icon on the Group toolbar, called *Group Selection On/Off*. You can do the same by typing in `pickstyle` and entering `0` when prompted. Pickstyle is a system variable, and more is said about system variables in Chapter 15, Advanced Design and File Management Tools. For now, go ahead and modify any of the shapes, and when done, press the same icon or type in the same system variable (entering `1` this time) to get back to the grouping. Fig. 7.35 shows the editing in progress.

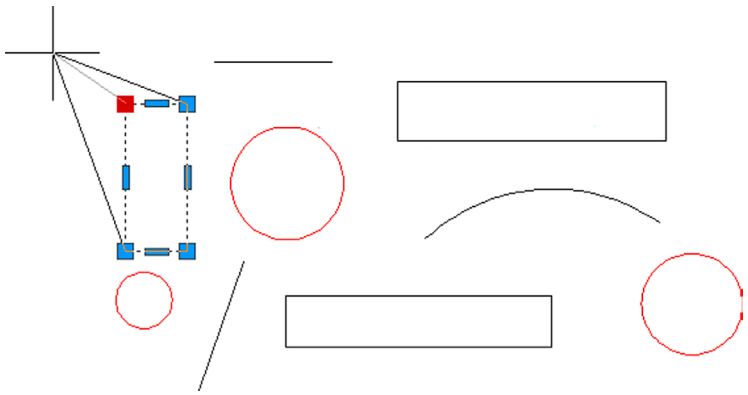


FIGURE 7.35 Group object editing.

Take a look at the other toolbar icon. The second one from left, called *Ungroup*, as you would expect, removes the grouping from the objects. Note that exploding the group does *not* delete the group itself but rather just explodes the objects inside of it—only the rectangles in this case.

The next icon to the right (with the ±), called *Group Edit*, allows you to add an object to or subtract an object from the group. Go ahead and try this on your own by deleting a rectangle via the Remove objects option and adding a new object (a polygon in this example) via the Add objects option. An example of the result is shown in Fig. 7.36.

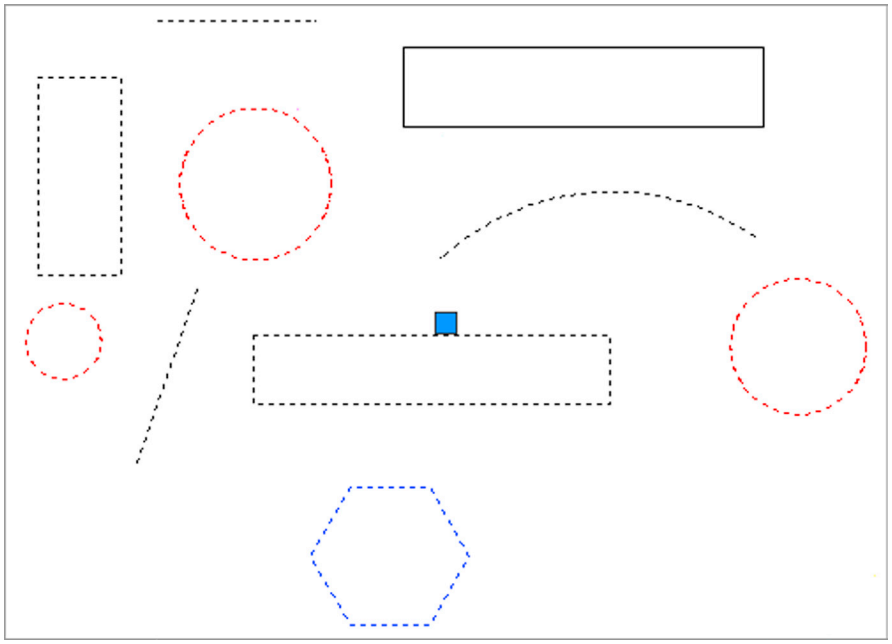


FIGURE 7.36 Object added (polygon) and another subtracted (rectangle).

Finally, it is necessary to mention the legacy Object Grouping dialog box (Fig. 7.37). It serves two purposes here. One is to provide you an alternate way of managing your groups. The second is to familiarize you with the process if you end up using an older version of AutoCAD, prior to the 2012 release, where this was the only way to create and manage groups.

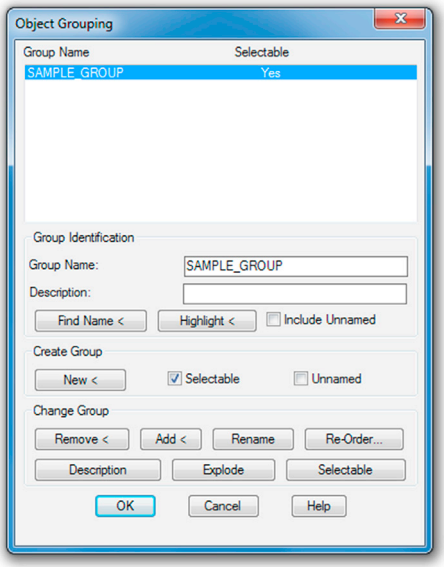


FIGURE 7.37 Object Grouping dialog box.

The Object Grouping dialog box can be called up by typing in `classicgroup` and pressing Enter. Note that, in this example, `SAMPLE_GROUP` is already created and is highlighted in the dialog box to reveal a multitude of options that should generally be familiar to you, based on the preceding discussion.

From this dialog box, you can create new groups, remove and add objects, and execute many other functions. Take some time to go over this dialog box on your own to complete your tour of the group command.

7.7 SUMMARY

You should understand and know how to use the following concepts and commands before moving on to Chapter 8, Polar, Rectangular, and Path Arrays:

- Block
  - Select objects
  - Name the block
  - Select a base point
- Wblock
  - Select objects
  - Select path and enter block name
  - Select a base point
- Options
  - Retain
  - Convert to block
  - Delete
- Insert command
- Purge command
- Dynamic blocks
  - Parameters
  - Actions
  - Visibility
- Groups

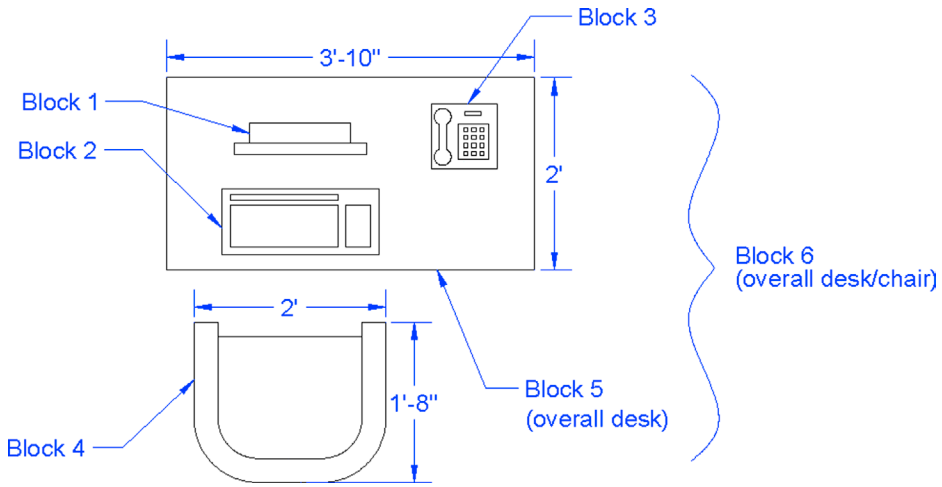
Review Questions

Answer the following based on what you learned in this chapter:

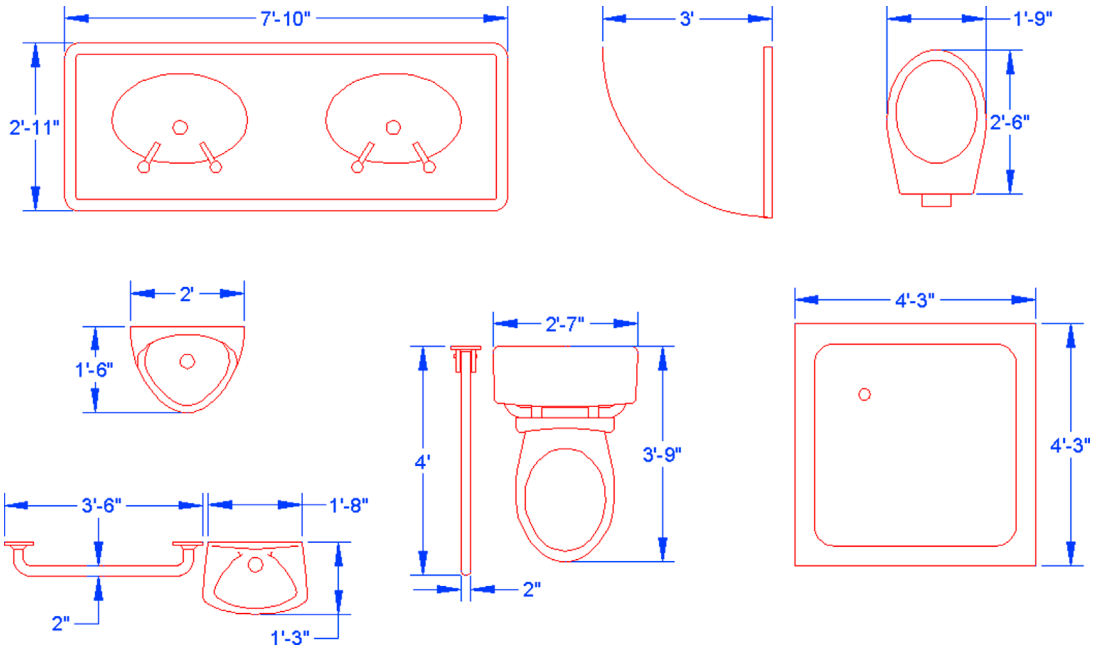
1. What are blocks and why are they useful?
2. What is the difference between blocks and wblocks?
3. What are the essential steps in making a block? A wblock?
4. How do you bring the block or wblock back into the drawing?
5. Explain the purpose of purge.
6. Explain the basic idea behind dynamic blocks. What steps are involved? What is visibility?
7. What are the advantages and disadvantages of using groups?

Exercises

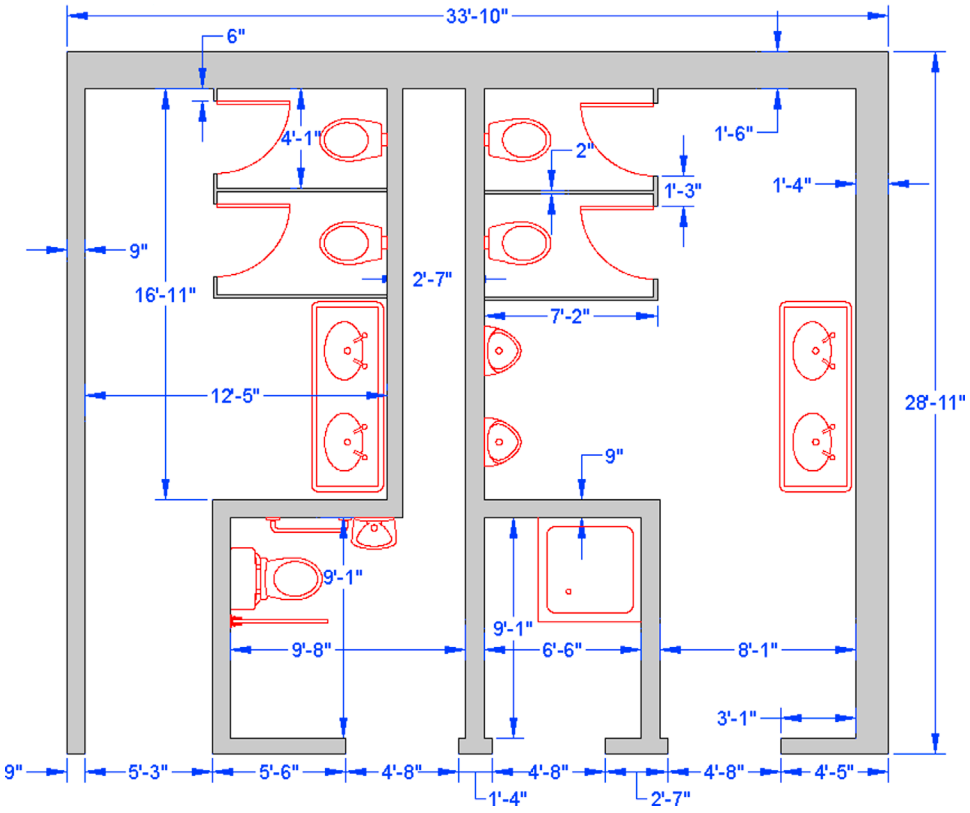
1. In a new file, create the following—architectural table, chair, keyboard, phone, and computer screen—and make a block out of the individual pieces as well as the overall "assembly." Nested blocks, such as these, are quite common in AutoCAD design. Insert the block back into the file. Only general dimensions are given, so you have to improvise for the individual pieces. Also, naming the blocks is up to you, just make them descriptive. (Difficulty level: Easy; Time to completion: 15–20 minutes.)



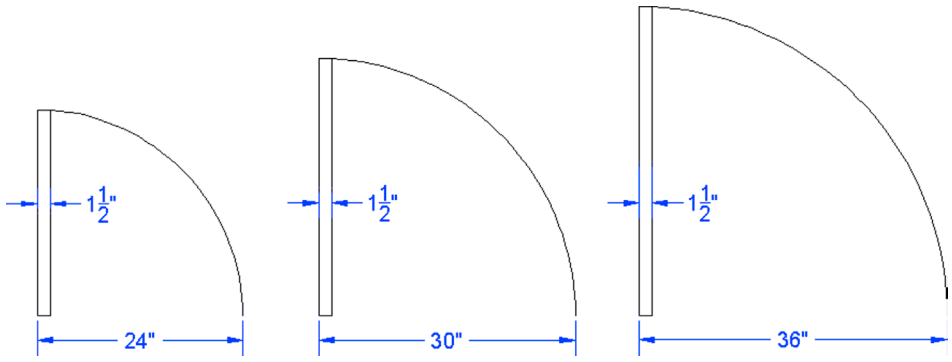
2. In a new file, create the following set of bathroom toilet and sink fixtures, doors, and other items. The overall dimensions are provided, but you need to create the details on your own. Be sure to use the appropriate layers. Create basic blocks out of the fixtures, and copy as needed. (Difficulty level: Easy/Moderate; Time to completion: 30–45 minutes.)



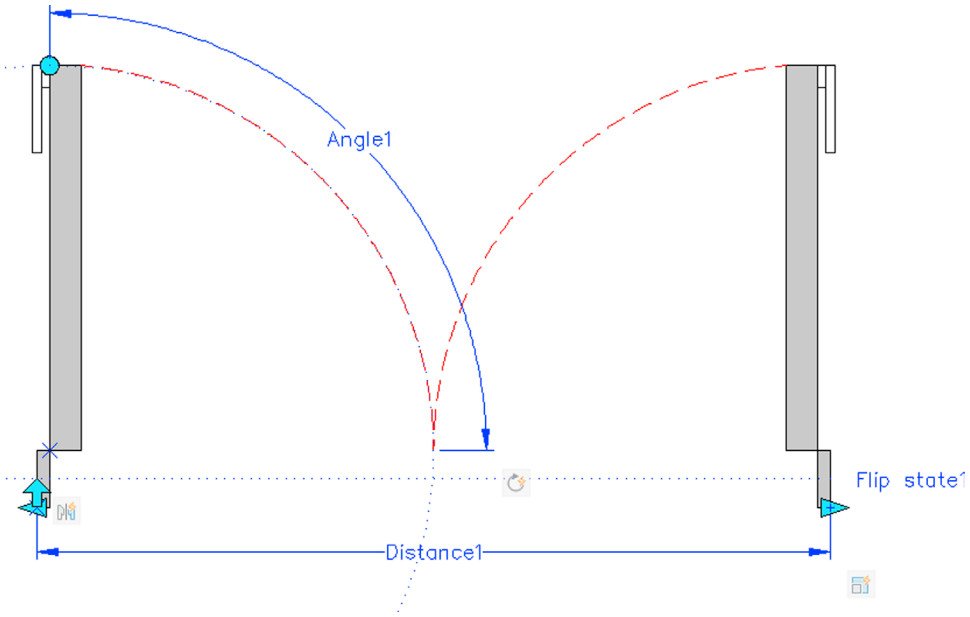
3. In a new file, create the floor plan depicted next. Insert the blocks from Exercise 2 into the floor plan as shown, adding the wall hatching as a last step. (Difficulty level: Moderate; Time to completion: 20–30 minutes.)



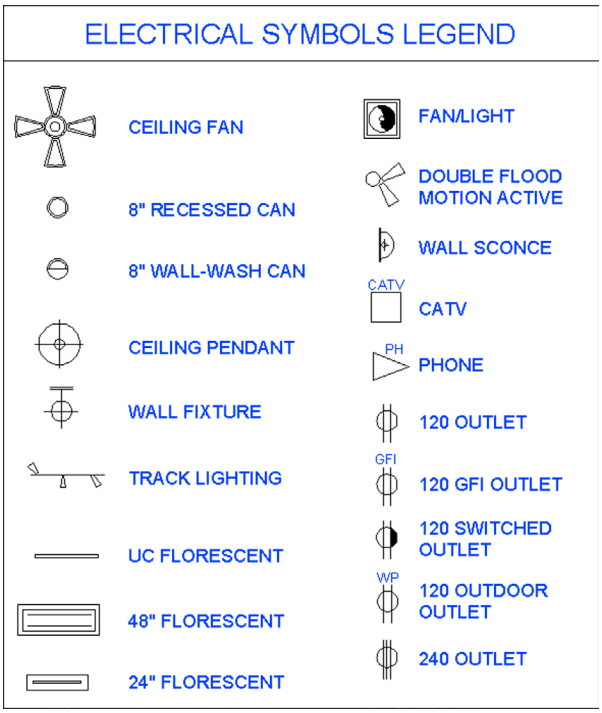
4. In a new file, create the following set of doors and make a wblock out of each one (`Door_24in`, `Door_30in`, `Door_36in`). Drop them all into a premade folder called `Doors`. Insert each wblock back into the file. When all three doors are inserted, explode them all and purge the file. Then, create dynamic blocks out of all three. Add the linear parameter followed by the scale action. (Difficulty level: Easy; Time to completion: 5–10 minutes per block set.)



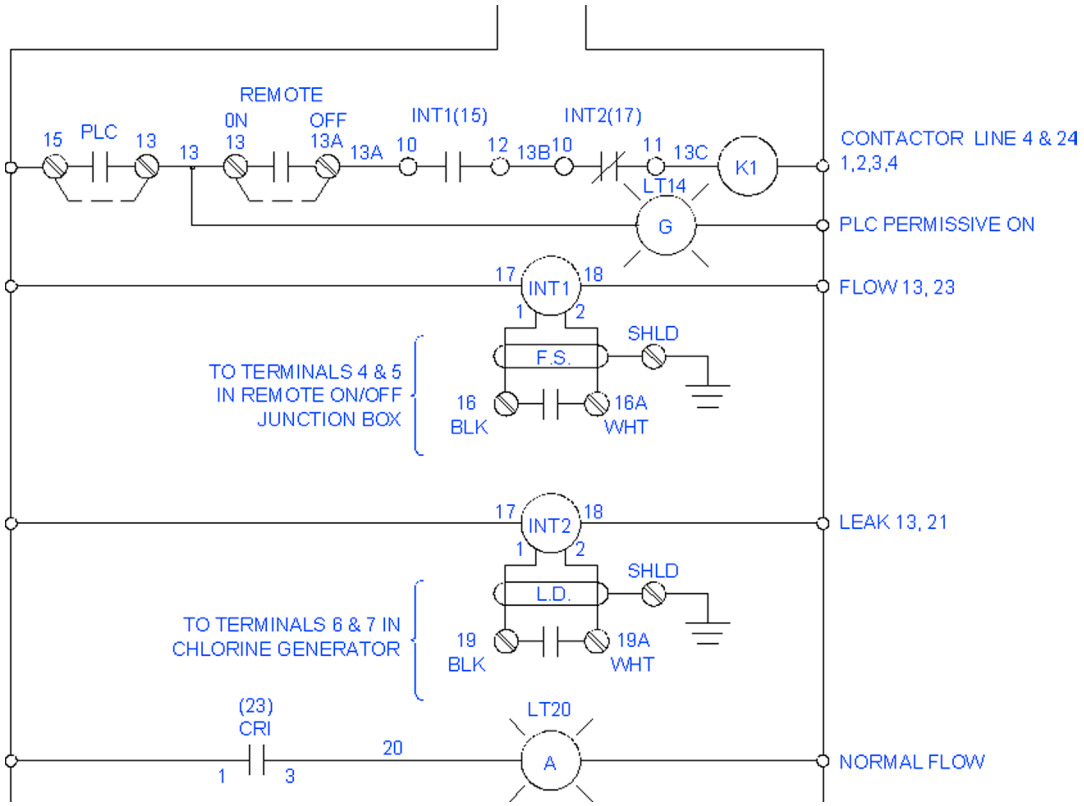
5. Create the following dynamic-block double door. The basic dimensions are 2″ × 24″, with the door handle sized off of that. Add in shading, arbitrary-sized door jams, and the hidden line door swing. Create a variety of parameters and actions, including Flip, Rotate, Scale, and others as you see fit. (Difficulty level: Easy; Time to completion: 15–20 minutes.)



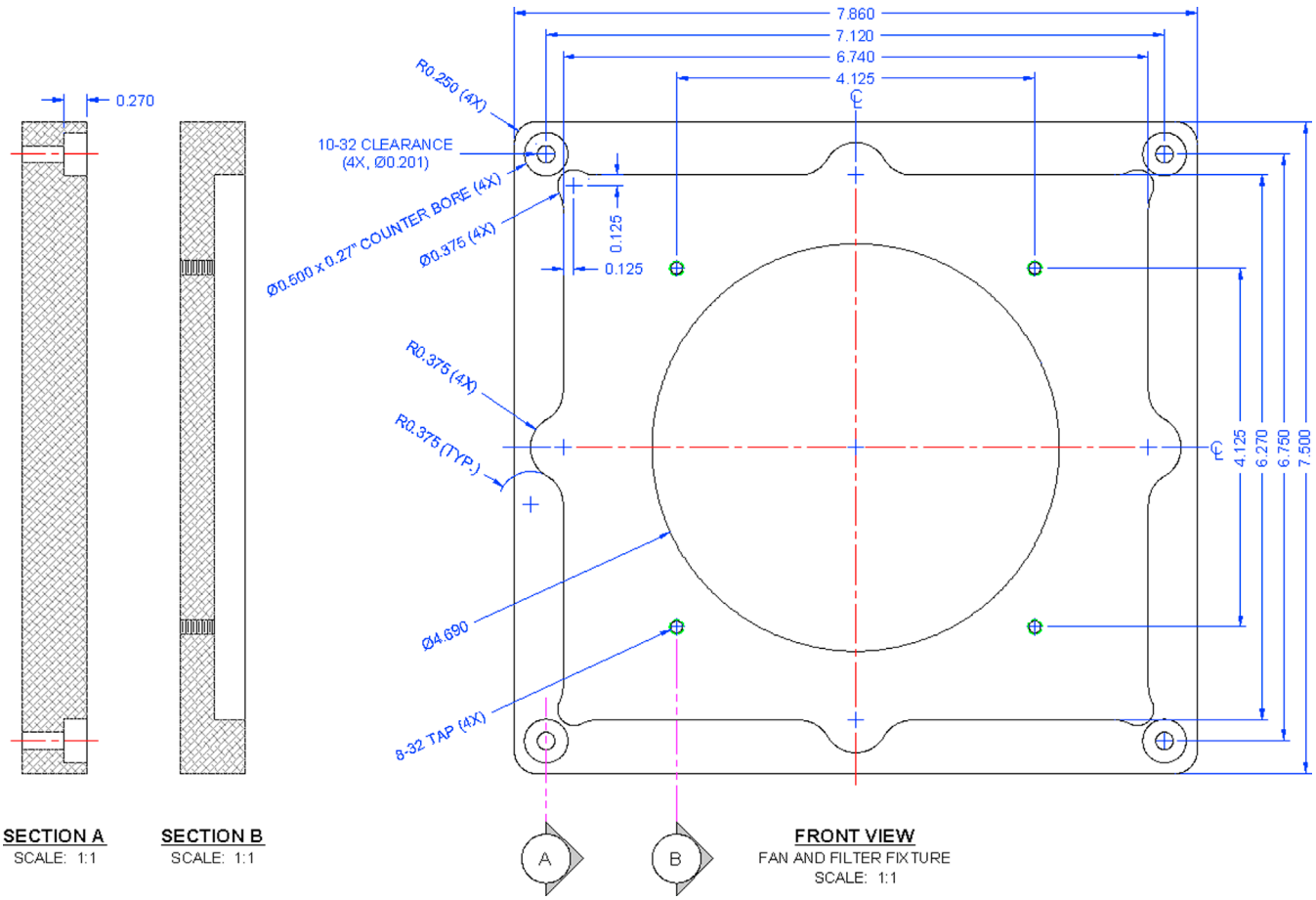
6. Electrical symbols are all usually blocks and electrical designers/draftspersons have an extensive library at their fingertips. You have already drawn a few electrical symbols in the previous chapter; now let us turn our attention to a few more. Draw the following symbol library and create a block out of each symbol. While you are drawing these symbols, try to memorize them as well. You will see them again in almost any architecture-related field. (Difficulty level: Moderate; Time to completion: 30–45 minutes.)



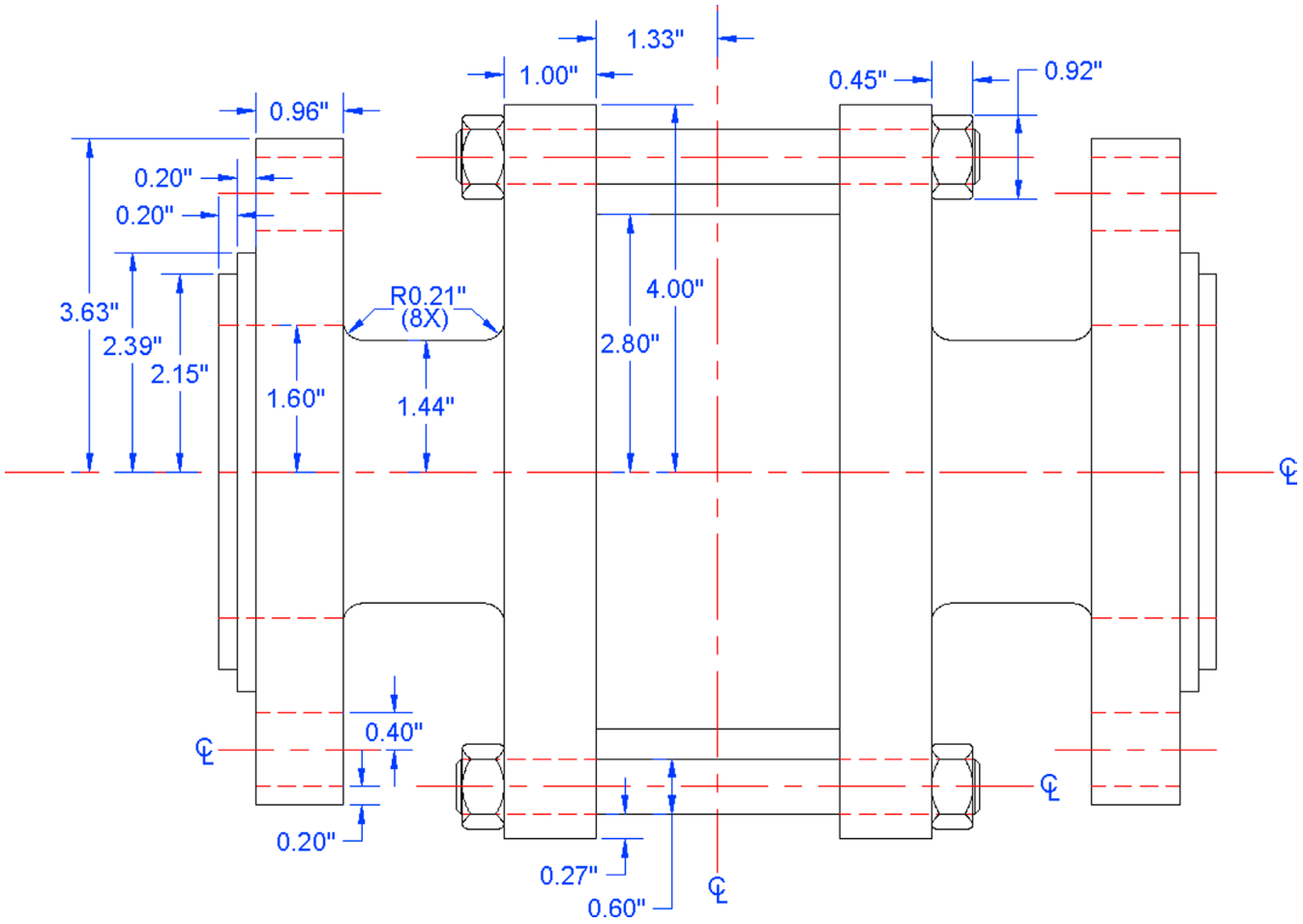
7. This exercise is another example of electrical schematics. Draft all the linework and add text as shown. Like most schematics, the drawing is not to any scale, but accuracy must still be maintained as far as connecting lines and other shapes, as well as text placement. Create blocks whenever practical and make use of the copy and mirror commands to expedite the drafting. (Difficulty level: Moderate; Time to completion: 30–45 minutes.)



8. For this exercise, we go back to mechanical drafting while incorporating elements of what we just covered. Set your units to Mechanical; set up all appropriate layers, fonts, linetypes, and mechanical dimensions (with 0.000 accuracy); and create the following mechanical fixture, including the two section views. The symbols used for sections are typical in the industry. Be sure to make a dynamic block out of them with the Flip action, so they can easily be reversed to point in the other direction if needed. For additional practice, add in all hatching and dimensions/text. (Difficulty level: Moderate; Time to completion: 45–60 minutes.)



9. Valves are an important part of P&IDs. In a new file, set up all appropriate layers, fonts, linetypes, and mechanical dimensions (with 0.00 accuracy and a " suffix), then, draw the valve shown. Be sure to use the mirror command wisely to avoid duplication of drawing effort. The hex nuts are not specifically detailed out; you have to improvise. Make a block out of each hex nut with the Flip action added. (Difficulty level: Intermediate/Advanced; Time to completion: 45–60 minutes.)



10. The final exercise of this chapter focuses on the basic drafting of an architectural detail. Set up the proper layers and draft what you see, based on the given primary dimensions. Where none are shown, you have to do a reasonable estimate. Note how lines and arcs were used to create borders for some of the hatches, then deleted. This is a common technique that we try again with plines and splines in Chapter 11, Advanced Linework to achieve a smoother effect. (Difficulty level: Moderate; Time to completion: 60–90 minutes.)