New Commands in AutoCAD 2010: Part 1
Dimensional Constraints, Part 1
by Ralph Grabowski

Introduction

One of the really significant new features of AutoCAD 2010 is parametric drafting. This
technology allows you to create drawings that are parametric in nature, changing designs semi-
automatically.
Parametric drawings look just like dimensioned drawings; the difference is that the parametric
dimensions control the size of location of objects. This is the reverse from what you may be
used to, where the size of objects determined the value reported by associative dimensions.
(Parametrics are not new. They have been around for two decades, and were first invented by
programmers from Russia for a program known as Pro/Engineer. AutoCAD 2010 is limited to 2D
constraints.)
Parametrics are made up of three parts: dimensional constraints, geometric constraints, and
parametric equations.
Geometric constraints are like semi-permanent object snaps: perpendicular, midpoint, and so
on. Whereas osnaps last only for an instant, geometric constraints last for as long as you need
them to.
Dimensional constraints are like associative dimensions but work in reverse: whereas objects
determine the values reported by associative dimensions, dimensional constraints determine
the sizes of objects.
Parametric equations let you link constraints between objects, and specify relationships, such
as one object being half the size of another.

Experienced users typically follow this series of steps in creating new drawings:
a. Sketch the new part with elementary objects using commands like Line, Arc, and Circle.
   Designers don't worry about sizes of the objects or how they are connected -- yet.
b. Apply geometrical constraints to position the basic objects and attach them to each other
   with the new GeomConstraints command. For example, the designer may want two lines to
   be perpendicular to each other, or for a circle to be centered between two arcs.
c. Apply dimensional constraints to specify sizes and distances with the new DimConstraints
   command. For example, one line might be the same length as another, and the circle might
   have a radius that is the distance between the two arcs.
   At this point, the designer has geometric constraints positioning objects, and dimensional
   constraints sizing them. There is one more optional step:
d. Add or edit formulas (parameters) to that define relationships with the new Parameters
   command. For example, one line might be 1/4 the length of another, and the circle might
   have a radius that is 2x distance between the two arcs.

I find it easier to understand this big topic of parametrics by beginning with dimensional
constraints. And so this tutorial starts with the basic operation of a single command that does all
the work:
DimConstraint – adds dimensional constraints to objects and geometric features within objects;
converts associative dimensions to dimensional constraints.

Step-by-Step Tutorial: Applying Dimensional Constraints

In this tutorial, you learn about dimensional constraints and how to add them to drawings.
1. Start AutoCAD 2010 with a new drawing in the default 2D Drafting workspace and the acad.dwt template file.
2. Draw a circle with the **Circle** command, and then a rectangle with the **Rectang** command. The sizes do not matter, but it is helpful if the circle is roughly inside the rectangle.

![Figure 1: Rectangle and circle of arbitrary size.](image1)

3. Use the **DimLinear** command to dimension the width of the rectangle. (The **ENDpoint** object snap is helpful here.)

![Figure 2: One size of rectangle dimensioned associatively.](image2)

4. Let’s convert the linear dimension to a dimensional constraint. Start the **DimConstraint** command using one of these methods:
   - Enter `dimconstraint` at the command prompt.
   - Or enter the `dcon` alias.
   - Or from the ribbon’s Parametrics tab, choose **Dimensional**.
Notice the prompts:
Command: **dimconstraint**
Current settings: Constraint form = Dynamic
Select a dimension or
[Linear/Horizontal/Vertical/Aligned/Angular/Radial/Diameter/Form
<Aligned>]: *(Pick the linear dimension.)*

**TIP** This command operates almost exactly like associative dimensioning commands, except that this one command handles all forms of dimensional constraints. The DimConstraint command places the following kinds of dimensional constraints:
- **Linear** – placed horizontal, vertical, or aligned, depending on how the extension lines are dragged. These are placed on objects and between two objects.
  - Horizontal or Vertical – placed horizontally or vertically, respectively.
  - Aligned -- placed at an angle.
  - Angular -- placed between two lines, on arcs, and on vertices.
  - Radial -- placed on arcs, circles, and polyarcs.
  - Diameter -- placed on arcs, circles, and polyarcs.
- **Form** – specifies the form of constraint, dynamic or annotational (see the follow-up tutorial).
  - Select a dimension – converts associative dimensions.

5. At the `Select a dimension' prompt, choose the linear dimension you drew earlier, and then press Enter. Notice that the dimension changes its look. The distinctive look indicates dimensional constraints.

![Figure 3: Associative dimension changed to dimensional constraint.](image)

Let’s take a look at what the changes mean:
- **Gray color** – the dimensional constraint is assigned a unique style that cannot be changed by you. These dimensions do not show up in paper space layouts and are not plotted. (Dimensions in drawings from earlier releases of AutoCAD can be converted to dimensional constraints.)
- **Text** – the dimension text changes to d1=12.6550 (or whatever size at which you drew the rectangle’s width.) The “d” refers to distance, while the “1” means this is the first distance dimensional constraint in the drawing. The "12.6550" is the distance.
- **Padlock icon** – alerts you that this is a dimensional constraint. (The padlock can be turned off with the DimConstructionIcon system variable.)
Consistent size — the visual size of the dimensional constraint remains constant, no matter the zoom level.

TIP If a circle is dimensioned with a linear dimension, the DimConstraint command refuses to convert it to a geometric constraint, complaining with this error message: “The object selected is not an associative dimension or cannot be converted to a dimensional constraint.” It is the part after “or” that applies in this case.

Similarly, you cannot apply linear dimensional constraints to circles. These two restrictions do not apply to arcs.

7. As I mentioned in the introduction, dimensional constraints control the sizes of objects. To see this at work, change the value of d1 to change the size of the circle, like this:
   a. Double-click the text of the dimensional constraint. Notice that you can now edit the text. (Alternatively, use the TextEdit command, new to AutoCAD 2010.)
   b. Enter a larger value, such as 14.
   Notice that the one side of the rectangle instantly increases in length.

![](image)

Figure 4: Dimensional constraint changes the size of objects.

8. Now reuse the DimConstraint command, but this time apply a diameter dimensional constraint directly to the circle. You could do this at the command line, as shown below, but...

Command: dimconstraint
Current settings: Constraint form = Dynamic
Select a dimension or
[Linear/Horizontal/Vertical/ALigned/ANgular/Radial/Diameter/Form
<Aligned>: (Type D to specify the Diameter option.)
...but it is much easier to use the ribbon for applying dimensional constraints, like this:

a. Choose the **Parametrics** tab.
b. In the Dimensional panel, click the **Diameter** button.
c. Select the circle, as prompted by AutoCAD, “Select arc or circle.”
d. Choose a location for the dimension line, and then press **Enter**.

Notice that the dimensional constraint is named “Dia1,” which indicates the first diameter one in the drawing. As before, you can change the size of the circle by changing the value of dia1.

9. A second powerful aspect of dimensional constraints is that they can be linked to each other. This is easy to do, because AutoCAD automatically assigns names to dimensional constraints, as we saw earlier. The names allow you to reference the constraints. Here is the complete list of names:

<table>
<thead>
<tr>
<th>Name</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>d</td>
<td>Distance (length)</td>
</tr>
<tr>
<td>rad</td>
<td>Radius</td>
</tr>
<tr>
<td>dia</td>
<td>Diameter</td>
</tr>
<tr>
<td>ang</td>
<td>Angle</td>
</tr>
<tr>
<td>user</td>
<td>User-defined</td>
</tr>
</tbody>
</table>

Let's make the size of the circle depend on the length of the rectangle. (In a later tutorial, we will make the parallel sides of the rectangle the same size.)

a. Double-click the circle's dimensional constraint. (Alternatively, select the constraint, right click, and then choose **Edit Constraint** from the shortcut menu.)
b. Enter the following formula, and then press **Enter**:

\[
d_{1/2}
\]

This formula means that the radius of the circle will always be one-half the length of the constrained side of the rectangle. Notice that the circle instantly change its size.
Consider how the two objects now relate to each other through the dimensional constraints:
   The width of the rectangle is constrained by its dimensional constraint. You can change the value of d1, but you cannot change the width on its own, such as through grips editing.
   The diameter of the circle is constrained by the rectangle. You cannot change the diameter of the circle.

Now you understand the purpose of the padlock icon: those entities are locked in size.

11. Play with the dimensional constraints, changing the size of the rectangle and circle through d1.

12. This concludes the tutorial. Save the drawing as "DimConstraint.Dwg," and then exit AutoCAD if you wish. You will use this drawing for the follow-up tutorial, in which you learn about variations on dimensional constraints.

**Command Reference**

**DimConstraint** commands adds dimensional constraints to objects and geometric features within objects.

**DimConstraintDisplay** system variable toggles the display of dimensional constraints.
**DimConstructionIcon** system variable toggles the display of the padlock icon.

**DynConstraintMode** system variable toggles the display of hidden dimensional constraints during selection.

**ConstraintNameFormat** system variable specifies the text format of dimensional constraints.

**ConstraintRelax** system variable toggles the enforcement of constraints during editing.

**ConstraintSolveMode** system variable toggles constraint behavior during editing.

**Test Yourself**

1. What is the purpose of the **DimConstraint** command?
   a. To freeze the text of associative dimensions.
   b. To constrain dimensions to paper space only.
   c. To constrain the sizes of objects.
   d. To convert associative dimensions from model to paper space.

2. Which abbreviation refers to linear dimensional constraints?
   a. D1
   b. Dist1
   c. Dia1
   d. Rad1

3. The values of different dimensional constraints can be linked.
   True / false.

4. Associative dimensions cannot be converted into dimensional constraints.
   True / false.

5. Which system variable turns off the padlock icon?
   a. DimConstructionIcon
   b. DimConstraintDisplay
   c. DimConstruction
   d. DynConstraintMode

[Answers]

1. c. To constrain the sizes of objects.
2. a. D1
3. True.
4. False.
5. a. DimConstructionIcon

[end of file]