Up And Down The Scales In AutoCAD®

Bill Fane – BCIT, CADALYST, etc

GD319-2   One of the first "Learning Curve" columns written for "CADalyst" magazine 20 years ago covered the topic of handling drawing scales in AutoCAD software. Well, the more things change, the more they stay the same. Drawing Scales is still a hot topic, even though AutoCAD has added several new functionalities to assist us. One problem, however, is that some of these functions appear to contradict each other, causing confusion as to which is the "correct" way to handle Drawing Scales. This class will follow a chronological pattern, starting with the earliest model-space-only drawings (of which there are millions still in existence) through Paper Space layouts, trans-spatial dimensions, and the latest self-scaling annotation objects in AutoCAD. We'll explore the operations, pros, and cons of each option so you can decide which one is best for you.

About the Speaker:

Bill is a recovering doorknob designer, having been a product engineer and then product engineering manager for Weiser Lock in Vancouver, Canada, for 27 years. An AutoCAD user since 1986, Bill has taught AutoCAD and mechanical design at the British Columbia Institute of Technology for more than 10 years, and teaches Autodesk Inventor classes at the Institute's training center. He is an active member of the Vancouver AutoCAD Users Society, and an equally active writer: He has authored "The Learning Curve" column for "CADalyst" magazine since 1986; written about Autodesk Mechanical Desktop (now AutoCAD Mechanical) and Autodesk Inventor for Autodesk's "Point A Toplines," and writes for "Inside AutoCAD Journal," "Design Product News," and "Cutting Tool Engineering." He has been lecturing at AU since 1995.

bill_fane@bcit.ca

NOTE

This course is intended primarily for BEGINNERS,
plus those more-advanced users who want to learn
about the new annotative features in AutoCAD 2008.

Once Upon A Time…

…a long time ago, back in the days when AutoCAD was young, I started using AutoCAD version 2.17g in September of 1986. In those days there were very few night-school courses, and the supplied documentation was "slim" in that it explained what the commands were and what they did, but nothing on the philosophies to be followed. "Drawing scale" was one of those hot-topic items that was not in the book, but was obviously of major concern to everyone. We blundered along, and with the support of local user groups and CADALYST magazine we managed to sort it out.

Over the years Autodesk implemented several new functionalities with the intention of fixing things. They often helped, but were not the final solution.

Fast-forward to AutoCAD 2008, wherein I firmly believe the problem has been solved once and for all.

This course will take us on a walk through history, as we analyze the functionality, the techniques, and the pros & cons of the different approaches.

You may well ask *"Why not just jump to AutoCAD 2008, if it is so much better?"* The answer is that there are millions of drawings out there that have used the varying techniques, and you will probably encounter every technique at one time or another. You would do well to understand them.

There have been three basic technologies, plus sub-variants. We'll explore each variant in turn, beginning with a brief overview followed by specific techniques for each one.

AutoCAD's Top Model

Initially there was only "model space". All drawing and annotation was done in this single environment. Dimensions were not associative, and in fact were just dumb blocks composed of lines and text. At least they were "semi-automatic" in that they would measure the distance involved and then fill in the appropriate text value.

It didn't take users long to figure out that the best way was to always draw everything full size, be it a map of the universe or a drawing of an atom. This way the automatic measurement functionality worked properly.

Obviously we then had to scale things somehow to get them to fit the sheet of paper. The PLOT command handled this by asking for a scale factor at plot time, and then scaling everything to suit.

The problem was that it scaled <u>everything</u>, including text and dimensions.

Normal drafting practice is to use a height for text objects and for the text in dimensions of about 1/8" (0.125). Okay, so we are drawing a building that we want to plot at a 1/4"=1' (1:48) scale. To make things work, we would have to set the text size to be the desired final plotted size multiplied by the inverse of the plotting scale. Thus, text in a 1:48 drawing would have to be 0.125 x 48 = 6 units (inches?) tall, so that when the *PLOT* command shrunk it by 1/48 it would come out 1/8" on paper. Sometimes two wrongs do make a right… (it usually takes 3 or 4). The same holds true the other way: a 10:1 drawing of a small mechanism part would need a text size of 0.125 x 0.1 = 0.0125 units, because it will get plotted 10 times bigger.

That's fine for text sizes, but how about dimensions?

The good news was that there was (and still is) a system variable called *DIMSCALE*. This scale factor was applied to each dimension as it was created, where it scaled the size of the text, the arrowheads, the dimension lines, and the extension lines accordingly. To make things work, it would be set to the inverse of the desired plot scale. Thus, a 1:48 drawing would have a *DIMSCALE* value of 48 while the 10:1 drawing would use 0.1.

Oh, by the way, the same values apply to *LTSCALE* (LineTypeSCALE) for non-continuous line types and to hatch pattern scales.

So far so good, but what about details at other scales?

Ah, that gets a little messy. Now you have to take two factors into account; the main plotting scale, and the detail scale.

The first issue is the creation of the detail itself. Don't forget, we are working only in model space so to get a detail at another scale we need to break the first rule and draw it at something other than full size.

The easy way to do this is to create the detail full size, and then use the *SCALE* command to re-size it. Another possibility is to create a full-size *Block* of the desired details, and then to *Insert* it at a suitable scaling factor. So what scaling factor do we use? It's the ratio of the detail scale divided by the main plotting scale.

Assume you have a 1:48 drawing, and now want to put a 1:12 scale detail in it. You need to use the *SCALE* command and re-size the detail to 1/12 (=0.8333333…) divided by 1/48 (=0.02083333…) which comes to 4. Obviously a 1:12 detail is four times as big as one at 1:48.

Similarly, a 5:1 detail in a 2:1 drawing would be scaled by 2.5.

Now comes time to dimension the detail. Oops, all the values are wrong by the scale factor.

Fortunately there was/is a system variable called *DIMLFAC* (DIMension Linear FACtor). When you apply a dimension, the value that it displays is the measured distance multiplied by the *DIMLFAC* value. Because our details are bigger than the main drawing, we need to set *DIMLFAC* to the inverse of the value we used during the *SCALE* or *INSERT* command.

Thus, a 1:12 detail in a 1:48 drawing needs a *DIMLFAC* value of ¼= 0.25, and the 5:1 detail in a 2:1 drawing would use 2/5=0.4.

Hey, I never said this was going to be easy!

To make matters worse, the earlier releases didn't have named text or dimension styles, so every time you flipped between dimensioning the main view and dimensioning the detail view you had to manually reset *DIMLFAC*.

Even worser yet, dimensions were not associative. When you *MOVE*d anything, or after the *STRETCH* command arrived, if you wanted to change anything, you had to erase the dimensions and re-apply them.

You didn't even want to think about changing the overall scale of a drawing or of a detail after the text and dimensions had been applied.

I saved the best for last. By default, text height was 0.200 (and you were asked for a size every time you inserted text) while dimension text defaulted to 0.180. The solution was to change the default text height from 0 (zero) to a specific value, say 0.125 or as corrected for plot scale. Now dimension text came out the same size as regular text and you were not prompted for the text size with each insertion.

The arrival of associative dimensions made editing a little easier, but the fundamental problems remained.

As I said earlier, the main reason for discussing this in such detail is that there are millions of drawings out there that were created that way, and you will probably have to edit some at some point. Know your enemy.

Spaced Out on Paper.

When paper space arrived, the theory was that model space showed the object, while paper space showed all the documentation. Dimensions could be applied by reaching in through the paper space viewport to use object snaps on the model.

The advantage to putting the model in model space and the annotations in paper space was that paper space always plotted 1:1, so there was no need to change text sizes or dimension scale factors to suit the plot scale. Everything can simply be set to the desired final size; normally 1/8" for text, for example. The plot scale was set by the zoom ratio of the viewport window; set this by double-clicking inside the viewport and then entering the command *Zoom nXP*, where *n* is the desired scale factor. Better yet, click on *Modify | Properties*, select the viewport boundary, then select a standard scale value from the *Standard scale* drop list. It's also a good idea to then turn on *Display locked* so the scale factor won't accidentally get messed up.

Using paper space also has a big advantage when creating details at other scales. You still draw everything full size in model space, but then use viewport scaling to set different scales for different details. In fact, you can even show the same objects at two or more different scales.

Another big advantage to using paper space is that you now only need one title block and border for each sheet size. When everything went in model space we had to have different borders for each sheet size and each plotting scale. Five sheet sizes times six standard plotting scales needed thirty borders!

Ah, but what do we do about the *DIMLFAC*? Surely dimensions will measure different values as they reach through viewports at different scales. No problem; just set *DIMLFAC* to 0 (zero) before creating the dimensions and AutoCAD takes care of everything. You do this in the *Dimension Style Manage*r (*DIMSTYLE*) dialog; under the *FIT* tab, click on the *Scale dimensions to layout* button.

Note that all of the foregoing applies to any <u>layouts</u>, all of which are in paper space.

Using paper space has its pros and cons:

Pro:

- Simplified text and dimension size settings.

- Much simpler to create views and details at different scales.

Con:

- You can't see the dimensions when you are working in model space

- Dimensions have lost their associativity with the objects they are dimensioning. The newer "Trans-spatial Dimensions" have overcome the associativity problem, but there are still no dimensions when you are working in model space.

This is still a hot-button issue, with the debate raging on both sides. One side swears by the procedure of putting all annotations in paper space, while the other side swears at them and puts the dimensions in model space. This latter technique basically reverts us back to the earlier model-space-only scenario, except that borders, title blocks, and general text notes can go in paper space. It works, but it is a little complex.

To make this technique work effectively, we usually end up with a number of different layers for dimensions. These layers are frozen and thawed as appropriate so that only the desired dimensions appear in each viewport. This applies particularly to details at other scales.

Annotative: The Final Solution

AutoCAD 2008 delivered our salvation (finally) to the 25-year-old issue of drawing scales. We can now put all dimensions, and some text items where appropriate, in model space. They are visible when we are working in model space. They automatically size themselves to suit the correct scale in model space and in paper space layout viewports.

The title block and border go in paper space, along with general text notes and things like revision tables.

Better yet, block insertions and/or attributes attached to block insertions can be placed in model space where they automatically size themselves according to the current scale setting.

At first glance, annotative objects can seem a little intimidating because of their many options and variants, but they are actually the essence of simplicity.

1. Start a new drawing, using the standard *acad.dwt* as the template.

2. Draw a circle 25 units in diameter and a horizontal line 10 units long.

3. Apply a diameter dimension to the circle and a linear dimension to the line. Almost invisible, aren't they?

4. Select *Format | Dimension style*.

5. Change the current dimension style from *Standard* to *Annotative*. The triangle icon, used throughout AutoCAD 2008 to indicate annotative, is meant to show the end of a triangular engineer's scale. Click *Close*.

6. In the lower right corner of the AutoCAD window, note the *Annotation scale* item and the value 1:1 beside it. Click the triangle and select 1:20 from the drop list.

7. Apply a diameter dimension to the circle. Note how it sizes itself accordingly.

8. Change the Annotation Scale to 1:10.

9. Apply a linear dimension to the line. Once again, it sizes itself correctly for the selected scale.

10. Switch to paper space layout number 1.
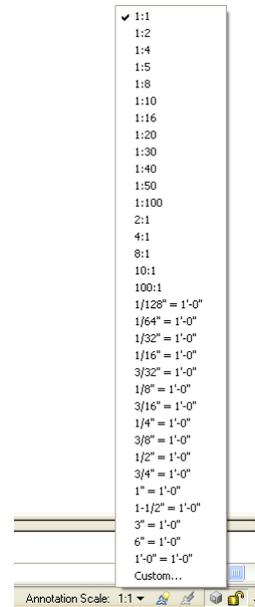
11. Click on the viewport boundary, and select 1:10 from the drop list beside the VP Scale item that appears in the lower right corner of the AutoCAD screen. The viewport scales itself accordingly, and only the dimension on the circle is visible. The dimension is scaled properly.
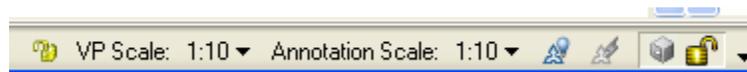
12. Click on the viewport boundary, and select 1:20 from the drop list beside the VP Scale item that appears in the lower right corner of the AutoCAD screen. The viewport scales itself accordingly, and only the dimension on the line is visible, scaled properly.

13. Click on the viewport boundary yet again and set the VP scale to 1:16. The viewport scales itself, but now neither dimension is visible.

14. Re-size the viewport to about half its size, and then Copy it (yes, a viewport is an object than can be copied and edited like any other).

15. Set one viewport scale to 1:10 and the other to 1:20.

16. Double-click inside the 1:10 viewport and pan until the circle and its dimension are nicely framed.

17. Double-click inside the 1:20 viewport and pan until the line and its dimension are nicely framed.

18. Double-click in an open area of the layout to return to paper space.

19. Bingo! Two details at two different scales and we did not need to do any calculations or switching of styles. Everything was drawn full size and was dimensioned in model space.

A single annotation item can have more than one scale applied to it, and it can be in different positions at the other scales. It will appear in any viewports whose scale matches one of the applied annotation scales. Nonetheless, it is still a single item, so that changing the size of a dimensioned object changes the dimension value at all scales.

20. Return to model space.

21. With the annotation scale set at 1:10, place an aligned dimension between the circle and one end of the line.

22. Go to the paper space layout and observe how this new dimension only appears in the one viewport.

23. Return to model space.

24. Double-click on the aligned dimension to invoke the *Properties* tool.

25. Expand the *Misc*. section.

26. Click on *Annotation scale*, and then on the "three-dot" button at the right-hand end of the *Annotative scale* window. This brings up the A*nnotation Object Scale* dialog box.
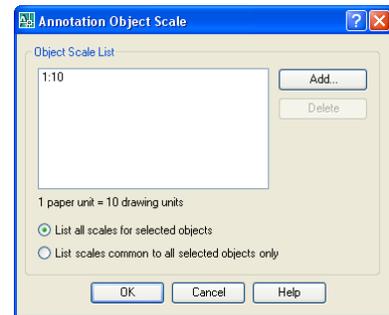
27. Click on Add, and then select 1:20 from the scale drop list. Click on OK.

28. Close the Properties tool and observe how your drawing now has two dimensions between the line and the circle, at two different sizes.

29. Switch to the layout. The aligned dimension appears in both viewports, correctly sized to suit each viewport scale.

30. Double-click inside one viewport to enter model space and use grip editing to change the location of the aligned dimension.

31. Double-click in the layout to return to paper space. Observe how the aligned dimension has a different location in each viewport.

32. Now comes the magic part. Return to model space and edit the line and circle to change their sizes and relative positions.

33. Go back to the layout. Magic! Everything has updated! Historically double-dimensioning can be very dangerous, but annotative dimensions can solve this.

Automatic annotation scaling can also be applied to text and mtext, blocks &/or attributes, line type scales, and hatch patterns.

Let's start with **text** and **mtext**.

1. Invoke the *STYLE* command (*Format | Text Style…*).

2. Create a new style, or edit an existing one.

3. Turn on the *Annotative* check box.

4. If you want to, set a *Paper Text Height*. This is the height at which you want text to appear in paper space. If you leave it at zero then you will be prompted each time, and the default is 0.200. If you set a finite height (say 0.125) then you will not be prompted and all text using this style will be the specified height.

5. Click *OK*.

6. Done! Now, any text or mtext that you create using this style will be annotative, and will scale itself accordingly.

7. Like dimensions, annotative text objects will only appear in those viewports whose VP scale matches an annotative scale of the objects.

8. An annotative dimension style scales its text no matter if its associated text style is or is not annotative.

9. Conversely, a non-annotative dimension style does not scale its text even if its associated text style is annotative.

NOTE

Setting an existing text or dimension style to *Annotative* is not retroactive, but you can change the properties of existing objects to be annotative, and you can add scales to them

Next come **attributes** and **blocks**.

1. The attribute definition dialog box (*ATTDEF*) now includes an *Annotative* check box.

   a. If you specify a text style that is annotative then this box turns on automatically. You can turn it off even if the text style is annotative, and vice-versa.

   b. If Annotative is ON then the text size is automatically corrected for the current scale setting.

   c. If an attribute definition is Annotative then it will size itself to the current scale each time it's mating block definition is inserted.

2. The block definition dialog box (*BLOCK*) now includes an *Annotative* check box.

    a. For the least amount of confusion, the current scale should be 1:1 when creating annotative blocks.

    b. Annotative block definitions can also contain annotative or non-annotative text and attributes in any combination.

    c. When an annotative block is inserted, the basic AutoCAD objects such as lines, circles, arcs, and so on will be scaled. Text and attributes will scale or not, depending on if they are annotative or not.

    d. When a <u>non</u>-annotative block is inserted, the basic AutoCAD objects such as lines, circles, arcs, and so on will <u>not</u> be scaled. Text and attributes <u>will</u> scale or not, depending on if they are annotative or not.

    e. Annotative blocks would normally be used for "explanatory" objects such as north arrow, surface finish, and welding symbols. They would not normally be used for real objects such as tables, chairs, and machine components.

Now for **hatch** objects:

1. The *HATCH* dialog box now includes an *Annotative* check box. It must be checked manually each time hatching is applied.

2. If you set a specific scale in the Hatch dialog box then it will be multiplied by the annotative scale when the hatch is applied.

3. Annotative hatching will only appear in those viewports whose annotative scale matches the VP scale of the viewport.

4. The Properties tool can be used to add additional scales to a previously-applied pattern.

Now that we have seen the fundamentals, let's take a look at some of the details:

1. The Properties tool can be used to apply additional annotative scales to more than one object at a time, and/or to convert multiple objects to or from being annotative.

2. Non-continuous linetypes are automatically annotative. The annotative scale is multiplied by any global scale factor (*LTSCALE*) that may be set.

3. The list of standard scales can be edited to add new ones or to delete unwanted ones using the *SCALELISTEDIT* (*Format | Scale List*) command.

4. By default, the *Standard* text & dimension styles are current and are <u>not</u> annotative. I assume this was done to maintain compatibility with older customizing, such as scripts, menus, and AutoLisp.

5. The new Multi-leaders (*MLEADER*) can be annotative, as set by the *MLEADERSTYLE* command.

6. Tables and fields are not annotative, even if their associated text style is.

Now we come to a couple of interesting buttons, down in the lower RH corner of model space.

7. *Annotation Visibility…* . When this is ON, then AutoCAD displays all the annotative objects regardless of their scale setting. Those objects that have an annotative scale that matches the current setting are shown at that scale, while others are shown in the scale at the top of their individual scale list. If you select an annotative object when this is on, then AutoCAD will highlight all the different scaled versions of it.

   It is off by default, but turns itself on as soon as you add a second scale.

8. *Auto Add new scales…* . This one can be really dangerous. It is off by default, and rightly so. When it is on and you change the model space scale then <u>all</u> existing objects automatically acquire the new scale. The problem here is that now every object matches every viewport scale and so appears in every viewport.

Finally, there is the question of compatibility with earlier releases. AutoCAD 2008 is able to save back to earlier releases as far as 2000, but these releases didn't have self-scaling annotations.

No problem. AutoCAD simply creates new layers for each of the variants of the annotative objects. The layer names are the existing names plus "@nnn" added at the end, where "nnn" is the scale factor. A copy of each annotative object is placed on the appropriate layers according to their scales.

When the drawing is re-opened in AutoCAD 2008, it brings everything back together again.

And In Conclusion…

Self-scaling annotations are invoking a massive paradigm shift in how all AutoCAD drawings are created from now on.