AS502286

Taming Parametric Curved Geometry in the Revit Family Editor

Paul F Aubin (Author/Consultant)

Paul F Aubin Consulting Services, Inc.

Learning Objectives

- Create reliable, flexible, parametric curved forms in the Revit Family Editor.

- Componentize your families: reuse the parts and pieces in multiple families and contexts.

- Discover what it takes to make a properly flexing compound curve.

- Use formulas and other advanced techniques to control family curves.

Description

Have you ever tried to control the shape of a curved form parametrically in the Family Editor? If so, you've no doubt discovered that flexing them sometimes throws you a curveball. In this session we'll explore several techniques to tame your unruly parametric curves. We'll look at examples of circles, arcs, quarter round, half round, arches, and we'll even check out some splines. We'll look at both simple and compound curves. We'll work primarily in the traditional Family Editor but most techniques apply to the massing Family Editor environment as well. We'll explore curvature and rotation, using both armatures (or rigs) and formulas to constrain flexing behavior. We'll even throw in some trigonometry for good measure! After this session, I cannot guarantee that you'll never have another misbehaving curve in your family content, but what I can promise is that you'll come away with several useful tools to help you tame them when curve-mischief strikes!

Speaker

Paul F. Aubin is the author of many Revit book titles including his "deep dive" into the Revit family editor: Renaissance Revit. He has also authored dozens of Revit video training titles on LinkedIn Learning (powered by lynda.com content) covering all levels of Revit, Dynamo and ReCap. Paul is an independent architectural consultant providing content creation, implementation, and training services. His career of over 30 years, includes experience in design, production, BIM management, coaching, reality capture and training. Paul is an active member of the Autodesk user community and has been a top-rated speaker at AU and other industry conferences for many years. Paul is an Autodesk Expert Elite and the founder of ChiNamo; the Chicago Dynamo Community. He was recently named a member of the board of directors for the Volterra-Detroit Foundation. He lives in Chicago with his wife and their three children are currently pursuing their careers around the country.

Visit: www.paulaubin.com

Introduction

Controlling the location and flexible behavoir of straight lines is easily accomplished in the family editor using simple reference planes or reference lines and labeled dimentions. When you introduce curves (i.e. arcs, circles and ellipses) it becomes a little more difficult to control them parametrically in a *reliable* and *stable* way. Techniques and procedures designed to address this issue are what this class is all about.

Dataset of Practice Files

A collection of dataset files is provided to supplement this class and paper. All starting files for each exercise are provided and several "catch-up" files are also provided. In this way, you can use this paper as a hands-on guide after the class is complete and try each of the techniques I show you first-hand. You can download a copy of the dataset at: https://www.paulaubin.com/au

Curvature in the Traditional Family Editor

This class focuses on techniques in the traditional family editor environment. I assume that you are familiar with the basics of the traditional family editor such as: establishing reference planes, creating labeled dimensions and parameters and building 3D solid and void forms. This class will *not* explore the massing (conceptual/adaptive component) family editor environment. As such, when speaking of curvature in the family editor in this paper, it will be referring to two-dimensional curves confined to the active work plane that include: circles, ellipses, arcs (portion of a circle), elliptical arcs (portion of an ellipse) and splines (Bezier splines with control handles) (see Figure 1).
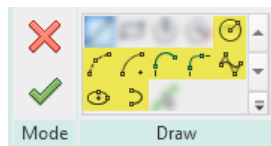

Figure 1—Curves available in sketch mode for solid and void forms in the traditional family editor

These curved forms can be used in each of the five solid and void forms available to us in the traditional family editor[1]. We will explore several isolated scenarios in the beginning of the paper and then combine techniques later on.

General Rules and when to Break Them

The general rule-of-thumb in family content creation is to create a clear and consistent hierarchy between the references, constraints and geometry. Typically, this means that you would want to lay down your reference planes and/or reference lines first. You would next apply constraints and parameters to these references and flex them to be sure that they function properly. Finally, you would build your geometry and lock it to this properly flexing armature of references. This is

---

[1] While I am limiting my explorations here to the traditional family editor, I should note that all of the techniques showcased in this class will work in the massing (conceptual/adaptive component) family editor environment as well. It is just that I will not be exploring any techniques that are unique to that environment here. There are several of these to be sure and they could easily fill another entire class. Maybe next year…

the so-called "bones/muscle/skin" analogy and it is often the best-practice approach in *most* situations.

However, as with any rule or guideline there are *always* exceptions. In our first few lessons, we'll look at a few examples where the dimensions (be they constraints or parameters) will be applied *directly* to the geometry instead of a reference. Now please allow me to stress that we are not throwing out the general rule. It should still be followed wherever possible. Specifically, if you can dimension your reference lines or reference planes first, and then attach geometry to them, this is generally the preferred (and better behaving) approach.

But as we will see, when curves are involved, this is not always possible or desirable.

Create a Parametric Circle

Two forms that will use the direct attachment method will be circles and ellipses. This is because you cannot align and lock curved geometry to linear reference planes. But you can use dimensions. Therefore, to parametrically flex and or reliably constrain curved elements, we often need to apply the dimensions directly to the forms.

1. From the File menu, choose **Open > Family**.

2. Select the file named: *_01 Seed (Instance Based).rfa* and then click Open.

3. Save the file as: **Circle**.

All solid and void forms in the family editor can use circles. So you can perform the following procedure on any kind of form. To keep the exercise simple, we'll use an extrusion, but feel free to practice the steps on other forms as well later on. Since the seed already contains an extrusion we can simple edit it.

Work in the *Ref. Level* floor plan view

4. Select the extrusion onscreen.

⇨ On the Modify | Extrusion tab, click the Edit Extrusion button.

⇨ Select and delete the four existing sketch lines.

5. On the Draw panel, click the Circle icon.

⇨ Draw the circle centered on the reference planes in the file. Snap the size of the circle to the width and depth defined by the reference planes (see Figure 2).
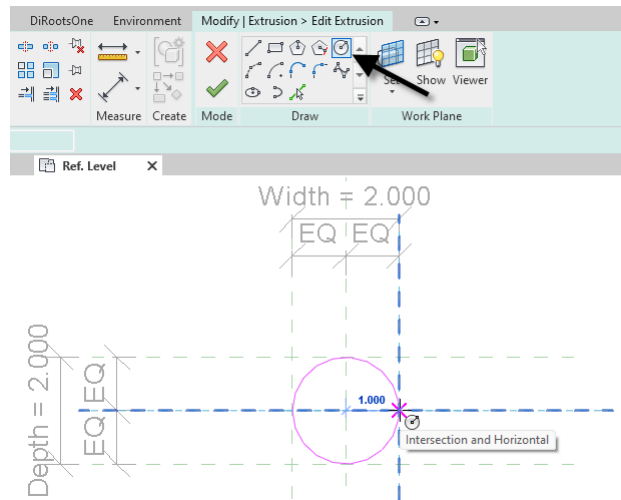
Figure 2—Draw a circle and snap it to the reference planes

Even though we snapped the circle to the reference planes, it will not flex when the parameters flex. Try it out if you like. Simply snapping to the reference planes will not be enough in this case. Instead we to make our intentions known to Revit. We can do this two ways. Either we can add a another parameter for the radius (or diameter) and directly apply it to the circle. Or we can use linear dimensions instead to attach the circle to the existing reference planes. Let's start with a new parameter for the radius.

6. Select the circle, and then click the icon beneath the dimension to make the temporary dimension permanent. Click the Modify tool to cancel (see Figure 3).
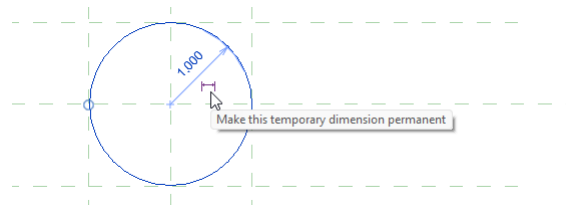

Figure 3—Make the temporary radius dimension permanent

7. Label this dimension with a new instance based parameter and call it: **Radius** (See Figure 4).
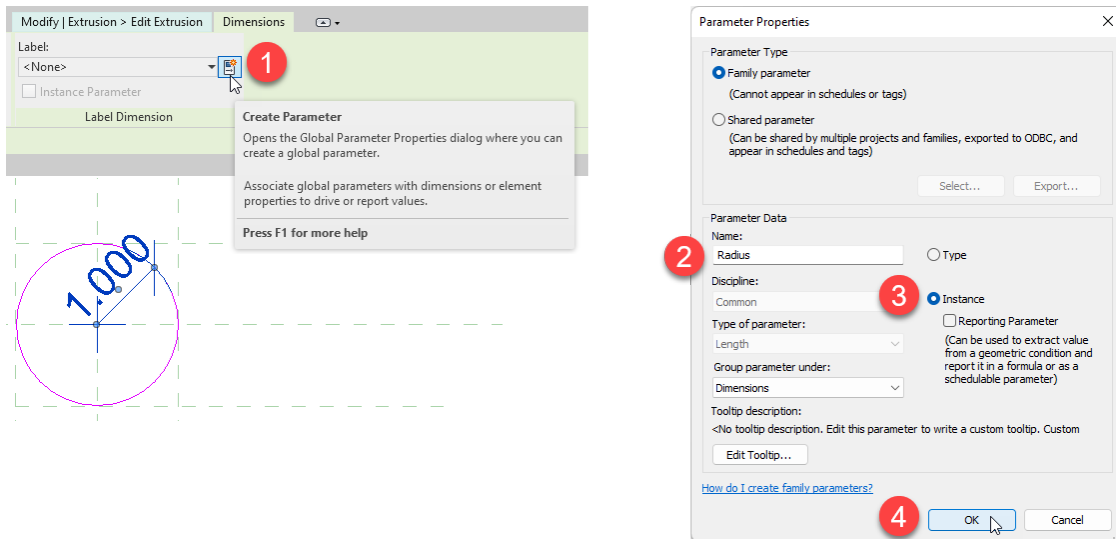
Figure 4—Create a Radius parameter and label the dimension with it

We could flex the extrusion now to be sure that it works, but since we already have the Width and Depth parameters that represent the overall extents of the circle, let's add a few simple formulas in the "Family Types" dialog to tie all three parameters together so they flex in a logical way. We want **Width** and **Depth** to be equal, and **Radius** to be half of these.

8. Open the "Family Types" dialog. In the Formula column next to **Width**, Type: **Depth**. For the Formula next to **Depth**, type: **Radius * 2** (see Figure 5).
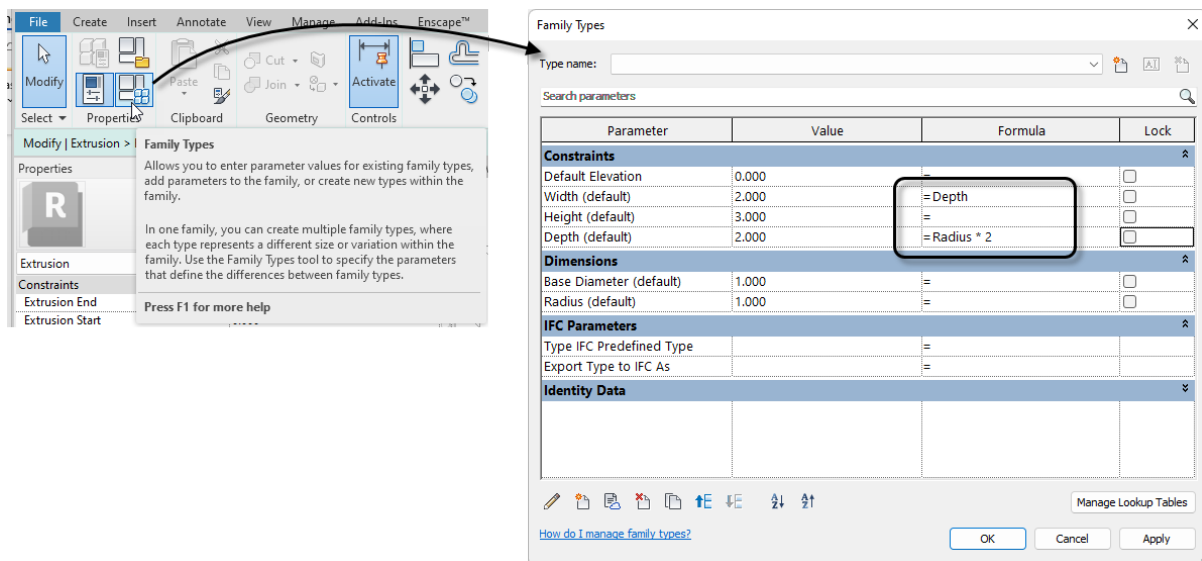

Figure 5—Link the three parameters together with formulas

Note: You will note that there is a parameter in this file called: **Base Diameter**. This parameter can be used as a multiplier to enable proportional scaling. For the time being we are ignoring this parameter. If you prefer, you can set the **Width** and **Depth** equal to **Base Diameter** times a multiplier.

9. Flex the family.

The circle should change size as you flex the radius. The width and depth should also flex with the radius.

> **Note:** If you prefer, you can use the Diameter dimension tool on the circle and thereby eliminate the need for the `Radius` parameter and the formula based on it. In this case, the diameter can be set equal to the either the `Width` or `Depth` (or even Base Diameter) parameter directly.

> **CATCH UP!** You can open a file completed to this point named: *A01_Circle.rfa*.

Create a Parametric Ellipse

Creating a parametric ellipse is very similar. For this one, we'll tie the `Width` and `Depth` separately to each ellipse axis. Otherwise, the procedure is nearly the same.

Continue in the previous file or open the catch-up file.

1. Save the file as: **Ellipse**.

2. Open the "Family Types" dialog.

⇨ Clear the formulas.

3. Select the `Radius` parameter and then click the Modify Parameter icon (at the bottom of the dialog).

⇨ Rename it to: **X** and then click OK. For the Formula, type: **Width / 2**.

4. Use the new icon and add another instance based Length parameter named: **Y**.

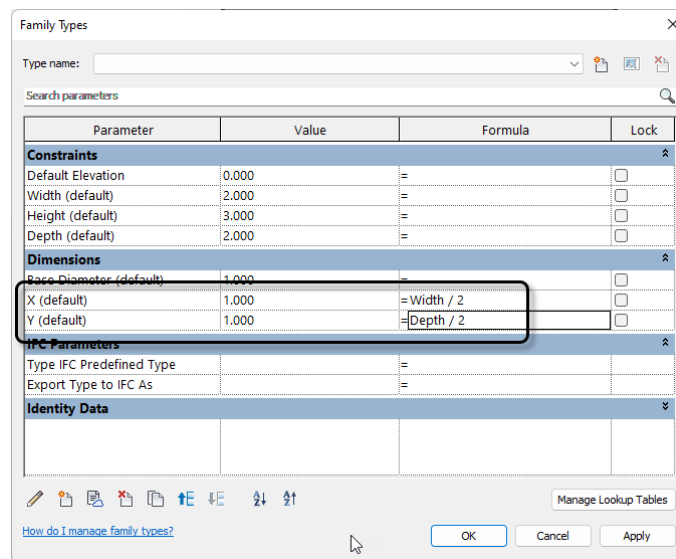⇨ For the Formula, type: **Depth / 2** and then click OK (see Figure 6).

Figure 6—Edit the parameters to prepare them for the ellipse

The parameters are now ready, let's edit the extrusion next.

5. Select the circle extrusion and then on the ribbon, click the Edit Extrusion button.

⇨ Select the circle sketch and delete it.

6. On the Draw panel, click the Ellipse icon and then click the center point at the intersection of the two center reference planes.

7. Snap the first ellipse axis to the intersection of the Center (Front/Back) and Right reference plane. Snap the other axis to the intersection of the Center (Left/Right) and Back reference planes.

8. Make both dimensions permanent (see the left side of Figure 7).
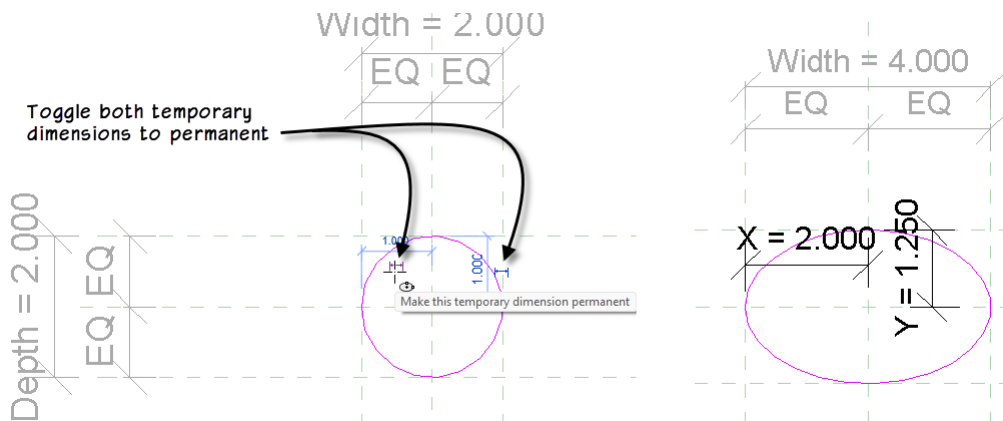

Figure 7—Draw an ellipse and make the dimensions for both axes permanent (left) Label the dimensions (right)

At the moment, this ellipse looks like a circle. Mathematically, a circle is really just a special case of an ellipse. So you can actually use an ellipse all the time if you prefer and when you need a circle just flex it so that both axes are equal. Let's finish it and flex.

9. Label the vertical dimension with the Y parameter and the horizontal one with the X parameter.

10. Open the "Family Types" dialog and flex (see the right side of Figure 7).

You should be able to input values for any of the four parameters. If you make the Width and Depth different, you will get an ellipse. Make them the same to get a circle.

> Note: An alternative approach to making the axis dimensions permanent and using half axis length parameters is to add a dimension on each of the four sides directly to the ellipse. You can do this by pressing TAB to highlight the endpoint of the ellipse before you click to place the witness line. Be sure to lock each resulting "zero" dimension (see the left side of Figure 8). You can also try this approach with the circle example earlier. In the case of the circle, you TAB until the circle highlights; endpoints will not appear (see the right side of Figure 8).
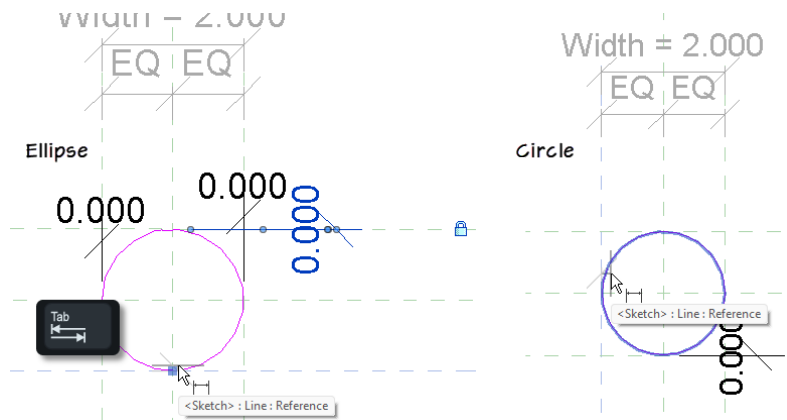
Figure 8—Alternatively, add dimensions between the reference planes and endpoints using the TAB key

CATCH UP! You can open a file completed to this point named: *A02_Ellipse_Half Parameters.rfa* and another version called: *A03_Ellipse_Zero dimensions locked.rfa.*

Both of these examples show situations where you will want to apply the dimention directly to the sketch lines as opposed to using the align and lock method that is more typically employed in family content creation.

Center Mark Visible

For curved objects like circles and ellipses you can display the Center Mark. This can be helpful to ensure that the element flexes in the way you expect and intend. Any curved object has this setting including arcs. (For ellipses, you can also display the foci). To display either, look for the checkbox on the Properties palette.
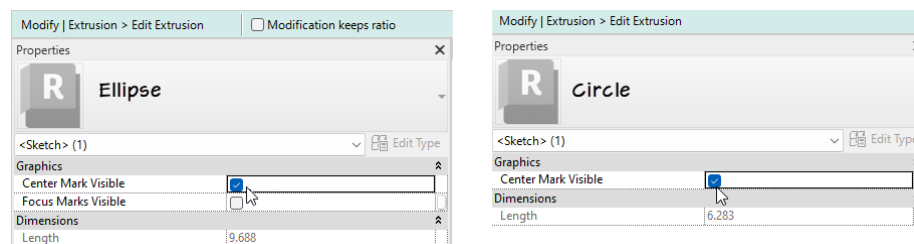

Figure 9—Enable the Center Mark or Focus Marks for Circles and Ellipses on the Properties palette

If in the course of flexing your family, you anticipate that the element might tend to shift relative to its center, enabling and constraining the center mark can be a good way to prevent undesirable effects. We'll look at some examples in the next topic.

Automatic Sketch Dimensions

You may have noticed that when you flex Revit families certain parts may be constrained automatically. Sometimes it behaves exactly as you want and expect, but not always. So how do you determine what automatic constraints are applied? The answer is "Automatic Sketch Dimensions." Automatic Sketch Dimensions can be made visible in the Visibility/Graphics Overrides dialog. Use the VG shortcut to open the dialog just like you would in a project. Once there, the Automatic Sketch Dimensions are a sub-component of Dimensions.
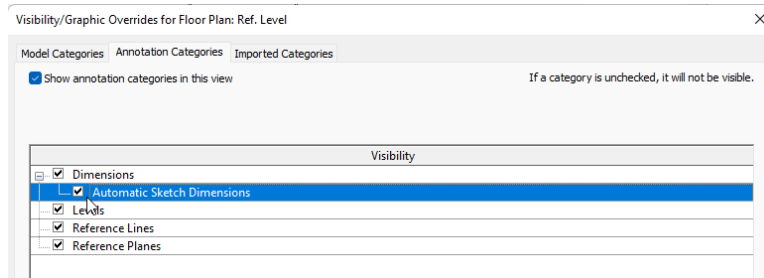
Figure 10—Enable the display of Automatic Sketch Dimensions in the Visibility/Graphics dialog

Using Center Mark Visible and Automatic Sketch Dimensions

Each of the previously mentioned tools are very useful when constraining flexible curvature in the family editor. Let's look a simple example.

1. Open the file: *AO1_Circle.rfa*.

2. Save the file as: **Center and Sketch Dimensions.rfa**.

3. Open the "Family Types" dialog. Clear the formulas and then click OK.

This means that the **Width** and **Depth** parameters are no longer linked together.

4. Select the extrusion onscreen and then on the ribbon, click the Edit Extrusion button.

⇨ Select the circle onscreen. On the Properties palette, check Center Mark Visible (see Figure 9 above).

   Deselect the circle, but do not finish the extrusion yet.

5. On the View tab, click the Visibility/Graphics button (or type VG) and then click the Annotation Categories tab.

⇨ Beneath the Dimensions category, check the Automatic Sketch Dimensions checkbox (see Figure 10 above) and then click OK.

Notice the two blue dimensions near the center of the circle. These are the Automatic Sketch Dimensions and by default have associated the center of the circle to the intersection of the two reference planes at the origin of this family.
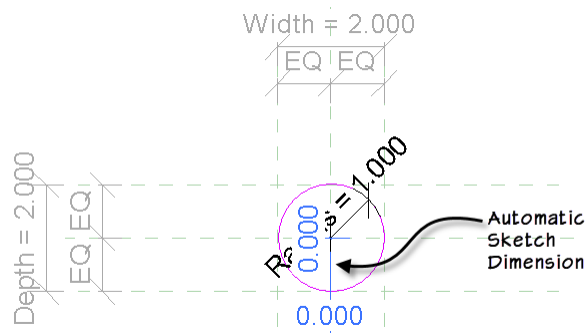

Figure 11—Automatic Sketch Dimensions display in a blue color in automatically determined locations

6. Return to the "Family Types" dialog and flex the **Width** and **Depth** parameters.

   As noted above, without the formulas, these two flex independently.

7. Flex the **Radius** parameter.

Page 9

Notice that the circle stays centered on the reference planes.

8. Click OK to dismiss the "Family Types" dialog.

Revit is always looking to establish logical relationships between elements. It does this by placing Automatic Sketch Dimensions in logical locations. In this case, Revit assumed we wanted to keep the center of the circle aligned to the reference planes at the origin (maintain a zero distance in each direction). This seems acceptable most of the time. For a circle, there aren't too many other logical assumptions to make, so constraining the center is a good bet. However, this simple example does not tell the whole story. Automatic Sketch Dimensions are *not* identical to constraints. They will adjust *on-the-fly* as you edit your family. Let's try another example to illustrate.

9. Drag the circle down about **0.500** units (it does not have to be exact, but if you drag more than .5, the Automatic Sketch Dimension will shift).

Notice that the vertical Automatic Sketch Dimension no longer reads zero, but now displays .5 (or whatever amount you dragged it). If you flex now, it will keep the center of the circle offset from the reference planes by this amount (not zero). Again, this may be your intent and may seem logical, but in some cases it may not be what you wanted.

10. From the File menu, Close the family without saving changes.

Let's return to our Ellipse family and see another example.

11. Open the file named: *A05_Ellipse (Sketch Dims).rfa* and tile the windows.

12. Open the "Family Types" dialog.

    Note that the formulas here have already been removed.

13. Flex the **Width** and **Depth** parameters.

Notice that this time, the object does not stay centered. Let's edit the extrusion and take a look at the Automatic Sketch Dimensions. They have already been turned on in this file.

14. Select the extrusion onscreen and then on the ribbon, click the Edit Extrusion button.

Notice that the Automatic Sketch Dimensions are attached to the left side horizontally and to the center vertically. Logical? Perhaps; perhaps not. The point is, that if you do not like the assumptions that Revit makes with the Automatic Sketch Dimensions, you cannot edit them directly. Instead you must add your own constraints and dimensions to make your desired intent known to Revit. As soon as you add a dimension or constraint of your own, the Automatic Sketch Dimension will be removed. You cannot simply delete them. You *must* add your own constraints or parameters to override (and therefore make unnecessary) the automatic behavior.

15. Select the ellipse onscreen. On the Properties palette, check Center Mark Visible (shown above).

16. Using the Align tool, align and lock the Center Mark to the Center (Left/Right) and Center (Front/Back) reference planes.

Notice that as soon as you lock the alignment, the Automatic Sketch Dimension disappears. The same is true if you add your own permanent dimension; even if you don't lock it. Any

dimension, constraint or parameter will have a higher priority than the corresponding Automatic Sketch Dimension and as a consequence will disable it.

17. Flex the ellipse after aligning and locking and note that it now flexes around the intersection at the origin.

18. Finish the Edit Mode, and close the Ellipse file. You do not need to save.

Automatic Sketch Dimensions are an important factor in being successful in the family editor. If you are unaware of them, it can make your work in the family editor frustrating as you make guesses on how to force Revit to behave the way that you need. There are a few schools of thought on the use of Automatic Sketch Dimensions. The first is to have the goal of eliminating them in all families. To do this, you would need to add your own constraints and parameters at all locations where Automatic Sketch Dimensions appear. This can be easy to accomplish in simple families, but in more complex ones, it can become quite difficult to achieve. So an alternate approach is to replace only those Automatic Sketch Dimensions that run counter to your design intent. In other words, if the family is flexing properly with the Automatic Sketch Dimensions, you can leave them alone. It is really up to you.

> TIP: In general, I prefer to eliminate the Automatic Sketch Dimensions where possible, but try not to become consumed by the task. If Revit insists on applying some Automatic Dimensions and they are not preventing my family from flexing properly, then I leave them alone. Your results may vary.

Working with Arcs

If you have successfully constrained, parameterized and flexed a circle or ellipse, you might next want to do the same for an arc. Arcs can be very similar to circles, but they do introduce an additional wrinkle; they have endpoints. If your center remains at a fixed location, arcs are pretty easy to control. If the center moves, you have the additional challenge of parameterizing the movement of the center. Let's consider the case where the center is fixed first.

Locking Endpoints

Arcs require you to constrain at least three geometric relationships. This can be any combination of the center, the angle, the length or the endpoints. Often however, the key to parameterizing arcs is constraining their endpoints.

1. Open the file: _02 Seed (Sketch Dims).rfa.

2. Save the file as: **Arc (Centered)**.

This file already has Automatic Sketch Dimensions enabled. There are no formulas in the "Family Types" dialog. To keep this example quick and easy, we'll build a model line. This way we can focus on just the arc instead of creating an entire form.

> Work in the *Ref. Level* plan view.

3. On the Create tab, click the Model Line tool.

⇨ On the Draw panel, click the Center-ends Arc tool.

4. For the center, click at the intersection of the Center (Left/Right) and Center (Front/Back) Reference planes.

⇨ Snap the first endpoint to the intersection of the Center (Front/Back) and Right Reference planes.

⇨ Snap the other point to the intersection of the Center (Left/Right) and the Back Reference planes.

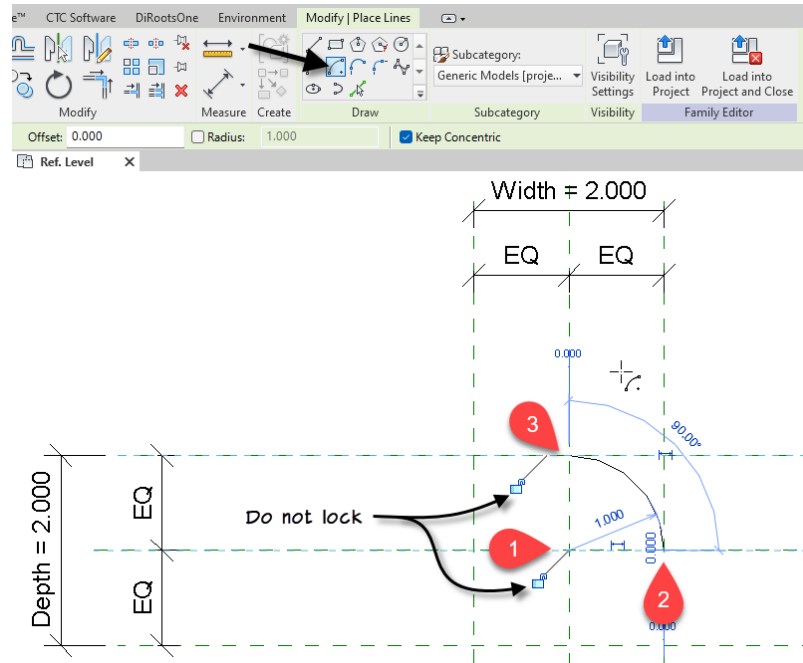Two lock icons will appear. **Do not** close them.

Figure 12—Create a Model Line arc centered on the reference planes

Notice all of the Automatic Sketch Dimension that appear. As noted above, the Automatic Sketch Dimensions try to anticipate how you intend to flex the object. They take into account the kind of geometry that you have. So in this case, since we have an arc, Revit is assuming we want to constrain the center point.

The more dimensions and constraints that you apply, the fewer automatic dimensions will be required. Of course there is a fine line we walk here. If you are not careful, it is easy to get the all too common "this would over constrain the sketch" error.

5. Add a radius dimension to the arc.

This removes one of the Automatic Sketch Dimensions. Let's add a parameter to the radius so we can flex the size of the curve.

6. Select the radius dimension and label it with the existing Depth parameter (see Figure 12).

Notice how the center remains fixed as the arc's radius increases. Revit tends to favor the center point location.
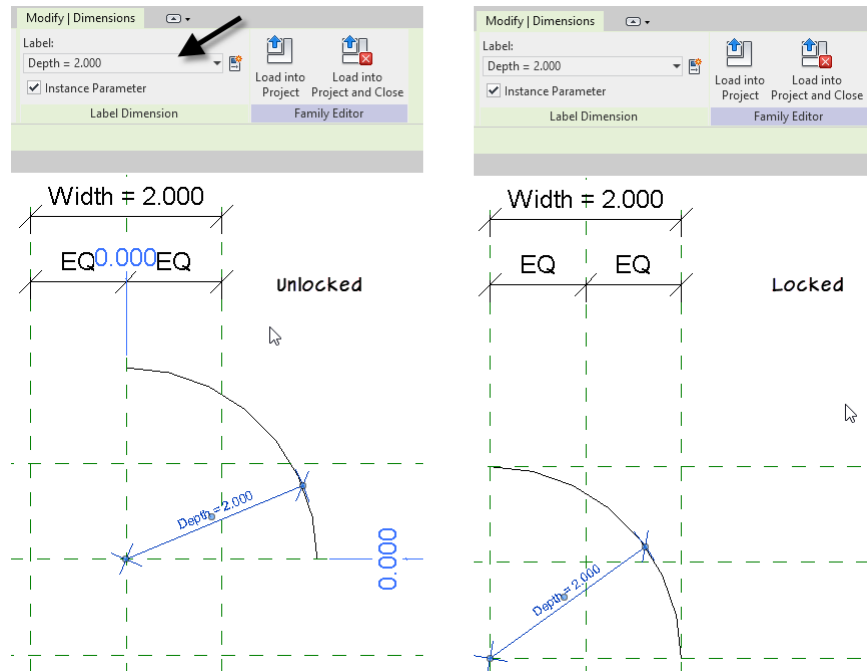
Figure 13—Relying on automatic dimensions vs. applying locks

7. Delete the arc and then add it again as above, but this time, **close the lock icons**.

8. Add the radius dimension and again label it with the Depth parameter (see Figure 13).

Notice that this time, since we applied the locks, the center moves when you flex it. Chances are, the first behavior was a little more expected than the second one. But potentially, both have their issues.

> TIP: I tend not to use the locks that Revit displays when drawing a shape. I use the Align tool instead to be more precise about where and what I am locking. With the Align tool, I see each lock being applied one at a time. I find this much more predictable then relying on the assumptions that Revit makes and offers.

9. Delete the arc and then add it one more time. Do not close the lock icons.

10. On the Modify tab, click the Align tool.

⇨ For the reference for alignment, click the Back reference plane.

⇨ For the entity to align, click the endpoint of the arc. Lock it (see Figure 14).

Alternatively, there is a Lock checkbox on the ribbon (see the upper-right corner of Figure 14). You can click this to align and lock in a single step.
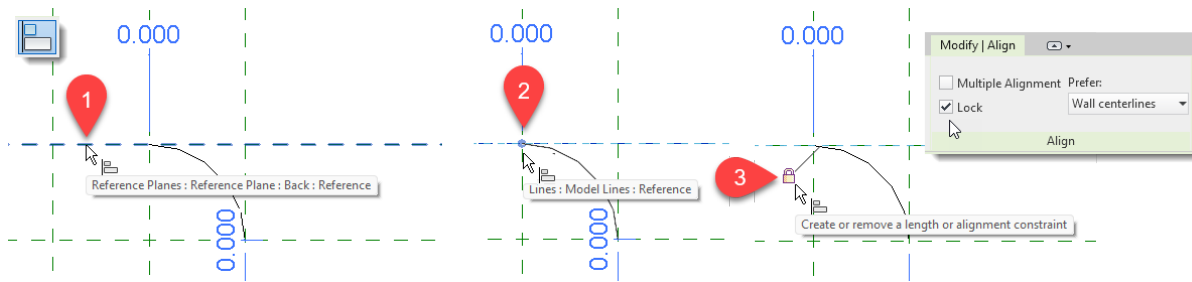
Figure 14—Align and lock endpoints

11. Repeat on the Center (Left/Right) reference plane and the same endpoint and again in both directions for the other endpoint. (This will be four alignments total).

Notice that as you finish aligning in all four directions, all of the Automatic Sketch Dimensions will disappear. The endpoints of the arc will now remain at the intersections of the reference planes when you flex.
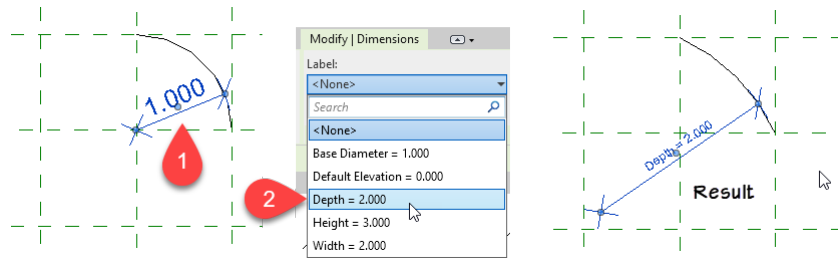
12. Add the radius and label it with the Depth once again (see Figure 15).

Figure 15—After aligning and locking endpoints, apply the Depth parameter to the radius; Did it flex as expected?

Were the results as you expected? This time, the center point moved. Considering that we just locked the endpoints to the intersections of the reference planes, there is really nothing else that could flex. I suspect however, that this may not be what you expected. It is likely that we would want the center to stay at the intersection of the two central reference planes and instead for the width and depth of the arc to flex. We can achieve this with a simple formula.

13. Undo the last step (the application of the label to the radius), but keep the radius dimension.

14. Label it with a new instance parameter instead and call it: **Radius**.

15. Open the "Family Types" dialog and add a formula to both `Width` and `Depth`: **Radius * 2**.

16. Flex the family.

Notice that now when you flex the radius, it stays confined to the upper right quadrant defined by the reference planes. So, depending on the specific behavior you require, the precise approach you use will vary and is a matter of preference.

You can also try dragging the arc's endpoints to produce a different angle. 90° increments should be very stable. But even other angles should work well since the main controlling parameter here is the radius. To add even more stability, turn on the center mark (see above) and then align and lock the center in both directions to the reference planes. With both center

and endpoints locked, you don't actually need the `Radius` dimension. The `Width` and `Depth` and flex the arc and the alignments will be maintained. Finally, all of this would also work with an elliptical arc as well. You can align and lock the endpoints and the center and let the `Width` and `Depth` parameters vary instead of forcing them equal with the formulas. Feel free to experiment further before continuing.

CATCH UP! You can open a file completed to this point named: *B01_Arc (Centered).rfa* and an alternate version called: *B02_Elliptical Arc (Centered).rfa*.

Arches

Look around the town you live and you are bound to find many different types of arches used in the various buildings you see. Each of these incorporates arcs to help form their shape. If you want to create one of these very common forms in your architectural project, you will need to ensure that when the opening that uses the arch flexes, that it does so predictably. The techniques covered with the arc examples above will work for some of the following arch forms, but not all (see Figure 16).
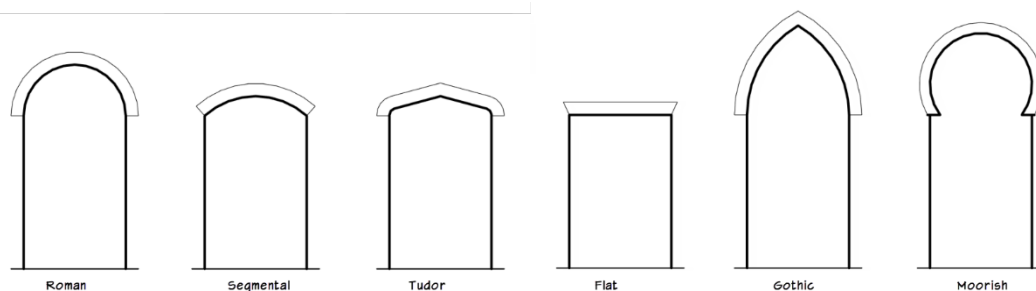
Figure 16—A selection of common arches used in architecture

These are just some of the more common arch forms. There are many more. Let's consider the ones shown here briefly. Starting with a Roman arch. A Roman arch is a semicircle, so the techniques discussed above will work well. Just turn on the center mark, align and lock it to reference planes and then lock the endpoints of your arc on both sides as well. You can flex its shape with standard width and depth parameters. You can also consider renaming depth to height and for a Roman arch, depth with always be half the width. You can and add a radius parameter (which would be equal to the height) if desired; but it is not necessary. The Moorish arch is actually very similar only its radius is less than its height (or depth). The flat arch is simple as there is no curve, so normal family editor techniques will work fine there. Similarly, the tutor arch really only has curves at the corners, so techniques that we will learn later on would work well for those, the rest of it is standard family editor stuff. That leaves the gothic and segmental. Let's address segmental first, which like Moorish, is similar to the Roman except that the radius is larger than the arch height. Once you understand how to make the segmental, to make a Gothic you simple place two segmental arches next to each other at opposing angles.

Naturally, we won't have time and space to do all of these in detail. Let's consider a foundational example, that will give us a framework which can be applied to many of these.

Create a Simple Parametric Arch

It can sometimes be tough to make an arch behave as it flexes. In this example, let's consider a simple arch that uses a segment of a circle. As noted above, if the radius of this arc is less than the height of the arch, then you will have a Moorish shape. If the radius is equal to the height (half the width) it will be a Roman arch, and it is more than the height, you will have a segmental arch. The main issue with such an arch is that the center point moves as the arch flexes. So since the techniques above relied on locking the center point, those techniques will not yield good results here. Now you certainly could add a reference plane for the moving center location and then use the techniques above, but flexing that reference plane into the correct location would require some more advanced formulas. However, there is an easier way. If we simply break our standard family editor rule about keeping constraints applied only to references (like the circle and ellipse above) we can easily get the behavoir we need. *Sometimes the key to success is in applying the labeled dimension directly to the geometry of the curve* rather than the traditional approach of dimensioning the reference planes and then letting them flex the geometry.

> **Note:** It is possible to stay true to the traditional approach and dimension only references. It would require formulas (often utilizing trigonometry) to calculate where the references need to be to give the correct curve. We will see examples of this below.

One last preliminary point. The following techniques work with any family template, but I think that in many cases, you will find a face-based or wall-based template a good choice for an arch. This makes it easy to place them on or in existing walls.

1. From the File menu, choose **Open > Family**.

2. Select the file named: *_03 Seed (Face Based).rfa* and then click Open.

3. Save the file as: **Simple Arch**.

Our seed families all have the two default reference planes at the center. In this case, let's use the Center (Front/Back) reference plane (the horizontal one in the middle) as the spring line of the arch, so we'll need to delete the bottommost one and rename the center one.

4. Delete bottom reference plane. Also delete the existing extrusion.

5. Select the Center (Front/Back) reference plane[2] and rename it to: **Spring Line**.

   ⇨ On the Properties palette, change its Is Reference setting to: **Bottom**.

When you do this, you will be left with a single unlabeled vertical dimension.

6. Select the vertical dimension.

   ⇨ On the Options Bar, label it with a new instance parameter and call it: **Rise**.

7. Flex the family to ensure everything is working. Set the Rise to: **0.750** (see Figure 17).

---

[2] If you are not able to select the Center (Front/Back) reference plane, please ensure that "Select pinned elements" is enabled. You can do this using the icon at the lower-right corner of the application frame, or the drop-down menu beneath the Modify tool.
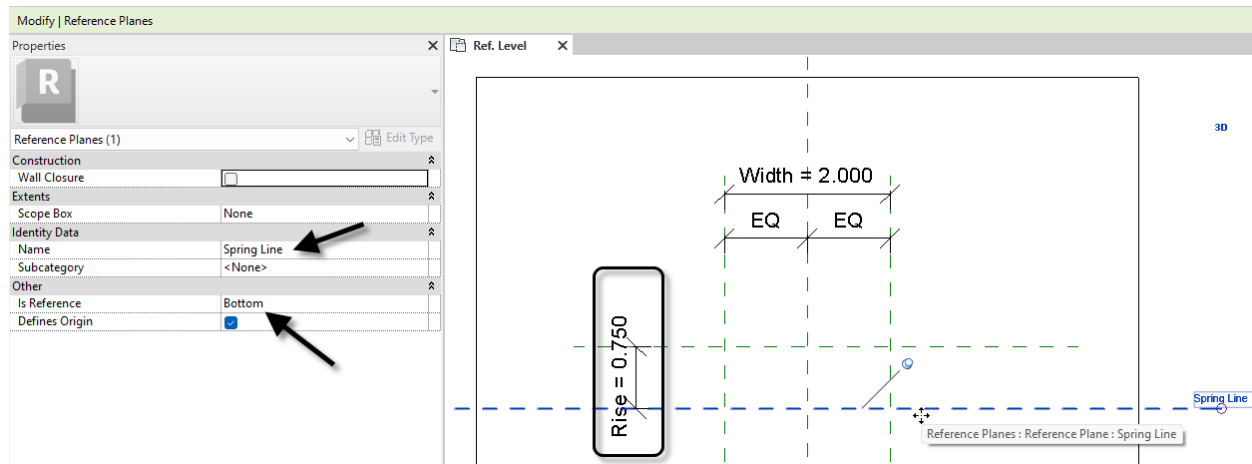
Figure 17—Configure the reference planes

With this framework in place, we are ready to build our geometry. While it is possible to use an extrusion here, it takes a little more effort to build and constrain one. This is because an extrusion would require two parallel curves; each needing to be constrained properly. Instead, let's create a sweep. The sweep is simpler because it is the path of the sweep that will form the arch shape. Then the solid form of the sweep will follow this path.

8. On the Create tab, click the Sweep button. Choose the Sketch Path option on the ribbon.

9. From the Draw panel, choose the Start-End-Radius arc.

10. Snap the start and end points to the intersections where the left and right reference planes cross the Spring Line reference plane (see points 1 and 2 in Figure 18).
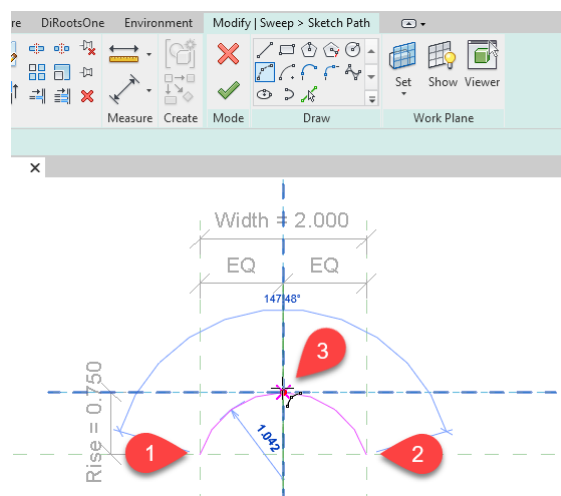
Figure 18—Create the path of the sweep with a start-end-radius arc

⇨ Snap to the intersection of the Top and Center (Left/Right) reference planes for the radius (see point 3 in Figure 18).

Page 17

11. Using the Align tool, align and lock the start and end points of the arc to the reference planes in both directions. (Four total). Use the tab key where necessary to ensure you are clicking the endpoint of the sketch line and not the dimension (see Figure 19).
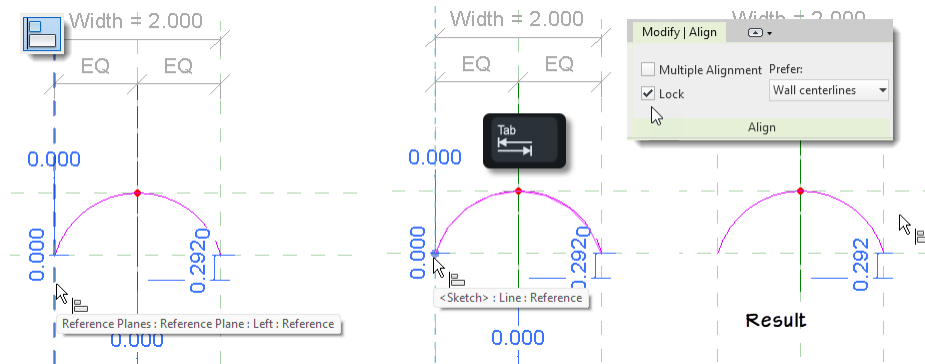
Figure 19—Align and lock the endpoints of the arc to the reference planes in both directions

Aligning and locking works fine for the endpoints. But notice that one automatic sketch dimension remains to the center of the arc which does not fall on any of our reference planes. We could use formulas to calculate what this dimension ought to be, but there is an easier way. It is the midpoint of the arc that we care about here. We always want the midpoint of the arc to stay with the Top reference plane. Unfortunately, we cannot align and lock the midpoint of an arc. Instead, we'll dimension the sketch line directly; like we did for circles and ellipses above.

12. Create an aligned dimension. For the first witness line, click the Spring Line reference plane.

⇨ Click directly on the arc next to add the second witness line.
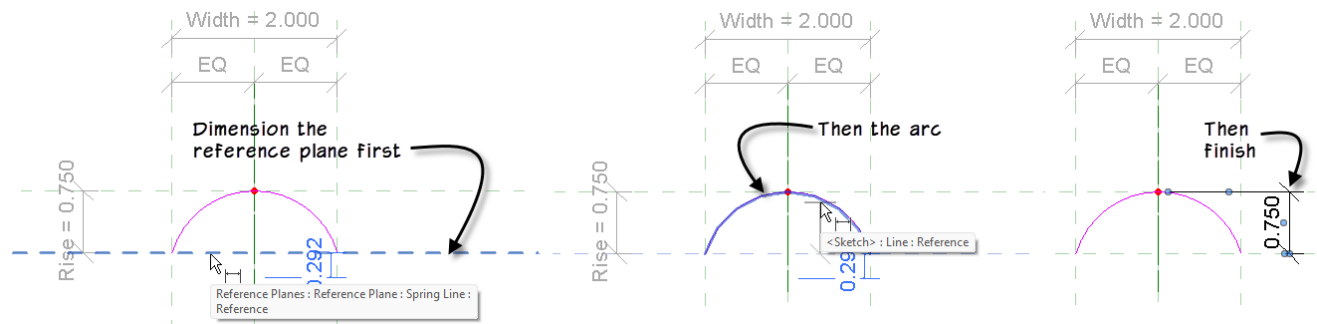
Figure 20—Dimension the arc directly

13. Label this dimension with the existing **Rise** parameter.

⇨ Flex the family.

That's it! The sweep can now have nearly any rise value you like to create arches of various shapes and proportions. In fact, this one construct can actually create three of our arch forms noted above. Keeping the **Rise** value smaller than half of the **Width** gives a segmental arch. If the value of **Rise** is equal to half of the **Width** parameter, you will get a Roman arch. If you go larger than this, it will make an arch that has a Moorish shape. While tempting to try a zero

value for Rise to get a flat or jack arch shape, an arc must be curved, so you cannot use a value of zero. Therefore, to make a jack arch simply make a separate family with a rectangular form.

To complete the arch, just edit the sketch for the sweep profile and create a simple rectangle. If you prefer, you can use a more complex shape, but a rectangle will do the trick for now. We'll do a more complex arch later on. Try different variations to ensure it is working properly (see Figure 21).
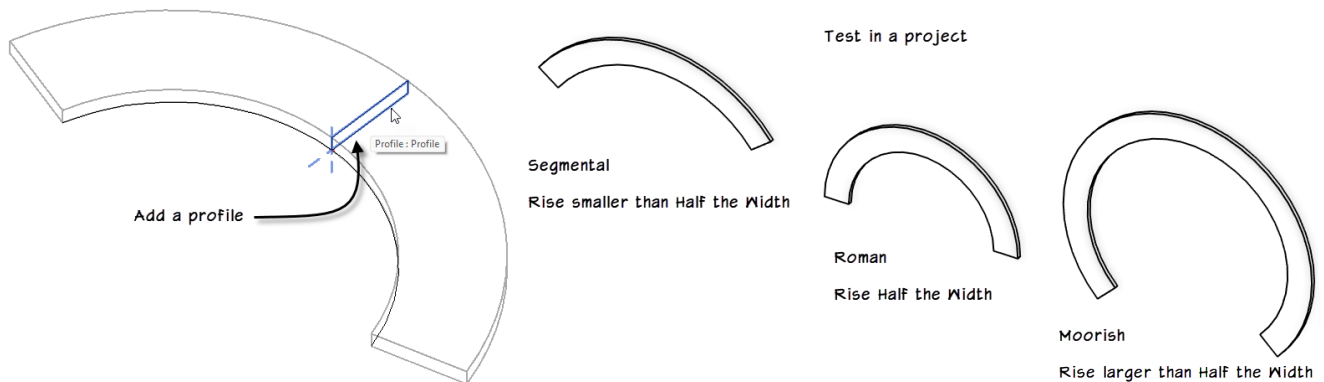

Figure 21—Complete the arch family and load it into a project to test

CATCH UP! You can open a file completed to this point named: *B03_Simple Arch.rfa*.

Profile Families

This topic will consider profile families. There are several advantages of profile families. Perhaps the most important benefit of profile families is that we can reuse them in several forms and families. For the task of creating complex flexible curved forms, it is much easier to build a profile family and get the shape flexing properly, and then apply it to any 3D forms required in the same or multiple files.

The remaining examples in this paper will use profile families.

Some challenges do exist. One is that profile families can contain reference planes, but not reference lines. Therefore, if you need to use reference lines like for parametrically controlled rotation (which typically uses reference lines), you will need an alternate approach when using profile families. There are some acceptable solutions. For example, you can use trigonometry to derive X and Y coordinates from any angle. This is very effective and very stable, however, trig can be challenging and if you need many formulas it can have a detrimental effect on overall performance. Another solution that we will explore is using a nested rig family. Nesting can also be detrimental to performance, so you will have to consider each use case carefully. We'll see some examples below.

Another issue is that only two of the five solid forms used in the family editor support profiles: Sweeps and Swept Blends. So if you choose an extrusion, blend or revolve, you will need to sketch the form instead of using a profile. So careful planning is required. The good news is that most forms that you can create with extrusions, revolves and blends can be easily emulated using sweeps and swept blends. So if you are committed to using profiles, favoring sweeps and swept blends over other forms should not be too much of an issue in most cases.

> **Note:** While we are not performing any examples in the massing environment in this paper, it should be noted that profile families are not available in the massing environment. The massing environment supports only 3D geometry and families. Therefore, to achieve the same result, create any "profiles" in a Generic Model family (unhosted or face-based—both work) and then draw the profile shapes using model lines. When loaded into a massing or adaptive family, a family created this way will be functionally equivalent to a traditional profile family.

Ovolo

Let's start with a practical profile family example and build some common molding shapes. In his excellent book on classical architecture: *The Classical Orders of Architecture,* author Robert Chitham details the construction of many common architectural moldings. (Chitham's book was a major resource for my book: *Renaissance Revit* from which most of the content in this handout is deriven). Below I have illustrated the first molding shape we will consider: the ovolo (see Figure 22). It uses a single arc curve, but constructing it introduces a challenge (an offset center point) that once solved, will help us create the more complex compound curves that follow.
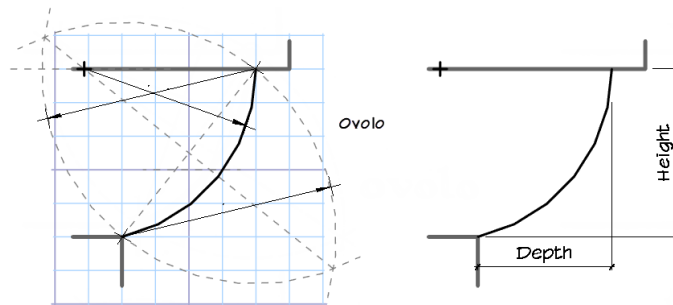
Figure 22—The Ovolo profile (analysis based on Chitham)

Consider the illustrations shown in Figure 22. The left side comes from Chitham and shows how to construct the curves using traditional hand drafting tools. A pair of diagonals are struck and this helps determine the center point of the arc required. The most likely way to describe the size of a molding like this and therefore the most likely way to flex this curve in Revit would be to ask for the depth and the height of the curved portion of the molding. This is shown on the right side of the figure. This height and depth of the curve are normally not equal, (A quick trip to your local lumber yard will allow you to verify this) meaning that the construction axes are along angled paths as the dimensions flex. This also means that the center point is not always in the same location. Typically, reference lines are the obvious choice for controlling angles in families. However, as noted above, reference lines are not available in profile families. We could abandon our use of profiles and resort to building each molding with its own integral sketch directly within the solid geometry, but this would not be ideal since we would lose all benefits of using profile families. One of the most important being that they can be reused in multiple families. Therefore, if we can define our required shape in a well formed flexible profile family, we can reuse it in almost unlimited ways in other content. So to build it as a profile family, we will have to resort to other techniques to control the shape of this family and ensure that it flexes properly. This means using some other method to control angles that does not rely on reference lines.

Using trigonometry to model traditional molding forms

In this example, we will turn to trigonometry for our solution. Depending on the complexity of the profile you are creating, the trig might be simple or complex. In this case, we have a simple case of similar triangles (shaded in gray in Figure 23). You will be asking your user to input the desired height and depth of the molding. We will call these **x** and **y** in the profile family (items 1 and 2 in the figure). When you make a triangle from **x** and **y**, you can easily derive the hypotenuse which we will call: **D** (for Diagonal–item 3). Using trigonometry, and these values, we can easily derive one of the angles of the triangle which we will call: **A**. Since we know all three sides, we have lots of options to choose from. For this example, I went with the **ATAN** function performed on **x** and **y**: **ATAN(Y/X)** to arrive at: **A** (item 4). This new angle, coupled with half of **D** (we'll call this: **HD**) (item 5) can be applied to the second (similar) triangle. Finally, with this information, the **COS** function will give us the length of hypotenuse of the smaller similar triangle, which also happens to be the radius: **HD / COS(A)** (item 6).
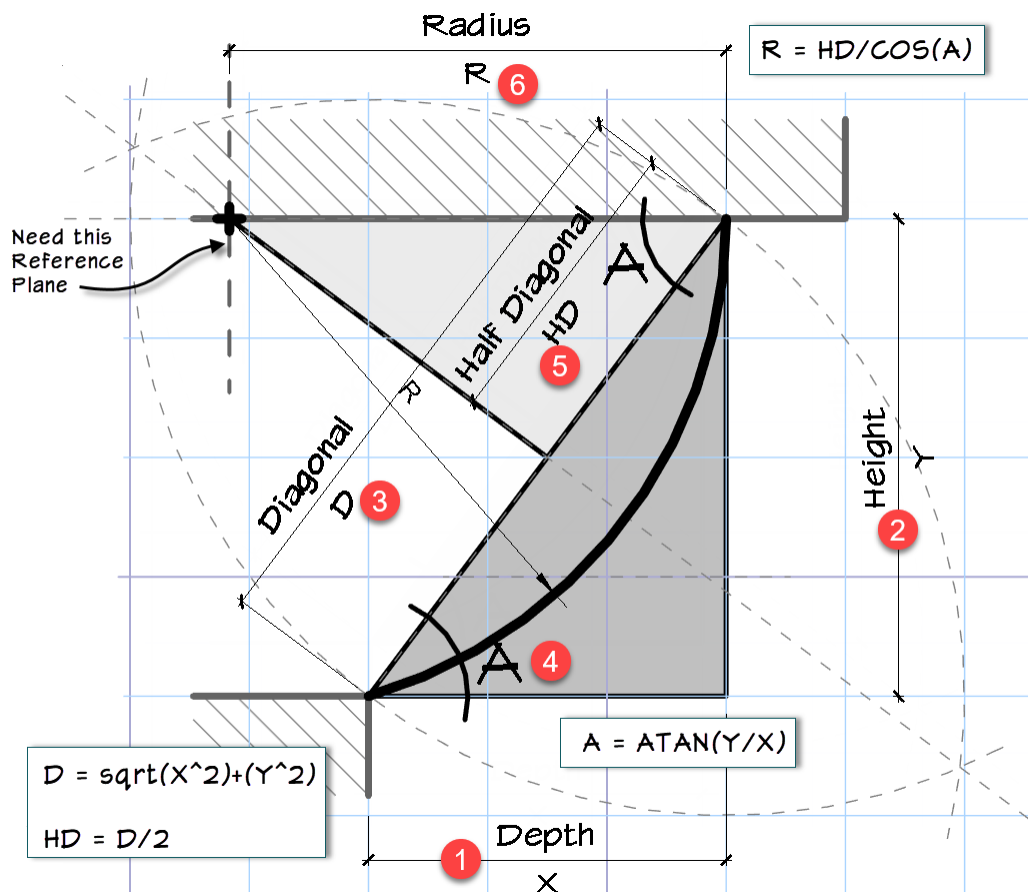
Figure 23—Using trigonometry to locate the required reference planes

Provided with the dataset is a seed file for a profile family. It has the two default center reference planes. Two additional pairs reference planes named: X Min and X Max, and Y Min and Y Max are placed above, below and to the sides of these. They are dimensioned and constrained equally already. The insertion is at the intersection of Y Min and X Min (lower left corner). To this framework, we need to add a vertical reference plane to left for the radius location indicated in the figure and our formulas to help us locate it.

1. Open the file named: *_04 Seed (Profile).rfa* and then Save the file as: **Ovolo Profile**.

2. To the left of the X Min reference plane, draw a vertical reference plane. Name it: **Radius**.

⇨ Add a dimension between the X Max reference plane and the Radius reference plane.

⇨ Label it with a new Type parameter called: **Radius**.

> **Note:** At the bottom of the "Parameter Properties" dialog, you can click the Edit Tooltip button and input a descriptive tooltip for the parameter. So for `Radius`, you might input something like: "Calculated radius of the ovolo arc." Tooltips are optional, but they can be helpful if you take the time to input them.

3. Open the "Family Types" dialog.

4. Using the New parameter icon (at the bottom of the dialog), create two new Type-based Length parameters: **D** and **HD**. Also create one Type-based Angle parameter called **A**.

   Group all of these under Constraints.

5. Input the formulas as shown in Table 1 and in Figure 24:

Table 1

Parameter	Formula
D	$\sqrt{(X^2) + (Y^2)}$
HD	$D/2$
A	$\mathrm{atan}(Y/X)$
R	$HD / \cos(A)$

6. Flex the family by creating a few Family Types.
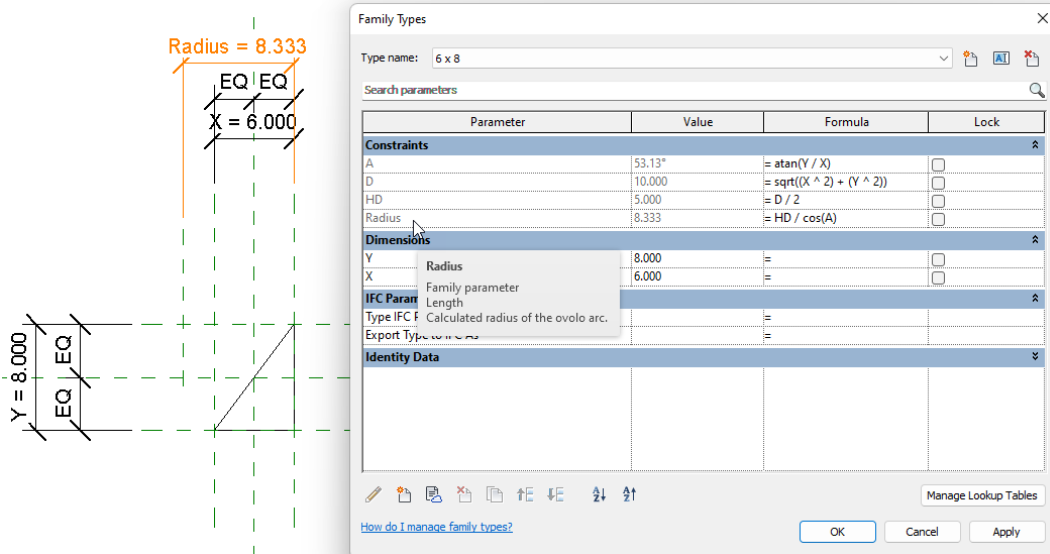
Figure 24—Create the parameters and assign the formulas

There is a simple triangle in this family already just to facilitate flexing. We can now delete this and draw the actual profile shape that we need.

7. To the left of the Radius reference plane, draw a vertical reference plane. Name it: **Projection**.

 ⇨ Add a dimension between the X Min reference plane and the Projection reference plane.

 ⇨ Label it with a new Type parameter called: **Projection**.

8. Delete the triangle onscreen.

CATCH UP! You can open a file completed to this point named: *C01_Ovolo Profile (Trig).rfa*.

9. On the Create tab, click the Line tool.

10. Draw a Center-ends arc with the center at the intersection of the Y Max and Radius reference planes.

    Be sure to turn on the center mark, align and lock the center in both directions as well as the endpoints.

11. Draw vertical and horizontal lines locked to the reference planes for the leftmost edge (Projection reference plane) and the top and bottom (see the left side of Figure 25).
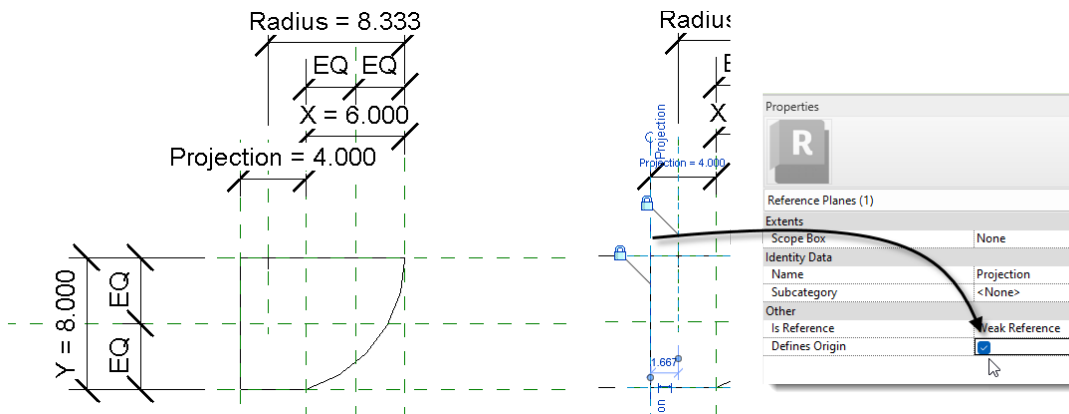
Figure 25—Create the profile lines

 ⇨ Select the Projection reference plane and then on the Properties palette, check the Defines Origin checkbox (see the right side of Figure 25).

CATCH UP! You can open a file completed to this point named: *C02_Ovolo Profile (Trig).rfa*.

12. Flex the completed version when done.

You can load this profile into any project or family now and use it shape anything from sweeps, to wall profiles to railings. A simple family file is provided here to test it out.

13. Open *_05_Sweep Flex.rfa*.

A version of the profile family is already loaded, but if you prefer, you can load your version instead. The pre-loaded version contains a family type we will use later, but for now, you can create custom types to flex the profile to any size you like.

14. Select the sweep and edit it. On the ribbon, click the Select Profile button and then from the drop-down list, choose one of the types you previously defined (Figure 26 shows the 6 x 8 type applied).
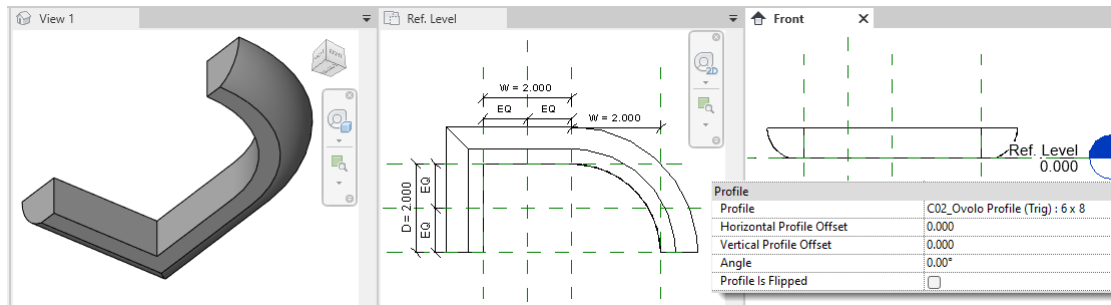
15. Finish the sweep and flex the family.

Figure 26—Apply the profile to the sweep form

This file is serving as a simple "sandbox" file. We are using it here to test out the profile and ensure that it works as expected. The family has two simple parameters: W and D. If you flex them, you will see the shape of the sweep path adjust. You can see by doing this how your profile follows along both straight and curved path edges. If you want to change the profile, edit the sweep again and pick a different type from the list. We can even edit the type of the nested profile family and link up the X and Y parameters with driving parameters here in the host family. I will not go into the steps at this time, but feel free to experiment with this if you like. Later, we will return to this file and add to it. But for now, there is no need to save the sandbox.

16. Close both family files.

Cavetto

In this example, we will look at the cavetto curve. It is exactly the same construction as the ovolo with the curve reversed. So the radius reference plane needs to be on the right with the center of the curve at the intersection of the Radius and Y Min reference planes. In all other ways, we can use the same strategy and formulas from the previous example (see Figure 27).
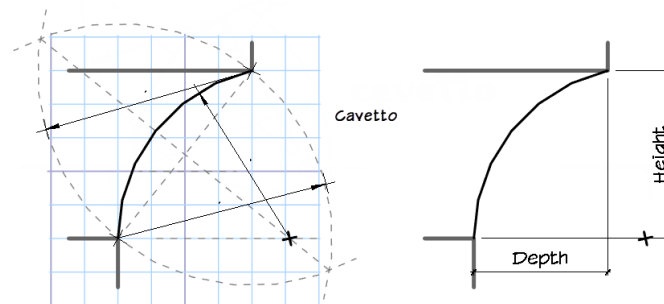
Figure 27—The Cavetto profile

If you wish to try your hand at creating this profile, you can follow a procedure similar to the one above. To save time, a family has already been provided for the Cavetto profile. If you wish, open the family now: *C03_Cavetto Profile (Trig).rfa* and study how it is constructed (see Figure 28).

EQ EQ

X = 6.000

Projection = 2.000

Y = 8.000

EQ EQ

Radius = 8.333
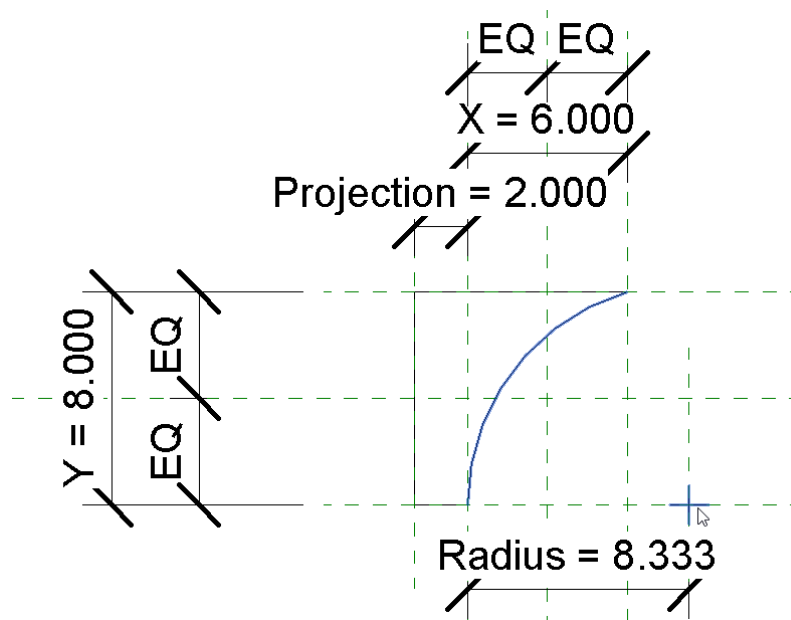
Figure 28—The Cavetto Profile family and its Projection parameter

Cornona

The curves are certainly the important part of these profiles and the focus of this paper. However, to make the profiles more useful as standard molding shapes, you can add other details that frequently accompany such molding profiles. In this example, we'll start with the Cavetto family considered in the previous topic and add a fascia and fillet (simple flat) molding to it to create a Cornona molding.

1. Open the file: *C03_Cavetto Profile (Trig).rfa*.

2. Save the file as: **Cornona Profile**.

3. Select the Y Max reference plane and copy it up a small amount. The exact distance is not important.

⇨ Repeat by copying Y Min down a small distance as well.

⇨ Rename each of these new reference planes to: **Fillet Max** and **Fascia Min** (see Figure 29).
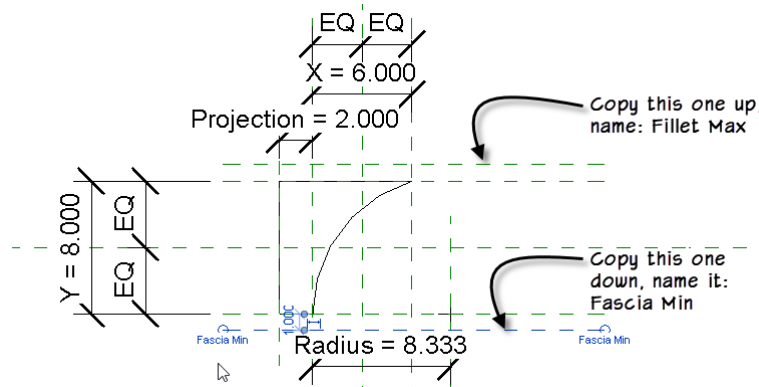
Figure 29—Copy and name two reference planes

4. Add dimension between Y Min and Fascia Min. Repeat with another one between Y Max and Fillet Max.

5. Label each of these dimensions new parameters: **Fillet** and **Fascia**.

⇨ Open Family Types and flex to be sure they are functioning properly (see Figure 30).
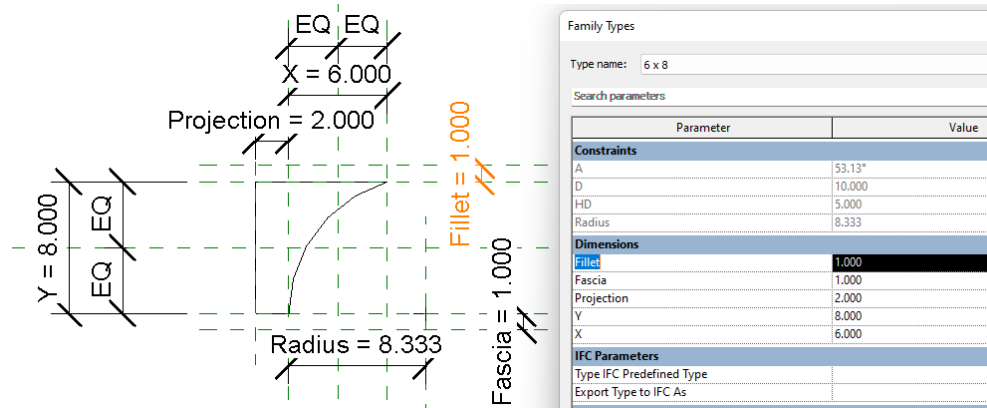
Figure 30—Add new parameters

We now need to add a few new lines to the profile. The two horizontal lines already in the family need to be associated to the new fillet reference planes instead of their current locations. However, if you try to move them or align them, they will attempt to remain connected to the curve causing errors. You can either manually stretch the endpoints to disconnect them from the curve and answer OK when prompted, then align and lock them to the required reference planes, or simply delete them and redraw them. I'll delete and redraw here.

6. Delete both horizontal lines.

7. Draw new horizontal and vertical lines aligned and locked to their respective reference planes as shown on the left side of Figure 31.

8. Use Trim/Extend to Corner to close up the profile shape. Align any element as required to remove any Automatic Sketch Dimensions (see the right side of Figure 31).
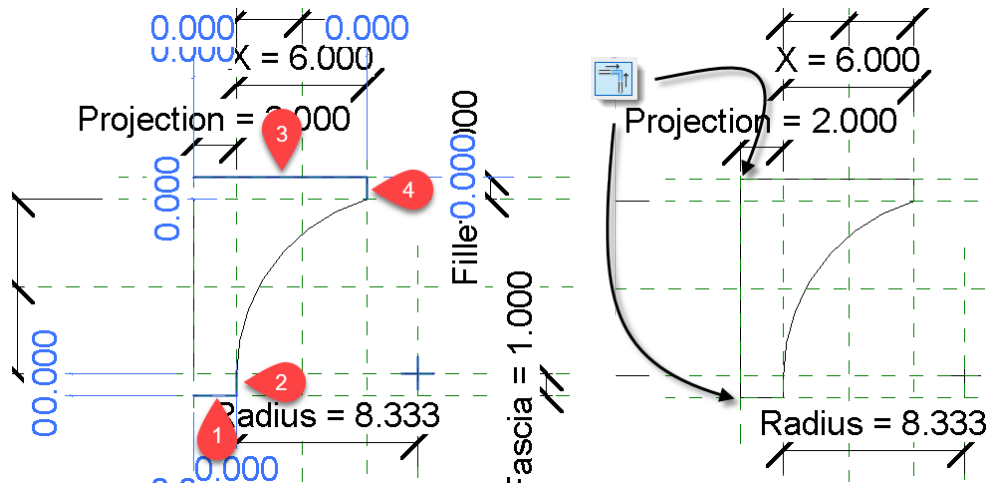
Figure 31—Add new lines, align lock and trim/extend the corners

⇨ Open Family Types and flex the family to ensure it is functioning properly.

CATCH UP! You can open a file completed to this point named: *C04_Corona Profile (Trig).rfa*.

To test it further, you can reopen the sandbox file: *_05_Sweep Flex.rfa* that we used above. The compled version of the profile is already pre-loaded in this file, or if you prefer you can load your own (see Figure 32). No need to save this file after flexing.
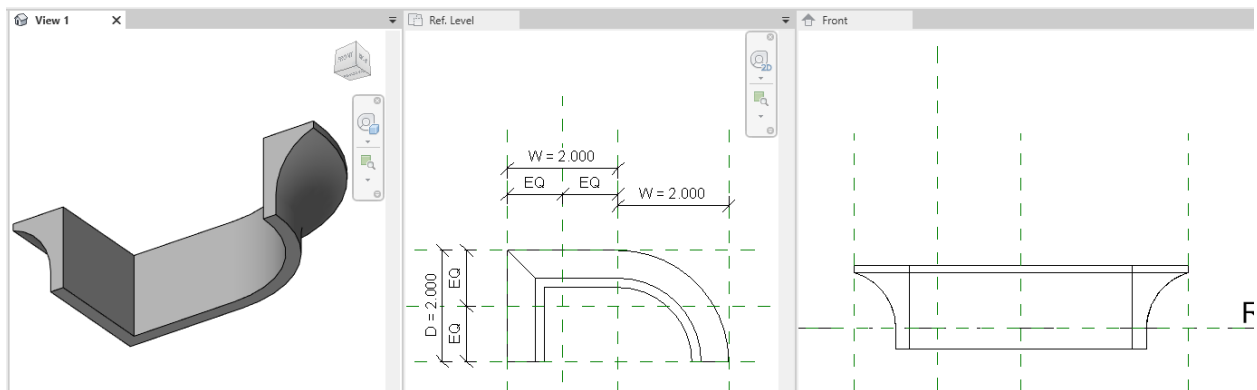
Figure 32—Test the new profile in the sandbox file

Complex Curves and Compound Curves

All of the examples covered so far were single curves; in other words, there has only been one curve that we were trying to flex. In such cases, if you ensure that you constrain and/or parameterize the key geometric aspects of the curve, you will usually get good results. For example, with a circle or arc, if you constrain the center and radius, it will usually flex properly. With an ellipse, the center and axes usually give good results. However, as the forms that you wish to flex become more complex in shape, sometimes these approaches will not be enough.

Consider situations where there is more than one arc segment making up a compound curve. Or even situations with a curve meeting a straight line at a tangent. There are endless possible

examples. In this topic we'll consider a common example. In similar fashion to the examples above, the key is going to be carefully constraining the curves so that you remove any ambiguity. **You want to make it very clear to Revit what your intentions are.** If you do this, everything *should* flex properly.

Cyma

Now that we have the basic techniques we need in hand, let's try our first compound curve. The next figure shows two very typical molding profiles: the Cyma and Cyma Reversa (see Figure 33). As before, the left illustration comes from Chitham and shows how the profiles are constructed traditionally, a circle is created whose diameter matches the diagonal between the X and Y (Height and Depth). This circle is intersected with two arcs of the same radius which, when intersecting perpendiculars are drawn, creates four equal segments along the diagonal. The points where the arcs intersect the circle are the centers for the arcs used to create the Cyma. Once again, our user inputs will be Height and Width of the molding (which will be parameters: **X** and **Y**).
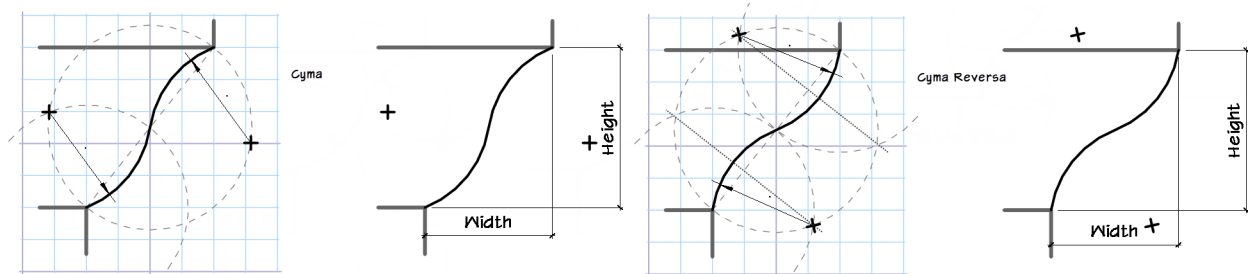
Figure 33—The Cyma and Cyma Reversa profiles

When looking for how to translate the traditional hand-drawn approach using a compass to something that would give us solvable triangles and yield the reference plane locations in Revit, I made a happy discovery. It just so happens that this construct also creates a regular hexagon whose vertices intersect the circle at the same locations as the intersecting arcs—compare the gray dashed construction arcs with the superimposed hexagon (see Figure 34).
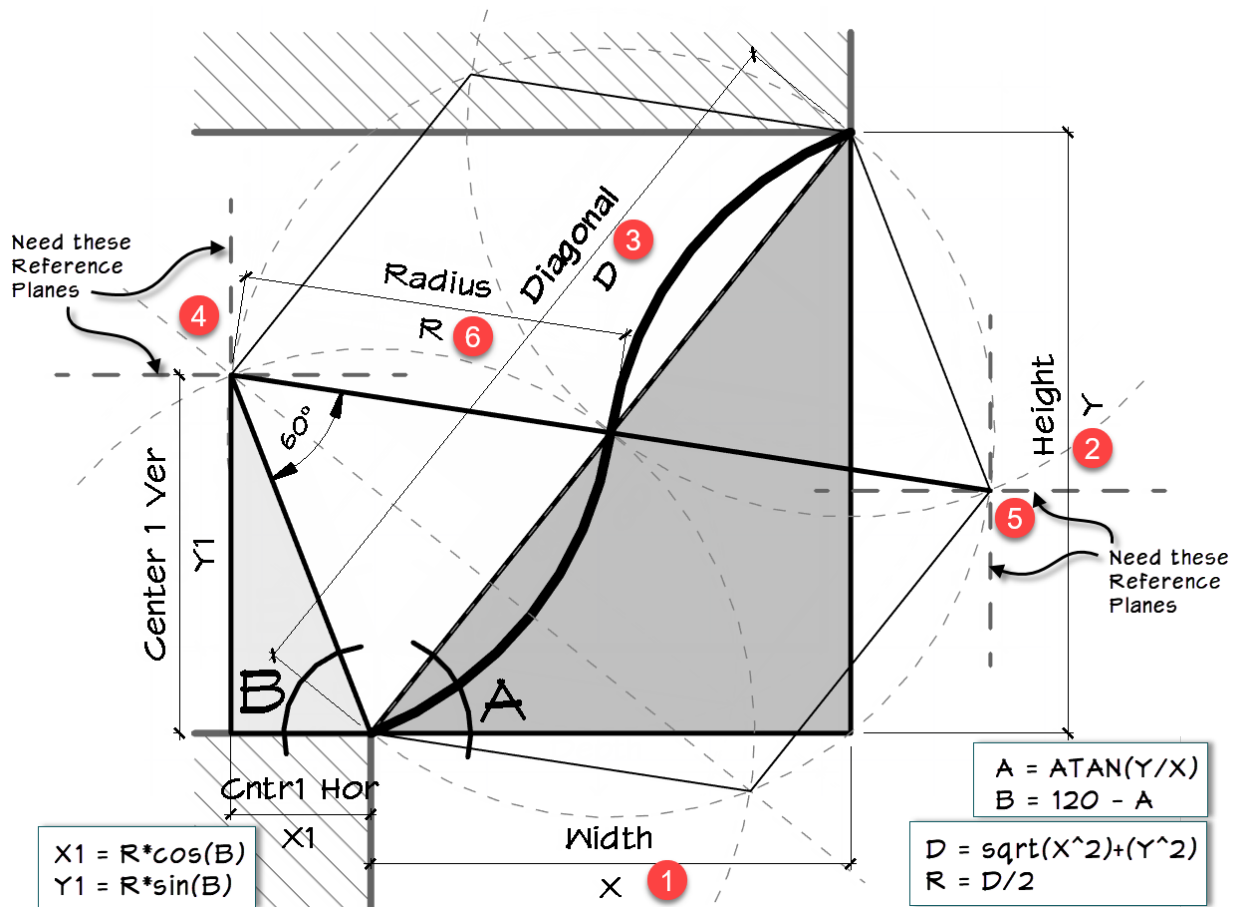
Figure 34—Applying trigonometry to locate the required Cyma reference planes

A regular hexagon can be divided into six equilateral triangles. The sides of these triangles are each equal to half the length of the diagonal (between **X** and **Y**). This distance (the side of the equilateral triangle) is also the radius of the arcs used for the cyma and cyma reversa profiles. We can use trigonometry to construct this profile family. The trigonometric formulas are shown in the figure and in the table below.

The basic idea is that the user input is the Height (Y) and Width (X) (items 1 and 2 in Figure 34). This is used to determine the angle of the diagonal (D) (item 3) which is in turn used to locate the center point of the two arcs of the compound curve (items 4 and 5) and their radii (item 6). We'll start with a file similar to the ovolo example above.

1. Open the file: *D01_Cyma Profile_Start.rfa*.

2. Save the file as: **Cyma Profile**.

Some of the work has been done here already. To create this file, a copy of the Ovolo file (created above) was saved. This means that some of the reference planes and some of the formulas were already in place. This includes the insertion point at the lower left corner, the X Min, X Max, Y Min and Y Max reference planes, and the parameters **X**, **Y**, **A**, **R** and **D**. (The parameter **HD** was not needed and has been removed). Additional reference planes have been

added: Arc Center Right Ver, Arc Center Right Hor, Arc Center Left Ver and Arc Center Left Hor, the parameters **X1** and **Y1** are applied to these reference planes already. These additional parameters are listed in the table below.
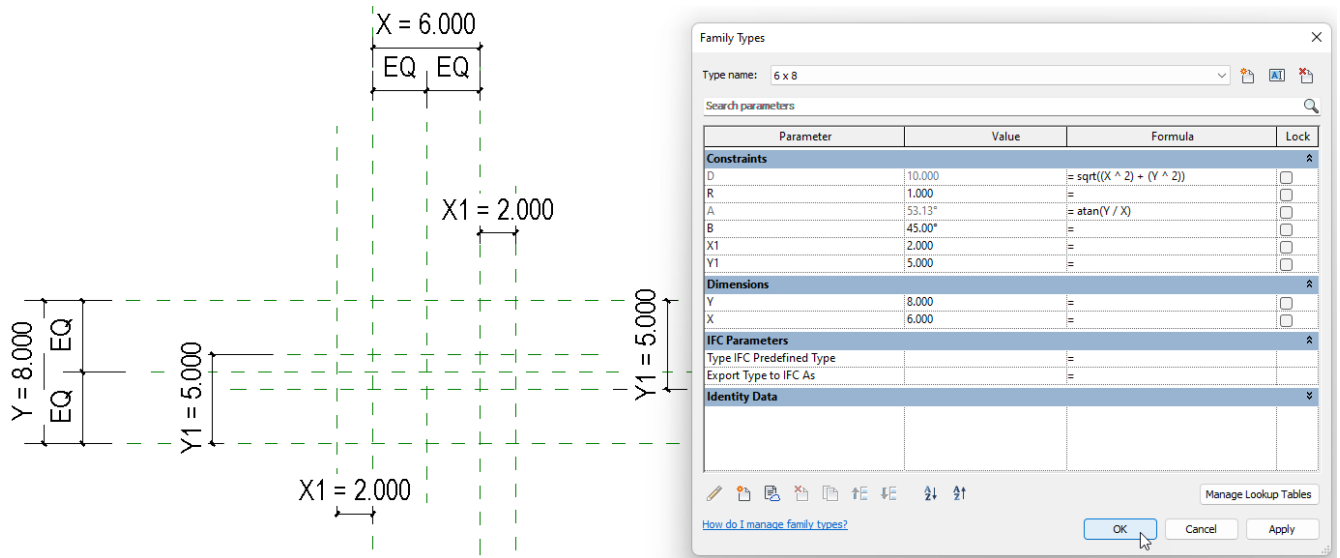
Figure 35—Starting family contains the reference planes and parameters

The formulas have not yet been added except those that came from the ovolo family. We will add them now.

3. Open the "Family Types" dialog. Using the following table, input the formulas shown for each parameter.

Table 2

Parameter	Formula
D	sqrt((X ^ 2) + (Y ^ 2))
R	D/2
A	atan(Y / X)
B	120 – A
X1	R*cos(B)
Y1	R*sin(B)

4. Click Apply to test the values.

The reference planes should adjust slightly.

5. Try flexing with each of the types in the family already. Return to the 6x8 Type and then click OK when finished.

6. On the Create tab click the Line tool and then click the Center-Ends arc icon.

⇨ Snap the center of the arc to the intersection of the Arc Center Left Hor and Arc Center Left Ver reference planes (see point 1 in Figure 36).

7. Snap the first endpoint to the intersection of the X Mid and Y Mid reference planes and the other to the intersection at Center (Front/Back) and Center (Left/Right) (see points 2 and 3 in Figure 36).
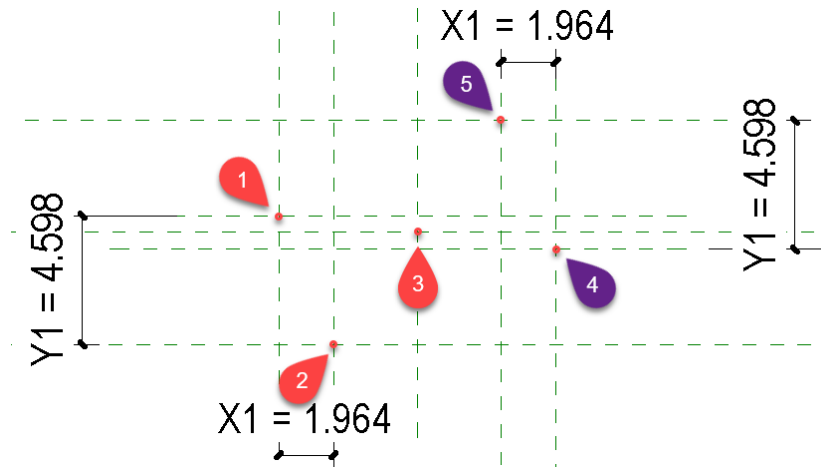
Figure 36—Intersections to snap arc lines to

⇨ Click the Modify tool, select the arc and on the Properties palette, check the Center Mark Visible checkbox.

⇨ Align and lock the center point and then each arc endpoint to the reference planes in both directions (6 alignments total) (see the left side of Figure 37).
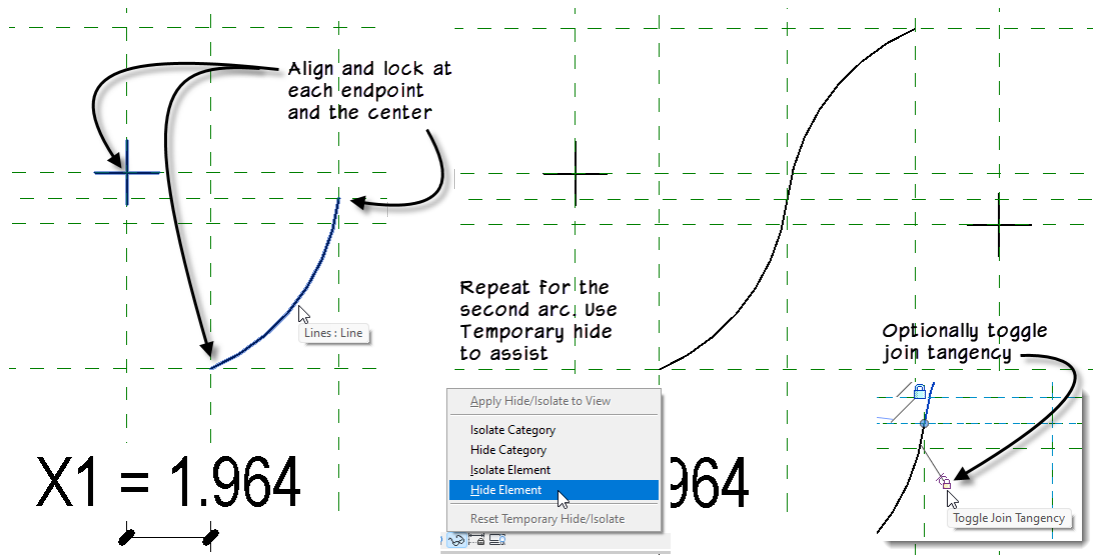
Figure 37—Draw the second arc and constrain it

8. Select the arc just drawn, and on the View Control Bar (at the bottom of the screen) use the Temporary Hide/Isolate icon and choose: **Hide Element**.

The previous step is not strictly required, but in compound curves, it can often make the step of applying constraints much easier. By hiding the items that you are not working on, it is much easier to align and lock the desired items without errors.

9. Repeat the process by drawing a second arc with center at point the intersection of the Arc Center Right Hor and Arc Center Right Ver reference planes (point 4 in Figure 36 above) and the endpoints at locations at the intersection of Y Max and X Max and Center (Left/Right) and Center (Front/Back) (points 5 and 3 in Figure 36 above).

⇨ Click the Modify tool, turn on the center mark and align and lock all points.

⇨ Reset the Temporary Hide/Isolate (see the right side of Figure 37).

When you select either arc, you will notice a small "Toggle Join Tangency" icon at the endpoints where they meet. This toggle ensures that the two lines remain tangent while flexing. In this case, toggling this is optional. Since we have set up the formulas with the trigonometry, our construct will always result in tangency between the two arcs regardless of the pressenc of the tangency lock. Toggling it in this case is therefore optional. In cases where you do not use trigononmetry to ensure that the curves maintain tangency, this toggle can be quite helpful.

10. Open "Family Types" and flex the curve.

The curves should flex properly and remain constrained to the reference planes.

11. Draw vertical and horizontal lines locked to the reference planes for the leftmost edge and the top and bottom.

CATCH UP! You can open a file completed to this point named: *D02_Cyma Profile.rfa*.

12. Flex the completed version when done.

You can load this profile into any project or family now and use it to shape anything from sweeps, to wall profiles to railings. Let's test it out in our sandbox file from above.

13. Open: *_05_Sweep Flex.rfa*.

A version of the profile family is already loaded, but if you prefer, you can load your version instead.

14. Select the sweep and edit it. On the ribbon, click the Select Profile button and then from the drop-down list, choose one of the types you previously defined (see Figure 38).
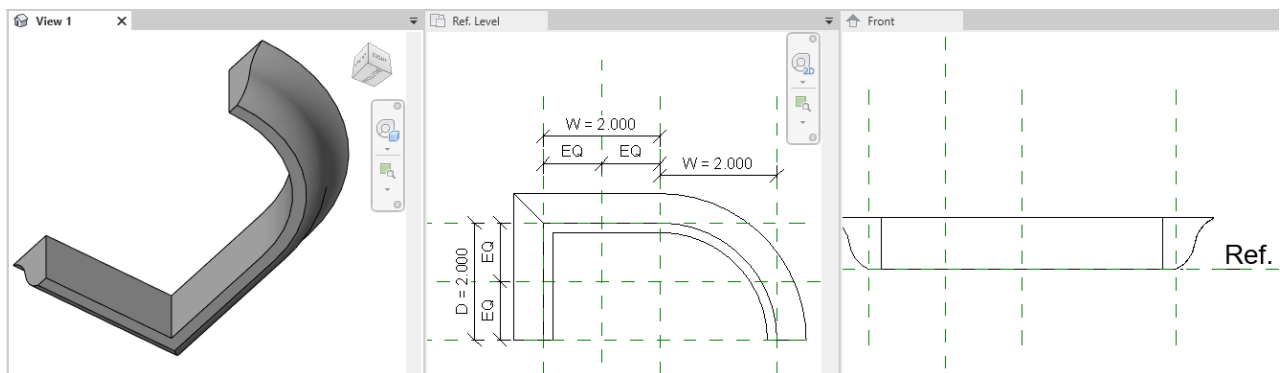

Figure 38—Test the profile family in the sandbox file

15. Finish the sweep and flex the family. Close all files.

Cyma Reversa

The Cyma Reversa is essentially the same shape just with the arcs reversed. So all we need to do to create one is save the Cyma family with a new name and redraw the arcs facing the opposite direction. However, the formulas do need adjustment due to the changed locations of the arc center points. The overall strategy and form is largely similar. You can try your hand at one if you like, or you can simply open the example provided here. There are two versions of the profile that you can edit from the Families branch of the Project Browser.

> **CATCH UP!** You can open the sandbox file which contains a completed version of the Cyma Reversa profile. Open: _05_Sweep Flex.rfa. Then on the Families branch of Project Browser, right-click the D02_Cyma Reversa Profile (Trig) family to open it.
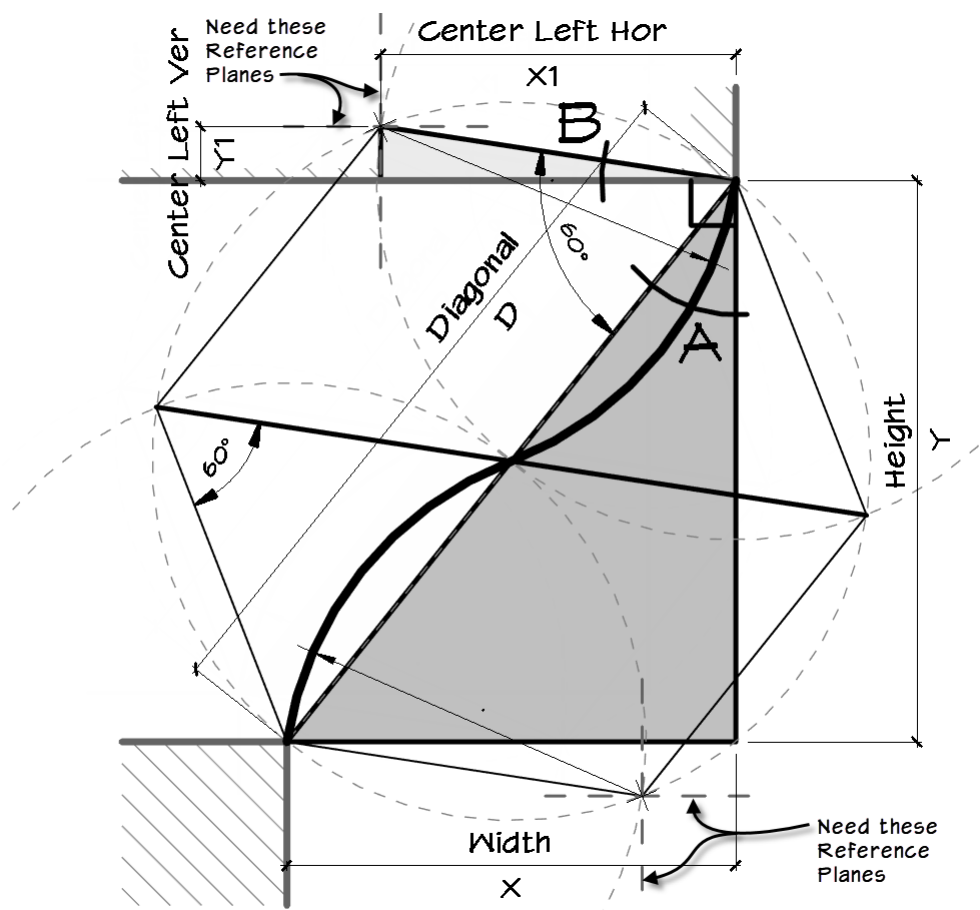
Figure 39—Cyma Reversa Construction

The critical angle is angle B. There are two known angles in its vicinity, the right angle between the reference planes X Max and Y Max and the angle between the diagonal and the top edge of the implied hexagon. The diagonal, as we saw above, is derived from the Height (Y) and Width (X) parameter inputs and the Pythagorean Theorem. As the diagrams for the cyma (above) and this figure illustrate, the diagonal's endpoints form two vertices and becomes the bisector of an implied hexagon. The hexagon's other four vertices determine the locations of the center points

of the arcs of Cyma and Cyma Reversa curves. Since a regular hexagon is easily divided into six equilateral triangles, we also know that the angle between the diagonal (bisector) and the hexagon's top edge is 60°. These known angles make it easy to calculate angle **B**, which in turn gives us the values **X1** and **Y1** and the locations of the required reference planes.

Torus

The last few examples have been some very common molding profiles. Here we will build one more. A Torus molding has a half-round shape. However, we will tie this one back to concepts discussed at the beginning of this paper by making the profile able to flex between a half-circle and half-ellipse.

1. Open the profile seed family called: _04 Seed (Profile).rfa_.

2. Save the file as: **Torus Profile**.

Recall from above that this seed begins with a few parameters and some stand-in lines in the shape of a triangle.

3. Delete the triangle.

⇨ Delete both of the horizontal dimensions (the one labeled with parameter: **x** and the EQ one).

4. Add two new horizontal dimensions: One from X Min to Center (Left/Right) and another from Center (Left/Right) to X Max.

⇨ Label the one on the left with a new parameter called: **Projection**.

⇨ Label the one on the right with the existing **x** parameter (it will flex) (see the left side of Figure 40).

5. Flex parameter **x** to a value of: **6** (see the right side of Figure 40).
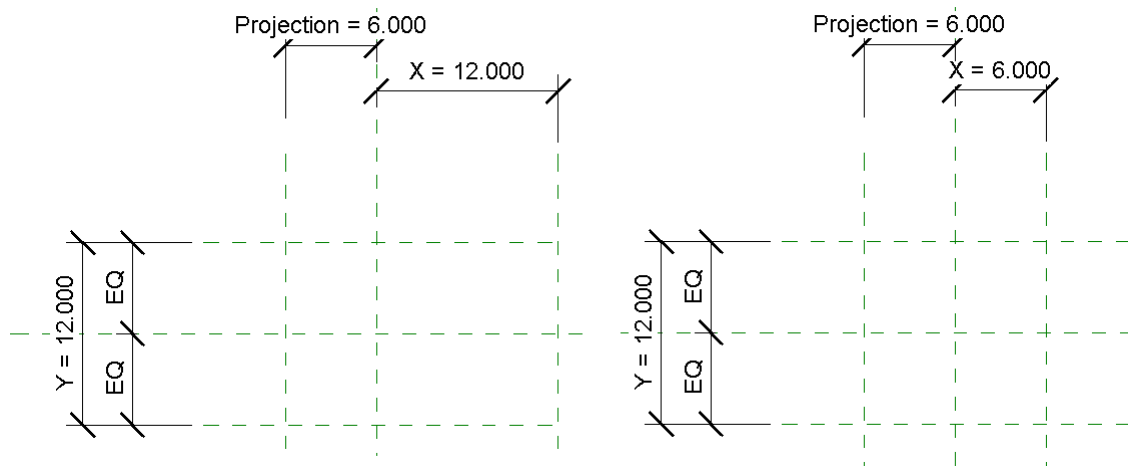
Figure 40—Reconfigue the horizontal dimensions and labels

Recall from the first exercise that an ellipse with equal axes is the same as a circle. Therefore, in this profile, we will use an elliptical arc. If we keep the value of **x** at half of **y**, we will have a circular profile. If you make them unequal, it will become elliptical.

6. Draw an Elliptical arc with the major axis running vertically along the Center (Left/Right) reference plane.

⇨ Make the minor axis to the right by snapping to the intersection of Center (Front/Back) and X Max (see the left side of Figure 41).
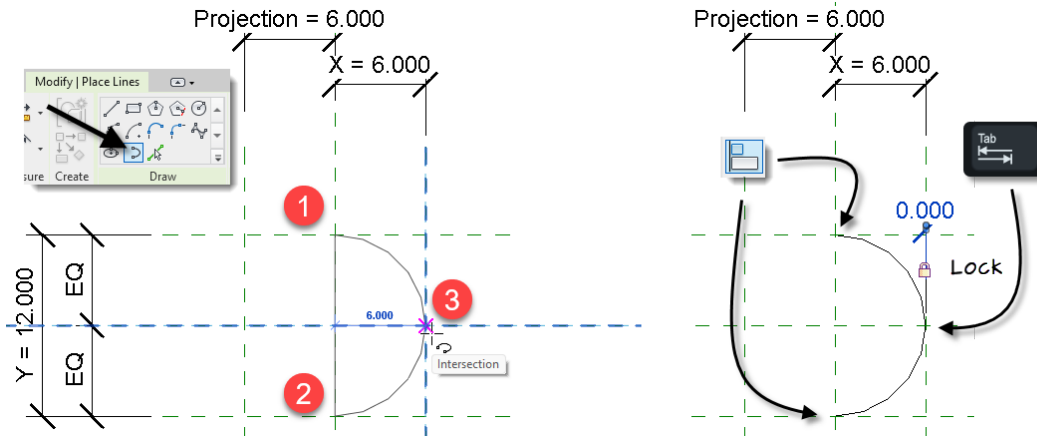

Figure 41—Create an elliptical arc and constrain it to the reference planes

7. Align and lock the endpoints in both directions to their respective reference planes.

⇨ Add a dimension between the X Max reference plane and the endpoint of the elliptical arc. Use the TAB key to assist (see the right side of Figure 41).

8. Draw three straight lines along the Y Max, Y Min and X Min reference planes to finish the shape (see).
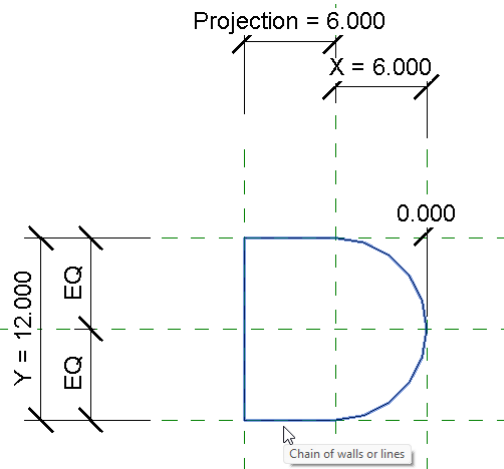
Figure 42—Complete the profile shape

Be sure to test the family in the Family Types dialog to ensure that it is functioning as expected. To make a circular torus, be sure to keep the value of **x** half the value of **y**. If you want an elliptical torus, vary this proportion. You can easily manage this manually.

However, if you want to control this with a formula, you can do so. First you must add a Yes/No parameter the to family. You can call it something like: "Circular". You will then need to replace the label of one of your parameters with a new parameter. For example, add a new parameter and use it to replace the label on the Y dimension. Call it: "Y Calc." Finally, the formula for **y**

`Calc` would be as follows: **if(Circular,X*2,Y)**. This means that if the `Circular` parameter checkbox is checked, then it will use a value of 2 times `X`. Otherwise it will use the value of `Y`. In other words, if `Circular` is not checked, it is not circular, so you can put in a alternate value for `Y`. When `Circular` is checked, the value of `Y` is ignored.

> **CATCH UP!** You can open a file completed to this point named: *E01_Torus Profile.rfa*.

Tying it all together

In the final exercise, we will create an opening family that uses many of the pieces we have already assembled. In addition to showcasing one more example of a compound curve, it will also showcase the modular approach that we have been building up towards. Since we now have many profile families, we will be able to use them to create forms in other families and projects. In this example, we'll create an arched opening complete with impost and archivolt moldings that utilize the profiles created so far.

Three-Center Arch

Before we begin assembling moldings, we have one more profile to construct. This profile will define the shape of the opening. As noted above, one of the advantages of using profile families is that you can reuse them in many other families and projects and their various elements and forms. This includes both solid and void forms. Above in the "Create a Simple Parametric Arch" topic, we looked at several common arch forms and created a simple family that could flex between a small collection of them. In this topic, we will explore another type of arch. This is the three-center arch. The three-center arch resembles an elliptical arch, but is more conducive to application of built-up moldings. This is because the geometry of an ellipse does not lend itself to concentric rings. You can get close, but there is no way to make an ellipse that is equally distant all the way around to another one. The three-center arch looks very close to an ellipse but is comprised of circular arcs. Therefore, creating concentric rings is much easier. The other advantage of using the three-center arch is that it will afford us one more example of creating a compound curve. This one will contain three arcs joined tangentally in series (see Figure 43).
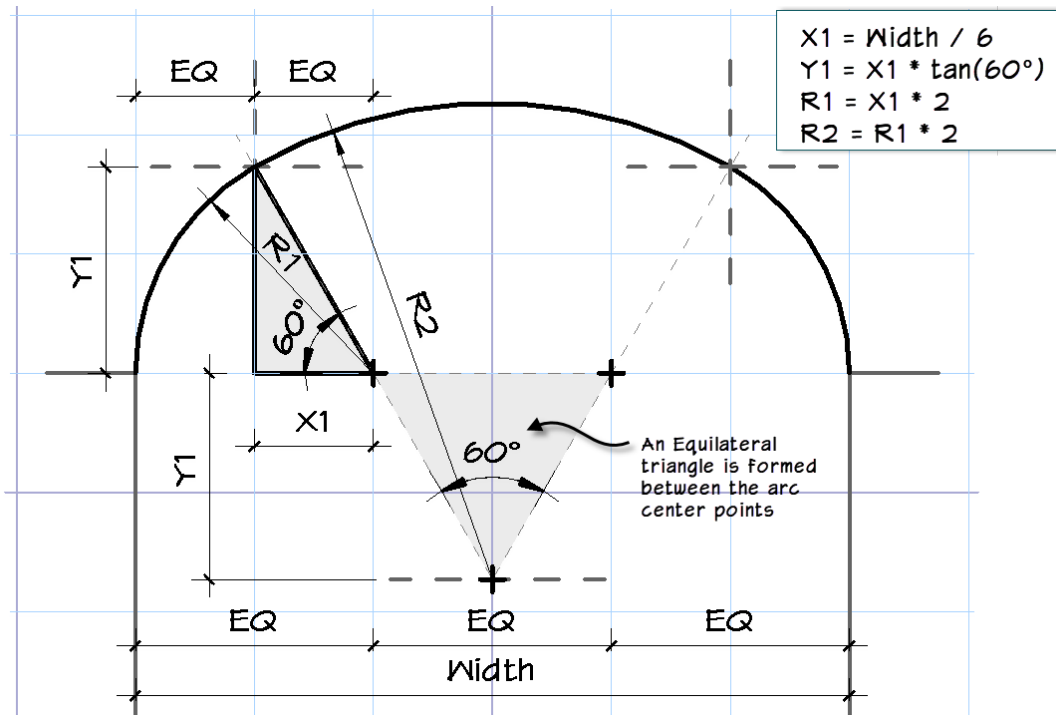
Figure 43—Construction of a Three-Center Arch

We'll begin in a profile family. Create the three-center form as a flexible family and then load that into another family where we will use it to create the opening and form the path for the moldings. This will also give us a nice example of using one of our profiles to create a void form. For this example, we are keeping the shape simple and fixing the angle at 60°. This keeps the rest of the geometry pretty simple. The two arcs on the left and right are 60° arc with centers along the spring line. The middle arc is similar to our segmental arch above. We could apply the technique used there to constain directly to the arc, but in this case we know where the center point of the arcs are located and so here we will instead contrain the arcs using their centers and endpoints.

The profile family that we will use has already been started to save some time. It contains the required reference planes and parameters. We just need to add the formulas and linework.

1. Open the family called: *F01_Three Center Arch_Start.rfa*.

2. Save the file as: **Three Center Arch Profile**.

Let's explore what is already provided in the file (see Figure 44).
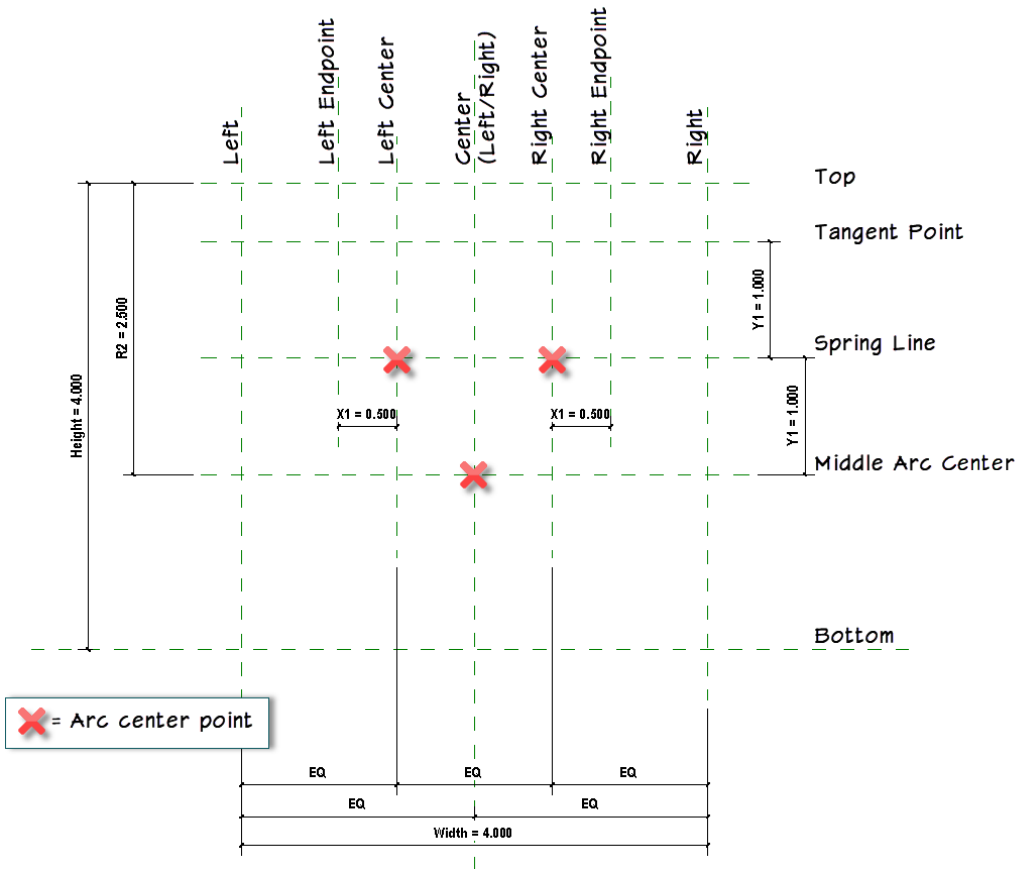
Figure 44—Starter file for the three-center arch profile

The figure shows the names of the reference planes included in the file. The red Xs indicate intersections that will locate the arc center points. Dimensions and parameters have also been included and applied to the reference planes. The equality dimensions keep things symmetrical running horizontally. Changing the Height will move the entire rig up or down. The rest of the parameters will impact the size and shape of the arch itself.

3. Open the "Family Types" dialog. Using the following table, input the formulas shown for each parameter.

Table 3

Parameter	Formula
X1	Width / 6
Y1	X1 * tan(60)
R1	X1 * 2
R2	R1 * 2

4. Click Apply to test the values.

The reference planes will adjust slightly based on the new values (see Figure 45).
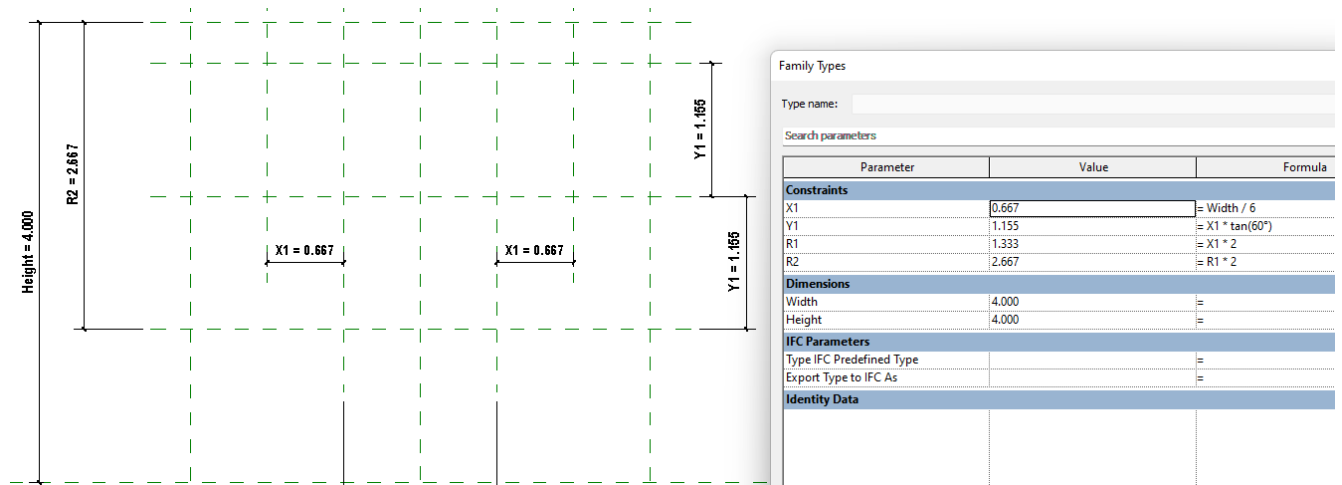
Figure 45—Add formulas to the parameters and apply

5. Close the Family Types dialog.

6. Using the line tool, draw a Center-Ends arc.

⇨ Place the center point at the intersection of the Spring Line and Left Center reference planes (point 1 in Figure 46).

⇨ Place the first endpoint at the intersection of the Spring Line and the Left reference planes (point 2).

⇨ Place the final endpoint at the intersection of the Left Endpoint and Tangent Point reference planes (point 3).

7. Repeat on the right side (points 4,5 and 6).
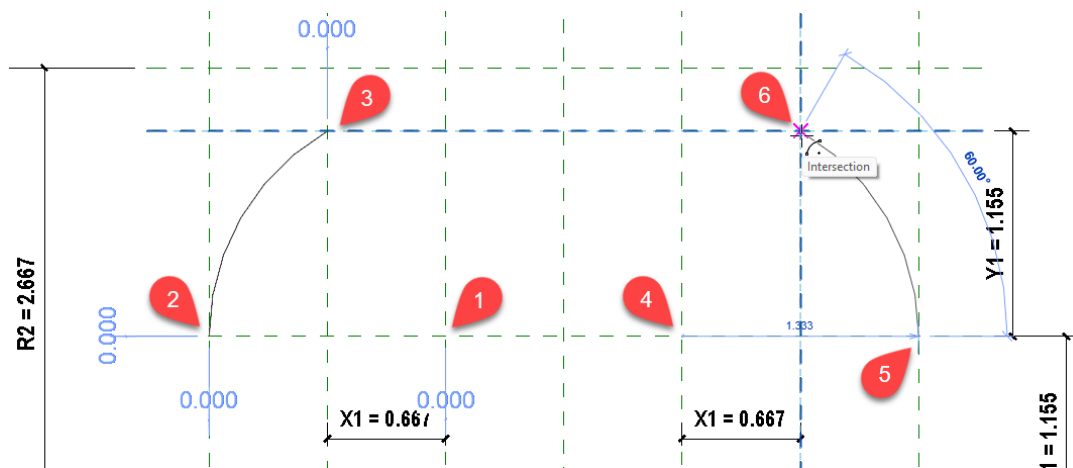
Figure 46—Draw the left and right arcs

8. Click the Modify to finish drawing. Select both arcs and on the Properties palette, check the Center Mark Visible checkbox.

9. Align and lock both centers and all four endpoints to the reference planes in both horizontal and vertical directions.

This should remove all automatic sketch dimensions. Feel free to flex the family before moving on the ensure that everything is working properly.

10. Select both arcs and on the View Control Bar, from the Temporary Hide/Isolate pop-up, choose: **Hide Element**.

11. Draw the last arc using the Center-Ends arc tool.

⇨ Place the center at the intersection of the Center (Left/Right) and Middle Arc Center reference planes.

⇨ Place the first enpoint at the intersection of the Left Endpoint and Tangent Point reference planes (point 3 in the previous figure).

⇨ Place the final endpoint at the intersection of the Left Endpoint and Tangent Point reference planes (point 3 in the previous figure).

12. Enable the Center Mark Visible and then align and lock the center and endpoints.

13. Reset the Temporary Hide/Isolate and then flex the family.

As before, tangency locks are optional in this family.

14. Draw the three straight lines at Left, Right and Bottom reference planes to complete the profile and then save the family (see).
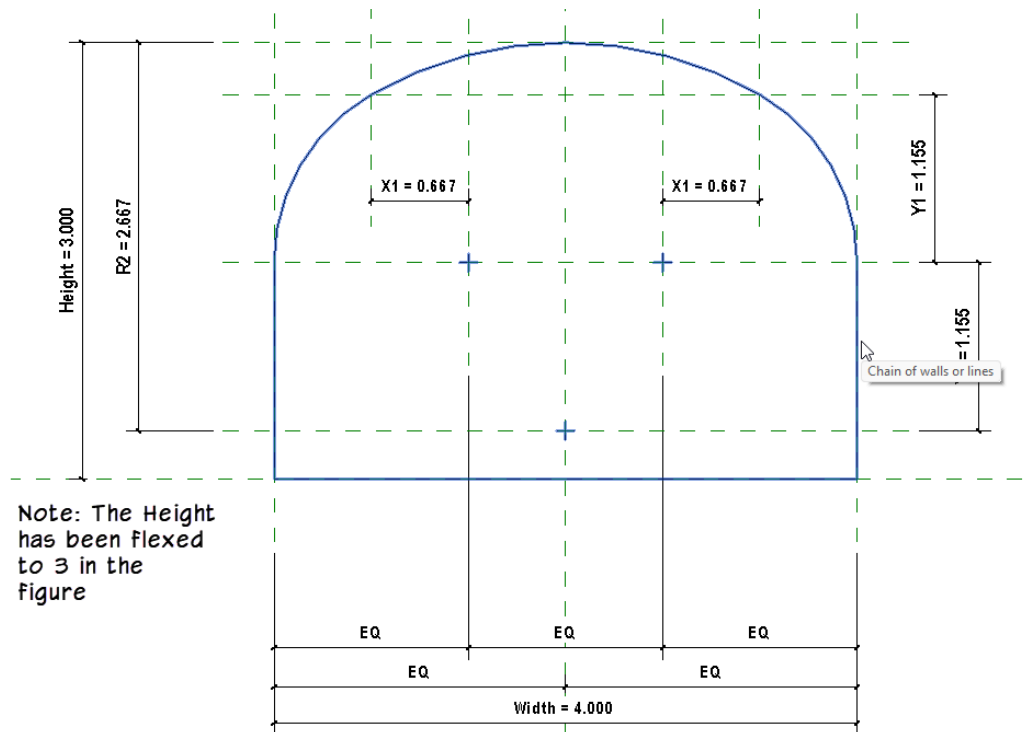
Figure 47—Complete the profle geometry and flex

CATCH UP! You can open a file completed to this point named: *F02_Three Center Arch.rfa*

15. Close all files.

Create the Opening Family

We have all of the pieces we need to create the final family. The three-center arch family created in the previous topic will be used to create a void element in a wall-based family. This will be the arched opening. Surrounding this opening, we will add a serious of moldings using the profile families created earlier. These will include both an archivolt, which is equivaelent to an architrave only following the curve of an arched opening. The Architrave is a series of moldings that comprise the lowest of three sections (architrave, frieze and cornice) of a typical entablature. The entablature spans across columns in a traditional colonnade. And at the springline of the arch we will add impost moldings. These serve as a capital to the piers that support the spring of the arch[3].

1. Create a new family using the family template: *Generic Model wall based.rft*. (A copy of this template is included in the dataset). Save the file as: **Arched Opening with Moldings**.

2. From the Insert tab, click the Load Family button.

⇨ Select the file named: *F02_Three Center Arch* and then click Open (see Figure 48).
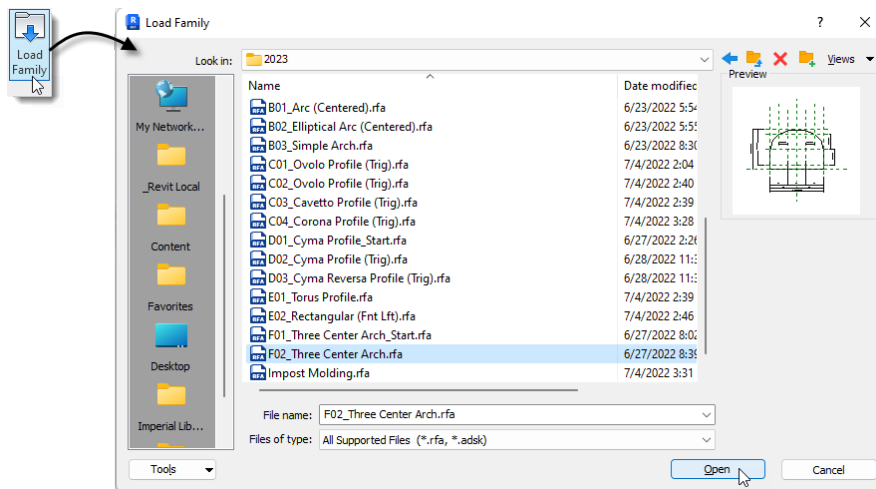
Figure 48—Load the previously created profile families

Create an Opening Using a Void

Let's start with the opening. We are using a Generic Model template here. So there is not an opening already in the file like there would be in a door or window template. But you can easily add one. There are two ways to do this. We can use the Opening tool on the ribbon, or we can create a void. Opening elements are appropriate when you want a cut that goes through the wall entirely and is perpendicular to the surface of the wall. This is suitable for many common openings and would be appropriate for our situation here. However, there is one additional criterion which we need to satisfy and that an Opening element will not. That is we want to use our previously create three-center arch profile family to shape our opening. The Opening tool does not support profile families. Therefore, we will instead create a void sweep.

---

[3] These definitions are paraphrased from *The Classical Orders of Architecture*, Robert Chitham.

Click in the *Ref. Level* floor plan view tile to make it active. If it is not open, then double-click to open it from the Project Browser.

1. On the Create tab, click the Void Forms drop-down and choose: **Void Sweep** (see item 1 in Figure 49).
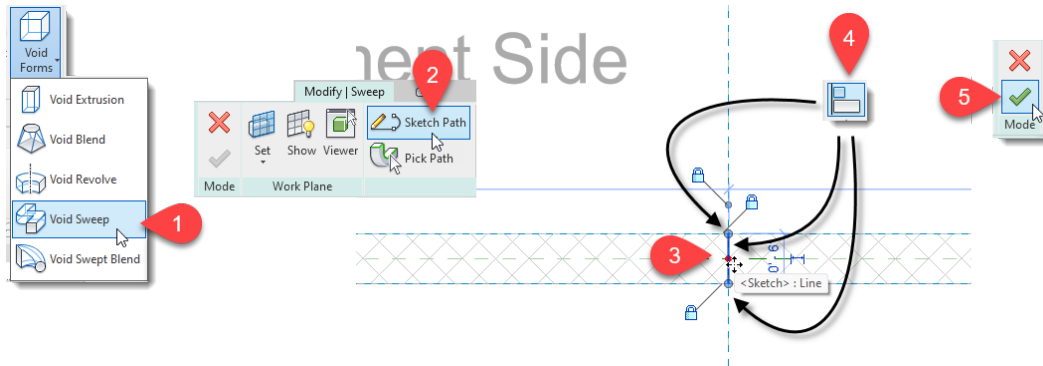

Figure 49—Sketch the path of the void sweep across the center of the wall

⇨ On the Modify|Sweep tab, click the Sketch Path button (item 2).

⇨ Draw a straight line along the Center (Left/Right) reference plane from face to face of the wall (item 3).

⇨ Align and lock the line to the Center (Left/Right) reference plane and align and lock each endpoint to the faces of the wall (item 4).

⇨ Click the Finish Edit Mode button (item 5) to complete the sketch.

2. On the Modify | Sweep tab, click the Select Profile button to active the Profile drop-down list (see item 1 in Figure 50).

⇨ From the Profile drop-down list, choose: **F02_Three Center Arch** (items 2 and 3).

⇨ Click the Finish Edit Mode button to complete the sweep (item 4).
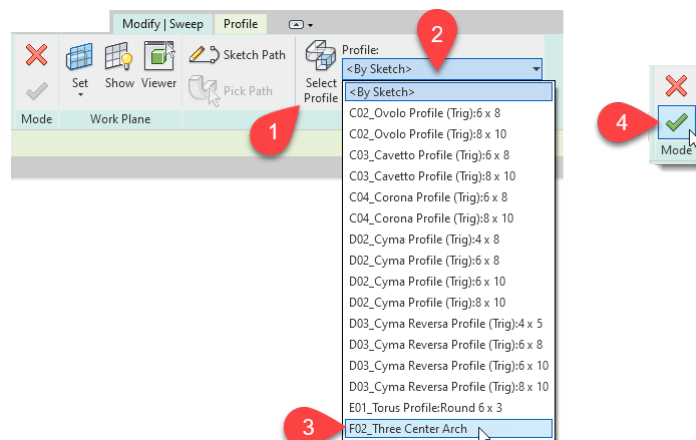
Figure 50—Choose a profile for the sweep

The void sweep will appear in all views superimposed over the wall. It will not engage with the wall automatically. So let's use Cut Geometry to achieve that.

3. On the Modify tab, click the Cut Geometry tool.

4. First click the wall, then click the void (see Figure 51).
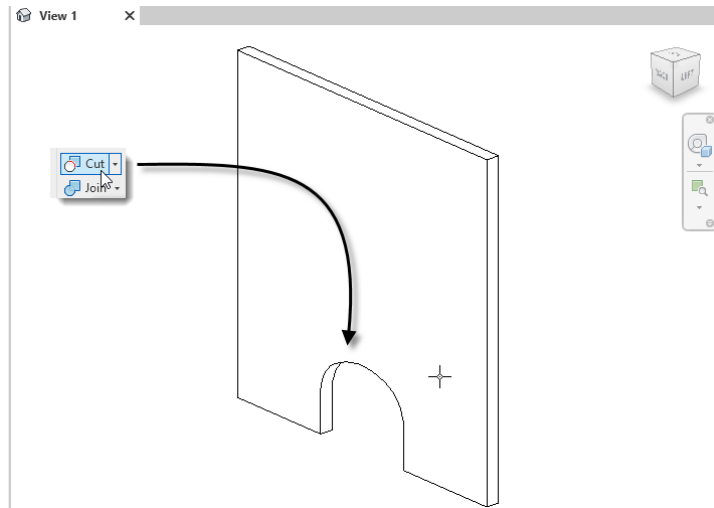
Figure 51—Use Cut Geometry to engage the void with the host wall

Now let's ensure that the size and shape of the arch flexes to the dimensions we require.

5. On the Project Browser, expand the *Profiles* branch, locate the: *F02_Three Center Arch* item and expand it as well.

6. Right-click the type named: *F02_Three Center Arch* and choose: **Rename**.

⇨ Call it: **Flex** and then press ENTER.

⇨ Right-click the newly renamed type: *Flex* and choose: **Type Properties** (see item 1 in Figure 52).

Figure 52—Create and associate family parameters to drive the size of the nested profile
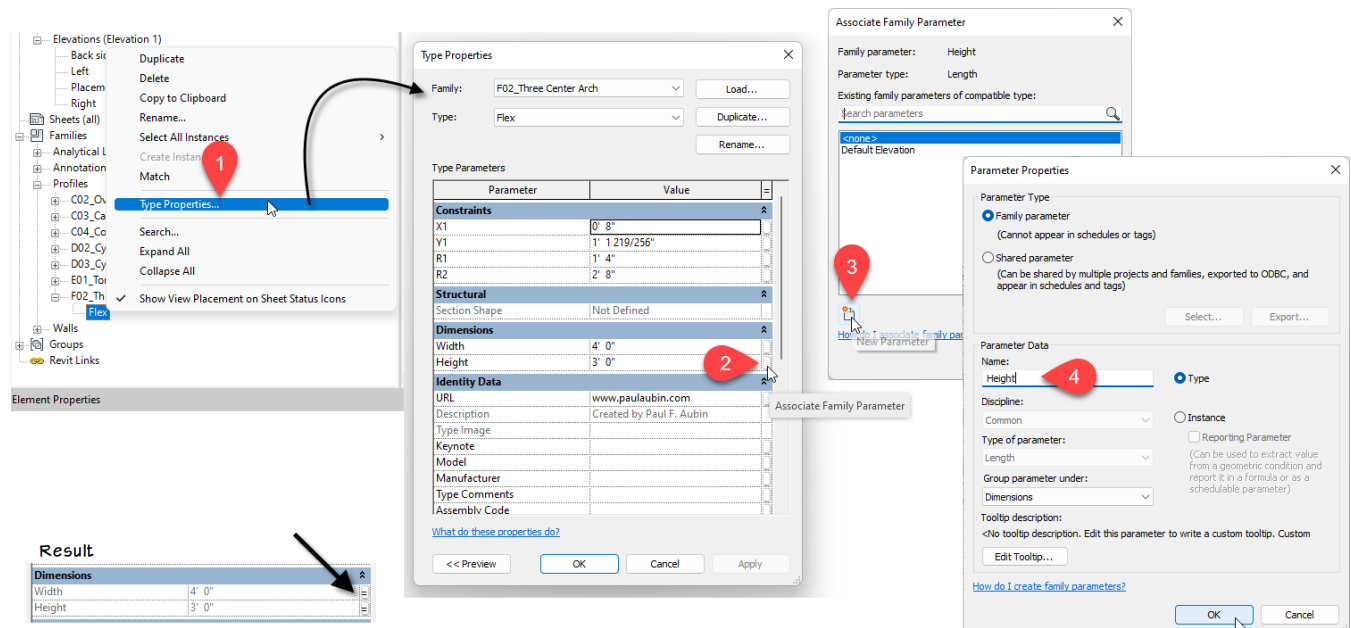
⇨ In the Type Properties dialog, next to the Height parameter, click the Associate Family Parameter button (item 2).

⇨ In the Associate Family Parameter dialog, click the New Parameter icon at the bottom (item 3).

⇨ In the Parameter Properties dialog, name the new parameter: **Height** (item 4).

7. Click OK twice to return to the Type Properties dialog and repeat the process to create and associate a new **Width** parameter.

When complete, both parameters in the Type Properties dialog will have small equals signs on the Associate Family Parameters buttons as shown in the lower-left corner of the figure.

8. Open the Family Types dialog and flex both the `Width` and `Height` parameters (see Figure 53).
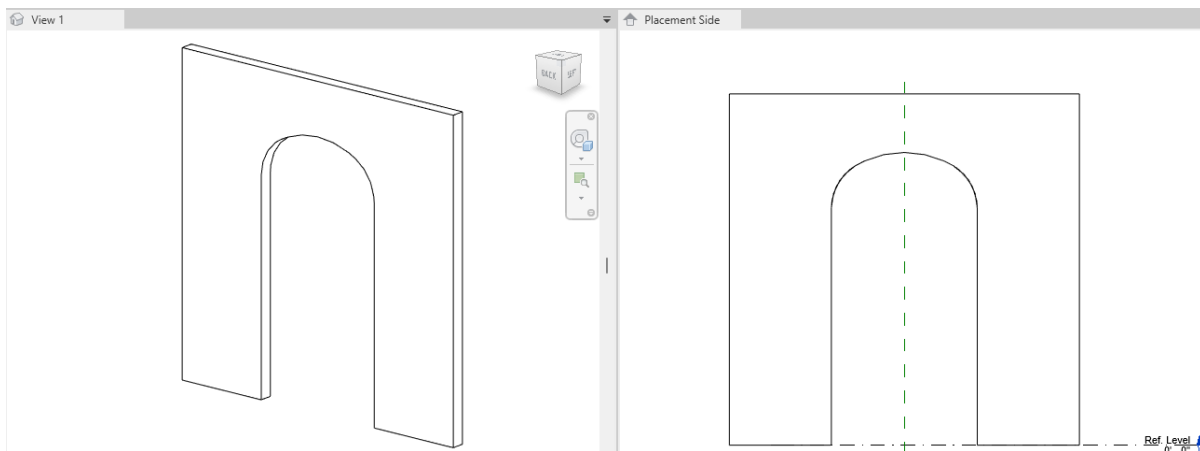
Figure 53—Flex the new parameters to ensure that the opening adjusts properly

Add Impost Moldings

Next we'll add moldings at the springline. These help define the line that arch springs from and create a kind of capital for the pilasters on either side of the opening. We will begin here in this file and add a few reference planes, then switch to our sandbox file, make a copy of it and set up the moldings in that file. Finally, we'll load them in and place them here.

Locate the Arch Springline and Molding Extent

Click in the *Placement Side* elevation view tile to make it active. If it is not open, then double-click to open it from the Project Browser.

1. On the Create tab, Click the Reference Plane button and add a horizontal reference plane (parallel to the floor) at the approximate location of the arch springline.

⇨ Name the reference plane: **Springline**.

In order to get this reference plane to coincide with the location of the springline, we will need to replicate the formula applied to a few of the parameters from the three-center arch profile family.

2. Add a dimension from the new reference plane to the Ref. Level.

⇨ Label this dimension with a new parameter named: **Springline Offset**.

Page 44

In the profile family, we have the overall `Height` parameter. From this location, we measured down to the location of the large radius. From there the springline is measured back up by the value of the parameter named: `Y1` (refer back to Table 3, Figure 44 and Figure 47). We could recreate each of these parameters and their formulas, but it is also possible to simply nest those formulas directly into our new `Springline Offset` parameter that we just created. The required formula is the equivalent of: **Height – R2 + Y1**, or: **Height – (Width/1.5) + ((Width/6)*tan(60))**.

3. Open the Family Types dialog.

⇨ For the formula of the `Springline Offset` parameter, input the following: **Height – (Width/1.5) + ((Width/6)*tan(60))** (see Figure 54).
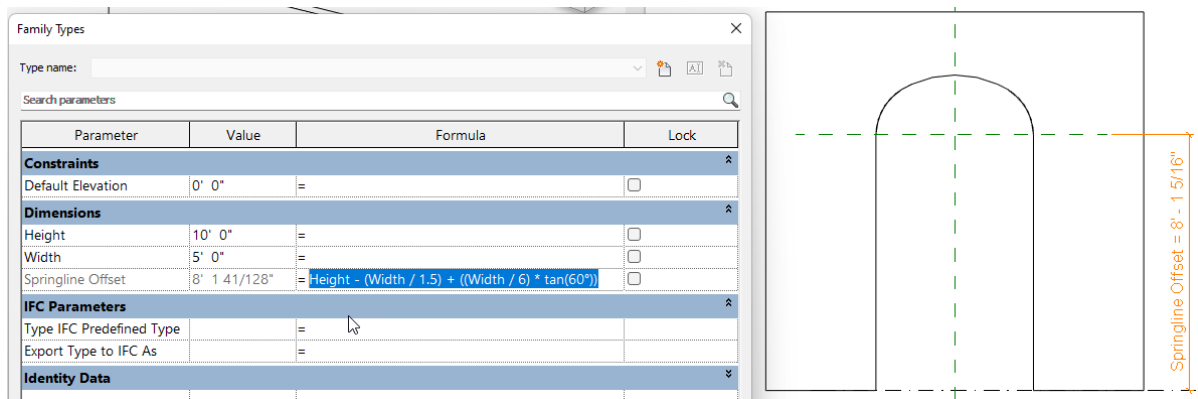
Figure 54—Input a formula derived from the three-center arch to locate the springline

4. Flex the family to ensure that the reference plane stays in the required location. Dismiss the dialog when finished.

5. In the Placement Side elevation view, add a new vertical reference plane parallel to and to the right of the Center (Left/Right) one. Name it: **Right Impost Extent**.

⇨ Add a dimension from this reference plane to the Center (Left/Right) reference plane.

⇨ Label this with a new parameter called: **Right Impost Extent**.

6. Repeat on the left side creating a reference plane and parameter called: **Left Impost Extent** (see Figure 55).
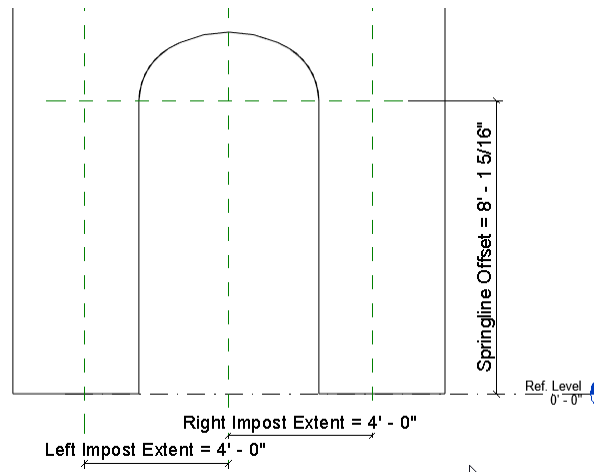
Figure 55—Add reference planes and parameters

CATCH UP! You can open a file completed to this point named: *F03_Three Center Arch.rfa*

Create Moldings in a Separate File

Next we'll reopen the sandbox file that we used to test our molding profiles. In that file, we will create a complex molding by building up several sweeps, each using a different one of our previously created profiles.

1. Open: *_05_Sweep Flex.rfa*.

⇨   From the File menu, choose: **Save As > Family** and then name the new file: **Impost Molding**.

We only need the two straight sides of the sweep that is already in this file, not the curve. Therefore, we will edit the sweep path and remove the curved part.

2. Select the rightmost reference plane (named: Arc) and delete it.

3. Select the Sweep and on the Modify | Sweep tab, click the Edit Sweep button.

⇨   On the Modify | Sweep tab, click the Pick Path button.

⇨   Select the curved portion of the path only and delete it.

    The two straight lines of the path will remain.

⇨   Click the Finish Edit Mode Button twice.

⇨   There will remain an arc model line. Delete this too.

If you move your mouse near the remaining segments of the sweep, you will find that there are two reference lines there. These are for the picked path in the sweep. Make sure that you do not delete these (see Figure 56).
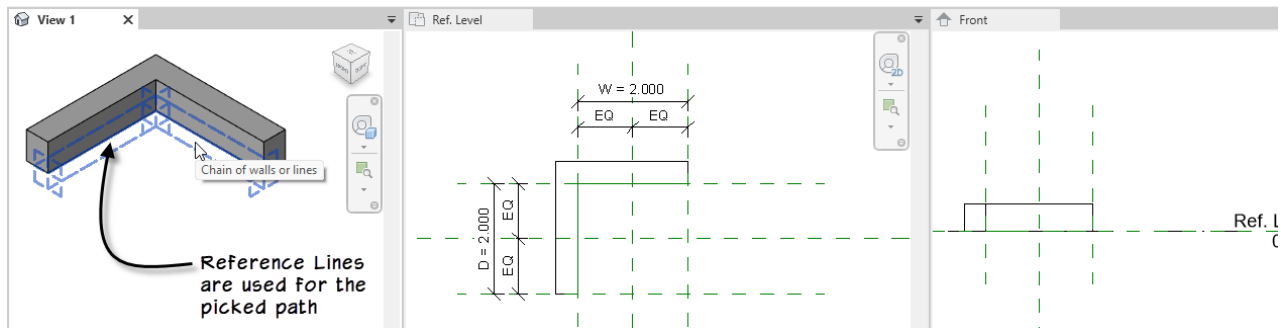
Figure 56—Delete the curved portion of the picked path

This file contains many profiles already representing the finished versions of each of the ones created above. In addition to those, there is also a simple rectangular one in the file as well. This is the one used by the sweep initially. If you changed it above, you can set it back to this one now. Also, there are types saved in this file for the size and proportions that we need for each molding.

4. Select the sweep.

⇨   On the Properties palette, from the Profile drop-down list, choose: **E02_Rectangular (Fnt Lft):Impost_01** (see Figure 57).
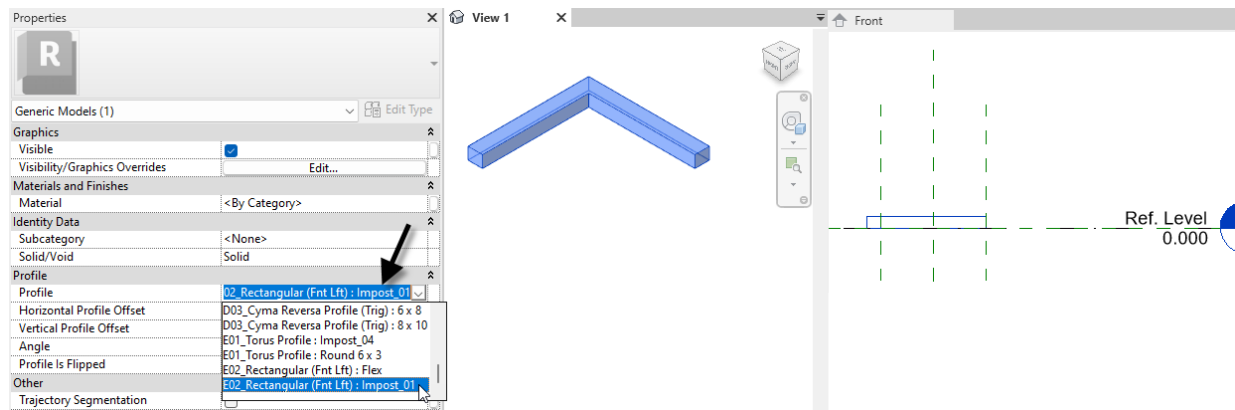
Figure 57—Change the profile of the sweep

Notice that the profile got a little smaller. Also notice that the height (as seen in the Front view) goes above the level. We need it to move down. There would be two ways to achieve this. The first would be to edit the profile family, change which reference plane in that file defines the origin and then reload it in and overwrite the existing.

Alternatively, we can simply offset the profile here for this sweep instance. This is what we will do here. We want the top edge of the profile to align perfectly with the level. Either edit the profile type called: Impost_01, make note of its Y dimension and then apply that as the Vertical Profile Offset in the negative direction on the Properties palette. Or you can edit the sweep and use the move command to achieve the same result.

5. Select the sweep.

⇨ In the *Front* elevatin view, click the Edit Sweep button on the ribbon.

⇨ Select the profile onscreen and then using the Move tool and object snaps, move it down so that the top of the profile aligns with the level.

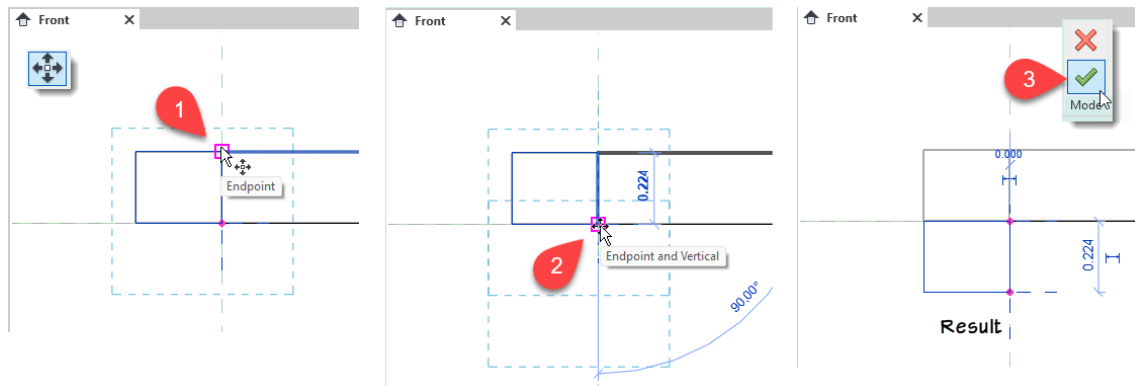⇨ Finish Edit Mode (see Figure 58).

Figure 58—Move the profile down to align with the level

Work in the *View 1* 3D view.

6. On the Create tab, click the Sweep button and then click the Pick Path button.

⇨ Use the TAB key to make sure that you are picking the reference line, then click (see the left side of Figure 59).

⇨ Again using the TAB key, click the other reference line and then click the Finish Edit Mode button (see the right side of Figure 59).
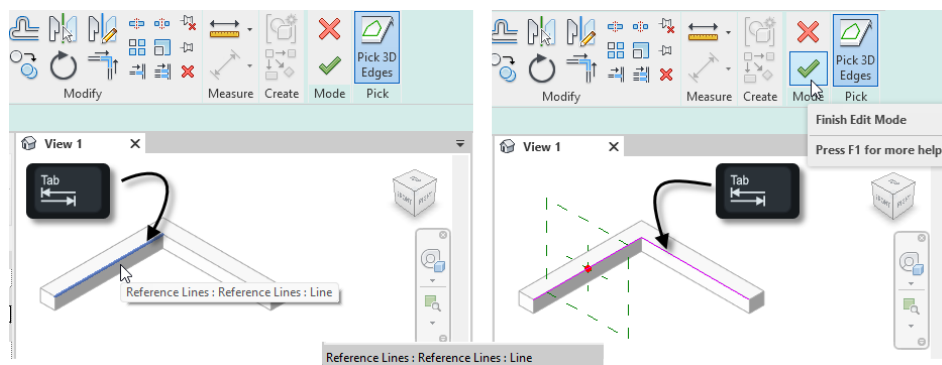
Figure 59—Using Pick Path, pick both reference lines

7. On the Modify | Sweep tab, from the Profile drop-dwon, choose: **C02_Ovolo Profile (Trig):Impost_02**.

Once again, you'll see it appear above the level line in the *Front* elevation.

⇨ In the Front view, using the Move command, move the profile down and snap it to the underside of the rectangular profile above it (see Figure 60).

If you have any trouble snapping to the profile, you can instead type in the Vertical Profile Offset of: **-0.443** on the Properties palette.
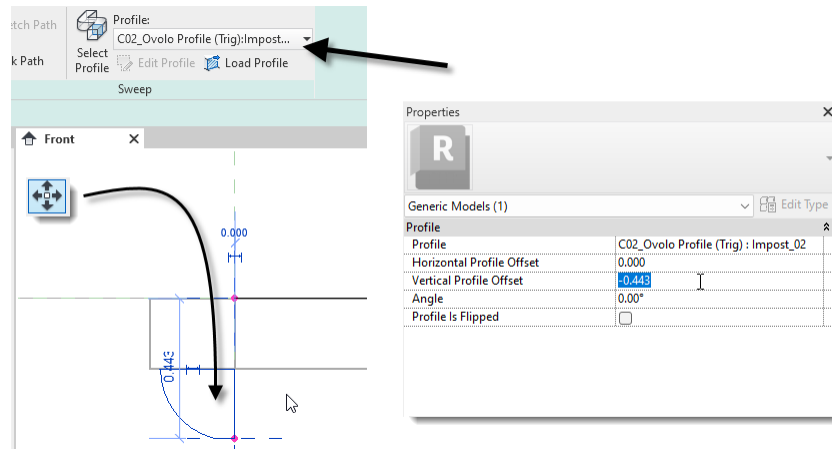
Figure 60—Shift the Ovolo profile down

8. Finish Edit Mode to complete the sweep.

9. Repeat this entire process twice more to create two more sweeps as shown in Table 4 and Figure 61.

Table 4

Profile	Type	Vertical Profile Offset
C04_Corona Profile (Trig)	Impost_03	**-0.683**
E01_Torus Profile	Impost_04	**-0.777**
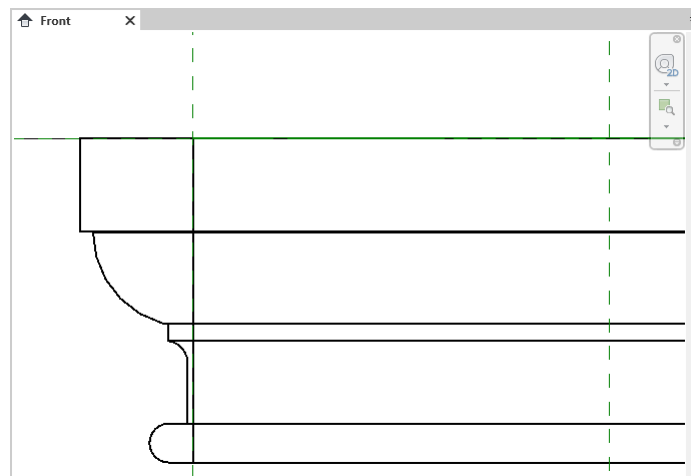
Figure 61—Add the remaining sweeps

10. Using Join Geometry, join all four sweeps together and then save the file.

CATCH UP! You can open a file completed to this point named: *F04_Impost Molding.rfa*

Assemble the Arch

We can now complete the arch by assembling the pieces we have so far.

1. Open the file: *F03_Three Center Arch.rfa*.

⇨ Tile the windows (WT).

2. Working in the *View 1* 3D view, create a new Sweep.

⇨ Click the Pick Path button, and then click the three arcs edges of the wall opening starting with the middle one (see Figure 62).
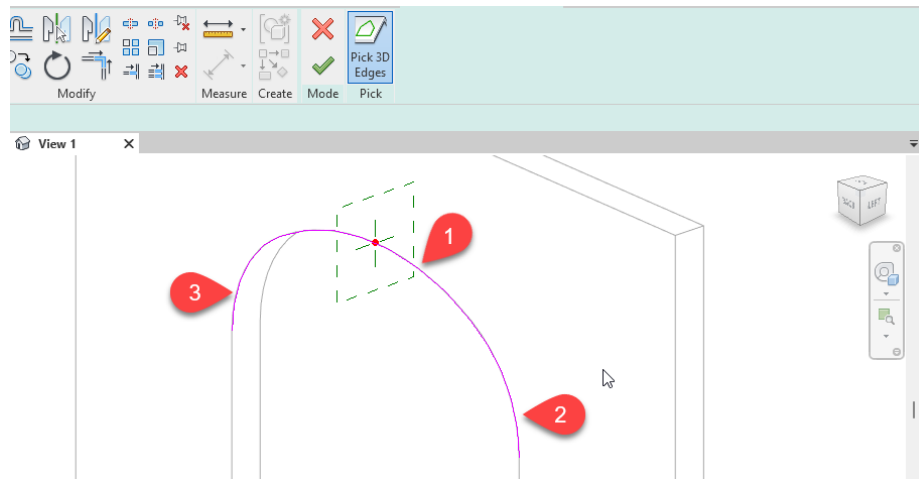
Figure 62—Using pick path, click the edges of the arch

3. Click the Finish Edit Mode button.

⇨ On the ribbon, from the Profile list, choose: **E02_Rectangular (Fnt Lft): Arch_Underside**.

⇨ On the Properties palette, set the Angle to: **180°** and then check the Profile is Flipped checkbox (see Figure 63).
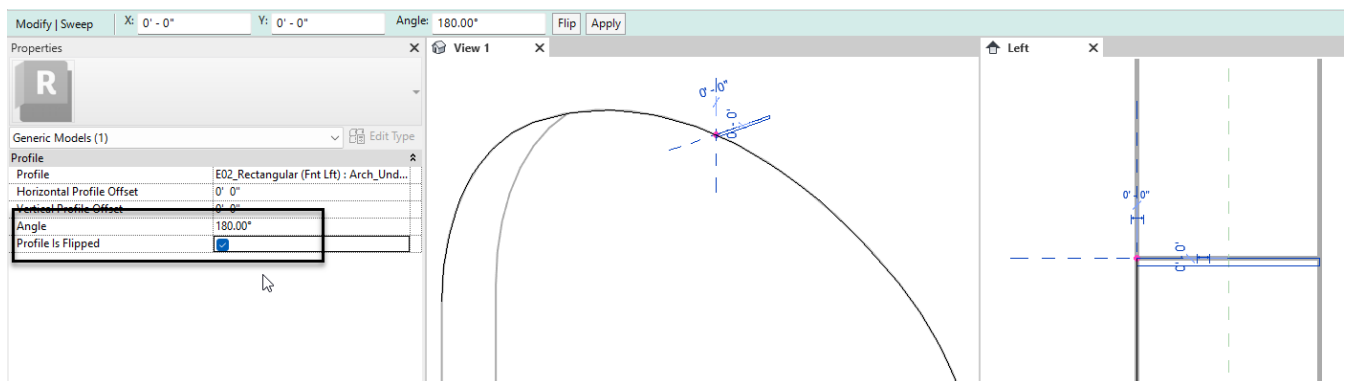
Figure 63—Adjust the profile

4. Click the Finish Edit Mode button.

5. With the new sweep still selected, in the 3D view, click the Temporary Hide/Isolate pop-up and choose: **Hide Element**.

⇨ Create two more sweeps. Pick the same path for each one (using Temporary Hide/Isolate inbetween). Use the settings shown in Table 5 and Figure 64.

⇨ Reset the Temporary Hide/Isolate when finished.

Table 5

Profile	Type	Angle	Profile is Flipped	Vertical Profile Offset
E02_Rectangular (Fnt Lft)	*Arch_01*	**0°**	No	**0**
C04_Corona Profile (Trig)	*Arch_02*	**90°**	No	**1 3/4"**
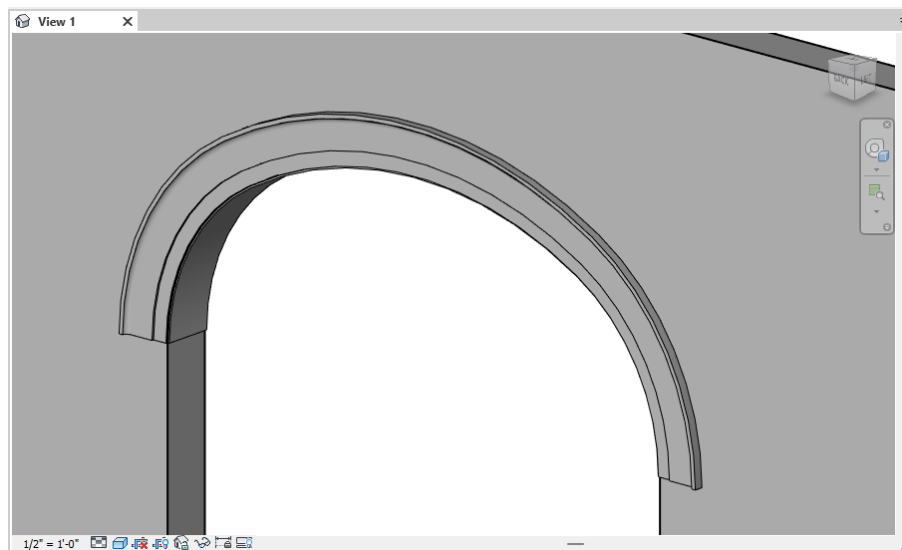
Figure 64—Add the remaining arch sweeps

Work in the *Ref. Level* plan view.

6. On the Create tab, click the Component tool. When prompted to load a family, click Yes.

⇨ Using the CTRL key, select both of the following families: *F04_Impost Molding.rfa* and *F05_Base Molding.rfa* and then click Open.

⇨ Click to place an instance of the *F04_Impost Molding.rfa* family at the intersection of the Opening Right and Wall Edge reference planes.

7. Align and lock each of these positions.

⇨ Continue with align, and align and lock the Inside reference of the family to the opposite face of the wall.

⇨ Align and lock the Right reference in the family to the Left Impost Extent reference plane (see Figure 65).
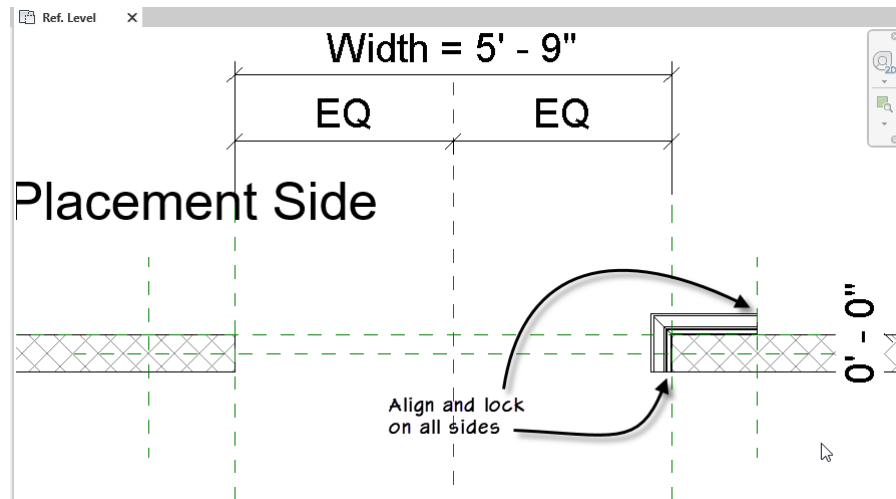
Figure 65—Place one of the molding families and align and lock all around

8. Mirror the family to the opposite side and then repeat the alignments on all sides.

    Looking at the *Placement Side* elevation, you will notice that the moldings have come in at the floor. Let's adjust that now.

9. Select both instances of the *F04_Impost Molding.rfa* family.

10. In the *Placement Side* elevation view, align and lock the two molding families to the Springline reference plane. Use the TAB key as needed (see Figure 66).
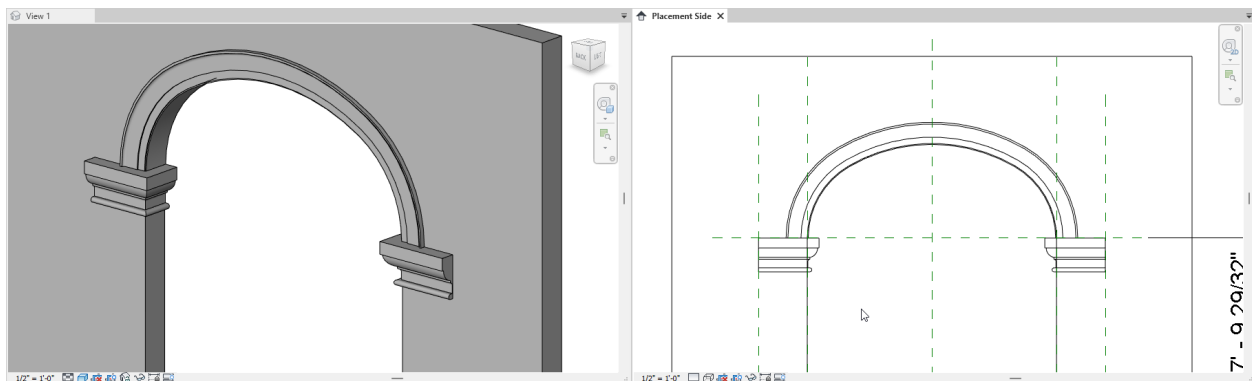
Figure 66—Move the moldings up to the springline

11. Repeat the process above to add the *F05_Base Molding.rfa* family to each side at the floor level.

12. Be sure to align and lock on all sides.

13. Flex the Right Impost Extent and Left Impost Extent parameters to ensure that the moldings are flexing properly (see Figure 67).
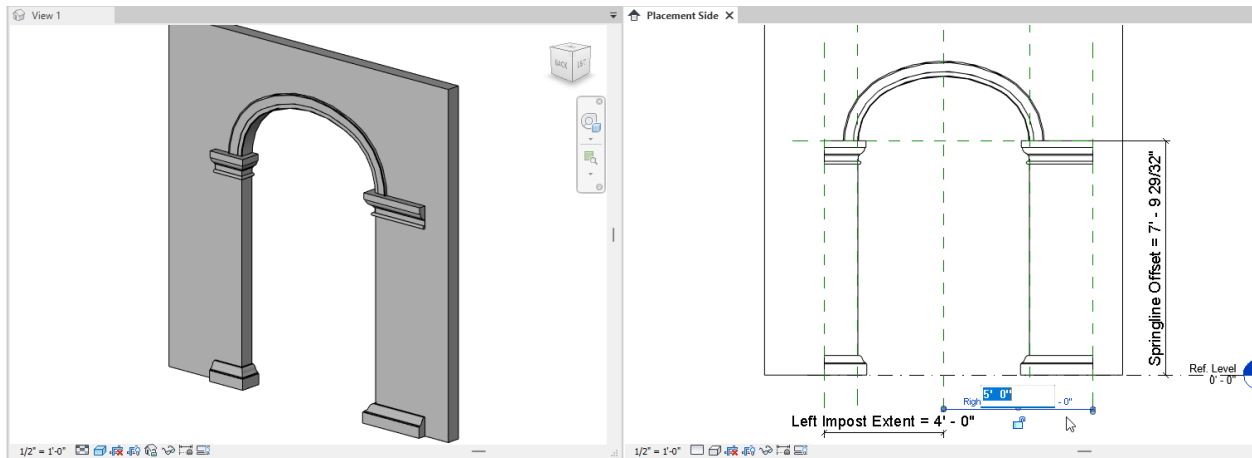
Figure 67—Complete the family and flex the molding parameters

⇨   Save the file.

CATCH UP! You can open a file completed to this point named: *F04_Three Center Arch_Complete.rfa*

Test in a Project

The final test for any family is to test it in a project file. Create a new project using the default template. Add some walls and then load in this family. Place it in one of the walls and then flex it to be sure it is working as expected. We did not make every parameter flexible in this short exercise. For example, all of the sizes of the moldings have been designed to work at this size opening. If you want to create proportional relationships between the opening size and the size of the moldings and their parameters, you can add additional multiplier parameters to do so. Detailed procedures to achieve this can be found in my book:

Renaissance Revit: Creating Classical Architecture with Modern Software. Visit my website: https://paulaubin.com/books/renaissance-revit/ to learn more.

More content on LinkedIn Learning[4]

I also have many video training courses available on LinkedIn Learning. LinkedIn Learning is a subscription based video training platform with courses across multiple disciplines and industries. You can see my complete list of the courses I have authored here:

https://www.linkedin.com/learning/instructors/paul-f-aubin

If you are new to Revit, I recommend my introductory courses available in Revit 2023 and many prior versions as well. The 2023 versions of the courses are Learning Revit 2023 and Revit Essential Training for Architecture 2023. Learning is a basic introductory course. Essential

---

[4] *Links given here are affiliate links and I will be compensated if you make a purchase after clicking on my links.*

Training is also introductory, but covers more content than the Learning course and is more in-depth. Both provide a good foundation for getting started with Revit.

Learning Revit 2023:

https://linkedin-learning.pxf.io/Learning2023

Revit Essential Training for Architecture:

https://linkedin-learning.pxf.io/EssT2023

To explore more on the content covered in this paper, check out my courses on the family editor. In particular, my course **Revit: Parametric Curvature in the Family Editor** covers much of the same material as this paper and would provide an excellent compliment and follow-up.

https://linkedin-learning.pxf.io/Curves

Revit Architecture: Family Editor (Imperial and Metric) is a general introduction to the family editor environment.

https://linkedin-learning.pxf.io/Families

HBIM: Historic Building Re-creation explores more advanced family editor topics and touches on some of the content of this paper as well by using the covered techniques to re-create a historic building façade.

https://linkedin-learning.pxf.io/hBIM

If you want to explore the massing family editor, my course **Revit: Conceptual Modeling Environment** will get you started.

https://linkedin-learning.pxf.io/ConModEnv

Finally, I have a weekly series that posts a new Revit tip each week on a variety of topics. New videos drop on Tuesdays.

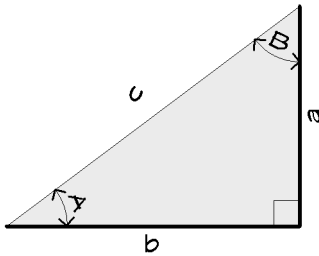https://linkedin-learning.pxf.io/AubinTips

Appendix

The following Trigonometry Cheat Sheet will help you determine which formulas are needed when using trigonolmetry in Revit families. (Thanks to Klaus Munkholm of revitforum.org).

Trigonometry Cheat Sheet for Revit

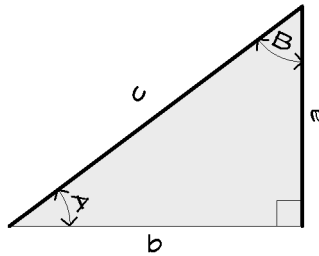Which parts are known?

Two Sides

Known: a & b	Known: a & c	Known: b & c

c = sqrt(a^2 + b^2)
A = atan(a / b)
B = atan(b / a)

b = sqrt(c^2 - a^2)
A = asin(a / c)
B = acos(a / c)
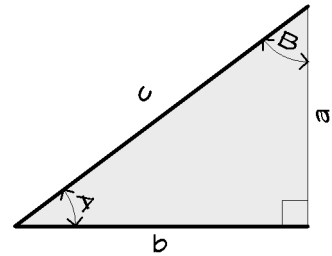
a = sqrt(c^2 - b^2)
A = acos(b / c)
B = asin(b / c)

One Side & One Angle

Known: a & A	Known: b & A	Known: c & A

b = a / tan(A)
c = a / sin(A)
B = 90° - A

a = b * tan(A)
c = b / cos(A)
B = 90° - A

a = c * sin(A)
b = c * cos(A)
B = 90° - A

Known: a & B	Known: b & B	Known: c & B

b = a * tan(B)
c = a / cos(B)
A = 90° - B

a = b / tan(B)
c = b / sin(B)
A = 90° - B

a = c * cos(B)
b = c * sin(B)
A = 90° - B