

AS319171-L

Revit Family Creation: A Step-by-Step Introduction (Just for Beginners)

Paul F Aubin

Paul F Aubin Consulting Services, Inc.

Learning Objectives

- Learn how to create a simple, fully parametric model family
- Learn how to use reference planes, constraints, and parameters
- Learn how to nest components and set up a parametric array
- Learn how to apply materials and other accoutrements

Description

Maybe you know the power of the Revit software's Family Editor but you've avoided it, or it's intimidated you until now. Wait no more—this lab will introduce you to the basics of Family Editor. You'll find that while Family Editor is extremely powerful, it can actually be great fun. In this short introduction to Family Editor, we'll create a Revit component family complete with constraints and parameters. Don't worry if you don't know what a constraint or parameter is—we'll cover that too. Whether you've never worked in Revit software before, or you've used Revit for a while but you've simply avoided Family Editor, this lab will teach you the basics of the Family Editor in a simple step-by-step fashion, and in the best way possible: hands on! (The session will use architectural examples, but the concepts apply to all flavors of the Revit software, including Revit LT.)

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 associate member of the AIA, 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 currently attend universities around the country.

Introduction

This class will introduce you to the basics of the Revit family editor. The family editor is extremely powerful but sometimes intimidating. In this introduction to the family editor, we'll work through the creation of a Revit component (loadable) family complete with constraints and parameters. Constraints and parameters can be thought of as adding “smarts” to the family.

Even if you are completely new to Revit, you have no doubt discovered how important families and the family editor are to your success in Revit. Everything you create in Revit is part of a family—consequently, understanding families and what it takes to manipulate them is a vital part of learning the software. Beginning with the difference between System and Component families this section will explore the critical concepts and terminology. We'll also take a quick look at what is provided in the Revit libraries and learn about family templates. After this brief introduction to terminology, the rest of the paper will focus on the component family creation procedures and strategies—presented in detailed step-by-step procedures. Using the concepts and techniques covered herein, you will learn how to begin tapping into one of the most powerful aspects of the Revit software package—the Revit family editor!

Everything in Revit is part of a Family

To get started with the family editor, it is important to understand some basic concepts and terminology. All elements in the Revit platform are part of a family and they fit into a clearly defined hierarchy. At the top level of this hierarchy, are **Categories**. Categories are pre-defined within the software and cannot be added, deleted or renamed. A wide variety of categories are included in Revit and distributed among a few overall master groups including model and annotation (but there are others). Model Categories include all elements that comprise your building model such as: Walls, Doors, Floors, Stairs and Beams. Annotation categories include items like Text, Dimensions and Tags. Categories are by definition very broad. It would not be enough to simply have a Walls or Doors category. These items come in all shapes, sizes and behaviors. Therefore, the next level of the hierarchy (the Family) is more specific. All Revit elements belong to a family. **Families are best thought of simply as a collection of like items sharing the same overall look and behavior.** Revit includes many families such as the “Basic Wall” wall family, the “Single-Flush” door family and many annotation families like “Text” or “Linear Dimension Style.” Even the views themselves like floor plans and sections are families in Revit.

All families can be thought of as one of two major kinds, based on their behavior: the **System Family** and the **Component (or Loadable) Family**. System families include anything that is built into the software and cannot be manipulated by the user in the interface. This can include model components like walls and floors, but also includes equally important items like floor plans, project data, text and levels. System families cannot be created or deleted. Their properties are pre-defined at the “factory.” However, most system families like walls, floors and dimensions can have more than one: **Type**. A type is our next level of hierarchy in Revit. **Think of it as a collection of variables (sizes, materials or other settings) saved to certain values and given a name for ease of reuse.** A Type provides a convenient way to switch several variables of a family at once. A family must contain one but can have several types; each with its own unique user-editable settings. So, while we cannot create or delete wall families, we *can* add, delete and edit the types associated with each of the provided wall system families. For example, “Basic Wall” is the most common wall family. In the out-of-the-box Imperial template files, there are several predefined *Basic Wall* types such as: *Exterior – Brick*

on CMU, Generic 6" and Interior – 5 ½" Partition (1hr). The Basic Wall definition establishes that its physical makeup has the same composition along its entire length and height and is composed of a layered assembly of one or more components. The specific details of this structure can vary widely from type to type as the names noted here imply.

Other system families vary considerably in their specific composition and features, but at the conceptual level they share the same basic characteristics: **the overall behavior of the object is defined by the system and cannot be redefined by the user**; however, the specific object-level parameters can be manipulated via the creation and application of type and/or instance variations.

As already noted, system families include both things that are part of the physical model in your Revit projects (like walls, floors and roofs) and other items that are not (like views, project data, text and levels). Another important characteristic in Revit is hosting. System families that also happen to be model elements are referred to as “**Host**” elements. A Host is an element that can receive or support or provide structure for other model elements. Hosts are often required for many of the component or loadable families like doors or windows which require wall hosts, or lighting fixtures which often require ceiling hosts.

Component (or “Loadable”) families include everything that is *not* a system family. Many component families are model elements, but they can also be annotation or other non-model elements as well. Component families can be “**host-based**” (require a host), or they can be “**free-standing**” (not requiring a host). Revit users *can* create, delete and modify component families (and their associated types). This is accomplished in the family editor and each family thus created can be saved to its own unique file (with and RFA extension). Like system families, component families can contain one or more types. They can also have instance parameters that vary from instance to instance (not part of the type). Unlike system families, they are *completely* customizable by the end user within the family editor.

Loadable component families; specifically, model components, are the primary focus of this lab and paper.

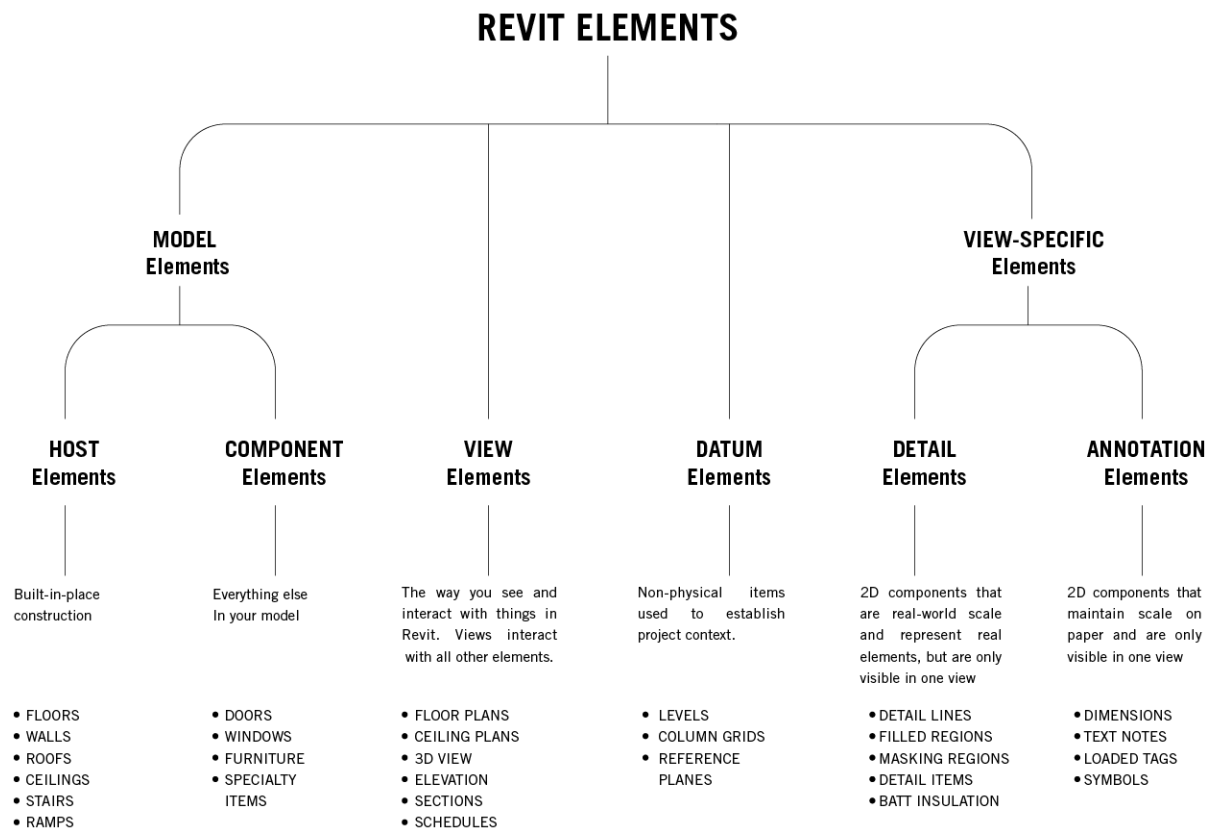
In addition to the system and component families, there is a third type of family in Revit called the “**In-Place Family**.” In-place families are similar to component families in terms of creation, editing and strategy. However, an in-place family is created directly within a project environment (not in a separate family file as component families are) and it *cannot* be exported to other projects. Further, you can create in-place versions of many system family categories like walls, roofs and floors. This capability allows the creation of custom or free-form shapes not otherwise possible in pre-defined system families. It also allows the specific surrounding context of the model to be referenced directly in its creation. You should only consider creating an in-place family for elements that are unique to a project with no possibility that you will ever want to reuse them in future projects. Also, in-place families as already noted, offer the only means to “customize” certain system families like walls or roofs. In-place families therefore prove effective for modeling unique existing conditions or very specialized and unique design scenarios. However, wherever possible, consider if the item you wish to create can be built using either predefined system elements or a loadable component family before resorting to an in-place

family. Often creating an in-place element seems like a good idea at the time only to later be the source of regret. We will *not* be exploring in-place families in this paper.

Revit Element Hierarchy

Here is a summary of the element hierarchy. The illustration is borrowed from the online help file a few releases back. There is a different version in the current help system, but I prefer this illustration as I believe it still does the best job of summarizing all the various kinds of elements in the Revit environment.

There are several kinds of elements. Each represents something fundamental to your project.



Element—Anything in your Revit project.

(Elements in italic below can be created and edited in the family editor)

Model Element—Something that represents the actual geometry of your building. It is created and displayed at full scale at its actual real life size.

Host Element—An element that can receive or support or provide structure for other model elements (built in-place construction) These are system families.

Component Element—An item inserted into a project (items that are pre-manufactured, purchased and installed). Can be freestanding or require a host as determined by the category and family.

Host Based Component Element—A Component Element that must be inserted on or into a Host.

Freestanding Component Element—A Component Element that can be inserted independently without a Host.

View Element*—An item in the Revit interface that allows you to see and interact with all other elements. Views conform to the characteristics of typical architectural and engineering drawing types like plan, section, elevation and schedule. Some View Element families allow customization of Types, many do not.

Datum Element*—Includes Levels, Grids and Reference Planes. These are used establish project context, limits, extents and the like. Datum Elements provide guidelines and limits for other elements within a project and can also include annotative qualities.

View-Specific Element—Something that is used to document, describe or embellish a view of your project. View-specific elements do not appear in any other views automatically. If you wish to repeat view-specific items in other views, you can copy and paste them.

***Detail Element**—A two-dimensional family typically representing a model element but at a level of detail that would be impractical to model. Detail elements appear only in the view in which they are added. Detail Elements remain their actual size as created and do not adjust scale with the view.*

Annotation Element*—Include text, dimensions tags and symbols. These items are view-specific (appearing only in the view in which they are added) and are used to notate, embellish, describe and document design intent within a Revit project. Annotation elements maintain a constant size relative to the plotting scale of the view to maintain a constant size relative the sheet on which they are placed.

**Level and Grid head tags, Section and Elevation head tags, model element Tags and Symbols (Generic Annotation families) can be created and modified in the family editor. Text and Dimensions cannot.*

Many of the branches in the diagram contain both system and component families. Naturally for a discussion on the family editor, we are therefore limited to considering only the non-system families. This includes all items on the Component Elements model branch, Detail Item families on the Detail Elements branch, Loaded Tags on the Annotation Elements branch and a few other miscellaneous elements as well like titleblock families or view tags and level head symbols.

Family Libraries and Resources

The first step to working in Revit in general and building families in specific is to become comfortable with the above terminology and hierarchy. Keep it handy as reference as you continue. But before you embark on the process of building family content, it should be noted that there are many families included with the software and many more resources available online. A quick search in Google will turn up hundreds of sites containing tips, tricks and downloadable content. Do take the time to explore the out-of-the-box offerings and some of many available sites as well if you have not already done so.

As has been noted, you cannot create or delete system families. All system families will already be in your project file. To add types that are not present to a system family, you either must

duplicate an existing type, rename and modify it, or import one from another project. To import from another project, you can use Transfer Project Standards (Manage tab) or copy and paste.

To use a component family from outside the project in your current project, you can load it from a family file (RFA) or copy and paste from another project. To load a family file, use the Load Family button on the Insert tab of the ribbon, or the contextual ribbon tab when a command is active. For example, if you click the Door tool (Architecture tab), the Load Family button will appear on the Modify | Place Door tab. This lets you load a door family and place it all in the same procedure.

In many cases, a family like the one you wish to create will already exist somewhere in the product or online in one of the myriad online resources. Most companies also maintain their own libraries of office standard content on their internal servers. Check with your CAD/BIM Manager to see what your firm offers.

The Recommended way to Get Started

Practical wisdom says that it makes more sense to begin with something in the library and either use it as-is or modify it to suit your needs. Typically, this will be easier than starting from scratch. In your day-to-day work when you are up against deadlines, this is often the best approach. Just be sure to take a little time to “vet” any unknown or newly downloaded content to ensure that it meets your office standards before using it on a live project.

In contrast to this approach, if you are new to creating families in Revit (as I assume you are because you are attending this session), then I recommend that you create your first few families FROM SCRATCH. By building the entire family yourself, you will learn more than simply modifying one. Furthermore, families can include very complex parameters and constraints that often link to one another in a chained and sometimes complex or even convoluted fashion. Even for seasoned family content authors, it can be difficult to dissect these often-complex relationships. Therefore, to avoid becoming discouraged, it is recommended that you start with a small simple example and work your way to more complexity over time.

For example, don’t start with a Door or Window family. These are more complex than they at first seem. Begin with something small, simple and boxy: like a simple piece of furniture or equipment.

Family creation procedures

The basic process for creating a family is as follows: *decide what type of family you need*. This will include deciding what it should look like, how much detail to include and whether the graphics or level of detail should change in different views. You can start by sketching out the family you intend to create (yes on paper – see Figure 1) and make notes about its requirements.

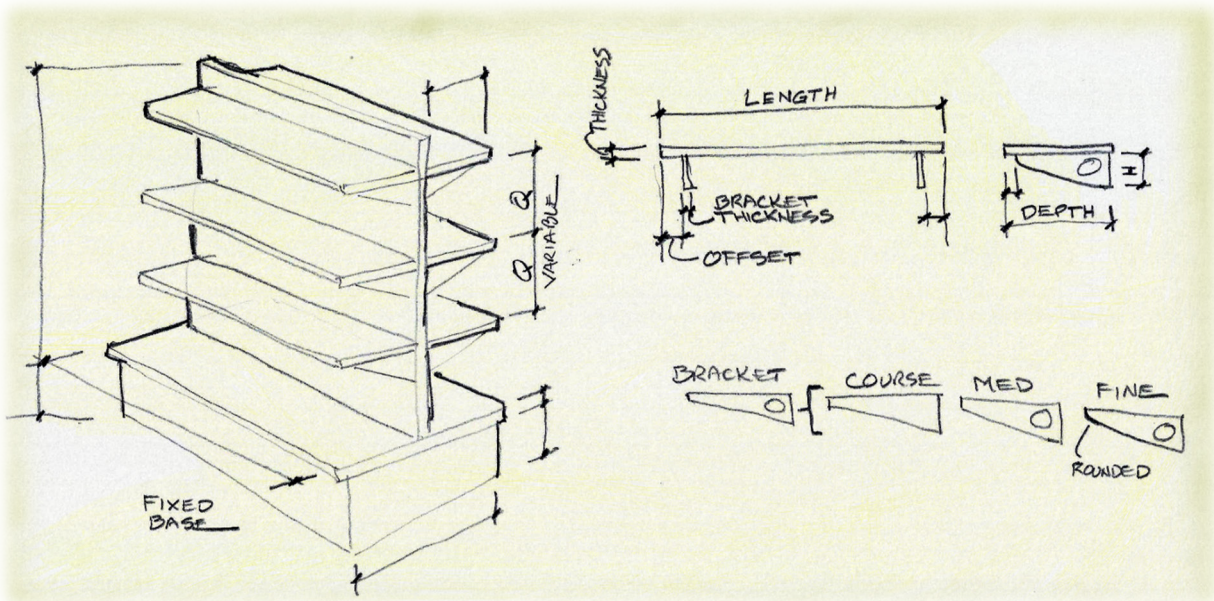


Figure 1

Next, create a new family file from the appropriate template or open an existing family file similar to the one you wish to create and save as. The choice of family template is important. The templates included with the software are provided with the product. Each contains basic settings, behaviors and in many cases some simple geometry or reference planes. The geometry included (like a sample length of wall) is only for reference and does not get inserted with the family when used in a project. While it is possible to change the category of family after creation, it is best to choose wisely at the start. Try to choose the most appropriate category selecting: *Generic Model.rft* only if no other suitable category can be determined. Unlike category, the hosting behavior of a family file *cannot* be changed after it is created. So, if you are not certain that you want the family you are creating to require a Host, it is safer to build it without one. In other words, if you choose *Casework wall based.rft* as the template, the family you create will *always* require a wall to be inserted. If you think you might like to use the cabinet as a freestanding piece of casework, choose the *Casework.rft* template instead. You can always use the Align tool to move the non-hosted cabinet to a wall face later. You cannot later decide to detach the hosted casework item from its host wall.

Once you have decided what you want to build and created a new family file based on an existing file or the appropriate new template, you are ready to create your family reference planes, parameters and geometry. It is usually best to start with the framework. If you begin with an existing family, delete anything you don't need first. Then *in both existing and new families, add the Reference Planes you will need*. Reference planes provide the skeleton for your family. Some templates already contain basic reference planes. You can use these as-is or modify them. *The proper procedure is to manipulate or create reference planes, optionally constrain or assign parameters to these planes, and then create geometry and lock it to the reference planes*. In this way, the reference planes drive the geometry. This is the most reliable, "best-practice" way to build your family files.

Once you have laid down your reference plane framework and assigned parameters and constraints, test the family by “flexing” it. This is done in the “Family Types” dialog which you can access from the Family Types button on the ribbon. To flex the model, simply try different values for each parameter and then apply. If the framework moves or adjusts the way you expect, everything is good. Otherwise, undo, and try to fix the problem. (We will see several examples below.)

When all geometry and parameters have been created, applied and flexed, you are ready to save the file and load it into a test project (below I use the **100 Sandbox.rvt** file for this purpose). If necessary, return to the family editor to make any adjustments and then reload, otherwise your family file is complete.

Constraints and Parameters

In its simplest form, a family can be a static graphic or symbol. Such a family would be drawn the way it was intended to look regardless of the circumstance. The out-of-the-box Chair-Breuer is one such example. There are no types or user-editable dimensions in this family. However, one of the things that make families so powerful is their ability to use variables to help them conform to varying circumstances. This is done using constraints and parameters. Any of the out-of-the-box door or window families would be good examples. Each includes named parameters that can receive different values. These values can be assigned at either the type or instance level and when “flexed” the family will adjust in response. Both parameters and constraints can be used for this purpose. While each of these terms has several possible meanings, in the context of Revit the following definitions are suitable to our discussion.

Constraint—is a fixed rule that can only be manipulated by editing the family file.

Parameter—creates a rule or relationship that has user-editable properties.

Each of these is a rule applied to some part of a family’s geometry or behavior, but a constraint cannot be manipulated by the end user, and a parameter can. For example, if you were working with a door family and you wanted to ensure that a vision panel was 10" from the door edge regardless of the door’s width, you would use a constraint within the family editor to achieve this. On the other hand, if you want to allow the same door family to have varying (flexible) sizes for height and width of the vision panel, you would use parameters instead. By making vision panel width and height parameters and using them to drive the geometry within the family, the user can exercise much greater control than would otherwise be possible. However, the location of the vision panel with respect to the edge of the door would remain fixed via the constraint.

Solid and Void Form Geometry Types

Geometry in families consists of solid and void forms. Solid forms represent the actual physical parts of the family and void forms are used to carve away portions of the solid forms. For example, you could create a solid form box, and then use a void form to cut a hole in it like a donut. Both solid and void forms come in five varieties. These include: Extrusion, Blend, Revolve, Sweep and Swept Blend (see Figure 2). We will use some of these in the tutorials below.

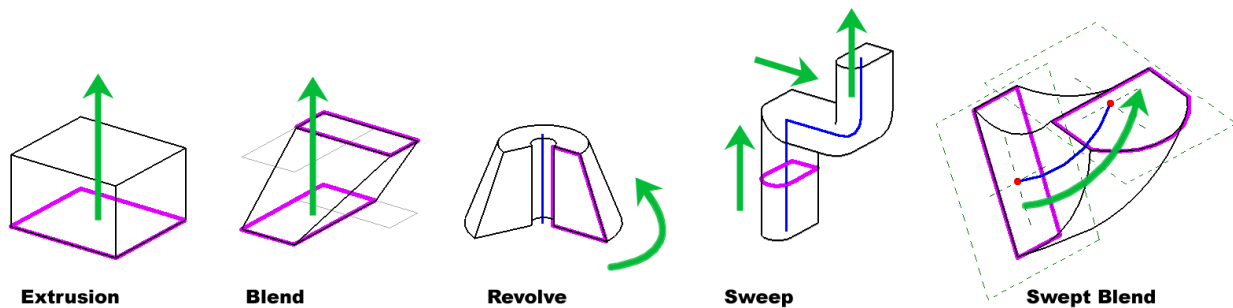


Figure 2 – Forms available in the family editor

An extrusion is a sketched shape pushed along a distance perpendicular to the sketch plane. A blend is similar except that instead of a single shape, you have both a top and a bottom shape and the 3D form transforms (or blends) from one to the other along the perpendicular height of the form. A revolve spins a sketch shape around an axis. The revolve can be a full 360° or a partial arc. A sweep pushes a shape (sketch or loaded profile) along a sketched path. The shape is perpendicular to the path. A swept blend combines features of both the blend and the sweep. The form morphs between two profiles or sketches as in a blend but can follow a non-linear path. Unfortunately, the swept blend path can only contain one segment unlike the sweep. This means that complex forms require a spline path. Using a combination of solid and void forms you can create nearly any three-dimensional shape.

Family Types

As we have already pointed out above, families can contain types. A type is a saved and named collection of values for the parameters within a family. You can add as many types as you wish. Types can be added within the family editor, in a catalog or even later in the project.

Nested Families

You can build complex forms using a combination of the solid and void forms available in the family editor as noted above. However, managing a complex form in a single family can become cumbersome. In many cases, it makes sense to break your object into discreet parts and build the parts as separate families. You can then insert these simpler families into another family that represents the whole. This is referred to as nested families. When you manage your complex families in this way, you gain more control and flexibility.

Subcategories and Visibility parameters

Any family you create or load from a library will belong to a certain category. Each of the elements within the family can belong to a subcategory within the family. Subcategories provide an extra level of visibility and graphical control over the parts of a family. For example, in the door families included with the software, there are several pre-defined subcategories. One such subcategory is the Plan Swing. Using this subcategory, it is possible to make door plan swings a lighter pen weight across a project regardless of the specific family. This helps enforce standards and simplifies such changes.

Visibility parameters are another way to control elements within a family. Sometimes it is useful to see part of the family only in certain circumstances. For example, you could create a casework family where hardware was an optional display component. In this case, a visibility parameter would be assigned to the hardware elements within the casework family and could then be toggled on or off by the user depending on whether they needed to show it.

Tutorial

That completes the introductory materials. The remainder of the paper is tutorial which will allow you to follow along with complete step-by-step instructions. Explanations are given in-line with the steps, but the steps are highlighted to help them stand out. Many of the concepts discussed in the preceding topics will be showcased in the tutorial that follows.

Before we begin

- This is an INTRODUCTORY class. If you already know how to use the Family Editor and have already created your own family content in Revit, most of the topics covered in this lab will likely be review to you.
- I assume NO PRIOR KNOWLEDGE of the family editor. We will start at the beginning.
- I will be teaching the course using Architectural examples. However, most if not all concepts should apply equally to other flavors of Revit. This includes Revit LT.
- **This is a Hands-on lab. If you don't wish to follow along, please pass the mouse to the person sitting next to you.**

Family Editor 000

Prerequisites and Setup

If you were going into the shop to build some cabinetry, you'd want to make sure you had all the tools you needed and that the shop was in order before you started. Let's do the same in Revit.

This paper was authored in Revit 2020.1, but most steps should work in other flavors or versions. (If you use an older version of Revit, you can visit: <http://paulaubin.com/au/> to find older versions of this class including the dataset in earlier Revit versions).

Steps have been kept brief to allow the tutorial to flow without interruption. Please refer to the topics above for further details on the concepts showcased herein.

1. If Revit is not already running, launch it now.
2. From the File menu, or on the Recent Files screen, in the Models area, click Open.
3. Browse to the folder where you unzipped the dataset files and open the file named:
100 Sandbox.rvt.
4. If necessary, open the *Axon without Roof* view. If other view tabs are open, close them.

The project file will remain open as we work through the next several lessons. As you build family content, you will want to test if frequently. Having a project file open in the background is an excellent way to do this. I like to minimize it so that it stays out of my way till I need it. There is nothing special about a "sandbox" except what you put in it. You can open your standard office template, add a few walls and other items and save it as a sandbox. *It is that simple.*

5. Click and hold down the *Axon without Roof* tab and tear it away from the application frame.

This will make this view a floating window.

6. Minimize this view.

And now, on with the good stuff! Let's start building some family content!

Family Editor 101

Getting Started with the Family Editor

It all starts with a box. A box you say? Yes, a box. The first thing you need to learn to build in the family editor is a simple box. But not just any box, we'll be building a **Parametric Box**. Ooo ahh.

7. From the File menu, choose: **New > Family**.

The "New Family – Select a Template File" dialog will appear. Template files do some very important things to get you off to a good start in your families. For now, we will just choose: Generic Model. (Don't be fooled by this choice however. Template choice is very important and was discussed in a little more detail in the "Family creation procedures" topic above).

8. Select: **Generic Model.rft** from the list and then click Open.
9. Type WT and then ZA.

These two keyboard shortcuts tile the windows and zoom them all to fit the available space. If you are not using keyboard shortcuts, you should consider learning some. Big time savers!

10. From the File menu choose **Save** (or CTRL + S) and name the file: **Box.rfa**.

Create Reference Planes

Reference planes give our family its structure. Think of it as the skeleton of your family. Always start with reference planes.

1. On the Create tab, click the Reference Plane (*not* Reference Line) button (or type: RP).
2. Working in the floor plan view, draw a vertical reference plane on the left of the centerline. Edit the dimension to make it: **3'-0"** from center (see Figure 3).
3. Click in the "Click to name" onscreen control, type: **Left** and then press ENTER.

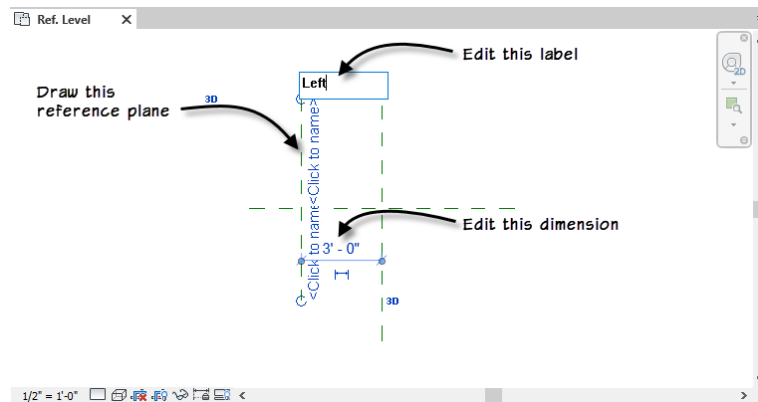


Figure 3

4. Repeat to create a horizontal one below at: **2'-0"** from center. Name it: **Front**.
5. Click the Modify tool to cancel.
6. Mirror the horizontal one to the opposite side and then name: **Back**.
7. Repeat for the remaining one, name it: **Right** (you'll end up with four total; see Figure 4).

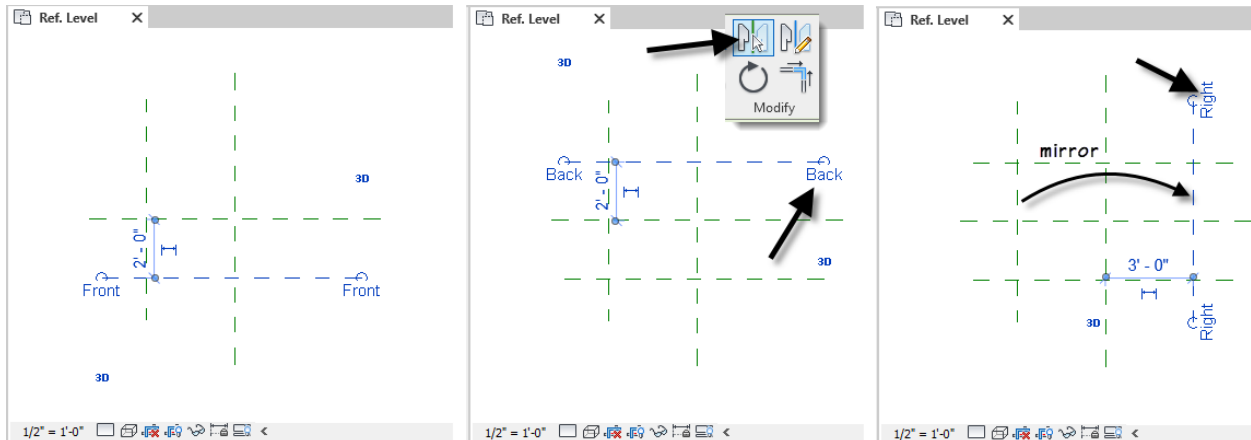


Figure 4

8. Add dimensions (Modify tab or type: DI) to each group of reference planes. Two vertical and two horizontal. There should be an overall and a continuous string including the centerline in each direction (see Figure 5).

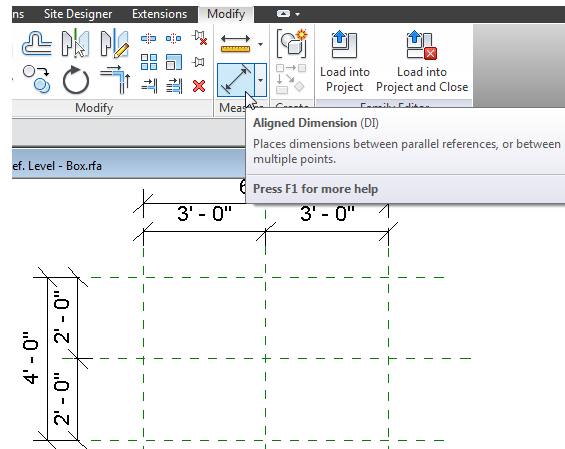


Figure 5

9. Select each of the continuous strings and toggle on the Equality (click the small EQ icon see the left side of Figure 6).

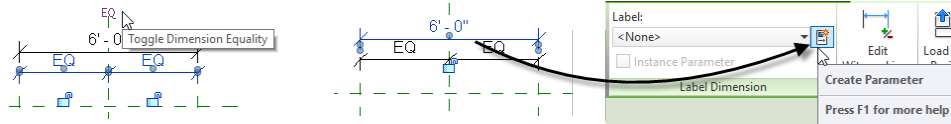


Figure 6

10. Select the 6'-0" overall dimension. On the ribbon, next to the Label drop-down, click the small Create Parameter icon (see the right side of Figure 6).
11. In the “Parameter Properties” dialog that appears, name the new parameter: **Width**, choose the **Instance** radio button (do *not* check “reporting”) and then click OK (see Figure 7).

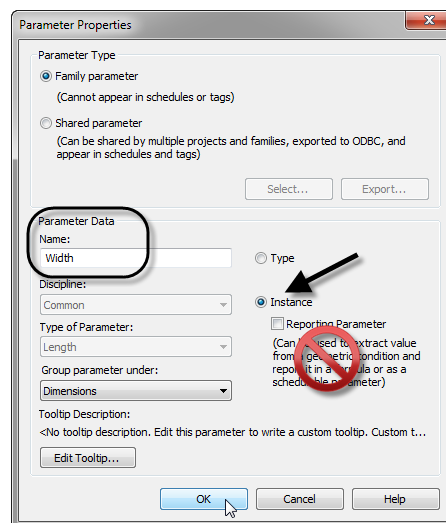


Figure 7

12. Repeat for the other dimension calling it: **Depth**. Also make it an Instance parameter.

Flex the model

What we have so far is a rectangular formwork defined by the reference planes. The two labeled dimensions are “parameters” that will allow these dimensions to be flexible. The two “EQ” dimensions are constraints that will keep everything centered. Let’s test it out. When you test your flexible family, it is called “flexing” the model.

1. On the ribbon, click the Family Types button (see Figure 8).

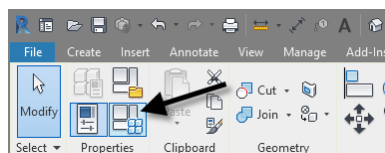


Figure 8

2. Input a different value for both the **Width** and **Depth** fields and then click the Apply button.

The locations of the Reference Planes should adjust but stay equally spaced from the center. If it does, congratulations, you have completed your first set of correctly configured parameters.

3. Reset the Width to **6'-0"** and the Depth to **4'-0"** and then click OK.

Repeat the process

Now we'll shift to elevation and add a reference plane to control the height.

1. In the *Front* elevation view, on the Create tab again, create another reference plane running parallel to and: **3'-0"** off the ground. Name it: **Top**.
2. Add a dimension between the new reference plane and the Ref Level already in the file. Label the new dimension with a new parameter and call it: **Height**. Make it an **Instance** parameter (not reporting).

CATCH UP! If you get behind, look for these boxes. I have saved versions of the files at various stages of completion. You can open the file completed to this point named: **101 Box_A.rfa**. First, from the File menu, choose **Close** (or press CTRL + W). Then open the file indicated here, press WT then ZA.

Add Geometry

How about some geometry? With a good framework in place, you are ready to add your first solid form to the Family.

1. In *Plan* view, on the Create tab, click the Extrusion button.
2. On the Modify | Create Extrusion tab, on the Draw panel, click the Rectangle shape (see Figure 9).

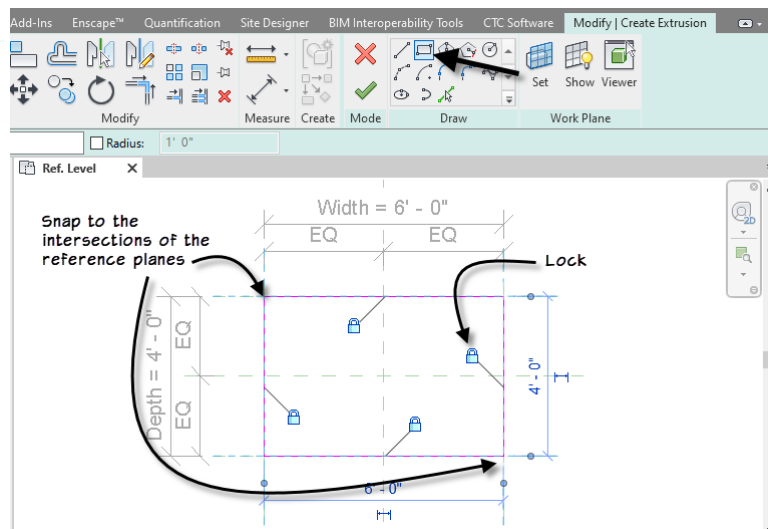


Figure 9

3. Snap to the intersection of two of the reference planes for the first corner, and then snap to the opposite intersection for the other corner.
4. Lock all four sides.
5. On the ribbon, click the big green checkmark button to finish (see the top of Figure 10).
6. Zoom the 3D view to fit. Click a corner of the ViewCube to do this quickly (see the bottom of Figure 10). Turn on Shading in the 3D view.
7. In the *Front* view, select the box. Drag the triangle shape handle grip at the top and snap it to the reference plane. Lock it (see Figure 11).

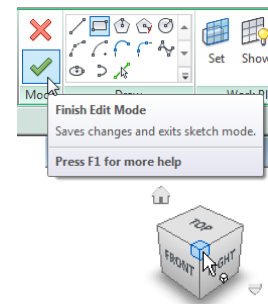
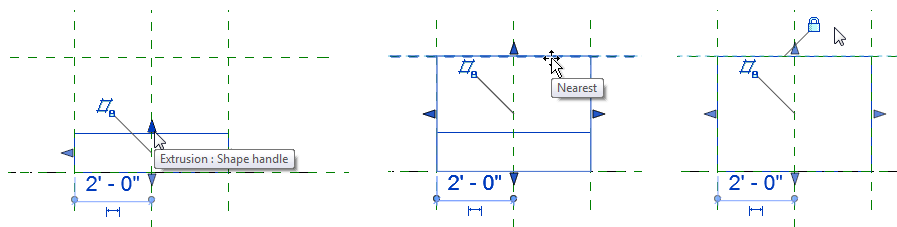
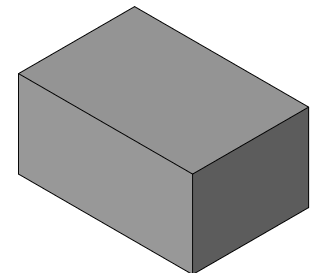


Figure 10



Stretch the height to snap to the Reference Plane and lock it

Figure 11



Load into a Project

The best way to flex any new family is to load it into a project. This is where the *Sandbox* file we opened earlier comes in.

CATCH UP! You can open the file completed to this point named: **101 Box_B.rfa**.

If you did not open the **100 Sandbox.rvt** project file earlier, please open it now.

1. With your Box Family active, on the ribbon, click the Load into Project button (see Figure 12). Place it anywhere inside the building.

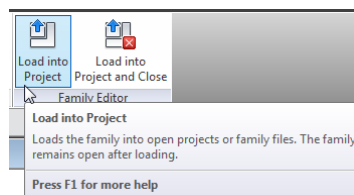


Figure 12

2. Make a couple copies. Try moving or rotating them.
3. On the Properties palette, edit the: Width, Depth and Height (see Figure 13).

Notice each instance can have its own dimensions (this is because we used Instance parameters). Expand the *Families* branch of the Project Browser. Under *Generic Models* you will see the *Box* family.

Congratulations. You have created your first parametric model family. Granted it is just a simple box, but that simple box can become lots of useful things as we'll see in the next topic.

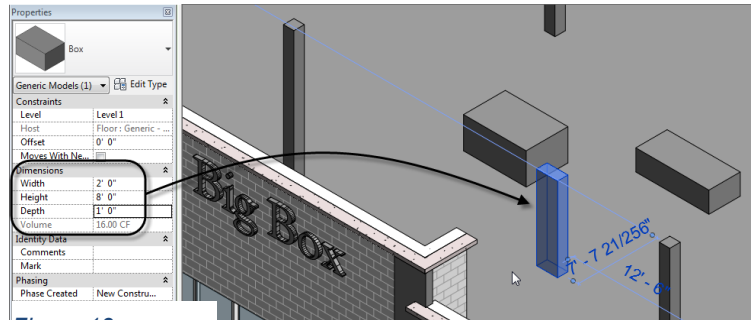


Figure 13

4. Minimize the *100 Sandbox.rvt* file view. Your Box should still be opened with views tiled.

Family Editor 102

More than a box

Think of the box we just created as a jig. Naturally there are many useful forms we could create. But perhaps none is more versatile than a simple flexible box. And this simple box becomes a seed from which we can derive many other families.

1. Save the file as: *Shelf.rfa*.

CATCH UP! You can open the file completed to this point named: *102 Shelf_A.rfa*.

2. In the *Front* elevation, create a new reference plane below the Ref. Level about: 4" away. Name it: **Underside**.
3. Select the Extrusion. On the ribbon, click the Edit Work Plane button. From the Name list, choose: **Reference Plane: Underside** and then click OK (see the left side of Figure 14).
4. At the bottom of the *Front* view, click the Reveal Constraints icon.

A red border will surround the view window.

5. Click the Top reference plane and then unlock the padlock icon (see the right of Figure 14).

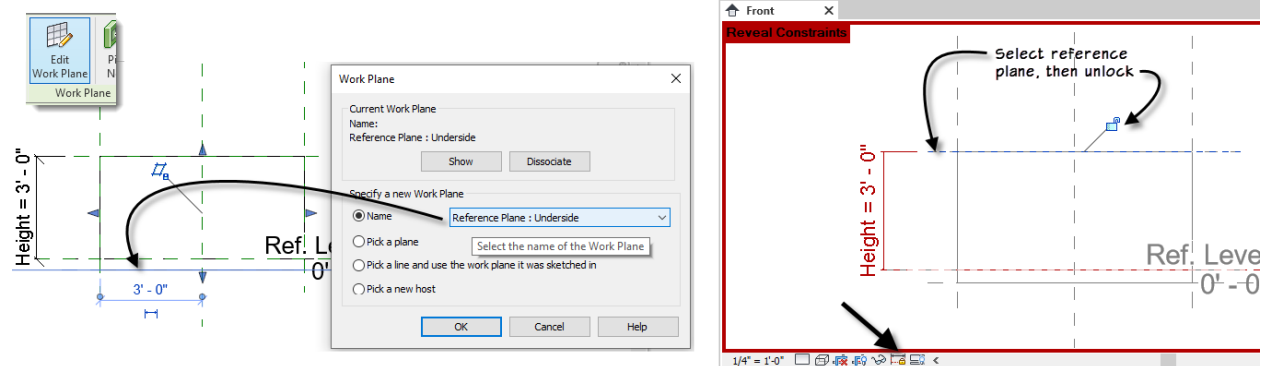


Figure 14

6. Deselect the reference plane. Select the extrusion and then drag the top control grip down and snap it to the Ref. Level.
7. Lock the new position of the Extrusion shape handle. You should have a very short box completely below the Ref. Level (see Figure 15).

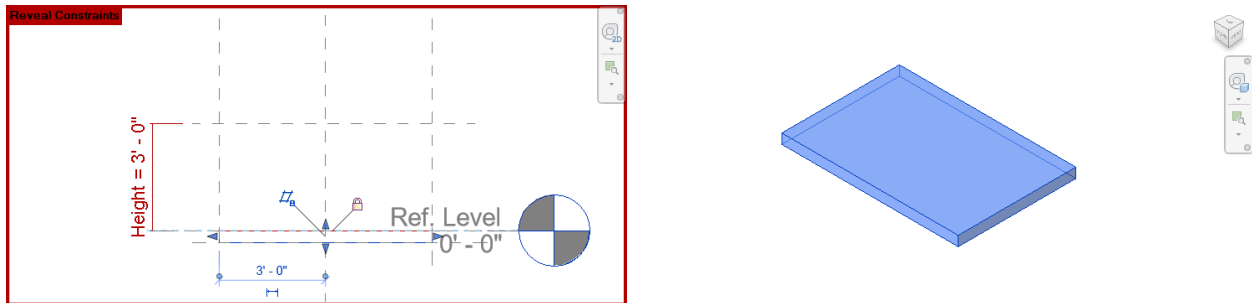


Figure 15

8. Click the Reveal Constraints icon in the *Front* view again to toggle it off.

Re-Constrain the Height

Let's apply the Height parameter to our new reference plane.

1. Delete the Top reference plane.

This also deletes the Height dimension.

2. Add a Dimension between the Underside Reference Plane and the Ref. Level.

Be VERY careful to dimension to the reference plane and level NOT the box. Use TAB as necessary.

3. **Label it** with the existing: **Height** parameter.

This will immediately “flex” the extrusion and make it 3'-0" tall, but in the negative direction!

There are other ways to achieve a similar effect. This approach was taken to keep the extrusion height a positive number while keeping the insertion point of the shelf at the top surface of the shelf.

CATCH UP! You can open the file completed to this point named: **102 Shelf_B.rfa**.

Flex and Save

While it is possible to have a 4 x 6 shelf, it is pretty unlikely that we would have a shelf that is 3'-0" thick. Let's return to Family Types and set the sizes to more reasonable values.

1. Open Family Types and flex. Input values of Width = 4'-0", Depth = 1'-4" and Height = ¾" and then click Apply.
2. Assuming there are no errors, click OK and then **Save** the file (see Figure 16).

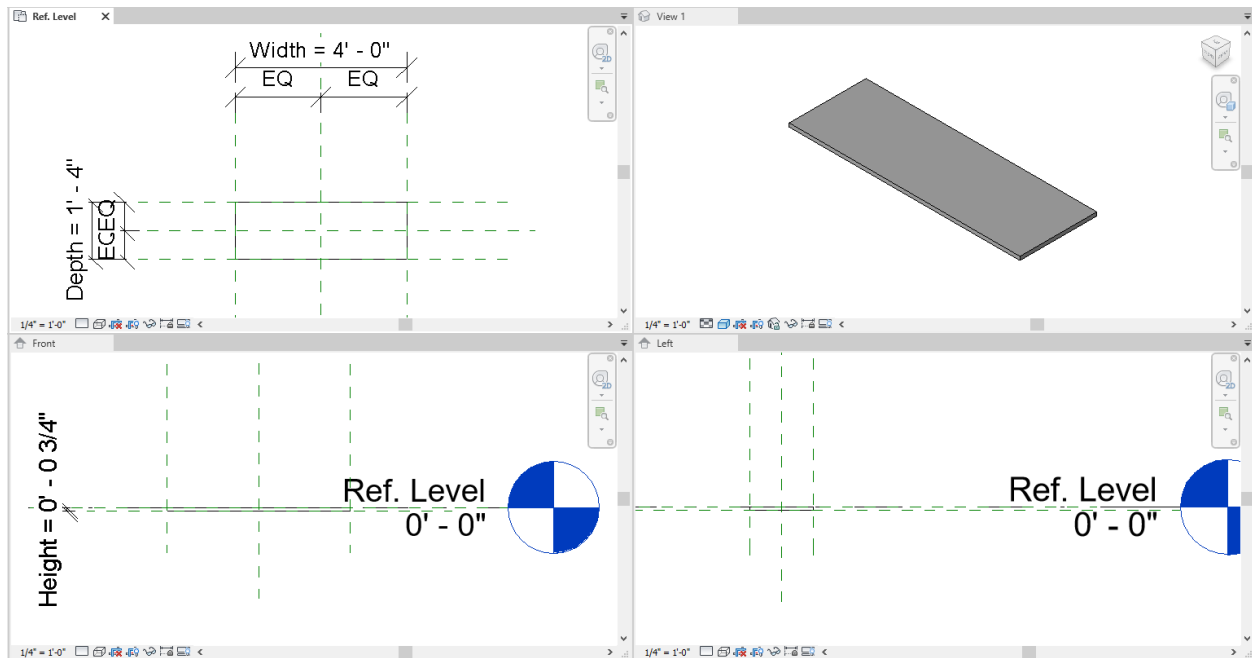


Figure 16

So, with minor adjustment, our simple box becomes a shelf. Now let's give it some supports.

CATCH UP! You can open the file completed to this point named: **102 Shelf_C.rfa**.

Family Editor 103

Solids and Voids

To create a support bracket for our shelf, we'll try a new form: a Solid Blend. However, it starts with the same basic armature that we used for the box. So rather than start from scratch, let's save it as a new file.

1. From the File menu, choose **Save As > Family**. Call it **Bracket.rfa**.

Since we are starting with an existing file, we need to make a few adjustments. Ultimately in your own work, you can proceed as we are doing here, or you can build each new family from scratch including all reference planes and dimensions. The choice is up to you.

2. Select the Underside reference plane and on the Properties palette, rename it: **Bottom**.
3. In the *Front* elevation, with the reference plane still selected, click on the Height dimension.

It will become an editable temporary dimension which allows you to flex it directly.

4. In the dimension, input a value of: **1'-0"** and press ENTER.

CATCH UP! You can open the file completed to this point named: **103 Bracket_A.rfa**.

OK, it doesn't look much like a bracket yet. But as we'll see it has all the raw materials we need to quickly transform it. We need to first add one more reference plane and adjust a few parameters.

5. In the *Front* view, zoom in if necessary and add a new reference plane: $\frac{3}{4}$ " below the Ref. Level. Add a dimension between this one and the **Ref. Level** (not the box itself – use TAB)
6. Instead of adding a parameter, simply lock it.

This will make this dimension “fixed” making it a constraint (see Figure 17).

Important: Be sure to dimension to the Ref Level and NOT the box use TAB.

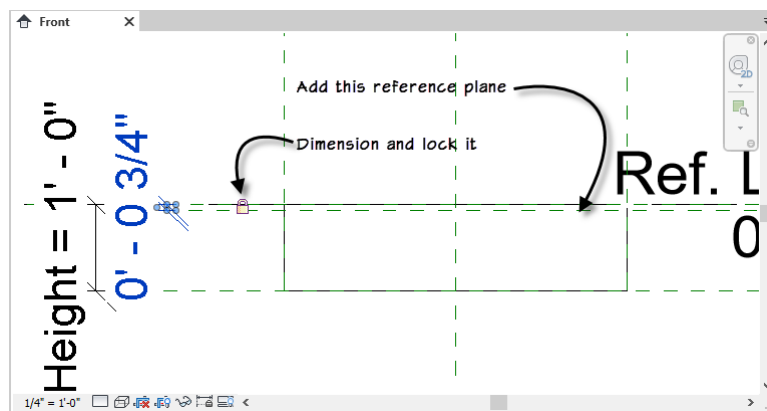


Figure 17

7. Open the Family Types dialog. Select the Width parameter and at the bottom, click the Modify icon (small pencil). Change the name to: **Thickness** and then click OK twice.
8. In the *Front* elevation view, **delete** the Extrusion. Save the file.

CATCH UP! You can open the file completed to this point named: **103 Bracket_B.rfa**.

Create a new Blend

Let's now build some new geometry.

Work planes are very important when creating solid forms. We saw above that we can edit the work plane of a solid element after it is created. Another approach is to set a work plane current *before* you create the form. This gives much more control. Any *named* reference plane can be used as a work plane.

1. In the *Front* view, on the Create tab, on the Work Plane panel, click the Set button.
2. In the “Work Plane” dialog, from the Name list, choose: **Reference Plane: Back** and then click OK (see Figure 18).

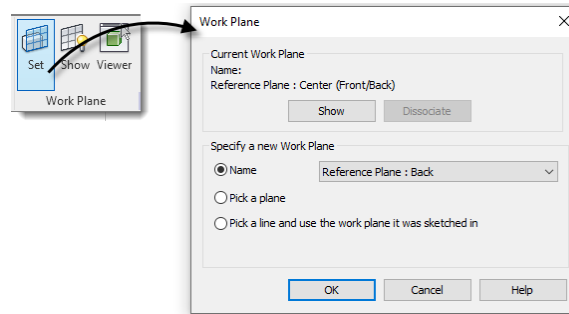


Figure 18

3. Stay in the *Front* elevation and then on the Create tab, click the **Blend** tool.
4. On the Draw panel, select the Rectangle shape and snap to the opposite corners of the rectangle defined by Ref. Level, Bottom, Left and Right Reference Planes. **Lock all sides** (see the left side of Figure 19).

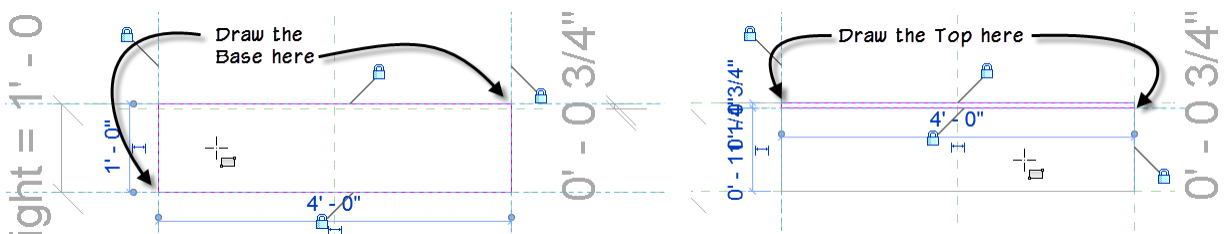


Figure 19

5. On the ribbon, click the Edit Top button.
6. Choose Rectangle again and create a thin rectangle between the Ref. Level, Left, Right and the new Reference Plane at $\frac{3}{4}$ ". **Lock all sides** (see the right side of Figure 19).
7. On the ribbon, click the green checkmark button to finish.
8. Zoom in on the Left elevation view. Use the control grip to drag the depth of the blend and snap it to the **Front** Reference Plane. **Lock it** (see Figure 20)¹

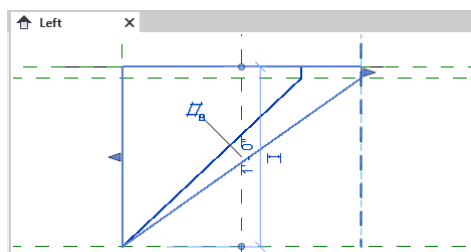


Figure 20

¹ If your blend is pointing the wrong way, select the Back reference plane and drag one of its endpoints all the way past its other end. This will "flip" the normal direction of the work plane.

CATCH UP! You can open the file completed to this point named: **103 Bracket_C.rfa**.

Flex the Blend

While it is certainly possible to have a bracket in this shape, its proportions are perhaps a bit off.

1. Open Family Types.
2. Change the Thickness to: $\frac{1}{4}$ ", the Height to: 4". Leave the Depth at: 1'-4" and then click OK (see Figure 21).

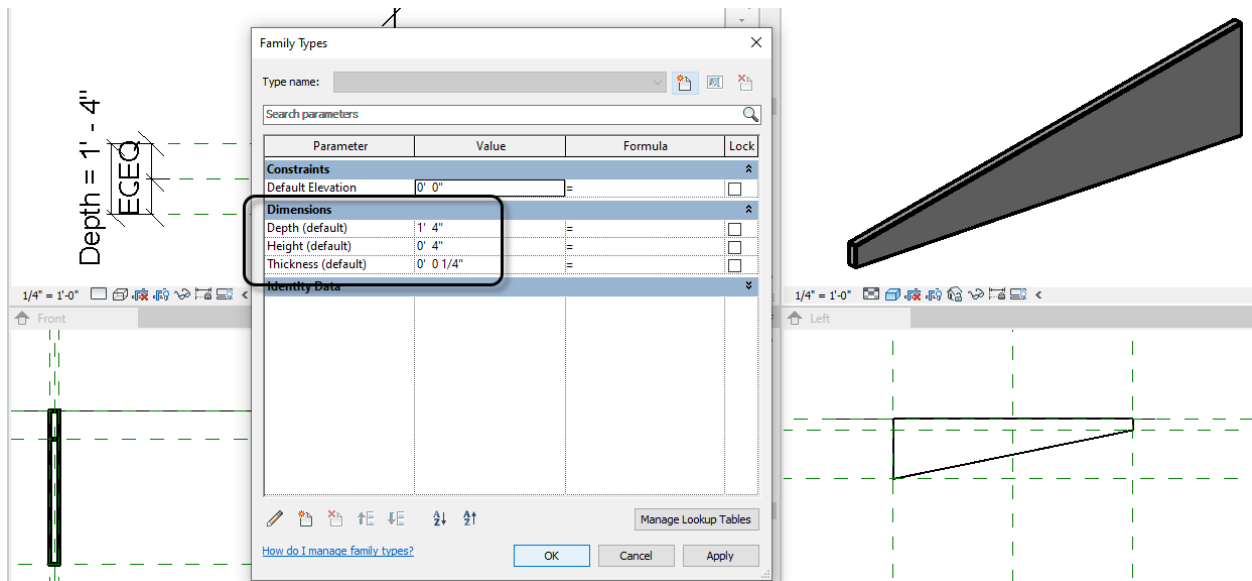


Figure 21

3. Save the file.

Add a Void Form

There are many other solid forms available. Feel free to explore more on your own. There are also void forms. Voids cut away from solids. Let's do a simple example.

1. In the *Left* elevation view and zoom in on the bracket toward the back.
2. Add a vertical reference plane 2" to the right of the Back reference plane. Name it: **Void**. Dimension and lock it to the Back.
3. In the *Front* view, on the Create tab, click the Set Work Plane button and chose the new Void reference plane.
4. Add another reference plane 1 $\frac{3}{4}$ " below the Ref Level. Dimension and lock it to the level. Name it: **Void Center**.
5. On the Create tab, click the **Void Forms** dropdown and then choose: **Void Revolve**.
6. On the Draw panel, click the rectangle tool and draw a small rectangle from the left to Right reference planes. Snap the bottom edge to the Void Center reference plane and make the top edge 1 $\frac{1}{4}$ " above this.

7. Click the Modify tool to cancel. Select the circle and on the Properties palette, check the Center Mark Visible checkbox (see the left side of Figure 22).
8. On the Modify | Create Void Revolve tab, click the Axis Line button.
9. In the Draw toolbox, click the Pick Lines tool and then click on the Void Center reference plane (see the right side of Figure 22).

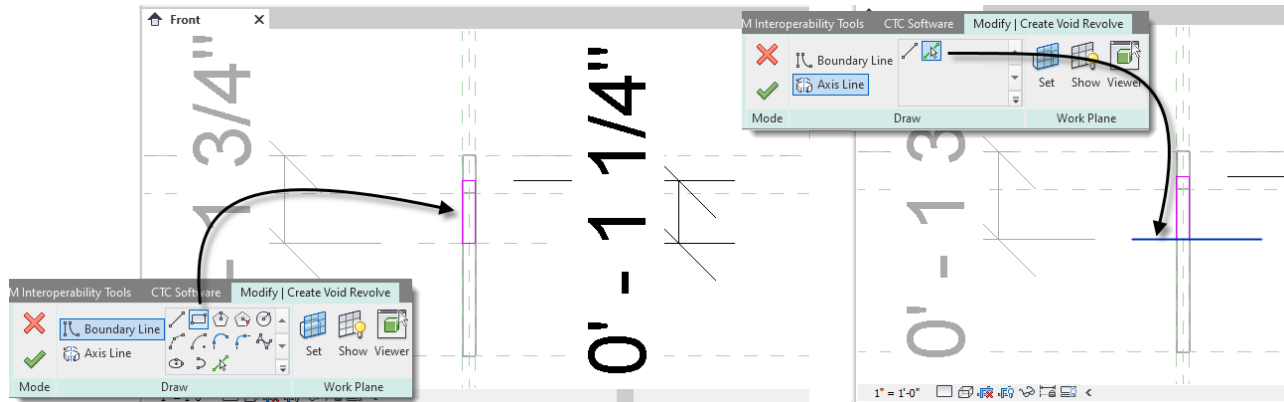


Figure 22

10. Click the Finish Edit Mode button.

The result of this sketch is a circular form. A revolve takes your sketch and spins around the axis line. The default is 360°.

11. Click away from the form anywhere in empty space to deselect it.

Notice how the void immediately engages with the solid and cuts a hole from it. This is just one way to create such a hole. We could also have sketched the bracket form in *Left* elevation with a circular inner loop instead. When you sketch a shape with two loops, the inner one becomes a hole within the shape of the outer one.

Adjust the Origin

The way our family is currently defined, the origin point which controls the insertion of the family is at the intersection of the two center reference planes. For a shelf bracket, the Back plane and Center (Left/Right) would make a better choice.

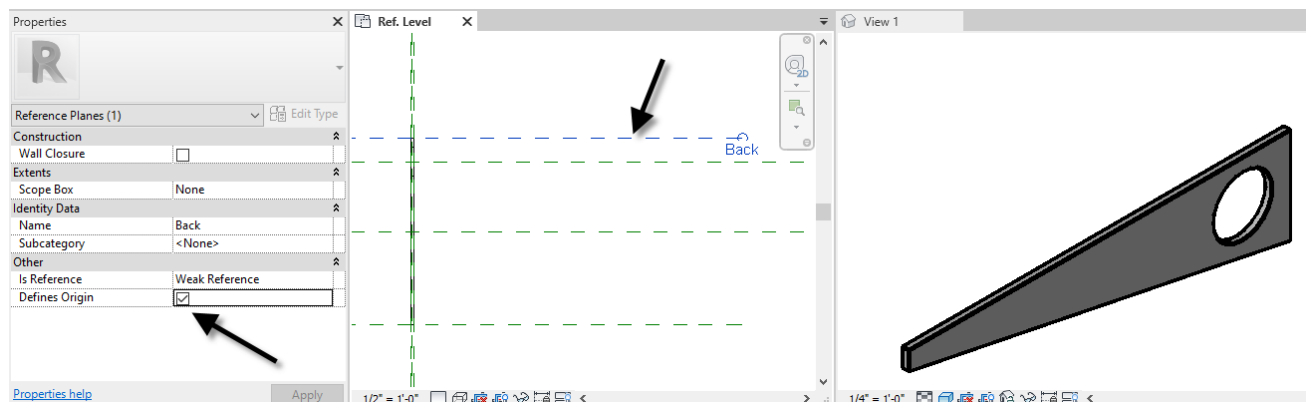


Figure 23

1. In the Plan view, select the Back reference plane. On the Properties palette, check the **Defines Origin** checkbox (see Figure 23).
2. Save the **Bracket.rfa** file.

Family Editor 104

Nesting Families

To make more complex forms, you will often want to model it in pieces. Break a large item into smaller parts, model each as a separate family and then pull them all together in a single host family. This is referred to as “nesting” families.

1. Keep your Bracket family open. Reopen your **Shelf.rfa** Family file.

CATCH UP! You can open the file completed to this point named: **104 Shelf_A.rfa** for the shelf and **104 Bracket_A.rfa** for the bracket.

2. Using Switch Windows, switch over to one of the views for the Bracket family.
3. On the ribbon, click the **Load into Project and Close** button.

If you click the Load into Project button instead, it will still load the family, but the other family will remain open and you will need to later manually close the Bracket family.

4. In the dialog that appears, check the **Shelf** family only (*not the Sandbox*) and then click OK (see Figure 24).

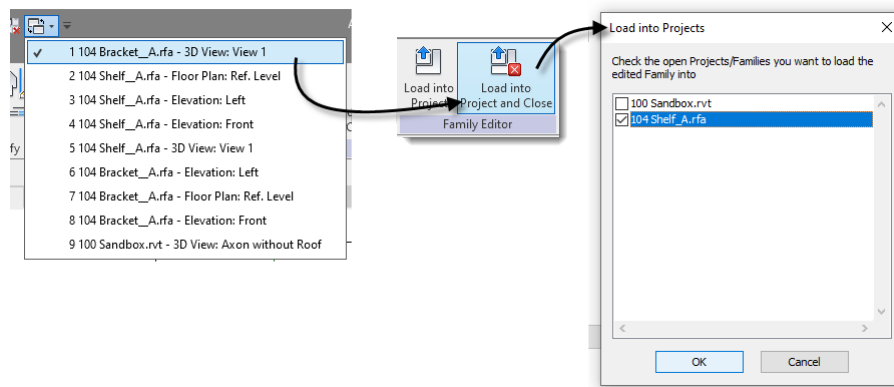


Figure 24

This should run the Component tool. If the plan view did not activate, click in the plan to activate it. (If you need to start the tool again, on the Create tab, click the Component tool).

5. In the *Plan* view, place an instance of the bracket. The exact location is not important. Get it close.
6. Use the **Align** tool to snap the back of the bracket to the Back reference plane in the Shelf family. **Lock it**.

7. Create a dimension between the Left reference plane and the center of the bracket.
8. Cancel the dimension command, select the bracket and then edit the temporary dimension to: **4"**. **Lock it** (see the left side of Figure 25).
9. Open the *Left* elevation view, on the Modify tab, click the Align tool.

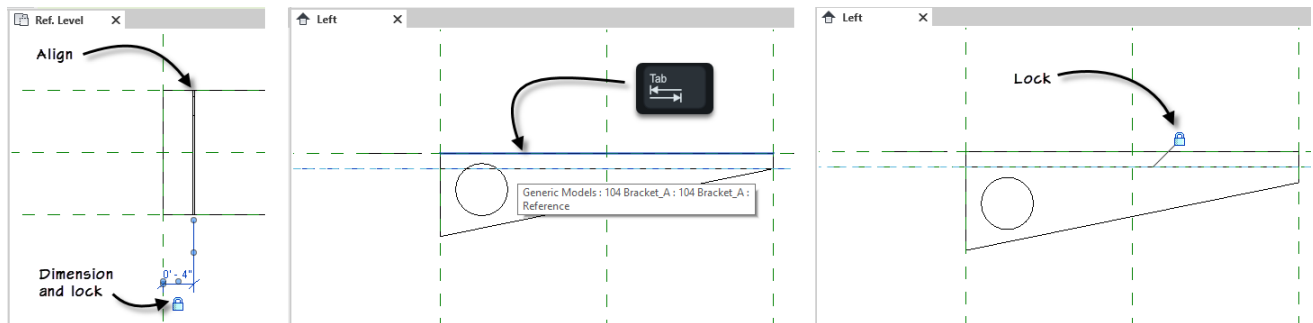


Figure 25

10. First select the Underside reference plane for the alignment reference. Then be sure to use the TAB key and make sure you are aligning the top of the Bracket and *NOT* the Extrusion: Shape Handle. Lock it (see the right side of Figure 25).
11. Mirror the Bracket to the other side and constrain it the same way. You must repeat all alignment and dimension steps and lock them.
12. Open Family Types and flex.
13. Save the Shelf file and if necessary, close the Bracket file.

CATCH UP! You can open the file completed to this point named: **104 Shelf_B.rfa**.

Linking Parameters between Host and Nested

Using the dimensions and locking to the reference planes keeps the brackets positioned at a consistent location as the shelf changes length. You can also establish a relationship between the parameters in the host file (the Shelf) and the nested family (the Bracket).

1. Select both Brackets onscreen. *Left* view is a good place to do this.
2. On the Properties palette, locate the small Associate Family Parameter button to the right of the Depth field and then click it.

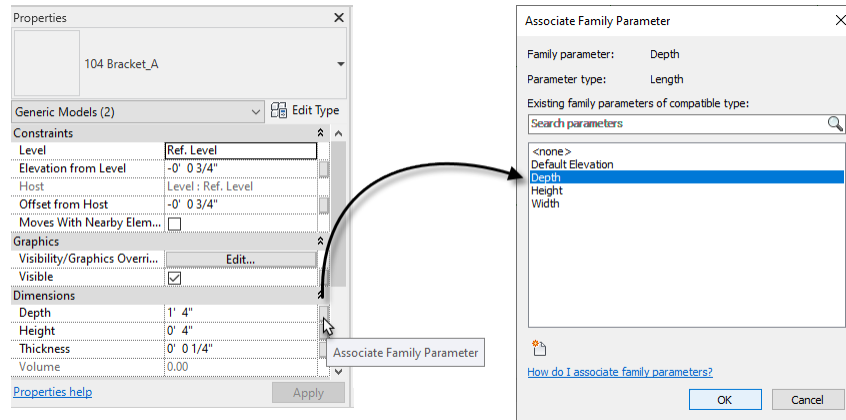


Figure 26

3. In the “Associate Family Parameter” dialog, choose **Depth** and then click OK (see Figure 26).
4. Flex the Depth.

Notice how the Depth parameter of the host family now also flexes the nested components.

5. Save the Family.

Family Editor 105

Creating Family Types

When you get tired of manually editing the values of each field when you flex, it’s time to create some family types.

1. Continue with your Shelf family. Save it as: ***Shelf w Brackets.rfa***.

CATCH UP! You can open the file completed to this point named: **105 Shelf w Brackets_A.rfa**.

2. Open the Family Types dialog.
3. At the top of the “Family Types” dialog, on the right, click the New icon. Input: **48" x 16"** for the name and then click OK (see Figure 27).

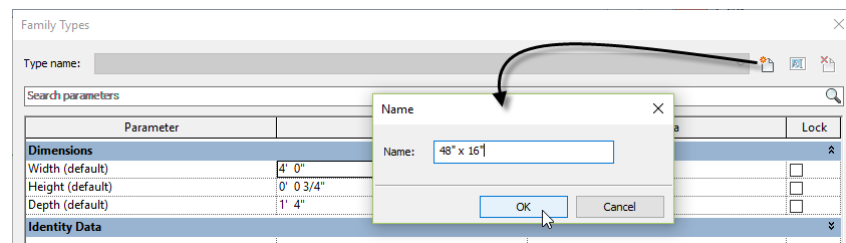


Figure 27

4. Click new again, input: **54" x 20"** and then click OK.
5. Change the Width to: **4'-6"** and the Depth to: **1'-8"** and then click Apply.

6. From the Name list at the top, choose: **48" x 16"** and then click Apply.

Notice how having types makes it easy to flex! Feel free to add more types if you wish.

7. Save the family.

CATCH UP! You can open the file completed to this point named: **105 Shelf w Brackets_B.rfa**.

Family Editor 201

Assemble a complex Family

The shelf we have could be loaded into our sandbox project and aligned with any of the walls and copied around as needed. But to make our family even more useful and powerful, we can nest it further into a new host family. In this segment, we'll build a shelving unit that repeats multiple shelves and exhibits several other useful features as well. For this we have a starter file.

1. Open the Family named: **201 Shelving Unit_A.rfa**. Save it as: **Shelving Unit.rfa**.

This family already has some reference planes, dimensions, parameters and a simple extrusion in it. Apart from a few extra reference planes and a couple extra dimensions, the setup in this file is very similar to our simple box family above. In fact, in the center we have a long thin box constrained to the reference planes. This box is the back wall of our shelving unit.

Do note however the difference in the way that the Depth parameter is applied. The shelving unit we will be building will have the option of being one sided or two sided. Applying the Depth parameter on each side facilitates this in a way that would be difficult with an equality dimension. But this simply illustrates how there is always more than one way to set things up.

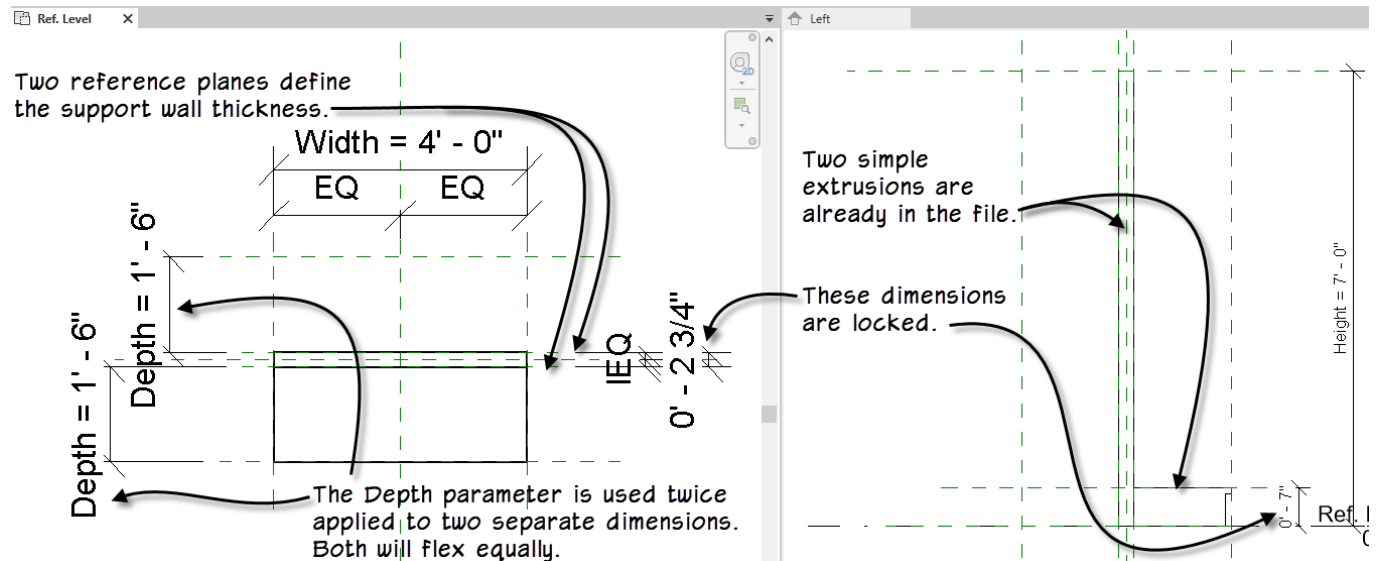


Figure 28

2. Directly above the Family Types button click the **Family Category and Parameters** button (see Figure 29).

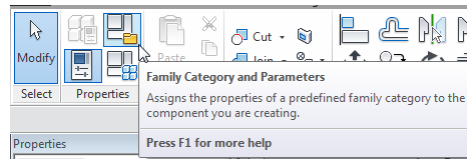


Figure 29

Notice that this family is set to the **Furniture** category. In the previous exercises, we used the **Generic Model** category. In general, it is preferable to choose a more specific category wherever possible. MasterSpec lists Retail Furniture in division 12: Furnishings and Storage Assemblies in division 10: Specialties. Translated to Revit categories, this gives us a choice between Furniture (assigned here) or Specialty Equipment. The choice should be considered carefully when devising your family content. (And there are several other issues that influence the decision which time and space will not permit us to delve into here). While possible to change your category later, it can cause issues if you change it too far into the development process for your family. So, try to choose early if possible.

3. Close the “Family Category and Parameters” dialog without changing anything.

Nest in and Position the Shelf

Load in your shelf w brackets and position it using the Align tool as we did above.

1. If you still have your shelf with brackets family open, you can simply click on one of its view tabs to switch to it. If you need to, you can open the provided file instead.

CATCH UP! You can open the file completed to this point named: **105 Shelf w Brackets_B.rfa**.

If you open the catch-up file, notice that all four of its view tabs will open in whichever tile was active at the time you opened. So, you will likely have five tabs in one tile and one tab in the other. The tabs only show the view names which can be confusing. To distinguish them from each other, hover your mouse over a tab name to see both the name of the view and the file name.

2. Following the steps above, set the Back reference plane as the Origin and Save the Family.
3. On the ribbon, click the: **Load into Project and Close** button. In the dialog that appears, check the *Shelving_Unit* family only (not the Sandbox) and then click OK.
4. If you are prompted to save the file or to overwrite the existing, choose Yes.
5. Place an instance of the Shelf in the plan view in the lower portion of the screen. (Do not try to line it up yet).
6. Using the Align tool, align the center of the shelf to the center of the host family. Lock it (see the left side of Figure 30).
7. Align the back of the Shelf w Brackets to the reference plane named **Single-Sided Shelf Plane**. Lock it as well (see the middle of Figure 30).

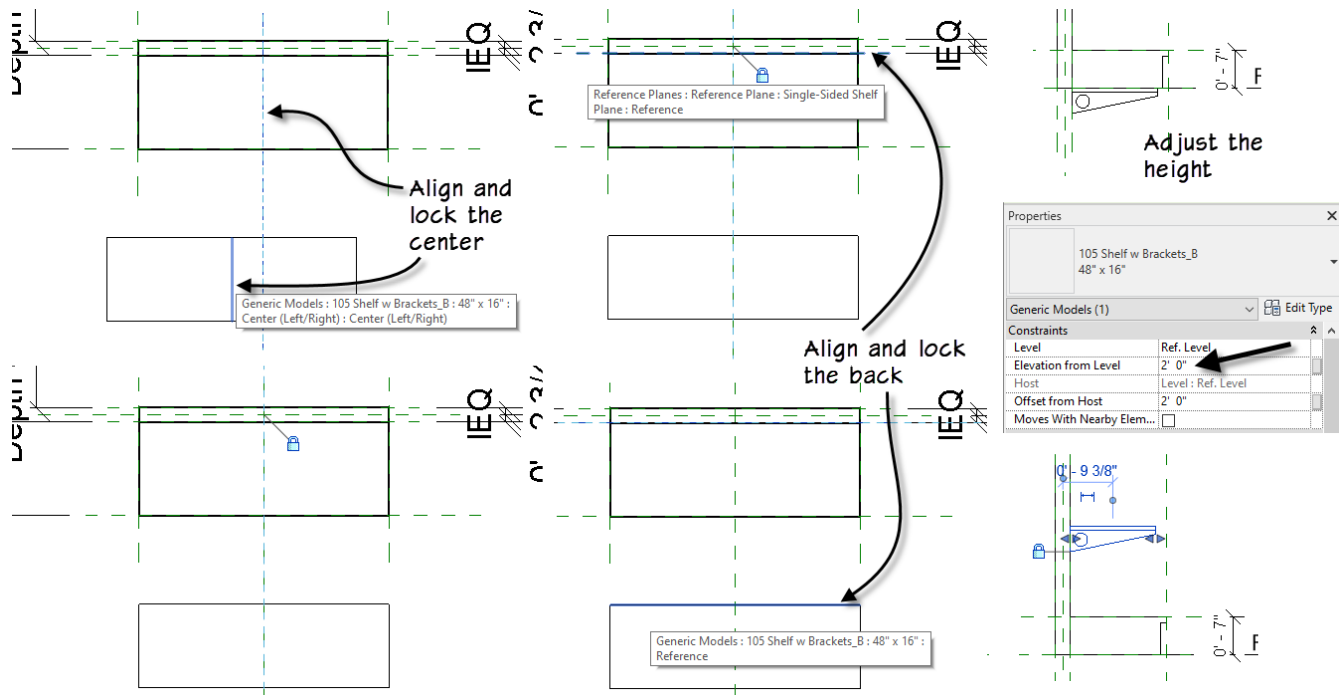


Figure 30

8. Cancel the Align command and then if necessary, Tile the windows (press WT).
9. In the *Left* view, select the Shelf. On the Properties palette, change the Elevation from Level to: **2'-0"** (see the right side of Figure 30).

This makes it appear in plan. Previously it was concealed beneath the other extrusion in the file.

CATCH UP! You can open the file completed to this point named: **201 Shelving Unit_B.rfa**.

Create a Parametric Array

When you create an Array in Revit, it does not have to be just a one-time copy command. You can leave it parametric! This means we can adjust quantity or spacing later and update the geometry accordingly.

1. In the *Left* view, select the Shelf. On the Modify | Generic Models tab, click the **Array** button.
2. On the Options Bar, verify that the **linear** icon and that the **Group and Associate** checkbox are both selected. Change the Number to **4**, choose the **Last** option and check **Constrain** (See Figure 31).

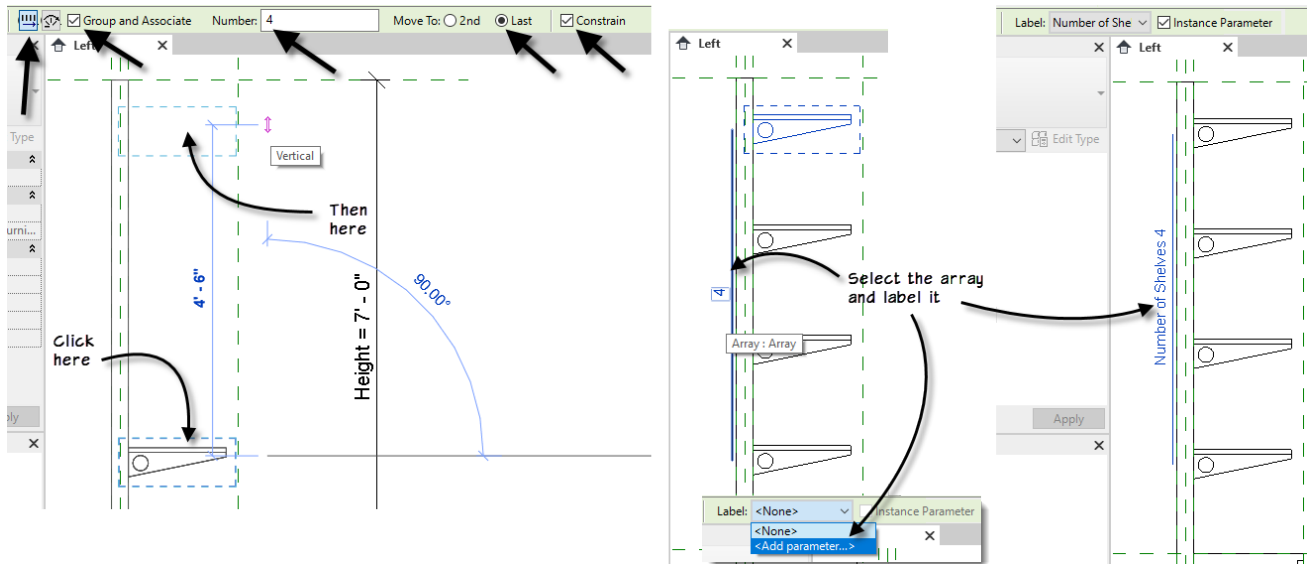


Figure 31

3. Click a start point near the shelf, drag up and click again about: **4'-6"** away.

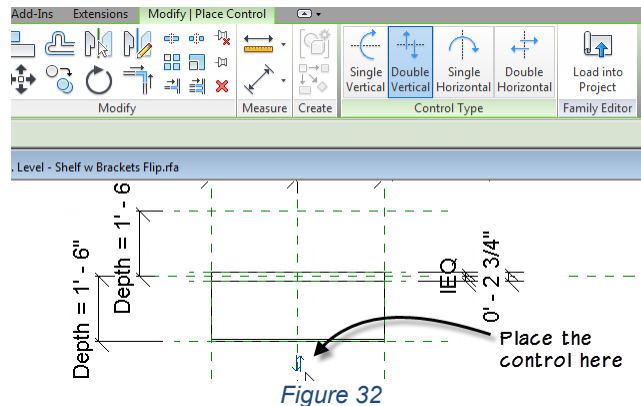
You can try different numbers for the array value to test it if you wish. Let's now assign a parameter to the quantity.

4. Select the Array dimension. (It will be a line near the quantity number onscreen).
5. On the Options Bar, click on the Label dropdown and choose: **<Add parameter>**. For the Name type: **Number of Shelves**, make it an **Instance** property, group it under **Graphics** and then click OK.
6. Open Family Types and flex.
7. Save the family as: **Shelving Unit**.

Add a Flip Control

A Flip control provides a quick and easy way to flip or rotate a family during or after placement. Let's add one to our Shelving Unit family.

1. Continue in your Shelving Unit family and make sure the plan view is active.
2. On the Create tab, click the Control button.
3. On the Modify | Place Control tab, click the **Double Vertical** button and then click beneath the plan geometry to place the control (see Figure 32).



4. On the ribbon, click the Modify tool and then Save the family.

Make sure to click the Modify tool, or you can end up creating multiple controls.

CATCH UP! You can open the file completed to this point named: **201 Shelving Unit_C.rfa**.

Load into the Sandbox

Let's test the family out in our Sandbox project.

1. On the ribbon, click the Load into Project button. Load it into the *Sandbox* file.

We are using Load into Project and *not Load into Project and Close* this time. We want to leave the Shelving Unit family open as we test it in the Sandbox.

2. Place an instance in the file.
3. Switch to plan view and try out the flip control. (You can also use the SPACEBAR).

The basic geometry of the family is complete. The next several topics will add more complexity and robust features to the family. We will try to get through as many as the lab time will permit. Feel free to use the steps outlined here to continue refining this collection of families after the lab on your own.

Family Editor 301 – Bonus 1

One-sided or Two-sided?

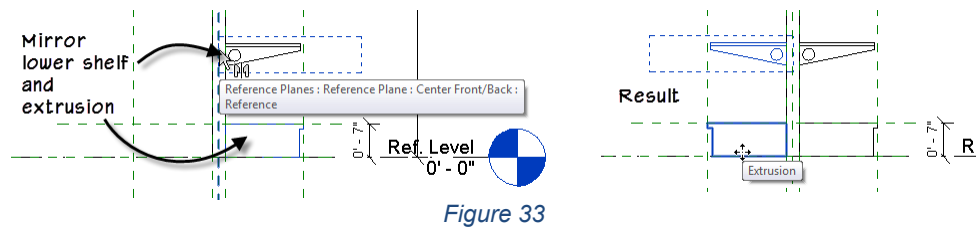
We have many ways to make this family more robust. Let's start by giving it the option to be one or two-sided.

1. Continue in your *Shelving Unit* Family. Close other families if any are open. Leave the *Sandbox* project file open and minimized.

CATCH UP! You can open the file completed to this point named: **201 Shelving Unit_C.rfa**.

2. In the *Left* elevation view, select the base shelf extrusion and the lowermost shelf in the array.

- On the Modify | Multi-Select tab, click the **Mirror - Pick Axis** tool (or type MM). Pick the Center Front/Back reference plane as the axis of reflection (see Figure 33).



When you array, it creates a group. Changes made to one instance of a group apply to all instances when the change is complete. This is a powerful feature for managing changes in both projects and families alike. Copying grouped elements do not keep them associated with the original array but will leave them grouped. So, if you made a change to the array (quantity, spacing, etc.) the four group instances on the right would react, but not the copied one on the left. However, if you perform an edit group on any one of them (left or right) the change would apply to all instances both inside and outside of the array. Furthermore, if you create a new array from an existing group it will simply use that existing group rather than create a new one. This means that if you want the shelves on both sides of the wall to be the same, you can make two arrays that use the same group. However, since they will be two separate arrays, each will be able to have their own quantity and spacing independently. Optionally, you can ungroup the copied shelf and when you array it, it would create a new group that is not tied to the first. Do this if your design calls for different shelf configurations on the two sides of the gondola unit. This is what we need in our case. This is because we will make the visibility of the shelves on one side of the wall optional. This will allow the same family to contain two options: one-sided and two-sided. But to do this, we must ensure that our new array uses its own group name.

- Select the new shelf w brackets created by the mirror (Only the mirrored one). It will still be grouped. On the ribbon, click the **Ungroup** button.

Be sure to ungroup **ONLY** the mirrored shelf. Not the ones on the other side of the wall. If you ungroup the others, you will break the array.

Create and Apply a Visibility Parameter

The items that we mirrored we want to display only some of the time. To do this, we can make a parameter to control element Visibility.

- Select the two mirrored elements (one extrusion and the ungrouped shelf w brackets).
- On the Properties palette, click the small Associate Family Parameter button to the right of the Visible checkbox. (Do not click the checkbox, click the small button next to it).
- In the “Associate Family Parameter” dialog that appears, click the **Add Parameter** button at the bottom.
- In the “Parameter Properties” dialog, name it: **Gondola Unit**, group it under: **Graphics**.
- Near the bottom of the dialog, click the Edit Tooltip button.

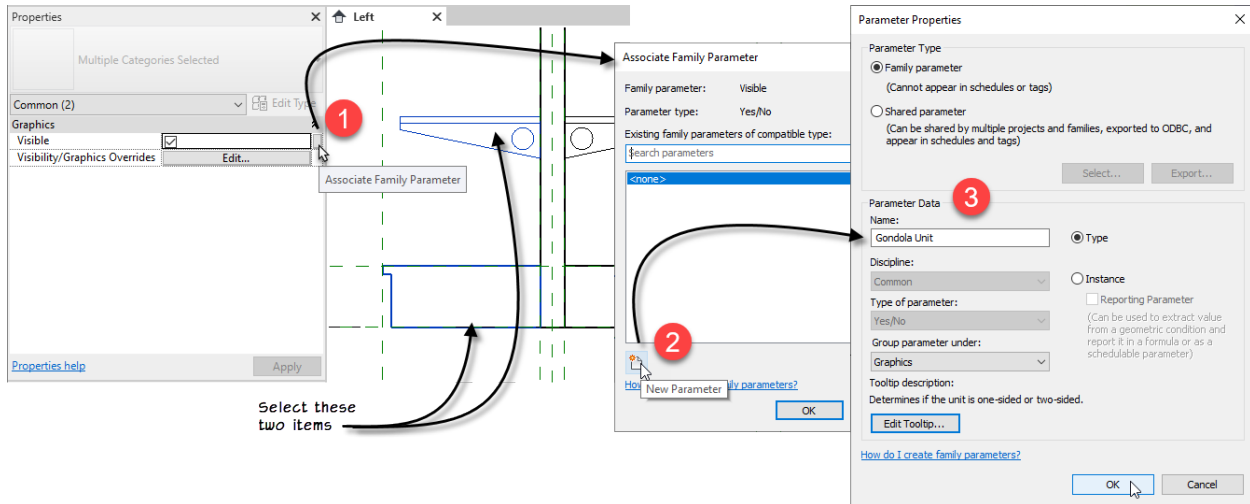


Figure 34

6. Select all the text in the “Edit Tooltip” dialog and then type: “Determines if the unit is one-sided or two-sided.”

Leave this parameter set to “Type” instead of Instance.

7. Click OK three times.

A small equals (=) sign will appear on the button to indicate that it is now driven by a parameter (see Figure 35).

8. Select just the ungrouped shelf. Repeat the process above to **array** it using the same settings as before.
9. Label the array with the same **Number of Shelves** parameter (see Figure 36).

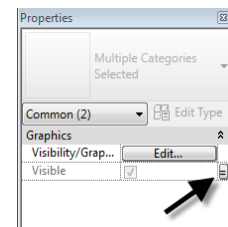


Figure 35

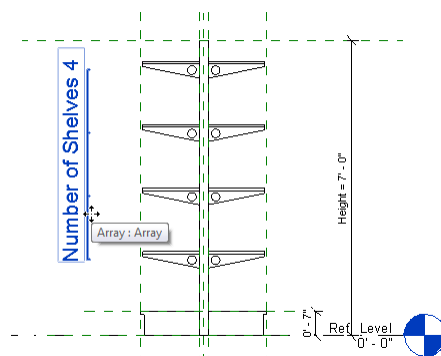


Figure 36

Optionally you could make a new parameter. Do this if you wish to control the quantity of shelves on each side separately.

CATCH UP! You can open the file completed to this point named: **301 Shelving Unit_A.rfa**.

Create Family Types

To test our new parameters, let's create a few family types. Last time, our family types were changing instance parameters, so they were not "true" types but rather a convenient way to flex the family. (When types are assigned to instance parameters, they only set the default values. Users can change those values to anything they like in a project without affecting the other instances). This time, we will be changing type parameters. This means that changes made in the future will affect all instances of the type.

1. Open the Family Types dialog.
2. Click the New icon. Name the new type: **Wall Unit** and then click OK.
3. Uncheck the Gondola Unit checkbox and then click Apply.
4. Click New again. Name it: **Gondola Unit**. Check the Gondola Unit checkbox and then click OK.

You will not see the effect of the visibility checkbox initially. To see it in the family editor, you can use the Preview Visibility mode.

5. On the View Control Bar, click the Preview Visibility pop-up and choose: **Preview Visibility On**.
6. Open Family Types and then choose one of the types and click Apply.
7. Choose the other type and then click Apply (see Figure 37).

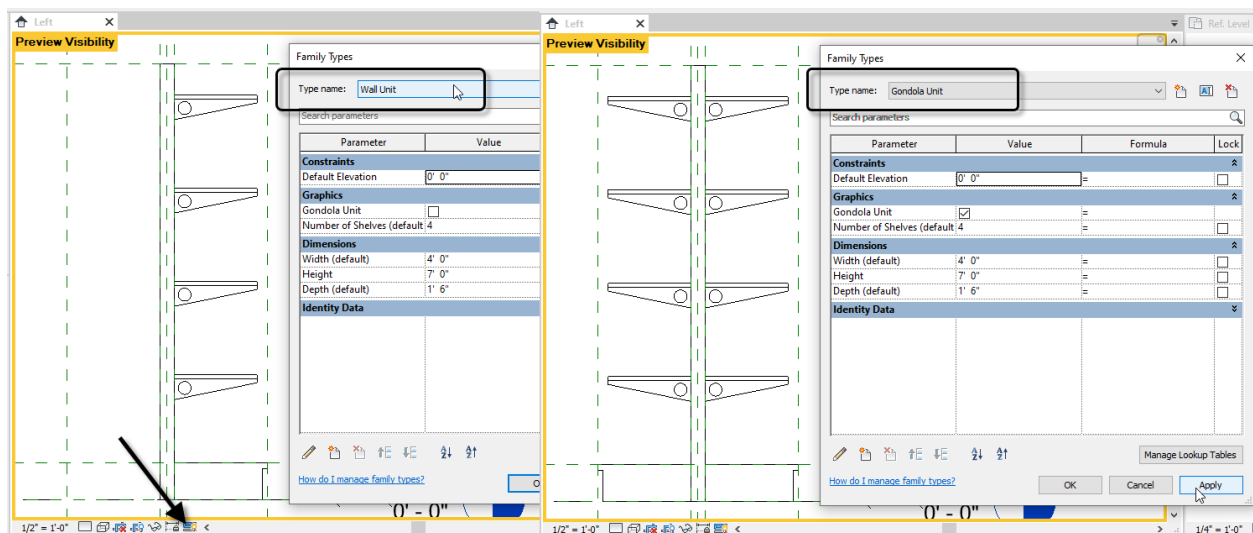


Figure 37

The other way to preview the family is to load it into the project. If you have been continuing in your own file throughout, you may have an extra type on the Type Selector with the same name as your family. This was created automatically by Revit when we loaded the family earlier to test

the flip control. If a family has no types (as was the case before we added them just now) Revit creates a type for you with the same name as the family. You can expand the *Families* branch of the Project Browser, expand *Furniture*, locate and delete this extra type by right-clicking on it.

Family Editor 302 – Bonus 2

Tweaking the Graphics

The Visibility parameter immediately introduces a great deal of flexibility to the family. But we have only scratched the surface. In this segment we'll look at some ways to refine the graphics.

1. Remain in the *Sandbox* project file. Open the *Level 1* plan view (see Figure 38).

Notice the double line at the front edge of the shelf. One of these lines is the edge of the base shelf and the other is the upper shelves. In small scale plans, you may not wish to see both lines. We can adjust the elements Revit displays in several ways.

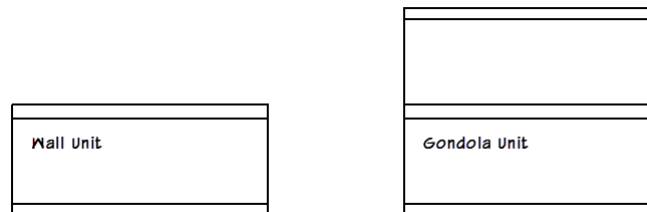


Figure 38

2. Close the floor plan view tab, minimize the 3D view and switch back over to the *Shelving Unit* Family file. (Or open the catch-up file).

CATCH UP! You can open the file completed to this point named: **301 Shelving Unit_B.rfa**.

Adjust Visibility

The visibility settings of each element in your family can be customized to determine when they display. Conditions available include Plan/RCP, elevation views and Course, Medium and Fine detail levels. We'll just do a simple example here. Feel free to experiment further.

1. Select one of the shelves. (It will highlight with a dashed box around it indicating that it is part of a Group – the array group in this case). You can select any instance of the shelf.
2. On the ribbon, click the **Edit Group** button.
3. In Edit Group mode, select the nested shelf family instance onscreen. On the Modify | Generic Models tab, click the **Visibility Settings** button (see Figure 39).

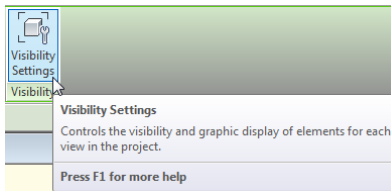


Figure 39

4. In the “Family Element Visibility Settings” dialog, uncheck the Plan/RCP box and then click OK (see Figure 40).

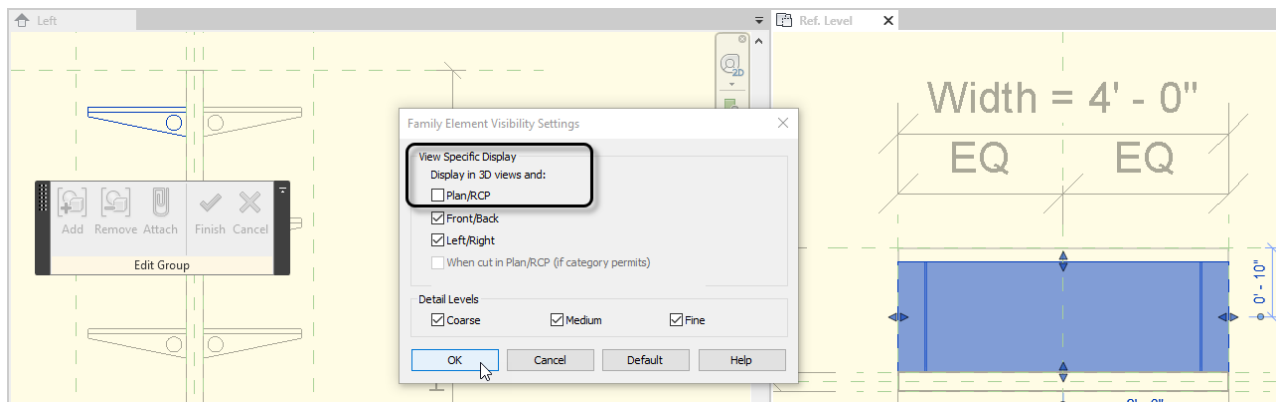


Figure 40

5. Click the Finish button on the Edit Group panel.
6. Repeat for the Group on the other side.

If you watch for the names of groups as you select them, you will see that the first array is called: Array Group 1 and the other is called: Array Group 2. This is why we have to do the edit twice. However, you do not need to do every shelf. Edits to a group apply to all instances of the group.

7. Save the family and reload it into the project to test it. When prompted, choose Overwrite Existing (see Figure 41).

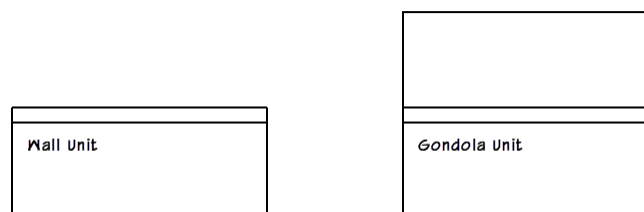


Figure 41

Family Editor 303 – Bonus 3

Adding Materials

Materials can be assigned to the elements in a family just like elements in a project to give them more realism. The family editor usually includes only a few materials however, so it is often best to use a Material parameter as a placeholder and then assign the actual material once the family is loaded into the project. This also makes the family more flexible and reduces its size.

1. Remain in the *Shelving Unit* Family file.

CATCH UP! You can open the file completed to this point named: **302 Shelving Unit_A.rfa**.

2. On the Project Browser, expand *Families* and then *Generic Models*. Right-click on *Shelf w Brackets* and choose: **Edit**.
3. Select the shelf extrusion solid form. On the Properties palette, click the small Associate Family Parameter button in the right column next to Material.
4. In the “Associate Family Parameter” dialog, click Add Parameter. Name it: **Shelf Material** accept the other defaults and click OK twice.

For this example, we will only edit the shelf material. See the note below if you wish to also edit the material of the brackets.

5. Click the **Load into Project and Close** button.
6. Choose the *Shelving Unit* family only (not the sandbox). (Saving the file when prompted is optional).
7. Overwrite the existing version when prompted.

We must remember that the *Shelf w Brackets* family is nested inside the *Shelving Unit* family. In order to access this new Material parameter, we must create a matching parameter in the host family and make it drive the nested one.

8. On the Project Browser, expand *Shelf w Brackets* to reveal the two types. Right-click the *48"x16"* type and choose: **Type Properties**.
9. In the “Type Properties” dialog, click the small Associate Family Parameter button to the right of Shelf Material and repeat the steps above to create a new parameter in the host file. You can name it: **Shelf Material** again (See Figure 42).

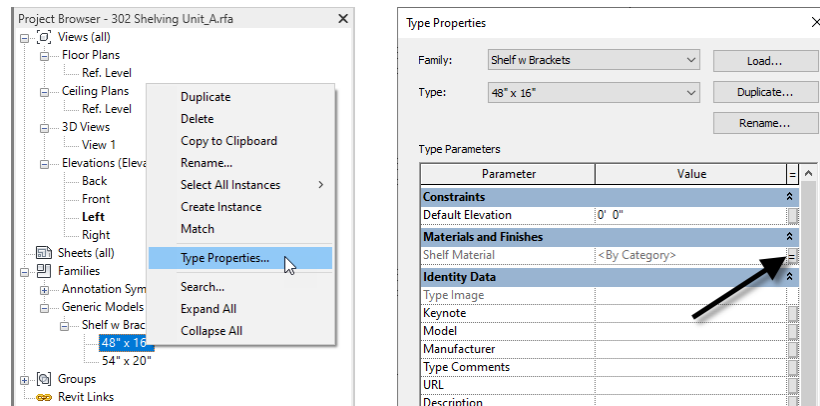


Figure 42

When finished, the equal (=) sign will appear to indicate the connection.

10. Edit the properties of the other type. Link up the Shelf Material parameter you just created to this one as well. (You do not need to make a new parameter this time).
11. Save the family, load it into the *Sandbox* and overwrite the original.
12. Open a 3D view, select the Shelving Unit and on the Properties palette, click Edit Type. You can now choose a material for the shelves.

To assign materials to other parts of the family like the support wall, the base shelf or the brackets, you must repeat the process. For the Brackets, you will need to edit the nested bracket family select the blend within that nested family and repeat the steps here to add the material parameter. However, you will need to link it to host parameters in both the **Shelf w Brackets** family and the **Shelving Unit** family. Go ahead and performs these steps if you like.

A version of the Family with Materials assigned to the brackets too is included with the lab dataset: **303 Shelving Unit_A.rfa**.

Family Editor 401 – Bonus 4

Formulas

One of the most powerful features of the family editor is the ability to use mathematical formulas to drive parameters in the Family Types dialog. Formulas can be simple arithmetic expressions that create a direct relationship between one or more parameters or complex conditional and trigonometry statements that trap errors and create robust relationships. In many ways, using formulas in the family editor can be considered the “deep end of the pool.” Therefore, in keeping with the course’s goal of being an introduction to the family editor, I am presenting a few straightforward examples here just to give you some idea of the possibilities.

Covering the topic of formulas in-depth would require an entire class or likely several classes to do it justice. (If you would like to learn more, you can explore my video training courses at LinkedIn Learning. There is: [Revit Architecture: Family Editor \(Imperial and Metric\)](#) course for an introduction through intermediate. To get a little more advanced, try: [Revit: Parametric Curvature in the Family Editor](#). Or for a really deep dive, get a copy of [Renaissance Revit](#).

1. Keep the **Sandbox** file open and minimized and close other files.
2. To try some of these ideas, a version of the original Box family has been provided. It is called: **401 Box_A.rfa**. Open that file now.

Simple Arithmetic

To use a formula in a family, you need to know the proper syntax and where to input it. The syntax is covered adequately in the help. Simply search for “[formulas](#)” and you will find the information required. To input a formula, open the Family Types dialog and type the formula into the field at the right of the parameter.

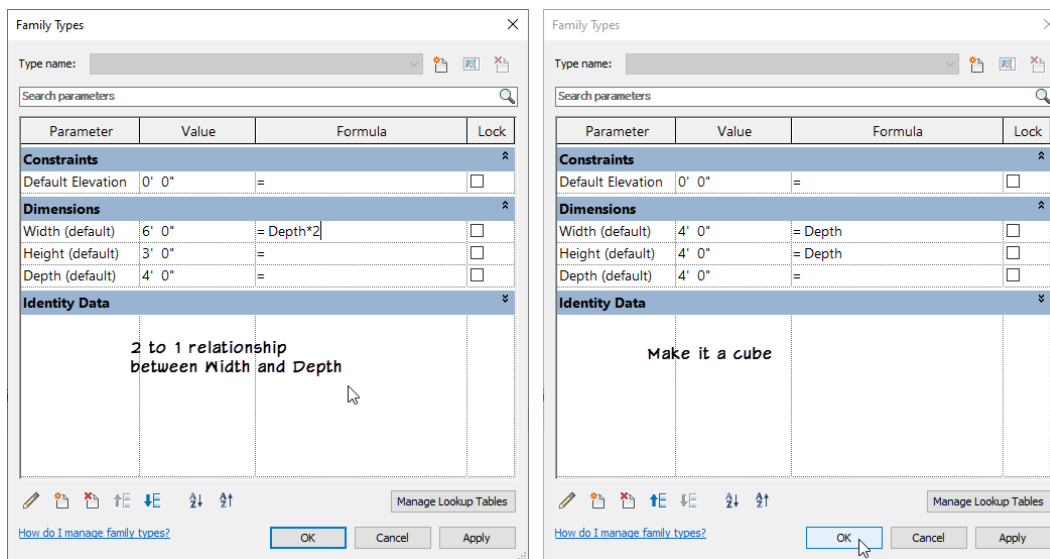


Figure 43

Arithmetic formulas include simple addition, subtraction, multiplication and division.

3. In the “Family Types” dialog, in the Formula column next to Width, type: **Depth * 2** (Formulas are case sensitive, so be sure to type: **Depth** and not: **depth** – see Figure 43).

A formula like this will work both ways. For example, if you type: **4'-0"** for the Width, the Depth becomes: **2'-0"**. If you type: **5'-0"** for the Depth, then the Width becomes: **10'-0"**. Try it out.

You can try other options as well. The figure shows an example to make a cube. You can use addition (+), subtraction (-), multiplication (*) or division (/). Try others.

Medium Complexity Expressions

More robust formulas are also possible. Let's do a couple more.

1. Close the box and open the file named *401 Shelving Unit_A.rfa*.

This is a copy of our **Shelving Unit** family with a few parameters added. **Bottom Shelf Height** sets the height of the first shelf in the array and is measured from the floor Level. **Top Shelf Offset** sets the distance below the Top reference plane for the topmost shelf. These give a little more control over the location of the shelving array both at the top and bottom (see Figure 44).

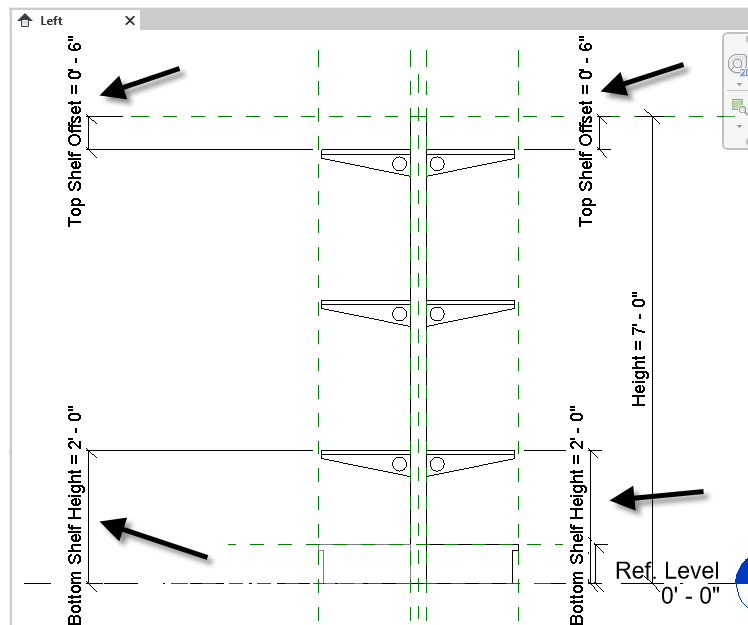


Figure 44

Using formulas, we can set the quantity of shelves in a few ways. One very common approach is to create a conditional statement that prevents the array from failing if someone inputs a bad value. In the case of a linear array, we need at least two items, so a bad value is less than: 2. To create such a formula, open "Family Types," and create a new parameter of the type: **Integer**. Make it instance based to match the Number of Shelves parameter which it will check. Name it something like: **Array Check**. In the formula column input:

if (Number of Shelves < 2, 2, Number of Shelves)

This is how it works. There are three parts; the "if" condition which in this case tests if the **Number of Shelves** parameter is less than 2. Next comes the value to be used if the test returns true, and the last part is the value to use if the test returns false. Read it like this and it makes more sense: "if the Number of Shelves is less than 2, then use a value of 2. Otherwise just use the Number of Shelves as is." Go ahead and input this formula and flex if you like (see Figure 45).

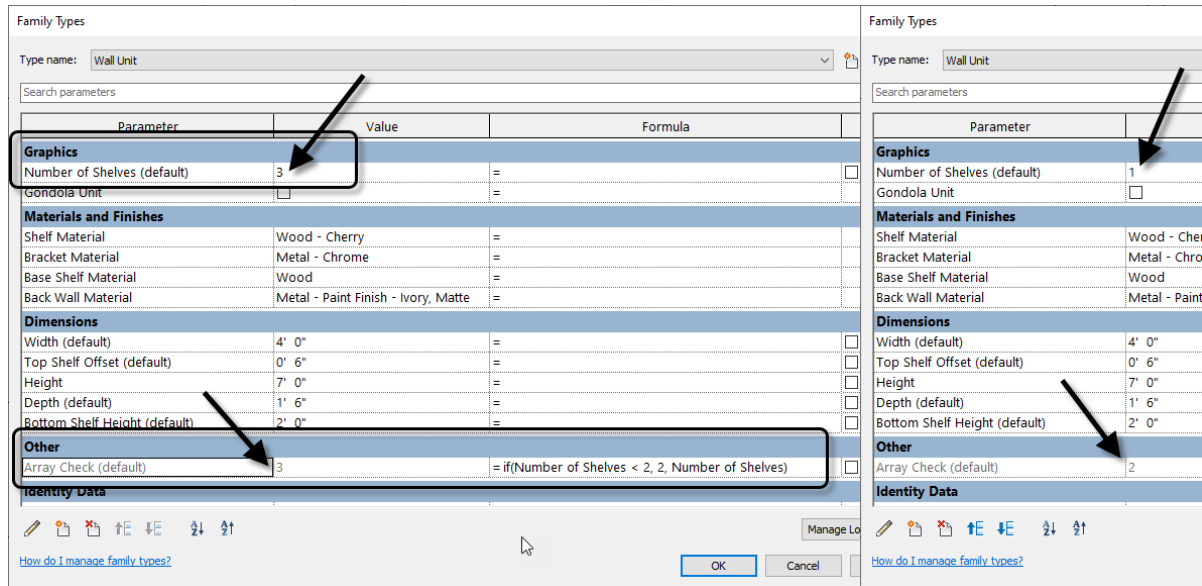


Figure 45

To use this new parameter, label the array dimensions in the family with this one in place of Number of Shelves.

CATCH UP! You can open the file completed to this point named: **401 Shelving Unit_B.rfa**.

There is another way to ensure not only valid array input, but end up with a logical quantity of shelves. We can use math.

2. Undo the previous, or simply delete the Array Check parameter and re-label the dimensions with **Number of Shelves** instead.
3. Open “Family Types” and in the formula field next to Number of Shelves, input: **roundup((Height - Bottom Shelf Height - Top Shelf Offset) / 1' 6")** (see Figure 46).

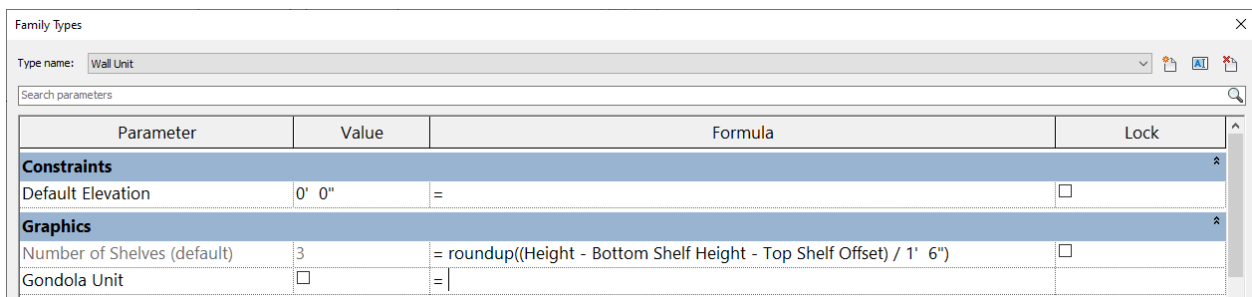


Figure 46

You will recall from High School math, that in mathematical formulas, parenthesis keep things organized and tell you which part of the formula to process first (order of operations I believe they called it...) So, let's start with the innermost parenthesis:

Height - Bottom Shelf Height - Top Shelf Offset

This is just more arithmetic. Simply subtract both **Bottom Shelf Height** and **Top Shelf Offset** from the **Height** parameter (remember that the parameter names are case sensitive). Assuming a 7' Height, 2' Bottom Shelf Height and 6" Top Shelf Offset, that is $7 - 2 - .5$ or: **4'-6"**.

Next, this result is divided by the distance we want between shelves or: 1'-6" in this example. You can make this distance anything you like (even another parameter if you wish). Now, since 4'-6" divided by 1'-6" is exactly 3, it works out very nicely. But what if you change any of the dimensions. For example, make the Top Shelf Offset: 1' instead of 6"? In that case, because the **Number of Shelves** parameter is an Integer, you would end up with a value of 2. So, the Roundup function effectively makes the 1'-6" value in the formula a "not to exceed" value.

SAMPLE FILE! You can open a file that includes the following ideas named: **401 Shelving Unit_C.rfa**.

Family Editor 402 – Bonus 5

Global Parameters

At this point it would be possible to create another family and nest the overall shelving unit into it and make parameters to control the sizes and quantities of the units. As an alternative, we can do some of the same things directly in the project with global parameters!

1. Load the final shelving unit **401 Shelving Unit_C.rfa** into the project and place an instance.
2. Select the shelving unit and then on the Properties palette, click the small Associate Global Parameter icon next to the Width parameter.
3. Click the Add Parameter button and create a new parameter called: **Shelf Width** (see Figure 47).

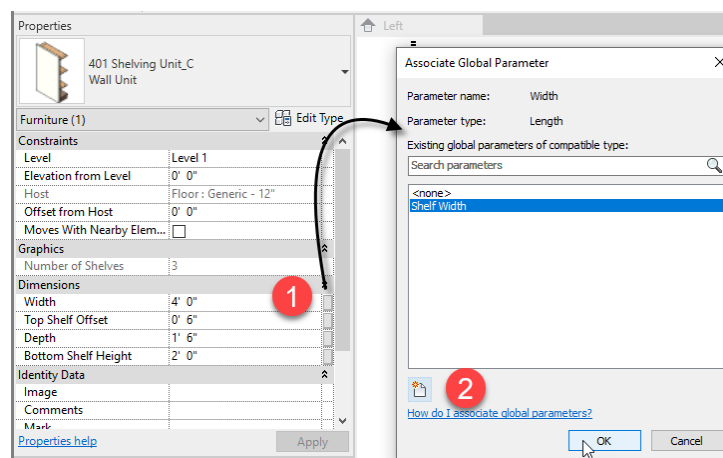


Figure 47

4. Array the shelving unit. Be sure Group and Associate it selected and keep the quantity at: 2.
5. Add a dimension between the two resulting array group elements.

6. Select the dimension and just like in the family editor, the Label dropdown will appear on the ribbon. Click the Create Parameter icon.
7. Name the parameter: **Distance Between Units** and then click OK (see Figure 48).

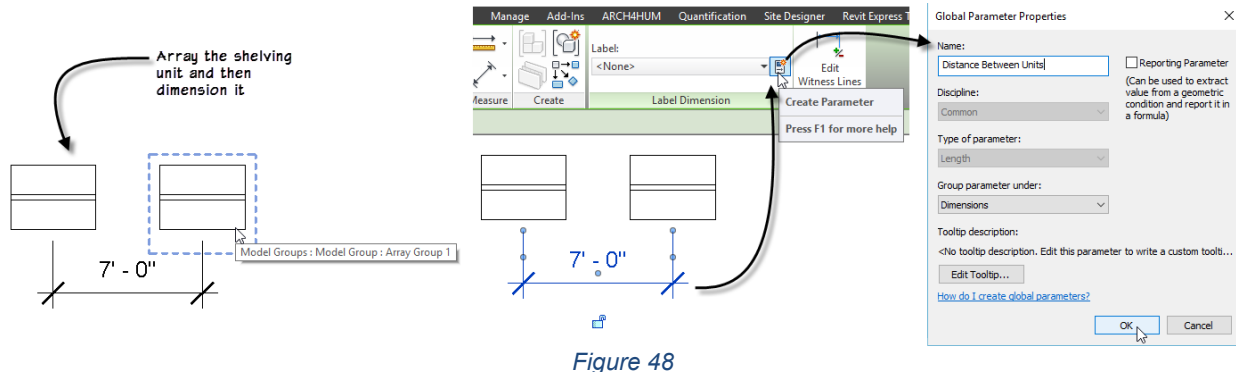


Figure 48

Now we can use the Global Parameters dialog to link these two global parameters together.

8. On the Manage tab of the ribbon, click the Global Parameters button.
9. In the Formula field next to Distance Between Units, type: **Shelf Width** (Case sensitive).

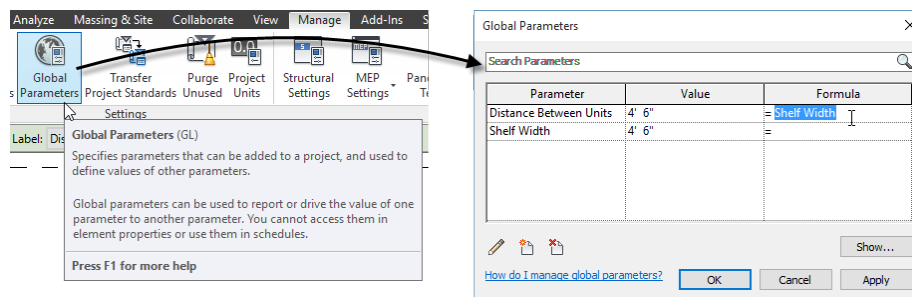


Figure 49

10. Click OK to finish (see Figure 49).

Your shelving units should now be flush to one another. Furthermore, you can now flex the Shelf Width (in the "Global Parameters" dialog) or adjust the quantity of arrayed elements (directly onscreen) and the shelving units will all flex together and remain flush to one another. This is just a simple global parameters example. But hopefully it is enough to give you some idea of the potential of this powerful feature.

If you want to see more global parameters examples, you can visit my website (<https://paulaubin.com/conferences/>) and download the handout from the session called: "Global Parameters, Global Control: Revit Global Parameters in Practice (AS121828)." I taught this session a few years ago, but most of the material in that paper should still be relevant to current releases of Revit.

Family Editor 501

What now?

Well, that is all we have time for. (Probably more than we have time for). There are so many ways to enhance even this simple family. Consider adding more parameters and formulas to control shelf placement, offsets, add end panels, etc. You can add more geometry to refine the design add moldings, different kinds of shelves, etc. You can add more family types with additional variations. The sky's the limit. Have fun with it and before you know it, you'll be an expert family author!

Further Study

You can find more information and tutorials in:



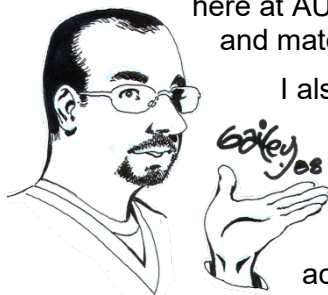
[Renaissance Revit: Creating Classical Architecture with Modern Software](#). This book can be thought of as a “deep dive” into the family editor. It starts with the basics but gets very advanced as well. The entire book is on family creation using classical architectural examples. Both the traditional and massing family editors are covered.

[The Aubin Academy Revit Architecture: 2016 and beyond](#). Chapter 11 is devoted to the subject of the family editor. The book is authored in an earlier release, but most functions are still relevant in current releases.



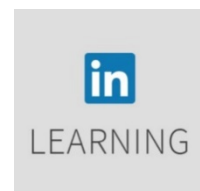
The Aubin Academy Master Series: Revit MEP. Chapters 12 and 13 are devoted to the subject of the family editor.

Other Autodesk University courses: I have taught this family editor lab before in previous years here at AU. I have also taught an advanced follow-up lab. Both classes have papers and materials available for download from my website: www.paulaubin.com/au



I also have Revit video training available at [LinkedIn Learning](#).

Lynda.com has recently been rebranded as LinkedIn Learning. If you are a Lynda.com subscriber, you can access all the offerings on that platform as well. If you are Professional member of LinkedIn, you already have access to the entire training library.



If you have any questions about this session or Revit in general, you can use the contact form at www.paulaubin.com to send me an email.

Follow me on twitter: [@paulfaubin](https://twitter.com/paulfaubin)