AUTODESK **UNIVERSITY** 2015

AS10641-L

Revit Families: A Step-by-Step Introduction

Paul F. Aubin
www.paulaubin.com

Learning Objectives

- Learn how to use reference planes, constraints, and parameters
- Learn how to create a simple, fully parametric model family
- 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 Revit software's Family Editor but you've avoided it or it has intimidated you until now. Wait no more. This hands-on lab will introduce you to the basics of Family Editor. Once you overcome your nerves, you'll find that while Family Editor is extremely powerful, it can actually be great fun. In this short introduction to Family Editor, you will 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 the Family Editor, this lab will teach you the basics of the editor in the most effective way possible—hands on!

Note: The session uses architectural examples, but the concepts apply to the flavors of Revit software, including Revit LT software.

Your AU Expert

Paul F. Aubin is the author of many CAD and Building Information Modeling (BIM) book titles, including the widely acclaimed Renaissance Revit, the Aubin Academy titles, and dozens of Revit software video training titles for lynda.com. Paul is an independent architectural consultant who travels internationally to provide Revit software implementation, training, content creation, and support services. Paul's involvement in the architectural profession spans over 25 years, with experience that includes design, production, CAD management, coaching, and training. He is an active member of the Autodesk, Inc., user community, he has been a top-rated speaker at Autodesk University and the Revit Technology Conference (RTC), and he has been a national speaker at the BIM workshops for many years. His diverse experience in architectural firms, as a CAD manager, and as an educator gives his writing and his classroom instruction a fresh and credible focus. Based in Chicago, Paul is an Autodesk Expert Elite, an Autodesk Certified Professional, and an associate member of the American Institute of Architects.

This part is only in the Paper

*I want to devote as much time as possible in the lab to working hands-on in the software. Therefore, this part of the paper which includes background and conceptual information is **only** being provided here in the paper. Since the papers are made available online before AU begins, you can read up on this background information before attending the lab. Therefore, in the actual lab time, I am going to dive right into the software and assume you have already read this first topic. However, if this is the first time you are seeing this and did not get a chance to read it before the lab, **don't worry**, you will do just fine in the session. I think it is good to provide you with some background information before beginning the process of creating family content, but you can begin without it. So if you are reading this before the class; read on and enjoy and come to class ready to start building. If you are reading this after the class, hopefully you find it a nice supplement to what you already learned in the session. And now on to some content building!*

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 hands-on introduction to the family editor, we'll work through the creation of a Revit component (loadable) 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. So whether you've never worked in Revit before, or have used Revit for a while but simply avoided the family editor, this lab will help get you acclimated to this critical and powerful aspect of using the Revit software.

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 and the lab will focus on the component family creation procedures and strategies. Using the concepts and techniques covered in this lab, you will learn how to begin tapping into one of the most powerful aspects of the Revit software package.

Everything in Revit is part of a Family

In order 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 a few 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 is the Family. 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 system families in Revit.

Families branch into two major kinds based on their behavior: the **System Family** and the **Component (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 less obvious items like floor plans, project data, 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 roofs can have more than one: **Type**. A type is our next level or 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 is basically a convenient way to switch several variables of a family at once. A family can contain one or more types; each with its own unique user-editable settings. So while for example we cannot create or delete wall families, we *can* add, delete and edit the types associated with each of the provided wall families. For example, "Basic Wall" is the most common wall family. In the out-of-the-box 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 simply means that it is a layered wall that has the same structure along its entire length and height. The actual make-up 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; 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, and levels). To distinguish further, 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 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 in the family editor.

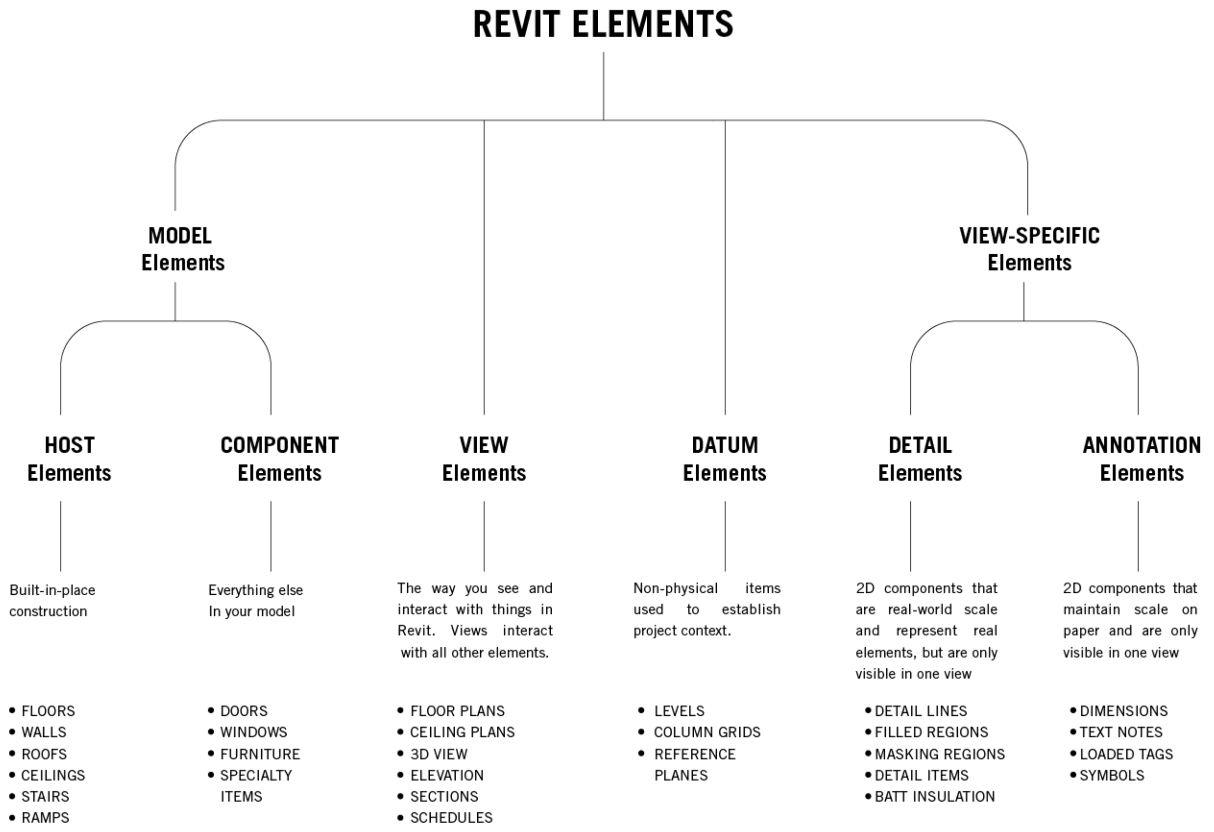
Loadable component (model) families are the primary focus of this lab.

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 (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. You should only consider creating an in-place family for elements that are unique to a particular project with little 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 component family first 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 lab.

Revit and Family terminology

Here is a brief summary of Revit Architecture critical terminology. 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 of 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 Architecture project.
(Elements in italic can be created and edited in the family editor)

Model Element—Something that represents the actual geometry of your building.

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

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

***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 drawing types like plan, section, elevation and schedule. Some View Element families allow customization of Types, many do not.

Datum Element—Include 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. The families and types of datum elements cannot be edited.

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 Architecture project. Annotation elements maintain a constant size relative to the plotting scale of the view in order to maintain a constant size relative the sheet on which they are placed.

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 this list of terms. 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 hits 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 have to 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. Also, on the Insert tab of the ribbon, on the Autodesk Seek panel, you can run a search from directly in Revit of the online Autodesk Seek website. You can also type: seek.autodesk.com into your web browser.

In many cases, a family similar to 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 by far 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.

If you are new to creating families in Revit (as I assume you are because you attending this lab), 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. If you are new to family editing and creation, it can be difficult to dissect these often complex relationships. To avoid becoming discouraged, it is recommended that you start with a small simple example and work your way to more complexity over time.

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. You can work your way up from there.

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 (yes on paper) the family you intend to create and make notes about its requirements.

Next, create a new family file from the appropriate template or open and 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 by Autodesk 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 in order 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 attach the non-hosted cabinet to a wall 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 actually 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 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 (this is our ***100 Sandbox.rvt*** file below for the lab). If necessary, make any adjustments, 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. 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.

Essentially 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, these would be parameters. 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 door would be fixed.

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 1). We will use an extrusion and a blend in the tutorials below.

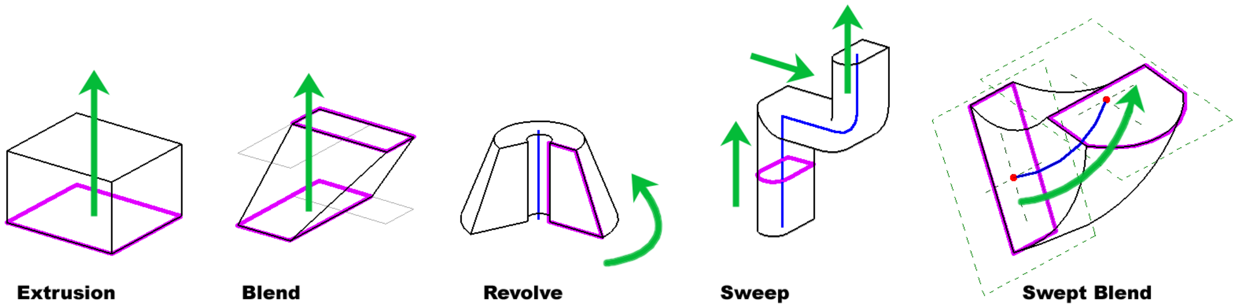


Figure 1 Forms available in the family editor

An extrusion is a sketched shape pushed along a distance perpendicular to the sketch plane. A blend is similar accept 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. Bear in mind however that spline paths can affect performance. 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 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. In some cases however, managing a complex form in a single family can become cumbersome. In some 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 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 door family where hardware was an optional display component. In this case, a visibility parameter would be assigned to the hardware elements within the door family and the visibility parameter could then be toggled on or off by the user depending on whether or not the needed to show it in a given situation.

This is the end of the introductory material. The rest of the paper contains the tutorial steps that will be conducted live in the lab. More lessons are included here however than our 75 minutes will allow. So feel free to perform the additional lessons on your own following the class.

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.

All steps and screen shots here are Revit Architecture 2016, but most 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).

The Steps outlined here are meant to supplement the live presentation given in the lab. Steps have been kept brief and much of the explanations accompanying the steps in the live lab have been omitted from the paper or included in the first section of the paper above. Please refer back 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 Application Menu (big "R"), or on the Recent Files screen, choose **Open** > **Project**.
3. Browse to the folder containing this lab's dataset files (I will have this posted up on my screen) and open the file named: ***100 Sandbox.rvt***.
4. Minimize the active view.

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*.

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*.

1. From the Application Menu (big "R"), 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).

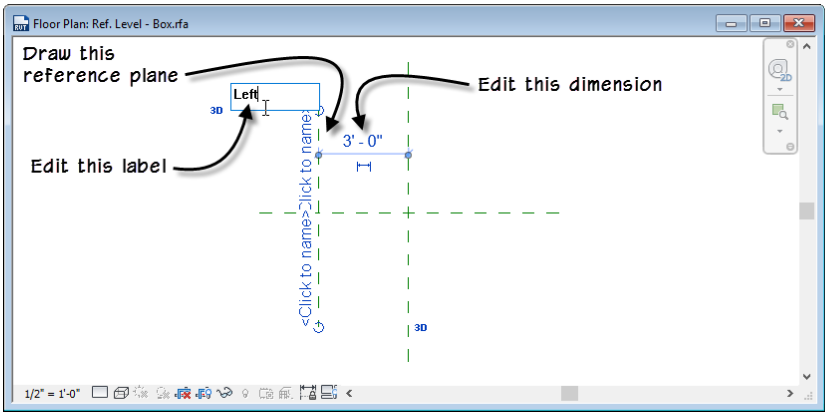
2. Select: ***Generic Model.rft*** from the list and then click Open.
3. Type WT and then ZA.
4. From the Application Menu choose **Save** (or CTRL + S) and name the file: ***Box.rfa***.

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!

Create Reference Planes

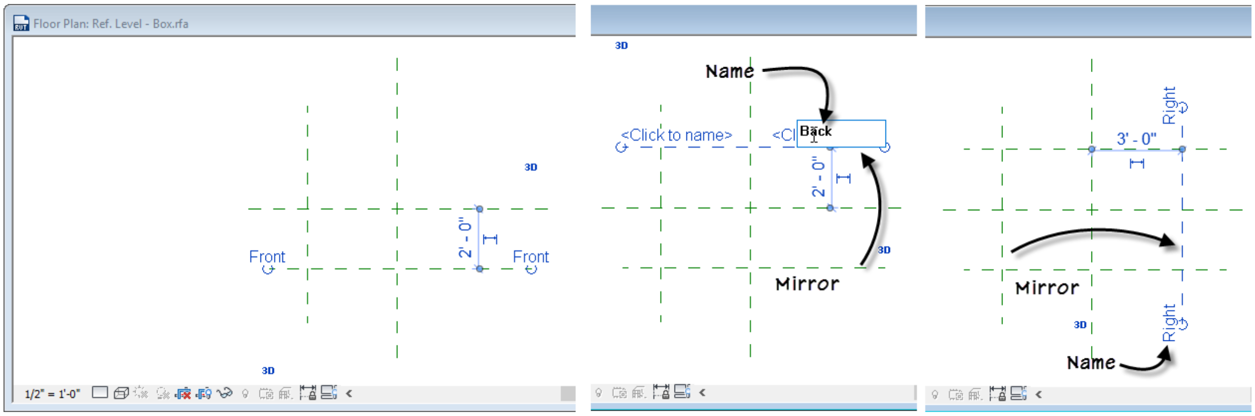
Reference planes give our family its structure. Think of it as the skeleton of your family. Always start with reference planes.

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.
3. Click in the "Click to name" onscreen control, type: **Left** and then press ENTER.

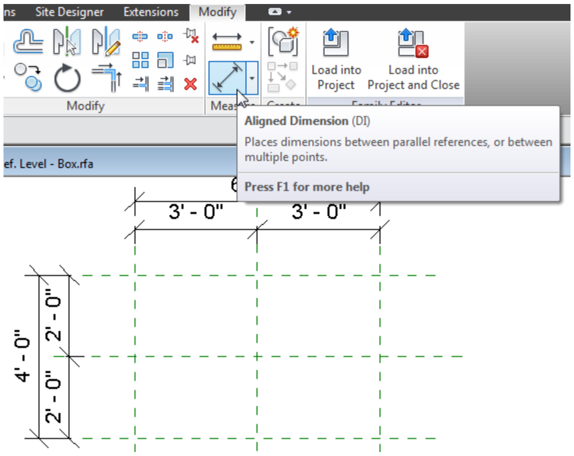


4. Repeat to create a horizontal one below at: **2'-0"** from center. Name it: **Front**.

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).



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.

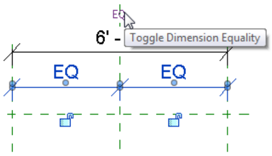


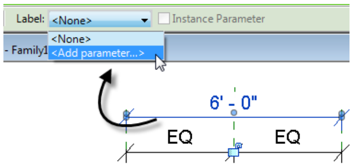
Aligned Dimension (DI)

Places dimensions between parallel references, or between multiple points.

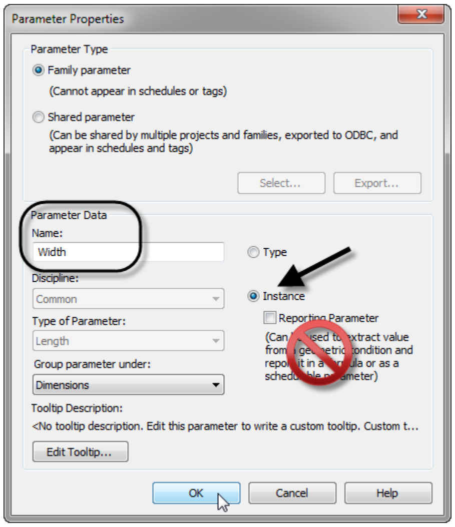
Press F1 for more help

9. Select each of the continuous strings and toggle on the Equality (click the small EQ icon).





10. Select the 6'-0" overall dimension. On the Options Bar, click the Label dropdown and choose: **<Add Parameter>**.
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.



● Family parameter
(Cannot appear in schedules or tags)

○ Shared parameter
(Can be shared by multiple projects and families, exported to ODBC, and appear in schedules and tags)

Name: Width

○ Type

● Instance

Discipline: Common

Type of Parameter: Length

☐ Reporting Parameter
(Can [illegible] used to extract value from [illegible] geometric condition and report it in a formula or as a schedulable parameter)

Group parameter under: Dimensions

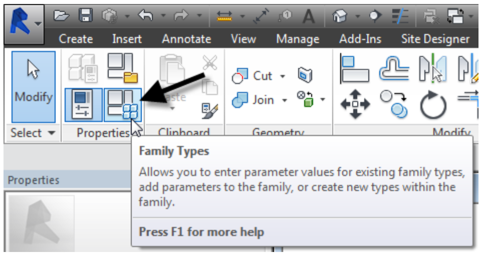
Tooltip Description:
<No tooltip description. Edit this parameter to write a custom tooltip. Custom t...

12. Repeat for the other dimension calling it **Depth** and also 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 will constrain the model in such a way as to 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.



Family Types

Allows you to enter parameter values for existing family types, add parameters to the family, or create new types within the family.

Press F1 for more help

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**.

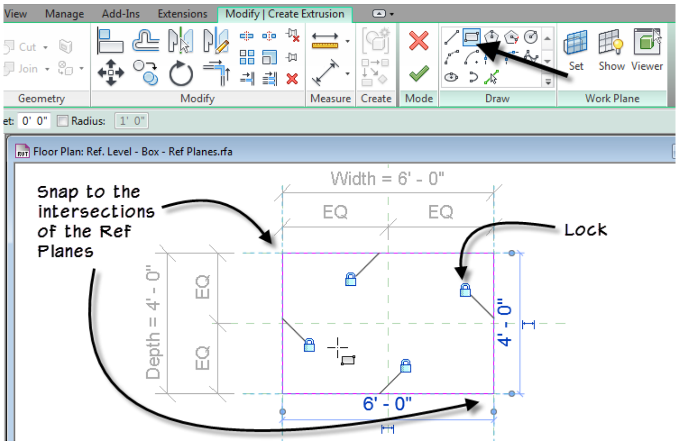
2. Add a dimension between the new reference plane and the Ref Level already in the file. Label the new dimension with a new parameter and call it **Height**. Make it an **Instance** parameter (not reporting).

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***.

Add Geometry

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

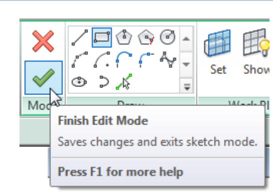
1. In *Plan* view, on the Create tab, click the Extrusion button.
2. On the Modify | Create Extrusion tab, on the Draw panel, click the Rectangle shape.



Snap to the intersections of the Ref Planes

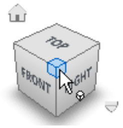
Lock

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.

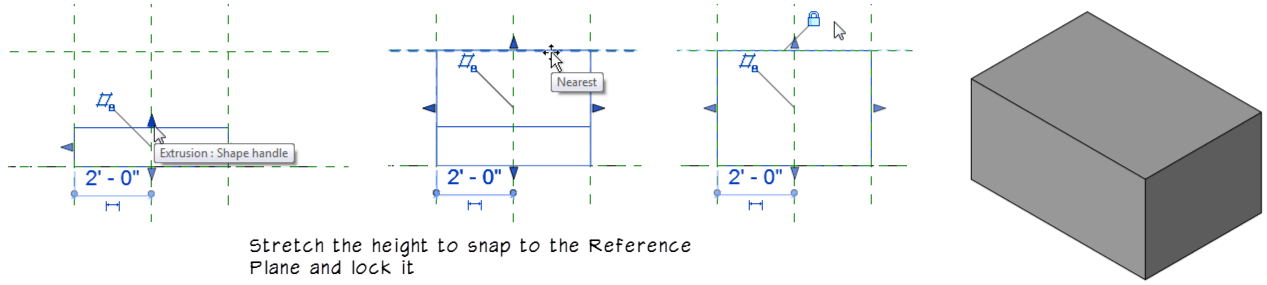


Finish Edit Mode
Saves changes and exits sketch mode.
Press F1 for more help

6. Zoom the 3D view to fit. You can do this quickly by clicking a corner of the ViewCube. 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.



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

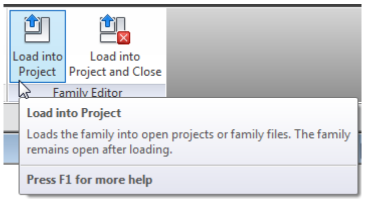
Load into a Project

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 and minimize the ***100 Sandbox.rvt*** project file earlier, please open and minimize it now.

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



Load into Project

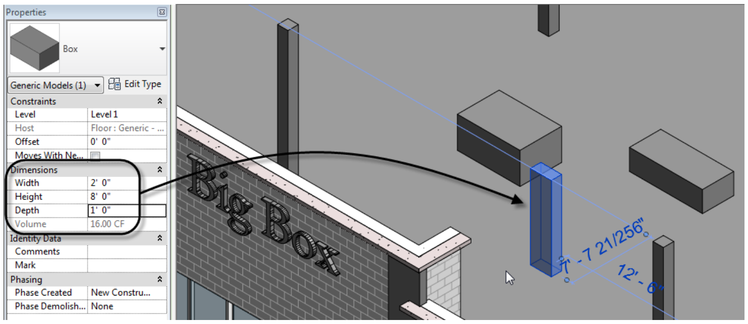
Loads the family into open projects or family files. The family remains open after loading.

Press F1 for more help

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

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 section.



Constraints	
Level	Level 1
Host	Floor : Generic - ...
Offset	0' 0"
Moves With Ne...	☐
Dimensions	
Width	2' 0"
Height	8' 0"
Depth	1' 0"
Volume	16.00 CF
Identity Data	
Comments	
Mark	
Phasing	
Phase Created	New Constru...
Phase Demolish...	None

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

Family Editor 102

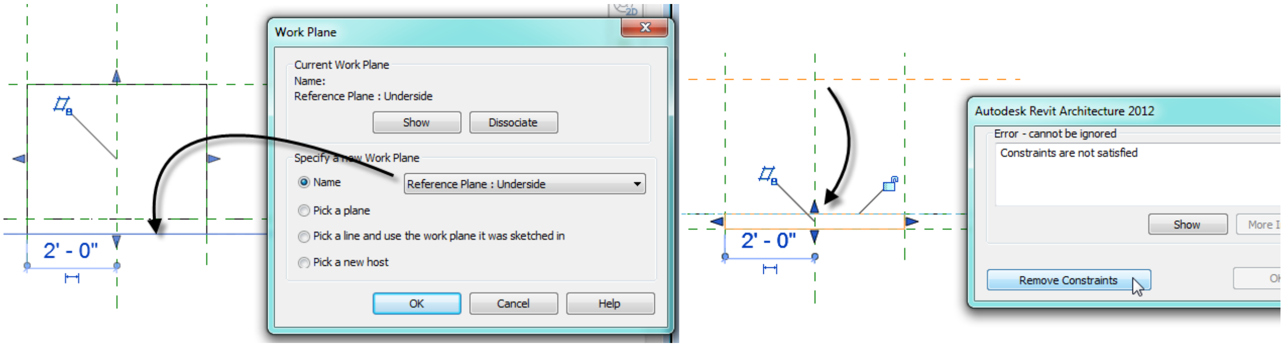
More than a box

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.

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.



Work Plane

Current Work Plane

Name:

Reference Plane : Underside

Specify a new Work Plane

● Name: Reference Plane : Underside

○ Pick a plane

○ Pick a line and use the work plane it was sketched in

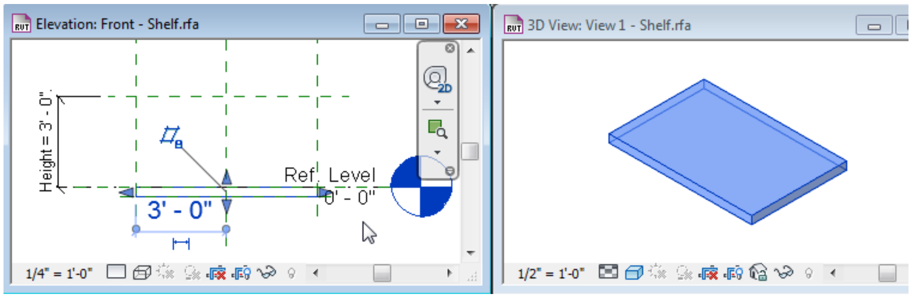
○ Pick a new host

Autodesk Revit Architecture 2012

Error - cannot be ignored

Constraints are not satisfied

4. Drag the control grip down and snap it to the Ref. Level. (This will cause an error). When the error dialog appears, click the: Remove Constraints button.
5. Lock the new position of the Extrusion shape handle. You should have a very short box completely below the Ref. Level.



Re-Constrain the Height

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

1. 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 if necessary.

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

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

3. Delete the Top Reference Plane.

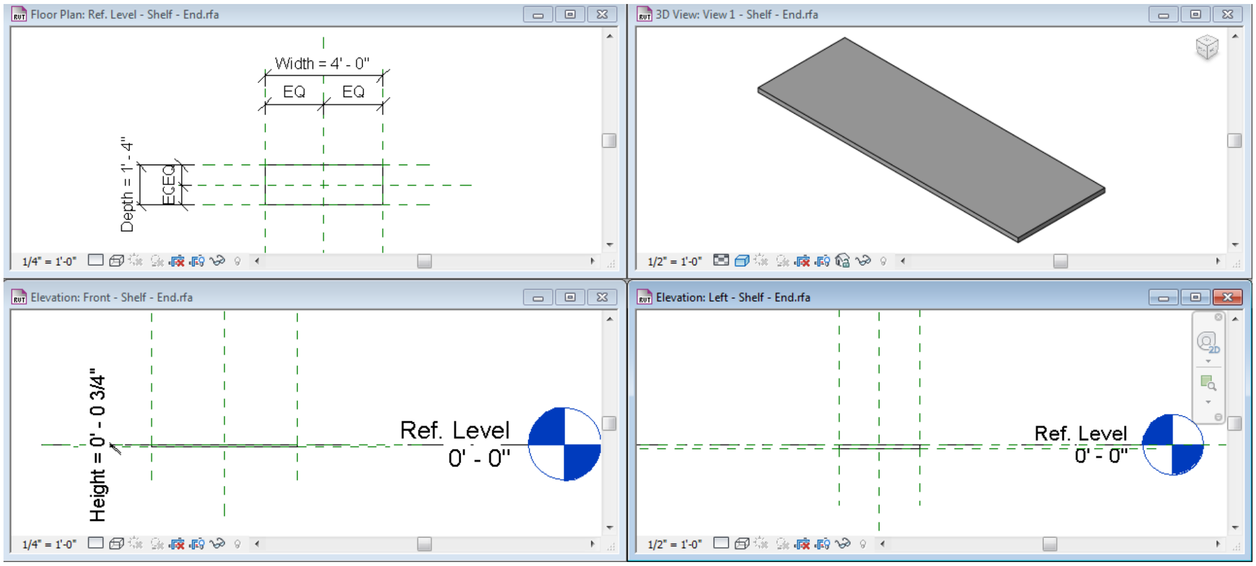
There are other ways to achieve a similar effect. This approach is taken to keep the extrusion height a positive number but make the insertion point of the shelf at the top surface of the shelf.

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.

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

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.


Floor Plan: Ref. Level - Shelf - End.rfa
Width = 4' - 0"
EQ
EQ
Depth = 1' - 4"
EQ
EQ
1/4" = 1'-0"
3D View: View 1 - Shelf - End.rfa
1/2" = 1'-0"
Elevation: Front - Shelf - End.rfa
Height = 0' - 0 3/4"
Ref. Level
0' - 0"
1/4" = 1'-0"
Elevation: Left - Shelf - End.rfa
Ref. Level
0' - 0"
1/2" = 1'-0"

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

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.

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

1. From the Application Menu (big "R"), 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.

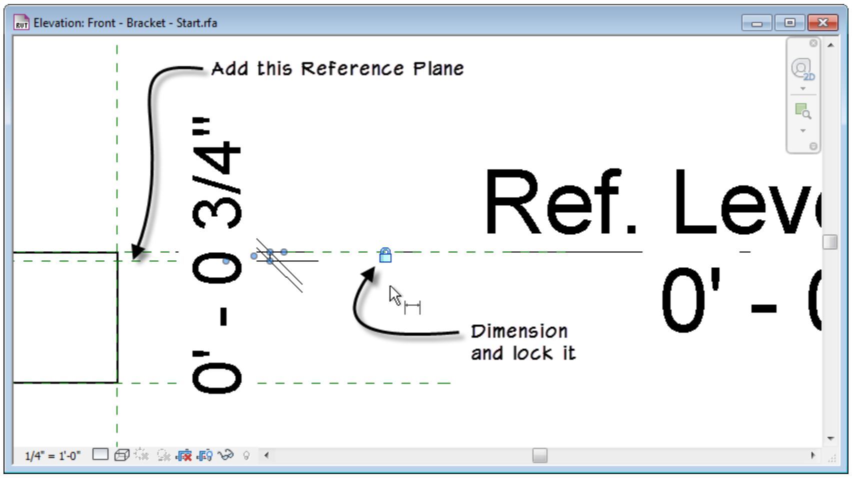
4. In the dimension, input a value of: **1'-0"** and press ENTER.

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: ¾" below the Ref. Level. Add a dimension between this one and the **Ref. Level**. Instead of adding a parameter, simply lock it. This will make this dimension "fixed" making it a constraint.

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



6. 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.
7. In the *Front* elevation view, **delete** the Extrusion. Save the file.

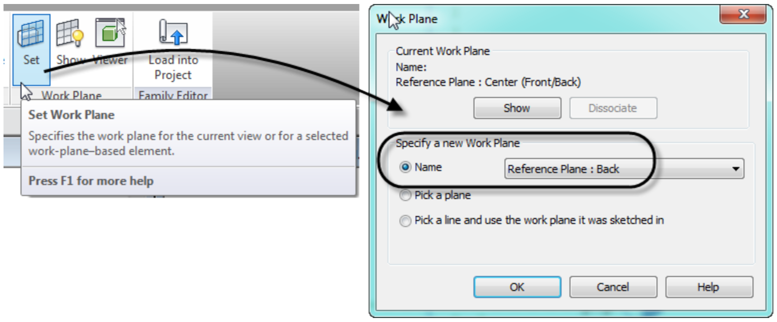
Create a new Blend

Let's now build some new geometry.

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

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. A better 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.



Set Work Plane

Specifies the work plane for the current view or for a selected work-plane-based element.

Press F1 for more help

Work Plane

Current Work Plane

Name:

Reference Plane : Center (Front/Back)

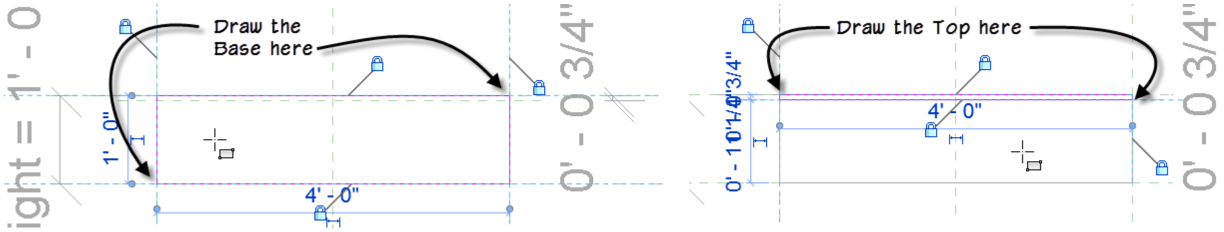
Specify a new Work Plane

● Name: Reference Plane : Back

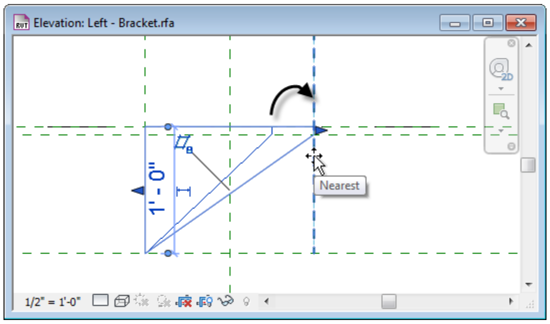
○ Pick a plane

○ Pick a line and use the work plane it was sketched in

2. In the "Work Plane" dialog, from the Name list, choose: **Reference Plane: Back** and then click OK.
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**.



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 ¾". **Lock all sides**.
7. On the ribbon, click the green checkmark button to finish.



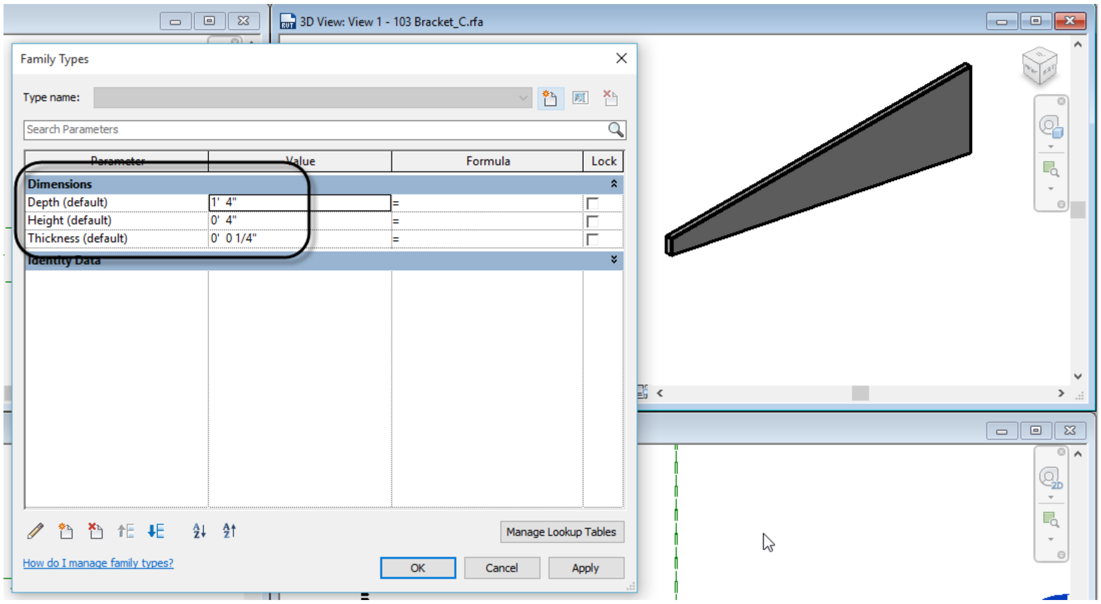
8. Zoom in on the Left elevation view. Use the control grip to drag the depth of the blend and snap it to the **Front** Reference Plane. **Lock it**.

Flex the Blend

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

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

1. Open Family Types.
2. Change the Thickness to: ¼", the Height to: **4"**. Leave the Depth at: **1'-4"** and then click OK.



Parameter	Value	Formula	Lock
Dimensions			
Depth (default)	1' 4"	=	☐
Height (default)	0' 4"	=	☐
Thickness (default)	0' 0 1/4"	=	☐
Identity Data			

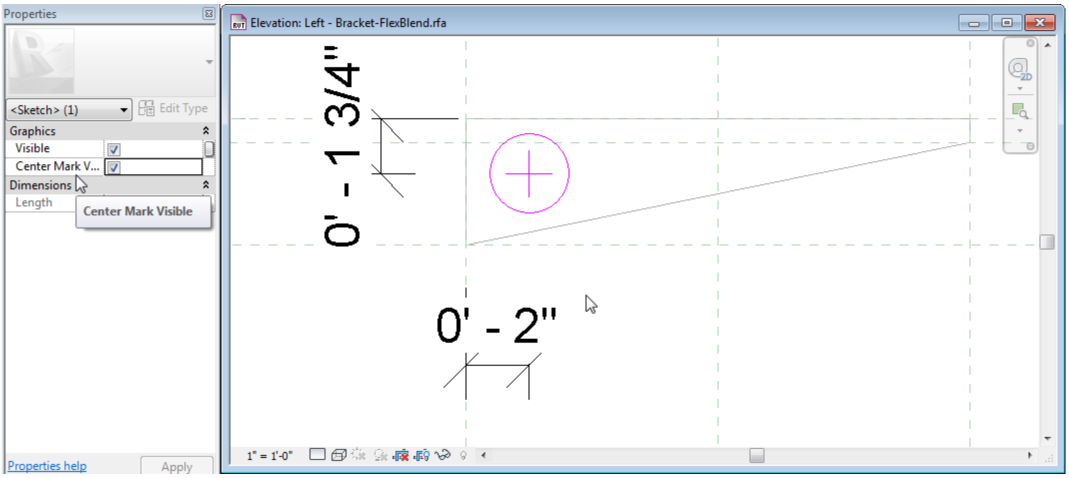
3. Save the file.

***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. Open the *Left* elevation view and zoom in on the bracket toward the back.
2. On the Create tab, click the **Void Forms** dropdown and then choose: **Void Extrusion**.
3. Create a circle inside the space of the bracket with a: **1 ¼"** radius.
4. Click the Modify tool to cancel. Select the circle and on the Properties palette, check the Center Mark Visible checkbox.
5. Add a dimension from the Ref. Level to the center of the circle. Select the circle and edit the dimension to: **1 ¾"**.

6. Add a second dimension from the Back reference plane to the center of the circle at: **2"** away.



<Sketch> (1)
Graphics
Visible ☑
Center Mark V... ☑
Dimensions
Length

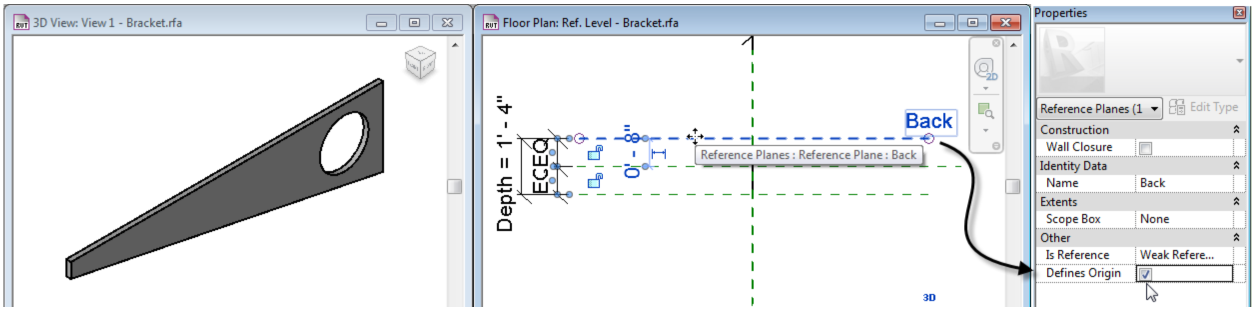
7. Click the Finish Edit Mode button.

Like any extrusion, this one defaulted to the last value used, or: 1'-0" here. It also uses a work plane of the Center (Left/Right) by default. You can edit the work plane, use the grips to edit the depth, or type in values on the Properties palette. We want the void to cut all the way through the blend.

8. In the *Front* view, adjust the depth of the Void to cut through the blend.

***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.


3D View: View 1 - Bracket.rfa
Floor Plan: Ref. Level - Bracket.rfa
Depth = 1' - 4"
EQ
EQ
0' - 0"
Back
Reference Planes : Reference Plane : Back
3D
Properties

Reference Planes (1
Construction
Wall Closure ☐
Identity Data
Name Back
Extents
Scope Box None
Other
Is Reference Weak Refere...
Defines Origin ☑

1. In the Plan view, select the Back reference plane. On the Properties palette, check the **Defines Origin** checkbox.

2. Save the ***Bracket.rfa*** file.

Family Editor 104

Nesting Families

To make more complex forms, you will often want to model it in parts. 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.

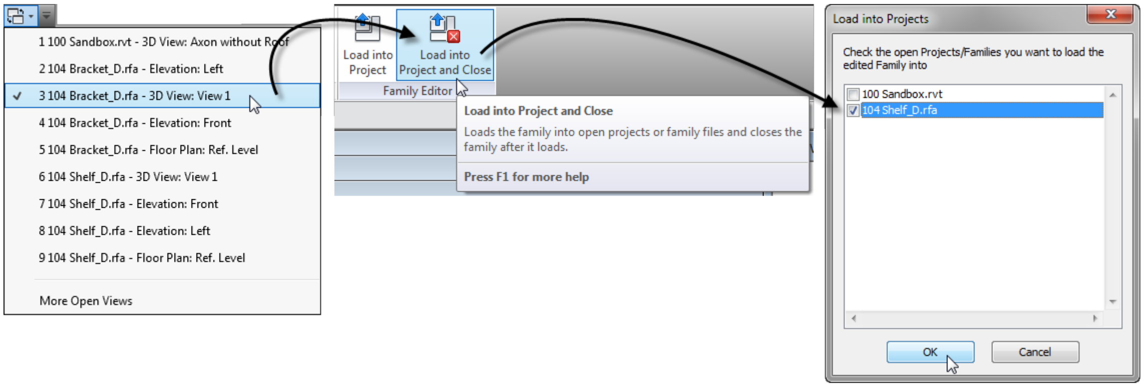
1. Keep your Bracket family open. Reopen your ***Shelf.rfa*** Family file.

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

2. Using Switch Windows, switch over to one of the views for the Bracket family such as: ***Bracket.rfa – 3D View: View 1***.
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.


1 100 Sandbox.rvt - 3D View: Axon without Roof
2 104 Bracket_D.rfa - Elevation: Left
✓ 3 104 Bracket_D.rfa - 3D View: View 1
4 104 Bracket_D.rfa - Elevation: Front
5 104 Bracket_D.rfa - Floor Plan: Ref. Level
6 104 Shelf_D.rfa - 3D View: View 1
7 104 Shelf_D.rfa - Elevation: Front
8 104 Shelf_D.rfa - Elevation: Left
9 104 Shelf_D.rfa - Floor Plan: Ref. Level
More Open Views
Load into Project
Load into Project and Close
Family Editor
OK
Cancel

Load into Project and Close

Loads the family into open projects or family files and closes the family after it loads.

Press F1 for more help

Load into Projects

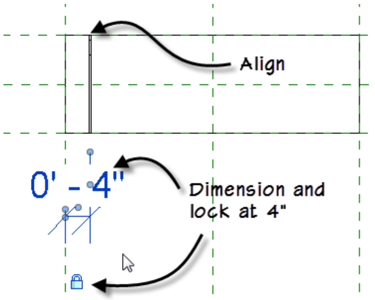
Check the open Projects/Families you want to load the edited Family into

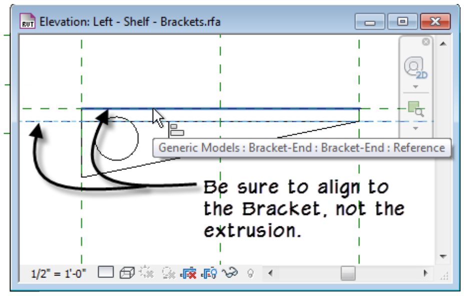
☐ 100 Sandbox.rvt

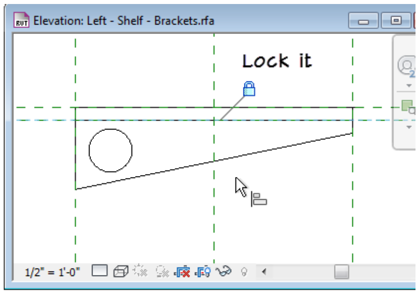
☑ 104 Shelf_D.rfa

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**.
9. Open the *Left* elevation view, on the Modify tab, click the Align tool.




Elevation: Left - Shelf - Brackets.rfa
Generic Models : Bracket-End : Bracket-End : Reference
Be sure to align to the Bracket, not the extrusion.
1/2" = 1'-0"



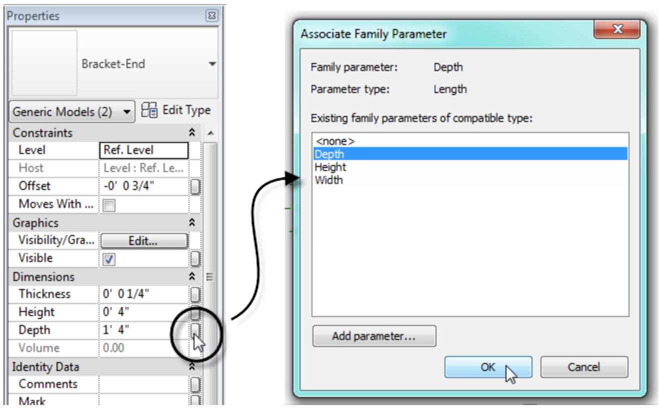
10. First select the Underside Ref. Plane for the alignment reference. Then be sure to use the TAB key and make sure you are aligning the top of the Bracket and *NOT* the Extrusion: Shape Handle. Lock it.
11. Mirror the Bracket to the other side and constrain it the same way. You must repeat all alignment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

***Linking Parameters between Host and Nested

Locking to the reference planes keeps the brackets positioned well as the shelf changes shape. If you like you can establish a relationship between the parameters in the host file (the Shelf) and the nested family (the Bracket).

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

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.



Constraints	
Level	Ref. Level
Host	Level : Ref. Le...
Offset	-0' 0 3/4"
Moves With ...	☐
Graphics	
Visibility/Gra...	Edit...
Visible	☑
Dimensions	
Thickness	0' 0 1/4"
Height	0' 4"
Depth	1' 4"
Volume	0.00
Identity Data	
Comments	
Mark	

Family parameter: Depth
Parameter type: Length
Existing family parameters of compatible type:
<none>
Depth
Height
Width

3. In the "Associate Family Parameter" dialog, choose **Depth** and then click OK.
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 open. 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.
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.

Family Editor 201

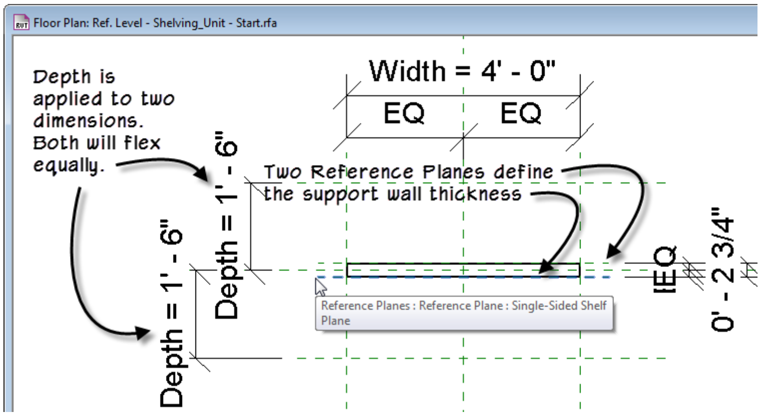
Assemble a complex Family

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.

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. With the exception of 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.



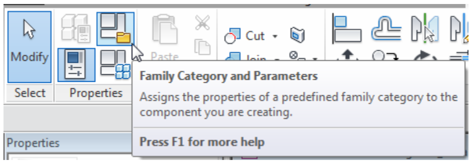
Floor Plan: Ref. Level - Shelving_Unit - Start.rfa

Depth is applied to two dimensions. Both will flex equally.

Two Reference Planes define the support wall thickness

Reference Planes : Reference Plane : Single-Sided Shelf Plane

2. Directly above the Family Types button click the **Family Category and Parameters** button.



Family Category and Parameters

Assigns the properties of a predefined family category to the component you are creating.

Press F1 for more help

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. 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.

***Nest in and Position the Shelf

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

1. Using Switch Windows, switch over to plan view of your **Shelf w Brackets** Family.
2. Following the steps above, set the Back Reference Plane as the Origin and Save the Family.

CATCH UP! You can open the file completed to this point named: ***201 Shelf w Brackets_B.rfa***.

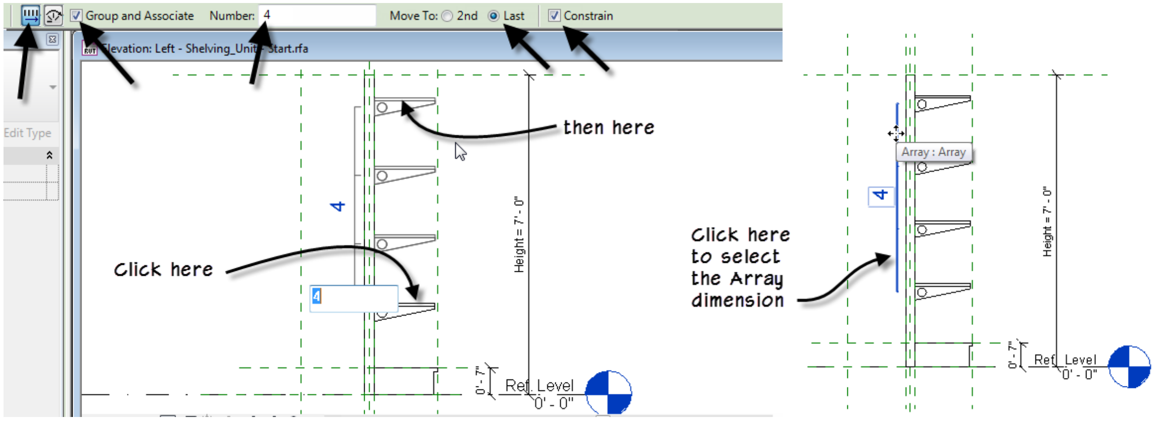
1. On the ribbon, click the: **Load into Project and Close** button. In the dialog that appears, check the *Shelving_Unit* family only and then click OK.
2. Place an instance of the Shelf in the plan view in the lower portion of the screen.
3. Using the Align tool, align the center of the shelf to the center of the host family. Lock it.
4. Align the back of the Shelf w Brackets to the reference plane named **Single-Sided Shelf Plane**. Lock it as well.
5. Cancel the Align command and then Tile the windows (press WT).
6. In the *Left* view, select the Shelf. On the Properties palette, change the Offset to: **2'-0"**.

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 later and update the geometry.

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

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**.



☑ Group and Associate Number: 4 Move To: ○ 2nd ● Last ☑ Constrain

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

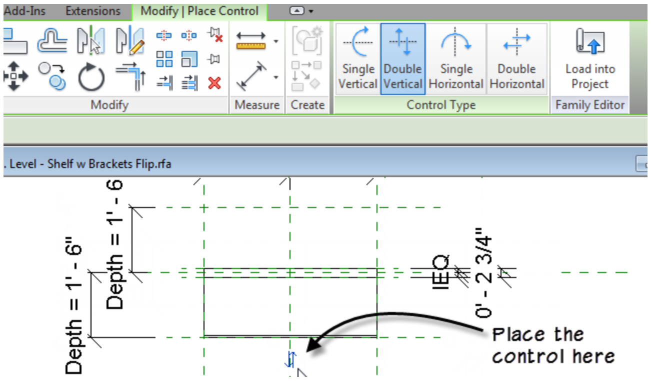
You can try different numbers for the array value to test it if you wish. Let's now assign a parameter to the quantity.

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.

***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.



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

Load into the Sandbox

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

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

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.

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

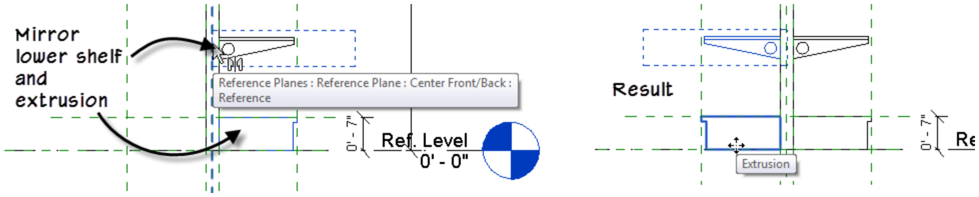
One-sided or Two-sided?

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: ***301 Shelving Unit_D.rfa***.

2. In the *Left* elevation view, select the base shelf extrusion (it was already in the file above) and the lowermost shelf in the array.
3. 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.


Mirror lower shelf and extrusion
Reference Planes : Reference Plane : Center Front/Back : Reference
0' - 7"
Ref. Level
0' - 0"
Result
Extrusion
0' - 7"
Re

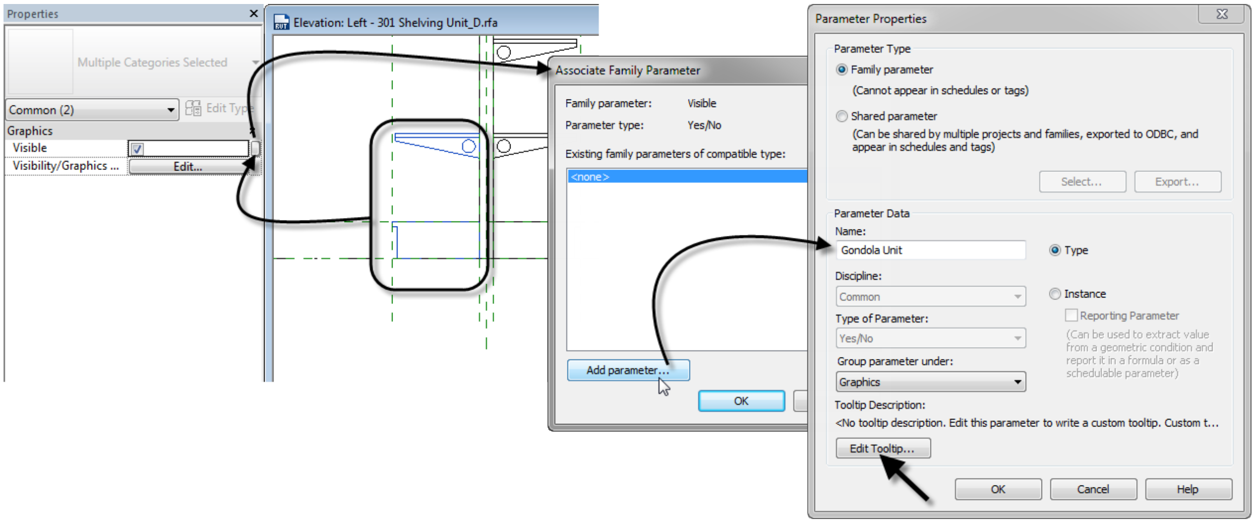
4. Select the new shelf w brackets created by the mirror (Only the mirrored one). It will still be grouped. On the ribbon, click the **Ungroup** button.

When you array, it creates a group. Copying grouped elements do not keep them associated with the original array but will leave them grouped. So ungrouping the mirrored shelf is important here. Be sure however to ungroup ***ONLY*** the mirrored shelf. Not the ones on the other side of the wall.

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.

1. Select the two mirrored elements (one extrusion and the ungrouped shelf w brackets).
2. 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).
3. In the "Associate Family Parameter" dialog that appears, click the **Add Parameter** button at the bottom.
4. In the "Parameter Properties" dialog, name it: **Gondola Unit**, group it under: **Graphics**.
5. Near the bottom of the dialog, click the Edit Tooltip button.


Properties
Multiple Categories Selected
Common (2)
Edit Type
Graphics
Visible
Visibility/Graphics ...
Edit...
Elevation: Left - 301 Shelving Unit_D.rfa
Associate Family Parameter
Existing family parameters of compatible type:
<none>
Add parameter...
OK
Parameter Properties
Parameter Type
Family parameter
(Cannot appear in schedules or tags)
Shared parameter
(Can be shared by multiple projects and families, exported to ODBC, and appear in schedules and tags)
Select...
Export...
Parameter Data
Type
Instance
Reporting Parameter
(Can be used to extract value from a geometric condition and report it in a formula or as a schedulable parameter)
Edit Tooltip...
OK
Cancel
Help

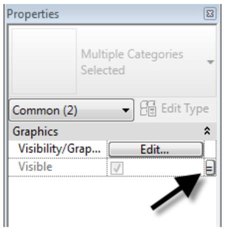
☑ Visible

Family parameter: Visible
Parameter type: Yes/No

● Family parameter
○ Shared parameter

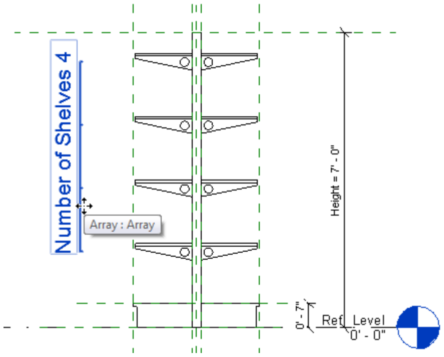
Name: Gondola Unit
● Type
○ Instance
☐ Reporting Parameter
Discipline: Common
Type of Parameter: Yes/No
Group parameter under: Graphics
Tooltip Description:
<No tooltip description. Edit this parameter to write a custom tooltip. Custom t...

6. Select all of the text in the "Edit Tooltip" dialog and then type: "Determines if the unit is one-sided or two-sided."
7. Click OK three times.



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

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.



***Create Family Types

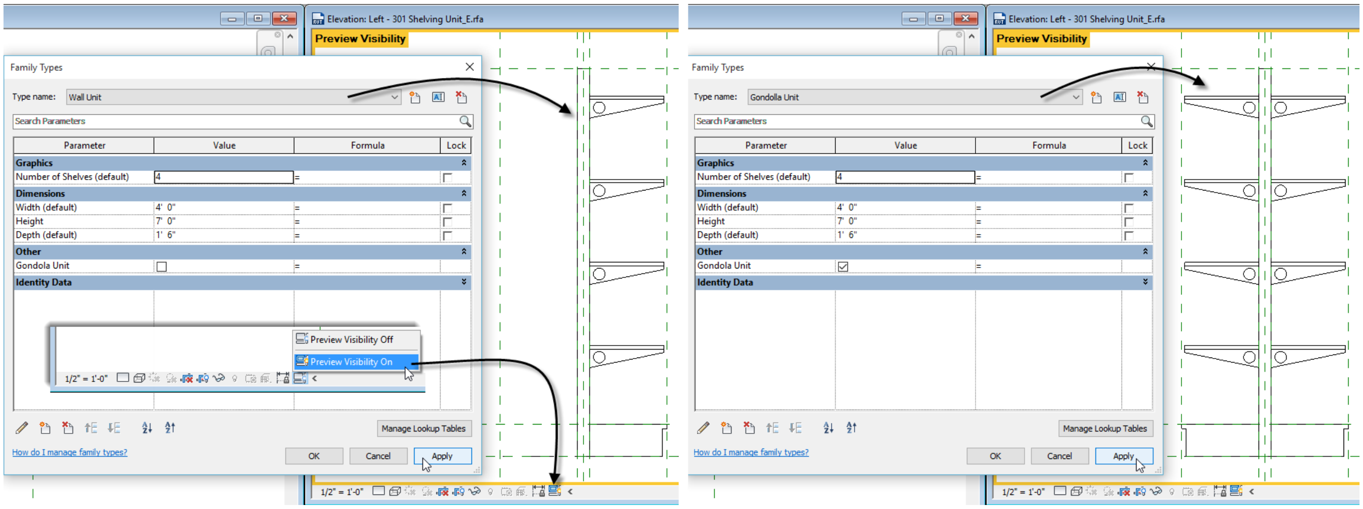
To test our new parameters, let's create a few family types.

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

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.



Type name: Wall Unit
Number of Shelves (default) 4
Width (default) 4' 0"
Height 7' 0"
Depth (default) 1' 6"
Gondola Unit ☐

Type name: Gondola Unit
Number of Shelves (default) 4
Width (default) 4' 0"
Height 7' 0"
Depth (default) 1' 6"
Gondola Unit ☑

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 will 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.

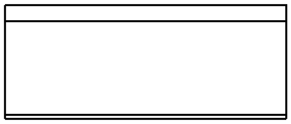
***Family Editor 302

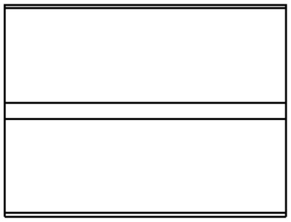
Tweaking the Graphics

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.

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 of these lines. Firstly, let's assume that the design is correctly modeled, and depth of the bottom shelf is a bit larger. We can adjust which elements Revit displays in a number of ways.



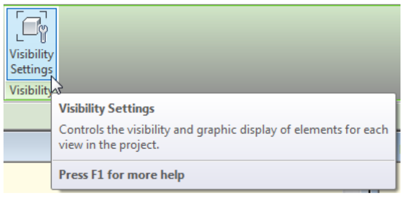


2. Minimize the floor plan to switch back over to the ***Shelving Unit*** Family file.

***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.

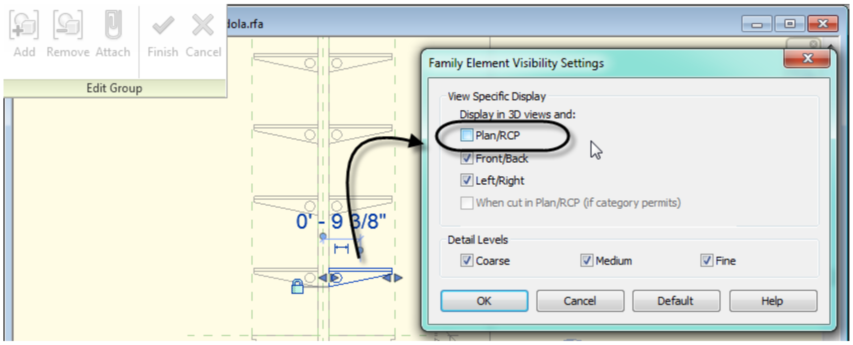


Visibility Settings

Controls the visibility and graphic display of elements for each view in the project.

Press F1 for more help

4. In the "Family Element Visibility Settings" dialog, uncheck the Plan/RCP box and then click OK.



Family Element Visibility Settings

View Specific Display

Display in 3D views and:

☐ Plan/RCP

☑ Front/Back

☑ Left/Right

☐ When cut in Plan/RCP (if category permits)

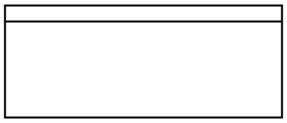
Detail Levels

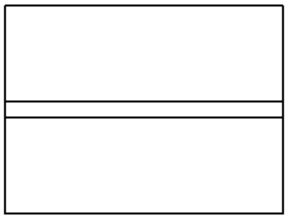
☑ Coarse ☑ Medium ☑ Fine

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 use Preview Visibility or reload it into the project to test it.





***Family Editor 303

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.

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

CATCH UP! You can open the file completed to this point named: ***303 Shelving Unit_F.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.

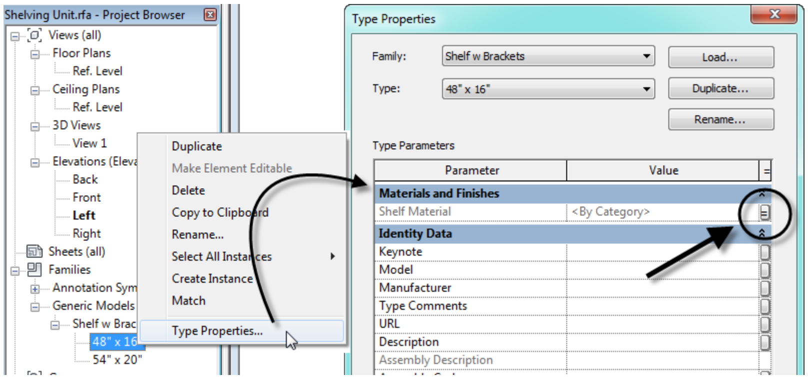
For this example, we will only edit the shelf material. See the note below if you wish to also edit the material of the brackets.

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 have to create a matching parameter in the host family and link it up with 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.



Family: Shelf w Brackets

Type: 48" x 16"

Shelf Material	<By Category>

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 have to 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 all pieces is included with the lab dataset: ***303 Shelving Unit_G.rfa***.

***Family Editor 401

Bonus Material – Formulas

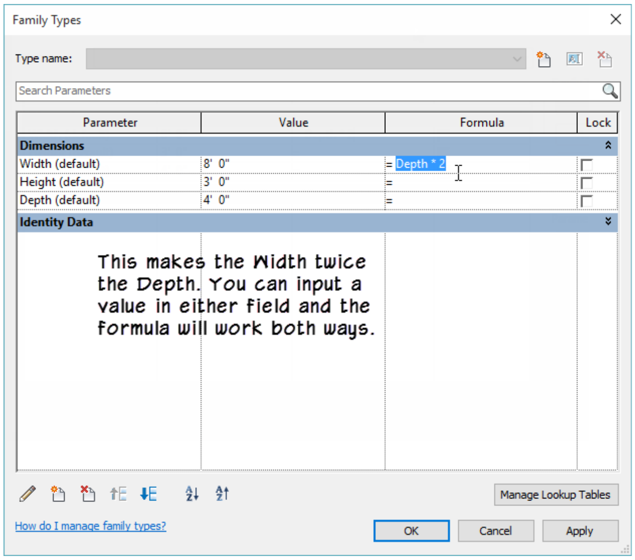
One of the most powerful features of the family editor is the ability to use mathematical formulas to drive parameters in the Family Types dialog. Formulas can be simple arithmetic expressions that create a direct relationship between one or more parameters or complex conditional and trigonometry statements 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 fairly straightforward examples here just to give you some idea of the possibilities. Covering the topic of formulas in-depth would require an entire lab or likely several labs to do it justice. (If you would like to see more, check out my lecture class tomorrow at 10AM: **AS10644** Taming Parametric Curves in Revit Family Editor).

1. Keep the ***Sandbox*** file open and minimized and close other files.

2. To try some of these ideas, a version of the Box file has been provided. It is called: ***401 Box_A.rfa***.

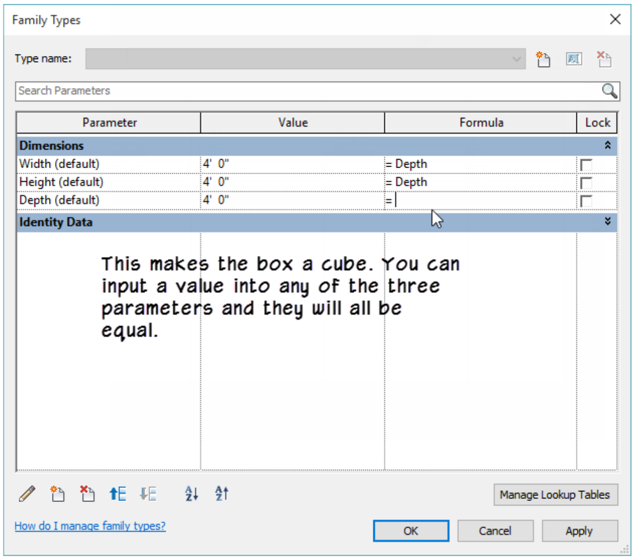
Simple Arithmetic

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.



Parameter	Value	Formula	Lock
Dimensions			
Width (default)	8' 0"	= Depth * 2	☐
Height (default)	3' 0"	=	☐
Depth (default)	4' 0"	=	☐
Identity Data			

This makes the Width twice the Depth. You can input a value in either field and the formula will work both ways.



Parameter	Value	Formula	Lock
Dimensions			
Width (default)	4' 0"	= Depth	☐
Height (default)	4' 0"	= Depth	☐
Depth (default)	4' 0"	=	☐
Identity Data			

This makes the box a cube. You can input a value into any of the three parameters and they will all be equal.

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

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

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.

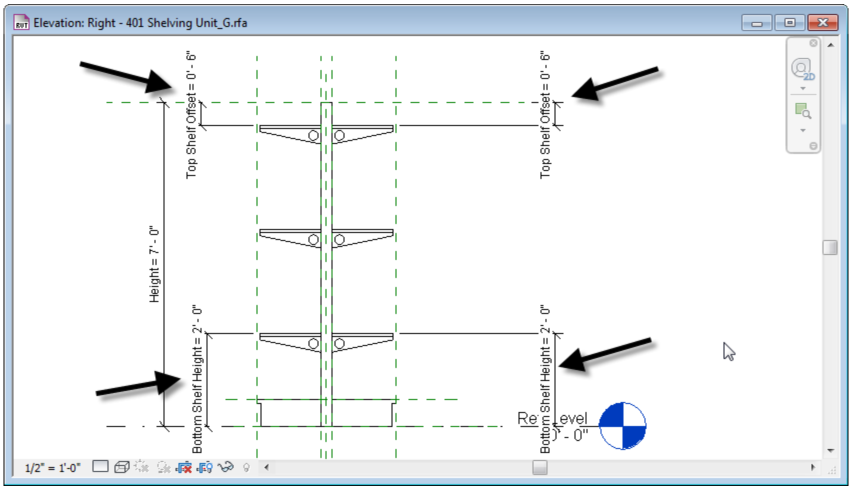
You try other options as well. The figure shows an example to make a cube. Try others.

***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_H.rfa***.

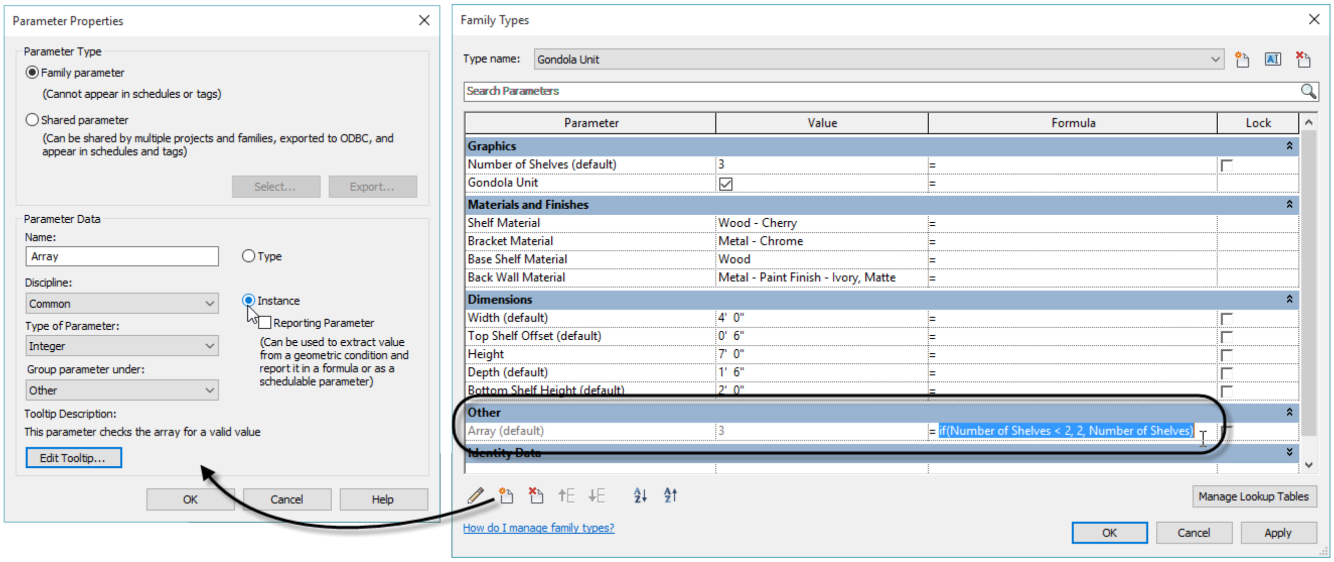
This is a copy of our ***Shelving Unit*** family with a few parameters added. **Bottom Shelf Height** sets the height of the first shelf in the array and is measured from the floor Level. **Top Shelf Offset** sets the distance below the Top reference plane for the topmost shelf. These give a little more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


Elevation: Right - 401 Shelving Unit_G.rfa
Top Shelf Offset = 0' - 6"
Top Shelf Offset = 0' - 6"
Height = 7' - 0"
Bottom Shelf Height = 2' - 0"
Bottom Shelf Height = 2' - 0"
Ref. Level
0' - 0"
1/2" = 1'-0"

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**. 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.



Parameter Type
● Family parameter
(Cannot appear in schedules or tags)
○ Shared parameter
(Can be shared by multiple projects and families, exported to ODBC, and appear in schedules and tags)

Parameter Data
Name: Array
○ Type
● Instance
☐ Reporting Parameter
(Can be used to extract value from a geometric condition and report it in a formula or as a schedulable parameter)
Discipline: Common
Type of Parameter: Integer
Group parameter under: Other
Tooltip Description:
This parameter checks the array for a valid value

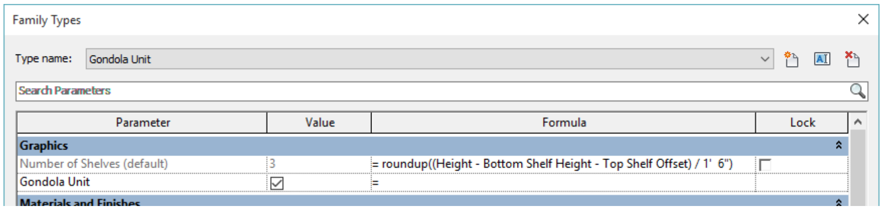
Type name: Gondola Unit

Parameter	Value	Formula	Lock
Graphics			
Number of Shelves (default)	3	=	☐
Gondola Unit	☑	=	
Materials and Finishes			
Shelf Material	Wood - Cherry	=	
Bracket Material	Metal - Chrome	=	
Base Shelf Material	Wood	=	
Back Wall Material	Metal - Paint Finish - Ivory, Matte	=	
Dimensions			
Width (default)	4' 0"	=	☐
Top Shelf Offset (default)	0' 6"	=	☐
Height	7' 0"	=	☐
Depth (default)	1' 6"	=	☐
Bottom Shelf Height (default)	2' 0"	=	☐
Other			
Array (default)	3	= if(Number of Shelves < 2, 2, Number of Shelves)	☐
Identity Data			

To use this new parameter, assign it to the array dimensions in the family in place of Number of Shelves.

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. In the formula field next to Number of Shelves, type this:
   roundup((Height - Bottom Shelf Height - Top Shelf Offset) / 1' 6")



Type name: Gondola Unit

Parameter	Value	Formula	Lock
Graphics			
Number of Shelves (default)	3	= roundup((Height - Bottom Shelf Height - Top Shelf Offset) / 1' 6")	☐
Gondola Unit	☑	=	
Materials and Finishes			

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. 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 "maximum" value or "not to exceed."

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

Family Editor 501

What now?

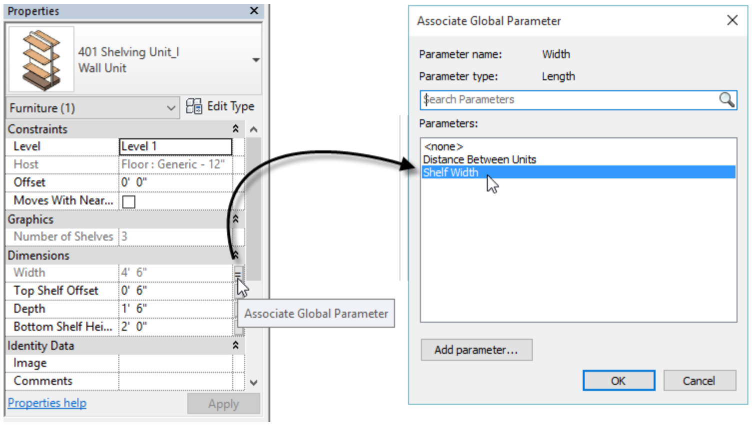
Well, that is all we have time for. (Probably more than we have time for). There are so many ways to enhance even this simple family. Consider adding more parameters and formulas to control shelf placement, offsets, add end panels, etc. You can add more geometry to refine the design add moldings, different kinds of shelves, etc. You can add more family types with additional variations. The sky's the limit. Feel free to send me a screen shot or finished family file of what you come up with. I'd love to see your progress.

***Global Parameters Bonus

Create a collection of shelving units

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! To do this, you must have at version 2016 R2 installed. This will give you the global parameter feature.

1. Load the final shelving unit *401 Shelving_Unit_I.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**.



Properties

401 Shelving Unit_I
Wall Unit

Furniture (1)

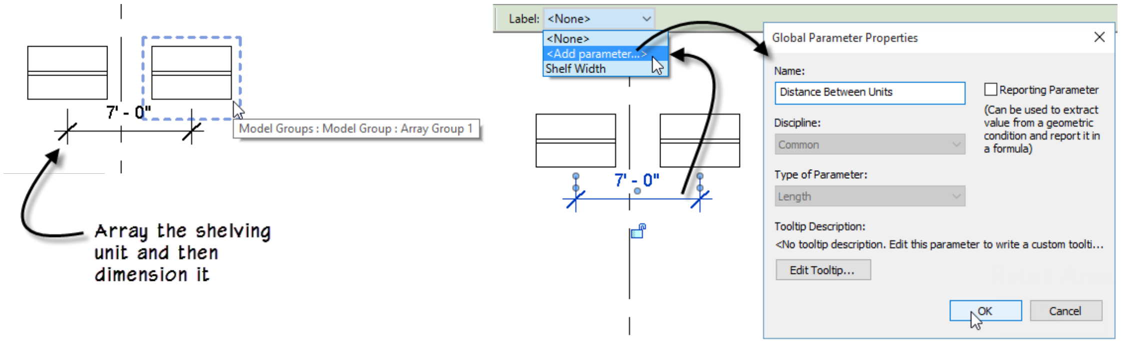
Constraints	
Level	Level 1
Host	Floor : Generic - 12"
Offset	0' 0"
Moves With Near...	☐
Graphics	
Number of Shelves	3
Dimensions	
Width	4' 6"
Top Shelf Offset	0' 6"
Depth	1' 6"
Bottom Shelf Hei...	2' 0"
Identity Data	
Image	
Comments	

Associate Global Parameter

Parameter name: Width
Parameter type: Length

Parameters:
<none>
Distance Between Units
Shelf Width

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 Options Bar. Click it and choose: **<Add Parameter>**.
7. Name the parameter: **Distance Between Units** and then click OK.



7' - 0"
Model Groups : Model Group : Array Group 1
Array the shelving unit and then dimension it
Label: <None>
<None>
<Add parameter...>
Shelf Width
7' - 0"

Global Parameter Properties

Name: Distance Between Units

☐ Reporting Parameter
(Can be used to extract value from a geometric condition and report it in a formula)

Discipline: Common

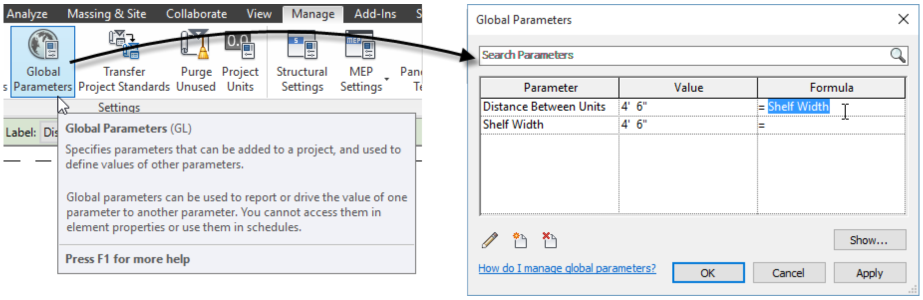
Type of Parameter: Length

Tooltip Description:
<No tooltip description. Edit this parameter to write a custom toolti...

Edit Tooltip...
OK
Cancel

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).



Global Parameters (GL)

Specifies parameters that can be added to a project, and used to define values of other parameters.

Global parameters can be used to report or drive the value of one parameter to another parameter. You cannot access them in element properties or use them in schedules.

Press F1 for more help

Parameter	Value	Formula
Distance Between Units	4' 6"	= Shelf Width
Shelf Width	4' 6"	=

10. Click OK to finish.

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 (direclty 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.

Please feel free to experiment further. Thank you for attending, and **please fill out your evaluation!**

Further Study

You can find more information and tutorials in:

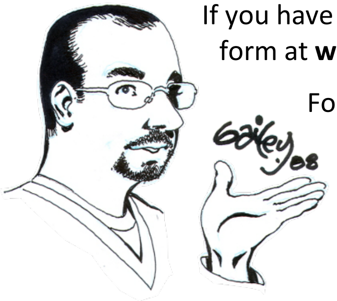
***Renaissance Revit: Creating Classical Architecture with Modern Software*.** This book can be thought of as a "deep dive" into the family editor. It starts with the basics, but gets very advanced as well. The entire book is on family creation using classical architectural examples. Both the traditional 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 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 class have papers and materials available for download from my website: www.paulaubin.com/au

If you prefer video training, I have several Revit video courses at: **www.lynda.com/paulaubin**. Check out: *Revit Essential Training*, *Revit Family Editor*, *Revit Family Curves and Formulas* and *Revit Advanced Modeling*.



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**

Thank you for attending. Please fill out your evaluation.