

# Supersize Me: Worksets with Links, Object Protection, Families Versus Groups Versus Links

Lonnie Cumpton – BIM9  
Bill Debevc – BIM9  
Joe Eichenseer – IMAGINiT Technologies

## AB3344

This Supersize Me class focuses on delivering large or mammoth projects with Autodesk® Revit® software. We will focus on three topics that can make or break a large Revit project. First, we will show you how worksets and links together unlock the gates of a large Revit project, and how these two areas represent one of the fundamental pieces of having a successful large Revit project. Next, on any Revit project, when something gets deleted by accident, it can be painful. On a large project it can cause days or weeks worth of re-work. We will show you how to protect Revit objects from being moved, deleted, or modified. Finally, we will take a hard look at families versus groups versus links. Unfortunately they have similar characteristics and often get used in place of each other. We will talk about each tool and discuss why and how they work, when to use them, and when not to use them. You will gain an understanding of why each tool can be both, destructive or constructive to any Revit project.

## Learning Objectives

At the end of this class, you will be able to:

- Describe worksets and links and how together they unlock performance on a mammoth project
- Protect Revit objects on massive projects
- Appropriately use families versus groups versus links on a large Revit project
- Develop best practices for a mammoth Revit project, but also apply these practices to projects of all sizes

## About the Speakers

*Lonnie Cumpston* has over 20 years of experience in consulting, implementing, training and corporate management at local and regional levels into the AEC community. He has a comprehensive understanding of building design and construction methods currently specializing in private BIM clouds. Lonnie is drawing on his experience with assessing needs, defining requirements and resolving challenges to deliver the optimized BIM9 solution to gain the benefits of cloud computing as it relates to BIM. His unique BIM9 strategy provides secure access to all your design data and applications from your office to mobile users around the world. You can find Lonnie on LinkedIn at BIM9.com and via email [Lonnie@bim9.com](mailto:Lonnie@bim9.com)

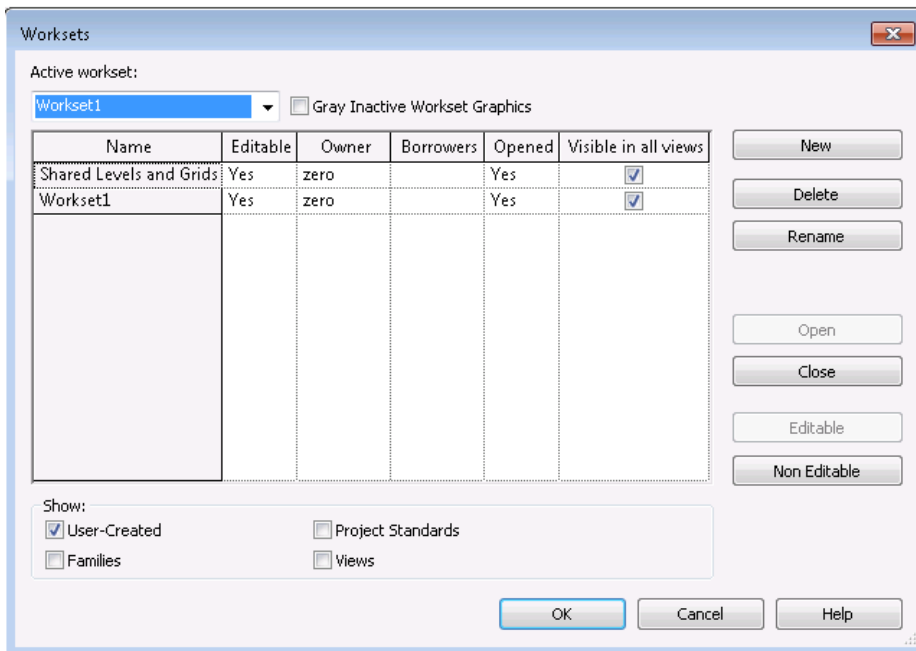
*Bill Debevc* is an expert in Autodesk software and Information Technology (I.T.) specializing in Private BIM Cloud technologies. He has over 20 years of experience using, supporting, and customizing Autodesk AutoCAD and Revit software. As a Microsoft Certified Systems Expert (MCSE) he uses his extensive I.T. experience to implement industry leading I.T. solutions for the AEC industry. Currently, as Technical Manager / Collaboration Solutions at BIM9, Bill is now sharing his knowledge about Private BIM Clouds helping the AEC industry reduce costs and increase productivity. You can find Bill at Bim9.com or via email at [bill@bim9.com](mailto:bill@bim9.com)

*Joe Eichenseer* manages IMAGINiT's North American team of building design and construction trainers and consultants, and is an Architecture, Engineering, and Construction (AEC) technology expert and consultant with more than 17 years of experience focusing on creating and managing environments that allows team members to be more creative, productive, efficient, and profitable. He has preeminent expertise in the implementation, integration, and training of advanced design and Building Information Modeling (BIM) software. Areas of expertise include System Integration and Data Sharing, Process Standardization and Documentation, Client and Team Productivity Improvement, Technical Problem Solving, and Coaching and Mentoring

## Worksets and Links the keys to working on a Large Project

### Worksets

If you have more than one person working on a Revit project at one time then you are using worksets. The process of enabling worksharing is the turning on of the worksets feature. This is also when you convert a standard Revit file into a central file. One of the challenges in companies today is that users are often working with Central files and worksets but they do not know what the real purpose of these tools are. In large part this has happened because the use of worksets has changed as Revit has been developed.



When worksets were first introduced you had to make sure the object you wanted to work on was on a workset and you had to check out and taken ownership of that workset to be able to modify those objects. That caused organizations to come up with several workset strategies allowing people to work on one project at

one time. When we checkout a workset the rest of the team could not make modifications to anything on the workset. This forced all of us to get creative and try to predict how the team would work over the course of a project. That all changed when element borrowing was added to Revit. With element borrowing the situation changed completely. The requirement to have object on different worksets so that more than one user could work at a time went away. In many situations today companies do not even create extra worksets. They just go with the default worksets and start modeling. To be honest on smaller projects this situation is fine. Unfortunately this has caused many users to not really pay attention to worksets and in many situations users do not even realize that worksets are enabled. It is surprising that the average Revit user today does not understand that everything you draw in Revit is being placed on a workset. This makes it very difficult for BIM managers to use worksets as part of a BIM strategy. Today BIM Managers use worksets for a number of different things. Some of these the community agrees with other are considered bad.

List of workset uses:

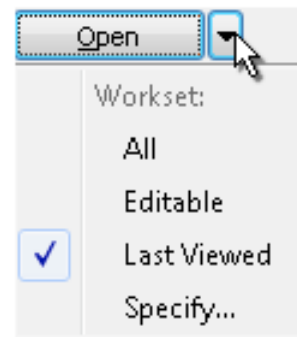
- Control Visibility of Objects
- Managing Project Size
- Discipline Separation
- Team Work Structure

For this session we are going to talk about worksets as they relate to file size and workflows on a large project. The #1 challenge on a large project in Revit is managing the Revit project file size. As you work on a large project you need to understand two key elements about your computer hardware.

1. CPU Speed
2. Amount of RAM you have.

With smaller Revit files CPU speed is not as noticeable because today's CPUs run so fast that a smaller model never really tasks the system. As a Revit model grows the CPU's ability to keep up gets tested. This is when you will start to notice how fast your CPU really is. You will see this in almost every action you take. Zooming in and out, panning, moving objects all of these operations are impacted by CPU speed. The faster your CPU is the smoother these operations are in larger projects. The second piece of the hardware puzzle is RAM. The speed of your RAM can have an impact on Revit performance but the important part is to know that you have enough RAM to load your project. Nothing will make your large Revit project run slower than your system running out of RAM. The industry uses a simple calculation we call the rule of 20 to determine if a computer has enough RAM to load a project. Just take your file size multiply it by 20 and you will roughly know how much RAM it will take to load that file. For example if your file size is 300MB, using the rule of 20 you will need 6000MB or 6GBs of RAM to load that file into Revit. Most computers today come with 8GB or 16GB of RAM. That usually means that you have enough RAM to load your files. The part that people forget are the other applications you have running. Every application you have running on your computer consumes RAM. So a 300MB Revit file, Outlook, Word, Internet Explorer, Windows and a couple other apps and the 8GB of RAM is gone. Leaving you with what we call disk swapping, that translates into everything on your computer running really slow.

You might be asking yourself what does all this hardware talk have to do with worksets. The answer of course is **"EVERYTHING"**. The key is in the knowledge that when a workset is "not loaded" it does not use any CPU or RAM. This is how we are able to work on Revit projects that are 500MB or more and our systems are still running at fast speeds. If you divide your project up with worksets and then use the "Open Specify" option you can tell Revit what worksets you want to open/load. This



means that Revit is working with a smaller data set and does not require as much CPU or RAM to open the file. This strategy used properly can keep a team working at top speed even when they are working on a massive Revit project. The key of course is to build a workset structure that supports the division of the project into areas that align with the team size and company workflow. The second part is to get the team educated so they understand and follow the workset strategy. A common example of this is in large tower projects. Usually in a tower project floors are grouped together because the core and shell footprint is the same for those floors. In a project like this you may find a workset list like this:

- Core and Shell 1 – 10
- Core and Shell 11 – 15
- Core and Shell 16 – 20
- Build out 1 – 10
- Build out 11 – 15
- Build out 16 – 20

With this type of structure a user that is working on the 18th floor can load two worksets and have complete access to a subset of the building. This subset will run faster and require less processing power. Ultimately letting the user feel like they are working on a much smaller file. This concept can be used on a large project that is spread out horizontally as well. Just divide the building into sections and create worksets for those sections. The key is to take some time early in the project and figure out how the project and work is going to be divided up and then build a workset strategy around that plan.



## **Links**

When working on a large project most users think about links as a way to reduce file size. The basic concept is to break the building into parts and then build a Revit file for each part. Then of course link them together to create the complete project. This concept seems very logical because that is how we work with consultants. Each discipline has its own model and then we link those together to create the complete project. As we covered in the section about worksets file size can be managed with worksets. When you are looking at ways to work with a large project we recommend that you look to worksets to manage file size before you start looking and dividing the model into

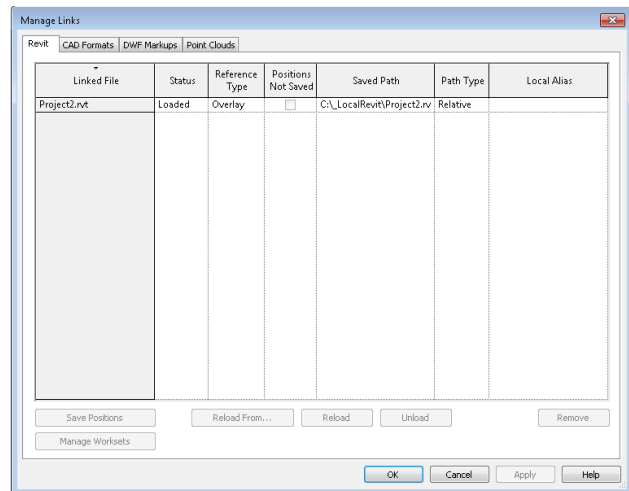
multiple Revit files. The general nature of Revit is to be a single file environment. As soon as you start breaking that concept you have trade offs. Autodesk has been working hard for the last several releases of Revit to make linked models work better together. In the end links work great and when used properly they can help you deliver a large project. At the same time the limitations of links can cause you frustration late in the project. The issue of course is the limitations of what you can bring across from a linked file into the current file. The most common issue is the coordination between annotation markers (section/detail markers) and views. One of the most powerful tools in Revit is that when you cut a section the marker that indicates the section and the view that is created are automatically coordinated. That way your sheets are always coordinated. When you start using links this



core functionality is not transferable from Revit file to Revit file. This can make the documentation process much more complicated. If you need to reference a detail or section in another Revit file you have to start creating dummy views and sheets to get them to reference properly. Then of course you have to remember to fix the dummy information if the original ever changes. This is one of several examples of how multiple Revit files can cause you difficulties. The key we have found is if you have to break your model into more than one Revit file do so based on deliverable packages. You want to think of each Revit file as its own complete documentation package. Most large projects are divided up into phases or sections. This allows the team to focus on a specific area. It is common these days to have a core and shell phase and then a build out phase. The deliverable packages are all inclusive packages. They have their own set of details, sections, and schedules and are typically submitted separately from each other. This type of deliverable package is a great place to look at linked models. This way the overall sections and details for the core and shell stay in the core and shell model and the details and sections for the build out stay in the build out model. This type of documentation separation will prevent issues later in the project. In any large project you will find that the project is divided into packages. This makes it easier on everyone in the process. If you need to divide you Revit file into multiple files look at these deliverable packages as a place of

model division. You will find that when it comes to deliverable packages the contractor will be in alignment with the designer. This is because of the natural flow of work. The contractor is going to have a team of people working on a specific area of the project at a time. If they can have a complete package for that area they can work more effectively. In larger projects were the contractor and designer are not working together you will find that the contractor often reorganizes the CDs to match the construction process. If you work together that construction process will give you perfect locations for model separation.

Of course all of the conversation about links does not mean much with out a key piece of information. When we talked about worksets we went over the how much RAM it takes to load a Revit file. We used to the rule of 20 to understand how much RAM it takes to load a Revit file. With a link the rule of 20 is not the same. In Revit 2010, 11, and 12 a link used a 1 to 1 ratio. Meaning that if the link was 100MB it would take 100MB of RAM to load that as a link. We started to see that change in Revit 2012 and now 2013 links are taking more like a 5 to 1 ratio. This varies a great deal based on different settings you may have turned on. For example turning on "Room Bounding" on a link will increase how much RAM the link uses. It also seem that as you pull more information from the link in schedules the more RAM the link requires. This is very logical in the fact that, as we use more and more information from the link the more RAM will be need to load it. The challenge of course is that the primary reason that we divide the models up and use links is to control the file size we are currently working with. If your model separation is based on deliverable packages it will reduce the amount of data you need to pull from the link. As a result the amount of RAM needed for the linked model will be reduced.

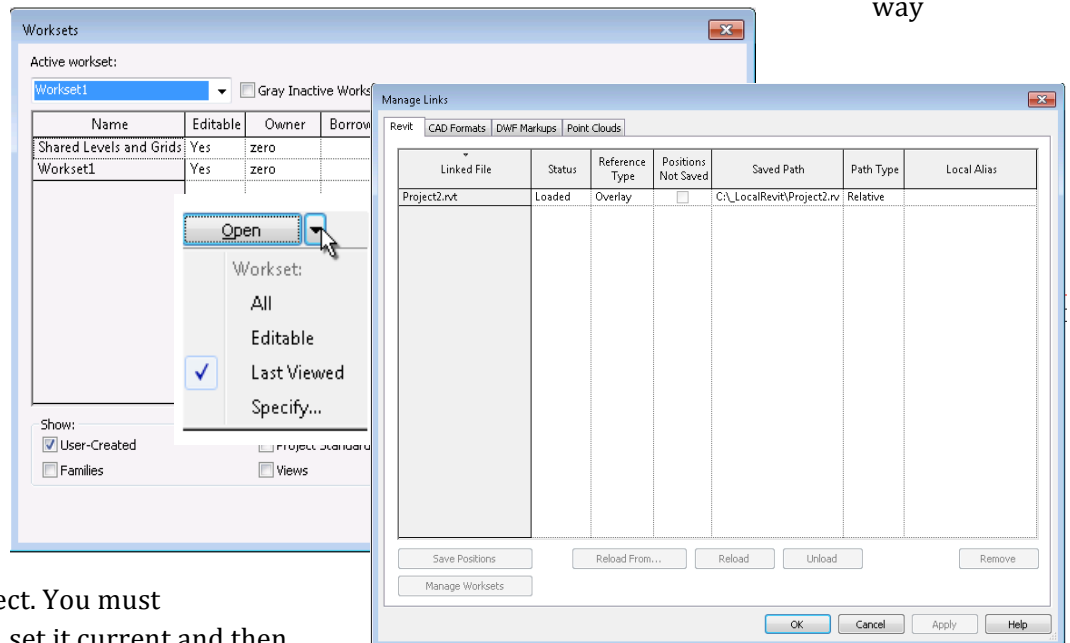


### **Worksets and Links Together**

Together worksets and links are the keys that can unlock success on a mega Revit project. As we talked about in the worksets section proper use of worksets lets you control how much of the model you want to load and work on. In the links section we talked about with the proper use of links we can further control the amount of information loaded. Together worksets and links give you total control of what is loaded in a session of Revit.

The very first rule about links is that you always link on a specific workset. Whatever workset is current when you link a model is the workset that link is on. If you close that workset the link will disappear. More importantly the link is unloaded from RAM. You will also find that information connected to the link will disappear with out error when the workset is closed and will reappear when the workset is opened. It you compare this to the load and unload tools found in the manage links dialog. When you unload a link you will often get errors from the object that are using the links and not all information will return when the link is reloaded. By having the link on a workset you will have a safe

way to turn on and off links without causing errors or issues with objects that use information from that link. We Highly recommend that you create a unique workset for each of your Revit file links. It is important to know that moving a link from one workset to another does not result in the desired effect. You must first create the workset; set it current and then preform the link commend.



When you use the link command you may notice that the open dialog box is used when you are searching for the file you want to link. If you look closely at the open button you will notice the same arrow that when selected reveals the same workset options we talked about in the worksets section. What this means is that you can not only link a file but you can specify what worksets from that file you want to link. This simple little button allows you to link a subset of another file. If you combine this with your workset strategy you can overcome what we typically refer to as no mans land. This is the space between to linked models when you often need some elements from both models but you do not need all of the information from the linked model. A common example of this is a project with a tower and podium. The transitions levels were the podium and tower meet are what we call no mans land. You do not need the overhead of all of the tower model, but you do need the lower levels that interact with the podium. If you used a workset strategy when building the tower model you can simply use the specify worksets tool when linking the tower file. This way you have the tower linked but only the levels needed for the transition. It is not uncommon for this transition area to be a separate model with elements from the other two models linked to it. A word of warning if you create a no mans land model you should make every effort to create a separate documentation package for that area. This is one area that some coordination with the contractor is

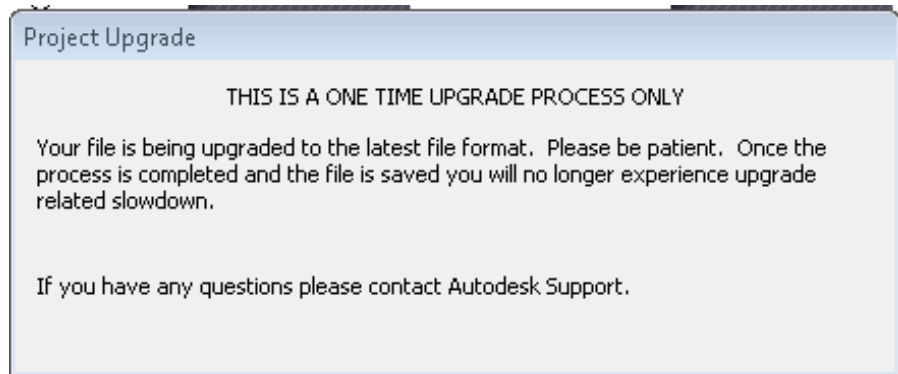


in order. It is not common for this area to have a separate documentation package but it will be very helpful to the modeling effort if it is.

A secondary benefit to following this overall process comes in when you upgrade your projects. If you have created worksets in your project a trick to upgrading your file is to close all the worksets during the upgrade process. This will greatly reduce the time it takes to upgrade the file. In some cases the upgrade time can be under 10% of the time it normally takes.

Also if you upgrade a file that has links the process temporarily upgrades all the links as well. If those links are on a workset and that workset is closed during the upgrade

process Revit will not do the temporary upgrade. We have seen projects that took days to upgrade not using the close worksets process being reduced to just a few hours.



## **Protecting Revit Objects**

When you talk about protecting objects in Revit different users will think different things. What we will be looking at in this section is the protection of objects from other users accidentally moving or deleting them. Some objects are more prone to this than others. On a large project the simple deletion of a floor object can cause days worth of rework and the effort to delete it can take seconds. Some objects just seem to be magnets to accidental selection. Once they are selected then the next command can be devastating. Floors seem to be one of the most common of the accidental selection process. You are just deleting some walls in a small area you use the crossing box to pick a few wall in one selection and then you hit the delete button. You go on working like everything is perfect. The challenge is when you did that crossing selection you unknowingly selected the floor and when you hit the delete command you deleted that floor. Your team may go on for days before they open the sheet where you have all of the slab depressions and openings dimensioned. Of course all of the dimensions were based on an object that no longer is in the model, so all of the dimensions are gone. We have seen this happen on several projects and the floor is not the only object this happens with. We have seen this with roofs, in-place families, grids and links. Of course this can happen with any object but these larger objects that cover large areas of the project seem to be the one that cause the most pain. This brings up the question of how do we protect these objects?

## **Workset Management**

The most common way to protect an object is to put it on a workset and then take ownership of that workset with a user that does not check that workset back in. This uses the workset tool as it was

originally designed. You create a workset place objects on that workset and then take ownership of that workset. If you do not check that workset back in then the only user that can modify those objects is the user that has ownership of that workset. Most companies do this by creating a fake username like ADMIN taking ownership with that username and then changing their username back to their normal name. This way if you want to modify those objects you must first change your username to ADMIN then open the project make the modifications save the changes without checking the worksets back in. Change the username back to the normal and then open the file again to start working. This process does a great job of preventing those objects from accidental modification. The challenge with this process is on a large project that takes 5 minutes to load you may be spending 15 to 20 minutes making a very simple change. Unfortunately with Revit you cannot change user name on the fly you have to stop and restart Revit. Some people over come this with a second session of Revit with a different username but a large project load one session of Revit with the project loaded it tough a second is often impossible.

### **Design Options**

A less common and somewhat controversial concept is the use of design options to protect objects. The controversial part of this concept is that it uses a tool in Revit for something that it was never meant to do. This idea came to us one day as we were looking for a better and faster way to protect objects. In the middle of the conversation a user asked for help saying that they could not select something that they needed to edit. When we got to their desk we quickly realized that the objects they wanted to edit were part of a design option and after coordinating with the designer we showed them how to make the changes to the object. It was not until later when we started talking about how to protect objects and we said if we just had a way that we could draw objects protect them so users could see them, dimension to them and even place objects based on them. At that a light bulb went off over our head. That was the exact situation the user earlier that day had encountered. Once an object is in an option set you can see it, you can pick it but you cannot select it unless you first activate the option. The other nice thing about options is if you have an option set with only one option then that is the one you see in all views by default.

With this newfound concept we started testing putting object we wanted to protect in an option set. The results have been spectacular. We have been able to protect all types of objects without any real negative issues. The biggest issue we found is that you need to name the option set in a way that it does not get confused with other design options. We now create an option set called protected objects and then move the objects that we want to protect to the option. In some cases we have created more than one to group object types together. We have also used this to protect links as well.

We generally do not like to use tools in Revit to do things that they are not designed to do but this is one case were it works and it does not seem to case any other issues.

## Families versus Groups versus Links

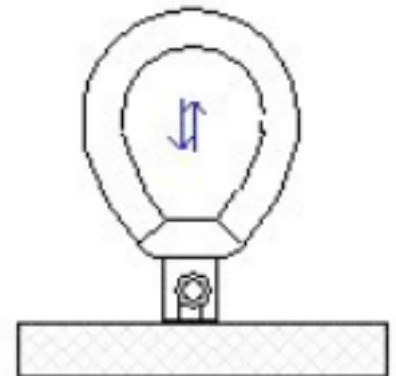
We often get questions about what is the best tool to use. When it comes to repeated objects in a project this is often a conversation about families, groups and links. In many situations you can use any of these tool to complete the process. Like most things in Revit on a smaller project any option typically works. One might be a bit slower than the other but over all not a big deal. Unfortunately on a large project that small difference in performance can easily translate into being able to open your model or not.

### Families

In almost every situation when dealing with repeated elements in a project a family is the right choice. Below we have list several rules about making families that we recommend.

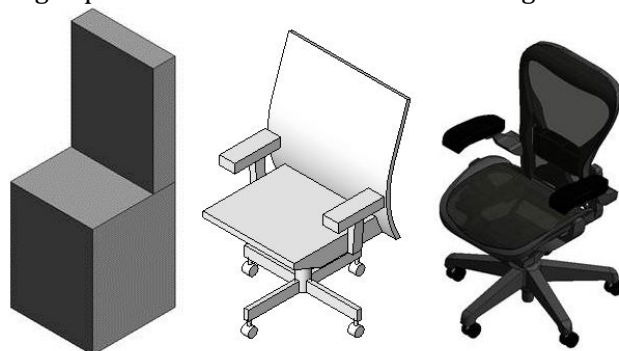
#### Rules of the Family

- Mastering families in Revit is critical for a successful mega project.
- File size
  - Try and keep your families under 2MB.
  - Although keeping the file size of each family small is important, regeneration / processing time is even more critical. Make sure you test your families for regeneration time.
    - Load your family into a blank project and copy is 3,000 times on level 1 and again on level 2. Compare the time it takes to change from level 1 to level 2 with and without your family copied 3,000 times. If the family was created properly you should have very little impact on speed.
- 2d Representations
  - Plan
    - When you have a family that is created with 3D geometry only Revit has to look at each instance of that family and process how it should look when you open, zoom or pan in a plan view. This causes Revit to slow down when switching between views. By simply adding a 2d representation to your families and turning off the 3D graphic in plan view Revit's performance will improve greatly.
  - Elevation and Sections



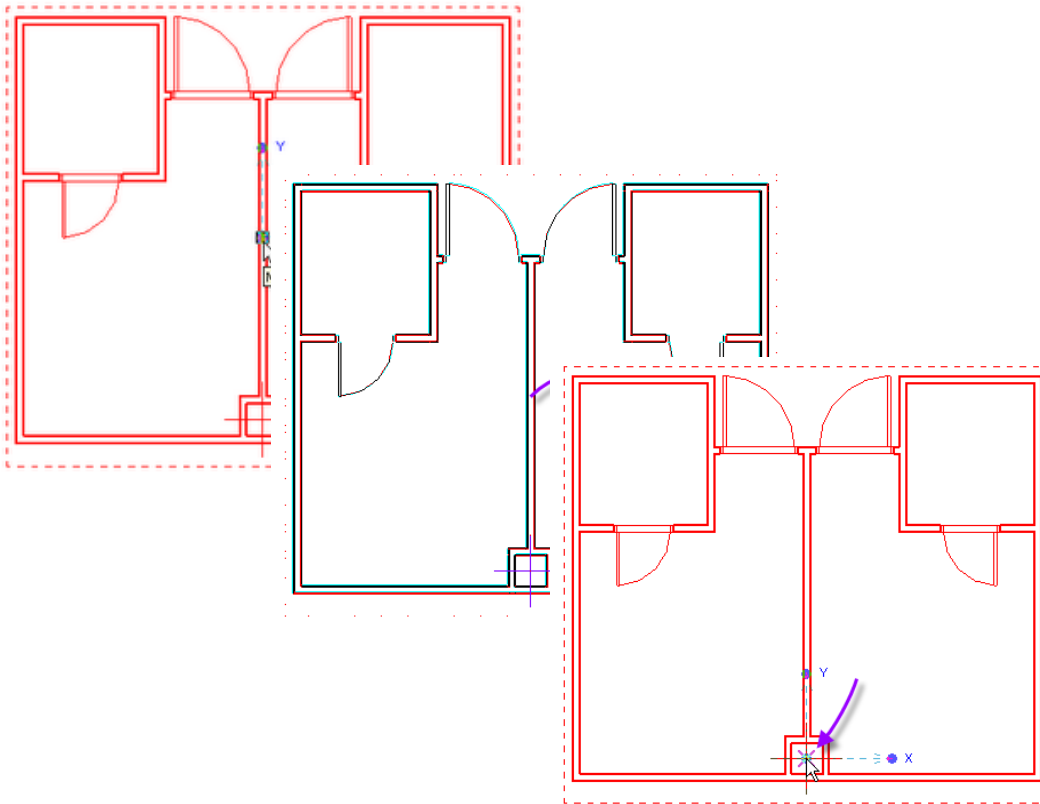
- 2d Representations in sections and elevations can also have an impact on performance, but not as dramatic as plan. This is simply because you are not looking at the whole level when you look at a section or elevation.
- Hosting
  - Some Families have to be hosted. For example a door must be hosted in a wall. Other families that do not affect geometry of the host should be face-based families. This is important on a mega project because the object that you want to host from may be in a linked model. A wall hosted fire alarm family will not work if the wall is in a linked model. A face-based fire alarm family will.
- Cutting
  - Avoid cutting the host object. As an example a can light fixture is a simple family and even though it maybe copied several thousand times on a level, it is only when you cut the ceiling that performance will be dramatically impacted.
- Level of Detail
  - Avoid modeling small details in 3D in your family. Anything smaller than 1" should be handled with 2d symbolic lines. Use the visibility detail level controls to manage when the 2d detail lines show up.
  - Model what you can see. It is easy to go overboard when modeling families. Think of families as what you see from the outside. Modeling the interior structure of objects is typically not needed. You can draw those details when you cut a section or detail with 2d lines. Only put into the family elements that you need to see in different views of the family.
- Limit arrays and all in one families

In-place families are often over looked as a way to create repeated elements in a project. In some cases when elements are based on other objects in the model in-place families are the right fit. It is very important to understand the in-place families do not have the same program level strengths as families. They do take more space and making copies of them is not as efficient as regular families.



## Groups

When dealing with repeating object in Revit the Groups tool is often looked at as an option. The challenge with using Groups is that most users think making a group has the same or similar benefits to creating a family. The reality is that groups in many ways cause files to get bigger and become slower. So although groups can be a very useful tool to grouping objects together for making copies, leaving the groups in place long term can have a negative impact. Working on a hotel where you have a unit that is repeated is a place that groups often get used. The challenge is if you use groups like you did blocks in AutoCAD your performance of the tower will suffer greatly. The strategies for this situation varies greatly based on the design. In general if you do not create groups based on the unit and look specifically at the objects that are repeated from unit to unit you can cut down on the number of groups and create a more database friendly model. Just remember that groups are not families and that groups are not the same as AutoCAD Blocks. Keeping the group limited to the bathroom and repeated elements. The demising walls can stay as part of the level plan. Also do some testing with your group before you start coping it to all the locations. Make sure the file size does not get out of control when you copy the group through the overall building.



## **Links**

The use of links to create repeating objects is something that we typically only see in a unit-based design. Hotels, Condos, Office Complex any design that has a base unit of a room or groups of rooms that are repeated throughout the design. For those of us that did unit based designs in AutoCAD based applications the links approach is one that looks and sounds the most like the Xref systems of the past. The challenge is that the links system works great for small to medium sized projects, but completely falls apart on large projects. If your project has 10 or less links with 5 to 10 copies of each, Revit performs ok. When you start exceeding that number Revit really starts failing very quickly. The links concept simple put, cannot scale. If you are looking to use the links concept we strongly suggest mocking up the system and testing it before you make it part of your process. The other challenge with the linked process is it is very hard to move away from it later in a project. It seems like it would be simple to just bind and explode the links but it is much more complicated than that. This is clearly a place where your past AutoCAD experience does not translate into Revit. Links are something that should be used very carefully.