AUTODESK. UNIVERSITY

AS227233

DeCAD Your BIM: Moving Beyond CAD Pitfalls in Revit

Michael Freiert
NewStudio Architecture

Learning Objectives

- Learn to identify workflows that are not BIM and Revit appropriate
- Find solutions to present a comparable workflow to end users
- Learn how to help other users adopt more-efficient practices
- Discuss ways to helpfully explain to stakeholders and designers why certain methodologies aren't the best solution in Revit

Description

BIM (Building Information Modeling) is a powerful workflow that's often sold as a massive time savings versus a comparable CAD workflow—but many firms see an increase in their project time when they migrate to it. There are a lot of CAD practices that, while very efficient in AutoCAD software-based projects, are a terrible idea to maintain in a Revit software-based project. This class will look at some of those common pitfalls, and examine ways to maintain the intent of the existing workflow while harnessing the "smart" features of BIM. This class is targeted at smaller to mid-size architectural practices that may not yet need everything smart and optimized in their offices, but are hoping to see some of the time savings and improved accuracy that BIM promises — but does not always deliver.

Speaker

Michael Freiert has a lot of hobbies, has had a few professions, and is a strong believer in multidisciplinary learning. He’s worked in ACAD since R12, and Revit since 6.1. Michael has learned many things over the a few decades by being paid to play with Legos, design clothing, work in music, theatre, special effects, explosives, woodworking, machining, sewing, cooking, and gardening that have influenced his design methodology, choice of tools, and creation of his own tools. Michael is a big believer in using the right tool for the job, even if that sometimes means using something in unintended ways, or making that tool himself. For the last few years he’s been the BIM Manager and a Job Captain at NewStudio Architecture in the Twin Cities. Prior to that he worked for a number of years as a BIM mercenary on a wide variety of architecture related projects, and is a founding board member of Twin Cities Maker, a member owned and managed non-profit that runs a pretty awesome makerspace.

BIM isn't CAD

Do not try to make something do something it wasn't designed to do, or do something that it performs poorly at. A F350 is not a great city commuter car and a Prius-C isn't going to move that ton of gravel well. Yes, both can accomplish each task, but different tasks work better with different workflows and different tools.

BIM is not merely an installation of Revit - Building Information Modelling is much more than one program. Revit is a solid foundation for BIM, but a full BIM workflow involves more than just one program and model. That does NOT mean that you can not make use of many *elements* of BIM only using Revit, or that you need to use every feature of Revit to be using BIM.

Computer Aided Drafting is simply put, just that. Drafting on a computer. Drafting, simply put is drawing (very accurate) pictures to describe a concept specifically for architecture, how to build something.

BIM is a very different means of communication. It is not *only* virtually building the building. It is not *only* a Model that looks good in a lobby on a pedestal. At the richest levels BIM contains* all of the information about all of the parts of the building, across time, and including theoretical variants. And from that model we can generate the same deliverables**.

> * that's a ***big*** *asterix.* It ***CAN*** contain *ALL* of the information we could possibly want about every element that goes into the building, but it does not have to. Trying to include all the information we could want may be a problem. In fact a very apt workflow may often function more as a reasonably clever model of a building rather than a truly "smart" model. Occasionally, it works out to be a pretty dumb and inaccurate model. And that is just fine for some types of project.
>
> **We can also generate significantly more interesting, well developed and accurate deliverables, incredibly useful information, facilitate design studies, and unfortunately wander down some deep rabbit holes of wasted time (and really cool stuff).

Use an apt tool for each task. If you have a workshop, you do not limit yourself to one hammer. Revit, like AutoCAD, is a very powerful tool that can do a lot of cool tricks. Even without the native integration of Dynamo, a not quite power user can automate information, how it is used and improve document accuracy in ways that a LISP master of 15 years ago could only dream about. Remember that not every 100 square foot backyard deck needs a 300 page CD set.

Why is this guy talking about BIM instead of just Revit? Because BIM is not just one program. Looking back at a CAD workflow we got away with thinking we were only using AutoCAD. In practice however we were using Acrobat, PhotoShop, BlueBeam, Word, Excel, MasterSpec and a variety of other software to support our CD sets and get them out the door whether those were pen plotted or PDFs. BIM has similar relationships with rendering, virtualization, Word, Excel, InDesign, Dynamo, web sites, building controls, proprietary software and internal databases tied into a Revit model.

How smart is your BIM?

How smart do you need it?

Should you use BIM (on everything)?

Probably both not smart enough **and** too clever;

I've got no idea;

and probably not- but yes.

If you are taking this class you probably understand that BIM (even as a stand alone Revit Model) can handle and manage a lot more information than you probably need. If you have used Revit for more than a couple of projects you have discovered that a lot of information you do not utilize is contained within content you have used. To be very frank, I have not found the need for a number of native family parameters that are included in the templates, but I can understand why others would want them. I have also added a number of my own parameters that drive elements within my own and my offices' workflows that the native, and third party standard writers, parameters do not cover. I can also see the strength of building a template or family that will nearly fully propagate a full prototypical model based on a client prototype **IF** the client required enough of those specific prototypes to be generated per year to offset the development time of that prototype.

Which brings us to the real second question- how smart does your BIM need to be? Are you saving yourself, your firm and your client time? The answer to that lies in *how* you implement BIM. If you are spending thirty (30) hours building automation to develop what would normally take forty (40) hours of annual of work, that is a very different answer than if building the automation replaces twenty (20) hours of annual work and will need twenty (20) hours of maintenance annually. Thinking through that return on investment and talking through implementation options with your stakeholders is important. How much information do you currently use? How much information are you maintaining that you are not using? Look at those two minute tasks that happen on every project at every step, can you cut that to five seconds? Can you cut a 5 seconds of scrolling that happens many times a day down to less than one?

So I should not use BIM on anything? NO. Use it on everything*, that is why we are here today (or you are reading this), but in implementation of BIM understand how much information you need and what you need to model. Way back in the days of drafting we did not re-draw related details and did crazy things such as note some details as "SIM" or even "REV" for similar or reversed elements. We, as a profession, carried that nomenclature on into CAD and we can still use such amazing anachronisms in BIM documentation. We can also use shortcuts and drafting methodologies but still model it quickly in Revit. We do not need to throw out the useful shortcuts and practices, **as long as they are still actually useful.**

*NapkinCAD, AutoCAD and other drafting and modeling solutions do have their place and are very apt as the correct tool for the certain projects, but we're looking at what BIM can do and trying to avoid less useful legacy workflows.

Finding the Problems

If you are reading this you are probably aware that your BIM workflow has some inefficiencies. Mine certainly does. BIM is a new workflow, and no, we did not always do it this way. There are a number of common broad groups of problems.

Awkward or Annoying tasks

Look for tasks that you or your users find frustrating. They may be tasks that you have not automated, poorly automated tasks, tasks which require too many clicks, or are just annoying or tedious to manage. Revit workflows do not have to utilize the stock Revit tool. The stock tool is great ONLY IF you use it as intended. Best practices are only best if they work for your team and project.

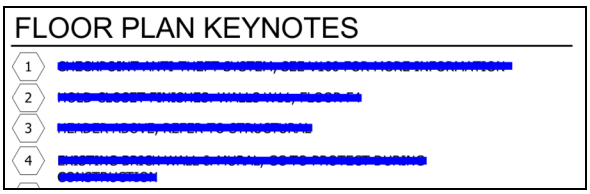
> Keynotes - Revit Keynotes tool is designed as a specification flag tool to be managed centrally by a spec writer, not as a classical “Keyed Notes” tool. An Annotation Symbol and note block schedule or Dynamo managed Annotation Symbol family and schedule *may* be a much more elegant solution for some offices as numbered notes within drawings.

Legacy Graphics

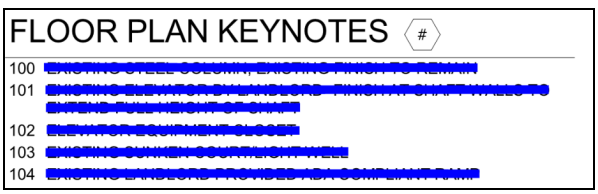
Printed Construction Documents need to communicate effectively. Each office, and often each architect, engineer, or client has their own well reasoned ideas about how to best communicate the information they need or want contained in the final documents.

We can very closely mimic the graphic standards that are well established, but changing to a new workflow is a great time to revisit portions of them that are not easy to integrate with BIM.

> Keyed Notes again - One Architect on the Revit migration team insisted that each keyed note in the schedule on each sheet would have not only the number adjacent to its text, but the number *MUST* be contained within the keyed note symbol. The keyed note annotation symbol contained the ability to show its text adjacent to the symbol (so you could check it on sheet without hoping in and out of the type editor) and that Architect opted to manually create a keyed note legend for each individual sheet, including duplicate tags for “not used” numbers on particular sheets as appropriate. Everyone else just created a schedule and was done. That user spent hours on double checking each note on each sheet, while the rest of the office gradually realized how much faster and easier it was to schedule it automatically - we included a generic # tag next to the schedule title. A minor change to the graphic standards resulted in a HUGE time savings.



Manually placed notes in Legend



Noteblock Schedule

It's *ALIVE*

CAD templates took years (if not decades) to build up. Even a well developed and thought out Revit template or BIM environment needs maintenance and upgrades. The first CAD project in your office took ages to detail everything, and those details were then saved, and reused, and adjusted and put in a library (even if that library consists of asking one of the senior team members what project had a particular detail hidden in it and copying that in - the card catalogue may be inefficient, but you are still copying in useful data).

You do not have to convert every detail in the library to Revit now. Do prepare to fix things - build in expansion capabilities, and plan for all the bells and whistles so you do not have to start from scratch. Look at all of your details and figure out how you are going to sort, find and update them. That may be naming convention, a "library file" or a third party content management add-in. You can change that, but thinking carefully about how your end users are going to gather that content into each project, how they are going to edit it, and how you are going to update it is critical. If you think you might want to be able to do something, build in that functionality now.

You are going to start from scratch again anyway. Sort of. You can copy in stock details from CAD, (remembering to use a dummy file to convert them to Revit elements) and once you have built a fancy new Family, it is often much faster to rebuild it cleanly than it may be to try to clean out all of your mistakes, so it is not truly from scratch, but as you are cleaning and upgrading, make sure you actually clean and upgrade.

Naming Conventions

CAD naming conventions do not translate well into Revit. Think about different object sort orders and how different user groups will want to access them. In CAD we sorted things by layers relative to how we wanted to display them, and named them in related hierarchies to make them easy to manage.

Revit deals with things very differently. We can use family category, sorting by family name, but that is not always ideal. Depending on your practice, it may be easier to find certain elements by a hierarchical description, by a specification section, or an internal code, and how you choose to name that will really depend on how your users are going to find and identify it.

Understanding how your naming conventions impact user interaction and ease of use can be significant. It is a new workflow, you can rename things, or add a prefix or suffix to clarify how users will locate or identify content.

What is in your template (Starter Project)?

Assess what is in the beginning state of each new project: Does each new project start with all of your stock sheets? Should you have all of those views on it? All of your families?
It is often faster to delete unused data than it is to import it later, however, if you can import as a batch that may be faster than deleting a lot of individual items. If every project has users adding and subtracting information from it, that can add up to a lot of lost time.

With good linked file management (or Sheet Sets) one could largely automate what sheets started in a project in CAD and we can certainly include stock sheets in Revit, or generate them as needed with an add in based on a family of project type. Do you need all of those? Is it faster to delete them or make them?

Do you need every content family loaded for every project, or just a flexible placeholder that might be swapped out later on when details are being ironed out and it will be changed anyway, or can it stay generic forever? Will including one unhosted toilet with standardized connections (if any) be easier or do you know it will almost always be one of three models and it will be faster for everyone if they are just there from the start?

Listen to the users.

They may be very very wrong about the solution, but odds are they are telling you something is annoying them. Figure out if the problem is workflow, graphics, GUI, hardware, etc. Users may want legacy workflows, but they may not know why. It may be because legacy workflows are what they are familiar using, because they were once taught to always do something "that way" or because the legacy workflows more efficient at getting the right data down faster, or If it is the last you need to figure out how to improve that workflow.

> Remember my Keyed Note Architect? They had a very valid point; the office was used to working one way and the notes needed to be clear as to what they related to. We added the symbol on the title because we understood what the Architect wanted to accomplish, and our document sets were clearer because of it. We also used Annotations instead of the Revit Keynotes tool because (most) users were frustrated at only one user being able to edit the text file and other users deleting notes with elements. They did not know that annotation symbols were a good solution for Keyed Notes, they just knew that they did not like fighting for control of the text file.

Identify Solutions

Certain problems seem to have blatant Revit solutions. Wall types are pretty much always going to be Walls, unless they are Curtain Walls, or generated from masses. Keynotes versus Keyed Notes as Annotations (again). Just because Revit gives you one way to sort things or one name to call something does not mean you have to use it for that.

> Our office does not do medical facility work, so we have a bunch of behind the scenes uses of "Nurse Call Devices" content that wrangle elements in ways that our day to day users never need to worry about. We needed a way to sequester that information from the day to day user, but still leave it so that they can edit it without fighting with it. This is NOT how Autodesk intended Revit to wrangle those things, but it works well (for us).

What are you trying to do?

Look at the input, modification and output you are trying to achieve - outside of what you expect the specific task to normally be achieved. Think well outside the box about related behavior that you want to see. Do you want it to be scheduleable? Should it pull information from another element? Should it be wall hosted or can it be surface hosted?

How does Revit want to produce that output (is another tool better)?

Revit is not great at editing text (it is much better than it used to be) - is Word, Excel or InDesign a better solution for a large text block? Do you need to push that information into or out of Revit? Can you move just part of it? Do you need to edit that text or is it always the same?

> Our office often uses spec on sheet, and has a pretty large set of code compliance notes and calculations on sheet in Revit. We have played with the text enough to get the

formatting easy for the codes, but can use the auto numbering and copy/paste of Word. We use Word with style management to edit more complex text such as specifications, and InDesign to get those onto our PDFs, but our code information is formatted in model code specific blocks of text to be edited natively in Revit. Side note, we use other solutions for larger or more complex project Spec books.

Revit does most of what I have ever needed for rendering, but that does not mean things do not get cleaned up in Photoshop, or that Enscape or InsiteVR and similar software are not amazing tools that can deliver well beyond what Revit by itself can.

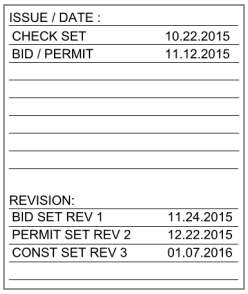
What can you use for Revit Input?

Are you using customized families? Pulling fixed schedules in from Excel with an add-in? Are you managing family data with Type Catalogues? Are you using Global Parameters? If you are using out of the box content and not looking at what you need and can use to automate or ease portions of your teams workflow you're missing huge potential efficiencies. Even the out of the box content has great possibilities there.

Is there a creative Revit workflow you can use?

Yes. Yes, there is. I have not yet found something that we actually needed to do in Revit that I could not find a way to accomplish. Often it was somewhat out of the box, and occasionally a request was something that we *should not* do in Revit because it would cause a lot of other problems.

NewStudio uses "Issues" and "Revisions" in two *separate* Revision Schedules - by breaking the Revit Revision Sequence count at (9) "Issues" we can force "Revisions" that are Sequence 10+ into a fully separate schedule. It's not exactly what Revit intended, but it works seamlessly. How do we do that? It involved copying elements from older family templates and pasting them into themselves and from there it gets arcane. So yes, you can make Revit do some crazy stuff if you get creative. That can be a very good, or very bad thing.



ISSUE / DATE :	
CHECK SET	10.22.2015
BID / PERMIT	11.12.2015
REVISION:	
BID SET REV 1	11.24.2015
PERMIT SET REV 2	12.22.2015
CONST SET REV 3	01.07.2016

Can you use an Add-in or Dynamo?

Renaming, Duplicating, Quality Assurance and Standards compliance. There are a lot of ways you can accomplish certain tasks. There are a LOT of free Add-ins plenty of pay Add-ins and a lot you can do pretty easily yourself with Dynamo with relatively minimal training. Figure out if an Add-in is worth it by looking at the hassle factor - does it actually save time, or would training on the task take more time? In general, things that are repetitive, or seem like they should be able to be automated are good candidates for finding an Add-in or creating a Dynamo script to perform. Remember, for your end user even if it is not a task they wanted to be "easier" if it cuts down on troubleshooting down time for the team, it might be "easier" for everyone even it it means an apparent extra step for them.

Help folks get on board

Everyone likes to do things the way they are most comfortable with or the way they have previously worked. Sometimes that is not ideal.

Show them an easier way

It actually has to be easier. Really. Whatever they complained about, your solution needs to be fewer clicks that are closer together, with less scrolling.

Some friends of mine developed a really awesome Add-in tool that is pretty affordable for what it does. I do not use it, and do not recommend it for one reason: It takes too many clicks. It CAN do (almost) everything that I want in that kind of content, but the amount of time it takes to train in a user and maintain with office standards, and then nonstandard project items it is not a practical solution for an office of our size. Instead, I built internal content that is highly configurable by the end user, and they just have to load one of four families to get almost anything their crazy hearts desire in that sort of content. It took half of a lunch and a learn Revit update to get two-thirds of the office trained. The other third of the office picked it up themselves. It took me roughly sixty (60) hours of development time and pestering a couple of my savvier users to do some beta testing, but for what we need, it generates all of the information necessary, for less than the cost of the Add-in. Folks saw an easy to configure family and ran with it. If you are getting push back you may need to simplify it:

If easier is more complex, break it down

When converting to Revit, I discovered an office wall naming convention that was at best loosely followed. At this firm, each Project Architect copied in the wall schedule from their last CAD project and modified it as needed. Each project's partition naming was similar to the "office standard" that was lonely but well documented in a dusty folder on the server. I used that standard as a jumping off point for a new Revit standard. The new partition naming convention was immediately unpopular. It was "too complex" to learn and not what each individual *thought* was the office standard. Folks got on board when I showed them the old "official" standard, and how I nudged each part of it to a more coordinated formatting. We talked about each specific element of the naming convention, what it did, and how everyone knowing that an "A" wall was a basic gyp stud gyp would be really easy to jump in and help on other projects. We worked through the full naming convention step-by-step until it all made sense to the team.

By showing everyone the building blocks of the new system, they were able to see that by everyone using the same system they could more easily inter-operate and they did not need to build things whole cloth over and over again. They were excited to be using a more complex system, *once they understood it.* Showing them a cheat sheet made it seem more complex than it was. I just had to "trick" them into learning it in pieces rather than showing them the whole shebang at once. Focus on the time savings and simplifying one pain point at a time, not the annoying part of learning something different.

Relate it to the old way

Do not name something identical to its replacement, but relate it - IF it is related. The new partition naming convention was similar to the old one, but there was no danger of CAD wall types being copied into Revit. Also, no one was really "used" to the old formal office standard so the new names being similar helped a lot with that issue. It seemed like I was not imposing something new and unproven. If I needed to start from scratch I would have approached the rollout more as "x" replaces "a" and "y" replaces "a1" - it is the same, but by using X and Y we can do this cool new thing and it is easier when we make schedules.

The Keyed Note use of annotations however was a big headache to have adopted. Not because most users did not like the functionality (for our office workflow) over the Revit Keynotes tool, but because they were certain *that* specific Revit tool had to be used for *that* specific task. By explaining that different drafting practices used the same term for different uses of similar notation styles the users got on board. Telling a metal fabrication and an architectural drafter to each fillet something will result in very different but similar drawings. In Revit, like in CAD, there are LOTS of ways to get something done, the trick is figuring out which is the "best", not just in this five minutes, but for the life of the project.

It's a different way to communicate

Remind users that the PDF is not the end product. The builder's understanding of that, and the building are the end product. BIM and Revit can produce a lot more than just a PDF set or prints. You do not always ***deliver*** more than that to the client, but that does not mean that you do not *USE* more than that if you are starting down even a limited BIM workflow. Just because you are putting a plan on a sheet does not mean that said plan contains ***only*** the same information as a 2D CAD plan. That floor plan is intrinsically linked to the elevations, sections, RCP and other views that relate to it. The floor plan view is NOT same CAD background copied into the RCP, but the walls are literally different parts of the same elements showing up in each view. BIM is not communicating in lines, but looking at lines that are the edges of physical things. Even if you do not drop one 3D view onto a sheet, and never use any 3D views, even compared to 3D drafting tools like AutoCAD Architecture (Architectural Desktop) it is a significantly more data rich and environment.

At a minimum, every view (except for drafting views) really is in 3D. It is easy to think of a floor plan as merely that, but when you are fighting with View Regions to get an odd height window to show up, or not show up, in a plan you are starkly reminded that the view has real depth.

Merely turning over a PDF CD set is fine, but the ability to spit out a cropped 3D PDF to the field for an RFI can be a huge time saver.

Listen. To. The. Users.

Yup, they are ***still*** *probably wrong* about their solution, but maybe not. Can you give them buy-in to a solution that helps everyone?

You've been to AU. You have fancy certifications. You have got $InsertQualificationHere and you have a brilliant solution to $TheProblem. A solution which all of your users hate. They know something is wrong or maybe just different, and they do not like it. Do not use an "appeal to authority" fallacy to defend your brilliant ideas.

If you have ever read the manual for an 1980's VCR manual you understand that technical writing does not necessarily communicate the intended message to the user. Similarly, a user who is frustrated with a task may not have the technical background to understand whether they are annoyed specifically with Windows, Revit, their graphics card, the number of clicks it takes them to do something, the amount of mousing, or that you have organized things by MasterSpec instead of Revit Family Template Class, or that old family used a legacy feature and they cannot control that setting anymore. *The user knows something is frustrating and are doing their best to communicate that* - you likely need to listen less to the verbatim "X is *BAD"* verbiage and more to the underlying systemic message that something around "X" is wrong.

The last bit was intentionally long to make you sift through seemingly extraneous information to get to the concept - that sometimes "helpful" information is not helpful itself, but there may well be useful content buried in the message.

Convince the doubters.

Not only end users but firm leaders need to be on board with adopting new workflows.

Help people see the benefit

You are going to have to do some math and analysis of workflow tasks. Is the promised average project hourly time savings not showing up? Explain where the hang ups are. Explain what added workflows have cropped up. If you never did renderings in CAD, or you are wasting time modeling a project in both SketchUp and Revit that is a time sink you need to account for.

Years back I worked at a firm where only a few of us had a second monitor. Each year those second monitors were borrowed for one week for use at a convention. I noticed that a routine task that I managed weekly was taking a lot longer than normal. I built a routine to count keystroke interval and mouse inputs. The week after the convention when I had my second monitor back I ran the same task, sure enough it took me forty percent (40%) as long for the same number of inputs. The next year I ran the routine on everyone with a second monitor for the two weeks before, during, and the two weeks after the convention, and the worst performance boost was only twenty percent (20%). Factored out over time, a second monitor for everyone in production paid for itself in about four months and was then pure efficiency. Showing the partners the math made it easy for them to spend the money.

Your task is to show users that they are doing less to get more in the workflow which you have assembled for them. This means you not only need to understand what they were doing, but how it is better under the new workflow.

Low hanging fruit on that tree is to use automation and good practices to reduce redundant information errors.

> Returning again to that Keyed Annotation example, by using a schedule users did not have to manually place notes and any note placed showed up in the schedule. Users can see immediately that a note number is not used or that a door is not scheduling because it was added in the wrong Phase.

The more you automate things and users follow at least office wide good practices the easier that is. There are *always* exceptions to the rules but if those can fall within the automation process it makes it easier for everyone, which is our goal. Simple things like adding stock

schedules to your template for “Print” (what’s needed on sheet) and “working” (everything for trouble shooting) can add up to huge time savings if people (or eve need those.

Look for quantifiable benefits - count those Clicks, Keystrokes, User minutes on a specific task, and look for tasks that can be automated and do not need user interaction anymore.

Manage early expectations

Until you have a solid road map do not expect time savings. Adopting a new tool or workflow takes time.

I have found that for an experienced Revit manager who has built a few templates and is jumping in blind, a solid eighty (80) to one hundred sixty (160) hours of time is not unreasonable to start a new office template. Time will be spent on messing with title blocks, figuring out what needs to be included in the template, why **THAT** was included in the CAD standard, and changing deeply nested system families to satiate firm graphics standards. It may not all happen by itself as things can be implemented across several projects (if the firm is willing to compromise graphic standards as things are developed). The first project, or three, takes double to triple the normal time, and the next half dozen usually take one and a half to two times as long as a typical similar project, by which time you are hopefully starting to hit a similar hours to deliverable time*.

That is a huge non-billable time sink. That does not mean that you are not going to see time savings or other benefits going forward. I have seen workflows drop to thiry percent (30%) of the previous CAD workflow for deliverables. That six weeks of non-billable development paid for itself handily within the first year of use. That was an absurd outlier, but your firm has spent years developing a CAD library that it uses on each project. It is not fair to expect that a decade or three of content creation is going to magically happen in Revit. Even if you automate the importation of the CAD details, you ****REALLY**** want to run them through a Quality Assurance/Quality Control protocol so that your new workflow only has vetted content in it - so you do not have folks copying in garbage from random projects.

Mission Creep

Just because we can does not mean that we should. Can we automate it? Can we bill for it?

That thirty percent (30%) of the CAD workflow I mentioned above *only* added one 3D view, and a sheet of automated details to the deliverable sheets versus the CAD workflow. It was a simple workflow that allowed for a lot of sheets and views to be pre-configured, and most notes and details to be minimally edited. If that firm was going from small single family residential to commercial tenant improvement work and back again it would never have worked. What we recognized in building that template was that the firm did not need anything fancier for deliverables, and adding bells and whistles was not going to add sales.

If switching to BIM means easier renders that can be used by clients for marketing, or even 3D clay render views to help a client visualize a concept that can be a huge reason to add scope even if it is not directly billable. Recognize though that those new features time to accomplish will come out of potential time savings.

Adding a BOMA schedule to a commercial retail project can be a simple value added proposition, or a billable service depending on how it is presented to a client, and how much work it takes you to accomplish.

If you add a lot of new features without new revenue, and it takes more time, that can be a problem.

Give users Buy-in to solutions

That user with the bad ideas? Thank them. Even if you did not use their solution. They showed you that something needed fixing or inspired you to look for different solutions. If you are lucky, they will appreciate that you have made a point to try to understand their problem. This can be as simple as saying at an office training session, “User brought this problem to my attention, and you would not have this fix without them.” They may have had nothing to do with finding the solution, but it helps them to see that even if you were not using their, possibly impractical, idea that they contributed in a constructive manner. It makes it harder for a user to complain about a new workflow if they helped to craft it.

Make it relatable

BIM is a new tool. Find a way in your own voice to explain that it is NOT just fancy Computer Aided Drafting. Try using analogies to woodworking, sports, baking or CMU vs ashlar stone work, or something else that is familiar to both you and your users.

I hear this a lot, but no matter what anyone says, **we didn’t always do it that way.**

For most of the last several millennia, architecture was practiced with sand, slate, chalk, charcoal, strings, and some fancy sticks. For only a few hundred years have we used pen on vellum, and about sixty (60) years ago we started in on rudimentary CAD. We have just passed thirty-five (35) years with AutoCAD for the bleeding edge adopters.

BIM is not the same as stepping from 2D CAD to 3D CAD. BIM is *literally* ***creating the building,*** which can include all of the manuals and specifications indelibly etched onto every element thereof on a computer instead of in person. We are moving from delivering a 2D technical drawing of a building to delivering a picture of a real building, a picture that we filter down into a technical drawing, andone that can come with a lot more information than a few lines.

Two more things - not necessarily in the overarching bullet points, but important differences from CAD. These may help some people see another side of several items I have attempted to address.

Gain structure.

In audio engineering you have a lot of places to control the volume of sound coming out of a speaker that went onto a particular microphone.
Typically you can adjust the volume at the:

→ channel input gain on the mixer (the little knob at the top of the channel);
↳ at the processor (does all sorts of things to tweak the sound);
↳ at the channel fader (that one mic on the console);
↳ at the group fader (backup vocals or whole drum kit);
↳ at the master fader (the whole band);
↳ at the amplifier or speaker (how loud the speaker should be).

Each of those steps is an adjustment that is closer to the end user, the listener. You want to make the dynamic changes in the easiest to control spot, and mess with the others as little as possible, but have them work with each other and support where you need the dynamic changes. At each step down the options there is a reasonably ideal sweet spot that lets you do more with it *at the next step, and often at previous steps*.

Revit has a similar series of visualization and data management processes that are far more complex than turning CAD layers off and on, or messing with volume. Between Family types, naming conventions, Filters, Phases, worksets and other options, you can control different things in a LOT of nested ways, and there is a trickle down “gain structure” in Revit in how the final deliverables come out and how your users access and utilize content within their workflow.

Get yourself and your users thinking of working from the top down rather from single instances. Instances are important, but they should be outliers rather than archetypes. Both are the right answer sometimes, and if you insist on doing it one way you are bound to be wrong sooner or later.

Revit is a database

Understanding the general concepts of how databases as a whole deal with subtables and the information contained within them will help you to understand some of the limitations that Revit seems to impose, and start to think about ways around those. It’s not exactly just some nested sub tables, but there are a lot of behavioral quirks that will make a lot more sense if you think of Revit files as databases rather than as collections of discrete elements.