AS502305

Federal Agency Stress Tests
Revit and BIM 360 Best Practices

Steven Register
CallisonRTKL

Robert Fink, AIA
Quinn Evans

Fotis Perizes, LEED AP BD+C
CallisonRTKL

Learning Objectives

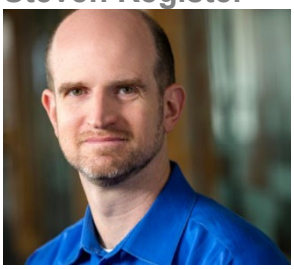
- Develop effective communication protocols for diverse teams
- Implement a model splitting strategy with confidence
- Identify appropriate methods for various design option scopes and scenarios
- Orient and align buildings and point clouds correctly

Description

A Prominent Federal Agency is renovating, preserving, and expanding two adjacent large historic buildings at the same time. The joint venture FORTUS assembled a 200-person, 23-consultant design team for this project. The team went on to develop 40+ Revit models totaling over 15 GB with an additional 4,295 scans of point cloud data. To facilitate such a large endeavor, FORTUS employed common best practices of Revit and BIM 360. In this class we will discuss the best practices that held up, expose the ones that struggled, and reveal newly discovered ones. The best practices we will cover focus on project setup, collaboration and communication standards, dividing up models and teams, alternate design options, point clouds, building orientation, BIM 360 Model Coordination, and QA\QC tools.

Speakers

Steven Register



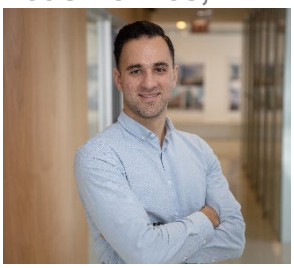
Steven is a Design Technology Specialist for CallisonRTKL in Washington DC supporting multiple offices. Steven has been a BIM Manager since 1999 supporting small Architecture and Landscape Architecture projects to very large multi-discipline campus projects. He is also responsible for creating and maintaining CallisonRTKL custom BIM addin/support programs using .NET programming.

Robert Fink, AIA



Rob is breaking new ground in architectural design, preservation, and the stewardship of historic buildings through the application of advanced technology and building information modeling. He combines his knowledge of building assessments, conservation, and restoration with technology-based skills to evaluate design strategies, coordinate disciplines, and create high-performing buildings. As Quinn Evans' director of design technology, Rob provides leadership in leveraging innovative technologies to better inform design projects. His experience in Historic Building Information Modeling (HBIM) has also helped clients document critical assets, prioritize maintenance and rehabilitation measures, and enhance access to historical records and stories for interpretive purposes.

Fotis Perizes, LEED AP BD+C



With a background in architecture and nine years of concentrated BIM leadership, Fotis is an expert in model delivery, review and coordination to ensure BIM industry standards are maintained on all project deliverables. He leads coordination meetings using Navisworks/BIM 360 for collaborations across multiple practice groups and ensures compliance with BIM execution plans. Fotis conducts BIM model QA/QC on a recurring basis throughout project phases, while coordinating BIM model content and approaches, model exchanges, quality control and delivery for both internal teams and external clients. He possesses extensive knowledge of developing and coordinating Construction Documents and is well-versed in project visualization through 3D virtual construction modeling. His skills are vital to the promotion and implementation of VR/AR throughout the design process.

Table of Contents

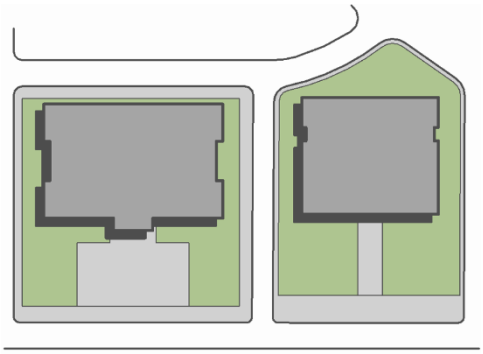
Learning Objectives 1
Description 1
Speakers 2
  Steven Register 2
  Robert Fink, AIA 2
  Fotis Perizes, LEED AP BD+C 2
Table of Contents 3
Introduction 4
  Sources 4
Project Setup 5
  Dividing Models and Teams 5
    Model Splitting 6
    Concurrent Users 7
    Each Link on Its Own Workset 8
  Collaboration \ Communication Standards 9
    BIM Meetings 9
    Tools 10
    Private Cloud Solution 12
  Point Clouds and Building Orientation 13
    Sharing Point Clouds 13
    Entire Building Is Slightly Off Axis 16
    Revit North Bug 18
    Visual Backcheck for Survey and Point Clouds 19
  BIM 360 Model Coordination 21
    Clash Coordination Setup 21
    Process 22
    Protocol 24
    Schedule 26
    BIM 360 Issues 27
    Insights 28
Alternate Design Options 29
QA \ QC 30
  Single Model Tools 30
  Model Checker Effort 31
  Model Checker Report 32
  Model Development Checker (MDC) Online 33

Introduction

This case study will present how well Revit and BIM best practices worked for a very large, complex project for a prominent federal agency.

Before covering the best practices, let us look at what makes this project a good stress test candidate for validating BIM best practices for large projects. Below is a summary of key numbers demonstrating its size and complexity.

- 2 Large Historic Building Renovations\Preservations
- 2 Major Additions (1 for each building)
- 1 New Underground Parking Garage
- 1 New Connecting Tunnel Under Existing Street
- 1M+ Total Building Square Feet
- 300K Square Feet Site Renovation\Preservation
- 2 Architecture Firms (Joint Venture)
- 23 Design Consultants
- 200+ Person Design Team
- 40+ Revit Models
- 15+ GB Total Revit Models
- 4,295 Point Cloud Scans



Beyond just the numbers, a key word in this list is “Historic” which is what made this project particularly complex. Some of the issues a large historic building like this can have are:

- Existing elevations across single floors can vary significantly (more than 6”) from one end of a large building to the other making aligning to new addition levels challenging.
- Deciding how to display important, historic objects in BIM needs careful attention:
  - Which historic objects are modeled and then to what LOD
  - Can point clouds be used in views and how will that affect performance
  - How many linked images and pdfs can be used before performance is significantly affected

Sources

The best practices presented are based on decades of large project BIM experience by the BIM team and Autodesk documentation. In the early days of Revit, Autodesk documentation was an object of our jokes. But today it is far from that and a great resource; and it aligns with our experiences very well.
These 2 Autodesk links in particular cover the best practices we will cover.

- Autodesk Help
  - Performance Optimization and Model Efficiency
- Autodesk Knowledge Network
  - Recommended File Sizes

Project Setup

The bulk of the best practices we evaluated center around setting up the project.

Dividing Models and Teams

When this project began, we knew it would be a BIM beast and that we needed to carefully think through how we would divide up the effort between models and teams. Early on we looked at anticipated file sizes and compared those with file size best practices. The best practice about file size is that there is no max project size but rather a RAM to project size ratio of 20 to 1, plus add in more for the operating system and other apps.

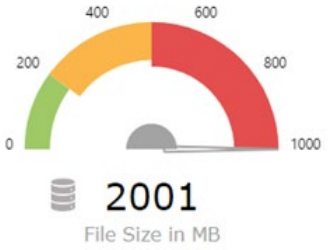
File Size

- No maximum unless non-workshared
- Use a RAM to Project Size ratio of 20:1, then add RAM for OS and other open programs



I think we anticipated 1 to 1.5 GB model sizes, possibly more, at least for the Architectural models if we didn't do any model splitting. So based on that and our typical computer with 64 GB RAM, we could easily handle a 3 GB project, which would not be near enough. We knew the project would total beyond 10 GB.

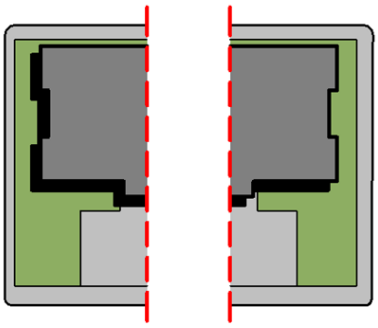
As it turned out, the project totaled 15 GB. The graphic to the right is from a model performance tool we used showing how far just one of the models went beyond our expectations.



Based on a 15 GB project, the best practice says our computers would need to have 300+ GB of RAM.

The Joint venture Architects had shared computers and Citrix virtual machines with hundreds of GB available we could rely on for intense tasks like printing, but not all our consultants had those kinds of resources. So, for daily work, especially for our consultants, we needed to consider model splitting strategies.

Model Splitting



Model splitting, or dividing large models into smaller ones, is a best practice; but it can have negative consequences. It does work well, when the smaller parts are more independent form each other, and links to them can be unloaded for most of the time.

You can divide models up by:

- Level
- Logical plan divisions like match lines or expansion joints
- Existing vs addition. Our Structural Engineer did that which was perfect for them.
- Team member scopes, like:
    - Discipline (typical)
    - Specialty teams within disciplines
    - Bid options
- Shell vs Core

As is typical, we split models by discipline; but beyond that, we did allow each discipline to use whichever model splitting strategies worked best for them.

The Architecture and Interiors models were by far the largest with many simultaneous model editors. That made model splitting interesting.

Looking at each of the above strategies, specifically for our anticipated largest models, we couldn't identify any clear divisions that would allow teams to turn off the links to the other parts for very long. Turning off links is key for model splitting to work. Any way we could think of to split the model, the resulting divided and split-up links would need to be loaded most of the time. For instance:

- The multiple atriums made splitting by level difficult for the Architecture team.
- The symmetry of the historic buildings meant teams often worked simultaneously across match lines on mirrored areas so dividing in plan was difficult.
- For the Architectural models, phasing better handled existing vs. new.
- Any Architectural sub-scopes we could think of, only benefitted 2 or 3 team members at the expense of the bulk of the rest of the team.

Almost every team member would often need to edit and coordinate elements across any scenario of host and link, meaning both would need to be open at the same time. Doing this would take up more resources than if everything was in just one model.

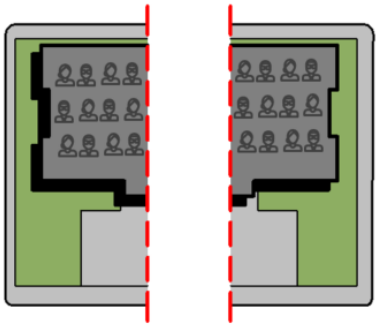
Then after factoring in the negatives of decentralized model splitting, which are the constant, manual coordination of shared:

- Content
- Parameter values
- Standard detail views

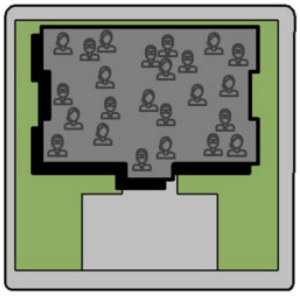
It made more sense for this project to just divide the Architectural models by building.

Concurrent Users

We had a very large Architecture joint venture team, so only splitting up the Architecture by building meant violating another best practice of keeping the number of concurrent editors in a single model to 12 and under. In the many decades of experience on large projects by the BIM management team, we have seen synchronization timing and object ownership problems when simultaneous editors goes beond 12, plus or minus. However, any model splitting strategy we thought of would result in no fewer than 20 concurrent editors.



Ideal but not possible for this project



Unfortunate Reality

So, to mitigate one of the biggest negatives of too many people in the model needing to synchronize simultaneously, we set up a sync sign-up Excel file hosted in MS Teams so it would be live for everyone in the joint venture.

	A	B	C	D	H
1	Write your name below and delete your row after your sync is complete. Be sure to alert the next person in line. Happy syncing!				
2	BE SURE TO ALERT THE NEXT PERSON IN LINE AND REMOVE YOUR NAME AFTER SYNCING				
3	Sign up to sync				
6					
7					
8	Name	Time Slot (EST)	Status (indicate ghost sync observed)	Next person informed?	
53	Steven Register	4:45 PM	Done	Y	
54	Rob Fink	5:00 PM	In Progress		
55		5:15 PM			
56	Fotis Perizes	5:30 PM			
57		5:45 PM			

Each Link on Its Own Workset

Another best practice related to dividing models is to put each linked model on its own workset. For large projects this is a must so that you only open the worksets, and therefore linked models, you need. This will speed up opening the model and reduce the need for lots of RAM while working.

But it's not all rosy for very large projects. Often, because of model sizes, you can't work unless you turn some of these links off. With context turned off, team members wind up working blind and creating clashes everywhere, like all the fires in the familiar KC Green comic, "This is fine."



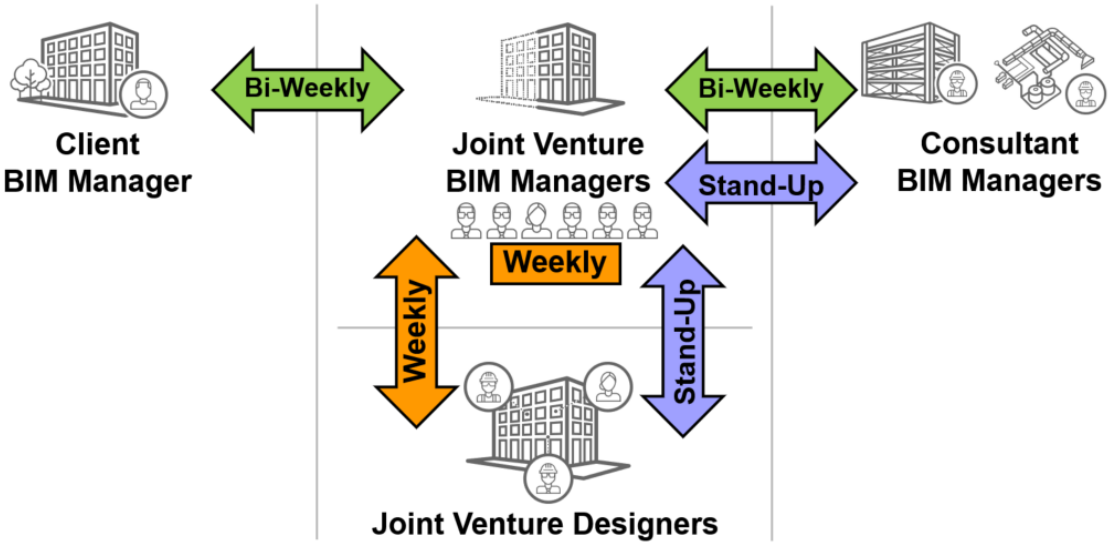
This kind of willful blindness is OK as long as you mitigate it by turning everything on and visually reviewing your work for the day.
Better yet, for very large projects, you should be doing frequent clash detection because you know, people need to work blind, and will miss things. So, make sure you budget for more clash detection so your team can have frequent feedback loops to put out fires as you go and not let them build up.

Collaboration \ Communication Standards

After we set out our model splitting strategy, another important strategy needed was how to keep all our BIM stakeholders connected and talking.

BIM Meetings

The diagram below was our BIM communication strategy.



At the center is our joint venture BIM Management team of 6 people who managed the back-and-forth flow of BIM data inputs and expectations, conceptually like a neural network. The difference between the above diagram and one we typically do for smaller projects was around frequency. For this large project we had bi-weekly BIM meetings with the client where normally client meetings happen infrequently at start-up and milestones. And we also added BIM standup meetings with the designers and consultants to increase BIM communication frequency.

The client bi-weekly meetings helped to move validation of the models sooner than just at deliverable submissions giving us more time and resources while everyone was still active in the models to make any adjustments. This helped us move closer to the digital DevOps concept of building quality into your process and not saving QA\QC until the end.

The standups were used more in the beginning of the project when you have more discovery and need more quick feedback loops. They were set weekly but only designed to last 15 minutes or less. Within that short window the team could have a 1-minute conversation with all BIM stakeholders present to get things resolved quickly or at least identify who was interested to attend a follow-up impromptu meeting. This freed people form long meetings that had nothing or little to do with their scope. I feel many disjointed email conversations with the wrong people cc'd were avoided because of these.

As we got into the CD phases where projects typically transition from the “Complex” space of discovery to the “Complicated” space of delivery, we found that just the weekly and bi-weekly meetings were sufficient. For more information on “Complex” vs. “Complicated”, search up Stacey matrix and Agile. Here’s a starting point from an article at Scrum.org:

> “Agile frameworks should be used to solve business problems in complex environments where more is unknown than known. For simple problems, where more is known than unknown, an alternative approach – such as Lean or Waterfall – should be applied. In this context, “simple” does not mean easy, instead it refers to the fact that more is known than unknown.”

Because of the joint venture, team distribution, and the Pandemic, all these communications happened over MS Teams. We also used Microsoft Planner to help make the standups and other meetings more successful.

Tools

Here is the list of the major tools we used to communicate and collaborate.

Tools

	WIP	Shared	Published	Archive
• **BIM 360**	RVT	RVT, Docs	RVT, Docs	
• **N2N**			Deliverables, Comments	Emails
• **MS Teams**	MS Office Docs, Planner, Chat			
• **Bluebeam Studio**	Sheet Set Review	Sheet Set		
• **Cintoo**			Point Clouds	
• **Keynote Manager**	Revit Keynote	Revit Keynote		
• **autoLink**		Standard Details		

Most of these we use on every project where there is design and engineering, like BIM 360, Newforma, MS Teams as I mentioned, and Bluebeam. However, for Newforma we went beyond the typical because of the joint venture and set up Newforma to Newforma connections between the joint venture companies and to the client.

Cintoo is also not typical for most projects, especially for new construction only projects; but on this project it was essential. See the “Point Clouds and Building Orientation” section later for more information.

The last 2 tools are from Revolution Design which were uniquely helpful to this project.

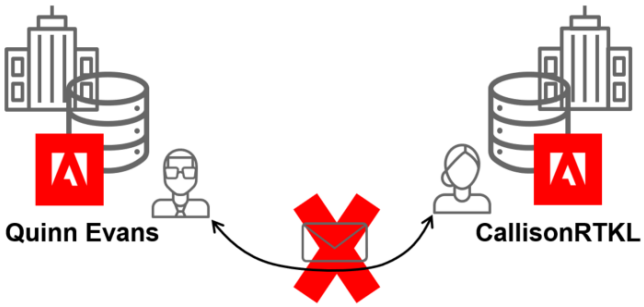
The Keynote Manager cloud solution gave us a central place to edit and share keynotes across the two joint venture companies CallisonRTKL and Quinn Evans.

AutoLink was valuable as we had two buildings that needed to share common details like typical walls, doors, etc. It gives you a way to reference, and keep synchronized, the sheet detail references found in linked models.

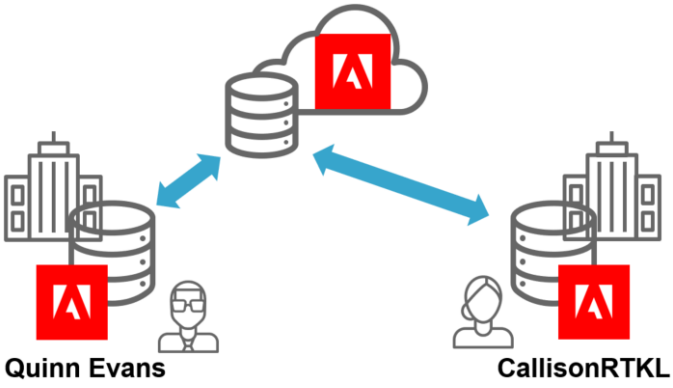
One important thing missing from this table is Adobe. We needed to use Adobe Illustrator and Photoshop early on in the project for conceptual presentations. Sharing Adobe files and libraries across two companies in a joint venture was not possible at the time with Adobe's service offerings. So, we looked for other options.

Private Cloud Solution

What we tried, but didn't work out so well, was a 3rd party private cloud solution to allow team members from the different design companies in the joint venture to work on Adobe InDesign and Photoshop documents together. Adobe products have resource pathing that is difficult to share across companies. We wanted to avoid manually sending libraries and data back and forth.



So, we looked at a few solutions and determined that a private cloud solution would give us the best value for the cost. It would allow team members from either company to work on the Adobe files in a cloud hosted external network while still maintaining secure links for each team member back to only their company network.



However, we couldn’t get it to work. The two biggest issues contributing to its failure were:

- Lengthy set-up time - We only needed this for SD so the need was almost over by the time we determined it could not be put in place
- Incompatibility - One of the joint venture companies could not get it to work with their network security solutions despite the 3rd party's assistance.

I think it could work in some cases, but it should be tested well in advance of needed timeframes for incompatibilities.

After we no longer needed a solution, Adobe came out with a cloud collaboration solution that we’ll be testing out on other projects.

Point Clouds and Building Orientation

This section covers best practices on point cloud distribution, management, and building orientation, starting with Cintoo.

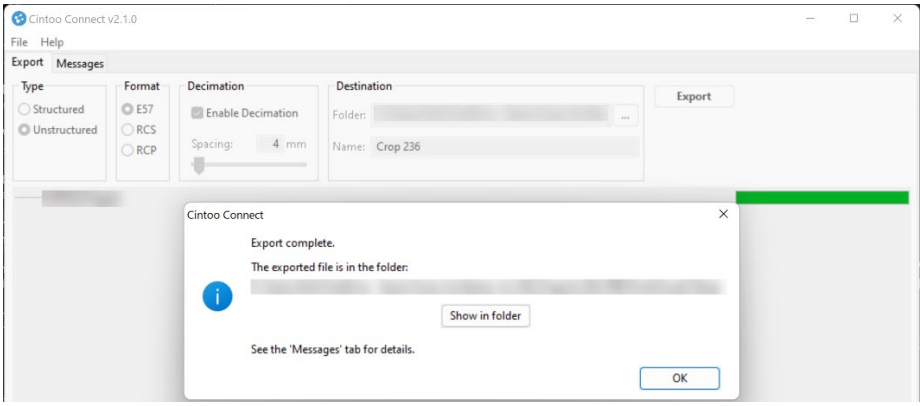
Sharing Point Clouds

Point clouds are large data files that can be unwieldy to manage on large projects. Distributing the files is difficult due to size limitations with most file sharing methods. Working with a point cloud in a model can be cumbersome if you are not aware of strategies to manage the display of the points.

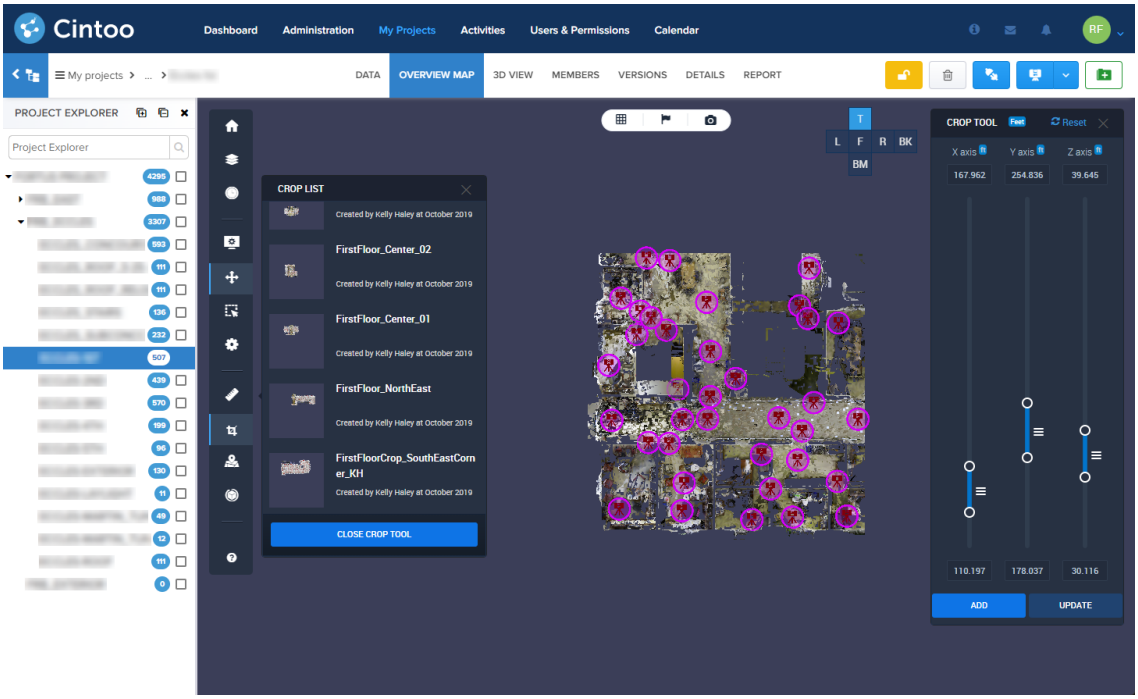
To address both issues, we use a cloud-based point cloud management service. On this project the provider we used is Cintoo. Cintoo proved itself effective in the ease of setting up the project, adding team members, controlling permissions, uploading scans, viewing and analyzing the scans, and downloading the scans. Unlike Revit or ReCap, Cintoo displays the point cloud as a mesh, not as points. This makes viewing the point cloud possible on a web browser and improves performance.

Scan data is uploaded via a WebGL browser by whomever registers the scans. In this case study a scanning company was contracted and granted permission to upload the scans. *Make sure the scan data is structured*; Cintoo does not accept unified, or unstructured scan data. Structured scans include information about the location of each scan, imagery, and the 3D point cloud. Once the data is uploaded it can be organized into work zones. If your project contains clearly delineated floors, consider setting up your work zones by floor.

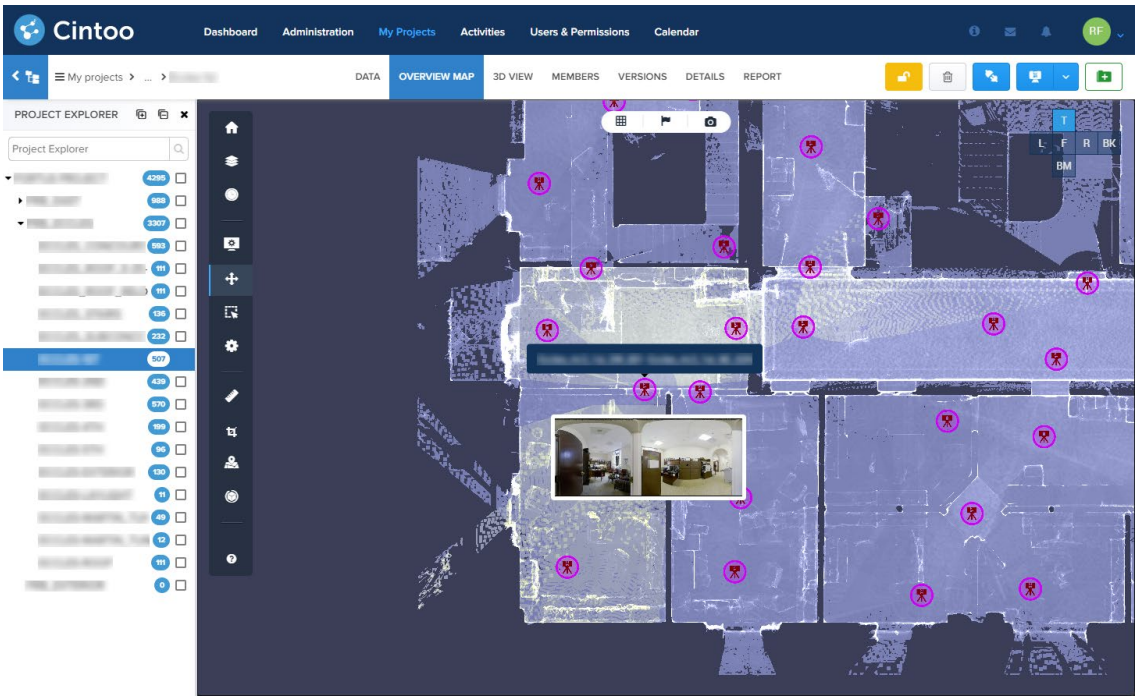
All team members can be added to the project and can gain access to the point clouds. Cropped areas of the scan data can be saved for easy reference and for exporting to an E57, RCP or RCS file. Having each team member download their own copy of the point cloud makes distribution easier, and team members only need to download what they need. It is a best practice when working with large point clouds to limit the point cloud data displayed in Revit to the specific area you are working in. To download a cropped section of the scan data you will need to install a utility from the Cintoo website. Within the utility you have the option of downloading a structured or unstructured file, an E57, RCP or RCS file, and you also have the option to decimate the point cloud data. Decimating the point cloud reduces the file size, but it will reduce the number of points. Typically, we found a decimation spacing of 4mm to be acceptable.



Below is an image of a cropped area of the point cloud. On the far left are the work zones. On the center left is a list of saved cropped regions. On the far right are sliders that can be used to adjust the extents of the crop area.



The magenta circles represent individual scan locations. Hovering over a scan location displays a thumbnail image. Clicking on a scan location takes you to a full 360-degree view of the scan.



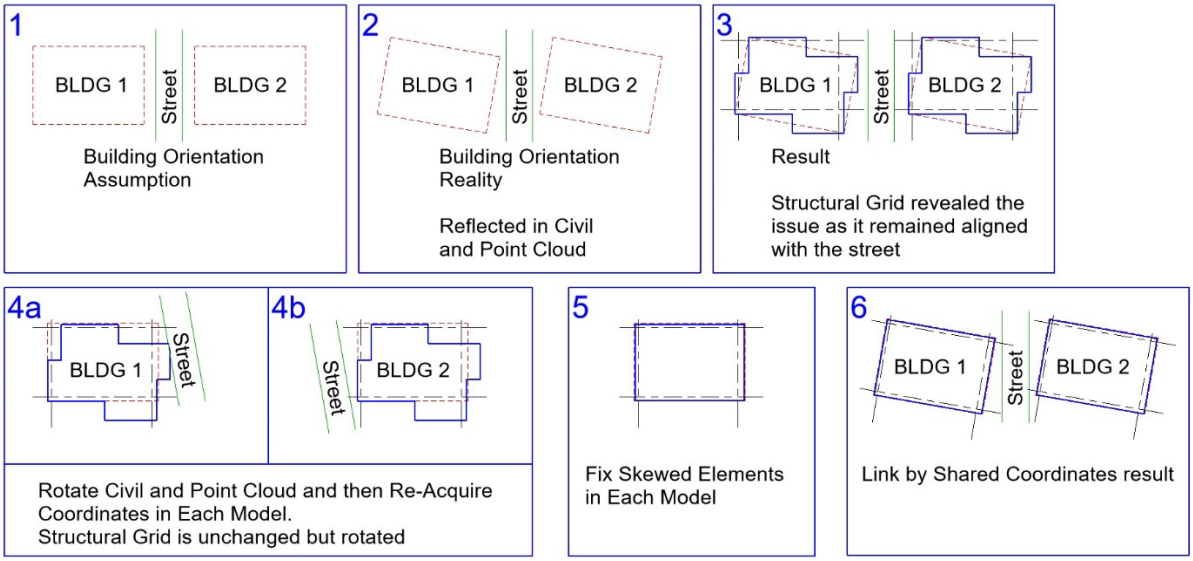
One of the liabilities of Cintoo, or most any cloud-hosted point cloud service, is the cost. We found it was worth the investment because of the efficiencies gained in managing the point cloud on such a large and complex project. By making the point cloud accessible to everyone on the project team, not just the BIM specialists, we found that more than just those working in Revit were using Cintoo to view the existing conditions. This not only gave the team a better understanding of the buildings, but it also cut down on the number of site visits needed. This in turn saved us money and helped offset the costs of the software.

A best practice learned from this is to invite all team members to use the service, regardless of their role. To increase the number of those who will use it, setup a training session to go over basic navigation and analysis tools.

The importance of making the point cloud accessible to the entire team cannot be understated. On this project the structural engineer was comparing the point cloud to the model and came to the realization that the buildings were not oriented perfectly orthogonal to north as was indicated in the original design drawings. This led to a discovery of a limitation in Revit that is covered in the next section.

Entire Building Is Slightly Off Axis

There was a slightly off access issue we had with this project as this storyboard illustrates.



Frame 1:
This project is two historic buildings oriented to the DC street grid which in turn is exactly oriented to the cardinal directions. The original historic drawings showed the buildings oriented to true north. The surveys and point clouds appeared to reflect that orientation as well. This was our assumption.

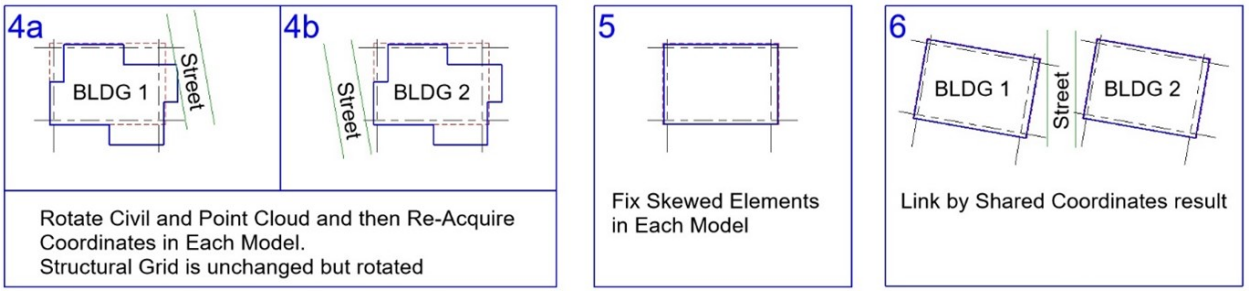
Frame 2:
However, both buildings were individually 1/16th of a degree rotation off of true north while the street was still actual true north.

Frame 3:
Not realizing this, we modeled the existing condition elements to best fit the point cloud, all while maintaining an assumed orthogonal orientation to true north for all objects. This created a lot of odd jogging of the layout as exaggerated in this frame. Due to the historic style of the buildings, there are no continuous wall faces across either of the 300' +/- long buildings, which would have made that 1/16th degree rotation obvious.

After completing the existing condition modeling, we brought in the Structural model. That's when we noticed the rotation. The Structural model used historic documents for their modeling instead of the point cloud since most of the structure was hidden from scanners within existing walls. So they were, fortunately, free of the influence of the orientation reality of the point clouds; and therefore, had all their objects correctly positioned relative to each other.

We couldn't get the Structural model and Architectural model to align properly. Encased columns were sometimes inside walls sometimes not. Simply rotating the structural model did not help since all Architecture objects were incorrectly modeled with each object essentially rotated slightly but centered in its correct space.

This resulted in having a compounded 10-inch discrepancy across the site, which was especially bad for aligning the tunnel between the buildings.



Frame 4:
The fix was, for each building, rotating the linked master shared coordinates file to align with the Structural model and acquire coordinates form it.

Frame 5:
We then proceeded to individually fix the orientation of every existing condition element we had already modeled.

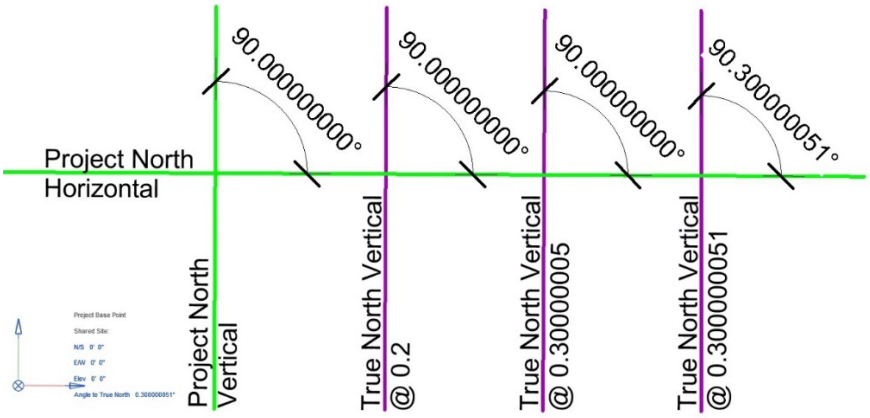
So, the lesson learned here is never assume a building is orthogonal to true north even though it was intended to be so and looks that way. Civil Engineers and Surveyors are thinking, "well duh"; and yes, it's a bit embarrassing. But that's not where this lesson ends. Revit had a hidden surprise for us.

Revit North Bug

Because we had to work with this ever-so-slight difference between project north and true north, we discovered that Revit cannot accurately draw or display a true north line when the difference between project north and true north is less than 0.300000051 degrees. Our project difference was less than that at .0625 degrees.

That is odd by itself, but stranger still is that if you export to dwg a view with vertical lines that look and measure incorrectly in Revit, those lines will be and measure correctly in the exported dwg in AutoCAD.

We demonstrated this to Autodesk, and they fixed it starting with Revit 2022. However, upgrading a project from prior releases does not fix it, so our files that started in 2020 are not fixed. Interestingly, if you upgrade a Revit template file to Revit 2022 or later, any project started with that upgraded template file will be fixed. That seems odd that upgrading an rte creates different results than upgrading an rvt.



Here are the steps to replicate the bug and explain it a bit better.

1. In Revit 2021 or prior, in a plan view set to Project North, use snaps to draw a horizontal and a vertical line (shown in green).
2. Rotate True North by 0.2 degrees or less.
3. Change your plan view to True North and draw a vertical line (shown in purple).

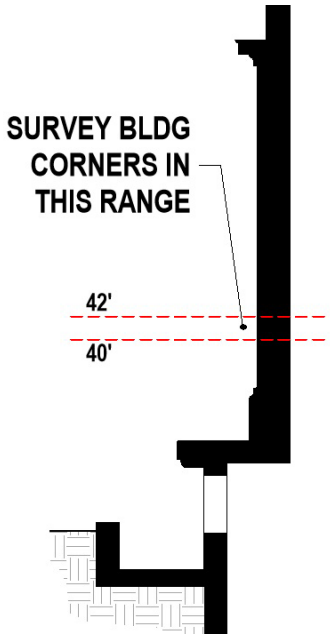
The purple lines should not be parallel or perpendicular to the green lines, but Revit measurably and graphically, through aliasing, shows they are. Rotating True North by 0.300000051 degrees or more displays and measures the lines correctly.

Another oddity is rotating true north for the first time in a project file using values greater than 0.20001 degrees will not exhibit the buggy behavior. However, if you first rotate true north less than 0.20001 degrees and then rotate again so that it is less than 0.300000051 degrees, you will get the bad behavior.

This issue became important to consider for the tunnel and sitework like aligning paving patterns to the buildings. To get around the issue, we snapped to known-good reference planes for true north oriented objects.

Visual Backcheck for Survey and Point Clouds

So, with that quirky north issue and because we initially didn't recognize the 1/16th degree rotation, everyone was a bit nervous about alignments. The BIM management team was confident the surveyor company and point cloud scanning company had everything properly aligned and bench marked, but we wanted to give the design team a visual backcheck so they could be confident too.

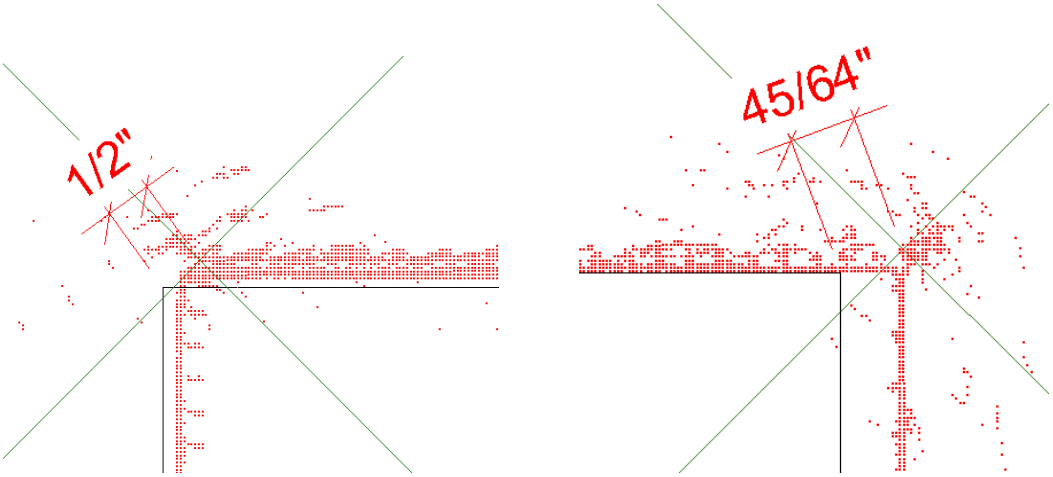


Because the outline of each building was drawn on the survey based on where the building touched the ground, deep window wells and ornamentation made it difficult to find, and especially visually display, the corresponding 3-dimensional location in the point cloud.

So, to create this backcheck, we found a narrow elevation range above watertables where the exterior wall geometry was free of ornamentation and where the point clouds were free of shadows and had good overlapping of individual scans.

Then we had the surveyors go back on site and measure the building corners at that specific elevation range relative to the site's benchmarks. With that data, we were able to accurately verify the alignment of the buildings to the site in a visual way that was easy to communicate to stakeholders and managers so we could proceed with confidence.

The images below are samples of the resulting data overlays we showed the design team.



- Green X = New Building Corner Survey in Specified Elevation Range
- Red = Point Cloud Slice in Specified Elevation Range
- Black = Old Revit Building Corners to be Adjusted to the Red point cloud

As you can see above, the new building corners survey and point clouds align well within the typical ½" tolerance.

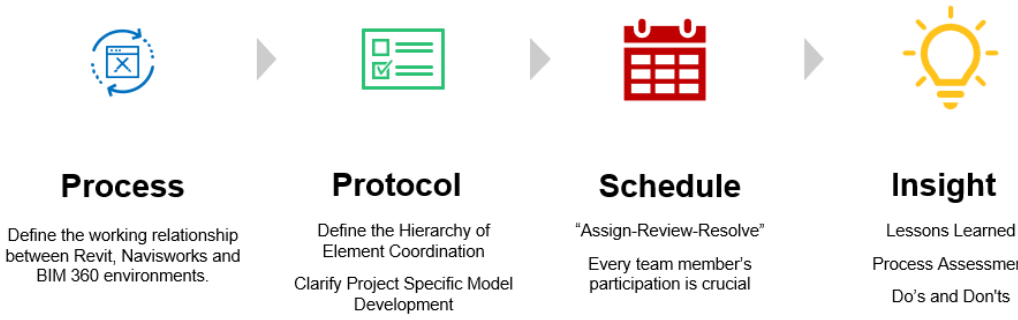
The best practice lesson here is that when you have surveys and point clouds done by 2 different companies where alignments are critical, make sure you have an alternate alignment method to backcheck the typical benchmark methods. Surveyors and point cloud scanning companies do a great job of orienting their work, but this technique just adds a simple QA\QC test to double check for any cross-company workflow inconsistencies.

This technique also fits the Visual Feedback theme throughout this document. Frequent visual feedback is important for any AEC project, especially large ones like this one. One prominent feedback loop which we will get to next, is Model Coordination.

BIM 360 Model Coordination

Model Coordination and Clash Detection also provide another layer of visual feedback loops. Intermittent workflow silos or even the slightest inconsistencies in building orientation can result in thousands of clashes which may not even be intended. In order to avoid this, we set up a detailed workflow to establish an efficient clash coordination protocol to pull team members out of those silos and discuss the bigger picture on a weekly basis. While we understand the importance of independent, focused design and engineering work sessions, we can't let independent work go on for too long before regrouping with the team.

Clash Coordination Setup



We first started by testing which process, and software platforms would best suit our needs for this type of communication. Here we defined the working relationship between Revit, Navisworks and BIM 360.

There were a set of model element hierarchy and client specific LOD clarifications that we first needed to establish with our team. These were documented in a Clash Coordination Protocol which is always a good starting point to address *expectation bias*. Similar to the term *selection bias* in Psychology, *expectation bias* as defined by Brad Hardin is "an error in expectations of the individuals or group who take part in a BIM project" (Source: BIM and Construction Management: Proven Tools, Methods, and Workflows). BIM expectations tend to be overlooked or miscommunicated by those who are managing the flow of information. When clearly documenting workflows and outlining the roles and responsibilities of each team member, the information manager has a higher chance of mitigating this unintentional miscommunication.

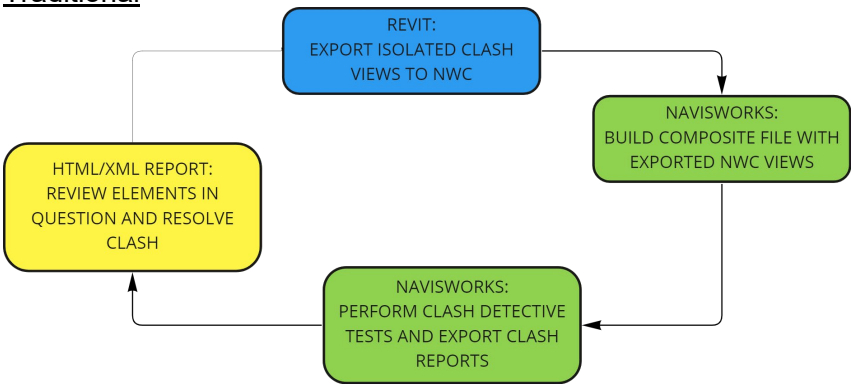
The Clash Coordination Protocol document also included a specific schedule of clash review meetings and the "Assign-Review-Resolve" timeline. Group participation in these meetings was crucial.

Finally, after much trial and error, we documented what worked and did not work and assessed how well a team this size can work on BIM 360's Unified Platform.

Process

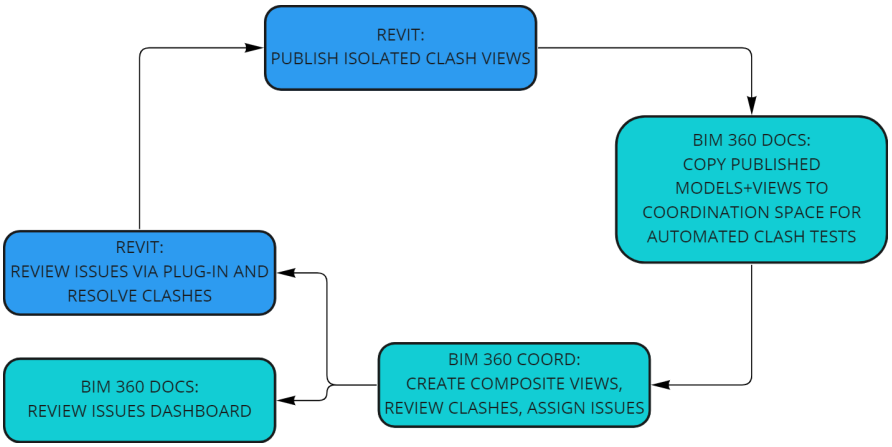
We started off with the simplest workflow that involved Revit and Navisworks. This "traditional" workflow was then built upon as we incorporated BIM 360's cloud tools to not only streamline the flow of information but also to enhance the team's experience during resolution work sessions.

Traditional



This process may be the most familiar to the user and begins in Revit where clash detection views are created. Each team member is responsible for the elements shown in these views so they must only show the model elements required for clash coordination and no other team member's elements or linked models. The views are exported in .nwc format and a composite .nwf file is created from these views to perform clash tests. These test results are then exported to a common reporting format (.xml or .html) so they can be reviewed. Note that there is no way for this report to be fed back into Revit where the resolution needs to be made. This is where BIM 360's cloud tools proved themselves very useful.

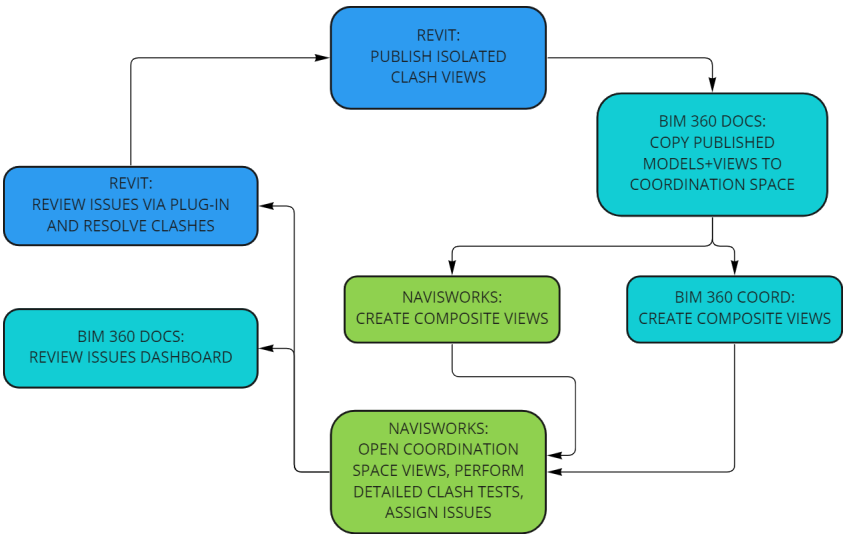
Cloud-Based



The next step was to study how BIM 360's cloud tools could handle the size and quantity of model views required for clash detection on this project. This workflow was completely

cloud-based where clash views were published from the cloud-initiated Revit model directly to BIM 360 Document Management (also known as BIM 360 Docs). These views were copied into each building's Coordination Space (see how to Create Coordination Spaces on the Autodesk Knowledge Network) and clash tests were automated through the BIM 360 Model Coordination service. In this module, the user can create composite views, review clashes and assign issues all within a web browser. The BIM 360 Issues (see Issues in Document Management on the Autodesk Knowledge Network) were assigned to team members and all users with the appropriate permissions could review the dashboard in the Document Management module. The major advantage this cloud-based workflow has over the traditional method is that these assigned issues can be viewed almost instantly in Revit via the Issues Revit add-in (see Revit Issues Add-In document on the Autodesk Knowledge Network).

Integrated



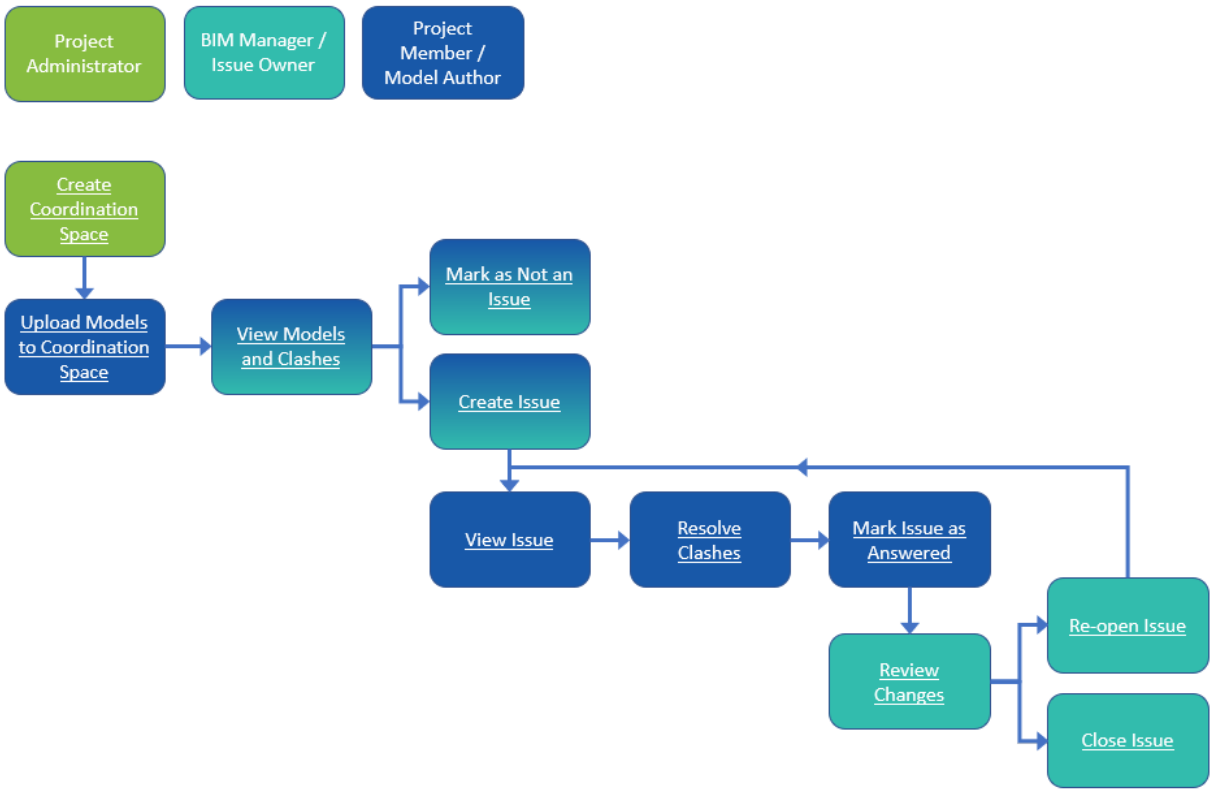
Integrating Navisworks and BIM 360's Model Coordination service brings the best of both technologies together: BIM 360's time-saving automation and cloud capabilities, as well as the sophisticated Navisworks workflows that users are already familiar with. In this workflow, we maintain the majority of the cloud-based process steps, but any team member with the appropriate access permissions could create a live connection with Navisworks. These is done via the Coordination tab (see BIM 360 Issues Add-in for Navisworks document on the Autodesk Knowledge Network). This is where we saw instant improvements in processing time as well as consultant participation as we employed the enhanced clash detective and viewpoints tools Navisworks has to offer.

Protocol

In addition to defining the software workflow, we issued a Clash Coordination Protocol document that also outlined how the team was to use the Issues add-in from Autodesk. This add-in was first available for Navisworks only, limiting the team's ability to participate in the clash detection process. When Autodesk released the add-in for Revit, we were able to execute and enforce the integrated process to its full potential. We based the Clash Coordination Protocol document on Autodesk's Model Coordination Visual Workflow shown below. This was a good starting point where team members clearly understood their roles and responsibilities in the overall process.

Within this document we issued a written explanation of the visual workflow as well:

> "In accordance with the Project Schedule, the BIM Manager will perform clash detection with the design models using BIM 360's cloud-based Model Coordination module. The BIM Manager will group and organize clashes for review by the Project Team. The clash groups will be assigned to each team's Clash Coordinator using the BIM 360 Issues. Each Clash Coordinator will have 1 week to acknowledge and respond to the assigned issue. Once clashes have been reviewed and a potential resolution has been determined by a specific team member, such team member shall provide a response in the 'Response' box in the Issue window, mark the issue as 'Answered' and resolve the clash or clash group in a timely manner and preferably prior to the next scheduled meeting time. The BIM Manager will then copy in the latest, published Navisworks view from the team member's model into the appropriate Coordination Space, verify that the clash has been resolved and mark the issue as 'Closed'."



Visual Workflow: Using Model Coordination.
Autodesk Knowledge Network

This document also was intended to:

- Establish a Hierarchy of Elements
- Provide Model Development Clarifications
- Mitigate Expectation Bias

Schedule

Clash coordination meetings require some ebb and flow between individual and group session work making sure that you don't spend too much time in each, respectively.

The individual tasks for our team mainly consisted of:

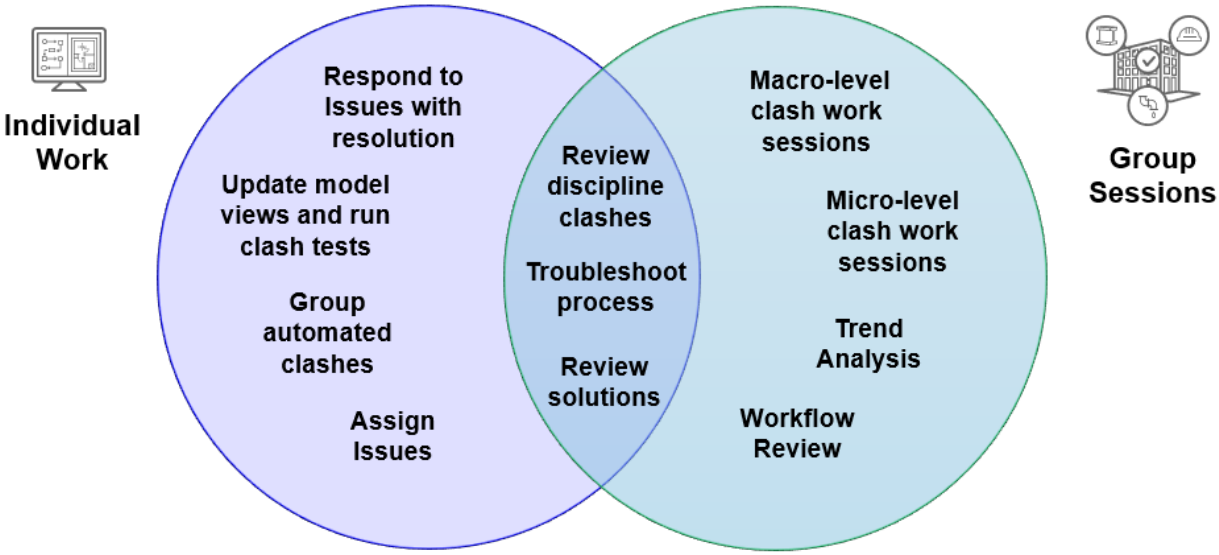
- Responding to Issues with a resolution
- Updating model views and running clash tests
- Grouping automated clashes
- Assigning Issues

The Group tasks were primarily focused on:

- Macro-level clash work sessions (entire team)
- Micro-level clash work sessions (two to three consultants)
- Trend Analysis
- Workflow Review and Feedback

Not only was there ebb and flow between both sides of the spectrum but there are critical activities that overlap such as:

- Review discipline clashes
- Troubleshoot process
- Review solutions



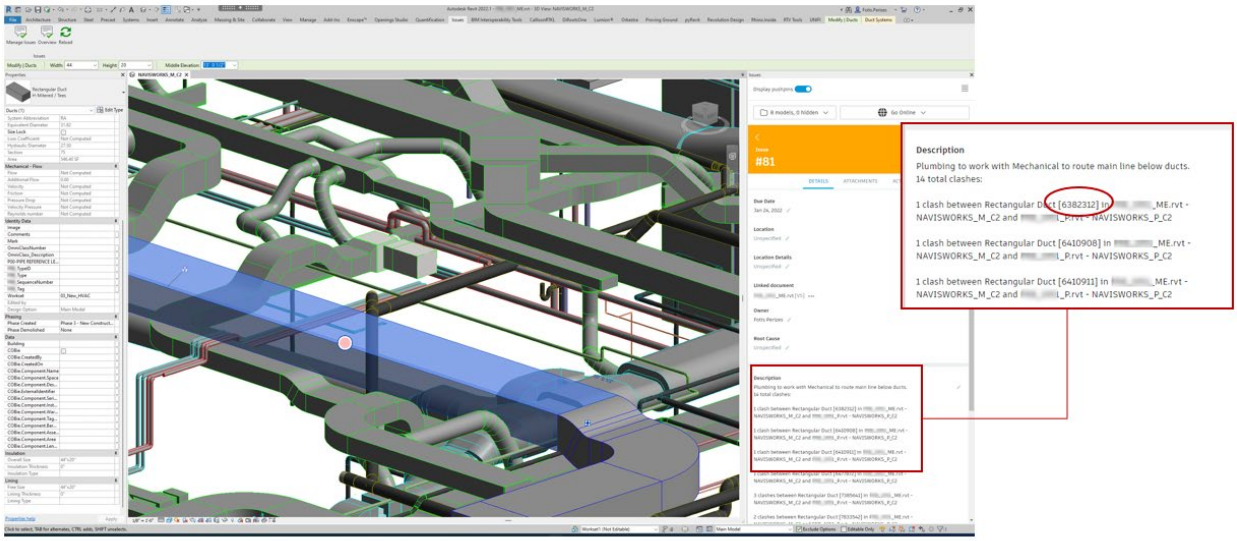
BIM 360 Issues

This invaluable add-in was first available for Navisworks only, limiting the team's ability to participate in the clash detection process. Soon after, Autodesk released the add-in for Revit as well. Our first pass at implementing this add-in was to provide a medium where all team members could access without specific software requirements. Just set the appropriate permissions in the Project Admin space in BIM 360, include View and Create permissions for your discipline specific BIM coordinators and you have a simple coordination workflow with just a few clicks. In reality, it's not that simple.

The BIM 360 Model Coordination environment was a great start to setup our collaboration views, get the lay of the land, maybe get rid of some extraneous clashes that were technically not issues at all. However, as the models grew in size, we felt that BIM 360's coordination space could not keep up. This is when the following error message kept rearing its ugly head time and time again.



Finally, the communication and discussions in our group sessions were able to come full circle with the Issues add-in for Revit. This enabled even greater participation from the entire team and through trend analysis, we witnessed an exponential increase in answered and resolved Issues. At this point, we were able to provide the same clash data that was originally pulled from each team members Revit models, something as simple as the element ID you see here:



This data was used to navigate and quickly identify elements in conflict during our group sessions in Navisworks, and finally reference it back in Revit where individual team members could quickly identify and resolve clashes within their authoring software.

Insights

So, what did we learn from all this?

- Model coordination module size limitations exist. We need to conscious of not only the model file size but also how many versions are piling up in each coordination space.
- Cumulative model size per coordination space matters. We experience a direct correlation with total file size in each coordination space and performance degradation.
- What else really made an impact on performance? Our choice in web browser - Chrome vs Edge

Here are a few Pros and Cons that we identified:

- BIM 360's model coordination is a great tool to quickly perform essential clash tests on any model set but once your LOD level and model size increases, it's best to start transitioning over to Navisworks for optimal performance during your clash review sessions.
- BIM 360 Issues was an invaluable tool that truly unified clash tracking in Revit, Navisworks and BIM 360.

Finally, we identified some important Do's and Don't's for future projects:

- Start by isolating what is not an Issue vs an actual Issue
- Assign, Assign, Assign Issues!
- Don't overload a coordination space (direct correlation with automated clash detection processing time)
- Use Clash Groups in Navisworks (see To Groups Clashes in the Navisworks Coordination Issues Add-in document on the Autodesk Knowledge Network)

So, these are all the lessons we learned that make the important feedback loop of clash detection work smoothly. Especially on a project this large, frequent feedback loops are a must, ensuring that individual work does not go on too long before you incorporate it back into the group.

Alternate Design Options



In response to the pandemic, the Client reconsidered the open office layout for a closed office plan. They asked us to develop an additional design: a hybrid open/closed office plan. This was to be developed in parallel with our current open office layout. By this point we were nearly two years into the design of the project. To further complicate matters, there were several smaller design decisions that were waiting for feedback from the Client. If the project were a tree, there were many limbs growing far from the trunk and it needed pruning.

For the hybrid office layout, we considered creating a separate model for the design. We decided to do this only for specific disciplines that were most affected by the task and could clearly delineate their scope. This included the Interior Design, Plumbing, Mechanical/Electrical, and Telecom models. The decision not to do it by other disciplines was made because it was going to be too difficult to separate the scope, which would lead to duplicate modeling efforts that would need to be developed in tandem. That wasn't practical, so we opted for heavy use of the Design Option tool.

What did we learn from our experience?

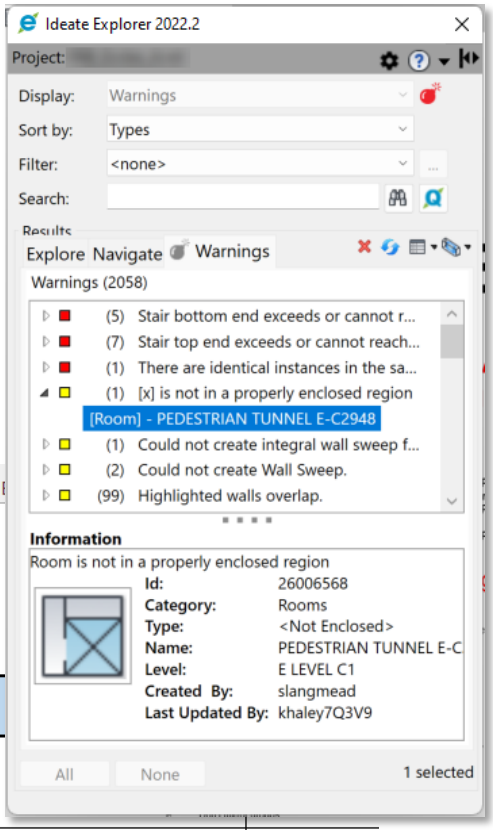
- Design Options are an extremely useful tool for exploring alternative designs. However, they can become a liability. On large projects they can proliferate quickly and often do not get resolved in a timely manner, resulting in a significant degradation of model performance.
- As a best practice, label each design option with the name or initials of the author and include the date created.
- Setup a routine where design options are reviewed at regular intervals, such as a weekly BIM coordination meeting. Encourage team members to explore and accept design options as quickly as possible. Don’t let the branches get too far from the trunk. Keep the tree trimmed.
- A linked model can be a better solution than design options, but the scope must be clearly separated.

QA \ QC

Just like managing design options, Quality Control is something you do as you go and not let things pile up at the end. We used a variety of QA\QC techniques and tools throughout the project, and we wanted to share with you some that we found particularly helpful in giving the team frequent feedback on how they did, as well as give the BIM management useable performance data.

Single Model Tools

Ideate Explorer was one very handy QA\QC tool helping us analyze and address Revit model issues. One aspect it helped us with, was managing warnings. It not only lists the warnings in your model but also gives them a color-coded severity weight. We had thousands of warnings to work through in just one of the Architecture models, and this helped us prioritize.



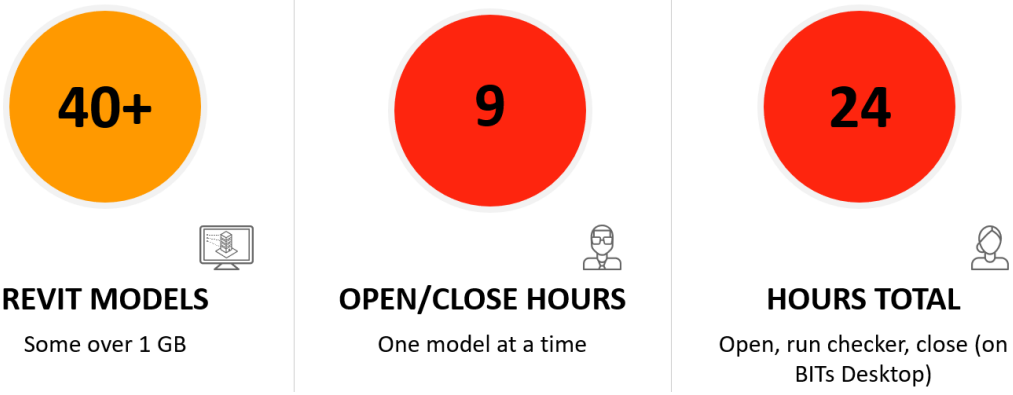
It also helped us put together a warnings explanation report required by the client.

	A	
1	Below are the compiled warnings from all models with an explanation of why they may exist within the models. Some warnings still need to be reviewed and/or corrected. The ones wothout an explanation are still under review. All the other tabs are the unique warnings that exist per model.	
2	**Warning**	**Explanation**
3	A Corner Mullion has been placed at the end of a Curtain Wall that is not joined to another Curtain Wall. The angle of the Corner Mullion will default to 90 degrees.	Intentional.
4	A wall and a room separation line overlap. One of them may be ignored when Revit finds room boundaries. Shorten or delete the room separation line to remove the overlap.	Existing conditions may require these conditions. Each should still be reviewed to ensure Rooms\Spaces are not affected.
	A wall and a room separation line overlap. One of them may be	Existing conditions may require these conditions.

It was a very good tool for our BIM coordinators who were responsible for a very small number of models.

Model Checker Effort

For our large project we needed insight into all the models and opening each one up individually to run a single model QA\QC tool was impractical. We had over 40 Revit models with some over 1 GB. Just to open and close each one without doing anything else would take 9 hours. If we opened each model, ran a single model QA\QC tool like the BIM Interoperability Tools (BITs) Model Checker desktop version, and closed the models, it would take 24 hours. With that kind of time sink, those single model QA\QC tools would only be used at submission milestones.



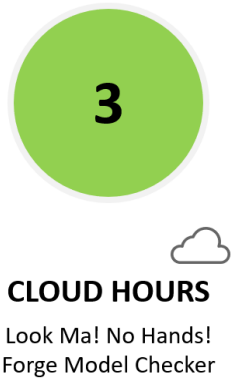
However, because this was a BIM 360 project, we were able to use a couple of Forge enabled tools to run QA\QC checks automatically.

The first one we used was similar to the BITs Model Checker, except you can schedule checks to run automatically in the cloud. With this Forge enabled Model Checker, we could click a few buttons, do something else for 3 hours, and then get our reports.

That made this tool useable at a frequency in-line with the Agile visual feedback principle that says:

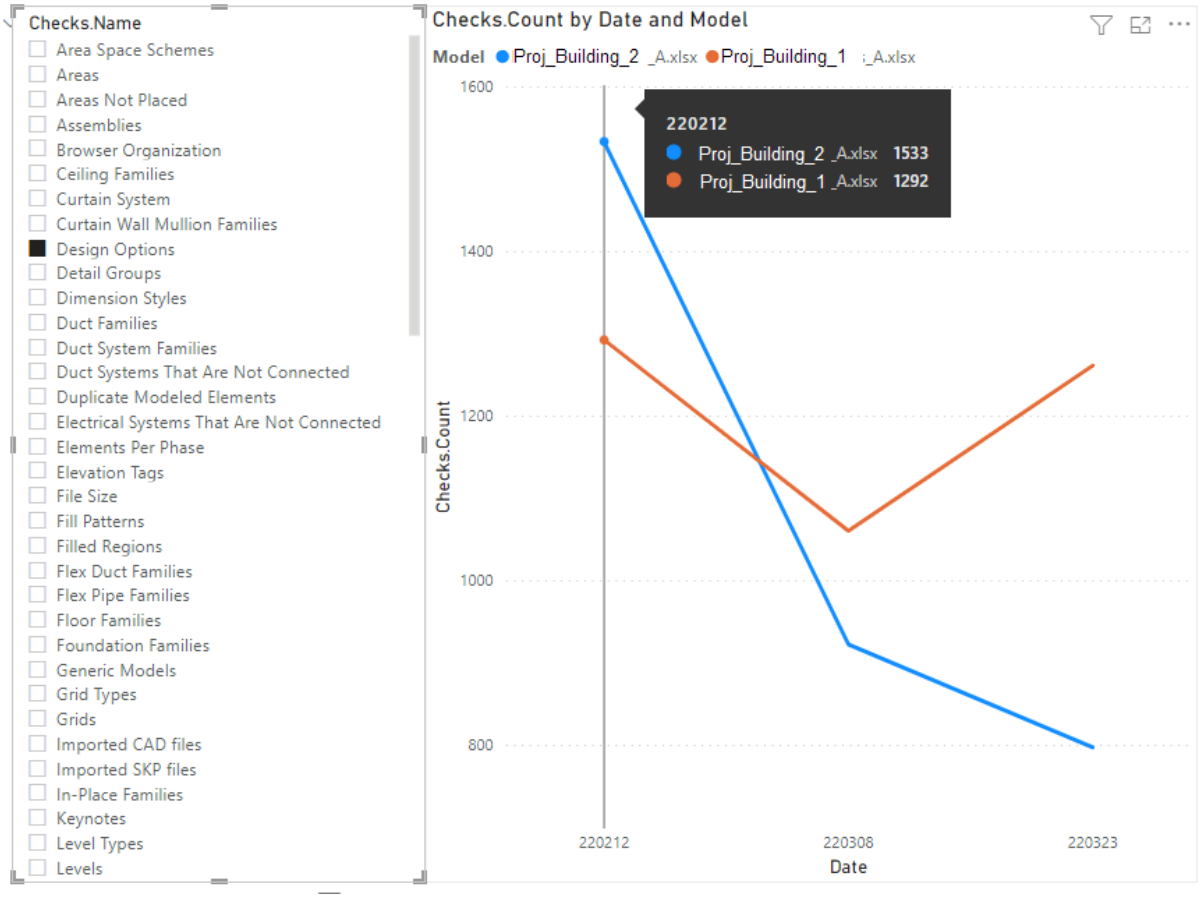
"At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly."

The Model Checker comes with a PowerBI dashboard we used to track model trends and to share that with the design team regularly so they could adjust accordingly.



Model Checker Report

Here is a sample of the PowerBI dashboard from the Forge enabled Model Checker data.



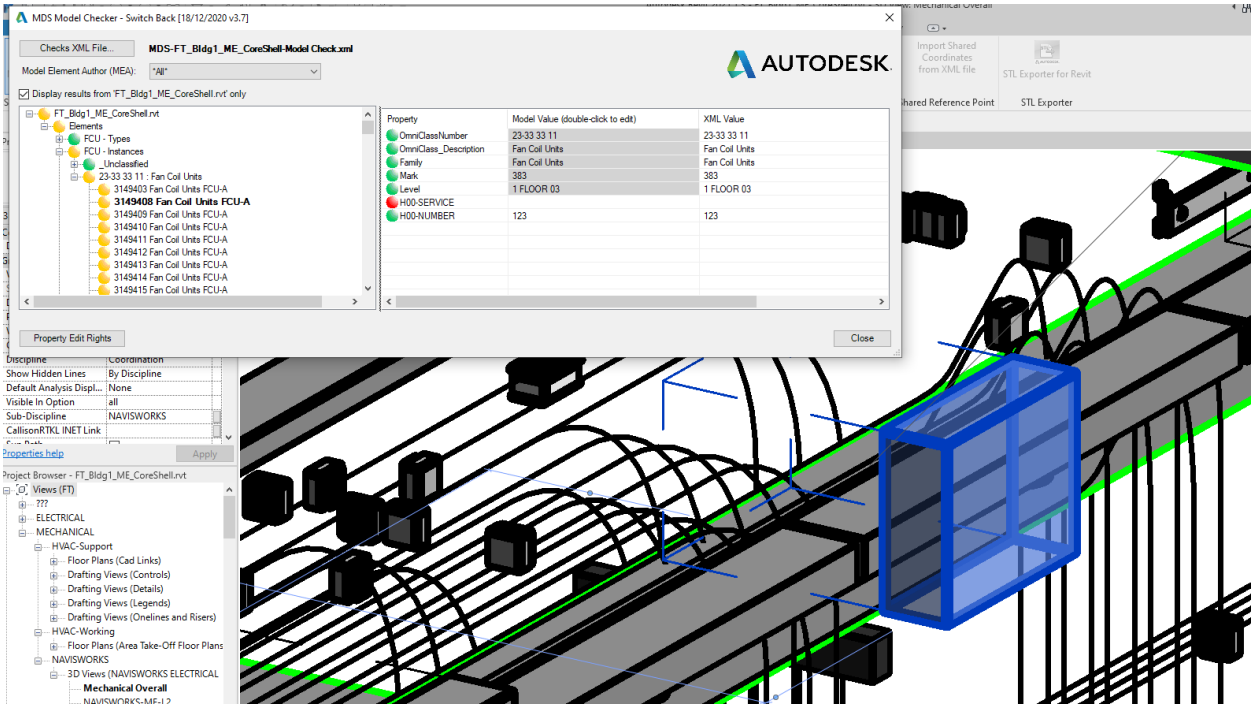
The chart above shows the number of elements in Design Options in our 2 Architecture models which we were very interested in bringing down. As you can see in the left menu there are many other performance data trends we could view as well.

We used this tool to look at individual model performance data and to compare trends across models as well. When everyone saw how their model compared health-wise to everyone else's, it got people to fix them immediately before the next report ran. I'm not sure if that was due to pride of workmanship, competitiveness, or peer-pressure. But, whatever the motivation, we saw things that could cause performance issues if left to linger, get resolved quickly.

By the way, this product is now called the Autodesk Validation Tool (AVT) and Autodesk is still working out the details of its availability.

Model Development Checker (MDC) Online

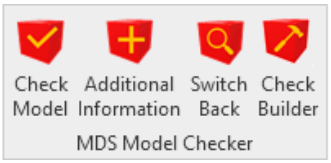
The other cloud QA\QC tool we used briefly was the Model Development Checker Online (or MDC Online). This was more of a proof of concept we did with Autodesk that let us check compliance with the client's family naming and classification standards for equipment to be included in the COBie deliverable.



There is a desktop version of this tool that has been around a while, but this was the Forge enabled version. It allowed us to simultaneously process multiple Revit models automatically in the cloud instead of one at a time manually. Its focus is not model health like the Model Checker, but rather compliance with information requirements.

With it you can check the values of parameters to make sure they follow naming patterns, value ranges, or just about any other test. There are some simple built-in tests, but it also recognizes Regular Expressions (Regex), and can run custom scripts you can write in Lua which is a popular scripting language with a large support community.

One cool feature is that it uses the Switchback tool to highlight non-compliance issues right within Revit, making it a contextual, visual feedback loop.



Unfortunately, we got it set up too late and only used it at the very end of the project. Even so, it did help automate the verification of how successful our COBie data complied. We will definitely use this on other big projects we have, and we'll keep trying whatever other experimental tools we can get as we constantly have large, challenging projects.