[AS226654]
[Cross-Platform Unitized Curtain Wall]

[Nate Holland]
[NBBJ]

Learning Objectives

- Learn how to create a reusable curtain wall rig with standardized parameters
- Learn how to build schedulable and tag-able assets into your curtain wall design
- Learn how to automate the instantiation of complex curtain wall modules in Revit using Dynamo-based data and data from cross-platform design models
- Learn different cross-platform design strategies to prepare for interoperability from day one

---

Description

The curtain wall tool is one of the most powerful features in Revit. When it's done right, it can drive efficiency from design to contract administration. Façade design inherently lends itself to modularity and repeatability—which makes the scope of building envelopes the ideal testing ground for new techniques in parametric design, as well as design computation. Thinking about and modeling modular elements in a manner similar to the way they'll ultimately be fabricated not only makes the documentation process smoother, but also educates young architects as they design. At NBBJ we've developed an ecosystem of tools and processes to make curtain wall design an easier and more nimble process. We recognize that designers prefer a variety of tools and that different software have different strengths. In this presentation, we'll demonstrate strategies and tools we use to create parity and enable interoperability between the programs we use for façade design.

Speaker(s)

Nate leads the Design Computation team at NBBJ. He is actively engaged in setting firm strategy for design computation, leading digital initiatives, as well as developing custom tools and cross platform workflows specific to project needs. He led the computational efforts for the design of the glass and steel shell of the Amazon.com Spheres project and the master planning of surrounding 3 block development. He's also worked on the headquarters for REI, the Two Union lobby renovation, and the Rainier Square Tower in the Seattle area. He has master's degrees in both architecture and business administration with a keen interest in all things digital.

Strategies for Cross Platform Design

Think like Revit

Revit is the documentation tool for almost all projects that go through NBBJ. Other tools come and go depending on the project team and client needs. When setting up a cross platform strategy it is important to plan ahead thinking about how the design will ultimately need to be modeled and documented in Revit. These choices made early in the design phase have a large impact on the amount of effort required to move between platforms.
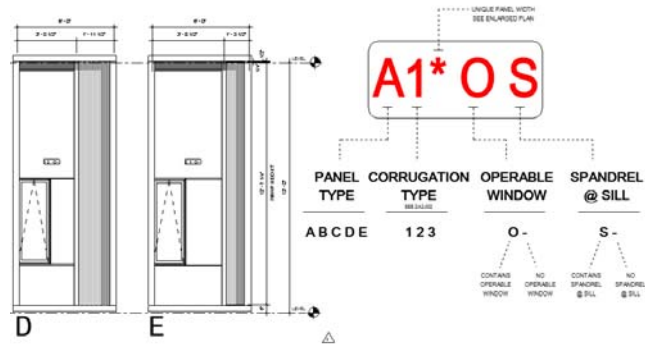
Model like it's built

Within Revit there are many ways to build and represent different building elements, especially facades. Each method, from walls with punched windows to adaptive components driven by dynamo has its place. For most of our projects, the façade will be fabricated and installed in a unitize manor, through unitized curtain wall, precast, or prefabricated stud framing. We aim to model and document our projects with a strategy that closely aligns with how the contractor will put the project together. This helps clarify the documentation of our intent, as well as, trains younger designers in what their drawing and how it goes together.
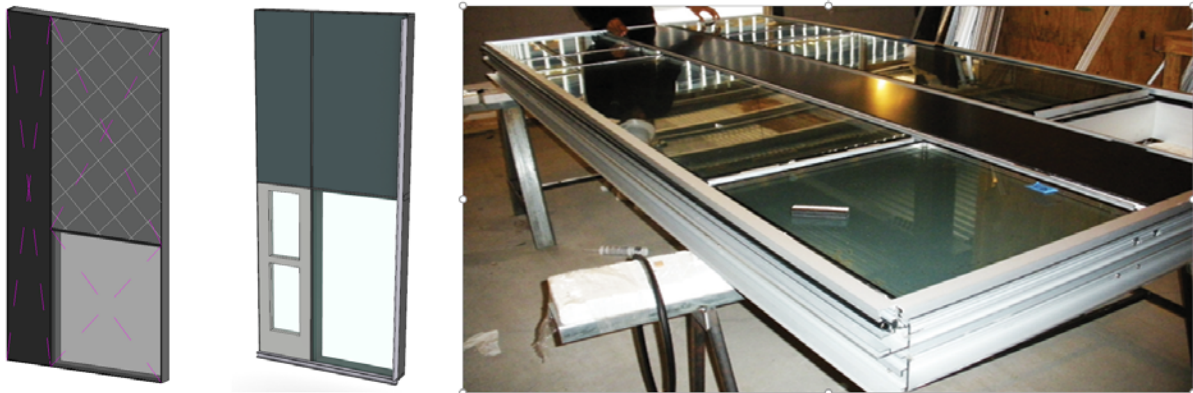
Create parity between design programs

Each BIM and 3d modeling tool has its strengths and weaknesses. Some are built for speed and flexibility, others for accuracy and data. To achieve these goals, the tools are fundamentally different under the hood. Moving data and geometry between platforms demands consistency in how things are modeled and represented digitally. By adopting a parallel modeling approach that mimics both the way it will be built and our best practices in Revit, our designers are set up for success in interoperability.
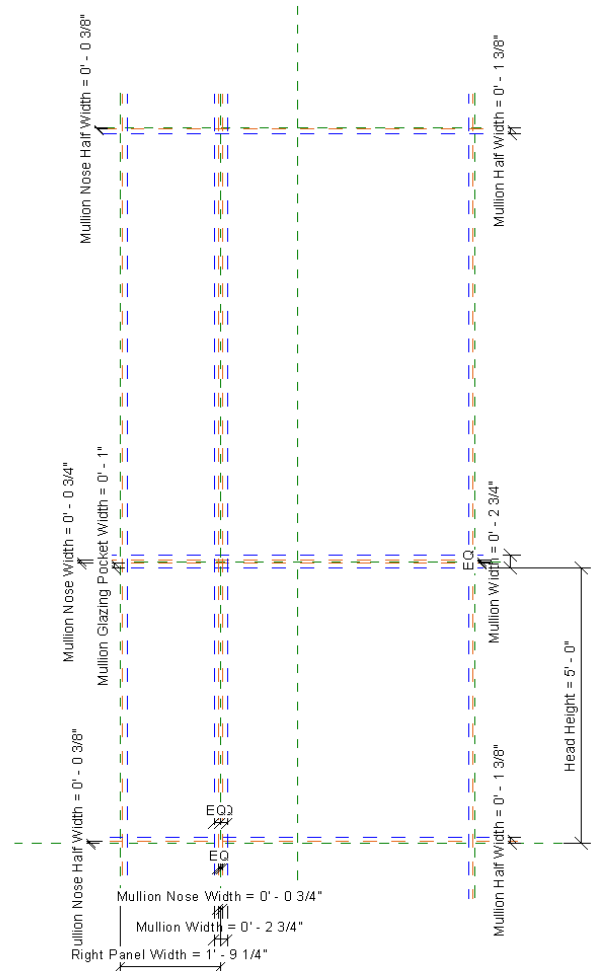
Reusable Curtain Wall Rig

Control the gridlines

When building complex families in Revit, reference planes are your best friend. Everyone's had that experience where an extrusion is snapped to another piece of geometry and the stretching or moving doesn't behave as expected. References allow you to establish the rules first and then you can set up the geometry to follow.

Use Sub-Object styles

A sea of green dashed green lines can be a challenge to navigate for experienced and new users alike. We adopted a purposeful-color coded system of sub-object styles for our grid lines. Green line represent the center of mullions, Blue lines are the edge of the mullion, and orange lines are the mullion nose width.

Sub object styles give you global control over elements within families. Within the primary model you can rapidly change their visibility settings, or even use the sub-object styles to create custom exports to layers. A little time spent setting them up early makes controlling the model much easier late in the project.

Standard Parameters

One template doesn't fit all panel shapes and configurations. Using standard naming conventions from one family to the next helps them feel familiar and less intimidating to designers. It also makes it easier to create schedules and tags that work with multiple families and types.

Shared instance parameters from nested families can be useful for extracting more customized data from the family such as material takeoffs and street level transparancy percentages.

Think beyond one platform

What are you trying to accomplish

This is often the most important and most challenging question with interoperability. Identifying what geometry and associated data will be used for is key to identifying the right strategy. Not every use case calls for the holy grail of continuous bi-directional instantaneous interoperability. Something faster and much lighter is usually more appropriate for our typical projects.

Visualization	Place holder Geometry	Setting Family Types
Documentation	Scheduling	Driving Parameters
Communicating	Coordinating	

UI/UX

Moving beyond the limitations of one platform allows you to explore different interaction and interface paradigms. When conceiving a façade:

- How do you want to design it?
- How should the tool interact with you?
- What data or parameters are important?
- How should they be displayed?
- Do all of the users need to be architects?
- Can someone in another city participate in the experience?
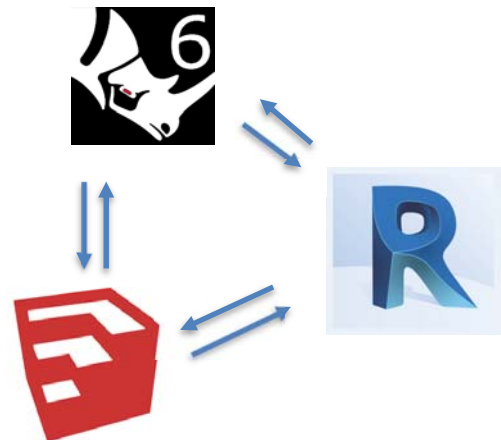
Make it re-usable

Document workflows

Last, and maybe the least exciting, but important to remember, architecture is a painfully slow process, often taking 5 or more years to complete a project. Between shaky memories and new team members, the agreed upon process can be lost. Setting up a series of documented workflows helps everybody involved know what's expected and how to move between platforms.