

# Autodesk® Project Vasari: Playing with Energetic Supermodels

Matt Jezyk & Zach Kron – Autodesk

**AB9660-L** In this hands-on lab, you will explore experimental tools and workflows using Autodesk Project Vasari. Project Vasari connects the parametric modeling capabilities of Autodesk Revit® with many of the analysis and simulation capabilities available in Autodesk Ecotect® Analysis and Autodesk Green Building Studio. You will also learn about and use new performance-based design tools available from Autodesk Labs. First, you will create a few parametric building models and simulations. Next, the class will cover more advanced topics, such as how to create automated feedback loops. You will explore workflows where changes you make to the model cause changes in the simulation results, which then drive changes back into the model. You will use both out-of-the-box tools and add-ons currently under development to create parametric building models that respond to environmental conditions through both automatic and semi-automatic feedback loops.

## Learning Objectives

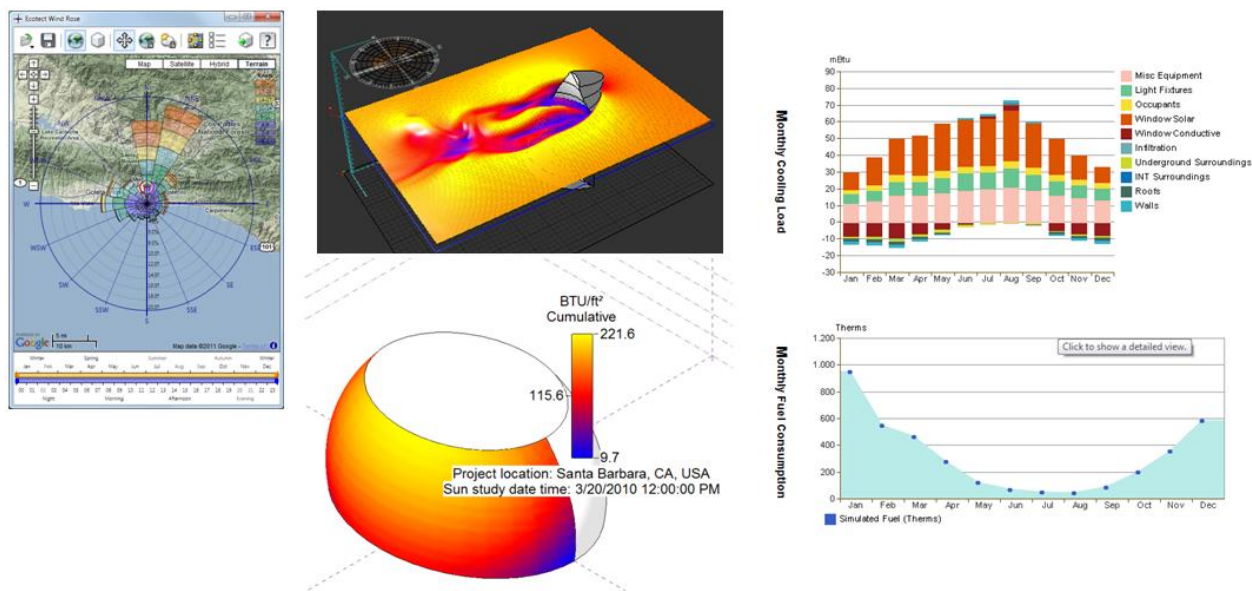
- Describe new types of analysis and simulation that are now accessible to building designers
- Combined parametric modeling with analysis to help drive decisions early in the design process
- Create feedback loops to enable analysis results to make changes to the building model
- Use advanced adaptive components and curtain panel techniques

## About the Speakers

*Matt Jezyk has been in the Architecture and Engineering industry for 16 years and has spent the past 12 years developing Autodesk Revit. He was one of the original architects hired by Revit Technology Corporation, pre-acquisition by Autodesk, and helped build Revit Architecture and Revit Structure. He is experienced in building parametric and geometric modeling tools for architects and engineers. Recently, Matt has focused on emerging markets and technology. His groups have designed and developed new parametric modeling and integrated energy analysis features. Most recently, he has lead the team working Project Vasari, a new technology preview available on Autodesk LABS.*

*Zach Kron is an architectural designer and a quality assurance analyst for Autodesk. For the last five years, he has worked closely with Autodesk® Revit® Architecture product designers and developers creating and testing the rendering, modeling, and analysis tools. In addition to internal teaching and curriculum development at Autodesk, Zach has helped create and run workshops at MIT & ACADIA, and maintains a personal blog. Before joining Autodesk, Zach worked as a designer in several Boston-area architectural offices on projects ranging from furniture to bridges. He has more than fifteen years of professional experience in form-making and visualization.*

*<http://buildz.blogspot.com>*



## Introduction: New types of analysis and simulation that are now accessible to building designers

### High Level Problem

Designers who are using local climate data to inform their designs sometimes have an enormous quantity of tabular or graphical data on their particular site and building form. The difficulty comes in finding avenues to rapidly explore the implications of this data for their buildings or to use the data itself to drive building form and detail level responses. After geometric reactions are derived from climate data, the designer often wants to verify the efficacy of changes by cycling this geometry back into an analysis application which often poses another set of data transfer problems.

### General CAD Solution

In CAD applications, one approach to this issue is a workflow where data that is either generated in an external analysis package (such as Ecotect, Equest, or any other number of applications) or derived from other external data sources is interpreted by a custom script to define points, lines and planes in space. More detailed building geometry can be hung on this skeleton at this time, but this requires substantial commitment to the current design and will require rework if the design changes. In order to go back to an analysis application to verify the efficacy of design decisions, this geometry often needs to be abstracted back to a format that can be consumed by the analysis package, and the existing geometry will need to be jettisoned and replaced with new geometry and documentation.

### Vasari Solution

Vasari has several built in analysis engines (Whole building energy analysis using DOE2, CFD "Wind Tunnel", Incident Solar Radiation) and access to various forms of local climate data. For many projects, this eliminates the need for external analysis packages. Because Vasari geometry is made up of analytic surfaces and solids, and that every element is a representation of an underlying relational database (as opposed to simple connected points in space) the analysis engines can mine the model geometry without the need for further cleanup or processing. The result of this is an ability to quickly cycle between climate analysis, geometric responses, and design validation, all within the same environment. At any time, this geometry can be used to host detailed building design and documentation in a standard Revit workflow.

### About the current prototypes

While the current Vasari environment allows for a much tighter relationship between analysis and design reactions, there is a need for designers to do more rapid and generative responses to climate data. With the prototypes explored in this lab, we will experiment with possible advantages and efficiencies to be had by leveraging custom Vasari families and standard parametric tools with built in analysis tools to make design moves that are either very cumbersome or impossible with standard tools.

Please keep in mind that these examples are very early stage proof of concepts and suffer from the fragility and limited range of pre-Alpha level software. What we wish to validate is the flexibility and robustness of using established analysis standards with the customizable aspects of basic parametric family behavior.

### Combine Parametric modeling with analysis to help drive decisions early in the design process: Authoring a Parametric Model

Revit and Vasari families are more than simple geometry. When they are properly constructed, they can contain and respond to a wealth of data. They can know their orientation, what they are connected to, what their dimensional aspects are, and so on. Most importantly for our current exercise is their ability to "listen" to external data and respond to it in an intelligent manner. While there are many ways to author intelligent families, a few things to always keep in mind

### Scale

What Scale of building intervention are you working on? Site and whole building mass explorations may require different strategies than paneling, shading, or other repetitive element designs. Although it is not always the case, Mass category should usually be chosen for

building, site, and planning scales, and detail elements should be created in panel and generic model templates.

Downstream use should also be considered. Do you want the models to create multiple design options or refine a single concept. Will the elements become detail elements in a final design, host final design elements, or are they throw away experiments?

### Rigging

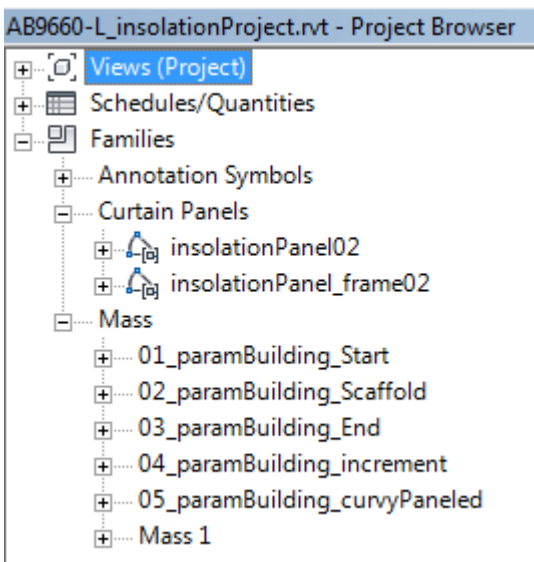
Rather than defining building form, think about modeling the constraints and relationships that will define the final building and building element forms. By creating frameworks and scaffolding upon which to hang actual geometry, you can create more explicit and flexible relationships and drivers for geometry and move quickly between design options.

### Data Receivers and Responses

What external data does your family need to listen to? Do you need a single piece of detail geometry to respond many times to slightly different inputs, or a whole building respond to a single data point that has not yet been defined? Is the dataset large, require multiple iterations to process, or otherwise complex? Is it relatively simple and could it be defined manually for each instance.

### Example Datasets

For this Lab, all the files we will use are contained in AB9660-L\_insulationProject.rvt.



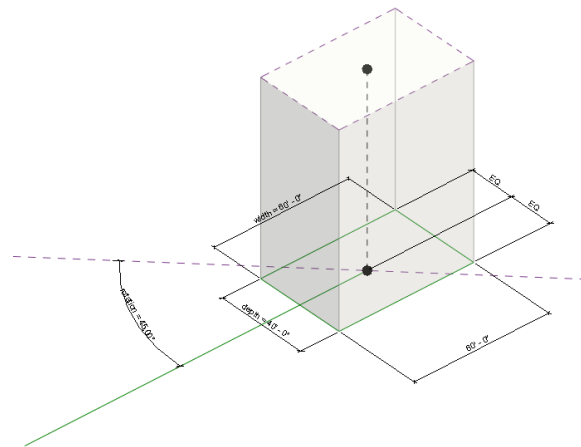
## Feedback loops: use analysis results to make changes to the building model

### Changing Form and Orientation Semi Automatically and Automatically

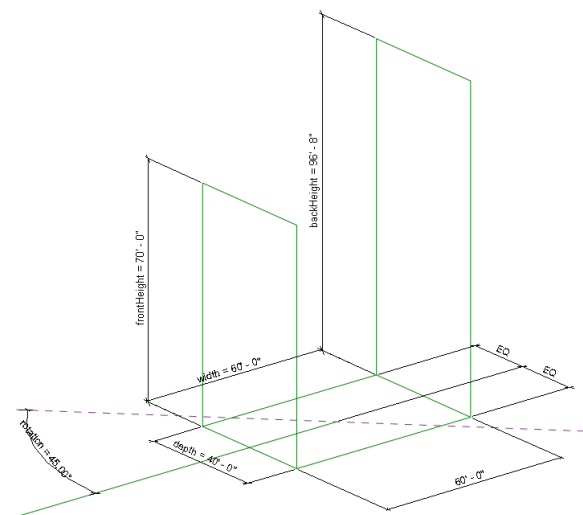
As a case study, we will look at a 20,000 sqft building in an urban area. The local climate and state incentives support the use of alternative energy sources to offset building energy consumption. We want to maximize the incident solar radiation on the roof of the building to make the most of a photovoltaic array, but because of some overshadowing by neighboring buildings, it isn't obvious where the "sweet spot" is for solar orientation.

### Manual Form Orientation and Configuration with Parameterized Geometry

**01\_paramBuilding\_Start:** The overall building size must always accommodate 20,000 square feet, but the shape of the building is not yet defined. This family establishes the basic set of constraints, by allowing a full 360 degree rotation of the building form, and maintaining a constant volume for the building, with height being driven by building footprint. This however does not allow us to explore the requirements of rooftop solar optimization



**02\_paramBuilding\_Scaffold:** Using the same basic skeleton of 01, we can now manipulate the angle of the roof while maintaining the overall square footage of the building. Add parameters for the front and back elevation heights for the building and create a new building form.



Reload (or use **03\_paramBuilding\_End**), activate Solar Radiation, analyze the roof, and manually find the optimal angle up and down, North/South/East/West by manipulating parameters and watching the analysis output.

The screenshot displays a software interface for solar radiation analysis. On the left, the **Properties** panel shows details for the selected mass **03\_paramBuilding\_End**.

Constraints	
Offset	0' 0"
Work Plane	Level : Level 1

Dimensions	
rotation	-35.000°
width	60' 0"
height	83' 4"
depth	40' 0"

Mass Floors	
Gross Floor Area	
Gross Surface A...	21693.03 SF
Gross Volume	200000.00 CF

Identity Data	
Comments	
Mark	

Phasing	
Phase Created	New Constru...
Phase Demolis...	None

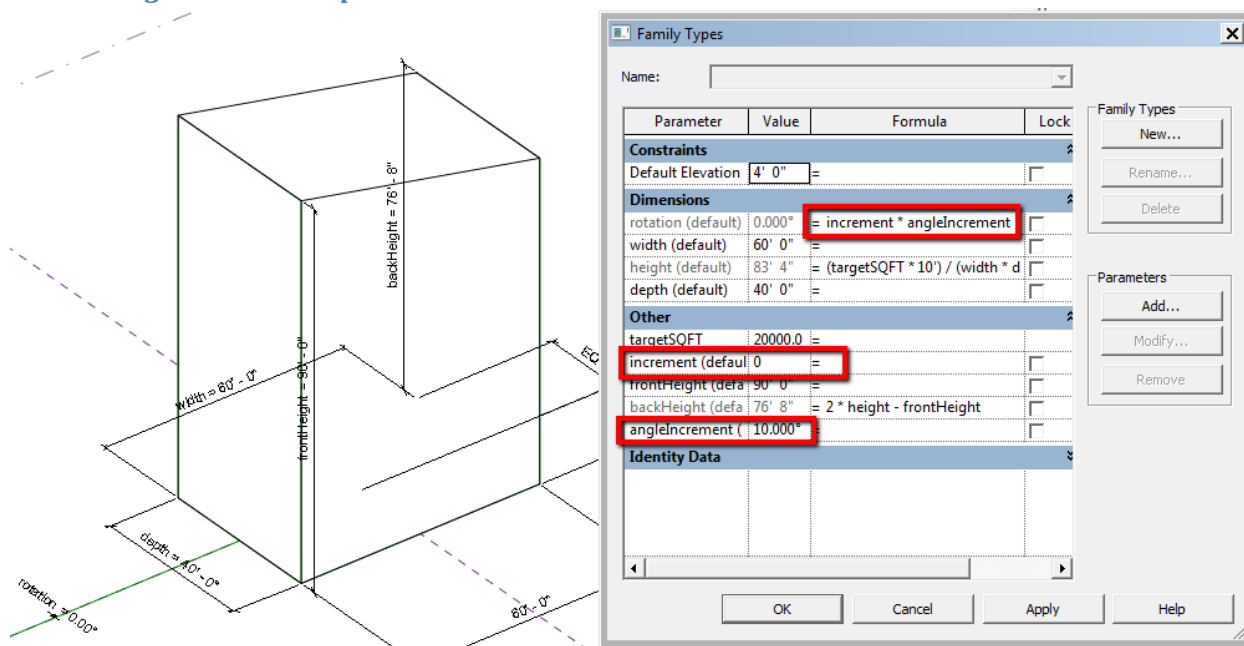
Other	
frontHeight	70' 0"
backHeight	96' 8"
Schedule Level	Level 1

The central 3D view shows a building model with a roof surface color-coded by solar radiation intensity. A color scale legend indicates values from 85081.0 (blue) to 206396.4 (yellow). A specific value of 145738.7 is highlighted on the roof. Project information is displayed at the bottom of the 3D view:

- project location: Boston, MA
- start date time: 1/1/2010 10:00:00 AM
- end date time: 12/31/2010 4:00:00 PM

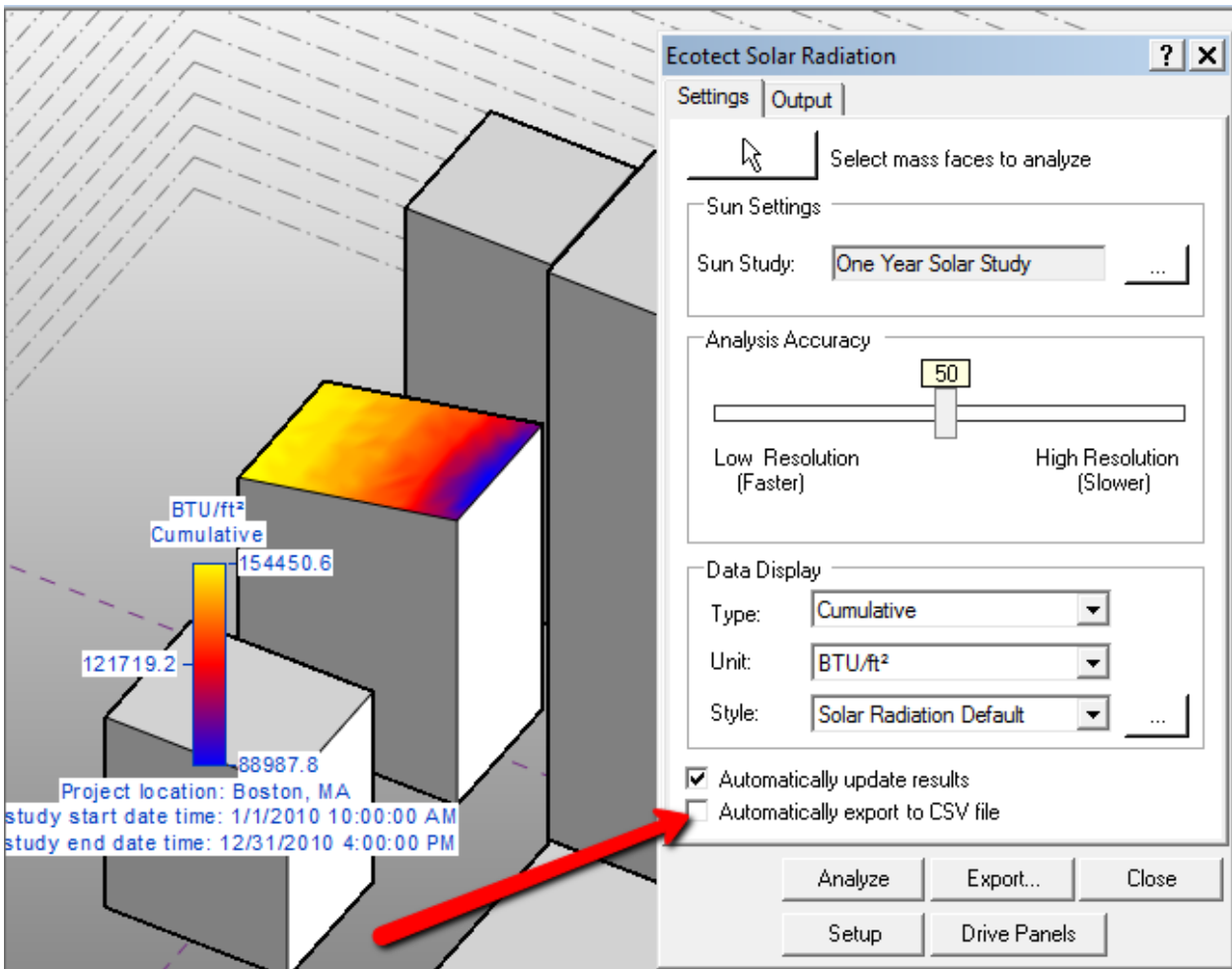
On the right, the **Ecotect Solar Radiation** dialog box is open, showing the **Settings** tab. The **Sun Settings** section includes a **Sun Study** dropdown set to **One Year Solar Study**. The **Analysis Accuracy** section features a slider set to **50**, with **Low Resolution (Faster)** on the left and **High Resolution (Slower)** on the right. The **Data Display** section shows **Type** set to **Cumulative**, **Unit** set to **BTU/ft²**, and **Style** set to **Solar Radiation Default**. Checkboxes for **Automatically update results** (checked) and **Automatically export to CSV file** (unchecked) are present. Buttons for **Analyze**, **Export...**, **Close**, **Setup**, and **Drive Panels** are located at the bottom of the dialog.

Automating orientation optimization



**04\_paramBuilding\_Increment.** This model is similar to 03\_paramBuilding, except if you change the increment parameter it will rotated the building however many degrees are defined by another parameter, incrementAngle. We will use Dynamo to iteratively rotate the building and find an optimal orientation relative to the sun while taking nearby building overshadowing into account.

After replacing the existing family with 04\_parambuilding\_Increment, start Solar Radiation and create an analysis for the roof.



### *Creating an external spreadsheet of solar radiation values*

Solar Radiation has the built in capacity to export tabular data for your analysis runs, writing the xyz coordinates and normal vector of the point sample as well as the insolation value (in BTU). We have added the ability to automatically update this file every time the analysis run changes. Later we will set up a tool to monitor these values and update the model. For now, enable the automatic update and identify a place on disk to write the file.

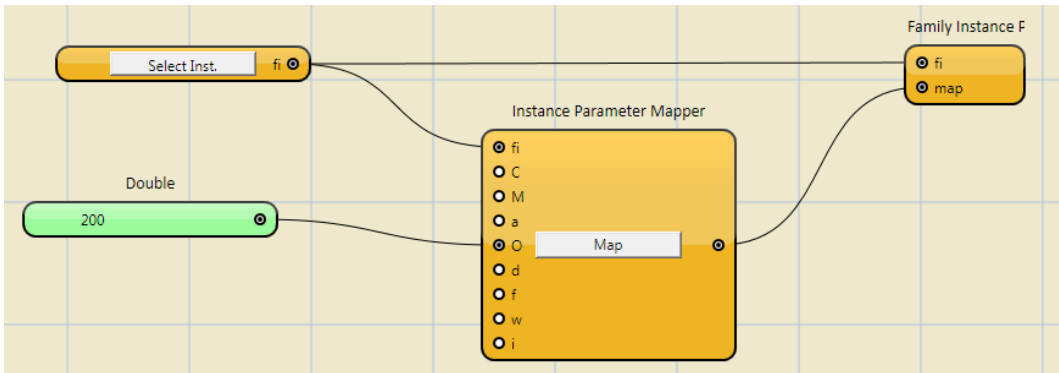
### *Start Dynamo*

In the Add-ins menu, select Dynamo. (Dynamo is a project started by Ian Keough to provide a code playground for building interesting parametric functionality on top of that already offered by Revit, and to do so with a graphical interface that allows you to share your work with others less inclined to write code themselves.)

From the Add menu you can drop actions, family component representations, and associations into the "workbench" to create simple or complex interactions for Revit and Vasari elements. Right click and drag on components to move them, left click from the output (right) side of a

component to connect it to the input (left) side of another. You can zoom in and out using the mouse wheel. If you fall behind in the exercises, you can use File>Open to browse to the files that stage the Dynamo workbench creation process.

### Create a Representation of a Placed Family Instance inside of Dynamo

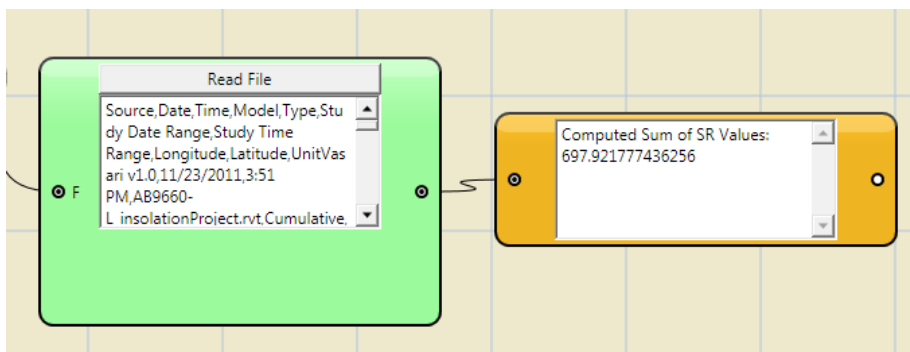


- Family Instance Creator by Selection: represents a previously placed family
- Instance Parameter Mapper: displays the instance parameters of a chosen family
- Family Instance Parameter Evaluation: resolves changes to the instance parameters of a family that may be different than their default settings.
- Double: a numerical value

These components allow you make a graphical representation of a placed family inside of the Dynamo interface. From here, you can manipulate the family and associate it with other components. By clicking on the Family Instance Creator by Selection, you can now select the placed 04\_parmbuilding\_Increment. After the family is selected, Click “Map” on the Instance Parameter Mapper to populate the parameters.

*Open 01\_Create placed family Representation to catch up to this step.*

### Connect to External Data



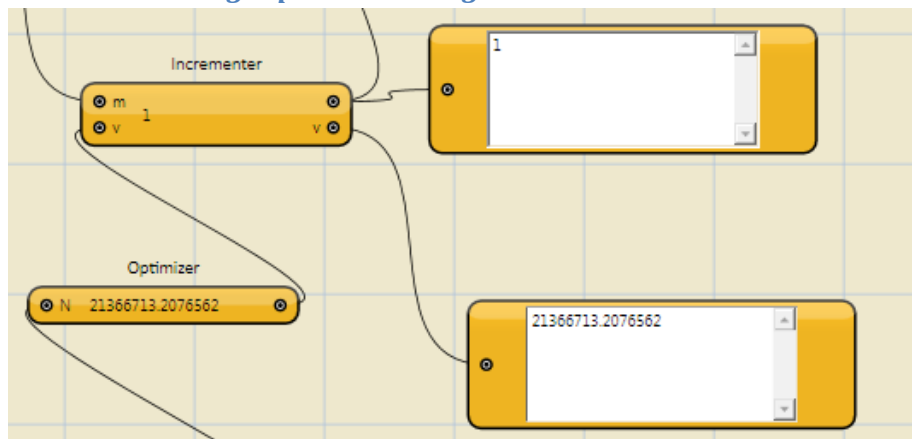
- Read and Watch Data from File: browse to and read a file on disk

- Extract Solar Radiation Value: computes the sum of insolation values in the specific file configuration used by the Solar Radiation .csv export

Any external file can be referred to by browsing to it, and the specific file we are using is a csv spreadsheet generated by Solar Radiation. The insolation values in that file are then summed to allow for use in our optimization process.

*Open 02\_Connect to External Data to catch up to this step.*

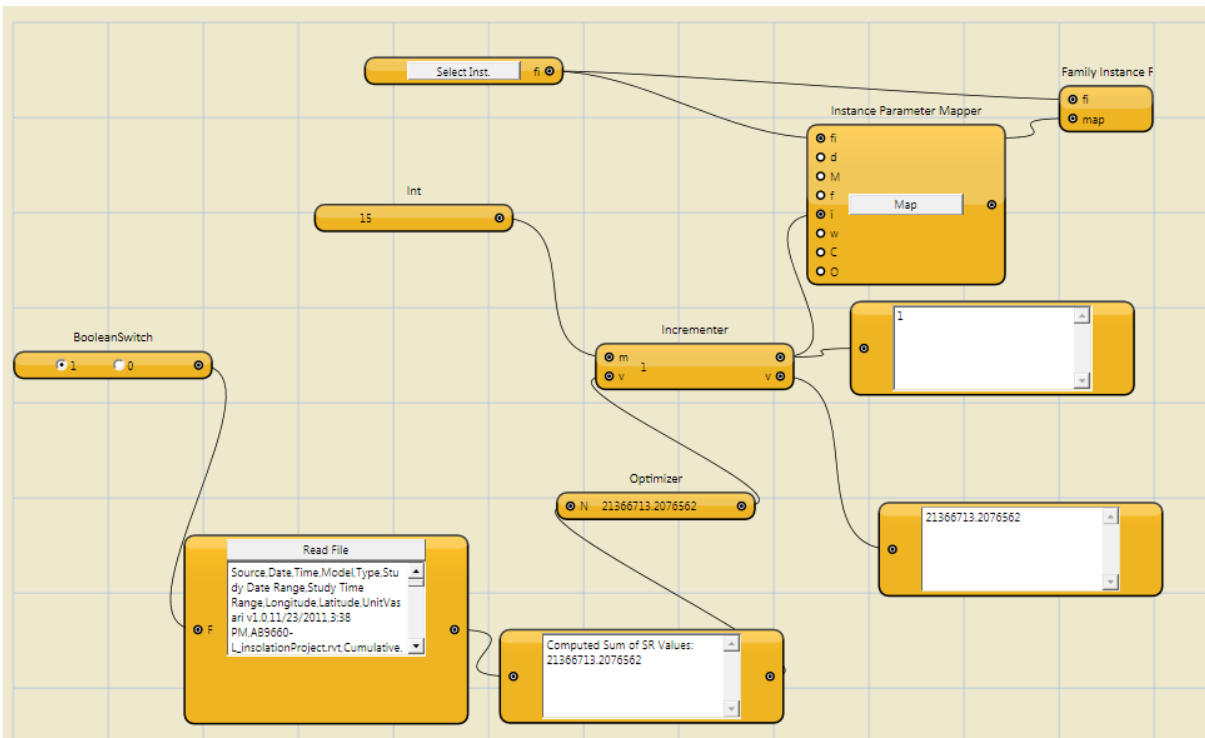
### Add "Hill Climbing" optimization logic



- Optimizer: evaluates an input number against a last known number. If the input is greater than last known, it passes the number to the output.
- Incrementer: for any upstream change, passes an increasing integer value for each time it is changed for m number of changes.

These components act as gatekeeper, only allowing downstream changes that increase the solar radiation values to pass through. When larger insolation values arrive at the Optimizer, it prompts the Incrementer to advance another step.

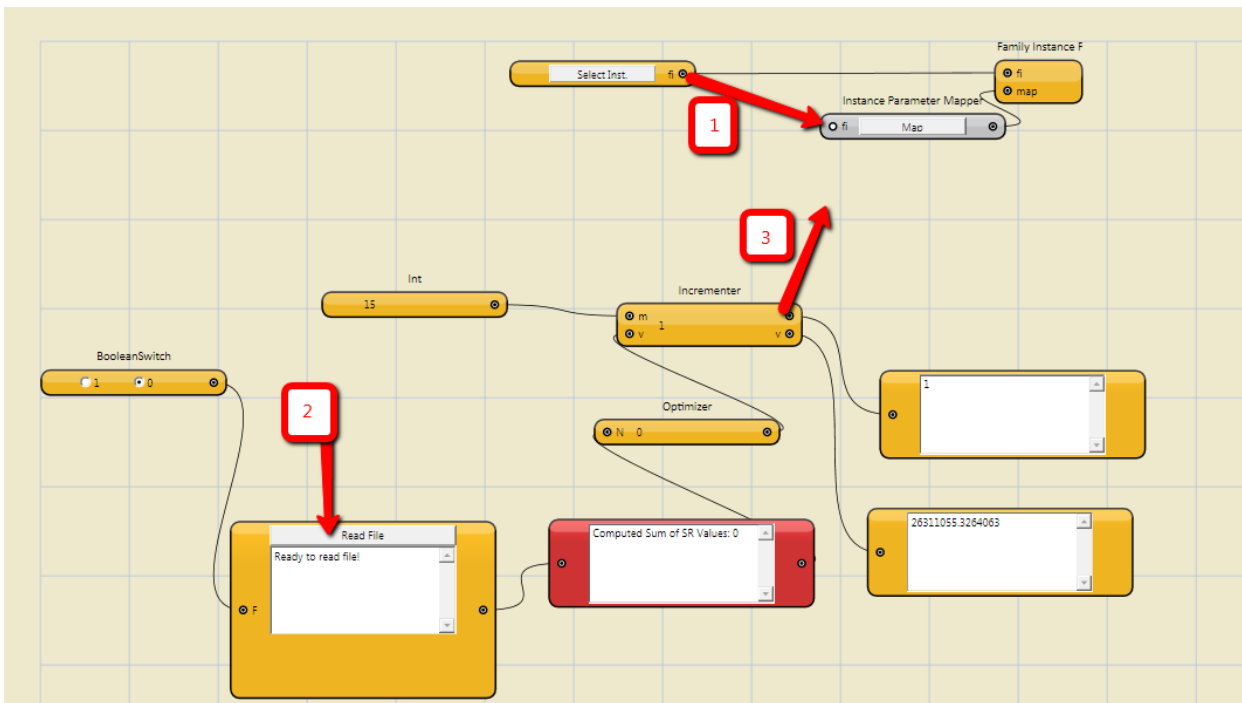
### Connect the Functional Areas and add Watchers



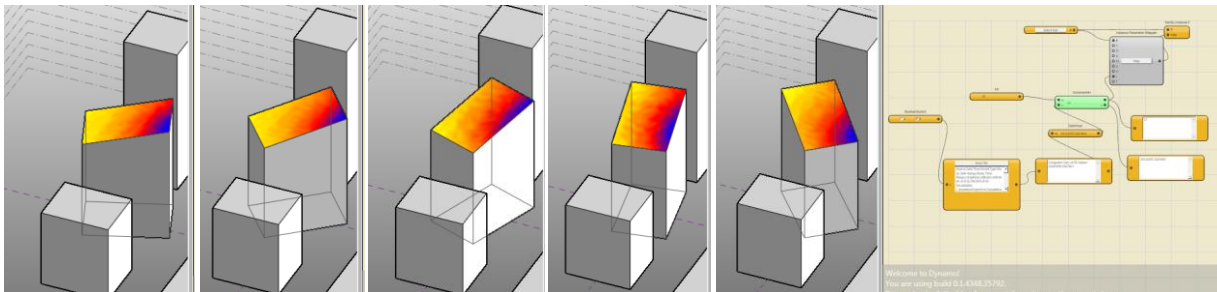
Wire all the components together, with the idea in mind that data is flowing from the Solar Radiation CVS file to the Optimizer and Incrementer, then into the Family.

- Connect the output from the Extract Solar Radiation Value to the input for the Optimizer
- Connect the output from the Incrementer to the “I” value on the Instance Parameter Mapper, which is the “increment” parameter.
- From the Add menu, add a couple of “Watcher” components and wire them to the output to the Incrementer, this allows you to monitor the otherwise difficult to see outputs.
- Finally, from the Add menu, get a Boolean Switch wire this into the Read and Watch Data from File. The “watch” part of this component is not working and it needs to be nudged to refresh the state of the file.

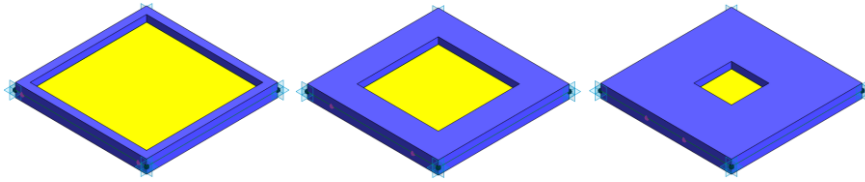
*Open 03\_Hill Climbing Logic and Watchers to catch up to this step. You will have to update certain parts of the definition to match the desired result. 1. Select the Family Instance in the Vasari screen and update the Map button. 2. Browse to the solar radiation file on the hard drive. 3. Wire the Incrementer to the “I” parameter.*



By clicking the Boolean Switch you should be able to propagate changes through the data structure and more the family at 15 degree increments until it finds the optimum incident solar radiation. When it has found the highest area, it will stop moving when the Boolean Switch is clicked.



This example is a relatively simple example of a "hill-climbing" optimization strategy. In a more complex situation, you would need to also control for local maximums. When the details of dynamic model update are figured out, the need to manually update the file read from disk will be removed.

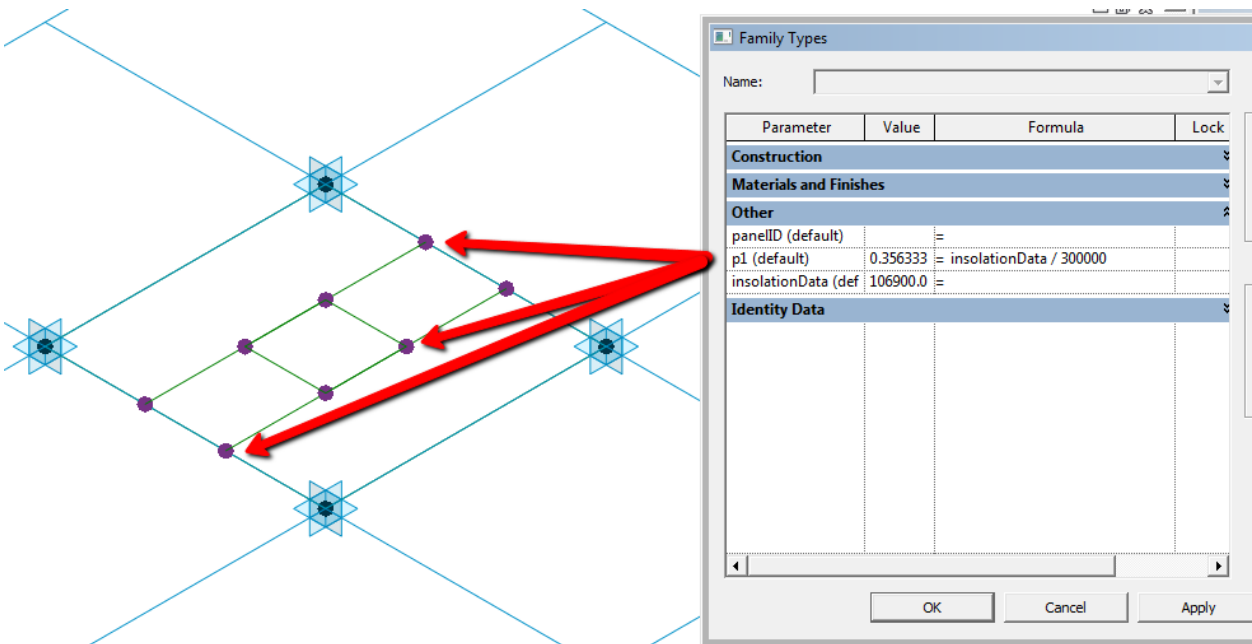
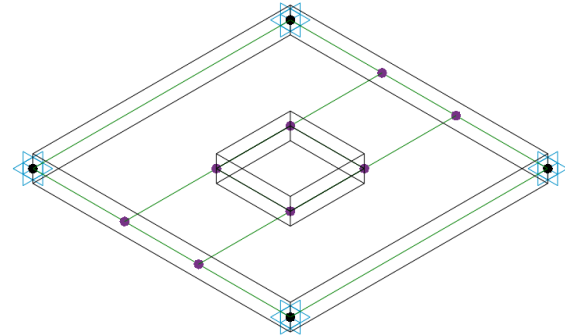


## Curtain panel techniques: Driving Building Facades with Solar Radiation Data

We have established a desired orientation and shape for our building and we want to manipulate the façade to respond to the specific solar conditions of the site.

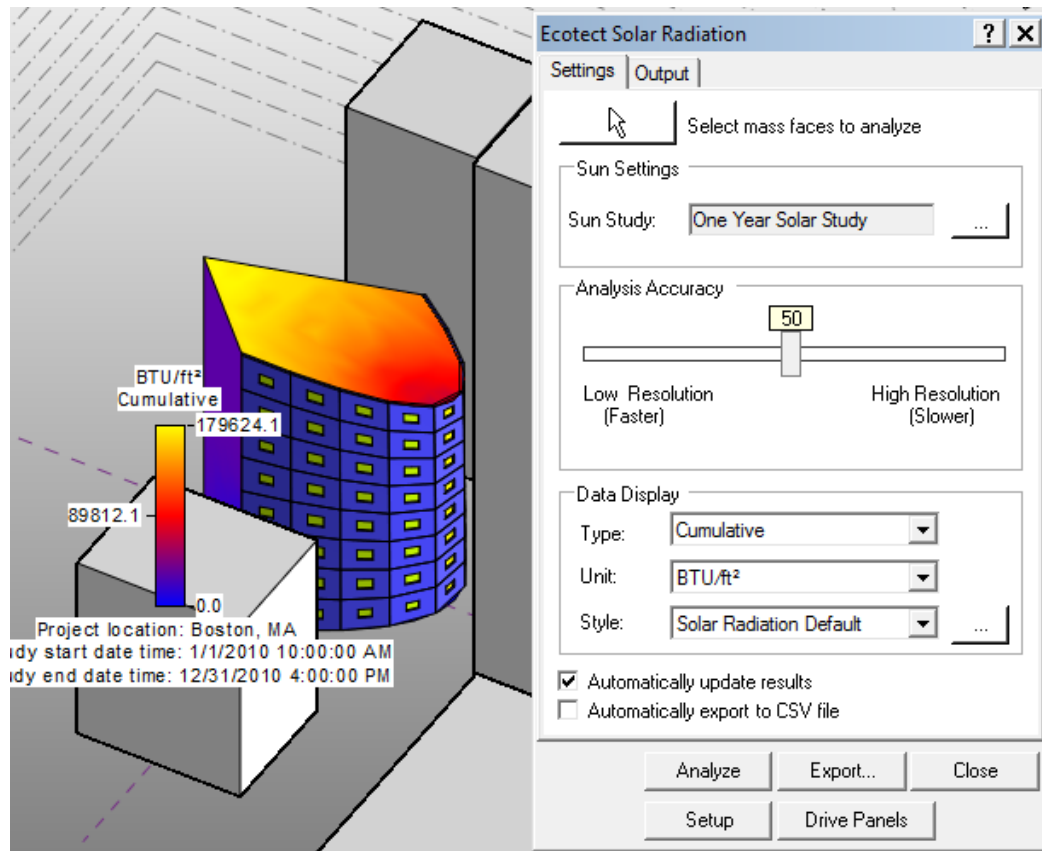
### Curtain Panel by Pattern family insolationPanel\_frame02.

In the family properties for InsolationPanel\_frame02 you will see that the aperture size of the window is driven by a parameter called insolationData. Change the values of insolationData to see the results.



We are going to populate all the panels on the building façade with individual solar radiation values. Our panel is going to "listen" for this data with the insolationData parameter, and take action based on the formula driving the aperture parameter.

**05\_paramBuilding\_CurvyPaneled.** Replace the existing family with 05\_paramBuilding\_CrurvyPaneled. This version is built on the previous examples but has a more interesting surface to work on. It is also panelized with an 8x8 grid. While a real building would likely have many more panels, this surface will respond more quickly and demonstrate the principles for the context of the Lab. Start Solar Radiation Analysis and select all the surfaces of the loaded family by box selection, complete the selection to create an analysis (only select the loaded family faces, not the surrounding context)

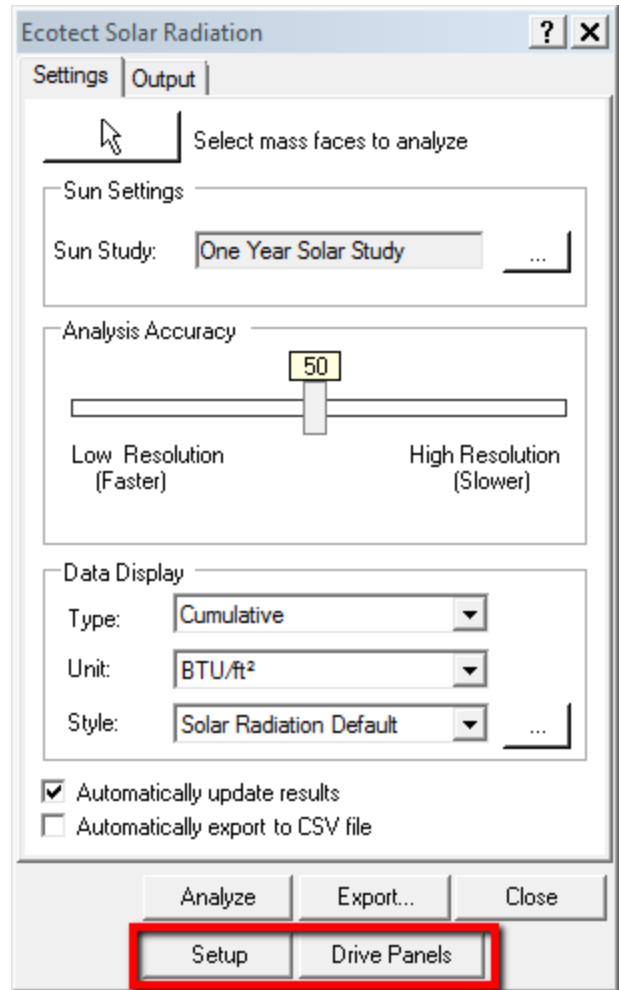
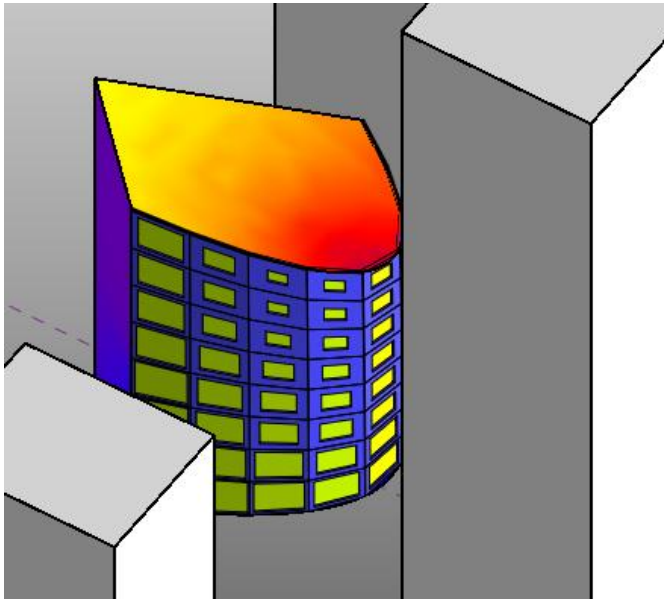


### Setup

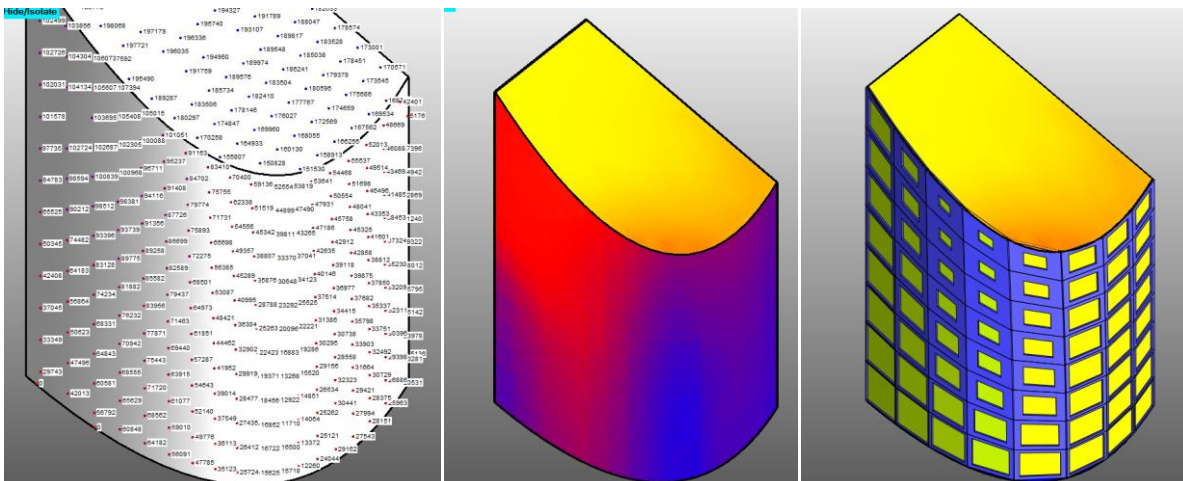
After running solar analysis, pressing Setup will prepare the panels to have a large amount of data written to them. Because panels exist as nested families inside of the loaded family, you can't write to them directly, but the application will create a conduit from the project to the family instance by creating a parameter in the mass family for each panel. For instance, there is a panel instance A hosted on a divided surface in the rfa file. Select the instance, and associate a new parameter in the host file to this instance. This is exactly what the application does through the API for each panel instance, whether there are 2 or 200. While it gets the job done, creating all these parameters is not ideal. However, until the Divided Surface exists in the project proper or there is API access inside of in-place masses, this was the best option. We could do a similar operation with standard curtain system families and avoid the extra parameters as well.

### Drive Panels

Now that there is a conduit from the project to the panel instances, Drive Panels will find the average incident Solar Radiation associated with each panel instance and write this data to the insolationData parameter, which will in turn drive the panel geometry.



A note on the accuracy of this functionality: The current implementation writes data with a bias down and to the left, as it takes a solar radiation sampling based on the origin of the panel, which is the first panel point and not the panel centroid. As a result, you will see a shift in the data up and to the right.



The Family parameters can be manipulated (height, width, rotation changes) as well as the sun study period and pressing the “Drive Panels” button will update the panel apertures accordingly.

## Conclusions

These examples demonstrate a possible hybrid application of custom tools and standard smart Revit/Vasari components with Vasari’s Analysis tools. While these tools are more than a little rough around the edges, we hope that they are convincing indicators of several things:

- Standard Family creation tools provide a generalizable interface for receiving and reacting to analysis data
- The analysis tools available in Vasari can be used to drive design responses to environmental conditions
- Smart families and built in analysis tools can be made to speak to each other through various methods. Dynamo in particular seems to be a promising way to leverage the existing strengths of Vasari and the Revit platform in general.