



Synchronizing Thermal and Lighting Analysis in Performance-Based Design

Kaveh Alagheband - PhD student, Teaching Assistant, Research Assistant, UNL

AB6905

Revit software and Green Building Studio web-based energy analysis service provide us with in-depth and overwhelming data in relation to building performance. Currently, however, organizing the data and decision-making is mostly a user responsibility. For example, although Revit software's full analysis provides us with lighting and thermal analysis, the results are not synchronized. To be more exact, the lighting part of the full analysis uses lighting power density (LPD), which is only related to the area and disregards other features such as window properties (even window size) or orientation, and so on. Using the Dynamo visual programming language extension and cloud-rendering services (illuminance calculation is currently a render operation) provided by Autodesk, Inc., it is possible to gather and organize illuminance values for the internal space. From there it is easy to calculate the internal lighting load, and accordingly it is possible to achieve a fully concerted thermal and lighting analysis, which is crucial in decision-making (especially in the design phase).

Learning Objectives

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

- Explore the energy-performance capabilities of Revit software and GBS
- Use rendering for gathering Illuminance values
- Use the Dynamo extension to read illuminance data and also to perform simple math operations
- Organize the full analysis and make conclusions

About the Speaker

Kaveh Alagheh is a PhD student at University of Nebraska-Lincoln, where he teaches Autodesk products to other students. He has also been involved in extensive research about the energy-saving potential of architectural forms. Some of the results of that research include being featured in certain publications and winning the Sefaira Perform Competition.

[illegible]

Figure 1: Demographic Data

Gender Distribution

Gender	Count	Percentage
Male	10	50%
Female	10	50%

Age Distribution

Age Group	Count	Percentage
18-24	10	50%
25-34	10	50%

Education Level Distribution

Education Level	Count	Percentage
High School	10	50%
College	10	50%

Marital Status Distribution

Marital Status	Count	Percentage
Single	10	50%
Married	10	50%

Employment Status Distribution

Employment Status	Count	Percentage
Employed	10	50%
Unemployed	10	50%

Potential Energy Savings

Figure 10: All Analyzed Building Features. This horizontal bar chart displays the current value (in %) for various building features. The y-axis lists the features: Plug Load Efficiency, Lighting Efficiency, Window Glass e-UC, Window Glass, Wall Insulation, Occupancy Sensors, Roof Insulation, Daylight Glass e-UC, Daylight Controls - DC, Daylight Orientation, and Daylight Glass. The x-axis ranges from -50% to 50%, with 0% in the center. A vertical line at 0% is labeled 'Current Value'. Bars extend to the left (negative) and right (positive). A green circle with '2' is on the right side of the chart.

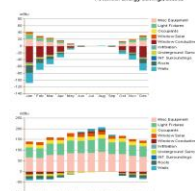
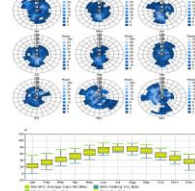


Figure 1 consists of two charts. The top chart shows the probability of infection (y-axis, 0 to 1,400) versus the number of days between the first and second COVID-19 tests (x-axis, 0 to 14). The bottom chart shows the probability of a positive test result (y-axis, 0 to 100) versus the number of days (x-axis, 0 to 14). Both charts show a U-shaped curve, indicating that the probability of infection and a positive test result is highest at 0 days and lowest at 7 days.

Figure 10 consists of two subplots. The top subplot is a line graph showing the 'Number of iterations' (Y-axis, 0 to 200) versus 'Iteration' (X-axis, 0 to 24). The data points are connected by a line, showing a fluctuating trend. The bottom subplot is a radar chart showing the 'Number of iterations' (Y-axis, 0 to 200) versus 'Iteration' (X-axis, 0 to 24). The radar chart has six axes, each representing a different iteration range. The legend indicates the following ranges: 0-25, 25-50, 50-75, 75-100, 100-125, and 125-150. The radar chart shows the distribution of iterations across these ranges for each iteration number.



Why Revit + Dynamo?

Perhaps similar approaches are employed in different software packages all the time, and they might be even simpler to work with. In Revit + Dynamo, however, not only can we work at the basic **conceptual level**, but also we are enabled to pursue our investigation into more **advances phases** of the project and we can control every small decision, the some of which may not be less important than the conceptual level. Also we can almost use real life properties for our **building elements** (such as walls, windows, etc.) and we also have all types of **parametric families** at our disposal. Adding **Dynamo** and the speed of **cloud services** there should be enough reasons to model more accurately in Revit. As everything is still developing, we might see, or create, the emergence of newer ways to do things that we can only do in Revit.

So at the heart of this investigation there is always two sides (corresponding to two types of analyses that we talks about earlier):

- *Conceptual Mass Modeling (think of what is usually called shoe-box modeling, with a low level of details)*
- *Using Building Elements Mode (for more advanced models)*

How?

In order for us to have a meaningful set of results, the process should be designed carefully. On the one hand, there should always be a 1. **Basis for the Comparison** as we would not want to compare apples and oranges, and on the other hand there are always 2. **Two types** of simulation, a lighting and a Thermal. Once we have the results in one place, think of charts, or numbers in an Excel sheet, etc. it is easier to make decisions.

This however, requires at least three steps:

- **Correct Modeling**

This is probably the most important part and it corresponds two both of the above rules. The models should be prepared with all the information needed both for thermal and Lighting analysis. At the same time some parametric logic should be built into the model in order for us to be able to iterate through some samples. We should also decide if we are working in Conceptual mode or in Building Element mode. Whatever our constant values (basis of comparison), we should make sure that we can exactly reproduce them in all our cases and that they appear the same on the result sheets.

- ***Management of information such as materials, location, weather data, zones, etc.***

In the current investigation, this might not be as important as the first part, as we want to keep them constant across the iterations. So while they should be correct and accurate, in most cases they remain constant. We should consider thermal (R, U, etc.) and lighting properties (reflectance, transparency, etc.) at the same time.

- ***Organizing the results in a useful way***

Once we have the results of our analysis, we can use a formula (a simple add function if they have the same units, kW for example) to gather everything together. We might also use, the cost of producing energy or any other similar thing that helps our goals the most.

Now that we know the process, let's get to work!

Parametric Families (System)

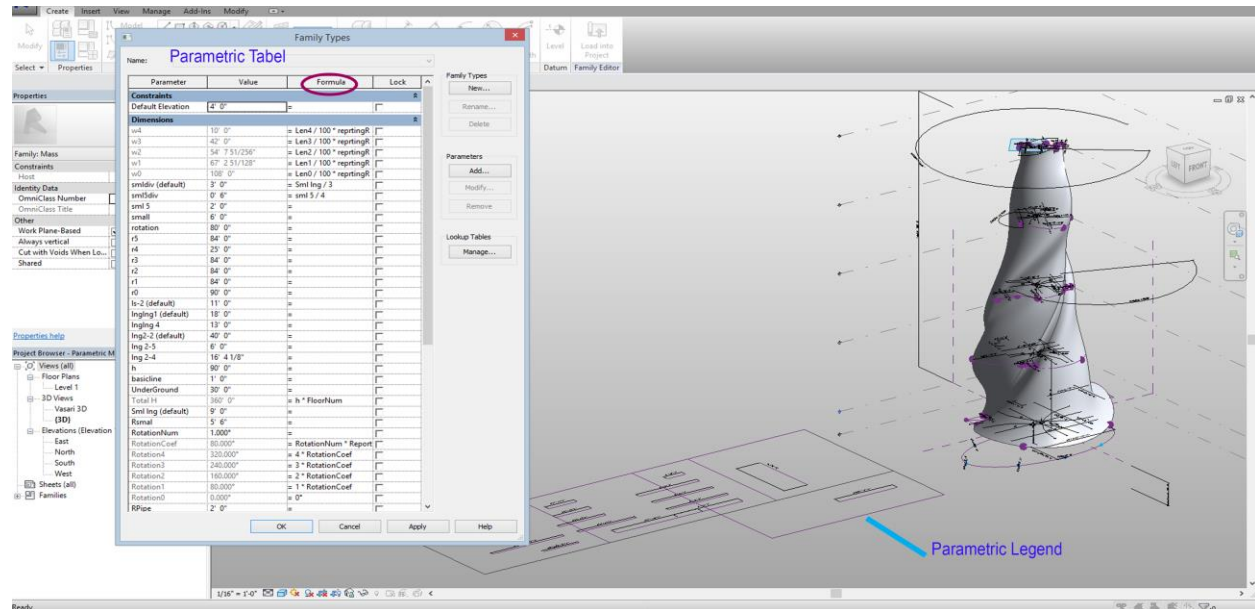
First things first! We have to design the variation producing parameter(s). Here we have a very important point.

Important Note: *since we are getting ready for analysis from the beginning we should note that when exporting a Revit model to gbXML, only **system** families, doors and windows are exported. All other families such as **Generic**, etc. are **missing** in the gbXML output. So if we want the Building element mode we should be really careful. In the conceptual mass mode we can still use usual mass families. Also in each case most probably we end up in Project space and we should be prepared for that.*

Are options are more varied when we use dynamo:

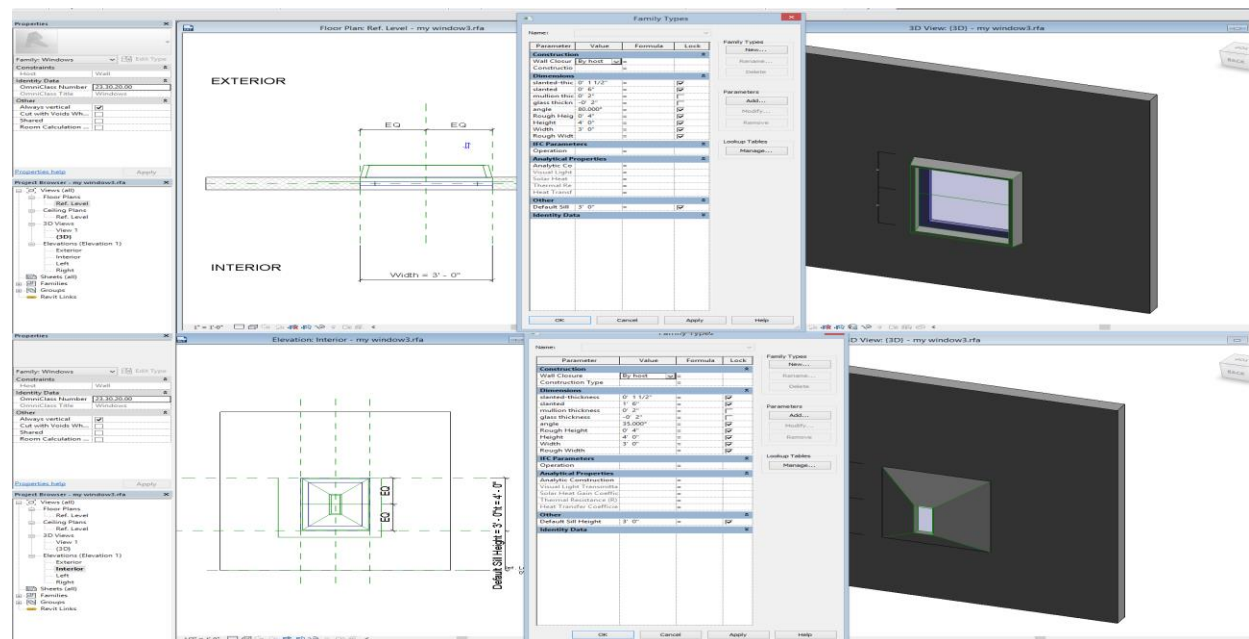
- *We can generate a family and we can access the parameters in Dynamo*
- *We can start by creating everything in Dynamo from the get go*

Here are some examples in each case:



Example one: Parametric Mass Family

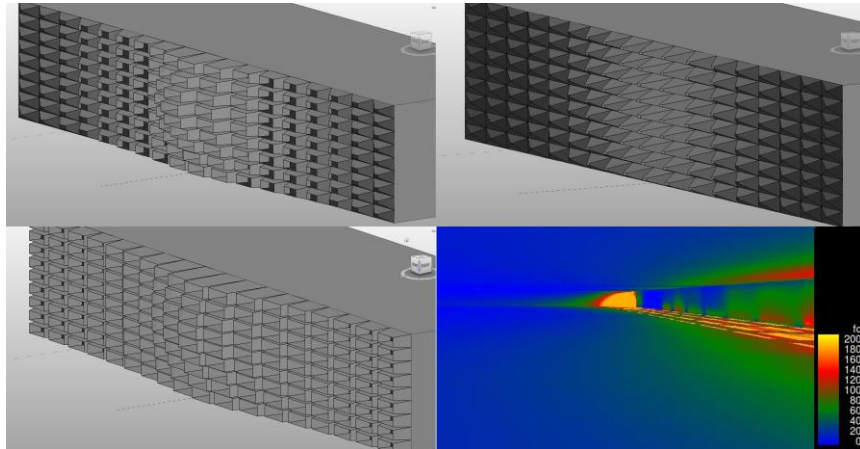
The above picture is an example of a parametric family where one is able to change the parameters on a legend or on the table (the usual method). The important thing here is using formulas to generate meaningful iterations.



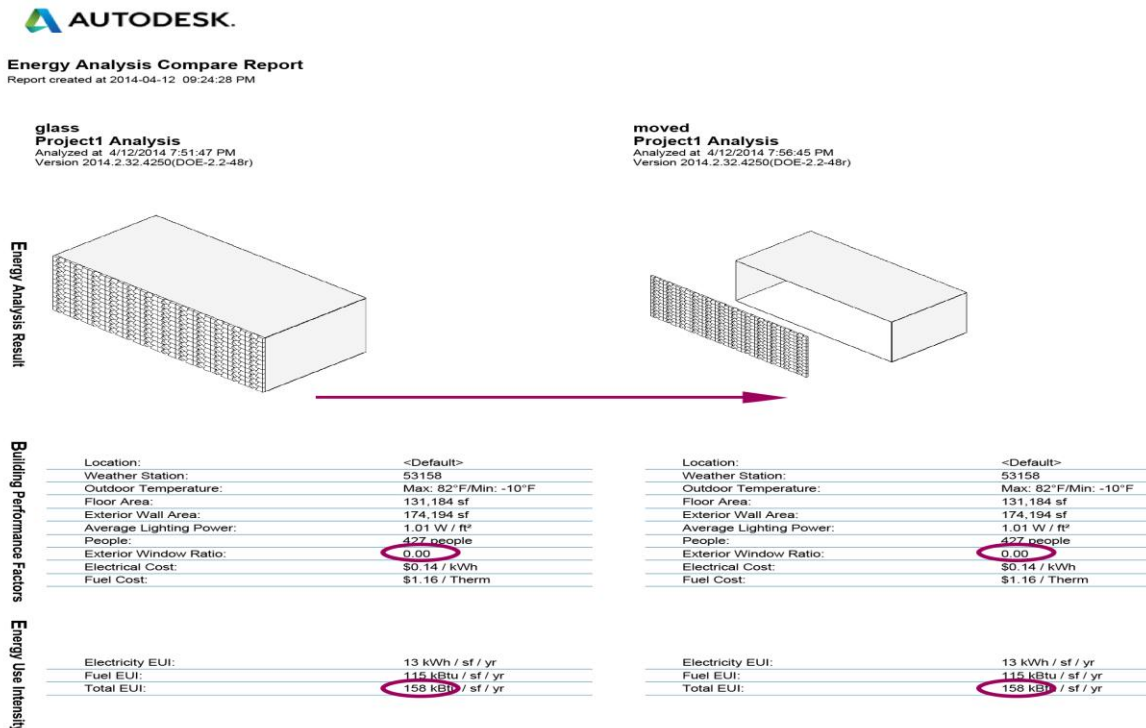
Example two: Parametric System Family

The second example is a system family (a window). We will see how we can reach the parameters later on, but note that:

- We have to have a goal in mind, for example the above window can be used to generate different lighting patterns (and different wvrs if needed) and it includes thermal and lighting properties (see next figure).



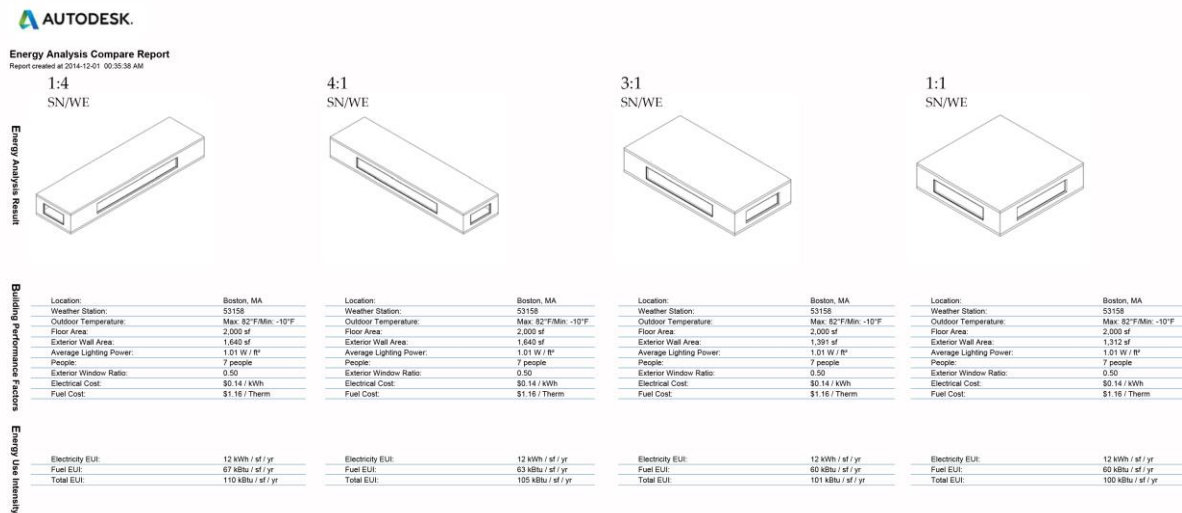
Note again that other family types won't work. Let's look at **Curtain Panel-Pattern Based Family** for example!



Failing Example

In this last example it is still possible to get lighting data (practically anything in the scene can be rendered, and hence analyzed in lighting) but the thermal analysis will fail. There will be no synch!

Full Dynamo Examples:



With the progress that we observe in Dynamo now generating everything inside dynamo is possible. Even at the element mode, some packages (Clockwork for example) give us some options in working with materials. At this moment, still a combination of parametric families (and the embedded materials) and Dynamo may be preferred. The above example is produced fully within Dynamo (Using Revit-Native Window family). Note that with simple mathematical relations, WWR, floor area and ratio are completely under control. Also note that at this level square performs the best.

Dynamo and the Packages

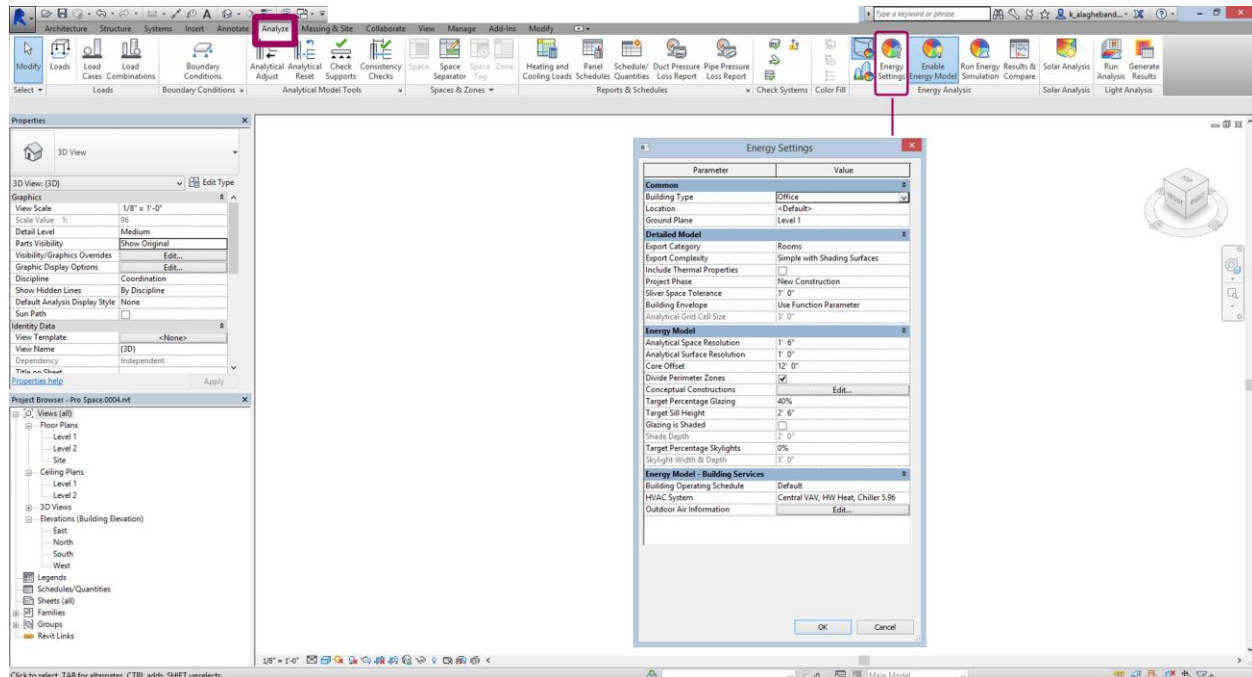
As we have already seen Dynamo can be very useful in recording a process. Dynamo can be used at two levels:

- Automating the processes that we usually employ in our approaches.
- Adding a level of control that is impossible, or really hard, to achieve otherwise

In the following examples I these packages are used:

- Energy Analysis for Dynamo
- Clock Work
- Lunch Box
- KOP

Most of what we have already seen deals with automation. Now we can start talking about enhancing the power of Revit + GBS. Let's consider a very basic energy analysis at the conceptual level. The following figure show some of the options available to us:



The next image shows some of the Nodes available to us in **Energy Analysis for Dynamo** package. This means that not only can we assign those parameters in dynamo, but also now we are enabled to change parameters such as shading depth or WWR on a per surface basis, while in Revit those parameters change globally.

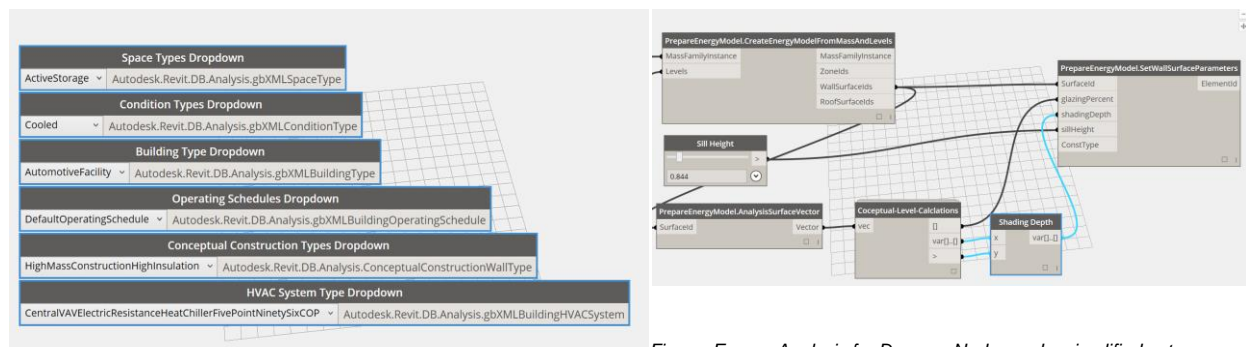
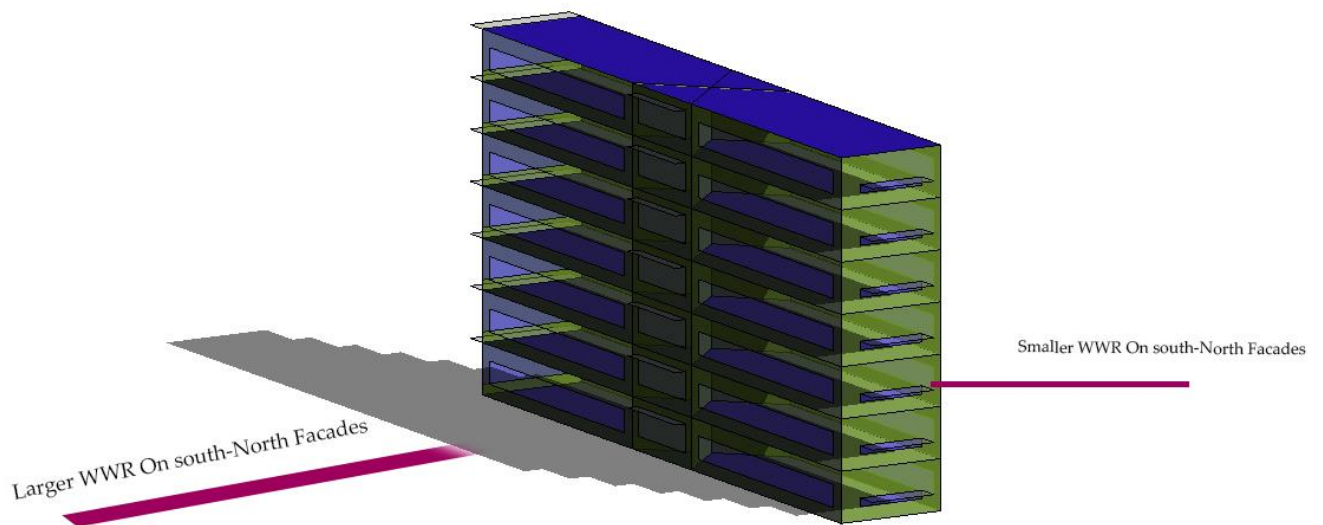


Figure: Energy Analysis for Dynamo Nodes and a simplified setup

Finally we can extract the result of our analysis and we can export it into excel charts for future reference. For more info on this visit: <http://core.thorntontomasetti.com/energy-analysis-for-dynamo-announcement-and-alpha-release>.

Synchronizing Thermal and Lighting Analysis in Performance-Based Design

To make it even more interesting, in each iteration we can capture a snapshot (or even a render) to include in our data set. Usual Dynamo workflow entails that we should make sure that the capturing (or data reporting) happens no sooner that the scene is updated. Transaction nodes and Function. Compose nodes are usually used for that purpose. The following series of images may illustrate the capabilities of this workflow. Note that WWR and shading can be varied on different surfaces of the same building.



{0.100000,1,10}.jpg



{0.100000,4,10}.jpg



{0.100000,7,10}.jpg



{0.300000,1,10}.jpg



{1.866667,1,10}.jpg



{1.866667,4,10}.jpg



{1.866667,7,10}.jpg



{3.066667,1,10}.jpg



{3.433333,1,10}.jpg



{3.433333,4,10}.jpg



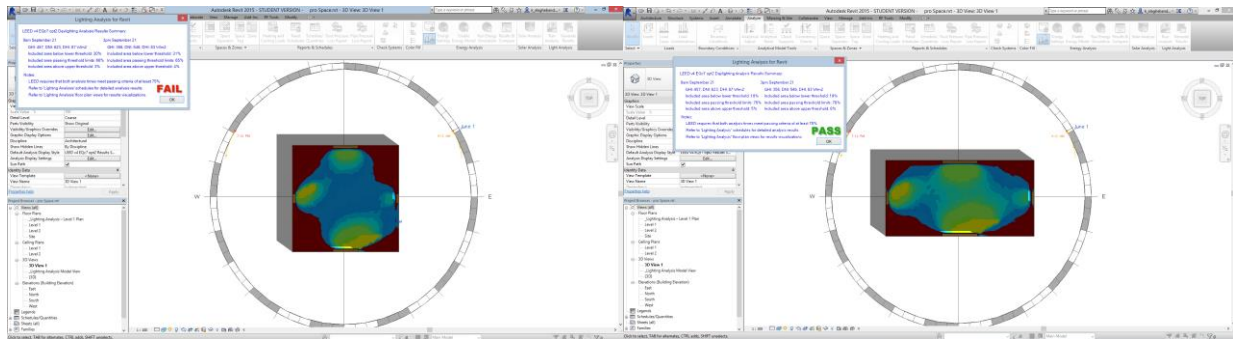
{3.433333,7,10}.jpg



{6.033333,1,10}.jpg

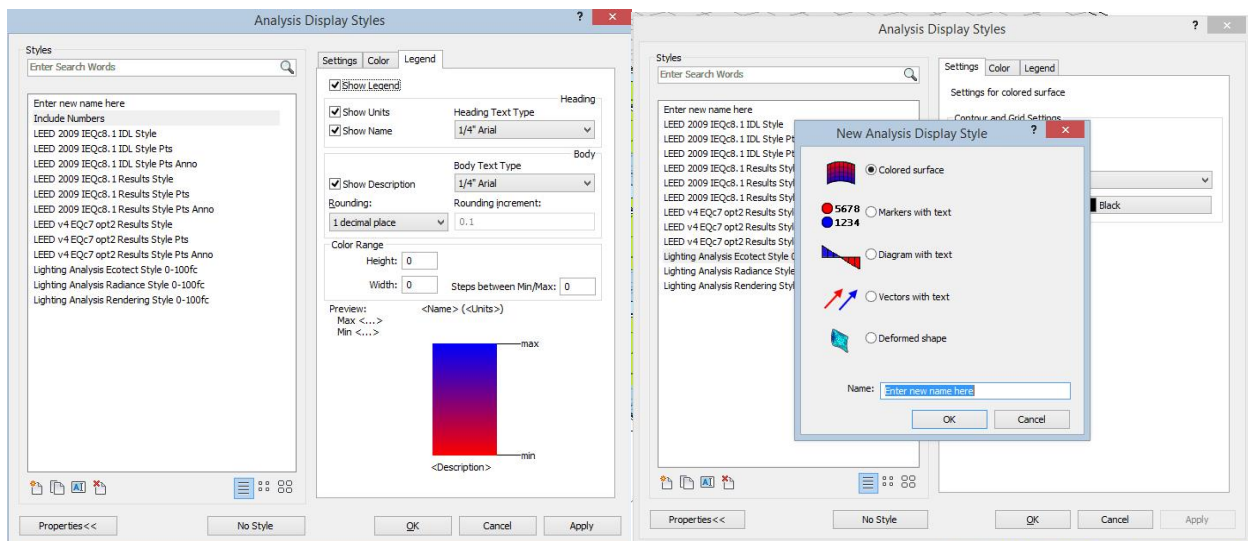
Include Lighting

If you remember (from page 7), a simple thermal analysis tend to favor shapes close to a square. That might not be true when we include lighting. We will get back to this with more explanation later on. But let's look at an example here.



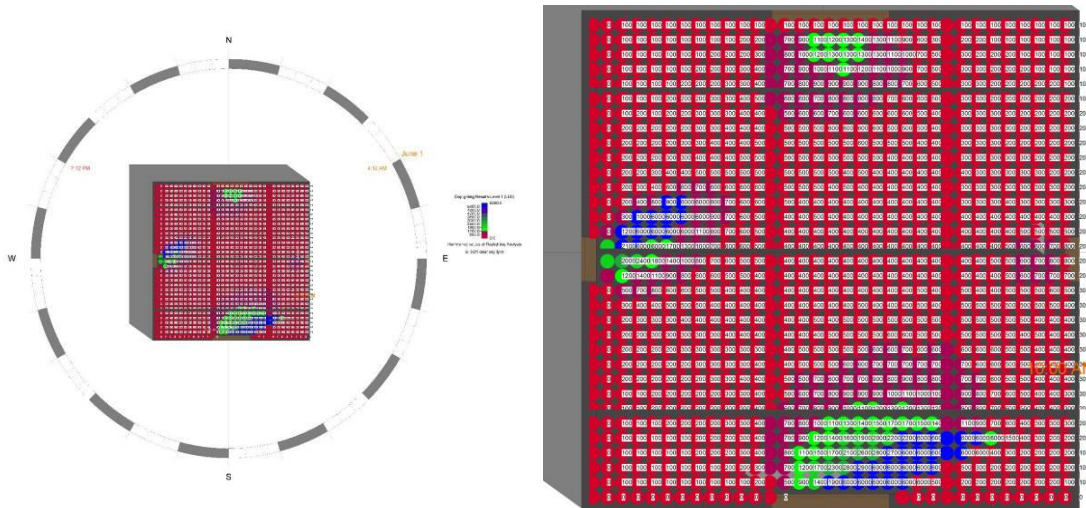
Lighting test as true false operation

The above images are produced via lighting extension for Revit. Immediately it is obvious that square does not perform better than the south-facing rectangle. So here the test can be used as a true false operation to validate our iterations. Looking back at our WWR=0 example (first one with which we started) we can omit it using this test. With a few modifications to the view we can even see the illuminance values:



Changing the style to see the values

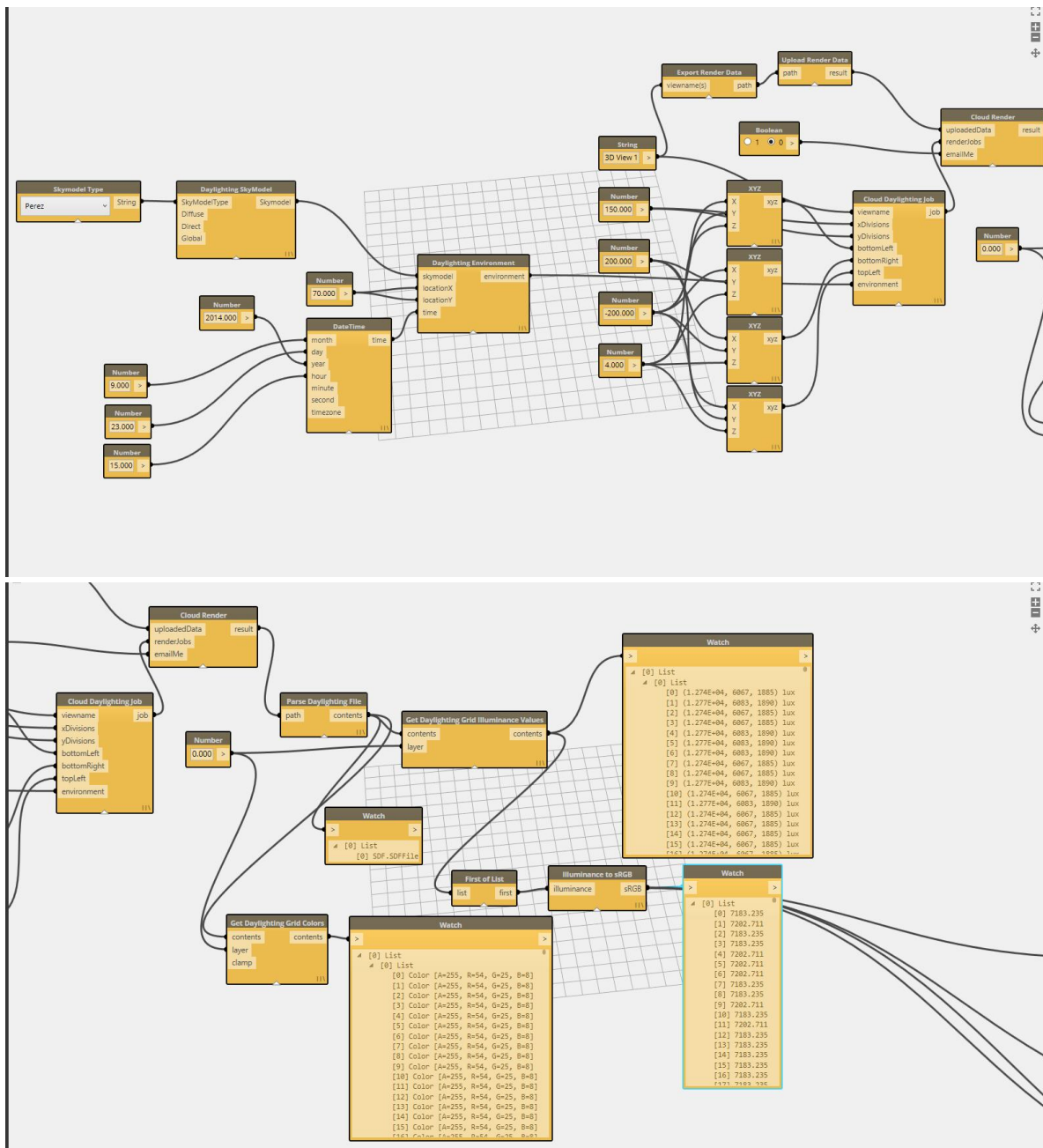
Synchronizing Thermal and Lighting Analysis in Performance-Based Design



Progressive zoom on embedded values

Synchronizing Thermal and Lighting Analysis in Performance-Based Design

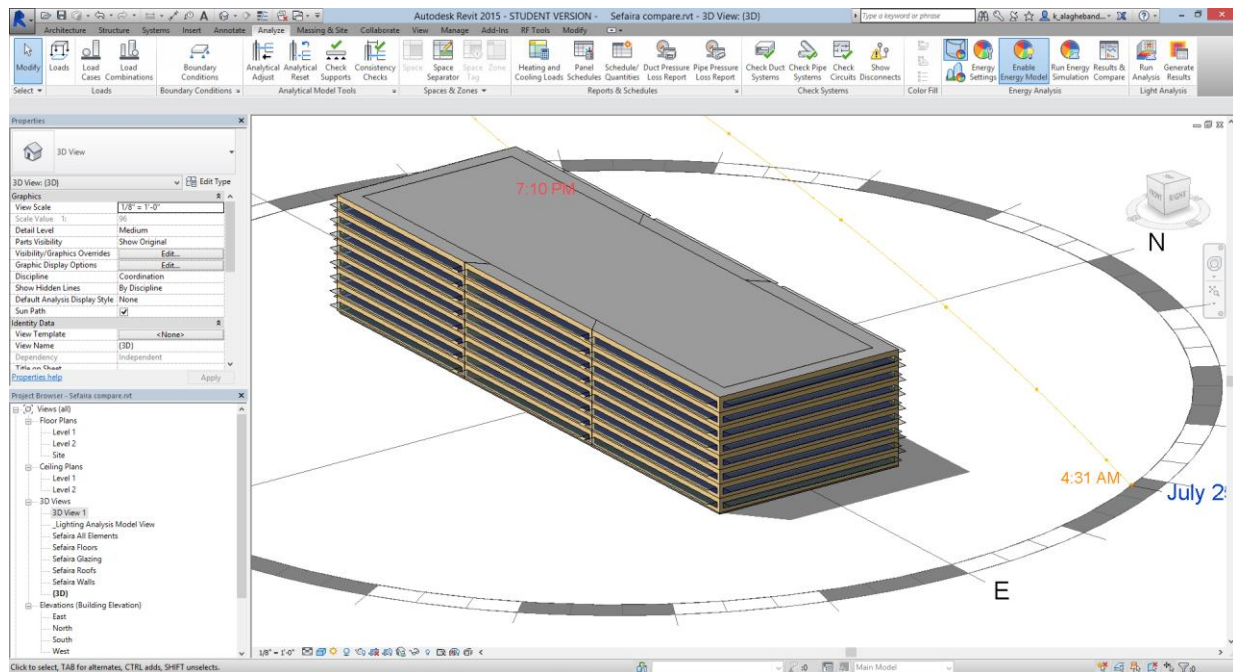
Using Dynamo we can access the data and we can validate our options in the same way. Once we have the data we should cut the results base on the upper and lower thresholds and we should calculate the percentages. Note that we should use the exact same dates. Because lighting test are more time consuming it is possible to apply this only to the x (subjective number) best cases of the thermal analysis.



Simplest setup to get the Values as numbers *(for now it does not work in Dynamo)*

Plugin: Sefaira

While setting up this process has its values (you do it once and you can use most of its parts for later projects as well), using a plugin like Sefaira can be also a fast solution. At the same time employing the same method for dynamic analyses metrics (such as sDA and ASE) can be more time consuming. In you are working in the building element mode, thus, Sefaira can produce fast results:



As you may see in the above picture, once you install the plugin, Sefaira will automatically generate additional views. As you work you iterations, you may gather the results from the cloud server associated with the plugin.

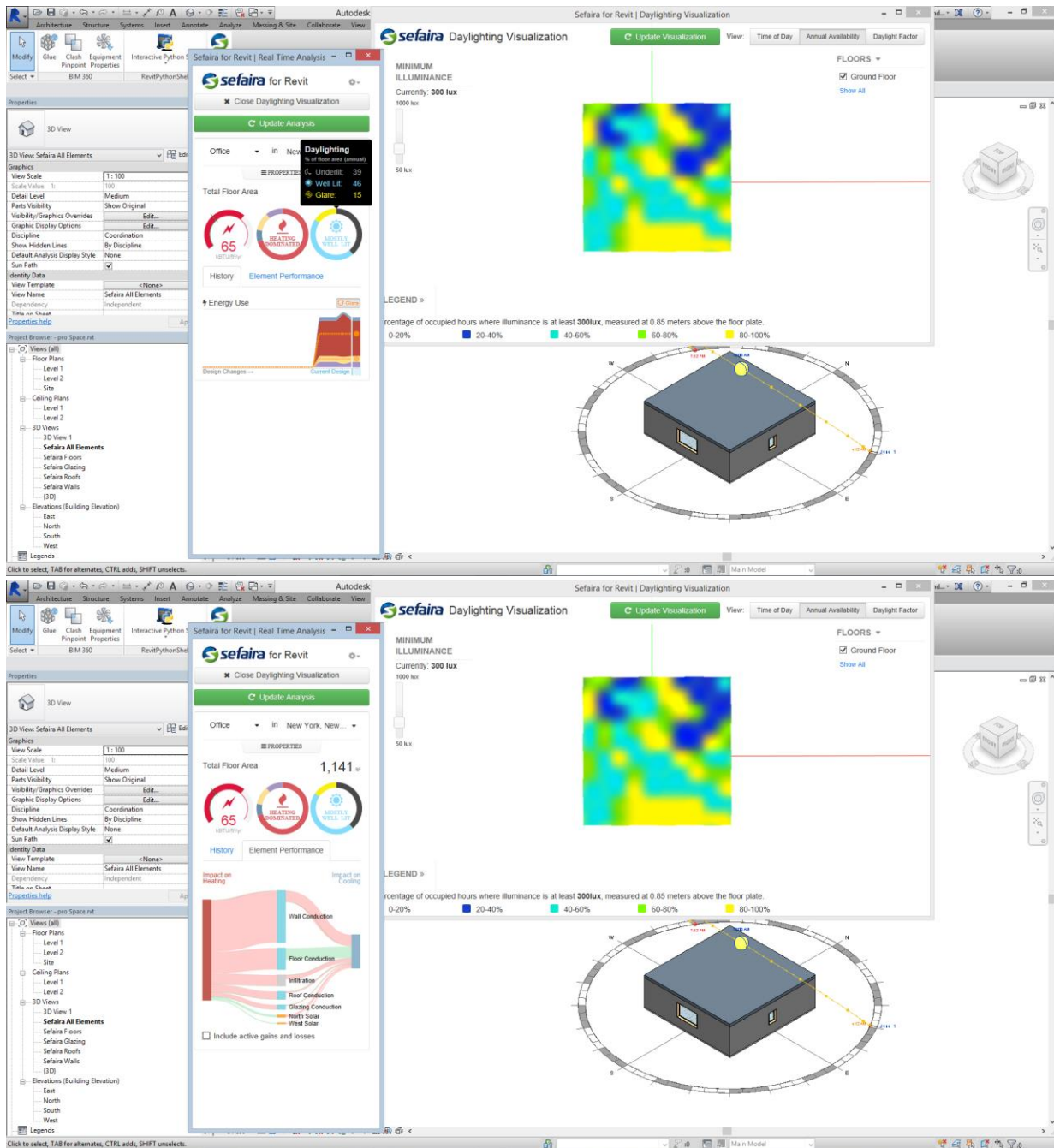
Following figures will indicate that besides the false color images of the sDA data mapped on the surface and key thermal metrics such as EUI, Sefaira also provides suggestions for improving the design (the element performance tab).

Because at this moment lighting data and Thermal data are already present, a very practical, if not accurate, synch is also suggested by Sefaira. Using the following formula can be really helpful especially in the early stages of the design.

$$\text{Reduced LPD} = [(1 - \text{sDA}/100) * 0.8 + 0.2] * \text{Base-line LPD}.$$

Note that in Sefaira the LPD value is accessible and adjustable. For more info on Sefaira see: <http://sefaira.com/tag/sefaira-for-revit/>

Synchronizing Thermal and Lighting Analysis in Performance-Based Design



Sefaira's way of synching