The Parametric Façade

Optimization in Architecture through a Synthesis of Design, Analysis and Fabrication

by

Peter C. Graham

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Architecture

Waterloo, Ontario, Canada, 2012
© Peter C. Graham 2012

Author’s Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Modular building systems that use only prefabricated parts, sometimes known as building "kits", first emerged in the 1830s and 1840s in the form of glass and iron roof systems for urban transportation and distribution centers and multi-storey façade systems. Kit systems are still used widely today in the form of curtain wall assemblies for office and condominium towers, yet in all this time the formal flexibility of these systems (their ability to form complex shapes) has not increased greatly. This is in large part due to the fact that the systems still rely on mass-produced components. This lack of flexibility limits the degree to which these systems can be customized for particular contexts and optimized for such things as daylighting or energy efficiency.

Digital design and fabrication tools now allow us to create highly flexible building façade systems that can be customized for different contexts as well as optimized for particular performance objectives. This thesis develops a prototype for a flexible façade system using parametric modeling tools.

The first part of the thesis looks at how parametric modeling can be used to facilitate building customization and optimization by integrating the acts of design, analysis, fabrication and construction. The second part of the thesis presents the façade system prototype and documents key aspects of its development. The façade system is modeled in Grasshopper 3D, a parametric modeling plug-in for Rhinoceros 3D. The model has built-in analysis tools to help the user optimize the façade for daylighting, energy efficiency, or views within any given context, as well as tools that alert the designer when fabrication or construction constraints are being violated.

Acknowledgements

I'd like to thank my thesis supervisor Marie-Paule MacDonald for her guidance and patience, and for helping me focus my efforts. I'd also like to thank my committee members Terri Meyer Boake and Mark Cichy for their helpful feedback and advice. Additional thanks go to Tim Verhey and the people at Walters Inc. for giving me a very interesting and valuable tour of their fabrication facilities. Finally, I'd like to thank my father and my brother for their unwavering support over the years.

Table of Contents

List of Illustrations

1. Introduction

Most Computer Aided Design (CAD) software used by architects today is still representational, meaning that it is essentially a form of visualization. But a new generation of parametric modeling software is now transforming CAD from a visualization tool into a flexible and powerful simulation tool.

Parametric modeling offers architects the ability to rapidly explore different design options, and when it is combined with flexible Computer Numerical Control (CNC) fabrication techniques, as well digital analysis techniques such as Computer Aided Engineering (CAE) tools, it opens up a great degree of formal possibilities for architects by reducing the time and costs involved in designing and creating custom building components and assemblies. But beyond just providing the ability to create novel building forms and material effects, parametric modeling provides designers with an improved ability to optimize buildings with respect to various performance goals, such as daylighting, structural efficiency, energy efficiency, ventilation, acoustics, and so forth.

This thesis explores the potential use of parametric design as an optimization tool. The thesis is divided into two parts: the first part of the thesis looks at the implications of mass-production on building customization and optimization and the role that parametric design plays in facilitating optimization; the second part of the thesis documents the development of a prototype for a flexible building façade system that can be optimized for daylighting, energy performance, or view availability within any context using built-in analysis tools. The façade system is modeled in Grasshopper 3D, a parametric modeling plug-in for Rhino 3D. There were four main objectives when creating the system:

1) To make the system highly flexible and scalable.
2) To integrate a range of analysis tools within the model.
3) To use fabrication and construction processes that are simple and consistent regardless of the façade configuration.
4) To keep the modeling workflow as fast and fluid as possible

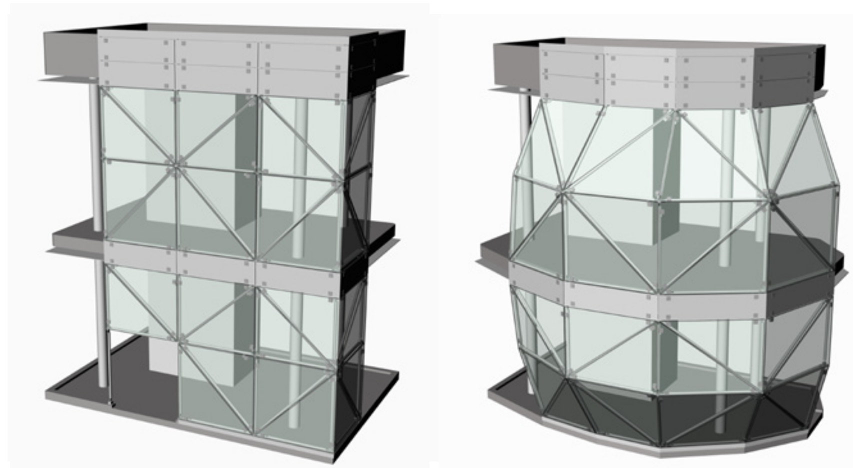


Figure 1-1 *The prototype façade system is designed to be highly malleable and to be able to form many different shapes.*

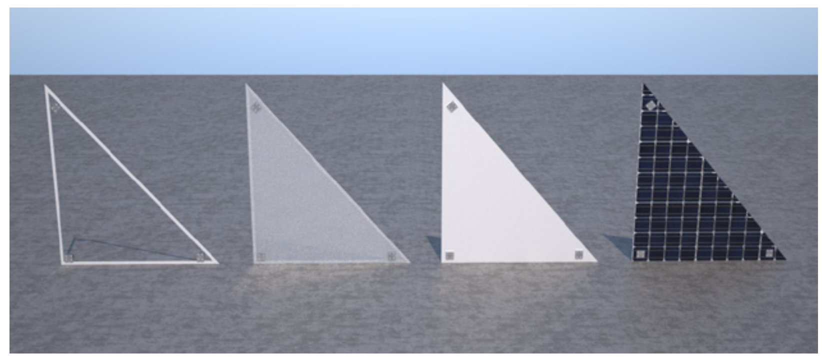


Figure 1-2 *The façade system has four different but interchangeable panel typess: glazed, opaque, translucent, and photovoltaic. Panel types can be changed any time using a spreadsheet.*

<u>Flexibility and Scalability</u>

The façade system is designed to be as flexible as possible so that it can be used in a wide range of design contexts as well as fine-tuned in response to analysis results. A triangulated panelling system allows the façade to be shaped into a wide variety of forms while at the same time ensuring that individual panels always remain planar and structurally stable, regardless of the overall configuration of the façade.

The system can be scaled according to the needs of any project. The number panels, the sizes of panels, and the number of storeys are all variable. Each panel can be any one of four distinct panel types: glazed, opaque, translucent, or photovoltaic. The panel types are controlled using an OpenOffice spreadsheet linked to the Grasshopper model.

Analysis Tools

A number of proprietary analysis tools integrated within the parametric model help the user optimize the façade for views, daylighting, and energy efficiency within any given context, as well as help identify potential problem areas during construction. The analysis tools range from rough, rule-of thumb tools for schematic design, such as vector-based view analysis tools, to more precise tools for design development phases, such as daylighting analysis tools. A number of key statistics about the model are also available to the user at any time, including the window-to-wall ratio, surface-to-volume ratio, average R-value, floor heights or panel heights, panel costs, and so forth. A utility in the model allows these statistics to be exported to a spreadsheet at any time for archival purposes.

Fabrication and Construction

To ensure that costs remain predictable and consistent regardless of the façade's configuration, all panels use the same straight-forward construction system. Additionally, most elements of the façade system can be made from planar materials or extruded profiles cut to size using CNC fabrication techniques such as laser cutting or plasma cutting. Currently, these CNC techniques represent the most accessible and economical forms of custom fabrication, as they use relatively common machines and require fairly minimal setup compared to other processes such as moulding or extruding.[1]

Workflow and Feedback Loops

Smooth workflow is important for any design process and critical when attempting to create feedback loops between design and analysis. The para-

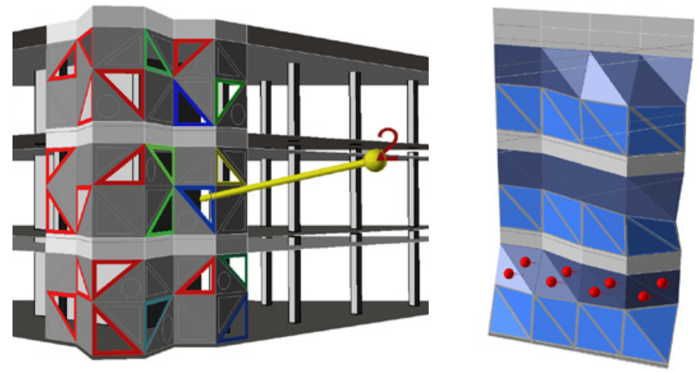


Figure 1-3 *Analysis tools in the model include vector-based view evaluation (left), and radiation mapping for solar panels using Radiance (right).*

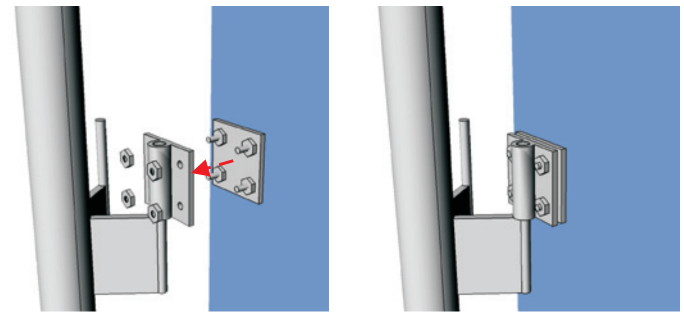


Figure 1-4 *Panel installation is a consistent and straightforward process.*

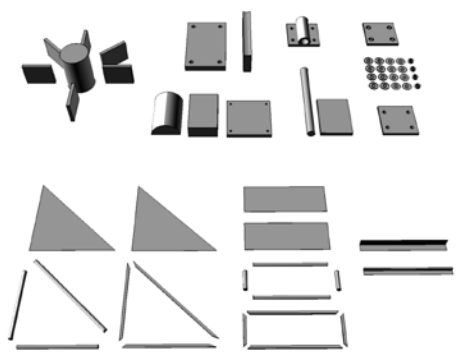


Figure 1-5 *The façade is composed mostly of simple extrusions and planar materials.*

metric model is designed to be simple and intuitive to use, and engineered so that the workflow remains as straightforward and as fluid as possible for the user at all times. More specifically:

- the model can be manipulated very easily by moving vertex points
- panel types and other features can be changed quickly using a spreadsheet
- most sub-assemblies in the model can be modified or replaced without greatly disturbing other systems (i.e. without propagation errors)
- all parameters and statistics are clearly organized and easy to find
- all objects in the model can be selectively turned on or off, or viewed in a simplified form in order to reduce computer processing loads and speed up modeling and analysis procedures.

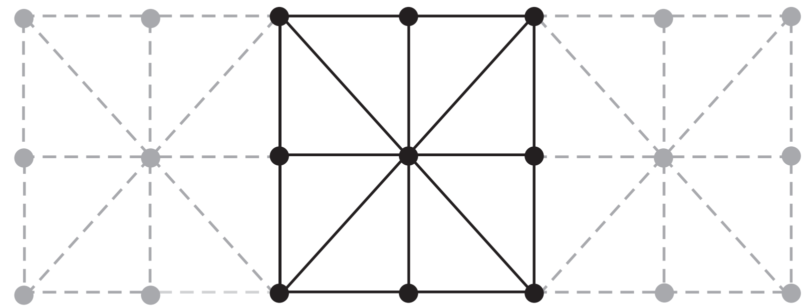


Figure 1-6 *A number of tools are implemented in the model to speed up workflow and feedback loops. For example, the user can choose to generate and view only part of the façade.*

"The sparse geometries of the twentieth century Modernism were, in large part, driven by Fordian paradigms of industrial manufacturing, imbuing the building production with the logics of standardization, prefabrication and on-site installation. The rationalities of manufacturing dictated geometric simplicity over complexity and the repetitive use of low-cost, mass-produced components"[1]

– Ruben Suare

"Hundreds of years ago, all of architecture could be held in the intelligence of a single maker, the master builder. Part architect, part builder, part product and building engineer, and part materials scientist, the master builder integrated all the elements of architecture in a single mind, heart, and hand. The most significant, yet troubling, legacy of modernism has been the specialization of the various elements of building once directed and harmonized by the master builder" [2]

– Stephen Kieran and James Timberlake

2. Towards Optimization in Architecture

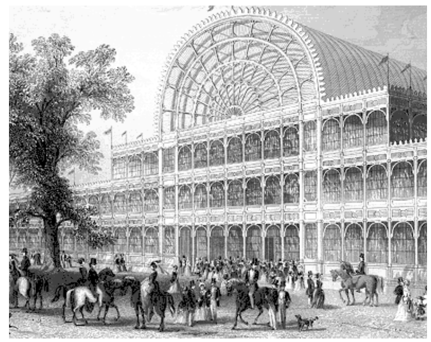
2.1 From Mass-Production to Digital Design and Fabrication

Early Beginnings of Mass Production

The creation of Joseph Paxton's Crystal Palace, a temporary structure designed for the Great Exhibition of 1851, marked a profound turning point in architecture. The entire building was a modular "kit" structure composed of industrially produced iron and glass components. While kit building assemblies had already existed for decades in the form of iron and glass roofs for railway stations, market exchanges and the like, these structures typically formed only parts of buildings, not entire buildings. With the Crystal Palace, modular, industrially produced building systems were no longer just a necessary means of augmenting older building types to accommodate the new spatial and structural demands of the modern economy, but rather, these systems were now being presented as the basis for entire buildings and an entirely new type of architecture.

Assembled in around four months and comprising some 93,000 square meters of glass, the Crystal Palace was a spectacular feat of engineering for the time. But a lesser-known fact is that the completed structure acted like a giant greenhouse with serious glare and solar heat gain problems that even an ad-hoc system of canvas awnings could not satisfactorily fix.[3] The Crystal Palace was thus emblematic of both the usefulness and the limitations of a systematized approach to building using mass-produced components.

Remarkably, today, over a hundred and fifty years later, we still design buildings in a very similar way to the Crystal Palace, only we use climate-control to offset solar heat gains and internal shading devices to attenuate excessive glare or hot spots. But with changing attitudes today towards energy conservation, environmental emissions, the sustainability of material resources, as well as occupant comfort, the idea of using climate control as a means of compensating for inadequate and contextually unresponsive designs is no longer an attractive option for designers. In a previous age where there was a perceived abundance of energy and less awareness of the environmental impacts of buildings, it made sense to view "efficiency" mostly in terms of labor productivity. Now however, our defi-



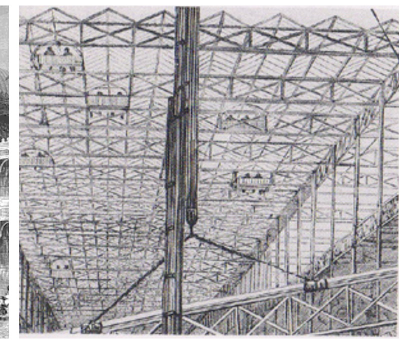


Figure 2.1-1 *Joseph Paxton's Crystal Palace (1851) was emblematic of both the potential and the challenges of using mass-produced building components.*

Figure 2.1-2 *P. Fontaine's Galerie d'Orleans, Paris (1829) was an early example of the use of modular, mass-produced building components to augment more traditional architectural forms.*

nition of efficiency is broadening to account for such things as energy usage, material usage, and even occupant comfort, which we now understand can translate back into increased labor productivity.[4]

Modularity

When modular building kits were first introduced, they offered a host of advantages over traditional methods of construction, including efficiencies in component production, predictable construction processes, compact and efficient transportation, and rapid assembly using general labor. These features allowed the building kits to be transported and built all over the world at low costs, even in non-industrialized nations.[5] The potential advantages of using mass-produced components in architecture were clear from very early on, and they persist to this day.

While few buildings today are "kits" in the strictest sense, many buildings use mass-produced, modular façade systems, and the majority of buildings in the developed world, regardless of size or purpose, are constructed using mostly modular, off-the-shelf, mass-produced components of some sort. In the design of a typical building today, deviation from the use of mass-produced components and their inherent economies of scale usually implies a significant price premium as well as introduces a level of uncertainty into the project with respect to such things as completion schedules, constructability, and envelope performance or durability. As such, highly customized building components tend to be found mostly in expensive projects or in high-profile spaces within buildings, such as lobbies. Henry Ford once famously commented, in reference to his company's Model T car, that "any customer can have a car painted any color that he wants so long as it is black".[6] A corollary for architecture might be this: any customer can have a window, door, tile, panel, or brick of any shape that they want, so long as it is rectangular or from a single family of parts.

Production Efficiencies vs. Design Requirements

Architectural design is the only major field of design that uses mostly standardized, off-the-shelf components both on the inside and outside of designs. While other areas of design, such as industrial design, deal with objects that are produced en-masse, today's buildings are typically one-off projects with limited economies of scale.[7] While there can be a certain art to designing with standardized parts, they tend to limit aesthetic and spatial options, as well as

Figure 2.1-3 *Mies van der Rohe's Seagram's Tower, New York (1958), was an iconic building that influenced the design of office towers for decades. Highly glazed uniform designs, while often cheap to produce, can present many challenges for energy efficiency and occupant comfort.*

the degree to which building shapes and materials can be optimized to meet client needs within specific contexts. This is especially the case with larger buildings where there are great economic incentives from both a design and construction perspective to use repetitive elements.

Architects have struggled with the relationship between design and mass-produced building components from their very introduction. In fact, a great deal of the ideology of Modernism explicitly revolved around architecture's relationship to its means of production. Examples of this range from the Art Nouveau movement's attempts to synthesize machine production with artistic expression and their desire to create artistic totalities in a context of commodified building components, to the Bauhaus school's initiative under Walter Gropius to make designers the master craftsmen of machine production, to Le Corbusier's visions of houses as mass-produced objects created by unskilled laborers in a similar manner as cars or appliances. It is nearly impossible, in fact, to separate the aesthetic sensibilities and social agendas of Modernist architects from the radical changes that were occurring in the way building components were produced, and it is no mystery that dominant aesthetic themes of the modern period -- abstraction, repetition and economy of form -- are inextricably related to mass production.[8]

In manufacturing, process specialization and part interchangeability are key to achieving economies of scale. By specializing in making a standardized part that can be used in a wide variety of end-products, such as a bolt, a manufacturer can gain the economies of scale that come with highly repetitive production processes.[9] By definition this system separates the producer of the part from the end-product, and the designer or creator of the end-product tacitly accepts a somewhat limited range of formal or functional possibilities as an acceptable trade off for keeping design and production costs predictable and low. Likewise, in the building industry, a manufacturer of building components has no real connection to the design of the building itself; their focus is on creating a product that will be applicable in the widest possible variety of situations, and this generally means creating a product that is simple and repetitive in form, self-supporting, and modular in design. As in manufacturing, such products afford the end-designer a somewhat limited range of formal and functional possibilities in order that the product can remain cheap, consistent, straightforward to install, as well as relatively interchangeable with other, similar products. A perfect example of this is the common brick, which is probably the most standardized component in the building industry. It is predictable, interchangeable, and cheap, but its use is generally limited to straight or gently-curved walls with vertical

Figure 2.1-4 *Victor Horta's Hotel van Eetvelde (1895) exemplifies the Art Nouveau movement's quest to infuse artistry into modern building materials such as iron, and to achieve artistic totalities at a time when building components were becoming increasingly commodified.*

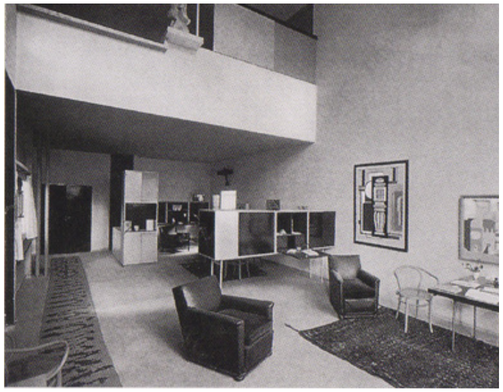


Figure 2.1-5 *Le Corbusier and Pierre Jeanneret's L'Esprit Nouveau Pavilion (1925), an adaptation of Le Corbusier's Maison Citrohan design, was conceived as a ready-made, standardized, and mass-produced object.*

joints and right-angled corners. Such a cladding material does not provide optimal solutions for all design problems, however we accept it because it offers adequate solutions to many problems, and does so very predictably and cheaply.

Unfortunately, most common structural and cladding systems have limited flexibility, and this can be a barrier to both customization and optimization. When construction systems are modular and standardized, the forms of buildings themselves naturally start to take on characteristics of standardized modules.

<u>Lack of Functional Integration</u>

While specialization and the use of mass-produced building components have brought undeniable benefits to the building industry in terms of lowered costs, higher predictability, and greater precision, one downside is that architects have become more and more dissociated from many key aspects of a building's production, most notably in the area of component or assembly design and fabrication. The structure of the building industry has itself come to mirror the way buildings are physically made, that is, it has come to resemble a system of generic and interchangeable services that reflect the generic and interchangeable building components used in buildings. Relationships between professionals and specialists within the industry tend to be hierarchical instead of reciprocal; instead of combining their expertise and knowledge to optimize solutions for specific problems, specialists tend to work relatively independently from one another in circumscribed roles.[10]

It has now long been the case in a standard open-bid construction contract that the "means and methods" of construction are the general contractor's responsibility, and neither the architect nor the engineer legally has any control over the construction process. Because of this arm's length relationship between architects and building construction, contractors are generally hired based on how cheaply and efficiently they can construct a building, not on the basis of whether they can bring special insights or knowledge to the design process.[11] The system tends to promote new combinations of architects, contractors and fabricators for each project and discourages long-term collaborative relationships. It also encourages the use of lowest-common-denominator approaches to construction and discourages experimental or progressive designs.

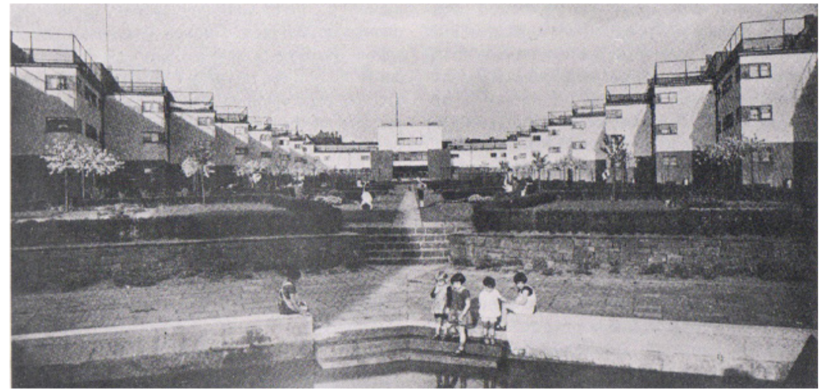


Figure 2.1-6 *Ernst May and C.H Rudloff's Bruchfeldstrasse Estate (1925): Modularity and repetition have economies of scale on every level of the design and construction continuum, but there are a finite number of arrays for standard modules,*

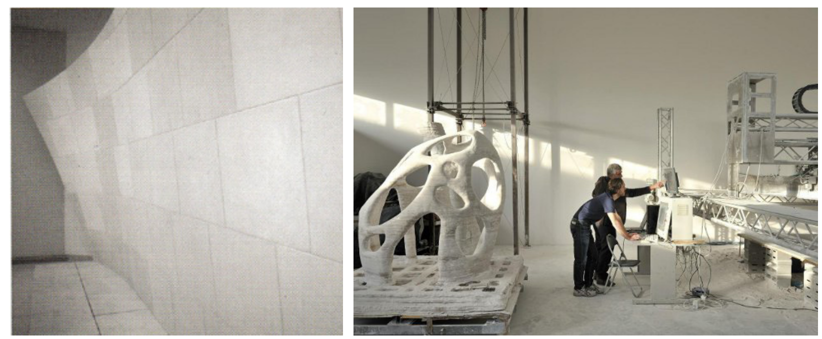


Figure 2.1-7 *Gehry Partners experimented with custom-milled masonry for the Walt Disney Concert Hall (left). In the future, additive fabrication processes, such as Enrico Dini's masonry "printing" machine (right) may allow designers to create large and complexly shaped masonry units without the need for milling or molding.*

The Emerging Role of the Computer

Optimization of buildings is essentially a form of customization, and customization today mostly exists within the bounds of whatever flexibility generic building components and systems offer us. Customization at the level of the building components themselves requires integrated knowledge and integrated processes within the industry. For example, without substantial communication with fabricators, it is difficult for architects to design a custom wall assembly since they do not know the constraints or limits of the fabrication processes involved. Attempts at integration in the design process fight against both the economies of scale of mass production and the entrenched industry divisions that have come to mirror the means of production. Fortunately, however, computers are beginning to substantially lower the barriers to customization and optimization as well as challenge the status quo of industry segregation. Architects now have at their disposal a host of powerful tools that can facilitate the re-integration of design, analysis and fabrication processes and work synergistically to help produce novel, non-repetitive building components or assemblies at reasonable costs. The revelation that efficiency in architectural design is no longer inextricably linked with sameness, and that building forms and materials can be made continuously variable instead of repetitive, is fundamentally altering how architects approach building design as well as building optimization.[12]

At the heart of the new digital toolset lies advanced 3D CAD utilizing solid or Boundary Representation (BREP) objects. Solid and BREP modeling allows designers to create complexly shaped volumetric objects and assemblies that are numerically quantifiable and can therefore be accurately analyzed using Computer Aided Engineering (CAE) analyses, or fabricated using using Computer Aided Manufacturing (CAM) processes. Many different types of analysis can be performed on either the object, assembly or building level, including Finite Element Analysis (FEA) for structural elements, thermal and energy efficiency analyses for building assemblies, daylighting simulations for building volumes, fluid dynamic analyses to optimize ventilation, and so forth. After geometries have been analyzed and optimized, information from the CAD model can be used to drive Computer Numerical Control (CNC) fabrication machines, such as plasma cutters or 6-axis milling machines. These machines can often be used to efficiently create non-repetitive, non-standard building forms and components.

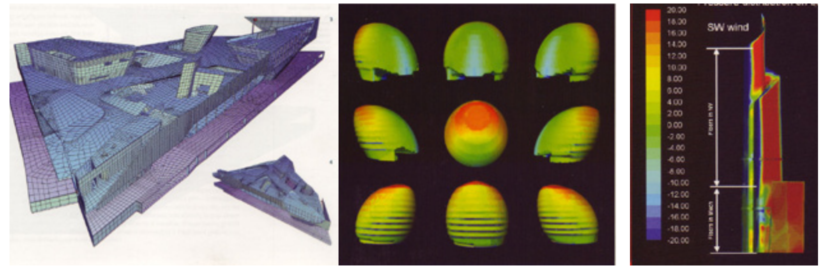


Figure 2.1-8 *Computer analysis techniques are now being integrated into architectural design. Left: Finite Element Analysis is used to evaluate the structure for Zaha Hadid's Phaeno Science Center; Center: solar radiation analysis informed the shape of Foster and Partners' GLA City Hall; Right: a wind pressure analysis of The Pinnacle, an office tower design by KPF.*

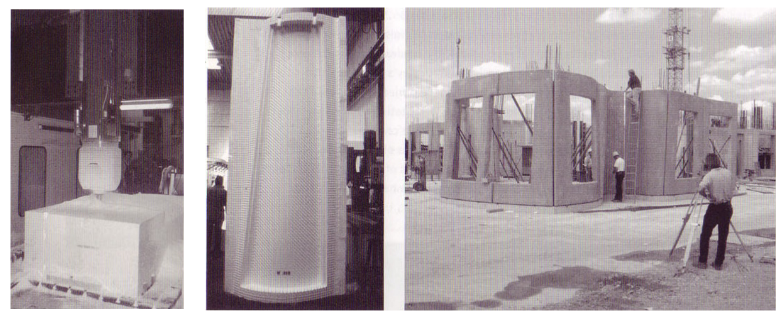


Figure 2.1-9 *For Gehry Partners' Zollhof Towers, the complexly-shaped formwork needed for the prefabricated concrete panels was created using a CNC milling machine.*

Figure 2.1-10 *The complex curvilinear metal panels that clad Gehry Partners' Experience Music Project were installed with the assistance of GPS satellite guidance.*

Parametric Modeling

The ability to model, analyze and fabricate complexly shaped objects and assemblies is important for building optimization, but these processes require integration, which is where parametric modeling comes in. With the use of parametric CAD, it is now possible to create 3D computer models of buildings or building assembles that behave like flexible virtual mock-ups. Parametric models are not simply representational - they are rule-based, relational constructs that allow for iterative testing of design options while preserving underlying component and assembly topologies. By combining parametric modeling with analysis tools and fabrication information, architects are able to rapidly test design permutations and understand how the optimization of one aspect of the design, such as daylighting, affects other aspects, such as structural efficiency or fabrication. The feedback and flexibility of parametric design allows designers to fine-tune specific performance parameters of buildings as well as find “best fit” solutions that satisfy many different design criteria simultaneously. In some cases a standard design process can be augmented by generative or relational algorithms that automatically optimize certain aspects of the design according to pre-established rules.

Building Information Modeling (BIM), a form of parametric object meta-data management, is also a very important tool in the optimization process, especially when used in conjunction with robust parametric modeling and analysis tools. BIM helps integrate and streamline the design, fabrication and construction process by facilitating such things as part tracking, real-time cost estimating, clash detection, trade coordination and construction sequencing, automatic code compliance checking, as-built feedback loops, and so on. BIM can also be used to facilitate things such as daylight, thermal, or acoustic analyses by embedding material and assembly information within CAD objects.

The use of sophisticated parametric modeling platforms as a means of close-coupling of design, analysis and fabrication processes is already well established in design industries with more vertical integration, such as the aerospace, industrial design, and automotive industries. In architecture, however, progress in this area has been relatively slow, in part because of procedural and legal frameworks that delimit the responsibilities of specialists, but also due to the simple fact that the majority of building projects are unique and tend to involve different combinations of specialists using different software packages. Much communication in the industry is also still handled through the exchange of two-dimensional drawings, which limits

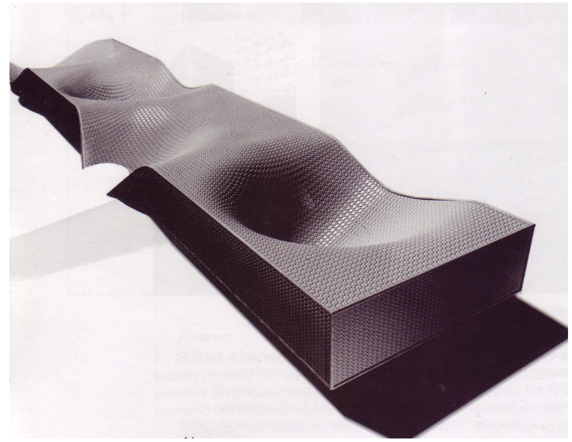


Figure 2.1-11 *For “Ecoscape”, a design by Chandler Ahrens, Eran Neuman, and Aaron Sprecher, the building’s shape was formed for optimal solar cell exposure using parametric and generative design techniques.*

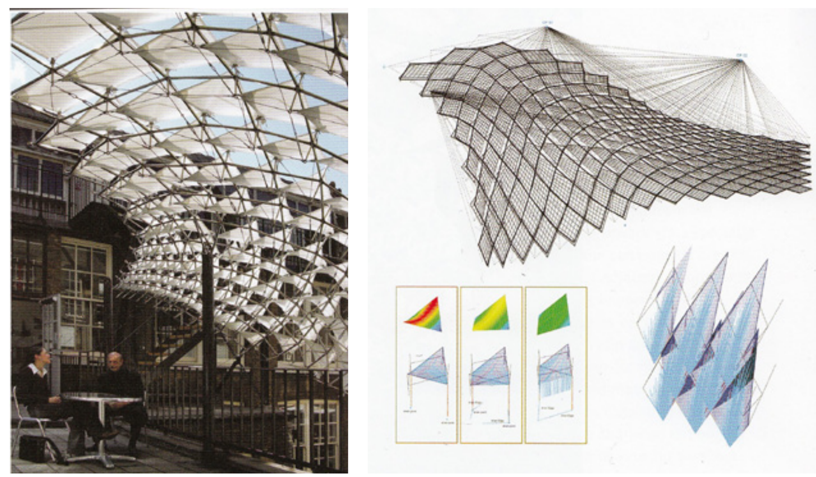


Figure 2.1-12 *For the “AA Component Membrane”, a terrace canopy designed at the Architectural Association, the form was simultaneously optimized for sun, wind, drainage and views using a parametric model combined with various computational analysis techniques including fluid dynamic wind flow analysis, precipitation analysis, stress analysis, and solar analysis.*

collaboration potential. While this is slowly changing, there are still many obstacles when attempting to exchange complex three-dimensional digital data, including the lack of a standard data exchange format.[13]

Toward Greater Integration

It will take time to overcome inertia in the industry and for legal and procedural frameworks to fully adapt to these new techniques, but some important precedents have been set. Design firms are beginning to circumvent the traditional building delivery model through the use of new collaborative models such as design-build models, multi-party joint ventures, and project alliance models.[14] Innovative firms such as Gehry Partners are pushing the boundaries of standard business models in order to find ways of doing business that are much more consistent with the level of technology that is now available. For example, instead of open-bid contracts, Gehry Partners uses forms of negotiated bidding, where contractors and sub-contractors have some involvement in the design process itself and take some responsibility for certain key aspects of the design.[15] While this type of business model runs very contrary to the status quo and requires a reassessment of the risks and responsibilities of industry specialists, it is clear that the technical and financial obstacles to such cooperation are becoming less and less relevant as digital tools and processes become more advanced and integrated.

Customization always entails risk and uncertainty, but the more that specialists are willing to cooperate early on in a project's development, the more the risks can be understood in advance and accounted for, and the more the uncertainty can be mitigated as a result. Ironically, in the end it may be the drive to find more predictability and efficiencies in design and construction process, rather than the desire to create progressive designs, that spurs more collaboration between various stakeholders in the building industry and encourages the adoption of integrative technologies such as BIM and parametric design[16]. BIM has already been found to increase construction margins by improving coordination and decreasing waste, and to significantly reduce requests for information (RFIs).[17]

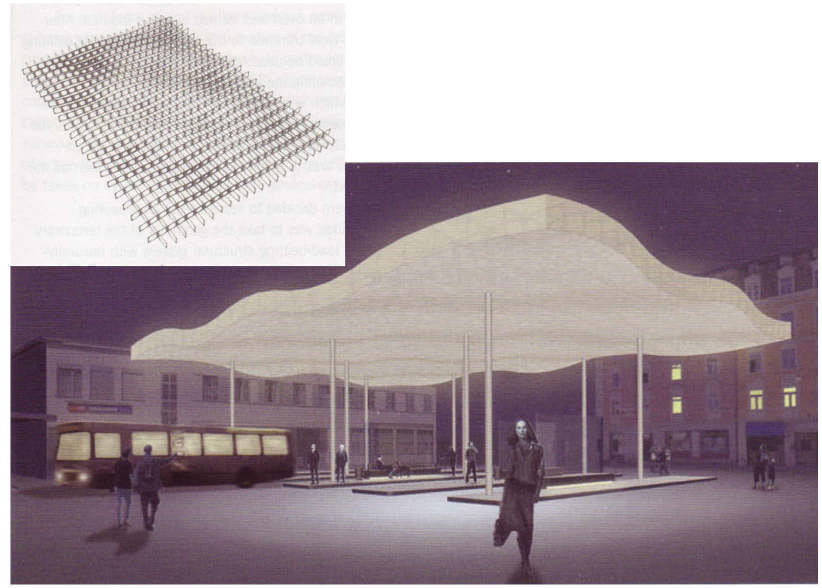


Figure 2.1-13 *OSD Structural Design used generative structural algorithms to optimize the design for a bus station by Hausmarke Architects. In order to achieve maximum stiffness with a minimum amount of materials, the depth of the structural members remains constant while the porosity (density) of the truss members changes according to localized stresses.*

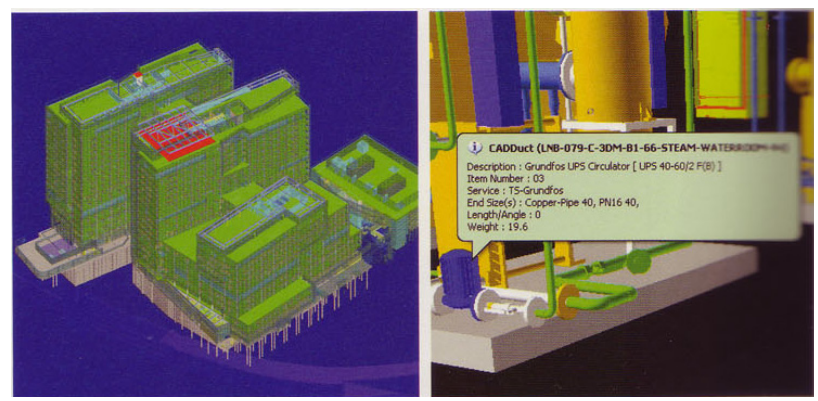


Figure 2.1-14 *For HOK's Royal London Hospital, BIM was used for everything from code compliance checking, to construction scheduling, to clash detection, to facilities management.*

"An architect must be a craftsman. Of course, any tools will do; these days, the tools might include a computer, an experimental model, and mathematics. However, it is still craftsmanship - the work of someone who does not separate the work of the mind from the work of the hand. It involves a circular process that takes you from the idea to a drawing, from a drawing to a construction, and from a construction back to an idea."[18]

– Renzo Piano

"Today, through the agency of information management tools, the architect can once again become the master builder by integrating the skills and intelligences at the core of architecture. This new master builder transforms the singular mind glorified in schools and media into a new genius of collective intelligence"[19]

– Stephen Kieren and James Timberlake

2.2 *Flexibility, Integration, and Feedback Loops*

The separation of design and fabrication in architecture has taught architects to see design as a somewhat abstract process disconnected from material realities. But in order to fully leverage the potential of the new digital tools for the purpose of customization and optimization, architects must cultivate a different sensibility. Architects must learn to see design, analysis and fabrication as interdependent processes rather than separate and abstract processes, and begin to understand the dynamic relationships between material properties, available tools, joinery, assembly processes, functional performance, context, and aesthetic vision. Through a deeper understanding of these complex relationships architects can better optimize buildings for their clients' needs, as well as actualize their own design ideas more elegantly and efficiently. This kind of thinking and approach is very similar to that of the craftsman. For the craftsman, the design process is seen as a circular and iterative one in which feedback loops between a host of diverse design considerations gradually lead the designer towards a satisfactory result. Craft and multi-faceted optimization are essentially the same thing.

Parametric Design

Historically, small-scale operations have been more favorable to craft because a craftsman must work on a scale where they are able to have oversight over an entire process and integrate the acts of design, analysis and fabrication. But today, parametric modeling is broadening the scope of craft by allowing highly specialized disciplines to work together in dynamic, flexible, and iterative design processes on projects ranging in scale from small objects to entire buildings. And while hands-on experience with such things as CNC machines can enrich a project as well as the designer's understanding of fabrication, direct engagement with all aspects of the building process is not a prerequisite for the successful integration of processes. The new craft is as much about encoding, organizing and coordinating information and the activities of specialists as it is about hands-on experience.[20]

Parametric design platforms provide three things that are essential to the development of this new craft and thus optimization: formal flexibility, process integration, and rapid feedback loops.

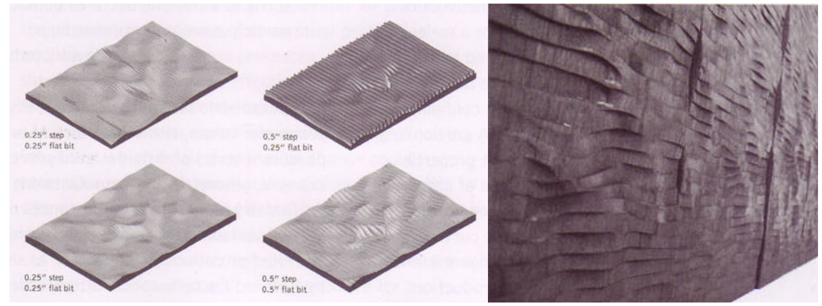


Figure 2.2-1 *In"Striations", designers Carmen McKee and Fuyuan Su experimented with a CNC milling machine to create wood panels. They used different input geometries, different types of wood, different milling machine bits, and different milling feed rates. This kind of direct feedback from machines can now be used as part of the building design process.*

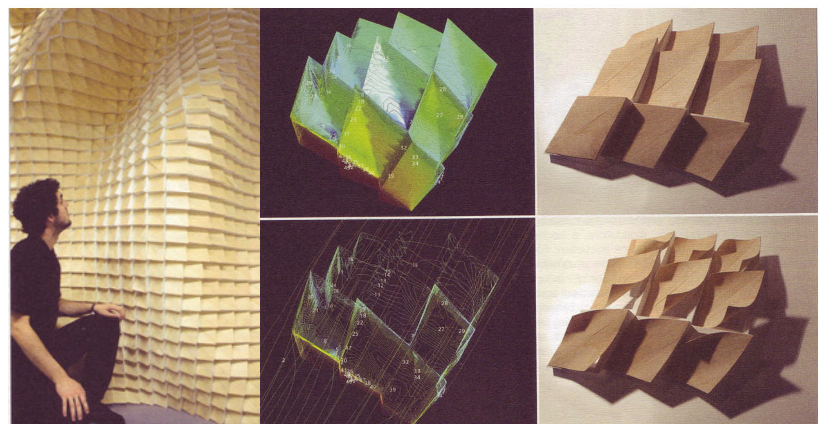


Figure 2.2-2 *In "Responsive Surface Structure", designer Steffen Reichert programmed the behavior of wood in response to moisture into a computer model and used the information to design a surface that responds to local humidity changes. The new craft is as much about encoding and manipulating information as it is about direct engagement with materials or assemblies.*

Flexibility

Formal flexibility is one of the most salient and essential features of parametric design. In a parametric model, building components and assemblies can maintain their geometric or topological integrity as various design configurations are explored. This allows many variations of a design to be rapidly tested once the critical rules and boundaries for the model have been set. Virtually any aspect of an object or assembly can be made variable, including its shape, location, quantity, scale, level of detail, or material composition. Models can be set up so that the characteristics of objects or assemblies are controlled either globally or locally, or both. When combined with flexible forms of CNC fabrication as well as sophisticated analysis tools, this formal flexibility that parametric modeling provides can be used to fine-tune building forms and materials very subtly in response to context, or in response to analysis results.

Figure 2.2-3 *In James Timberlake's SmartWrap Pavilion, a semi-theoretical construct, a layered composite material performs all weathering and service requirements and is varied according to local performance needs. Contextual sensitivity and responsiveness is one of the key ways in which digital tools can help optimize buildings.*

Integration

The integration of design, analysis and fabrication processes is a cornerstone of craft and optimization. Parametric design can facilitate this integration in three principle ways. The first way is by embedding knowledge as rules, relationships or constraints within a model, or within a series of models that reference one another. For instance, by following certain rules, a mechanical duct could change its profile automatically in response to the dimensions of the space around it. The second way parametric modeling can facilitate integration is through warning systems such as dynamic interference checks. In the above case of a mechanical duct for example, an automatic interference check could be used to warn the designer if there was a conflict between the path of the duct and another object such as a structural column. The third way in which parametric modeling can be used to integrate processes is by creating "snapshots" or animations of part of the model's geometry for the purpose of analysis, or for use as reference in the design of a sub-assembly. This is typically done for complex analysis processes or when some aspect of the design, such as the structural system, needs to be resolved independently of the main model.

Figure 2.2-4 *Grimshaw Architect's Fashion and Design Events Building: In order to explore structural options, the designers created simple geometrical elements that could be quickly exported from the Generative Components parametric model to Robot, a sophisticated structural analysis software program.*

Feedback Loops

Feedback loops are the final feature of parametric design critical for customization and optimization. Feedback loops allow designers to ascertain whether a design configuration will be successful or not. The more rapid and informative the feedback loops are, the easier it is to test propositions and understand their implications. In parametric design, feedback loops can occur on two principle levels. The first level involves the evaluation of a design configuration with respect to a specific objective such as structural efficiency. The second level of feedback is more dynamic and non-linear, and has to do with how the optimization of the design with respect to one particular objective affects other aspects of the design. There may be a sympathetic or synergistic relationship, or there may be trade-offs. For example, optimizing a building's shape and materials for thermal efficiency may or may not be conducive to optimal daylighting, depending on the context.

In today's large building projects it is virtually impossible for the architect to have comprehensive knowledge of every aspect of the building's design and construction. The key, therefore, lies in the careful coordination and integration of knowledge, ideas and information from the various disciplines within the entire design and construction continuum.

Modern Master Craftsmen

In *Refabricating Architecture: How Manufacturing Methodologies are Poised to Transform Building Construction*, Stephen Kieren and James Timberlake suggest that while architecture today is too varied and complex for architects to again become the masters of all aspects of building like the fabled "master builders" of yore, architects now have the tools necessary to re-integrate back into the design process the various disciplines that have been spun off from architecture, namely those of material science, process-engineering, and construction. The architect's new role is to integrate and leverage the "collective intelligence" of specialists within the building industry. This re-integration, they argue, will ultimately improve the quality of buildings just as it has improved quality in other design industries such as the aerospace and automotive industries. The ability to accurately simulate complex assemblies using 3D computer models now makes it possible to break a design problem of virtually any size down into manageable parts, or modules, which can be designed by small multidisciplinary teams and then seamlessly integrated back into the greater design, represented by a "master" computer model. The modules can then be fabricated independently, virtually anywhere in the world.[22] Unlike the sterile type of modularity that is often associated with the use of generic, mass-produced components, this

Figure 2.2-5 *Above: A prototype for a 3D spatial gesture design and modeling device by Doo Young Kwon and Markus Gross. Computer interface devices are becoming increasingly sophisticated and someday they will likely offer tactile feedback in addition to gestural input.*

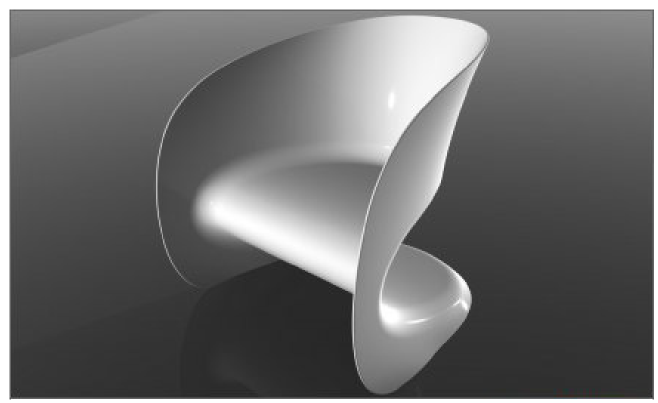


Figure 2.2-6 *FS1 chair by Torsten Plate: Fiber Reinforced Plastics (FRP) represent a good example of the dynamic relationships between material composition, form, assembly, and structure. Much of the rigidity of FRP comes from surface curvature, and therefore the more optimal the curvature is, the more simple and elegant the design can be, and the less need there will be for supporting ribs.*[21]

type of modularity is instead dynamic, integrated, and geared towards creating optimized solutions for specific problems.

On the surface it might seem that such an approach would lead to increased costs for buildings given the fact that most buildings are one-off ventures. But there are a number of factors that can help mitigate the increased design costs. Most significantly, customization can be used as a means of streamlining the fabrication, construction and the servicing of building assemblies. For example, assembly and inspection tasks can be simplified by using low part counts, low part complexity, and by consolidating part functionalities[23]. Building assemblies can now also increasingly be prefabricated off-site, and this can provide a number of advantages including better quality control, less material waste, faster production using shift work in weather-controlled environments, reduced complexity and reduced bottlenecks at the final point of assembly, and the ability to create certain assemblies in parallel that would normally have to be constructed sequentially.[24]

Over the long term, customization can decrease costs and increase profits when it is used as a means of improving the overall quality of a building by making it more energy efficient, more comfortable, more flexible in use, or simply more interesting. And from the designer's personal perspective, proprietary building systems developed for one project can often be used in other projects, and there is always the possibility of creating patentable or marketable building components or systems.[25]

Gehry Partners' MIT Stata Center project provides a glimpse of how parametric design and other digital technologies may in the future be used to coordinate collaborative ventures between architects, contractors, and fabricators in ways similar to what Kieren and Timberlake suggest. For the Stata Center, Gehry Partners engaged the contractor Beacon Skanska and various fabricators through at-risk contracts early on in the project's development and coordinated all design activities using a single 3D parametric master model created in the software program CATIA. While not every sub-contractor was able to use information from the master model in its native CATIA format, the model allowed all parties involved to collaborate and coordinate most of their activities using three-dimensional data instead of two-dimensional drawings. A. Zahner, the sub-contractor in charge of designing and fabricating the façade panels, was able to work in CATIA directly, and consequently they were able to provide Gehry Partners with specific design rules for the panels that could be encoded as constraints in the master model. The contractor, Beacon Skanska, was able to coordinate construction

Figure 2.2-7 *Dividing a design problem into modules reduces the bottleneck at the final point of assembly and puts the responsibility for quality and accuracy in more hands.*

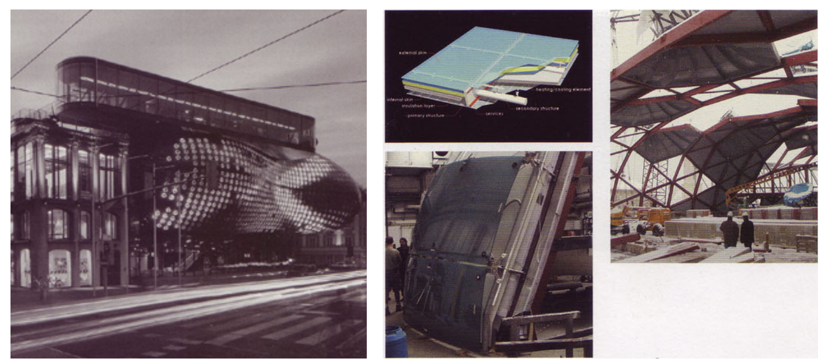


Figure 2.2-8 *The envelope of the Kunsthaus Graz, by Peter Cook and Colin Fournier, is a complex, multi-layered construction incorporating many different elements, including electrical and lighting systems (internal and external). Some elements were prefabricated while others were created or modified in-situ. In the future, completely integrated, prefabricated building assemblies may lead to very short construction times.*

activities for the complexly- shaped building with the master model using sophisticated three-dimensional site-surveying techniques. This allowed , among other things, for information from the construction site to be fed back into the master model as construction progressed so that adjustments to the shapes of prefabricated elements could be made on the fly based on discrepancies between as-built construction and the original computer model.[26] The latter is an excellent example of how integration can actually reduce uncertainty and risk by introducing flexibility and feedback loops into the entire design and construction chain.

The Stata Center project was not without problems however. Not long after the building's completion in 2004 a number of problems began to occur including leaks, mold, cracking walls, excessive snow accumulation, and so forth. As a result, MIT sued Gehry Partners in 2007, claiming that the firm had "breached its duties by providing deficient design services and drawings".[27] These types of problems are a reminder that while digital tools can facilitate the creation of novel and complex building forms, such forms still require careful analysis and attention to detail.

Projects like the MIT Stata Center represent important steps towards creating the coordination and feedback loops necessary for a synthesis between design, analysis, fabrication and construction, as well as creating the collaborative relationships between specialists and stakeholders that are necessary for pushing the boundaries of form and optimizing buildings. When industry specialists work together in such a manner, not only do more sophisticated designs become possible, but the insights and knowledge shared in the collaborative process helps inform future ideas and decisions for all parties involved. While industry integration presents risks and uncertainty in the short term, there is reason to believe that in the longer term the benefits will greatly outweigh these risks. As information exchange between specialists becomes more direct and sophisticated and the entire building process, from design to construction, becomes more fluid and non-linear, the need to rely on standardized parts and processes in order to achieve predictable and economical results will be diminished. Integration will lead to opportunities for optimization on many levels, from the performance of building components or assemblies to the very processes of design and construction themselves. Ultimately, the design and construction processes will become more flexible and adaptive, able to embrace the nuances of context and respond to contingencies, instead of avoiding them.

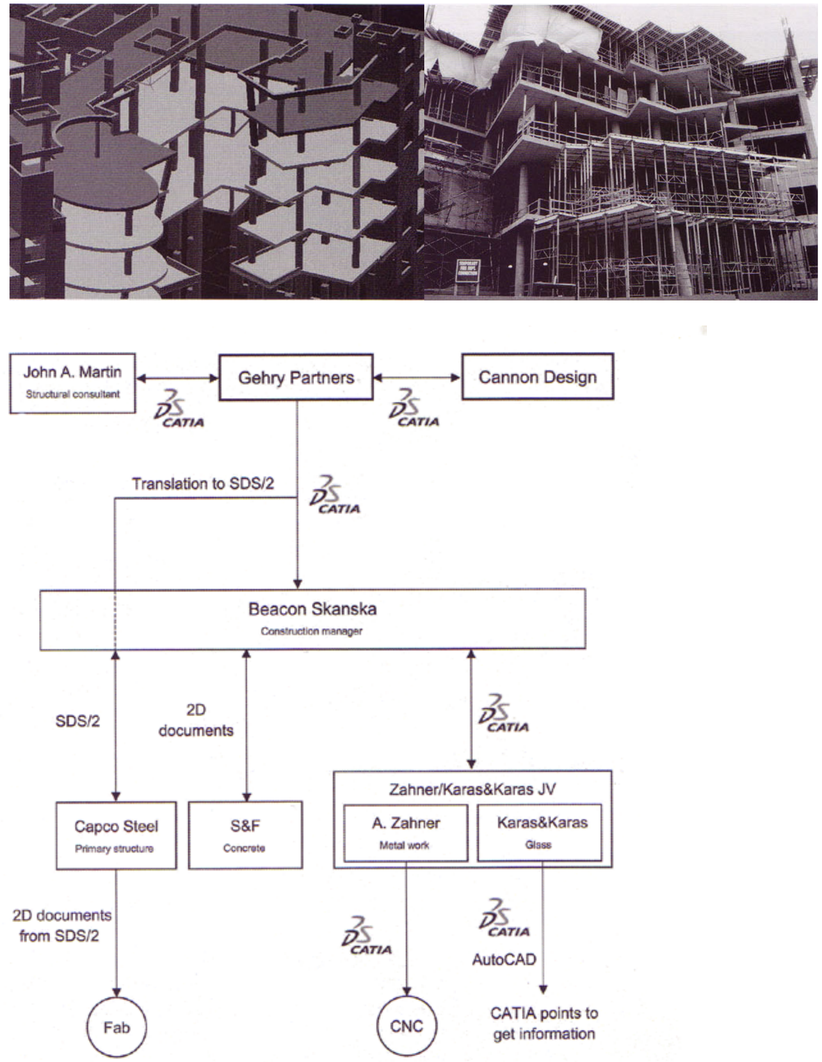


Figure 2.2-9 *Gehry Partners' MIT Strata Center was innovative in its use of negotiated bid contracts and a digital master model to coordinate all design, construction and fabrication processes.*

"Deep insight is not to be found in the lonely garrets but in the fabric of the world. It results from a way of thinking that is not confined to separated, divided truths but which seeks its way through significant connections, one whose ordering power is founded on its combinatorial capacity."[28]

– Earnst Jünger

"Just like the designers of the past, the craftsman of the digital age - the designer working with virtual representations of the material artifacts - seeks out unpredictable outcomes by experimenting with what the medium and the tools have to offer"[29]

– Branko Kolarevic

2.3 *Optimization with Parametric Design*

What distinguishes parametric modeling from standard CAD is the ability to create relationships between elements within the model, as well as rules that govern these relationships. With parametric CAD, the shape of a design is not necessarily defined initially, only it's parameters.[30] This has two important implications: firstly, it allows models to be set up in such a way that design iterations can be explored without the need to erase objects, and secondly, it allows various types of knowledge, such as fabrication constraints, to be embedded in the model.

In parametric CAD, an object's form and behavior are defined and constrained by a combination of its internal logic and its input variables. An object's input variables can be determined by a number of different means: by direct user input or an input sensor, by using a mathematical expression, by using an attribute of another object in the model, by referencing a list of data, or by some combination of these things. Parametric CAD systems are usually propagation-based systems that calculate a new "solution" whenever a design change is made by starting with known quantities and proceeding towards unknown quantities, using topological ordering algorithms to do so.[31] Virtually any aspect of a design can be simulated in a parametric model insofar as its properties and behaviors can be expressed as a combination of input variables, constants, formulas, or relationships between objects or data sets.

While parametric modeling can be used to refine or optimize a design that is already relatively fixed in terms of its form or its fabrication, it is increasingly being used in a more deliberately open-ended way, where the final form or material assembly of the building is determined by an iterative process involving a dialogue between many different but interconnected design considerations, including site and programmatic constraints, structural or mechanical requirements, fabrication and transportation factors, energy performance and occupant comfort goals, and spatial or aesthetic ideas. By creating rapid, visceral feedback loops, parametric modeling allows designers to discover the trade-offs and synergies between a very wide variety of design objectives and constraints, and to use this feedback to work towards a solution that best satisfies both the design vision and the demands of the client or context.

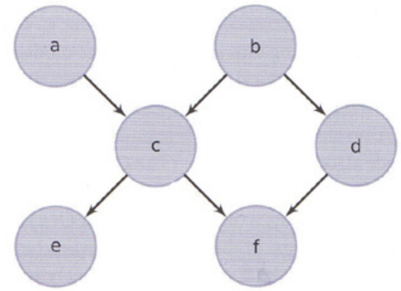


Figure 2.3-1 *A basic topological sort. Sorting must proceed from known quantities to unknown quantities. Some possible sorting orders are {b,d,a,c,f,e}, {a,b,c,d,e,f}, {a,b,c,d,f,e}, and {b,a,c,e,d,f}. If node "f" changes, there is no need to recompute the values of any other nodes. If node "a" changes, however, then nodes "c", "e" and "f" must also be recomputed.*

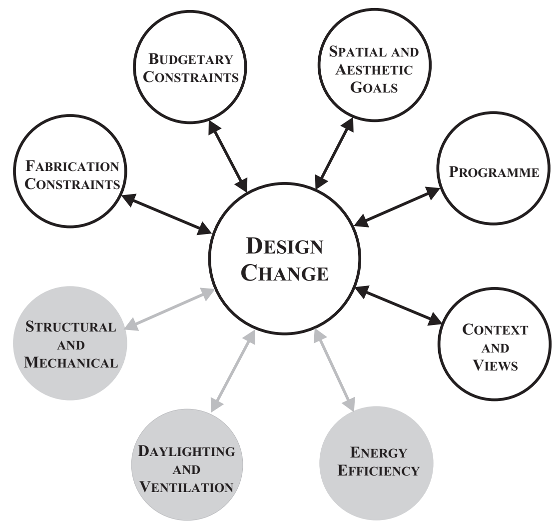


Figure 2.3-2 *When a design configuration is tested in a parametric model, feedback loops inform the designer of the implications of the choices they've made. Some feedback loops may be instantaneous, while others may be periodic or may require that an analysis or model update cycle be performed. In the above example, delayed updates are colored gray.*

Methods of Optimization

There are a number of methods by which parametric modeling can be used to optimize a design. The first method is to create explicit rules and constraints which either define the boundaries within which objects can be changed, or the shape or position of objects in relation to the current state of other objects or data in the model. Such rules can either be precise or approximate and "sketchy", depending on the stage of design's development and the complexities involved. The second way a parametric model can assist in optimization is through the automatic generation of statistics about the current state of the model. This data can be used to determine whether progress has been made towards particular goals. For example, model geometry or material properties can be used to generate real-time data on material costs or the overall thermal resistance of a building. The third means of optimization is through the use of proprietary or third-party analysis tools. Tools that integrate with the host modeling program are ideal but some forms of analysis may require that the model geometry be exported to another programme. Finally, the fourth means of optimization is through the use of mechanisms that alert the user when a design element has achieved an optimal position, shape or size, or alert the designer when objects or assemblies violate design constraints.

An important consideration when creating a parametric model is to set it up in such a way that the user can receive feedback appropriate to the stage of the design they are at, and at a speed that allows them to test propositions without frequent or lengthy interruptions. For example, in the early stages of design, a detailed daylighting analysis may be inappropriate and overly time-consuming, but a tool that gives the designer quick feedback using a common rule-of-thumb, such as the ratio of window height to room depth, might be very helpful. As the design progresses and the emphasis shifts more from exploration to refinement, a transition can be made from sketchy tools to more precise tools, or the same set of tools can be used, only with more precise geometry or more precise settings.

Because of the way objects are interconnected in parametric modeling, it can potentially be very computation-intensive. In order to maintain a smooth workflow it is often necessary to allow extraneous information to be turned off in order to speed up workflow. The detailed geometry that represents a building's actual construction can create large computational loads as well as visual clutter, and this geometry is not relevant to every design or analysis task. Sometimes it can be beneficial to create a second set of more

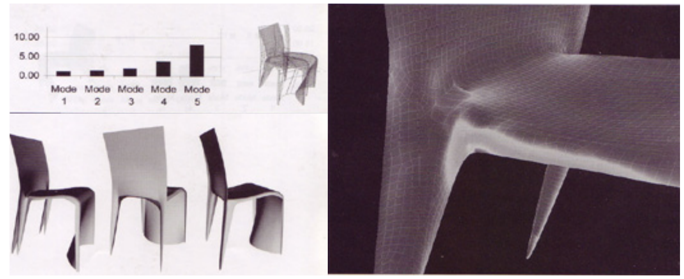


Figure 2.3-3 *A chair prototype is analyzed early in the design process using a "fuzzy" form of stress analysis known as "Static Eiegenvalue Analysis". The analysis is used to quickly indicate general strengths and weaknesses under different types of loading conditions.*[32]

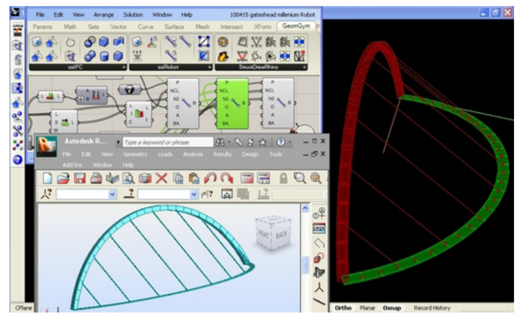


Figure 2.3-4 *The Geometry Gym plug-in currently under development for Rhino 3D facilitates the integration of Rhino and Grasshopper with third-party structural engineering programs such as GSA, Robot, SAP2000, Sofistik, Strand7 or SpaceGass.*

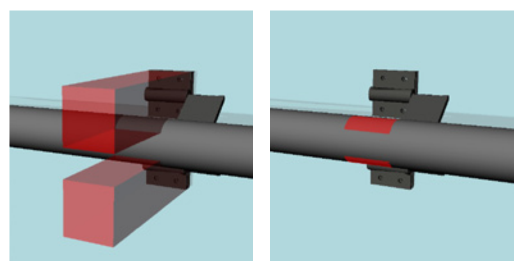


Figure 2.3-5 *The façade system developed for this thesis has a proprietary warning mechanism that informs the user when bolt access for the panels might be awkward or impossible during construction.*

abstract, simplified geometry to work with, as this can facilitate faster response times and therefore create better workflow and feedback loops. With large models, it can also be very helpful to create mechanisms for isolating portions of the model so that changes can be made to a particular part of the model without causing the software to recompute sections of the model that are not relevant to the task at hand.

<u>Two Case Studies</u>

The Smithsonian Courtyard Enclosure project by Foster and Partners presents an excellent example of the use of parametric design as a dynamic optimization tool. By embedding design, structural, fabrication and construction rules in a parametric model, the design team was able to edit and evaluate the roof design not just as a single monolithic form, but as a complex system involving many thousands of parts. This allowed the designers to optimize the structure's shape and composition with respect to many different design objectives simultaneously. The designers used the model to explore over four hundred different roof configurations over the course of six months, with each configuration having potentially over a hundred thousand unique components designed to be custom fabricated. The form and construction of the courtyard roof was influenced and constrained by many factors, including the visual profile of the roof as seen from outside the building, the need for acoustic dampening for concert events within the space, the need for water to properly drain off the roof through the columns, the need for the roof to behave as a grid shell structure that resolves to a minimum amount of support points independent of the existing structure[33], and the need to keep all roof components within certain construction, transportation and weight tolerances. As various roof configurations were explored, structural rules embedded in the model would appropriately increase the depth of the beams where stresses were greatest at the columns, as well as ensure that the beams always remained perpendicular to the roof surface. Meanwhile, an additional set of rules and constraints controlled the vertical positioning of panels relative to the beams and ensured that all panels remained planar. Ultimately, the parametric model allowed the design team to find a graceful form that satisfied all design objectives simultaneously.[34]

Foster and Partners used a somewhat similar approach to the design of their Great Canopy proposal for the West Kowloon Cultural District in Hong Kong. As with the Smithsonian project, the designers embedded structural and fabrication rules within a parametric model of the canopy, but

Figure 2.3-6 *For the Smithsonian Courtyard Enclosure by Foster and Partners a simple control surface in the 3D model controlled thousands of unique parts. An optimized solution for the roof shape was determined based on feedback from the model with respect to factors such as sight lines, drainage, structural performance, and acoustics.*

since the constraints of the project were less clearly defined at the outset, the designers intentionally made the parametric model more flexible and scalable by creating a modular system in which the roof could be extended horizontally in any direction by adding new sections. Additionally, the panel types could be changed anywhere on the surface of the roof according to requirements of the programme areas immediately underneath. Panel options included glass panels, aluminum panels, ETFE cushions, solar and photovoltaic panels, open trellis, and louvers. The structural depth of the space frame supporting the roof was designed to automatically vary according to local loading conditions and the positioning of columns, and this became another dynamic affecting the quality of the spaces underneath the canopy. The Great Canopy represents a good example of how the flexibility of parametric design can be used to optimize a structure for local conditions within the context of a greater design vision.[35]

Parametric design is as much about creating frameworks for design as it is about making the geometry itself, and the way in which a model is structured on a schematic level greatly affects how useful the model is in the long run. Even though parametric modeling allows structural, fabrication or construction constraints to be embedded in models at a relatively early stage in the design process, the various sub-assemblies in a design, such as cladding or structural systems, should generally remain modular and easily replaceable if possible. Construction assemblies are often provisional at first and need to be replaced as new information comes to light or as designers try different options. Designers may also wish to work on different systems within the model in parallel.

The associative nature of parametric design is its great strength, but can also be its Achilles' heel. If the organization of the model is such that it makes sub-systems difficult to modify or replace due to long-chain dependencies between objects, this can severely limit the flexibility of the system and make it impossible to replace systems or work on different systems in parallel. A fairly common and effective strategy for isolating assemblies is to attach each assembly directly to a "carrier" geometry instead of to one another. A carrier geometry is often the surface used to control the overall shape of the model, or some derivative of the control surface. By attaching assemblies to the carrier geometry instead of one another the relationships between assemblies can be made parallel instead of hierarchical, which allows individual assemblies to be removed or changed with fewer complications. This technique also allows sub-assemblies to be selectively disabled to reduce processing times when making changes to the model.

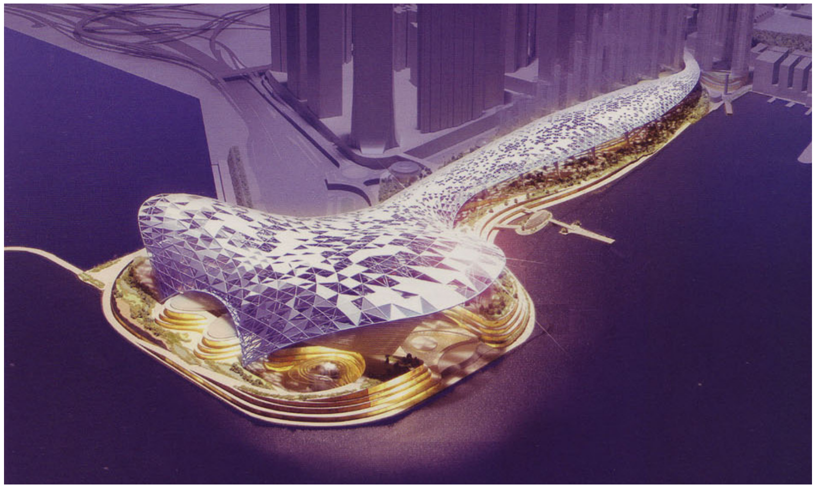


Figure 2.3-7 *Foster and Partners' "The Great Canopy" was a proposed for a continuously variable structure composed of hundreds of thousands of differentiated parts. The 40 hectare surface responds to local conditions and programme through a combination spatial variations and different panel configurations.*

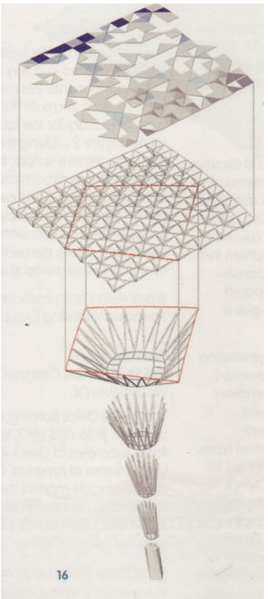


Figure 2.3-8 *Both the Great Canopy and the Smithsonian Courtyard Enclosure projects used strategies that isolated individual sub-systems as modular components so that they could be modified or replaced at will.*

In both the Smithsonian and Great Canopy projects, Foster and Partners used generative scripts to populate structural and panelling components onto carrier surfaces. The carrier surfaces were generally either the control surfaces used to shape the roofs, or derivative of these surfaces. Because of this topology, the generative scripts used to create sub-assemblies such as the truss systems could be made very modular, which in turn allowed designers to try out many different structural and cladding options as well as work on systems independently from one another.

When creating a parametric system, care has to be taken to find the proper balance between simplicity of inputs and sophistication of form. Too many parameters, or input geometry that's too granular, can make the system clumsy and cumbersome by under-constraining it and making it require too much user input. Too few parameters or overly-simplified input geometry, on the other hand, can severely limit the formal flexibility of the system and thus its customization or optimization potential. In the Smithsonian Courtyard project, the form and configuration of the roof system was determined by a combination of a simple curved surface for the roof, and a second surface that controlled the locations of the columns. The control/carrier surface was designed with a minimum amount of control points so as to assure ease of use and curvature smoothness. Ultimately the system provided a simple set of controls for a very complex structure with many thousands of components. In the Great Canopy project, the designers used a similar approach, only the canopy surface was broken down into smaller sub-sections, each having its own set control points in plan, section and elevation. Each section had a different amount of control points and was linked to adjacent sections using common surface tangency constraints to ensure continuity of the overall canopy surface. Like the interchangeable panels, this scalable and granular approach to input geometry allowed the design of the canopy to respond to and to be optimized for local conditions within the context of a greater design vision.

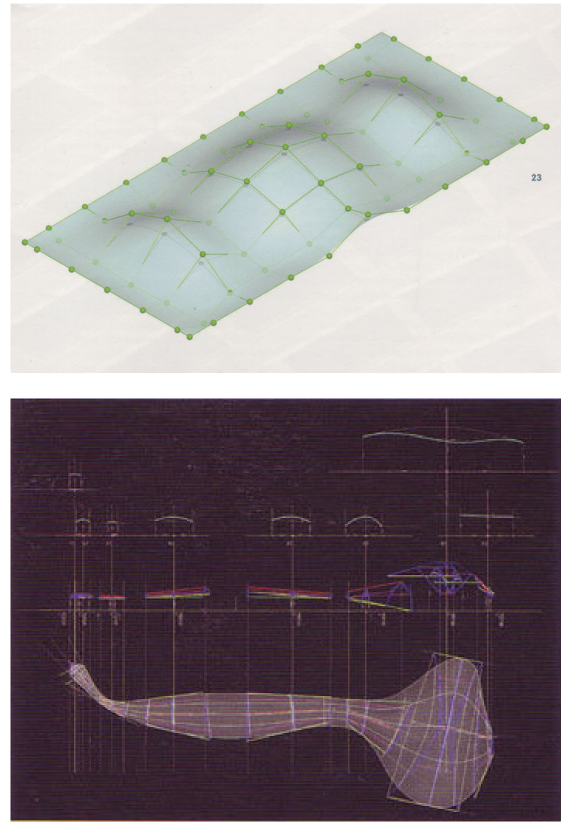


Figure 2.3-9 *The Smithsonian Courtyard Enclosure (top) used a surface with minimal control points to control the roof geometry, while the Great Canopy (bottom) used modules with plan, section and elevation control mechanisms that could be scaled according to local needs.*

3. Prototype Façade System and Parametric Model

3.1 Design Objectives and Overview

The façade system is conceived of as a flexible kit of parts and as an alternative to generic curtain wall systems. The parametric model is designed to be usable by someone with minimal Rhino 3D or Grasshopper 3D experience.

There were four primary objectives when creating the façade system which are outlined below. More in-depth descriptions of the façade system and its features can be found in the sections that follow.

Note: Many images show the façade geometry in a simplified state (e.g. without showing elements such as panel clips).

<u>1. Flexibility</u>

The first goal was to create a system that would be flexible enough to be used in many different design contexts as well as flexible enough so that it could be fine-tuned in response to the results of daylighting, energy or view analyses. Flexibility is approached in two principle ways: formal flexibility and material flexibility. Formal flexibility is achieved by using a triangulated paneling system that scales with the number input points created by the user. A triangulated system was chosen primarily because it can be used to create a wide variety of forms while panels always maintain planarity. Material flexibility is achieved by having four different possible panel types: opaque, translucent, glazed, and photovoltaic. All panel types are interchangeable, and the composition of any panel can be changed at any time by the user using an OpenOffice spreadsheet linked to the model.

<u>2. Integrated Analysis Tools</u>

The second objective when creating the façade system was to integrate custom and third-party analysis tools into the parametric model that would provide the designer with rapid feedback on view availability, daylighting, and energy efficiency. The analysis tools had to range from rough, rule-of-thumb tools which could be used in the more schematic design phases to provide very general and quick feedback, to much more precise and slower

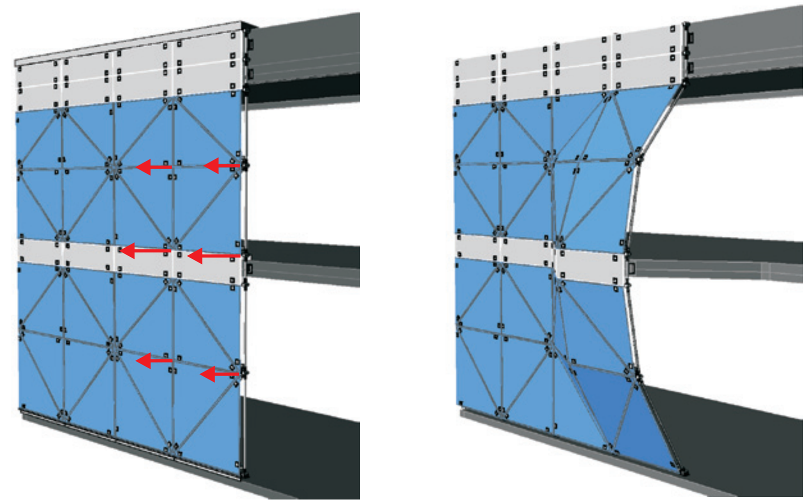


Figure 3.1-1 *The façade system uses vertex points to define the corners of panels, and these points can be moved freely horizontally in order to create many different shapes.*

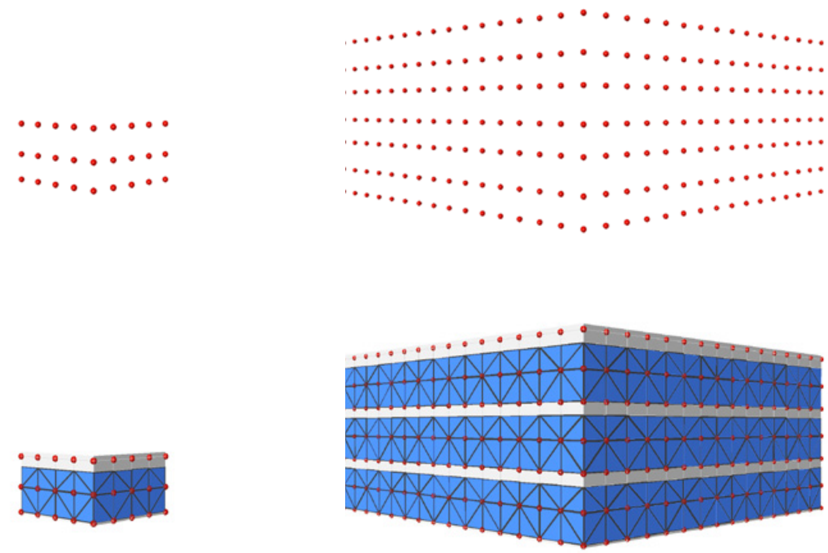


Figure 3.1-2 *The façade system is scalable based on the number of Rhino input points.*

tools that could be used to fine-tune the façade's shape and material composition in later stages of a design's development. These tools are discussed in more detail in section *3.8: Analysis Tools and Modes*.

3. Simple Fabrication and Construction

The third goal for the façade system was to engineer it in such a way that fabrication and construction processes would always be consistent, and that costs would always be relatively predictable regardless of the façade's configuration. To this end, the entire façade assembly is designed so that the majority of parts can be created from planar sheets of material or a simple extrusions by using 2-dimensional CNC cutting machines such as plasma, laser or water-jet cutters. Even with today's CNC technologies, the creation of custom moulds or extrusions still tends to be very labor intensive and costly. CNC cutting and milling, however, represents the "low hanging fruit" in the emerging flexible manufacturing economy since these techniques employ relatively common machines that are relatively straightforward to setup and can use simple, standardized feedstock.[36]

Panels are prefabricated off-site and then installed using a straightforward procedure that varies little with different façade configurations. This is discussed in more detail in section *3.5: Façade System Construction.*

4. Smooth Design Workflow

The final objective when designing the façade system and the parametric model was to make the workflow as smooth as possible for the user so that feedback loops remain as rapid as possible. Ther were three aspects to this:

1) Making all model inputs and parameters as simple and intuitive as possible.

The façade's shape can be manipulated using a simple series of input points, and all model parameters are grouped together in categories within Grasshopper for easy access.

2) Allowing the user to eliminate levels of detail unnecessary for given task in order to speed up computer response times.

The model has a number of built-in mechanisms for reducing the amount of

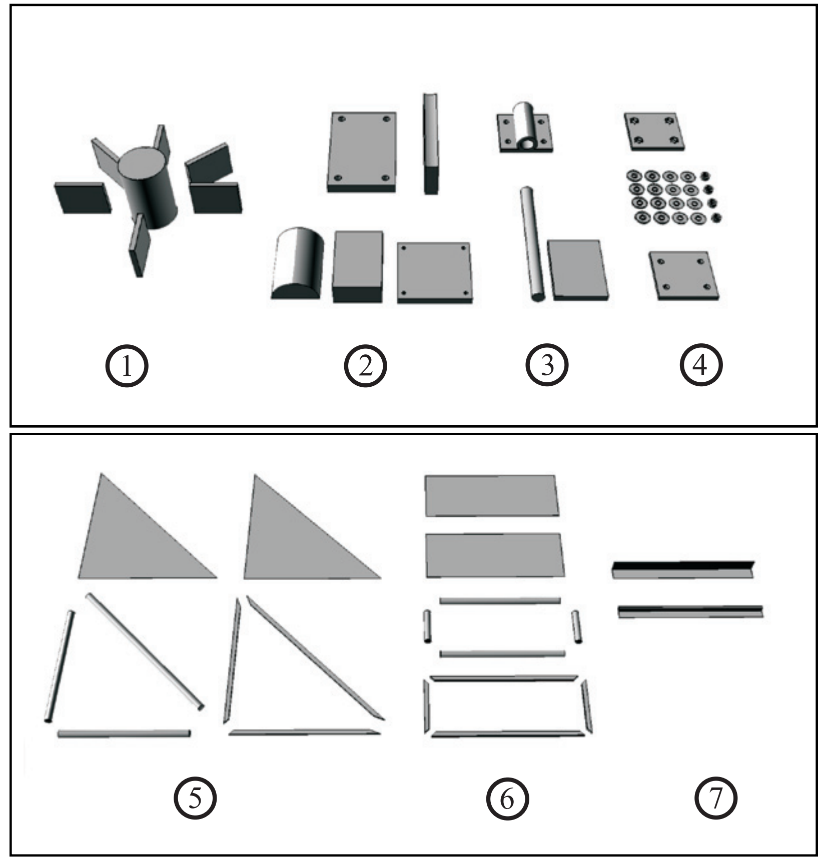


Figure 3.1-3 *Façade components that can be cut from simple extrusions or planar materials: 1) Truss hub components; 2) Truss-to-slab attachments; 3) Hinge assembly; 4) Panel clip components; 5) Triangular panel components and truss members; 6) Spandrel panel components and truss members; 7) Façade base panel assembly and floor-façade gap covers.*

calculations the computer has to perform when an input or parameter in the model is changed. These include: the ability to isolate and work on small sections of the façade, the ability to view simplified versions of the façade geometry, and the ability to disable or hide from view any sub-assembly within the façade system.

3) Segregating all sub-assemblies within the model so that they can be edited or replaced without greatly affecting other components or assemblies.

The model is schematically structured in such a way that all sub-assemblies in the façade system, such as the slabs or the panel clips, can be edited without creating cascades of dependency-related errors.

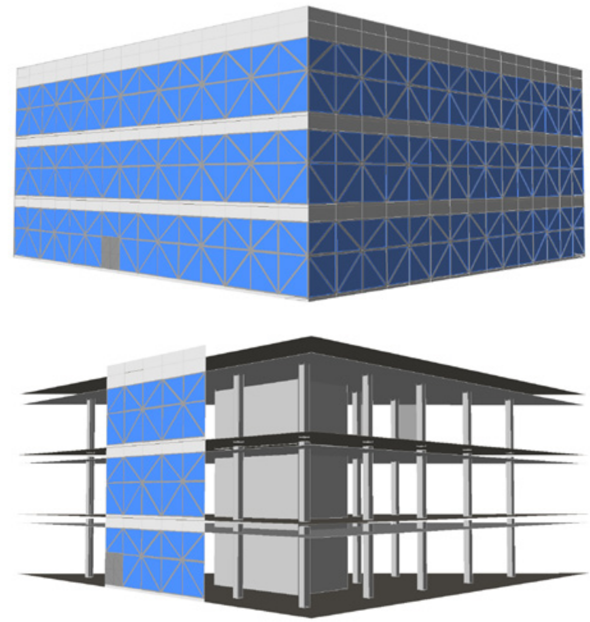


Figure 3.1-4 *Specific sections of the façade can be worked on in isolation.*

3.2 Choice of Software

Three different software packages were evaluated before creating the façade model: Generative Components, Grasshopper 3D, and Digital Project (an architectural version of CATIA). Grasshopper and Generative Components are "generative" modelers that use algorithms to propagate arrays of objects, while Digital Project has more of a traditional CAD interface. Ultimately it was decided that a generative modeler would be best for creating scalable façade system with panel types that could be changed rapidly.

Of the two generative modelers evaluated, Grasshopper was eventually chosen due to its more intuitive interface and the support offered by its community of users. One downside of using Grasshopper however was that the software was officially a Work In Progress (WIP) and not a commercial-grade product. While the software proved to be mostly stable and bug-free, there were some significant glitches that required time-consuming work-arounds (see Section 5: *The Modeling Process)*, and the program also had a tendency to crash when large amounts of geometry were created.

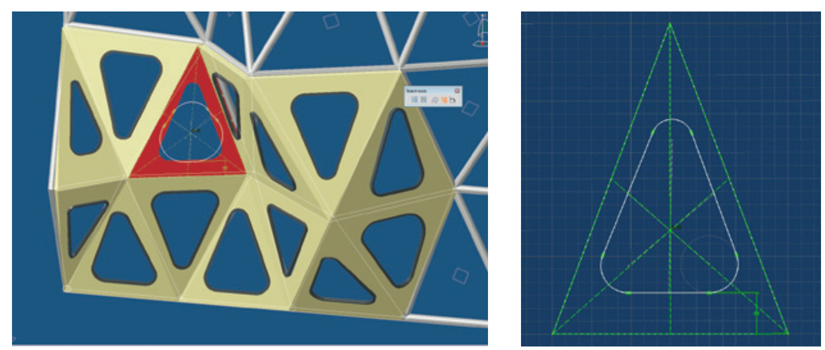


Figure 3.2-1 *An early prototype for the design used Digital Project (CATIA), which allowed for much more granular control of geometry but had less robust tools for propagating arrays of components. Left: a window shape is edited with the mouse. Right: unique two-dimensional drawings are linked to each panel face and used to control window shapes.*

3.3 Model Schematic Structure

The entire façade assembly is generated from simple rows of Rhino input points which correspond to the corners of the façade panels. At the heart of the model is a series of algorithms that reorganize the Rhino input points so that they can be properly referenced by various sub-assemblies within the model. For example, to create triangular panels, point rows are organized into pairs and then points within adjacent rows are organized into groups of

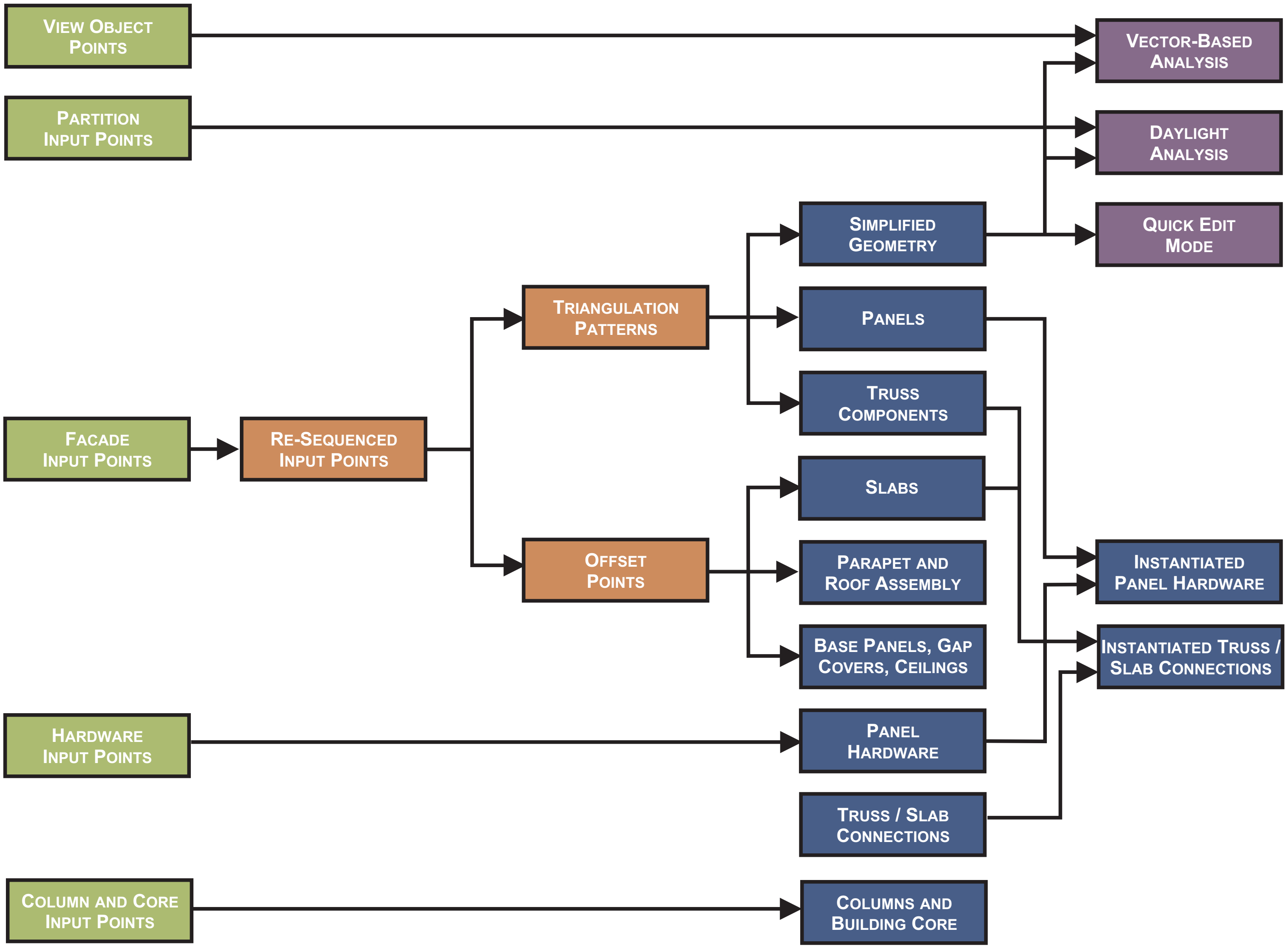


Figure 3.3-1 *A schematic flowchart of the Grasshopper model. There are four broad categories of systems: geometric inputs (green), point sequencing and offset algorithms (brown), façade sub-assemblies (blue), and analysis and statistical mechanisms (purple). Note how all sub-assemblies run in parallel to one another.*

three. If only part of the façade is being viewed, only portions of the point rows are used instead of the entire rows.

The model is designed so that all sub-assemblies are offset directly from the input points. This creates parallel as opposed to hierarchical relationships between sub-assemblies and eliminates most long-chain dependencies within the model. As a result, most sub-assemblies can be can be disabled, edited or replaced without much impact on other assemblies, and advanced users can fairly easily identify and replace sub-assemblies or systems within the model.

3.4 Model Inputs and Behavior

There are two types of inputs for the model: geometric inputs, which are point objects in Rhino that can be moved with the mouse, and numeric inputs, which are parameters controlled by numeric sliders or boolean true/false switches in Grasshopper. All input points, such as those that control the shape of the façade, are first created by the user in Rhino and then referenced into Grasshopper. All numeric parameters, such as the dimensional thicknesses of the panels, are located in the top-left corner of the Grasshopper canvas, along with all key model statistics such as the window-to-wall ratio and slab-to-slab heights.

Each row of points used to generate the façade corresponds to a dividing line between panel rows. The amount of points per row determines the number of panels per row. Moving any of the points moves the corners of adjacent panels and changes the overall shape of the façade, much in the same way that editing the vertices of a mesh in a typical 3D modeling program changes the shape of the corresponding object. Individual input points can be moved freely so long as they remain on the same horizontal plane. There are no limits to the amount of input points that can be used or the amount of panels beyond what the host computer can handle in terms of computation.

3.5 Façade System Construction

As discussed previously, two important objectives in the design of the façade system were to make it flexible and to allow it to be constructed in a straight-forward way using easy-to-fabricate parts. This posed a challenge when creating the panel system.

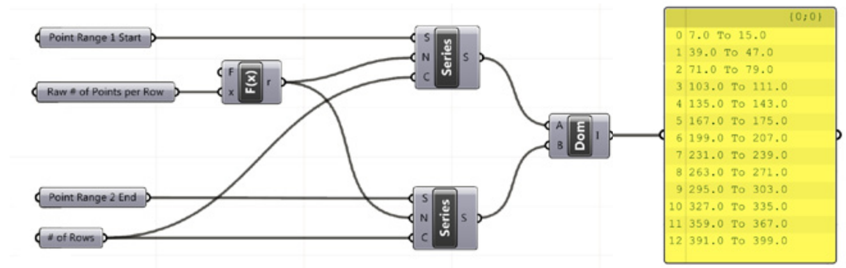


Figure 3.4-1 Abo*A point algorithm that creates a sub-set of the façade's input points in order to generate only a portion of the façade geometry.*

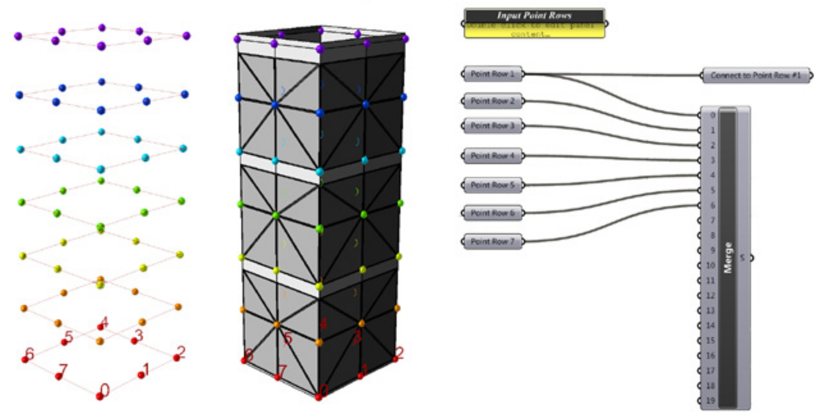


Figure 3.4-2 *Left and center: the input points for the façade correspond to the corners of triangular and spandrel panels. Right: a simple mechanism in Grasshopper allows point rows to be added or removed from the model.*

The use of unique mullions for each panel joint would have been impractical, because custom extrusions are expensive to fabricate in small batches. Identical extrusions can offer some degree of flexibility, in terms of panel angles and thicknesses, but not as much as was needed. A compromise may been possible - using a finite number of custom extrusions, each with some flexibility in the joints - but this would have created its own set of potential challenges. For example, from a modeling perspective, one of the challenges would have been to have the correct type of mullion be automatically generated when needed and then joined properly with other mullions. And from a construction perspective, the complexity of joints in such a system could make installation difficult as well as create potential leaks.

A clip-based system with a secondary support truss was thought to be a better option. The panels are attached to the truss via a hinge mechanism that can accommodate a very wide range of angles between adjacent panels as well as different panel thicknesses and different dimensions or offsets for the truss and panel clips. This system behaves somewhat like a spider joint system, but each clip pivots and attaches to only one panel in order to allow for different angles between panels.

While a square clip and hinge system was chosen for the façade system prototype, there are a number of different ways to approach the design of the hardware. For example, a combination of a hinge and a spider joint might be possible. The key design requirements are:

1) The distance between the clip and the truss must be variable due to the different panel thicknesses.
2) Each attachment must pivot to allow different joint angles
3) The system must allow the panels to be prefabricated
4) All hardware should be identical except for the hinges
5) Hardware should be accessible for installation, from inside the building

Figure 3.5-1 *Top: a typical articulated spider joint system for double-paned glass. Bottom: a mullion with a joint that allows some variation in panel angles.*

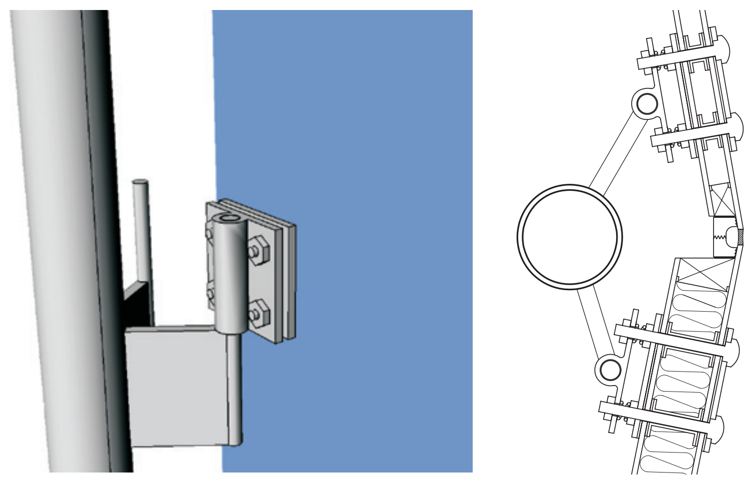


Figure 3.5-2 *The façade system utilizes a flexible hinge and clip assembly that can adapt to different joint angles as well as different panel thicknesses.*

Panel Assembly

There are four panel types: glazed, opaque, translucent and solar photovoltaic. All panels are of a sandwich-type configuration composed of an outer pane, a cavity with wood spacers, an inner pane, and clips to hold the assembly together. The hinges are attached later during panel installation. All panels are designed to be prefabricated off-site using parts that can be cut from sheet material or simple extrusions.

Panel Clips and Hinges

The number of clips per panel is variable. All clip components, with the exception of the bolts, can be fabricated from simple planar or extruded elements.

The clip assembly includes spacer elements with seals which allows it to be used for both standard insulated panels and gas-filled double glazed panels. The hinge mechanisms are attached to the panel clips on-site during the installation process (see *Panel Installation* below). Springs between the hinge plates and panel clips as well as over-sized bolt holes on the hinge plates allow for some fine-tuning of the alignment of each panel relative to neighboring panels. The number of clips per panel and the size of the clips will vary from design to design, depending on structural requirements.

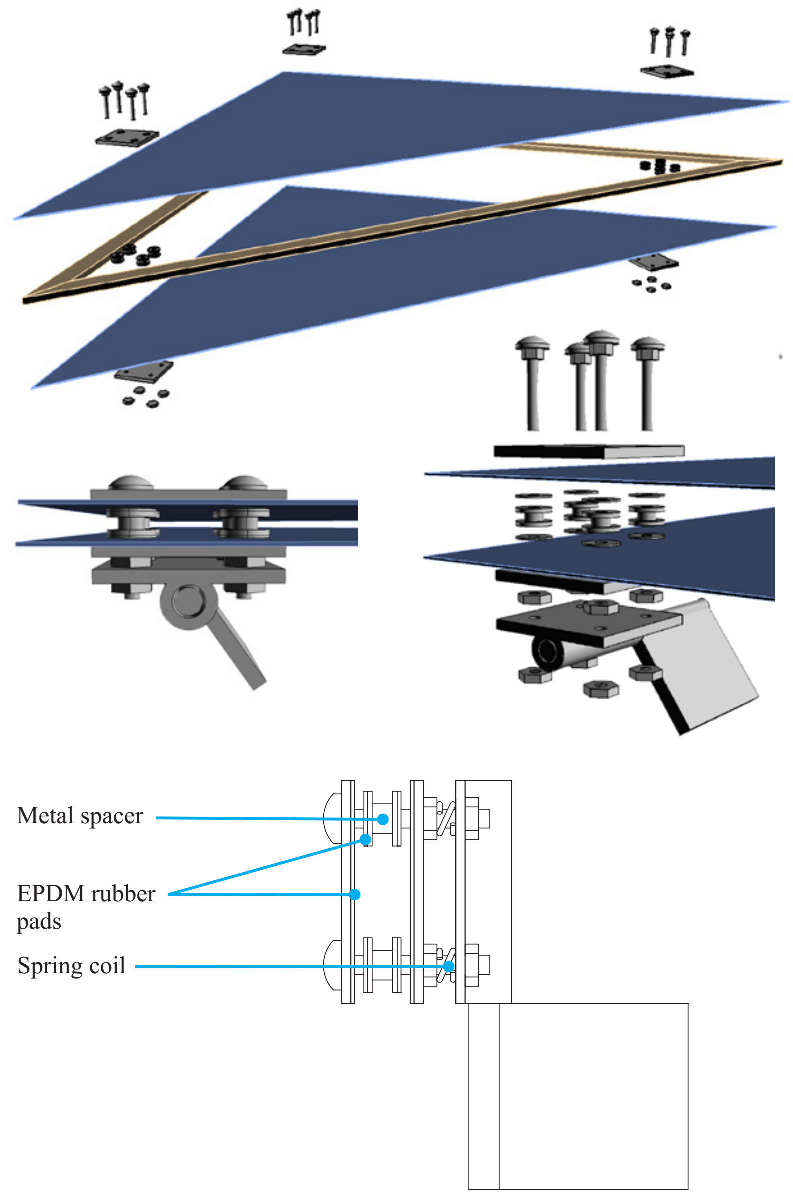


Figure 3.5-3 *Panel and clip assemblies.*

Panel Installation

The process of installing the panels for the façade system is fairly straightforward and does not change significantly with different façade shapes or panel types. The panels are installed by suspending them individually with a crane while the hinge plates are attached and tightened. An analysis tool in the parametric model allows the designer to ensure that the bolts for any given panel can be accessed easily during construction and will not obscured by truss elements (see *Panel Clip Inaccessibility Warning* in section *3.8: Analysis Tools and Modes*).

Panel Joints

The panel joints have a double-drained topology. The primary exterior seal for the panel joints is silicone, but because the panels can have semi-horizontal orientations like skylights, a secondary layer of protection is necessary to catch minor leaks. The secondary layer is composed of neoprene gaskets that are attached to each vertically-oriented panel edge. When the panels are installed, pressure between adjacent gaskets creates a seal and a drainage trough between the panels. These drainage troughs empty out at open troughs along the horizontal joints where the water can sit and evaporate.

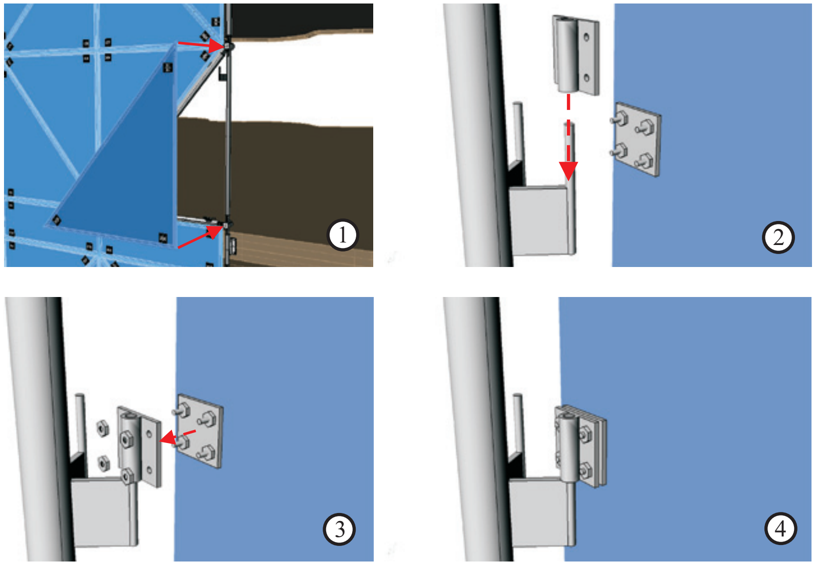


Figure 3.5-4 *Panel installation sequence.*

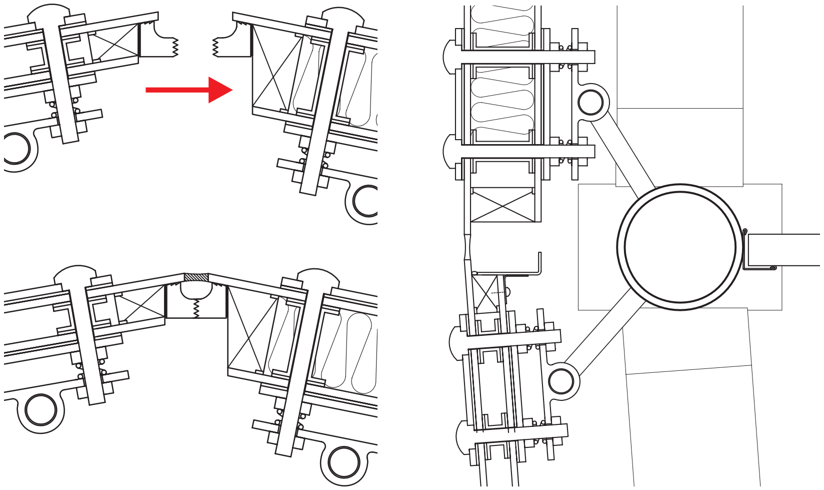


Figure 3.5-5 *Left: the secondary drainage system for vertical joints, created by the pressurized interlocking of two neoprene gaskets. Right: the drainage trough at horizontal joints.*

Panel Types

Any material composition and thicknesses can be used for the panels as long as the basic topology remains unchanged. The default panel types and material compositions are as follows:

Glazed panels

Outer layer: *High transmittance tempered glass*
Cavity: *Inert gas, sealed with gaskets*
Inner layer: *Low-e coated tempered glass laminate with SPD film.*

Opaque and spandrel panels

Outer layer: *Tempered glass with opaque film of any color*
Cavity: *Aerogel insulation*
Inner layer: *Tempered glass with opaque film of any color*

Translucent panels

Outer layer: *Tempered glass*
Cavity: *High-translucency aerogel insulation*
Inner layer: *Tempered glass*

Photovoltaic panels

Outer layer: *Photovoltaic cells embedded in high transmittance tempered glass*
Cavity: *Aerogel insulation*
Inner layer: *Tempered glass with opaque film of any color*

Instead of using internal shading devices, the glazed panels use Suspended Particle Device (SPD) glass, which allows users to control light transmission properties the of the glass electronically. While this technology uses a small amount of electric current (as little as 0.65 watts per square meter when the glass is clear[37]), the energy consumption can be offset by reduced cooling loads. Unlike venetian blinds, which do little to mitigate solar heat gains, SPD glass can prevent heat from entering the building.[38]

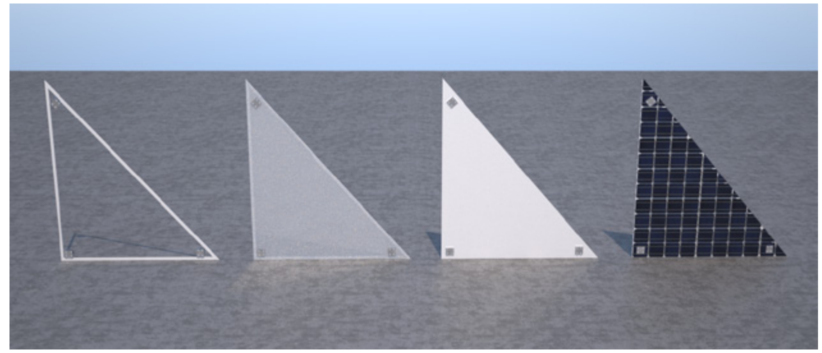


Figure 3.5-6 *Different panel types: glazed, translucent, opaque, and photovoltaic. Opaque and spandrel panels can be any color.*

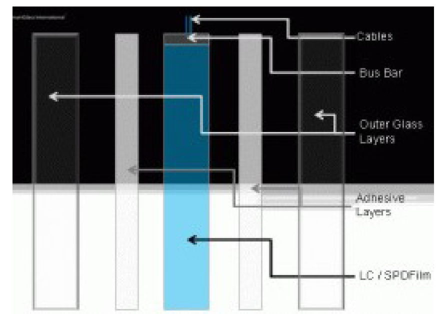


Figure 3.5-7 *Suspended Particle Device (SPD) glass uses a small current to control the opacity of windows.*

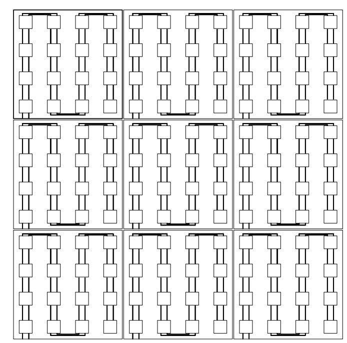


Figure 3.5-8 *Photovoltaic cells laminated in glass (left) and sample schematic (right).*

Solar panels use arrays of photovoltaic cells laminated in high transmission glass. Thick crystal photovoltaic arrays can produce around 10-12 watts per square foot, or 108-129 watts per square meter.[39]

Truss System

The complete resolution of the truss system is beyond the scope of this thesis, so the system must be considered provisional. A hub and spoke system with tubular members was chosen for the prototype, but there a number of possible approaches.

It is important that truss elements can be transported in pieces and assembled on-site without any need for complex alignment processes or elaborate welding during installation. A bolted joint may be ideal for this, because it allows truss members and hubs to be transported to the site individually while also allowing the most sensitive and precise alignment and welding tasks to be done in the controlled environment of the shop.[40]

It is critical that the slab truss-to-slab connections allow for construction tolerances and that the slab positions are verified before the truss connections are finalized.

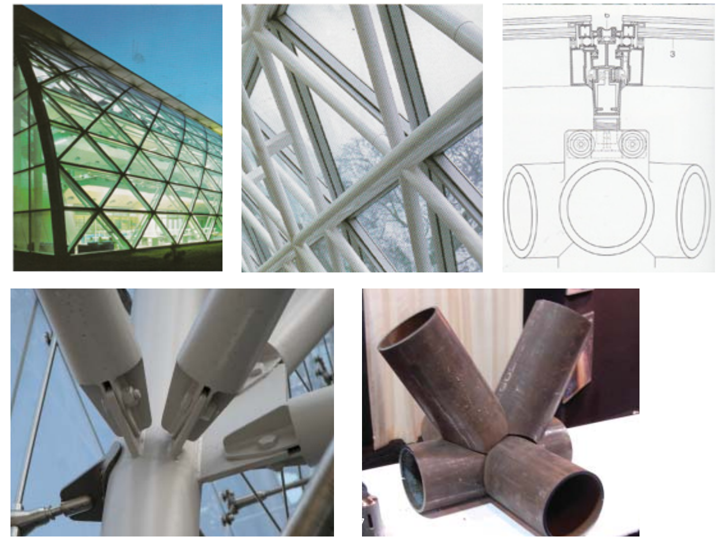


Figure 3.5-9 *Top: the Cambridge Faculty of Law building by Foster and Partners uses tubular truss with slotted joints. Bottom: examples of connections for tubular steel structural members.*

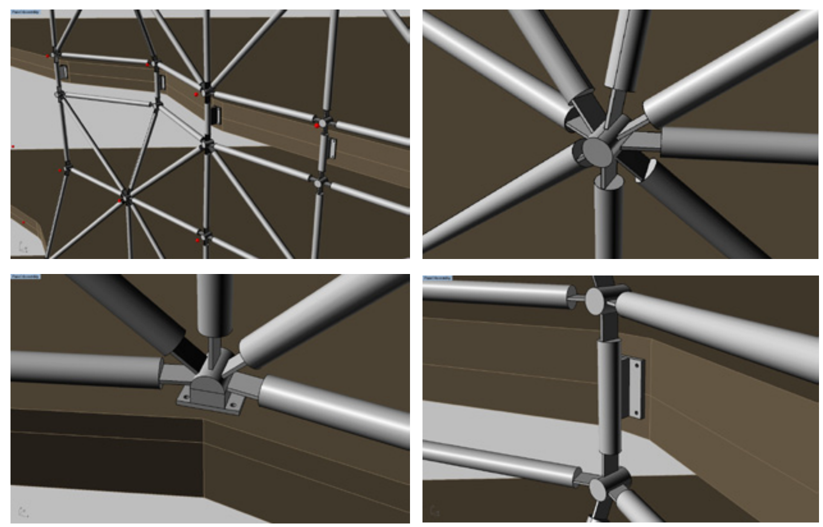


Figure 3.5-10 *The truss for the façade system is designed to be able to form a wide variety of shapes using mostly simple extruded metal elements.*

Parapet and Roof Assemblies

These assemblies are cut-to-fit on site and do not require elaborate off-site preparation or pre-fabrication processes. The assemblies are therefore represented in the model only as single objects and detailed in two-dimensional drawings instead. Key dimensions and characteristics of the assemblies can be controlled in the model, such as the angle and thicknesses of the parapet, the roof assembly thickness, and the thickness of the inner wall assembly.

Façade Base Panel Assembly

The façade base panel assembly is designed to provide thermal and weather protection as well as proper drainage under the bottom row of panels. Like the parapet assembly, the base panel components are cut-to-fit on site and the assembly is detailed in two-dimensional drawings. Key properties such as assembly thickness, drainage angle, material thickness, and drip-edge height can be controlled in the model.

Slabs

All floor and roof slabs are generated automatically according to parameters set by the user such as thicknesses, offsets from the façade, and curb dimensions. The ground floor slab extends outwards further than the other slabs and has a curb that extends to meet the truss from underneath. The roof slab has an extended curb that represents the core of the parapet. Note that the curb is not actually intended to be continuous like it appears in the model. In reality, it only needs to extend the full height of the parapet periodically where truss-to-slab attachments occur.

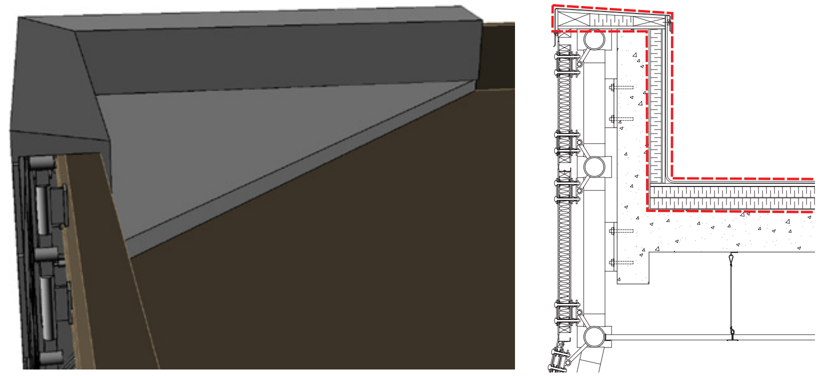


Figure 3.5-11 *The parapet assembly in the model (left) and a 2D detail (right).*

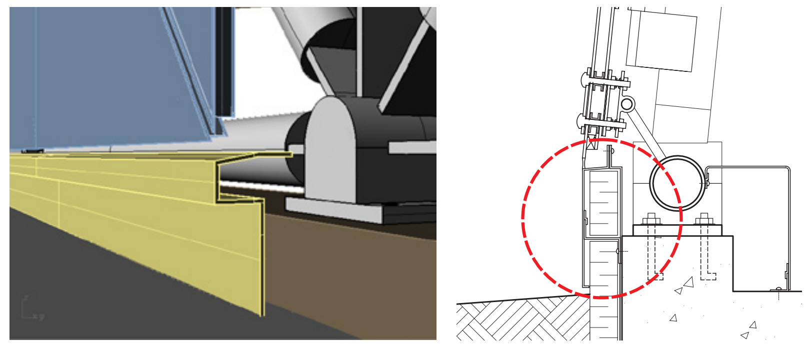


Figure 3.5-12 *The façade base panel assembly in the model (left) and a detail (right).*

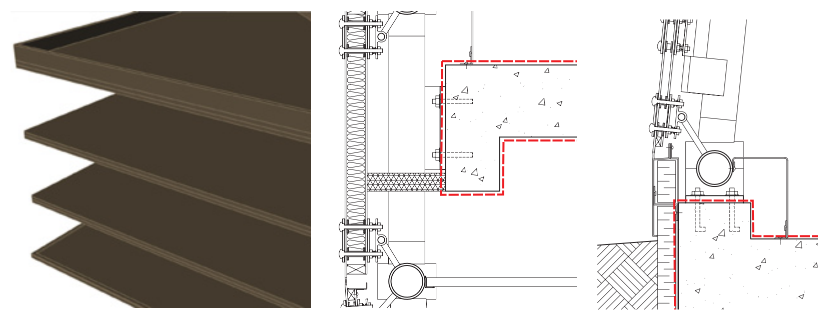


Figure 3.5-13 *Left: floor slabs in model. Center: typical floor slab. Right: at grade.*

Façade-Slab Gap Covers

These assemblies cover the gap between the façade and the floor slabs as well as provide a service space for heating ducts or for electrical conduits (e.g. for the solar panels or SPD glass). The assemblies are cut-to-fit on site and are represented in simplified form in the model. Parameters such as the width, material thickness, and clearance distances around truss hubs are controlled by the user. The height of the assembly is determined automatically by the distance between the slab and the center-lines of the horizontal truss members at the bottom of the first triangular panel row.

Ceiling Assemblies

The ceiling assemblies are standard, modular hung assemblies represented in a simplified form in the model and detailed in two-dimensional drawings. In the model, the user can directly control the thickness of the assembly as well as the clearance distance around truss hubs. The ceiling assembly lines up automatically with the center-lines of the top horizontal truss members on each floor. Ceiling cavity depth is determined by the distance from these truss members to the slabs above.

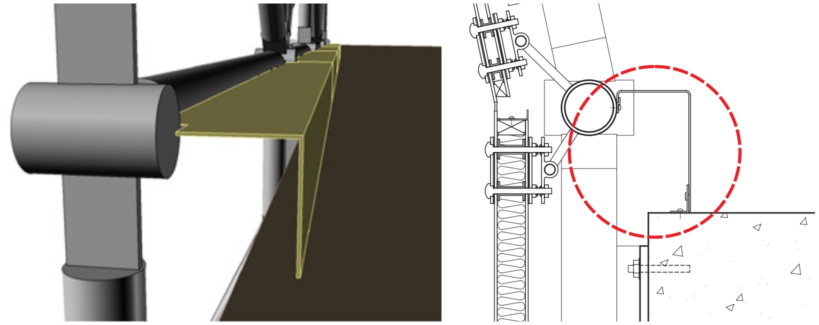


Figure 3.5-14 *Façade-Slab Gap covers span the gap between the slab and the façade and provide a service space for heating or electrical conduits.*

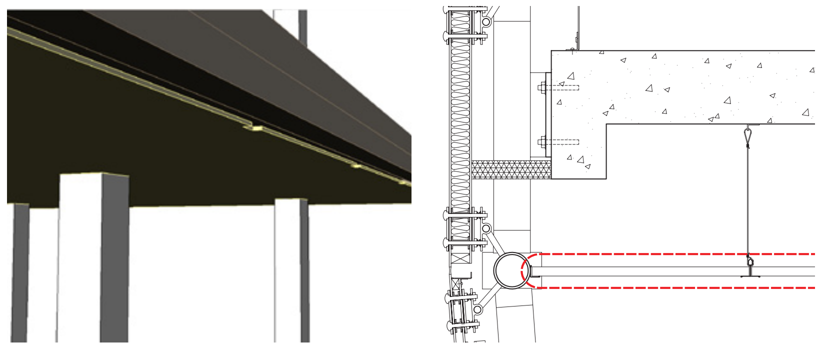


Figure 3.5-15 *Ceiling assemblies are automatically aligned with the top horizontal truss member.*

Columns and Building Core

Column locations and the building's core are generated using user-defined Rhino input points on the XY plane of the model. The columns and core are simple placeholders and contain no useful construction information. There is no limit to the number of columns or the number of columns and the core can be any shape.

Light Shelves

When the façade is configured for two rows of triangular panels per floor there is the option to add light shelves between panel rows at any location. Light shelf locations are designated using the same spreadsheet that controls panel types and empty bay locations (see section *4.3: Creating a Façade from Scratch* for more detail).

Basic dimensions of the light shelves can be controlled in the model as well as material properties for daylight simulations.

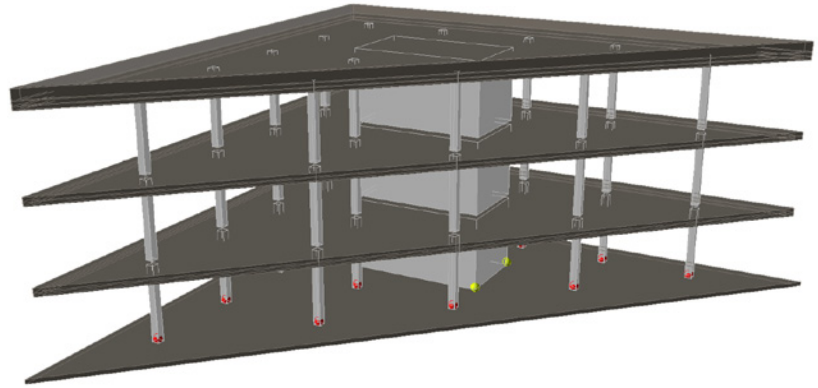


Figure 3.5-16 *Columns and building core are controlled by input points on the base plane.*

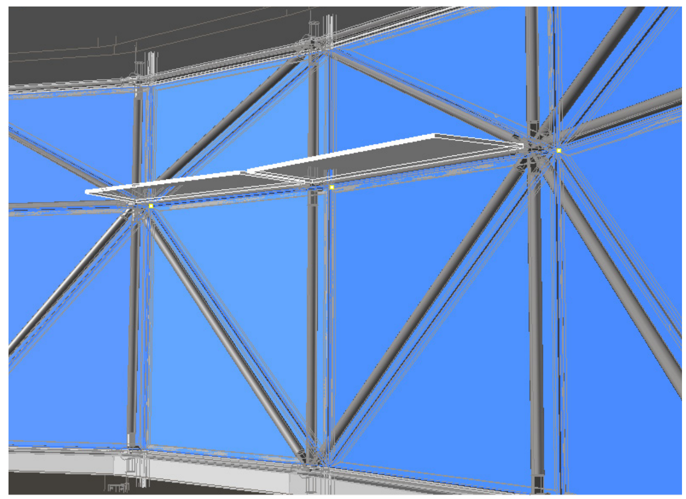


Figure 3.5-17 *Light shelves can be added in specific locations if there are two rows of panels per floor.*

Figure 3.5-18 *Rendered view of the façade system*

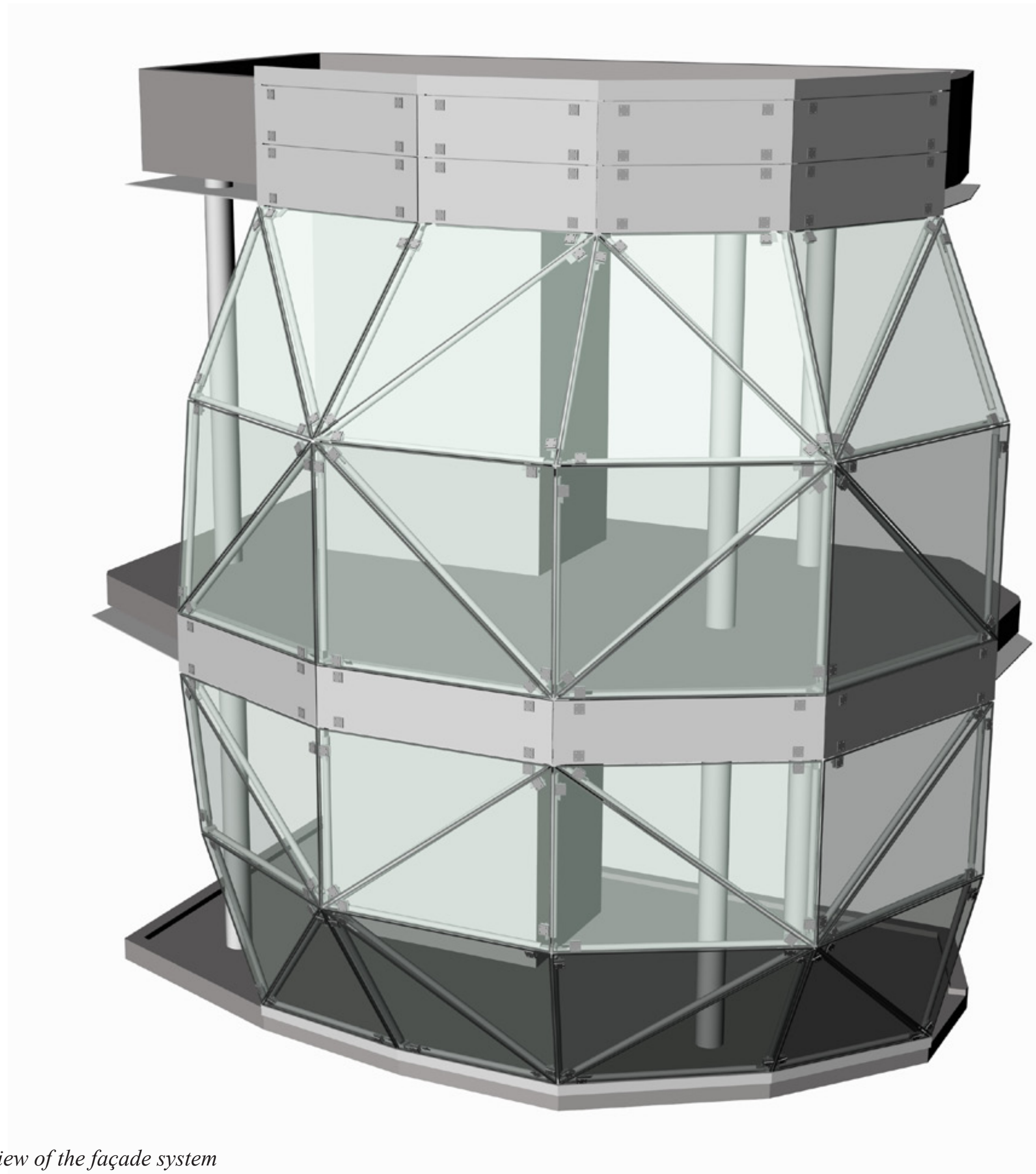


Figure 3.5-19 *Rendered view of the façade system*

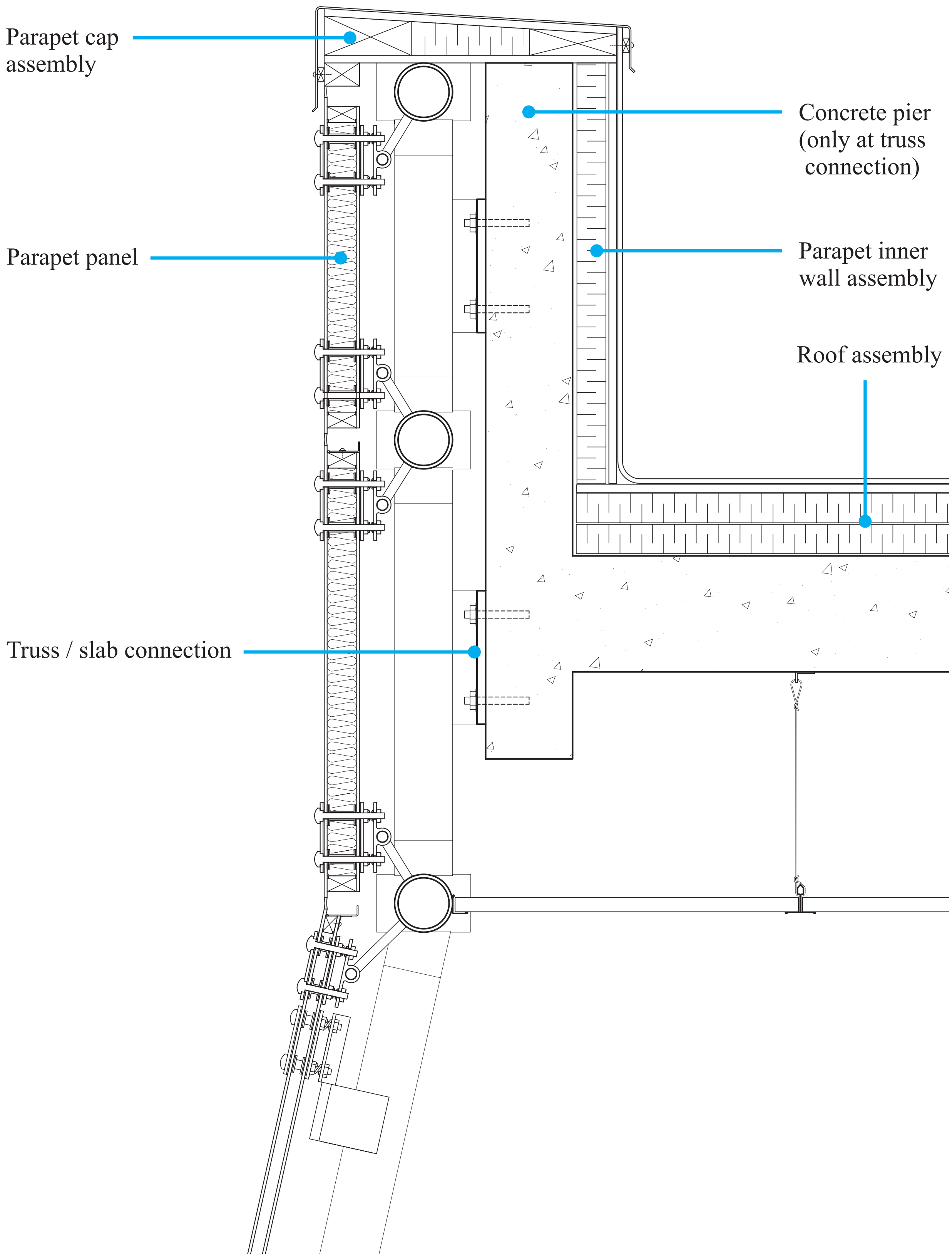


Figure 3.5-20 *Typical wall section at parapet. Scale 1:10 (sizes of façade components will vary according to structural requirements).*

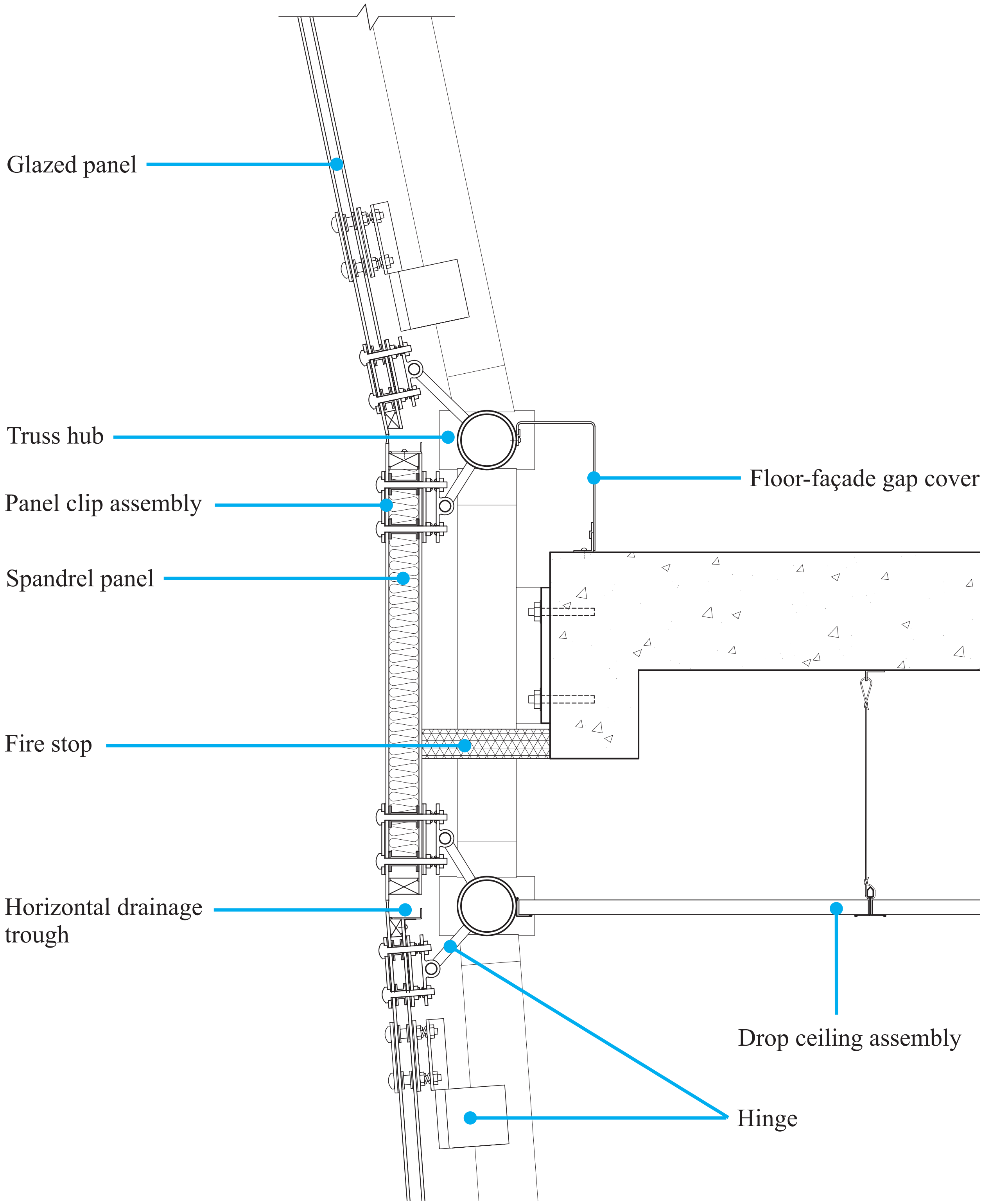


Figure 3.5-21 *Typical wall section at floor slab. Scale 1:10.*

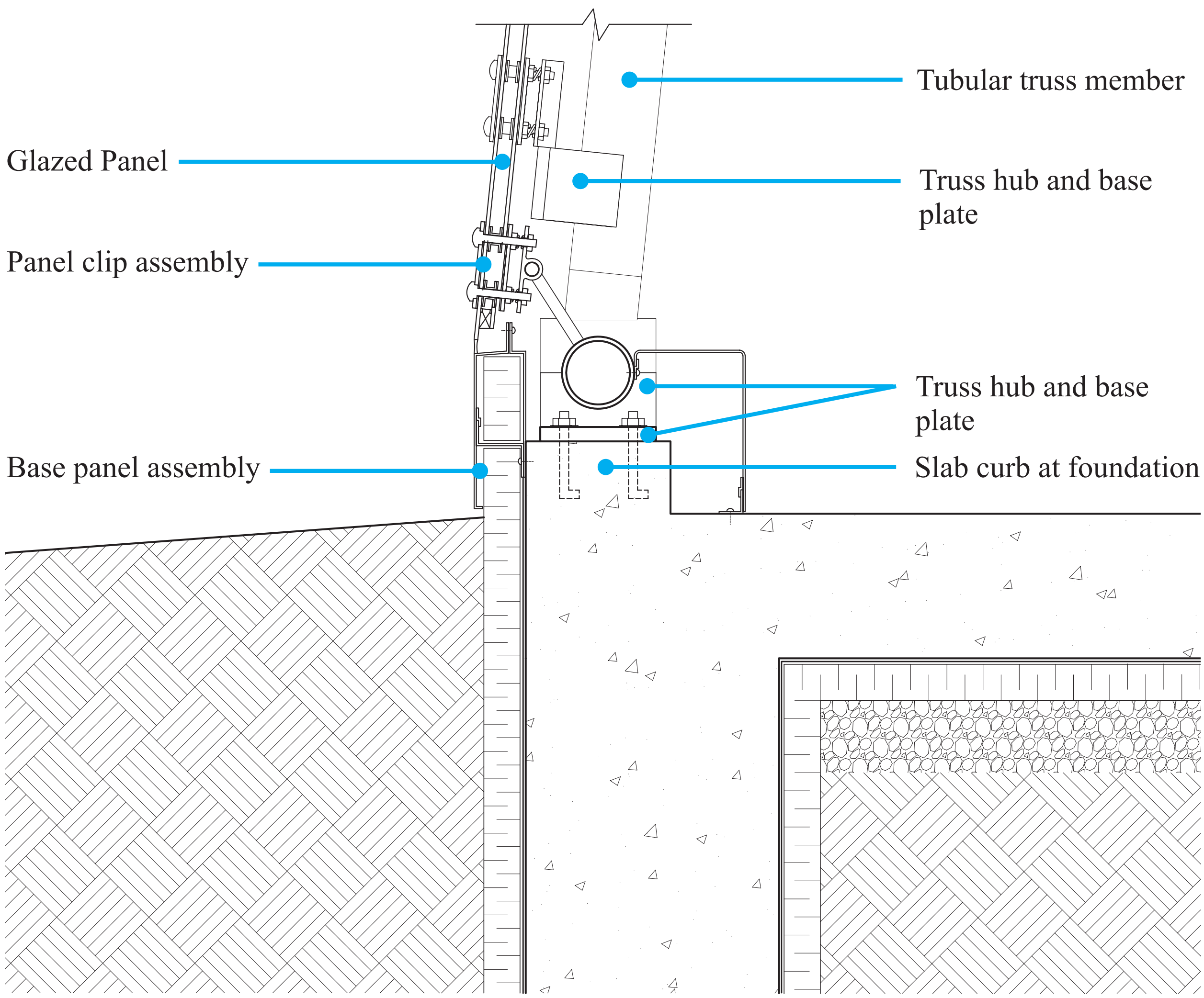


Figure 3.6-22 *Typical wall section at foundation. Scale 1:10.*

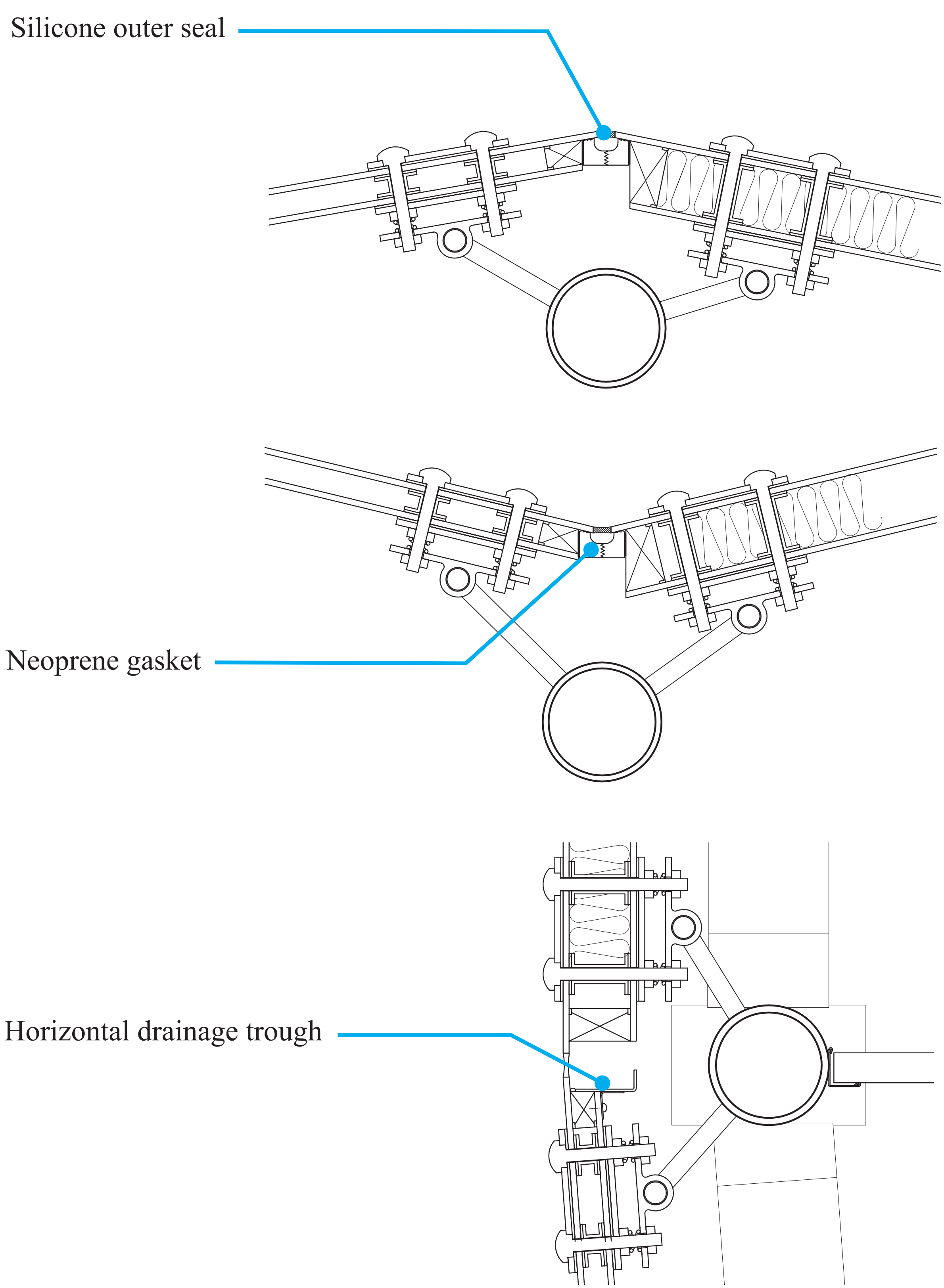


Figure 3.6-23 *Vertical joint details (top, center) and horizontal joint with drainage trough detail (bottom). Scale 1:5*

3.6 Key Model Parameters

There are over two hundred adjustable parameters in the model. Some of the more critical variables affecting the character of the façade as well as aspects of the model workflow are explained below. Detailed descriptions of every model parameter can be found in an *Appendix A: Model Parameters and Statistics.*

Evaluation / Modeling Mode

This parameter determines what modeling or analysis mode is active and turns geometry or features in the model on or off accordingly. The options are:

0. Hardware Definitions Only
1. All Detailed Geometry Enabled
2. Quick Edit Mode
3. Panel View Angle Evaluation Mode
4. Panel Vertical Angle Evaluation Mode
5. Solar Panel Evaluation Mode
6. Daylight Analysis Mode
7. Panel Southern Exposure Evaluation Mode
8. Panel Direction Evaluation Mode

The "Hardware Definitions Only" mode disables everything in the model except for the geometry that defines the panel clips, panel hinges, and truss-to-slab attachments. This mode is used to edit these elements quickly without simultaneously instantiating them (which requires a lot of processing). The "All Detailed Geometry Enabled" mode enables the full geometry of the façade, however the user can still enable and disable various components manually if they choose.

"Quick Edit Mode" uses simplified geometry for the panels but maintains a sense of the look and feel of the actual façade. This mode is designed to allow the user to change the shape and material composition of the façade very quickly. The various analysis modes also substitute the normal façade geometry with simplified or specialized geometry in order to give the user rapid feedback when manipulating the shape of the façade. These modes are discussed in more detail in section *3.7: Analysis Tools and Modes.*

Figure 3.6-1 *There are over 200 model parameters grouped according to function on the Grasshopper canvas*

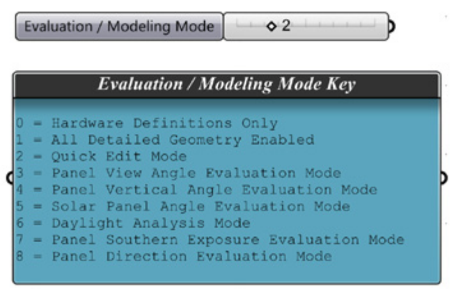


Figure 3.6-2 *There are nine different modes in which the model can be evaluated or manipulated*

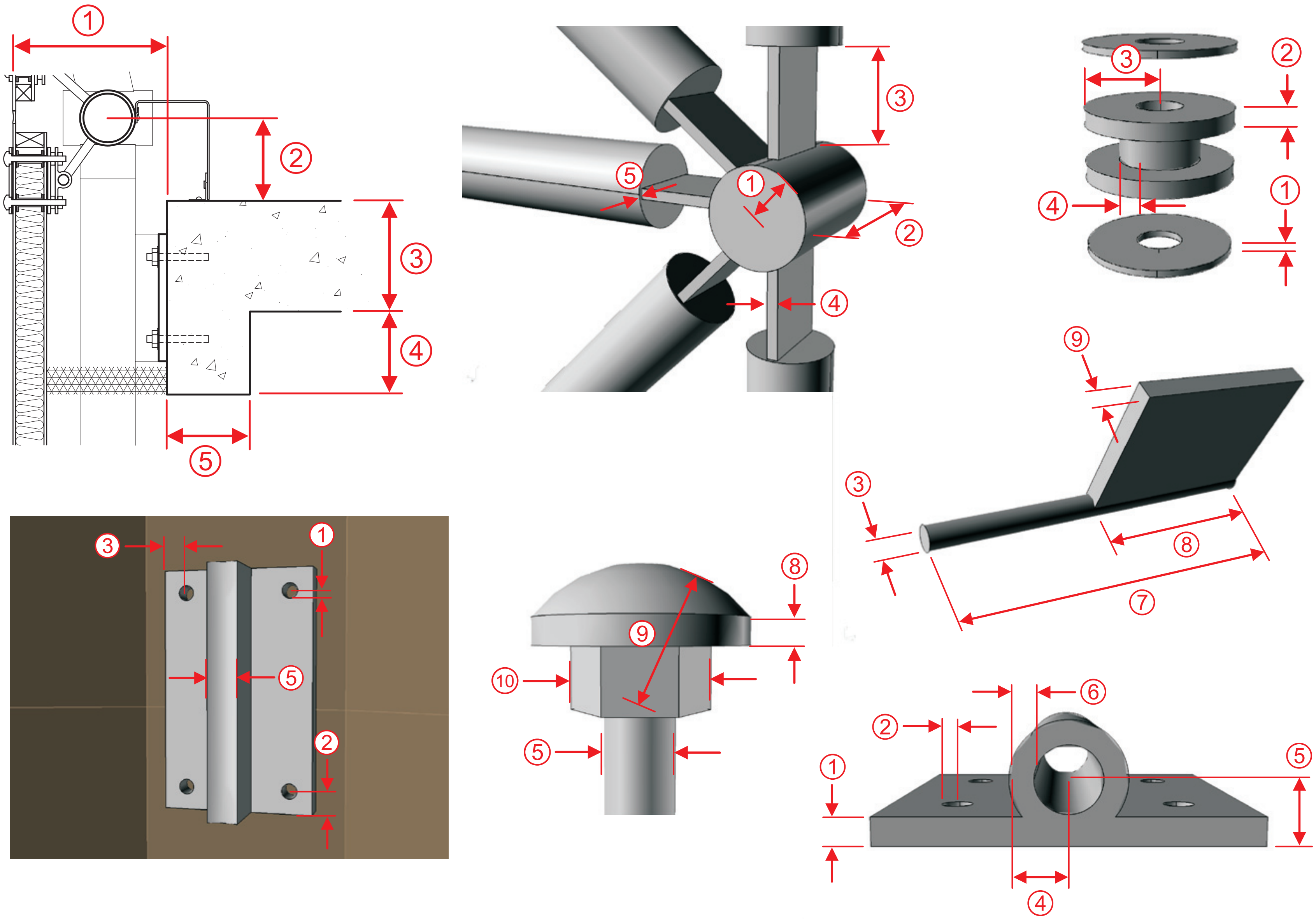


Figure 3.6-3 *Virtually every dimension in the model is variable. For a full description of each parameter, see Appendix A.*

Full Façade View On / Off

In most cases, only a portion of the building needs to be evaluated at a given time, and turning this switch off can save a very substantial amount of computer processing time when changes are made to the model by disabling unneeded geometry.

Partial Façade View Start and End Point

When "Full Façade View" is off, these points determine the beginning and end of the area to be evaluated. The vertex numbers corresponding to the first row of façade input points from Rhino are used as reference when designating the section of the façade to be evaluated. These point numbers can be made visible with the *Façade Input Point Number* toggle in the view parameters.

Double Rows of Triangular Panels per Floor

This toggle determines whether there is a single row or double row of triangular panels per floor. This affects the ways in which the façade's shape can be manipulated and determines whether or not light shelves can be added.

Panel Types

Panel types are changed via an OpenOffice spreadsheet linked to the Grasshopper model. The locations of panels, empty bays and light shelves are all controlled via the same OpenOffice spreadsheet. Detailed information on how to use the spreadsheet can be found in section *4.3: Creating a Façade Model.*

Empty Bays

Panel bays can be designated as empty so that they can be used for other purposes such as doorways. Geometry that is not needed, such as crossing truss members and slab curbs, is edited out automatically when a bay is left empty.

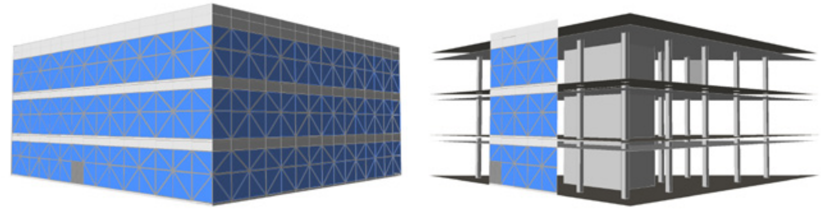


Figure 3.6-4 *Turning the "Full Façade View" parameter off allows specific sections of the model to be viewed and manipulated.*

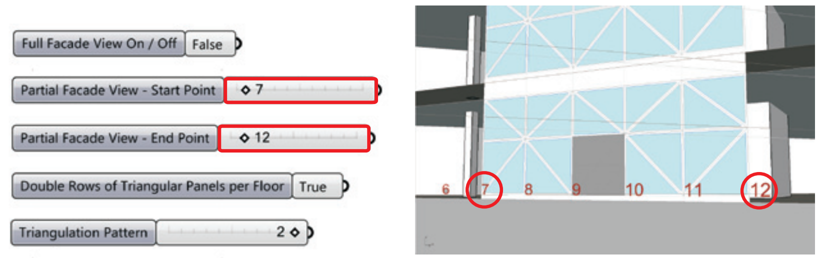


Figure 3.6-5 *Partial Façade View uses the first row of input points as reference.*

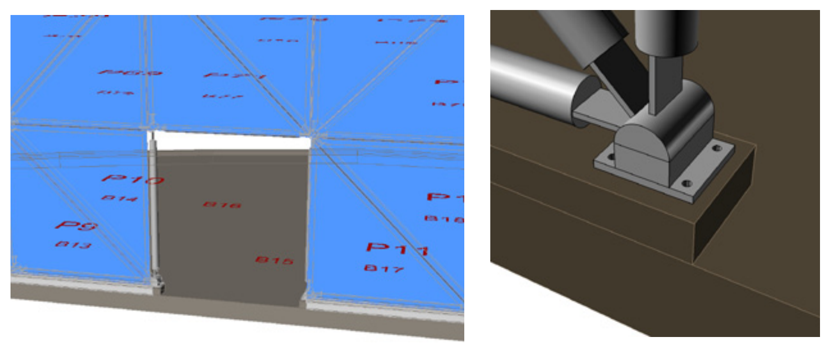


Figure 3.6-6 *Empty bays are used for things such as entrances, which must be designed separately by the user.*

Triangulation Pattern

This parameter allows the user to choose between two triangulation patterns for the panels. The first pattern is an asymmetrical sawtooth-type pattern which is well-suited for free-form shapes. The second pattern is a doubly-symmetrical pattern that is suitable for certain types of regularized façade shapes.

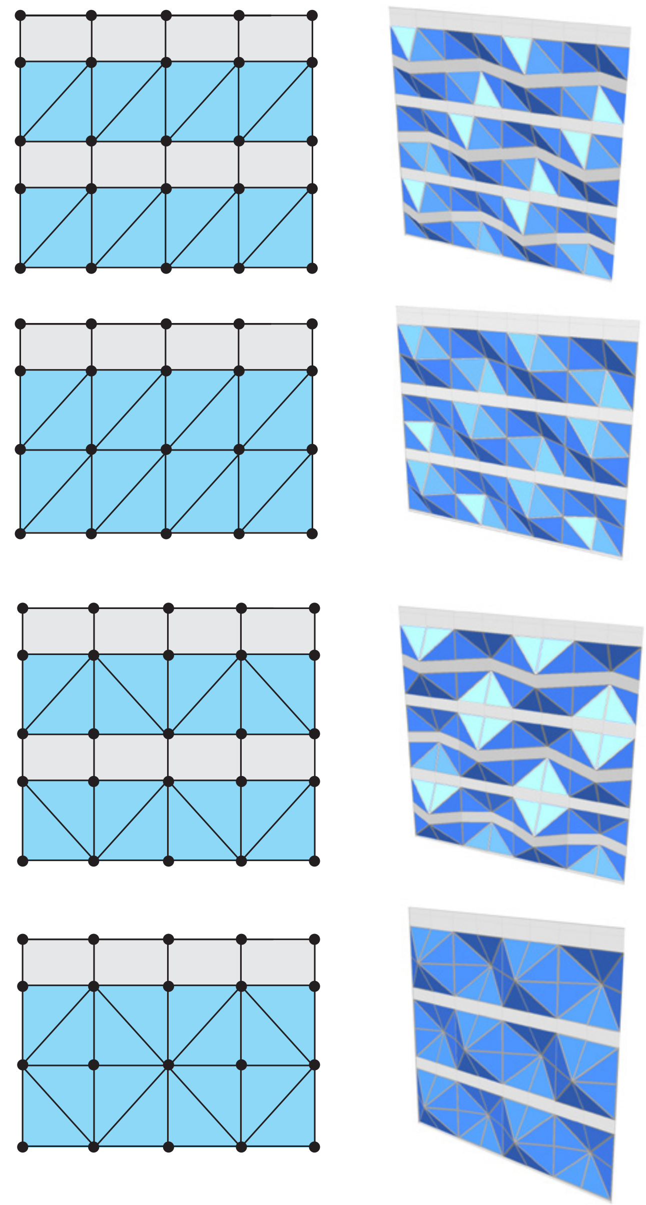


Figure 3.6-7 *Single and double row asymmetrical and symmetrical panel patterns.*

3.7 Analysis Tools and Modes

The model has six analysis modes that can be accessed using the *Evaluation Mode* slider. There are also three additional analysis tools that can be used in conjunction with other evaluation modes.

A number of the analysis modes, such as the Panel Orientation Evaluation Mode, are designed to give quick "rule-of-thumb" feedback on decisions in the early stages of design. More sophisticated tools are also available, such as the Radiance daylight simulation engine accessed through Daylight Evaluation Mode. These tools are more informative but also take a lot of time to give results, so they are best used to perform periodic calculations, or used to make refinements in the latter stages of design.

The analysis modes use simplified geometry to speed up regeneration times. Vector-based analysis modes, such as the View Evaluation Mode, use symbols instead of colors to identify the panel types because colors are used to convey analysis results. The symbols for the panels in vector-based modes are as follows:

1) Empty bay - inverted triangle
2) Opaque panel or spandrel panel - blank
3) Solar panel - circle
4) Translucent panel - outline of a window frame
5) Glazed panel - empty window frame

A filtering mechanism allows the user to turn on only specific types of panel types for an analysis if they choose. Disabled panels appear as gray in the viewport.

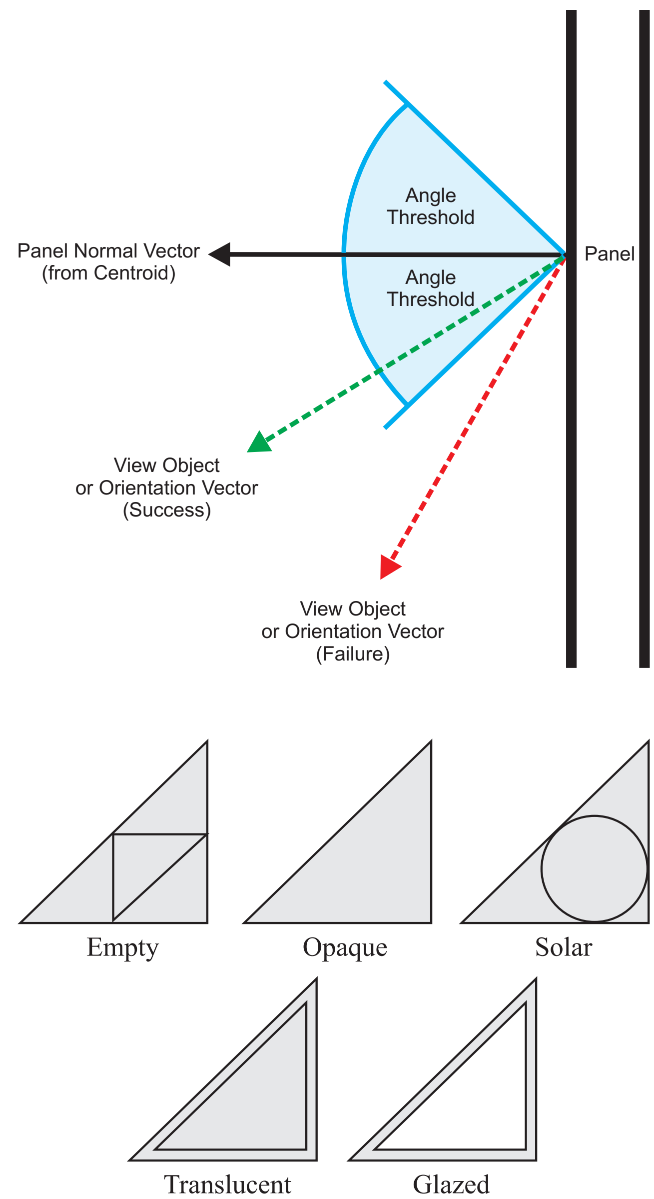


Figure 3.7-1 *Vector-based analysis (top) and panel symbols (bottom).*

View Evaluation Mode

This mode compares a vector perpendicular to each panel's centroid to a series of user-defined points which can be positioned anywhere in the Rhino viewport to represent views. If the angle between the panel's normal vector and the view object exceeds a threshold set by the user, the panel turns red, signifying that there is no view or the view is poor. If the angle is within the threshold, the panel will turn a different color depending on how large the angle is. If the angle is close to zero, the panel will turn white or yellow to indicate a very good view. If the angle is closer to the threshold angle, the panel will turn blue to indicate a marginal view. Shades of green indicate angles in between these extremes.

If the user wishes to see the view vector for an individual panel or check for obstructions between a panel and a view object, the panel can be selected using its index number and a view vector for the panel can be displayed as a line in the Rhino viewport.

The user also has the option to automatically disable all non-glazed panels when performing this analysis.

Vertical Angle Evaluation Mode

This mode compares a panel's normal vector to a vertical vector to determine the panel's deviation from an upright position. If the panel's deviation from vertical exceeds a threshold set by the user, the panel turns red, otherwise the panel displays as green. This mode can be used to restrict panel angles so that snow won't accumulate on the façade, to prevent excessive buckling strain on the truss, or to prevent panel angles that would create awkward or unusable interior spaces.

Solar Panel Angle Evaluation Mode

This mode compares the angle of a panel's normal vector to two different reference angles. The first angle is the ideal vertical angle for a solar panel for a particular location (designated by the user). The second angle is the direction of due South. If the angle between the panel's normal vector and both of these reference angles is within tolerances set by the user, then the panel turns a color ranging from yellow (good) to blue (marginal) depending on how close the vertical angle is to the optimum vertical angle value. If the angle does not fall within both of the thresholds, the panel turns red. Note

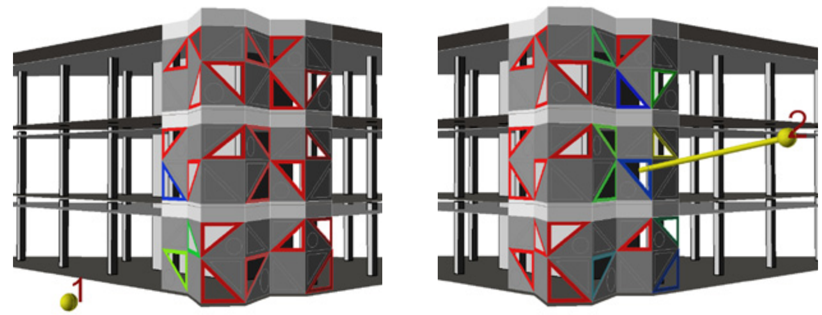


Figure 3.7-2 *Different view targets can be selected for View Evaluation Mode, and sight lines to specific panels can be checked by displaying a vector line.*

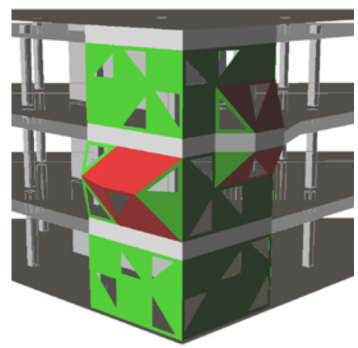


Figure 3.7-3 *Vertical Angle Evaluation Mode is used to spot potential issues like snow accumulation or uninhabitable spaces.*

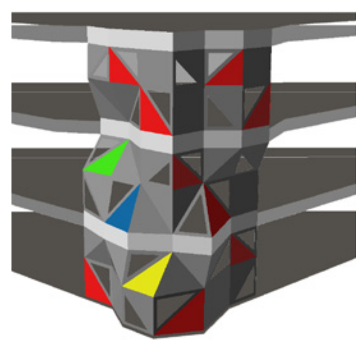


Figure 3.7-4 *Solar Panel Angle Evaluation Mode allows the user to quickly see which panels will get optimal solar exposure.*

that the *North Angle Correction* parameter can be used to change South to North for buildings in the Southern Hemisphere. Non-solar panels can be also be disabled automatically for this analysis.

<u>Southern Exposure Evaluation Mode</u>

This mode simply compares the panel's normal vector to a vector representing due South. If the resulting angle is within the tolerance set by the user the panel turns green, otherwise it turns red. This tool can be used to maximize the amount of panels receiving sun in the winter months at high latitudes. If the building is in the Southern Hemisphere, the south direction can be inverted using the *North Angle Correction* parameter.

<u>Panel Orientation Evaluation Mode</u>

This mode changes the panel's color according to it's orientation relative to the cardinal directions. The panel will turn y ellow if pointing due South, blue if pointing due North, and red if pointing either due East or due West. Angles between the cardinal directions are represented by gradients between the primary colors. For example, a panel pointing South-West would be a shade of orange, while a panel pointing north-east would be a shade of purple. This mode can be used to get a quick sense of the type of sun exposure panels will get. The *North Angle Correction* parameter can be used to invert South and North for buildings in the Southern Hemisphere.

<u>Generic Reference Façade</u>

In order to provide a baseline for daylighting performance comparisons, a standard rectilinear façade can be created independently of the main model and used for performing daylighting simulations. The generic façade can be generated from either the first row of input points used to generate the façade or from its own set of unique input points. The sizes and positions of windows on the generic façade can be adjusted as well as its material composition. Some key statistics are also available for generic façade including window-to-wall ratio and average R-value.

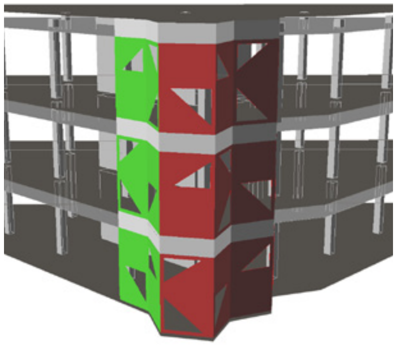


Figure 3.7-5 *Southern Exposure Evaluation Mode allows the user to quickly see which panels will get good light during the winter months.*

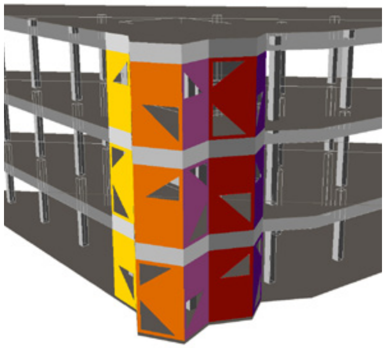


Figure 3.7-6 *Panel Orientation Evaluation Mode color codes panels based on their North, South, East and West orientations.*

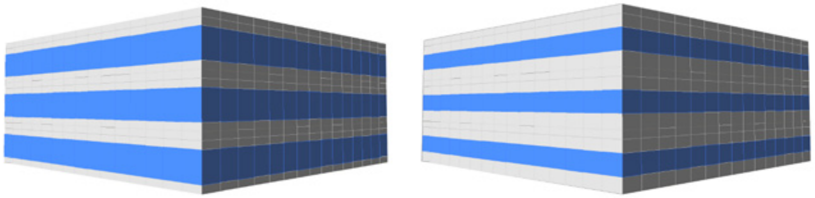


Figure 3.7-7 *An generic façade can be created independently to use as a point of reference for daylighting and energy performance.*

Daylight Evaluation Mode

This mode provides an interface between Grasshopper and a Radiance/Daysim pluggin for Rhino called Diva. This mode can be used to fairly accurately evaluate the building's daylighting performance using metrics such as Daylight Factor or Daylight Autonomy. The mode can also be used to perform glare analyses, or to generate heat maps that show the intensity of solar radiation on various parts of the façade and on individual solar panels.

To speed up analysis, this mode allows the user to isolate individual floors of the building as well as isolate specific parts of each floor using Partial Façade View mode and a set of a moveable partition wall points.

Several of the daylight evaluation routines use light receptacles or "nodes" to determine the amount of light incident on a particular spot in the model. These nodes can be generated either in a plane at a specified height above the floor (e.g. to represent a work plane in an office), or can be placed on individual panels on the exterior in order to optimize the placement and orientation of panels (e.g. optimize placement of solar panels).

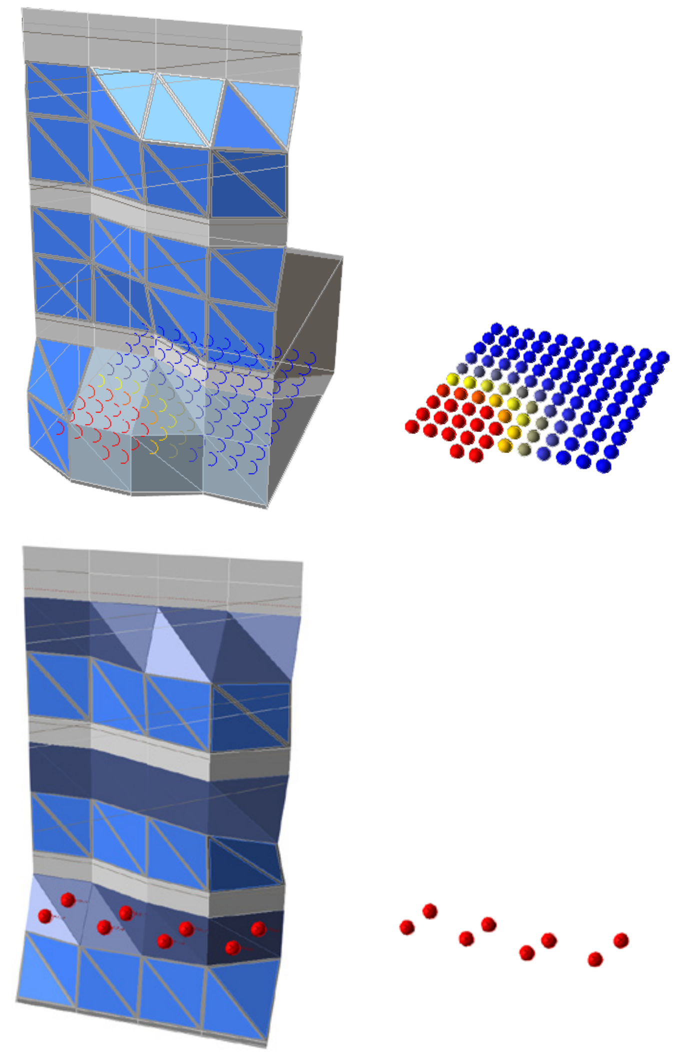


Figure 3.7-8 *Daylight Evaluation Mode configures the geometry to be used in Radiance simulations. Receptor nodes can be configured for interior spaces (top) or for the surfaces of panels (bottom).*

Panel Clip Inaccessibility Warning Tool

This tool allows the user to monitor whether a panel clip's bolts can be accessed directly or not during construction by determining whether a truss member will be obstructing the path to the bolts. The warning mechanism has two different modes, one in which conflicts are displayed as boxes projecting from the panel clips, and one that shows exactly where the truss is blocking access to clips. This analysis tool only works when the *All Detailed Geometry Enabled* mode is selected.

Panel Joint Angle Warning Tool

Overly acute angles between adjacent panels can cause problems including excessive structural loads, difficulties with assembly, snow accumulation, and so forth. This analysis tool allows the user to set minimum angles for both horizontal and vertical joints. If the angle between two adjacent panels exceeds the threshold, a warning indicator will appear on the panel joint in the form of a colored pipe object. Since this warning uses a similar mechanism as the *Panel Edge Length Warning* tool, it is best not to use the two warning modes simultaneously. This analysis tool can work in conjunction with all other evaluation modes.

Panel Edge Length Warning Tool

This tool warns the user when a panel length exceeds a certain threshold by creating a colored pipe object along the edge of the panel. This allows the user to see if dimensions of the panel exceed transportation or fabrication tolerances (set by the user). This analysis tool works across all evaluation modes.

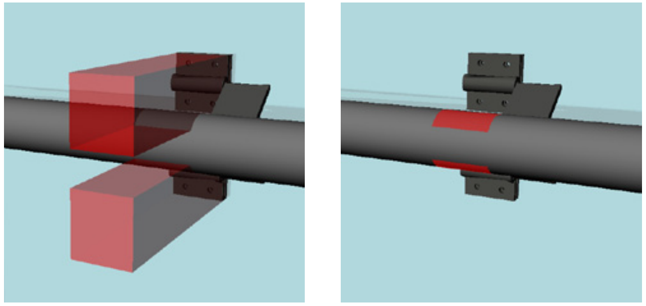


Figure 3.7-9 *The Panel Clip Inaccessibility Warning tool includes a general warning mode (left) and a more precise interference indicator (right).*

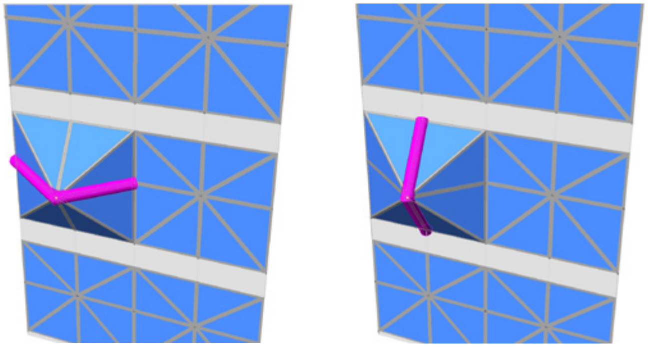


Figure 3.7-10 *The Panel Joint Angle Warning Tool alerts the user to overly-acute panel angles that may pose construction or occupational problems.*

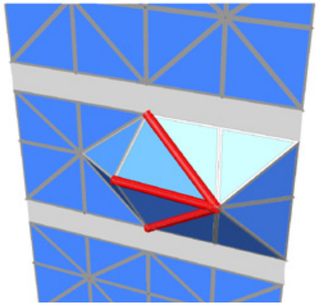


Figure 3.7-11 *The Panel Edge Length Warning Tool alerts the user when the panel size exceeds fabrication or transportation constraints.*

3.8 Model Statistics

The model dynamically tracks several statistics to assist with design and analysis. The following is a general description of the available statistics by category. A comprehensive list of the statistics available can be found in *Appendix A: Model Inputs, Parameters and Statistics.*

1. Number of Panels

These statistics track the total number of panels, the number of each type of type of panel, and the number of panels currently being viewed if *Full Façade View* is turned off.

2. Datum Points and Areas

These statistics track key datum points, dimensions and areas within the model including: top of parapet height, tops of slab heights, slab-to-slab heights, ceiling heights, panel row heights, gross floor areas, total gross floor area, roof area, and ceiling cavity depths.

3. Window to Wall Ratios and Thermal Resistances

These statistics track the window-to-wall ratio and approximate R-value of the façade by floor. The values are also tracked for the *Generic Reference Façade*.

4. Vector-Based Analysis Statistics

These statistics track the number and percentage of panels that currently fall within the designated angle threshold when using vector-based evaluation modes.

5. Daylighting Statistics

When performing Daylight Factor or Daylight Autonomy calculations, the user can set minimum daylighting requirements for the area under evaluation. The number and percentage of analysis nodes that meet the minimum requirements is displayed, as well as the average value of all nodes.

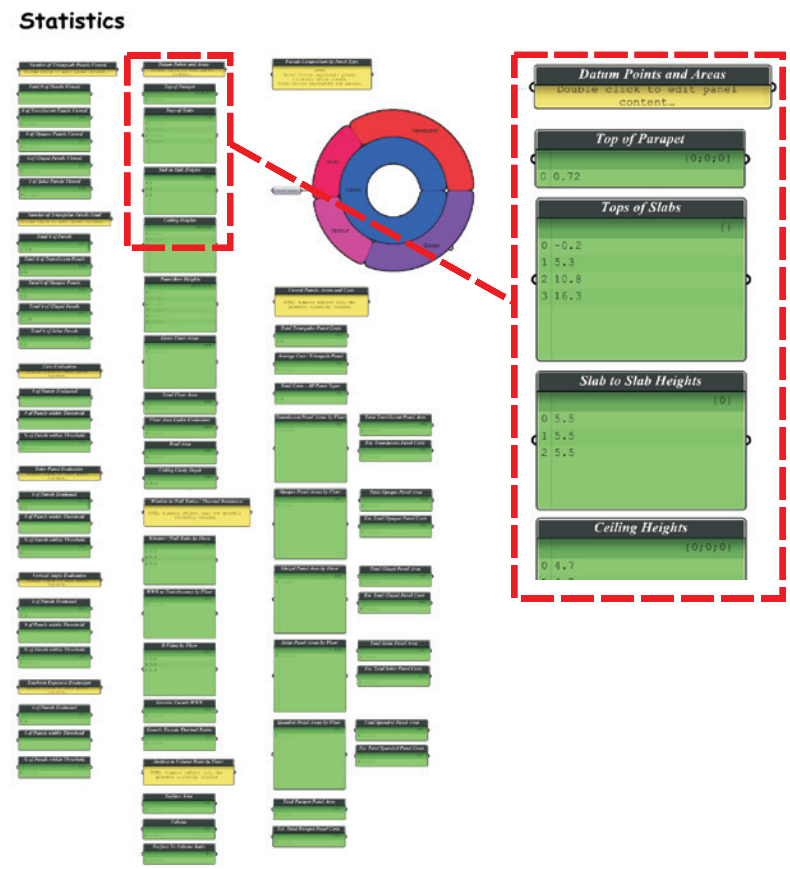


Figure 3.8-1 *A wide variety of model statistics are generated automatically and displayed in the upper left area of the Grasshopper canvas.*

6. Surface and Volume

These statistic track the surface area, volume, and surface-to-volume ratios of the entire building or for a particular portion of the building defined by the partial façade view parameters and the interior partition.

7. Façade Composition by Panel Type Diagram

This pie chart gives a quick visual indication of the relative proportions of each panel type used in the façade for either the current view or the entire building. The inner circle represents panels currently being viewed (i.e. using a partial façade view) and the outer circle represents the total façade.

8. Paneling Cost Factor

These statistics represent calculations of the panelling costs based on cost-per-square meter values entered by the user for each panel type .

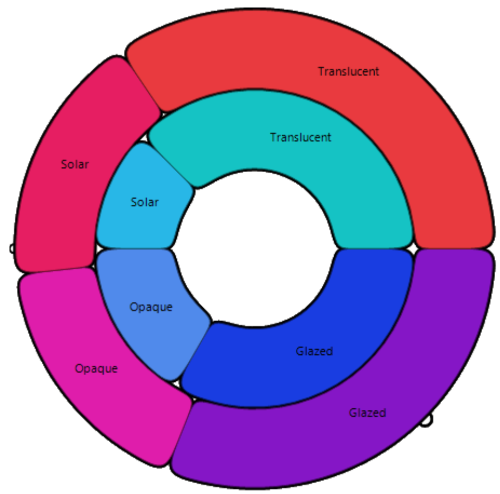


Figure 3.8-2 *A pie chart provides a quick visual indication of the relative proportions of different panel types used.*

3.9 Model Utilities

Statistics Report

When developing a design it can be very helpful to be able to take snapshots of model statistics so as to gauge progress from one design iteration to the next. For this reason, there is an option to export key statistics from the model to an OpenOffice spreadsheet. The statistical information is then automatically labeled and formatted in a manner suitable for printing or archiving.

Grasshopper-to-Rhino Geometry Conversion

In order to create quality renderings, edit geometry directly, or export geometry into another 3D program for analysis, it is necessary to first convert ("bake") Grasshopper geometry to Rhino geometry. This process has been facilitated in the model by routing all key geometry into a single area on the Grasshopper canvas.

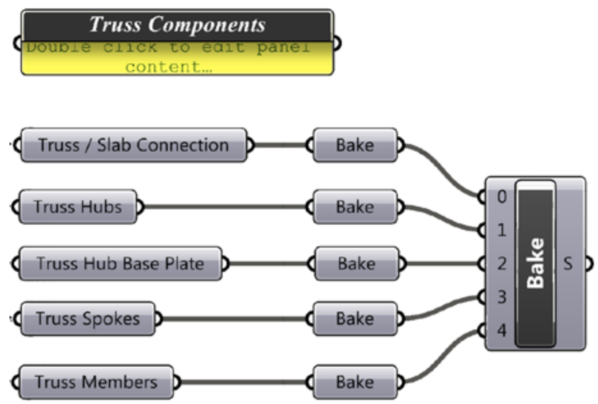


Figure 3.9-1 *All model geometry is consolidated in one area for easy "baking".*

Building Statistics – General	
Full Facade View (1) or Partial Facade View (0)	1
Partial Facade - Start Point	0
Partial Facade - End Point	33
Building Height	11.23
Number of Floors	2
Average Floor Height	5
Partitioned Floor Area	0
Total Floor Area	3232
Building Surface Area	3396
Building Volume	16481
Surface to Volume Ratio	4.85
Window to Wall Ratio	0.82
Window to Wall Ratio w/ Translucency	0.82
Translucency Factor	0.2

Panel Statistics – General	
Number of Translucent Panels	0
Number of Opaque Panels	0
Number of Glazed Panels	526
Number of Solar Panels	0
Total Number of Panels	526

Thermal Resistance Statistics	
Translucent Panel R-Value	3
Opaque Panel R-Value	5
Glazed Panel R-Value	2
Solar Panel R-Value	5
Spandrel Panel R- Value	5
Average R-Value of Building Envelope	2.55

Panel Costing Statistics	
Translucent Panel Cost / Sq.m	100
Opaque Panel Cost / Sq.m	100
Glazed Panel Cost / Sq.m	100
Solar Panel Cost / Sq.m	100
Spandrel Panel Cost / Sq.m	100
Total Translucent Panel Cost	0
Total Opaque Panel Cost	0
Total Glazed Panel Cost	72961
Total Solar Panel Cost	0
Total Spandrel Panel Cost	16587
Total Cost of Trangular Panels	72961
Average Cost per Triangular Panel	276.37
Total Panel Costs	97106

Analysis Statistics – Views	
View Angle Threshold	45
View Object Evaluated	2
Number of Panels Evaluated	264
Number of Panels Within Threshold	95
Percentage of Panels Within Threshold	35.98

Analysis Statistics – Solar Panels	
Optimum Vertical Angle	45
Optimum Vertical Angle Threshold	40
South Orientation Threshold	45
Number of Panels Evaluated	1
Number of Panels Within Thresholds	0
Percentage of Panels Within Threshold	0

Analysis Statistics – Panel Vertical Angles	
Vertical Angle Threshold	25
Number of Panels Evaluated	1
Number of Panels Within Threshold	0
Percentage of Panels Within Threshold	0

Analysis Statistics – Southern Orientation	
Southern Orientation Threshold	30
Number of Panels Evaluated	1
Number of Panels Within Threshold	0
Percentage of Panels Within Threshold	0

Analysis Statistics – Daylighting	
Floor Under Evaluation	2
Number of Light Shelves	0
Window Sill Height	0.2
Window Aperture Height	4
Window Tops (from slabs)	4.2
Daylight Autonomy Threshold (lux)	33
Daylight Factor Threshold (%)	2
Number of Evaluation Nodes	121
Number of Nodes within Threshold	67
Percentage of Nodes within Threshold	55.37
Average Node Value	35.41

Figure 3.9-2 *A statistics report in OpenOffice spreadsheet format generated automatically by Grasshopper.*

4. Using the Façade Model

4.1 Façade System Configurations

The façade system is designed to be extremely versatile in terms of the forms it can take. Below are some examples of this versatility.

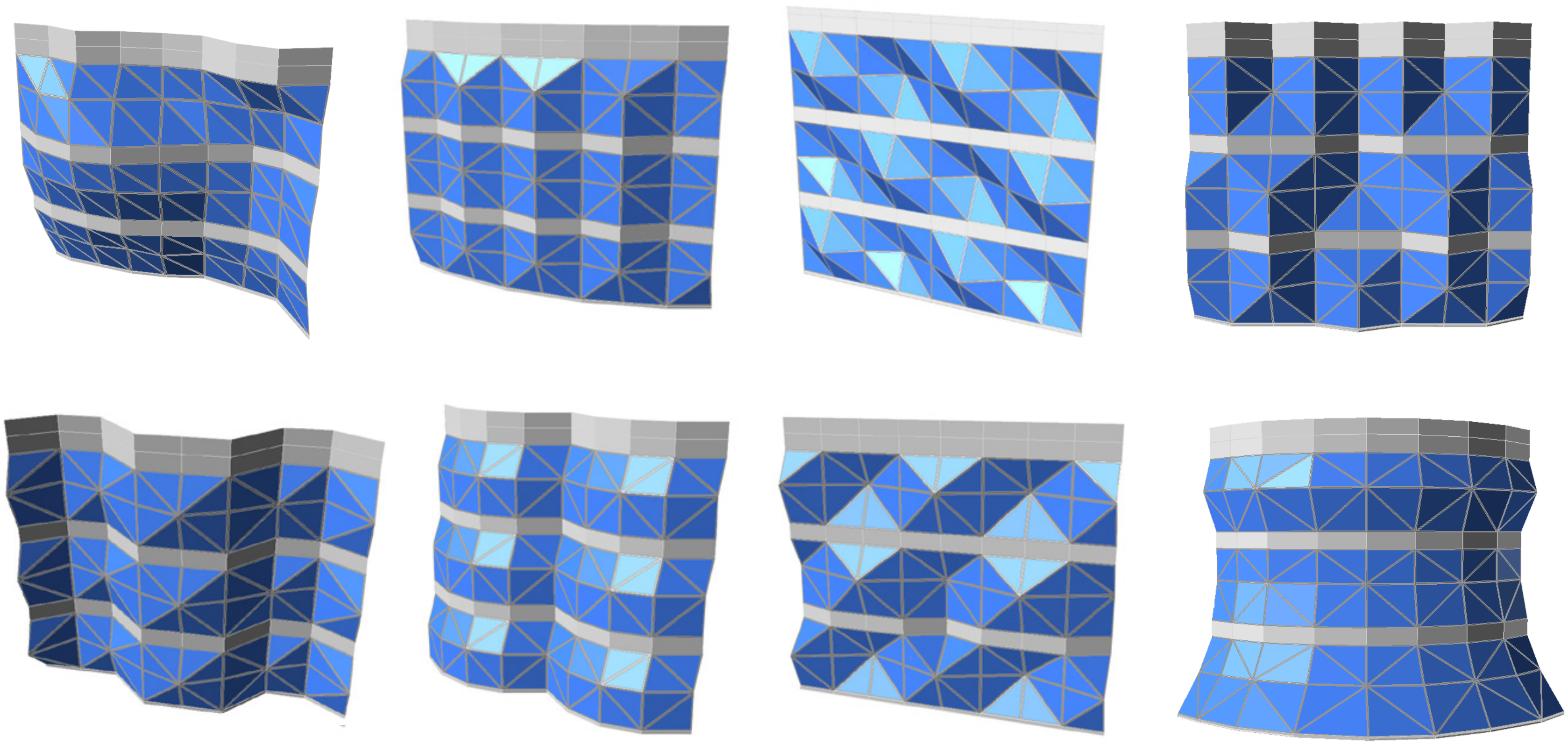


Figure 4.1-1 *Example façade configurations.*

4.2 *Façade System Limitations and Constraints*

While the façade system is very versatile, it has some limitations. This section looks at some of the key constraints and limits that the user needs to be aware of when configuring the system.

1) Movement of points

Points in the model within any given point row can be moved freely on the horizontal plane so long as long as they stay within sequence. If points are moved vertically, however, the entire row must be moved or the geometry may not generate properly.

2) Variations in panel size

All clip and hinge parameters are global. This means that wide variations between panel sizes may cause problems when trying to space clips properly around the perimeters of the panels.

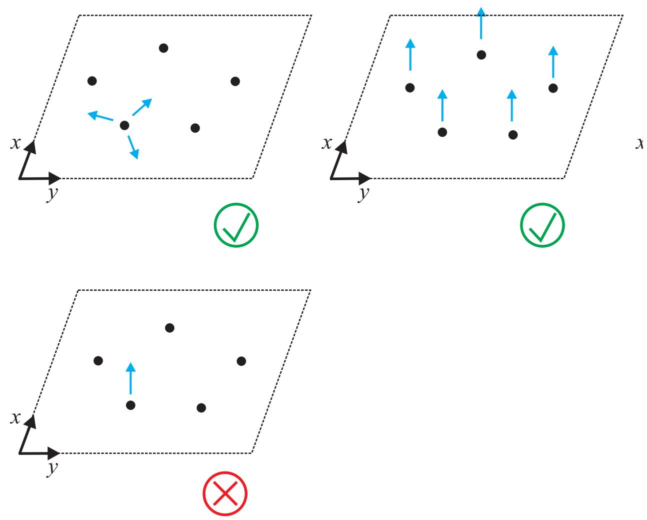


Figure 4.2-1 *In-plane movement of points: input points can be moved freely in the horizontal plane, but if points are to be moved vertically, the entire row must be moved.*

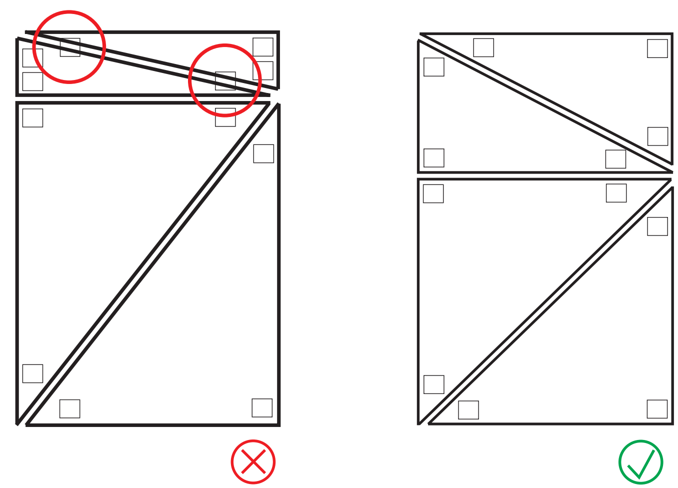


Figure 4.2-2 *Clip parameters cannot be adjusted on a per-panel basis, meaning that large variations in panel size can cause problems when positioning clips.*

3) Rectilinear and curvilinear forms

It is relatively simple to create precise angular or rectilinear forms by moving points precise distances horizontally or vertically in Rhino or using grid snaps. Curvilinear forms, on the other hand, are more challenging to execute with precision because the input points for the model are not attached to lines or splines.

Parallel sets of lines or arcs with evenly-spaced points can be created as guides for aligning the points, but care has to be taken to make points in different rows perpendicular to one another. Dividing parallel arcs is one way to yield properly perpendicular points, but it will only work if all arcs are treated separately (i.e. not attached to one another). Alternatively, perpendicular lines can be created between the guidelines at even intervals in order to generate properly aligned point rows.

4) Clip accessibility during construction

The *Panel Clip Inaccessibility Warning* mechanism will warn the user when the bolts on a clip assembly may be difficult to access during installation. All clips should be accessible with a torque wrench, but if there is enough space between the truss members and the panels it is not necessary that all clips pass the warning test.

The warning tool does not cover spandrel clips at all. Spandrel clip positioning is based on global variables such as the ceiling cavity dimension, and thus it remains constant regardless of the shape of the façade.

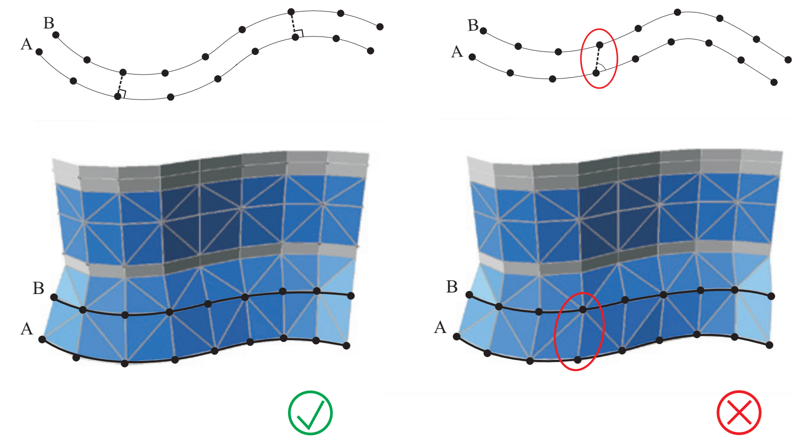


Figure 4.2-3 *Spline-based curves can cause unpredictable results. Concentric arcs are the most reliable way to create curves in the façade.*

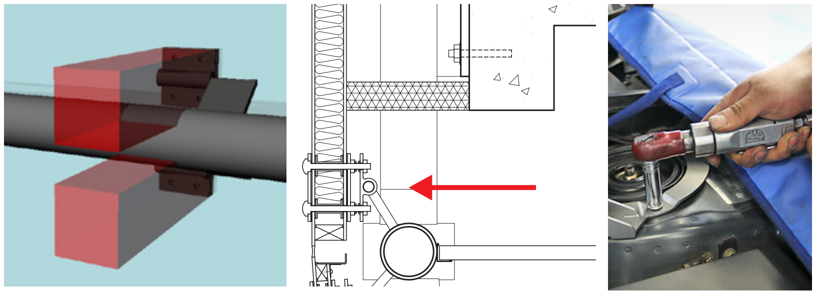


Figure 4.2-4 *The bolts for the panel clips must generally be accessible by torque wrench, including those of the spandrel panels (center).*

5) Panel clip edge offsets

Clips should be offset from the edges and corners of the panels so that they are clear of the spacers on the edges of the panels. Clips should also be offset so that hinges do not overlap with truss spokes.

6) Interior Partitions

If internal walls or partitions are to be used, panel edges and truss members must be made perfectly vertical periodically. These joints can slant inwards, but should be aligned vertically along the axes of the intended partition walls.

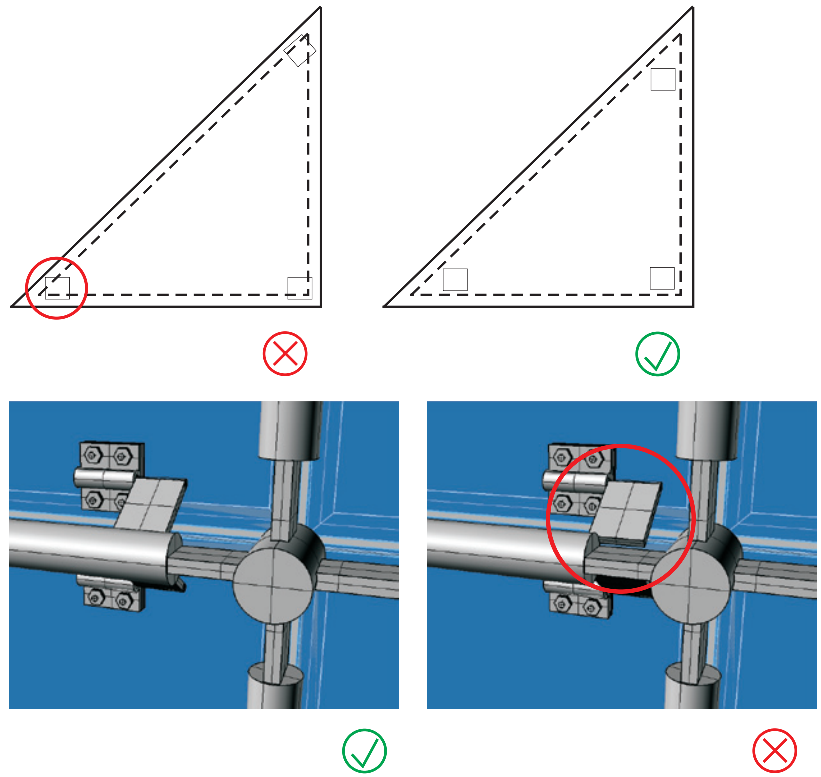


Figure 4.2-5 *Clips need to be positioned so that they do not overlap with the spacers around the edges of the panels and so that they don't overlap with the truss spokes.*

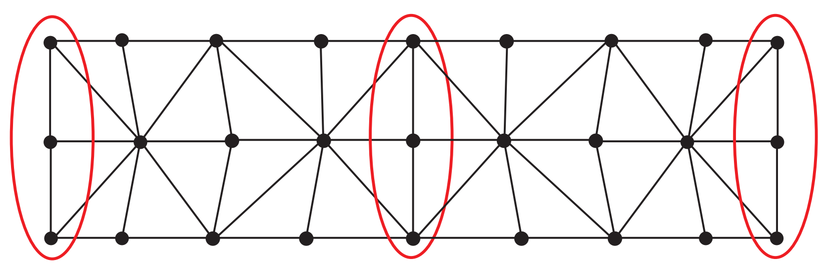


Figure 4.2-6 *If the space is to be partitioned, it is necessary to have vertical seams periodically in the directions of the intended partition walls.*

7) Joint angles

Because the edges of panels are always parallel to neighboring panels, the panel joints and gaskets can accommodate wide ranges of concave and convex angles. Joint angles limited more by other factors, such as truss hub limitations (see "Truss configurations" below) and the relative positions of the clips and the hinge pins. Convex joints are generally less limited than concave joints, and corners of up to 90̊ can easily be achieved for convex vertical joints.

8) Empty bays

Empty bays are intended to be used only on the first row of panels and do not work reliably on upper floors or panel rows. Empty bays can be placed in series', but there will always be vertical truss members in between bays.

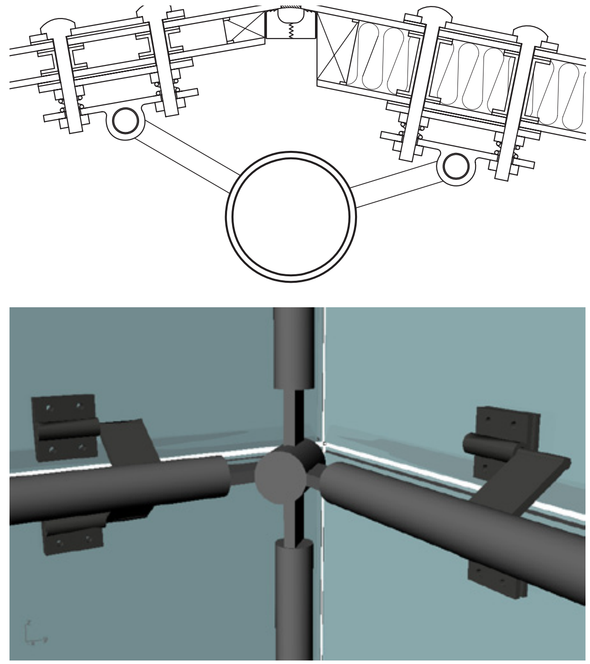


Figure 4.2-7 *The joint topology will accommodate convex corners on vertical joints (bottom).*

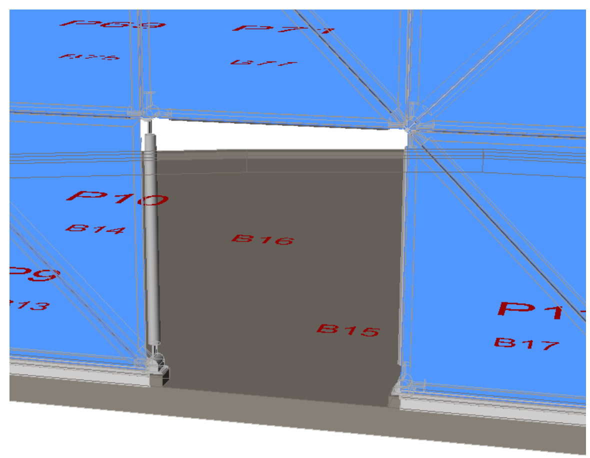


Figure 4.2-8 *Empty bays are restricted to the first floor but can grouped together.*

9) Truss configurations

Hub and truss parameters should be adjusted so that truss spokes always intersect the hubs from on top. Increasing truss hub sizes will allow more acute angles, but it will also require a greater offset distance between the panels and the truss.

Since the truss has limited structural depth, it is not designed to bow-out very far or take excessive eccentric loads. Care should be taken to resolve forces efficiently and to avoid excessive buckling stresses on the joints.

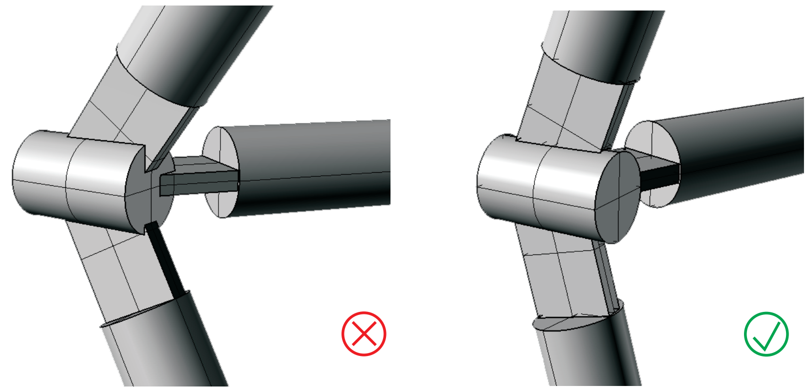


Figure 4.2-9 *Extreme joint angles may cause the truss spokes to intersect with the ends of the hubs.*

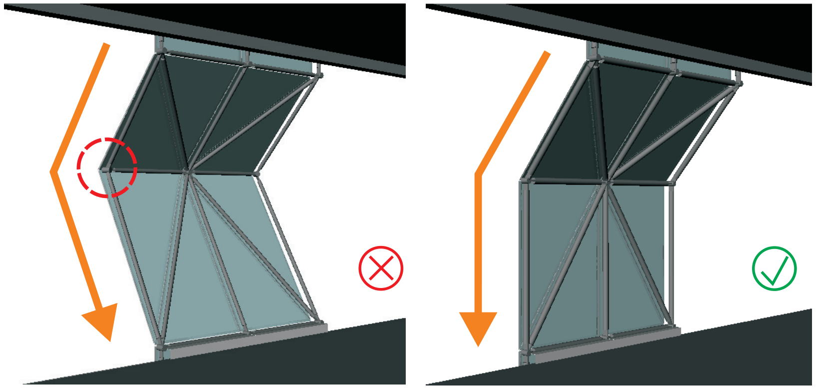


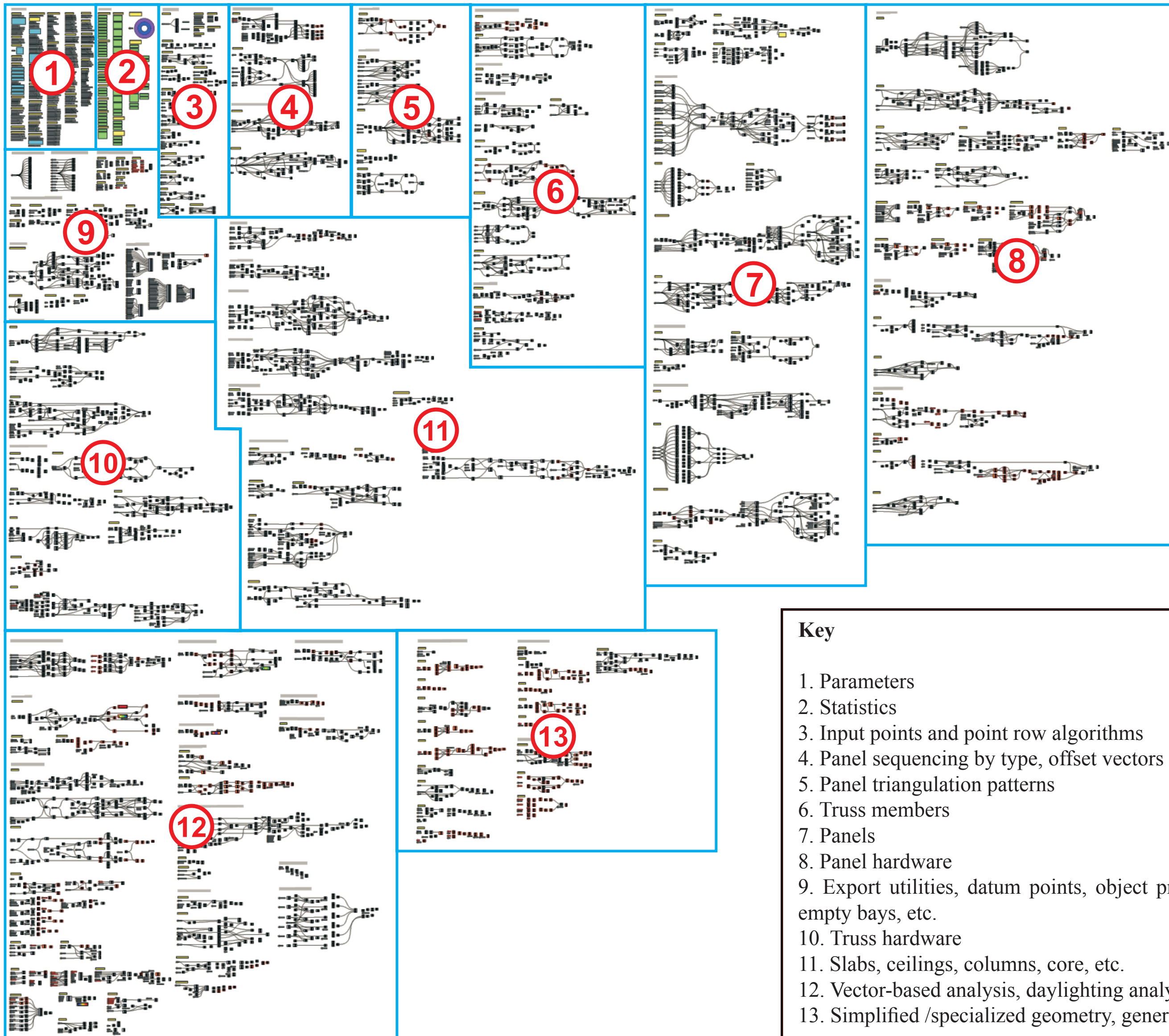
Figure 4.2-10 *Shifting the center of gravity too far outwards from the slabs can cause undue stress in the truss joints.*

4.3 Creating a Façade from Scratch

The model is designed to be usable by anyone with a moderate amount of Rhino 3D experience and little or no Grasshopper experience. This chapter provides a series of simple steps for creating a façade model from scratch.

The two-dimensional workspace in Grasshopper is referred to here as the "canvas", and individual elements in Grasshopper are referred to as "components". See figure 4.3-1 on the next page for an overview of the Grasshopper canvas and the locations of various elements on the canvas.

Note: The façade model requires the following software to run: Rhinoceros 3D (version 4), Grasshopper 3D (build 0.8.0051), and OpenOffice (version 3.2). Other versions of these software packages may not work as expected. DIVA, the Radiance plugin for Rhino and Grasshopper, is also needed if any daylighting analyses are to be performed. DIVA is currently in beta development and the software has a tendency to change significantly from one version to the next, often causing it to stop functioning until it can be properly re-integrated with the Grasshopper model. The model was created using DIVA version 1.2c beta.



Key

1. Parameters
2. Statistics
3. Input points and point row algorithms
4. Panel sequencing by type, offset vectors
5. Panel triangulation patterns
6. Truss members
7. Panels
8. Panel hardware
9. Export utilities, datum points, object properties, view management, empty bays, etc.
10. Truss hardware
11. Slabs, ceilings, columns, core, etc.
12. Vector-based analysis, daylighting analysis, statistics
13. Simplified /specialized geometry, generic reference façade geometry

Figure 4.3-1 *Overview of the canvas for the Grasshopper model. Most components and parameters that need to be adjusted to create a façade are in sections 1-3.*

Step 1: Create Input Points

Create a new Rhino file and then create a number of point objects on the XY plane. The points can be in any configuration except a straight line (because the façade must form a loop).   Copy the points upward to create additional rows of points. If the planned number of rows of triangular panels per floor is one, then the number of point rows needed will be equal to the number of building storeys, plus one. For example, a five-storey building with one row of triangular panels per floor will need six rows of input points. If using two rows of panels per floor, then the number of point rows needed will be equal to twice the number of building storeys, plus one. A five-storey building with two rows of triangular panels per floor would therefore require eleven rows of input points.

Step 2: Create a Grasshopper Point Component

Create a point component in Grasshopper. Right-click on the component and rename it to “Point Row 1”.   Create additional point components for each point row created in step 3 and name them “Point Row 2”, “Point Row 3”, etc., in sequence from the bottom row to the top row.

Step 3: Reference the Rhino Points into Grasshopper

Right-click on the first point component and select *Set Multiple Points*, then select the points in the first row in sequence, making note of the position of the first point. Repeat this step for each point row using different point components. Be sure that the first points of each row align vertically.

Figure 4.3-2 *Step 1: Create a series of points on the XY plane and copy them upwards.*

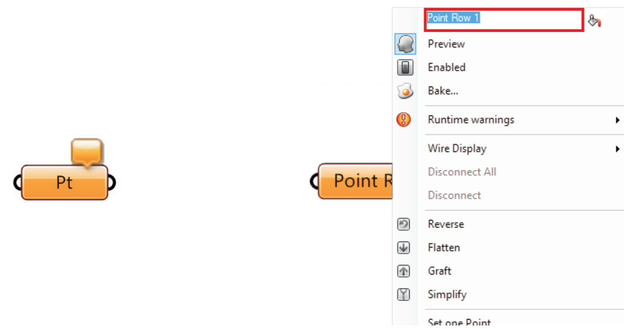


Figure 4.3-3 *Step 2: Create a point component in Grasshopper.*

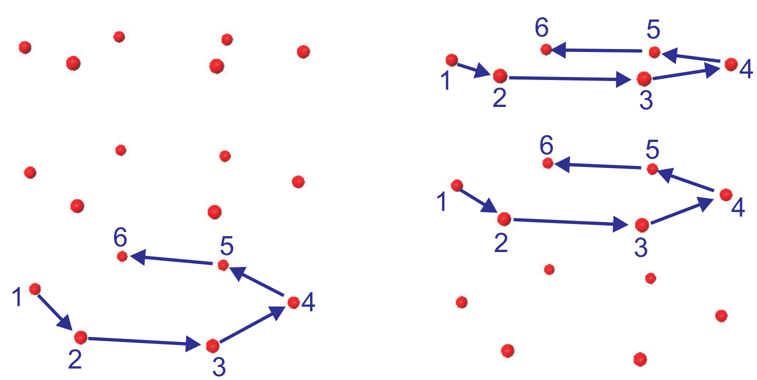


Figure 4.3-4 *Step 3: Reference the Rhino points into Grasshopper*

Step 4: Set Parameters

Change the following values in the *Parameters* section (section 1) of the Grasshopper canvas.

1) Under the *General Parameters* sub-section:

   a) Set the *Evaluation / Modeling Mode* slider to *2: Quick Edit Mode*.

   b) Set the *Full Façade View On / Off* parameter to *True.*

   c) Set the *Double Rows of Triangular Panels per Floor* parameter to *True* if using two rows of panels per floor, or *False* if using one row of panels per floor.

   d) Set the *Triangulation Pattern* parameter to *1* for an asymmetrical panel pattern or *2* for a symmetrical pattern.

2) Under the *Input Point Parameters* sub-section, set the *Number of Input Point Rows* parameter to match the amount of rows of input points created previously.

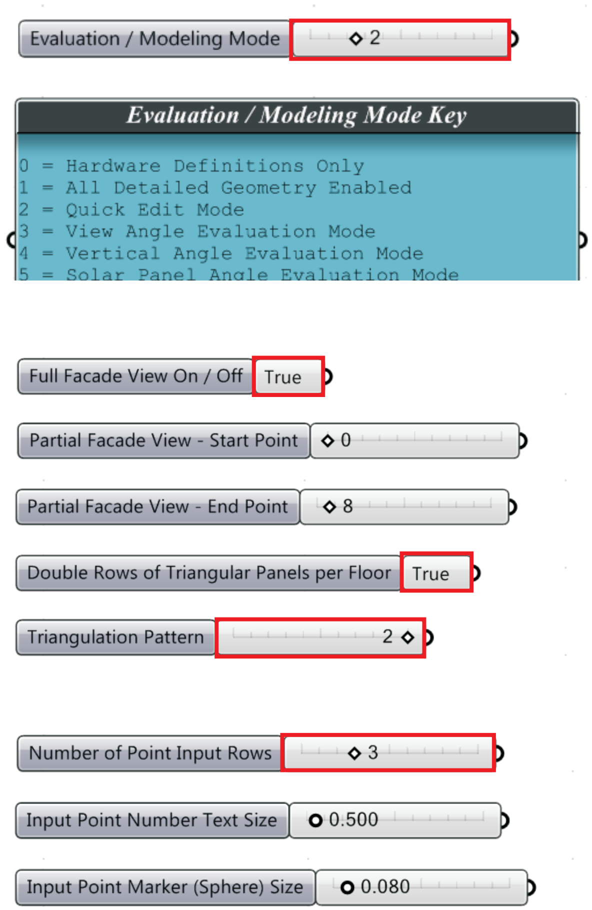


Figure 4.3-5 *Step 4: Set Parameters*

Step 5: Connect Point Row Components

Within the *Input Points and Point Row Algorithms* section of the model (Section 3), there is a sub-section under *Input Points* called *Input Point Rows*. Drag the point row components created in step 2 to this area and connect them to the large vertical *Merge* component. Connect the *Point Row 1* component created previously to input 0 of the *Merge* component, connect the *Point Row 2* component to input 1, and so on.

Next, connect the *Point Row 1* component to the input of the *Connect to Point Row 1* component.

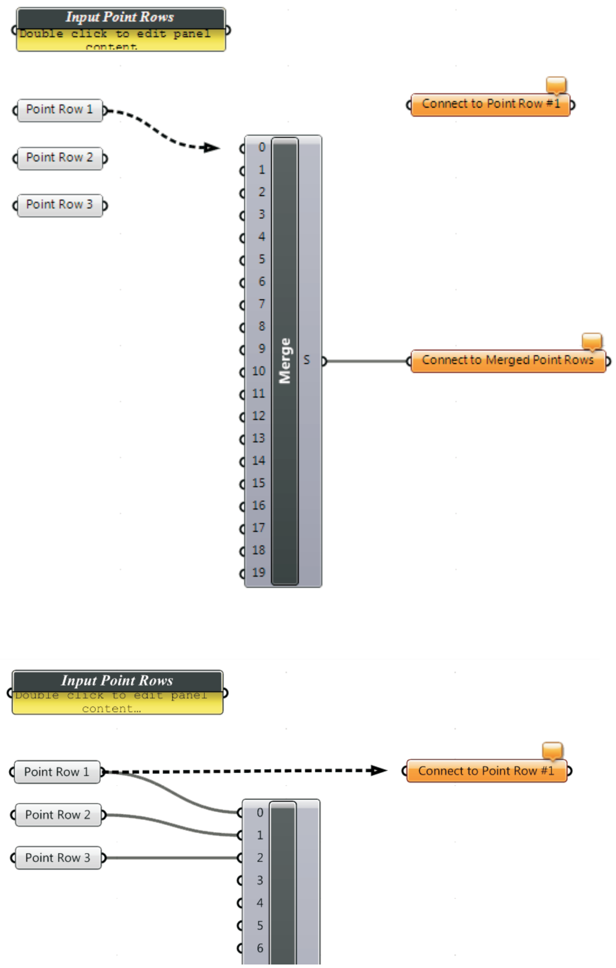


Figure 4.3-6 *Step 5: Connect Point Row Components*

Step 6: Create a Panel-Type Spreadsheet

First, in the *Statistics* section of the model (Section 2), under the sub-section *Number of Triangular Panels*, note the number of bays in the *Total Number of Bays* statistics box.

Next Create an OpenOffice spreadsheet file and save it by any name. The first column on Sheet 1 will represent the panel list. The rows in the spreadsheet represent the panels by number, and amount of rows should equal the amount of panel bays initially. To see the panel numbers in the model, turn *Panel Text Visibility* toggle on under *Component Visibility Parameters*.

To designate panel types, fill in the first column in the spreadsheet with numbers from 1 to 4. The numbers correspond to the different panel types, as follows:

1 = Translucent panel
2 = Opaque Panel
3 = Glazed Panel
4 = Solar Panel

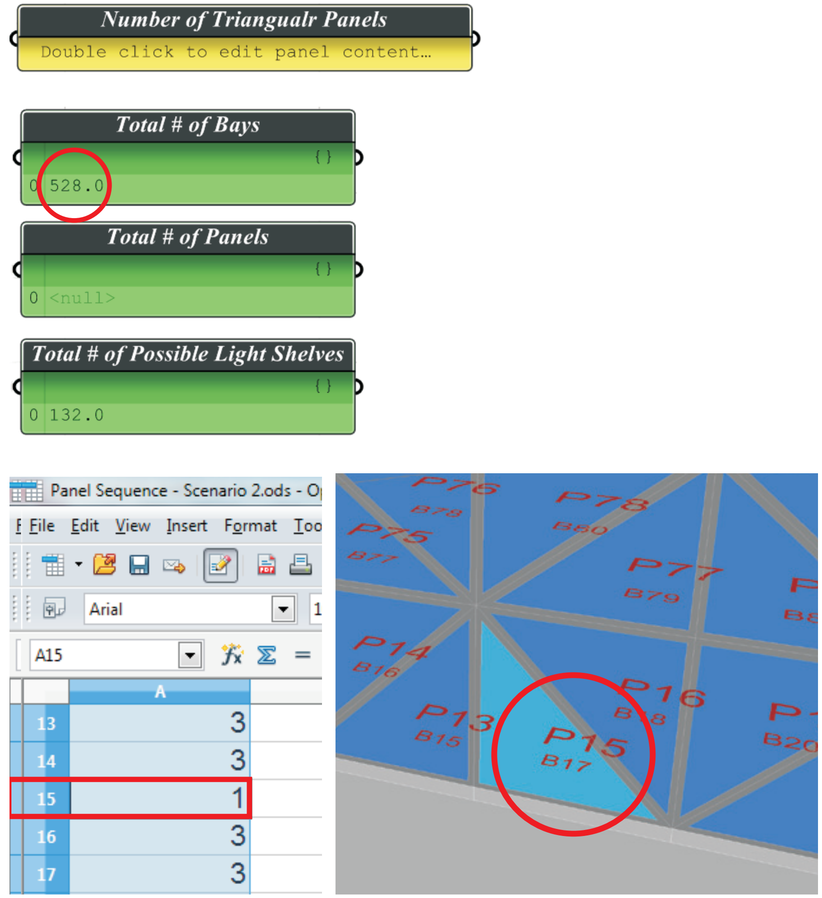


Figure 4.3-7 *Step 6: Create a Panel-Type Spreadsheet. Top: number of panels statistic. Left: the panel list in the spreadsheet. Right: corresponding panel index numbers displayed in the model.*

Step 7: Indicate Empty Bays and Light Shelves

Sheet 2 of the spreadsheet determines which bays will be kept empty so that they can be used for other things such as doorways, while Sheet 3 of the spreadsheet determines which windows will have light shelves. Enter the numbers of the bays that will be empty in numeric order. The row numbers do not represent anything on this sheet. The bays should always be listed in pairs, starting with odd-numbered bays. Note that adding empty bays will reduce the overall panel count, and Sheet 1 will need to be updated accordingly.

Sheet 3 designates which parts of the façade will have light-shelves. First determine the total number of possible light shelves by looking at the *Total # of Possible Light Shelves* statistic in the model. Then, in the first column of Sheet 3 enter the value of *0* for those bays where there will be no light shelf, and a value of *1* for those bays which will have light shelves, making sure that the total number of rows in the sheet matches the total number of possible light shelves. To see which number in the spreadsheet corresponds to which light shelf location in the model, turn on the light shelf number tags in the model under *Component Visibility Parameters*.

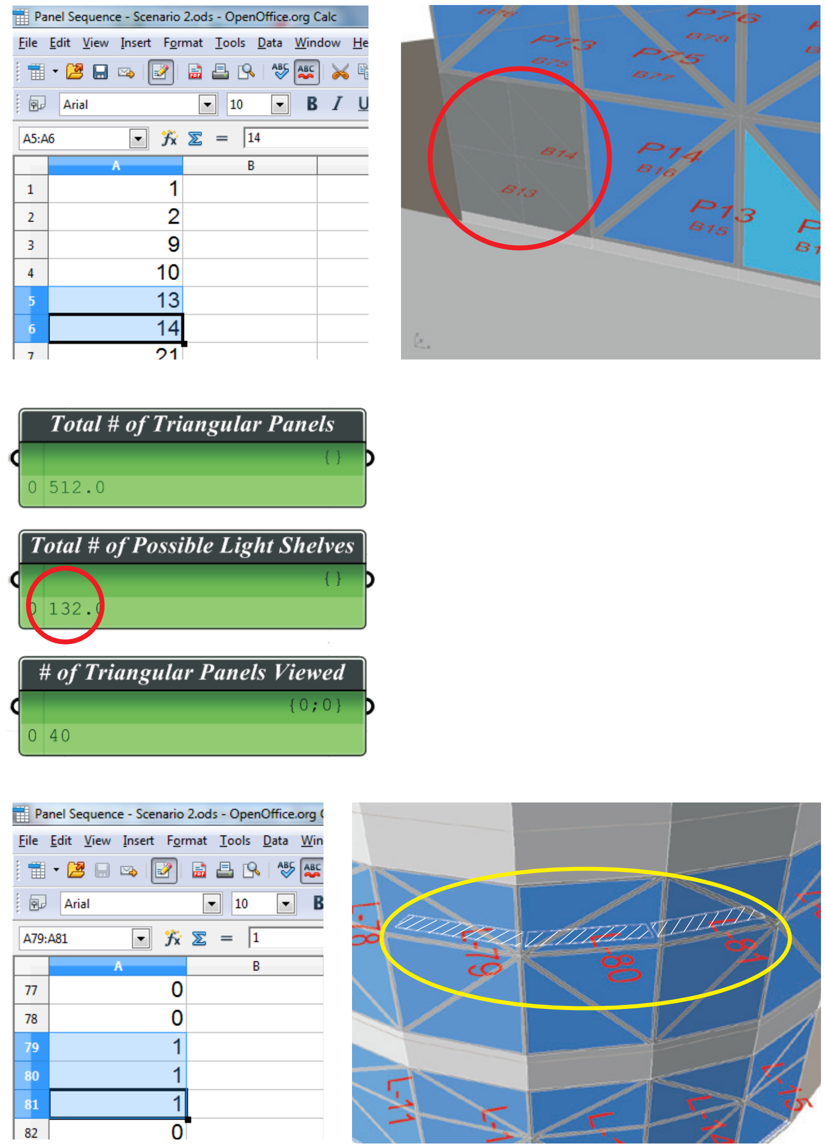


Figure 4.3-8 *Step 7: Add empty bays and light shelves. Top: empty bays controlled on sheet 2. Middle: statistic for total possible number of light shelves. Bottom: light shelf on/off toggle on sheet 3.*

Step 8: Enter Spreadsheet Information into Grasshopper

Open the Grasshopper model, and in the *File Information Parameters* of the canvas, double-click on the *Panel Sequence File #1* box and type the full path and filename of the spreadsheet file (including the .ods file extension). Set the *Panel Sequence File #* slider to *1*.

The panels should become visible at this point. Changes can be made to the panel types at any time by modifying the spreadsheet, then using the *Recompute* command in the *Solution* menu in Grasshopper.

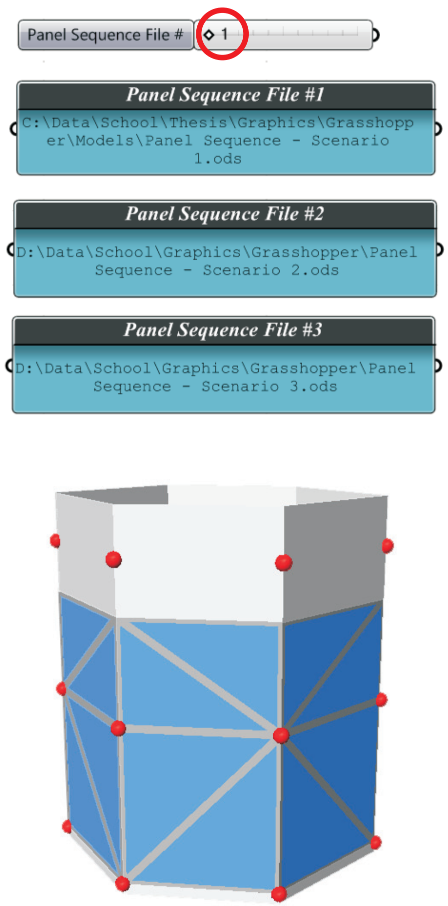


Figure 4.3-9 *Step 8: Designate the name and path of the spreadsheet to apply the panel list to the model.*

Step 9: Create Hardware Input Points

Panel clips and hinges, as well as truss-to-slab attachments, are created independently from the façade geometry and then “populated” onto the façade in the appropriate places, and thus these elements require their own input points.

Rhino points can be referenced into Grasshopper for these components in the same way that the façade input points are. To reference points into Grasshopper, use the pre-made point components in the *Hardware Input Points* sub-section of the *Input Points and Point Row Algorithms* section (section 3) of the Grasshopper canvas.

1) Truss-to-slab Attachment plate

The truss-to-slab attachment plates are generated using four points to define the corners and one point to define both the center of the plate and the location where the plate is to be created. Use the pre-made *Truss/Slab Attachment Input Points* component to select the corner points, and the *Truss/Slab Attachment Plane Origin* component to select the center point. Note that a misalignment of the center point can cause anomalies when the plate is populated onto the model.

2) Panel Hardware

The panel clips and hinges are generated in a similar fashion as the truss-to-slab attachment. Use the pre-made *Panel Hardware Input Points* component to define the corners of the clip and hinge plates (selecting points in the order illustrated in *Figure 4.3-10*), and use the *Panel Hardware Plane Origin* component to define the center point. Note that, as of this writing, the boolean operations used to create the clip components are unreliable. Some experimentation may be necessary to find a combination of parameters that works.

3) Truss Hub Base Plate

The truss hub base plate only requires a center point. Its dimensions are determined solely by numeric parameters.

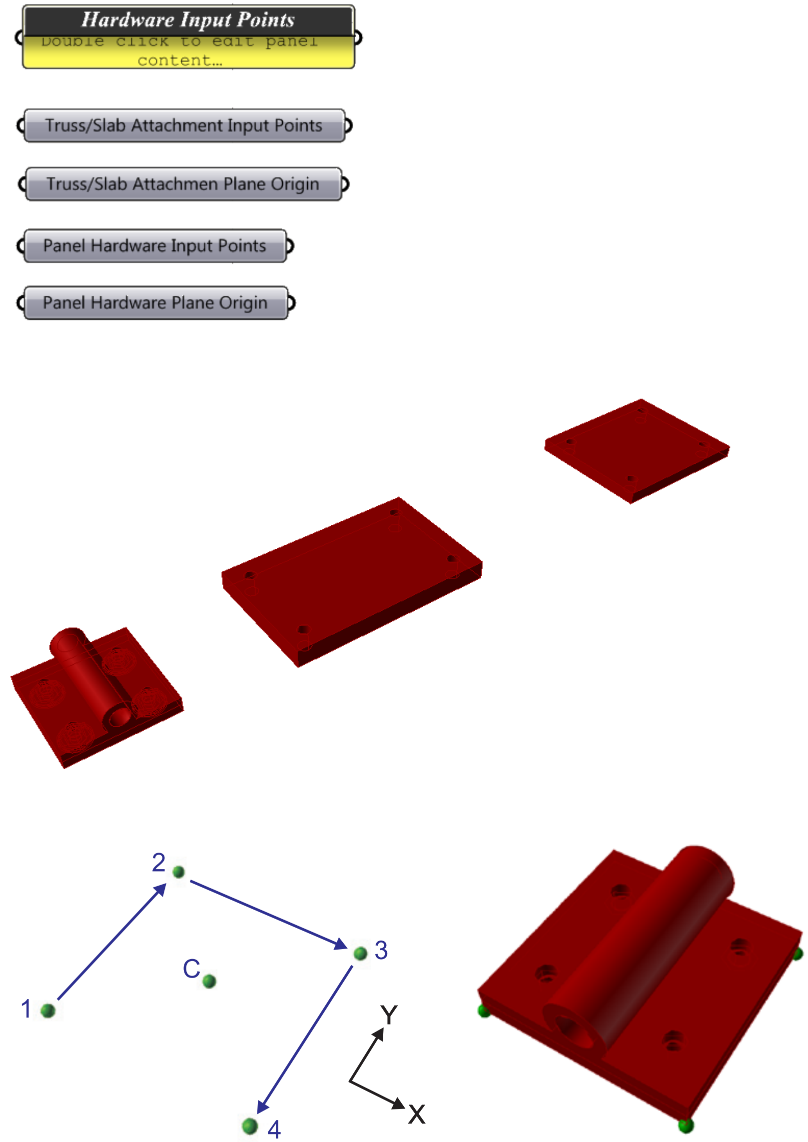


Figure 4.3-10 *Step 9: Create points for panel and truss hardware.*

Step 10: Enable Detailed Geometry

Once the panel hardware has been properly defined, the detailed model geometry can be viewed. Go back to the *Evaluation / Modeling Mode* parameter under *General Parameters* and choose *1: All Detailed Geometry Enabled* to see the complete façade geometry. Adjust the clip positioning and number of clips per panel according to the sizes and shapes of the panels.

For a full list of parameter list and explanations of how to add elements such as partition walls, a building core, columns, view objects, and so forth, see *Appendix A: Inputs, Parameters and Statistics.*

Note that enabling detailed geometry forces the computer to make complex calculations any time some aspect of the model is changed, and this can lead to very long pauses. As such it is advisable to only enable detailed geometry when necessary and to use "Quick Edit Mode" whenever possible.

Also note that, as of this writing, Rhino and Grasshopper will frequently experience memory problems and crash if the detailed geometry is enabled for too many panels simultaneously. It is advisable to disable the bolts and clip spacers when viewing detailed geometry, and to limit the amount of panels viewed whenever possible.

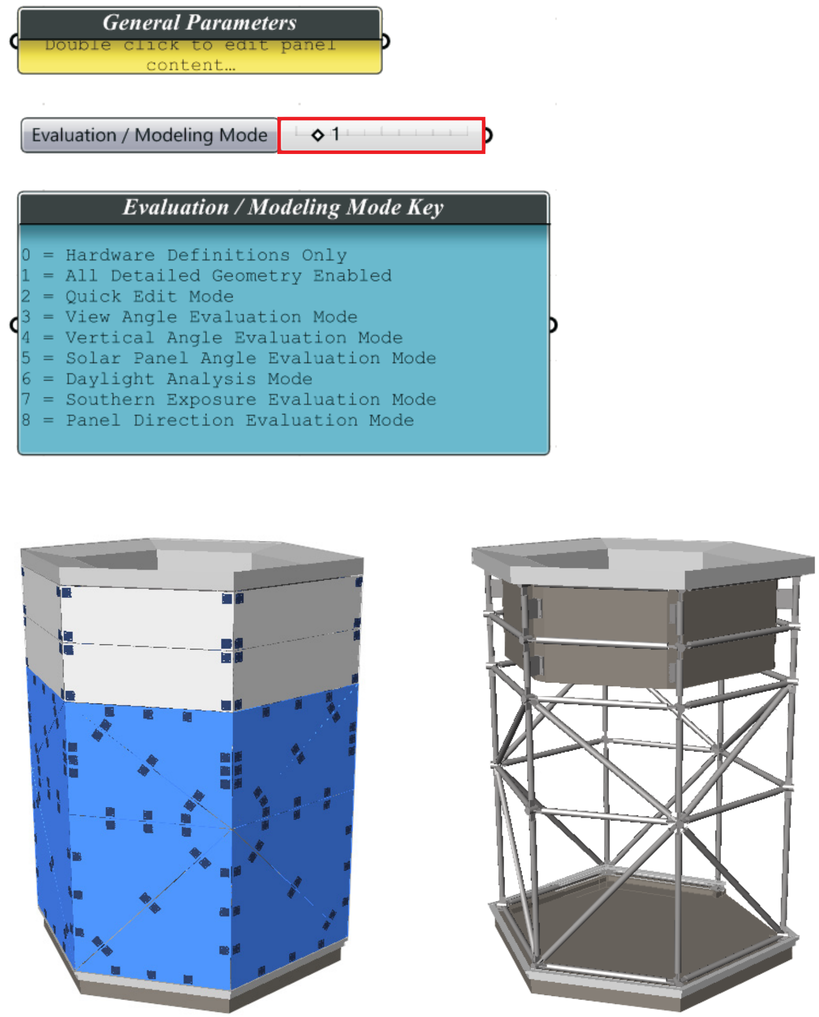


Figure 4.3-11 *Step 10: Enable detailed geometry*

4.4 *Statistics Reporting*

When an OpenOffice statistics spreadsheet is generated by Grasshopper, it contains only raw data. In order to put the data in a readable format, the data in the first spreadsheet needs to be brought into a second spreadsheet file containing labels. Unfortunately, the links between the spreadsheets need to be established manually any time a new filename or a different directory is used for the spreadsheet that Grasshopper creates. The process for linking the files is a little bit tedious, but very straightforward. The steps are as follows:

<u>Step 1 - Set the filename and path</u>

In the parameters section of the Grasshopper model under *File Information*, set the file name and path for the spreadsheet that Grasshopper will generate. By default the filename is Raw Model Statistics.ods. Make sure the *Statistics File #* slider is set to match the box in which the data was entered.

<u>Step 2 - Generate the spreadsheet from Grasshopper</u>

Set the *Create Statistics File?* parameter to *True*. If a file is not created automatically, go to the *Solution* menu in Grasshopper and select *Recompute*. Turn the switch back to *False* once the file has been created so that a new file won't be created again and again every time something changes in the Grasshopper model.

<u>Step 3 - Reference the spreadsheet file in the report spreadsheet</u>

Open the "Model Statistics Report" spreadsheet. Select the first cell in column B. Change the path and file name in the cell to reflect the name and location of the file just created by Grasshopper, making sure not to erase the sheet and cell number reference. Repeat this process for every cell in column B by copying and pasting the information. When finished, the spreadsheet can be saved in any location and under any name.

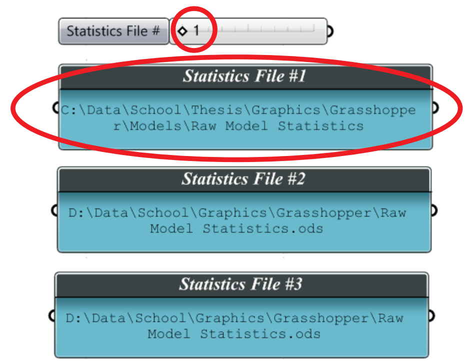


Figure 4.4-1 *Step 1: Set the filename and path*

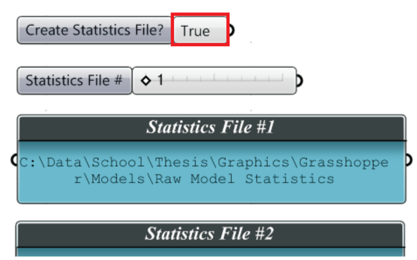


Figure 4.4-2 *Step 2: Generate the spreadsheet*

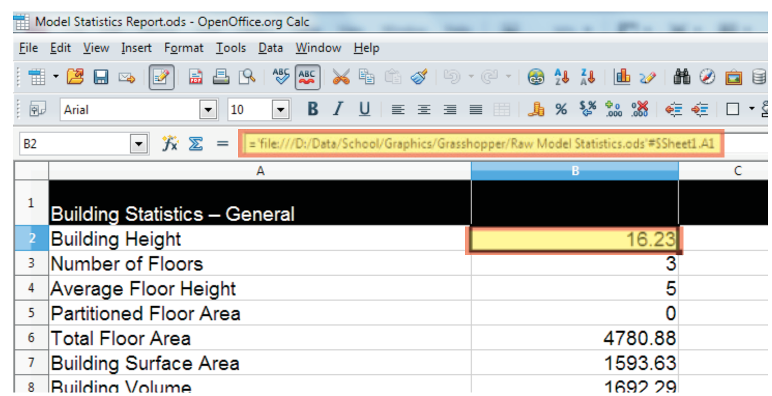


Figure 4.4-3 *Step 3: Change the file path for each cell (top highlight)*

4.5 *View Optimization*

The following is an example of how the view optimization tool can be used. A site in downtown Toronto on the North side of Queen Street West at Soho St. provides the context for this example. The site has the potential for panoramic view of the city's skyline, beginning with the downtown core in the East, and ending with the CN tower in the South.

This analysis example starts with a simple box-shaped building and then edits the façade shape in an attempt to improve views. The view threshold is set to 45-degrees. If an object is more than 45-degrees from a panel's normal vector, the panel will display as red. Otherwise, the panel will display as yellow, green or blue depending on the strength of the view, with yellow indicating a direct view and blue indicating a more oblique view.

Figure 4.5-1 *View South-East from the site towards First Canadian Place (white tower, far left) and the CN Tower (far right).*

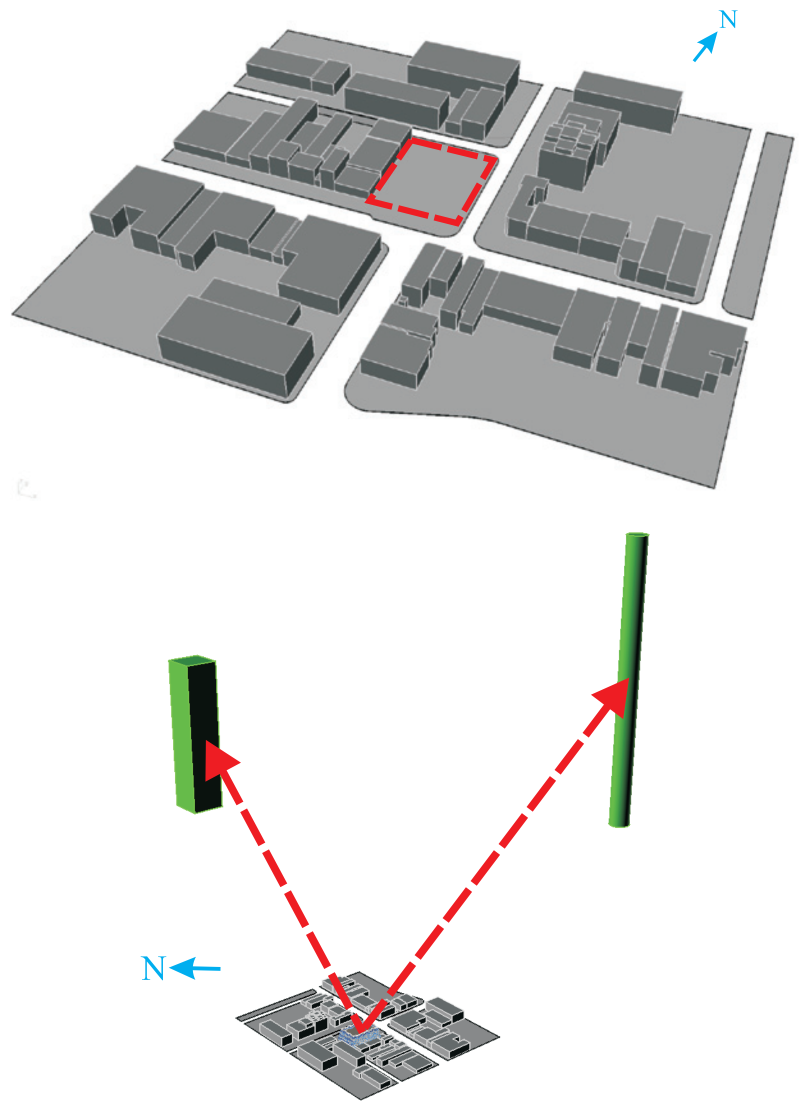
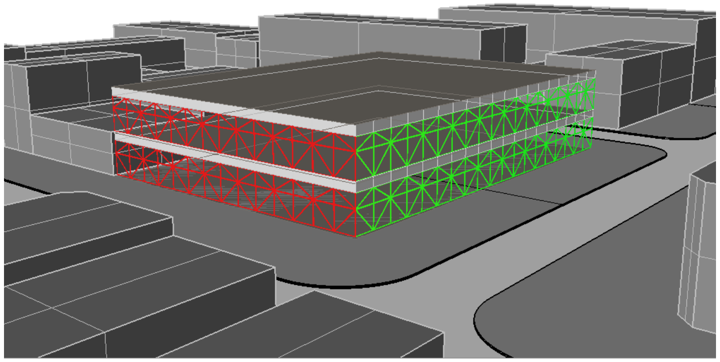
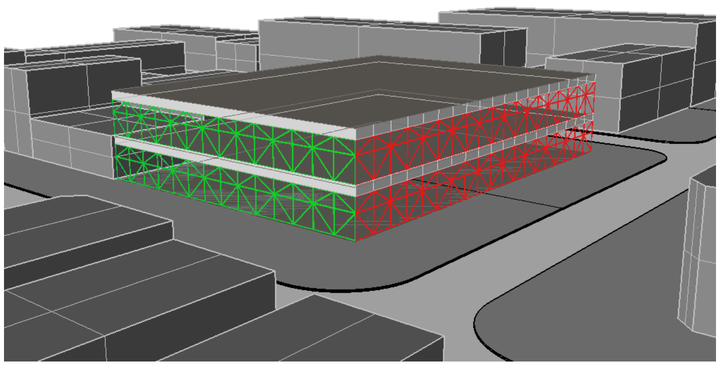


Figure 4.5-2 *A massing model of the site (top) and objects to the East and South representing First Canadian Place and the CN Tower, respectively. The view objects are placed at the mid-points of each tower (bottom).*

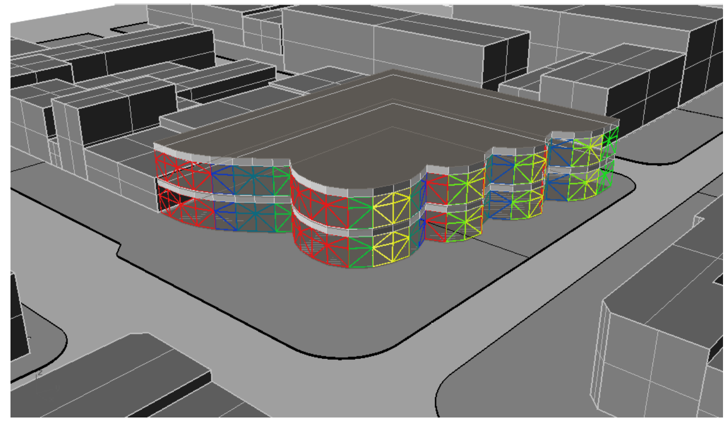


Analysis Statistics – Views	
View Angle Threshold	45
View Object Evaluated	1
Number of Panels Evaluated	264
Number of Panels Within Threshold	152
Percentage of Panels Within Threshold	57.58

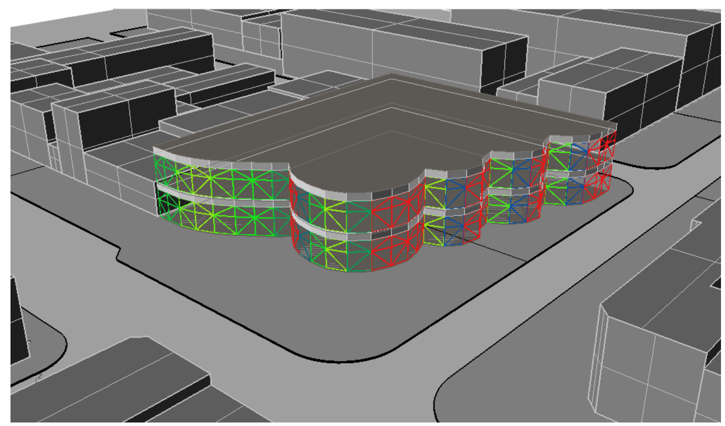


Analysis Statistics – Views	
View Angle Threshold	45
View Object Evaluated	0
Number of Panels Evaluated	264
Number of Panels Within Threshold	136
Percentage of Panels Within Threshold	51.52

Figure 4.5-3 *A flat façade with First Canadian Place selected as the view target (top), and the CN Tower set as the view target (bottom).*



Analysis Statistics – Views	
View Angle Threshold	45
View Object Evaluated	0
Number of Panels Evaluated	264
Number of Panels Within Threshold	144
Percentage of Panels Within Threshold	54.55



Analysis Statistics – Views	
View Angle Threshold	45
View Object Evaluated	1
Number of Panels Evaluated	264
Number of Panels Within Threshold	168
Percentage of Panels Within Threshold	63.64

Figure 4.5-4 *A curved façade with First Canadian Place selected as the view target (top), and the CN Tower set as the view target (bottom).*

Compared to a flat façade, the curved façade increases the horizontal scope of the views from the two sides of the building being evaluated.

Since the top floor has a fairly unobstructed view of the skyline, two further refinements are explored in order to increase the vertical scope of the views on that floor: beveling the top row of panels, and sloping the top two rows of panels. The former offers better street views, while the latter provides more direct skyline views.

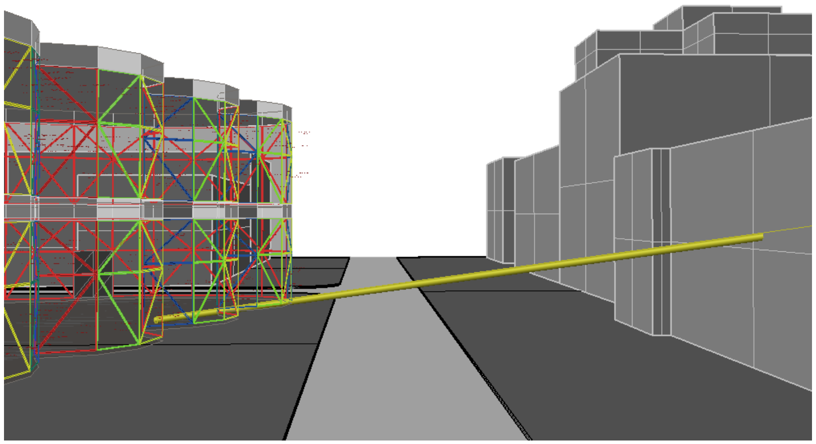


Figure 4.5-5 *The view of the skyline is more limited on the bottom floor and is obstructed in some cases.*

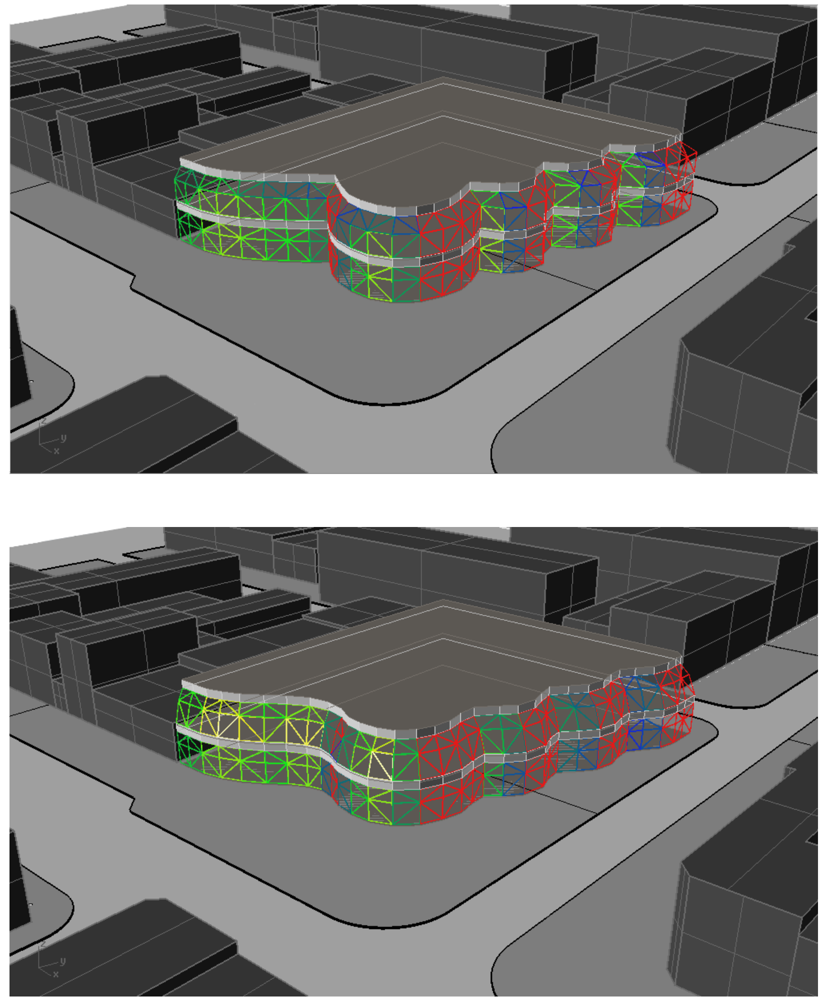


Figure 4.5-6 *Two options are explored for improving the vertical scope of the views.*

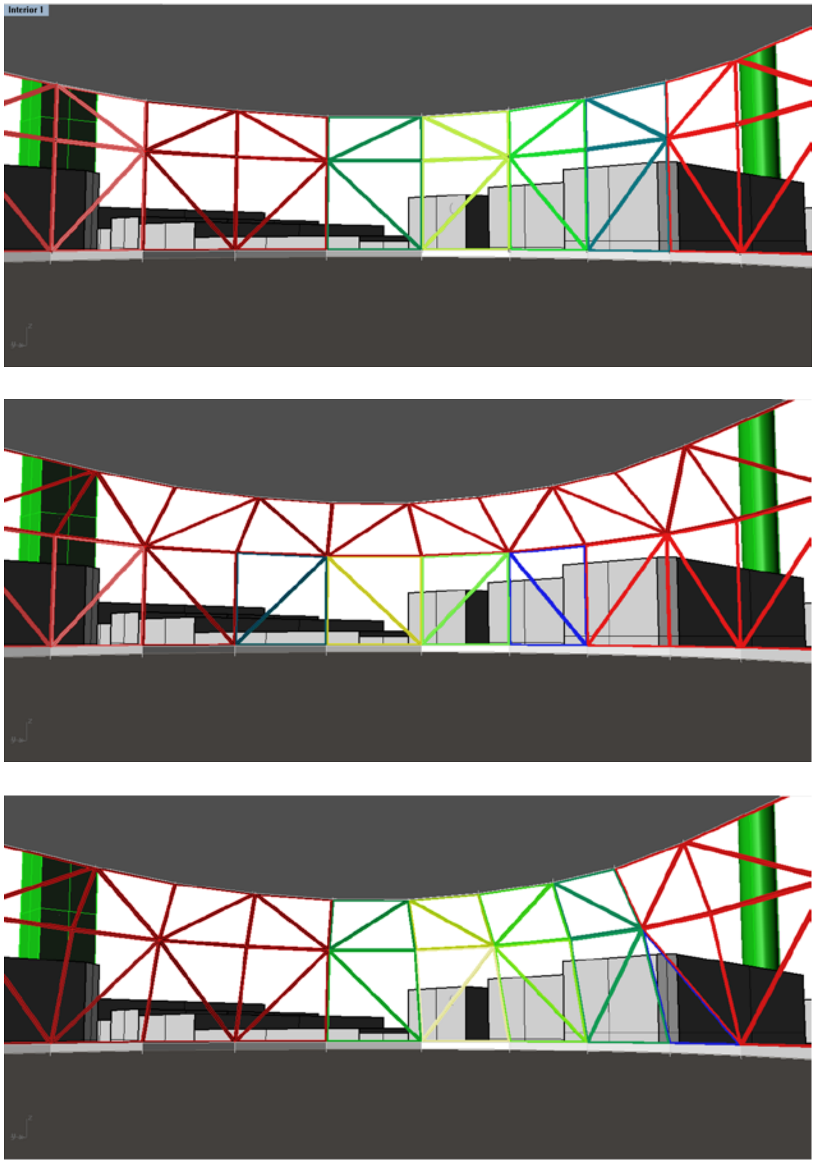


Figure 4.5-7 *Views from the interior: completely vertical panels (top), beveled top panels (middle), and sloped panels (bottom).*

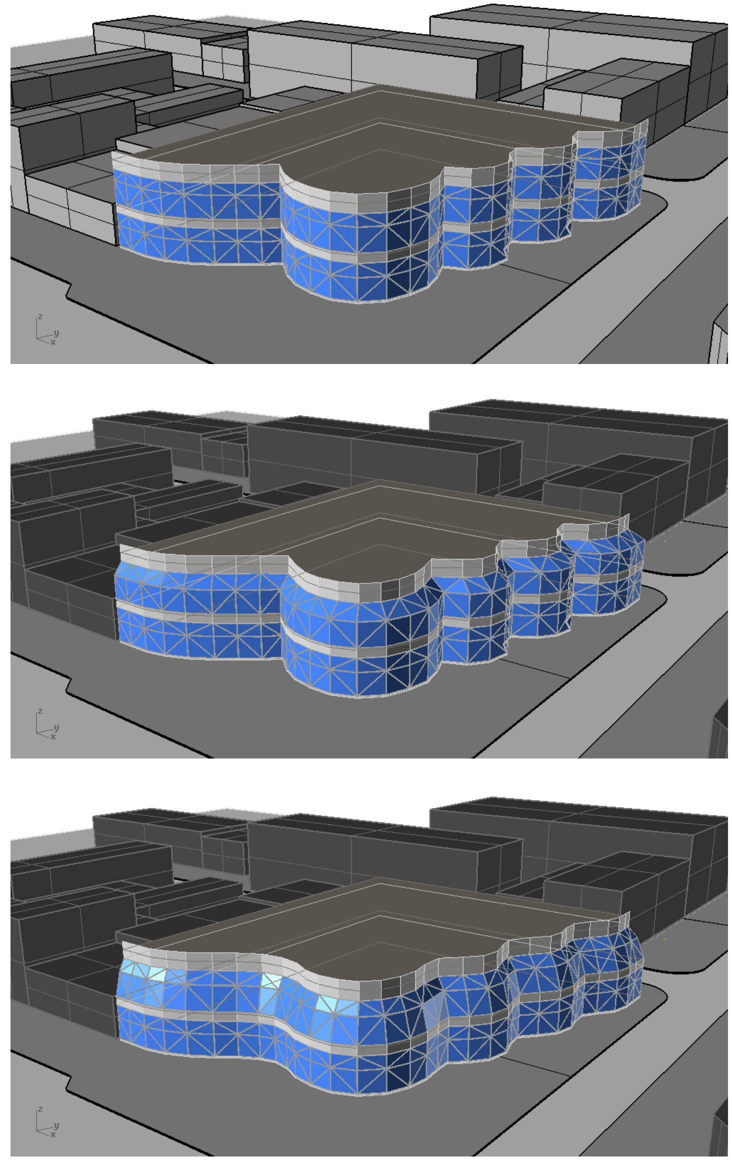


Figure 4.5-8 *The appearances of the options explored (Quick Edit view).*

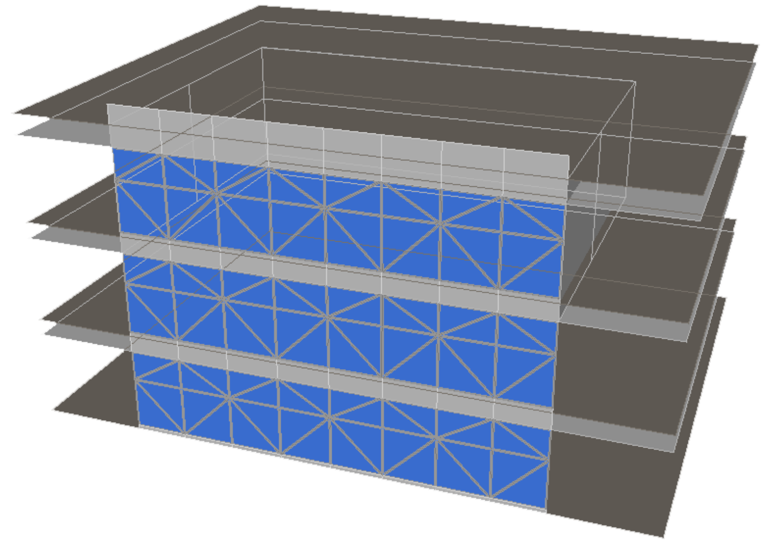
4.6 Daylighting Optimization

There are many factors to consider when evaluating daylighting effectiveness, including lighting levels, lighting angles, glare, contrast, light distribution, and solar heat gains[41]. This section looks briefly at how to perform a Daylight Autonomy or Daylight Factor analysis in the model. The analysis is somewhat simplistic and is only for illustration purposes.

Note: Diva, the Daylighting evaluation plugin for Grasshopper and Rhino, is still in development at this point in time and has many bugs and inconsistencies. Some analysis types are currently working in the model while others aren't. The node-based analysis modes, such as Daylight Factor and Daylight Autonomy are functioning reasonably well currently, while visualization-based analysis modes, such as glare analysis, are unreliable.

Daylight Factor and Daylight Autonomy

Daylight Factor measures the brightness of an area within a space as a percentage of the light outdoors, while Daylight Autonomy measures the percentage of time that an area within a building receives an amount of daylight above a certain threshold, measured in lux. The desired thresholds for Daylight Factor and Daylight Autonomy will vary depending on the building type and the tasks being performed, as well as the standards used. 300 lux is often considered an adequate amount of light for mixed computer and paperwork activity in an office environment for Daylight Autonomy purposes[42], and 5% is considered an adequate Daylight Factor.[43]



Analysis Statistics – Daylighting	
Floor Under Evaluation	2
Number of Light Shelves	0
Window Sill Height	0.2
Window Aperture Height	4
Window Tops (from slabs)	4.2
Daylight Autonomy Threshold (lux)	33
Daylight Factor Threshold (%)	2
Number of Evaluation Nodes	121
Number of Nodes within Threshold	67
Percentage of Nodes within Threshold	55.37
Average Node Value	35.41

Figure 4.6-1 *A partitioned, flat section of façade used as a baseline reference.*

<u>Example: Evaluating Daylight Autonomy on a South-facing Façade</u>

DIVA settings:

Minimum Illuminance: *300 lux*
Lighting Control: *Photosensor controlled dimming*
Radiance Parameters: *-ab 2, -ad 1000, -as 20, -ar 300, -aa 0.1*
Advanced Shading Devices : *No shading*
Geometric Density: *100*

Model parameters:

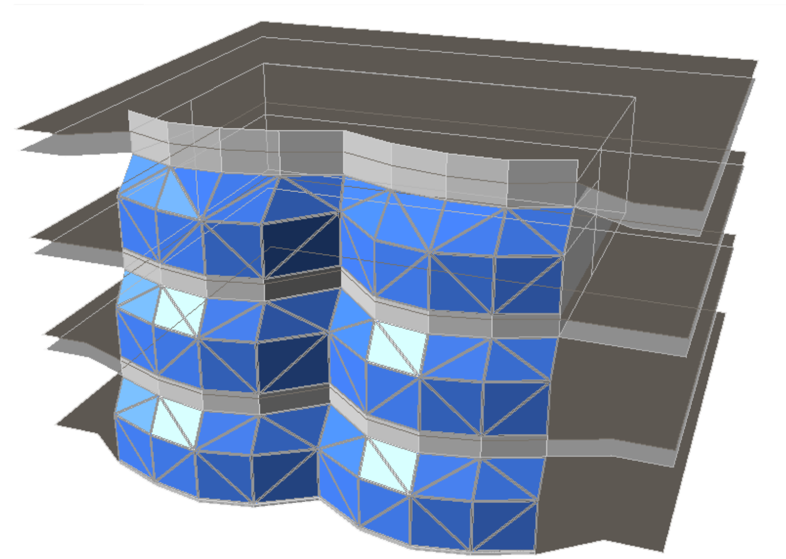
Daylight Autonomy Threshold (% time at minimum lux): *33*
Generate Analysis Nodes: *True*
Evaluation Grid Vertical Offset: *1 meter*
Evaluation Grid Node Count (U): *10*
Evaluation Grid Node Count (V): *10*
Use Panel Centroids for Nodes: *False*
Light Shelves: *Off*

Results: flat façade

Average amount of time at 300 lux for all areas: *35%*
Percentage of space with >33% Daylight Autonomy: *53%*

Results: curved/beveled façade

Average amount of time at 300 lux for all areas: *39%*
Percentage of space with >33% Daylight Autonomy: *68%*



Analysis Statistics – Daylighting	
Floor Under Evaluation	2
Number of Light Shelves	0
Window Sill Height	0.2
Window Aperture Height	4
Window Tops (from slabs)	4.2
Daylight Autonomy Threshold (lux)	33
Daylight Factor Threshold (%)	2
Number of Evaluation Nodes	121
Number of Nodes within Threshold	82
Percentage of Nodes within Threshold	67.77
Average Node Value	38.98

Figure 4.6-2 *Results for a curved and tiered configuration.*

Viewing the Results

The numerical results can be viewed either in the statistics section of the Grasshopper canvas or by generating a statistics report. The nodes themselves can be viewed by selecting *View Evaluation Nodes Only* toggle in the *Daylight Evaluation Parameters* section of the Grasshopper canvas.

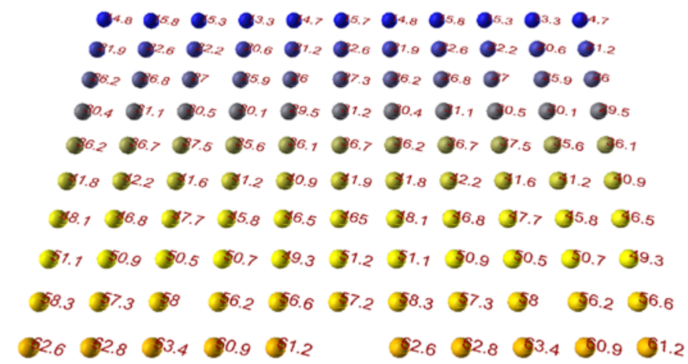


Figure 4.6-3 *Using the "View Evaluation Nodes Only" toggle allows the user to more clearly see the evaluation nodes and their corresponding values, in order to get a better indication of the overall light distribution.*

4.7 Optimizing Solar Panels and the Building Envelope

Solar Panels

There are three ways to optimize solar panels positioning:

1) Use the *Solar Panel Angle Evaluation Mode* to evaluate the positions of panels relative to South as well as a specified optimum vertical angle. This is ideal for quick positioning of panels.

2) In *Daylight Evaluation Mode*, enable evaluation nodes on the solar panels and use the *Radiation Nodes* Diva analysis to evaluate the amount of solar radiation hitting each panel. This gives a better idea of the relative performance of panels.

3) Export all of the geometry or just the solar panel geometry to a third-party program such as Ecotect which can evaluate the actual wattage each solar panel will generate given an efficiency rating. This can be used for a fairly accurate assessment of the effectiveness of the panels.

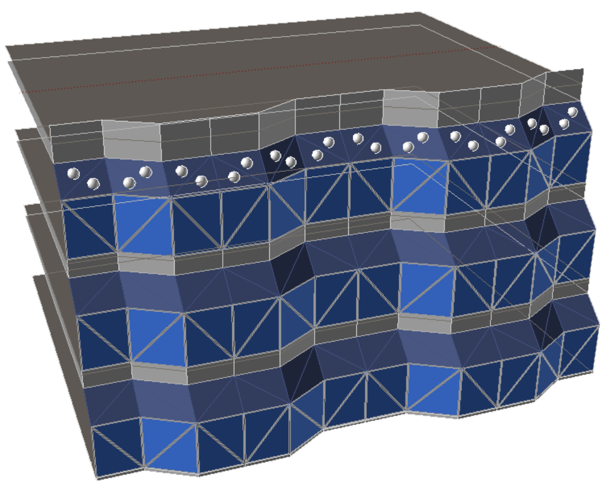
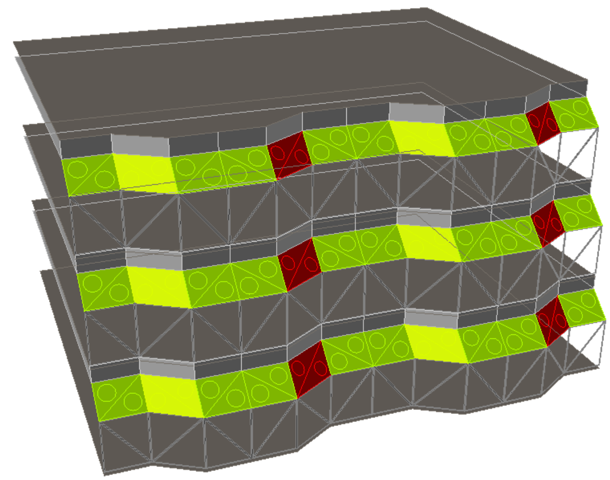


Figure 4.7-1 *Evaluation nodes are applied to solar panels on the top floor for a Radiation Node analysis.*



Analysis Statistics – Solar Panels	
Optimum Vertical Angle	45
Optimum Vertical Angle Threshold	40
South Orientation Threshold	45
Number of Panels Evaluated	72
Number of Panels Within Thresholds	60
Percentage of Panels Within Threshold	83.33

Figure 4.7-2 *A vector analysis showing which panels are at the most optimal angles according to the thresholds set by the user.*

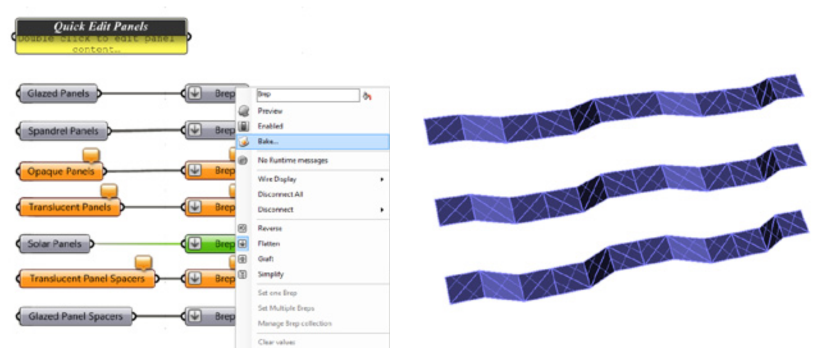


Figure 4.7-3 *Solar panels are "baked" so that they can be exported to Ecotect for an accurate annual energy gain assessment.*

Building Envelope Properties

Some of the statistics discussed below can be useful when trying to optimize the building for energy performance.

1) Surface-to-volume ratio

This can give an idea of how daylighting or other strategies that change the surface area of the building may affect the rate of heat transfer through the building envelope. This statistic can be calculated for the whole building or just a portion of it.

2) Window-to-wall ratio

This can give an idea of the thermal efficiency of the envelope given a particular choice of panel types and panel row heights, as well as help indicate how efficient a daylighting strategy is. The "translucency factor" can be used to to determine how the effects translucent panels are to calculated. For example, a 0.2 factor means that each translucent panel is counted as 20% of a glazed panel and 80% of an opaque panel.

3) Thermal resistance statistics

The overall thermal resistance of the façade can be approximated if R-values have been entered by the user for each panel type.

4) Panel costing

If panel costs per square meter are entered, the overall panel costs can be approximated. This can be useful for determining whether the solar panels are producing enough energy to justify their costs, for example.

Building Statistics – General	
Full Facade View (1) or Partial Facade View (0)	1
Partial Facade - Start Point	0
Partial Facade - End Point	12
Building Height	13.23
Number of Floors	3
Average Floor Height	4
Partitioned Floor Area	0
Total Floor Area	1746
Building Surface Area	2355.85
Building Volume	6948.3
Surface to Volume Ratio	2.95
Window to Wall Ratio	0.75
Window to Wall Ratio w/ Translucency	0.75
Translucency Factor	0.2
Thermal Resistance Statistics	
Translucent Panel R-Value	3
Opaque Panel R-Value	5
Glazed Panel R-Value	2
Solar Panel R-Value	5
Spandrel Panel R- Value	5
Average R-Value of Building Envelope	2.74
Panel Costing Statistics	
Translucent Panel Cost / Sq.m	100
Opaque Panel Cost / Sq.m	100
Glazed Panel Cost / Sq.m	100
Solar Panel Cost / Sq.m	100
Spandrel Panel Cost / Sq.m	100
Total Translucent Panel Cost	0
Total Opaque Panel Cost	0
Total Glazed Panel Cost	67813
Total Solar Panel Cost	12010
Total Spandrel Panel Cost	21700
Total Cost of Trangular Panels	79823
Average Cost per Triangular Panel	448.44

Figure 4.7-4 *Various statistics that can be helpful when trying to predict energy and cost efficiencies.*

5. The Modeling Process

This section gives a brief explanation of how Grasshopper works and then explains how some of the key elements of the parametric model were created.

Basic Grasshopper Principles

There are five basic types of building blocks or "components" in Grasshopper: geometric inputs, geometry generators, number generators, user parameters, and data filters.

Geometric Inputs

All Grasshopper models require basic geometric objects from Rhino, such as vertex points. The positions or shapes of these objects typically determines the overall shape of the model.

Geometry Generators

Geometry generators are components in Grasshopper that create geometry using either numeric information or by combining other existing geometric elements.

Number generators

Number generators are typically used for one of three purposes: to sort geometry into groups in order to modify the geometry in batches, to create lists of parameters for use in the generation of geometry, or to modify other number sequences.

User Parameters

User parameters can be used to change the input values of any type of component within Grasshopper except geometric input components. Number sliders and true/false boolean toggle switches are the most common way in which user parameters are changed within a model.

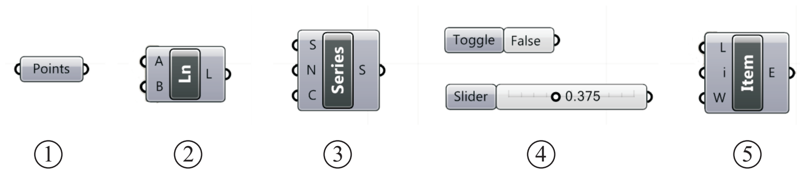


Figure 5-1 *Examples of basic Grasshopper building blocks: 1) Points component (geometric inputs); 2) a line component (geometry generator); 3) a series component (number generator); 4) a toggle switch and numeric slider (user parameters); 5) a list item component (data filter).*

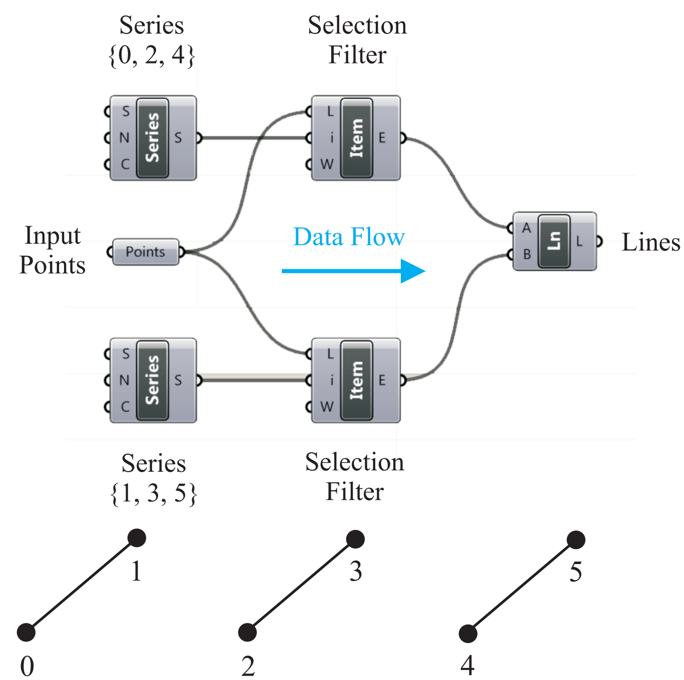


Figure 5-2 *A very basic point-sorting routine that selects specific points from a group of six points to create lines (bottom).*

Data Filters

All objects and number sequences are organized and stored in lists within Grasshopper. Lists are what allow geometry or numbers to be processed in batches. A list can be simple and linear or there can be sub-groups within the list.

With the use of data filter components, lists can be manipulated in a variety of ways: they can be reordered, items can be removed or inserted, items can be selectively copied from one or more lists to form a new list, and so on. These types of operations are necessary for most modeling and analysis routines.

Point Rows and Sequences

The following steps describe how an array of Rhino input points are sequenced for use in the model. The amount of points involved does not matter because all number generators scale their output according to the number of rows and columns in the array.

1) The Rhino input points are put in a single group. A subset of the input points is created using a number range. In this case the subset is {(1-3), (6-8), (11-13)}, representing a partial view of the façade. The new group of points is then organized into discreet rows, with all points in the same columns sharing the same index numbers. This allows points to be selected by column as well as by row, which makes complex multi-row point sequences possible.

2) If the full façade is being viewed, the first point in each row is copied and then added again to the end of the row so that the façade becomes a closed loop.

3) Point rows used to create the spandrel panels are offset from the original input points where necessary. The use of offset points for the spandrels, as opposed to moveable Rhino points, ensures that the spandrel panels always remain vertically oriented.

4) The point rows are divided into groups according to how they are used in constructing geometry. The points for the tops of triangular panels form one group while the points at the bottoms of the panels form another group.

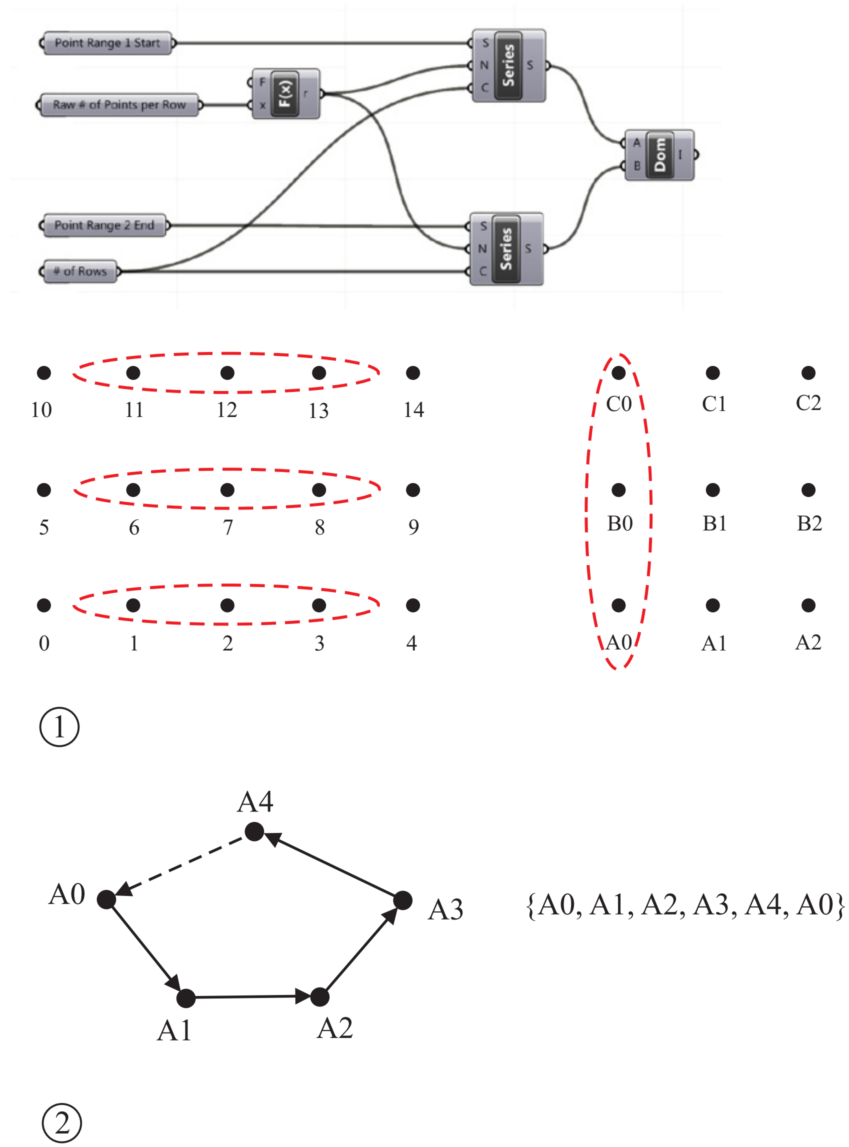


Figure 5-3 *Creating point rows and sequences, steps 1 and 2.*

If the symmetrical triangulation pattern is being used, the panel pattern is mirrored vertically and the point rows need to be organized into four groups instead of two.

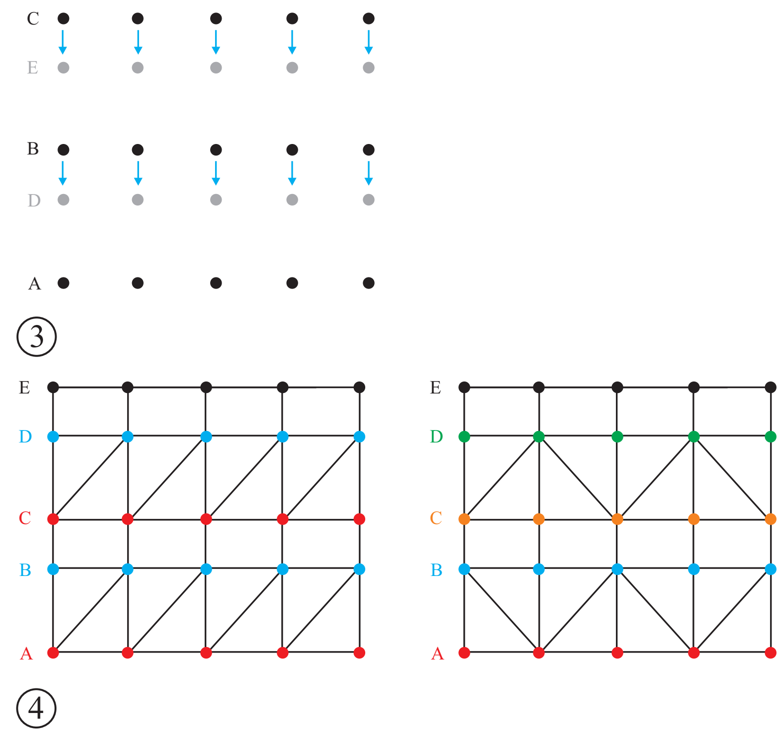


Figure 5-4 *Creating point rows and sequences, steps 3 and 4.*

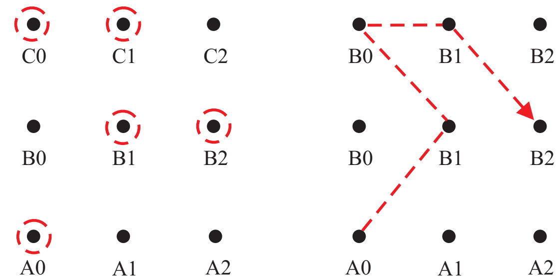


Figure 5-5 *Once points in the same column of each row have matching index numbers, virtually any sequence of points can be created by weaving together point number sequences (see "Triangulation Patterns" section below).*

<u>Triangulation Patterns</u>

The creation of panel and truss patterns involves weaving together sequences of points from each row. This section describes the process of creating the symmetrical triangulation pattern in detail.

1) The point rows are divided into four groups, denoted here by different colors (the top row is excluded). The point generators have to query the total number of rows and total number of building storeys to properly perform the calculations. This is because when there is an odd number of storeys and only one row of panels per storey, as is the case in this example, the groups will be uneven in size.

2) Points from adjacent rows are woven together in order to create triangles. The pattern must be optimized for the creation of unified normal vectors for the panels. This means always sequencing the points in either a clockwise or counterclockwise direction (clockwise in this case). The sequences are created as follows:

Sequence A: {1,0,1,2,1,1,3,2,3,4,3,3,5...}

> Combine sequences a and d using the weave pattern {a,a,d}
> a: {1,0,2,1,3,2,4,3,5,4...}, cull first and last numbers
> d: {1,1,3,3,5,5...}, cull last number if the number of points per row is an even number

Sequence B: {0,0,1,2,1,2,2,2,3,4,3,4,4...}

> Combine sequences a and b using the weave pattern {b,a,a}
> a: {0,0,1,1,2,2,3,3,4,4...}, cull first and last numbers
> b: {0,0,2,2,4,4...}, cull first and last numbers

Sequence C: {1,0,0,2,1,2,3,2,2,4,3,4,5...}

> Combine sequences a and b using the weave pattern: {a,a,b}
> a: {1,0,2,1,3,2,4,3,5,4...}, cull first and last numbers
> b: {0,0,2,2,4,4...}, cull first and last numbers

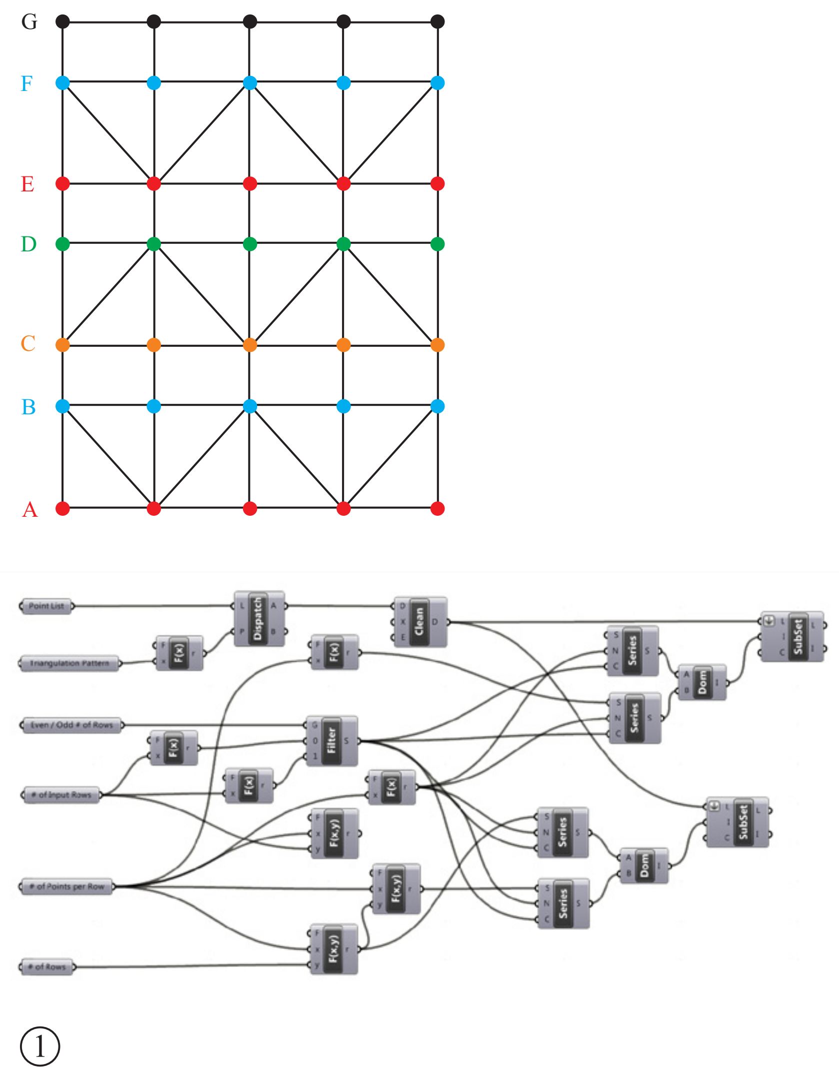


Figure 5-6 *Creating triangulation patterns.*

Sequence D: {1,0,1,1,1,2,3,2,3,3,3,4,5...}

Combine sequences c and d using the weave pattern {d,c,c}
c: {0,0,1,1,2,2,3,3,4,4...}, cull first and last numbers
d: {1,1,3,3,5,5...}, cull last number if the number of points per row is an even number

3) The final point sequences are woven together:

Sequence A-B: {B0,A1,A0,B0,B1,A1,B2,A2,A1...}

Combine sequences A and B using the weave {B,A,A,B,B,A}
A: {1,0,1,2,1,1,3,2,3,4,3,3,5...}
B: {0,0,1,2,1,2,2,2,3,4,3,4,4...}

Sequences C-D: {C1,C0,D1,C0,D0,D1,C2,C1,D1...}

Combine sequences C and D using the weave {C,C,D,C,D,D}
C: {1,0,0,2,1,2,3,2,2,4,3,4,5...}
D: {1,0,1,1,1,2,3,2,3,3,3,4,5...}

4. The sequences in step 3 are divided into groups of three in order to create triangles using closed polylines: {C1,C0,D1}, {C0,D0,D1}, etc. and {B0,A1,A0},{B0,B1,A1}, etc.

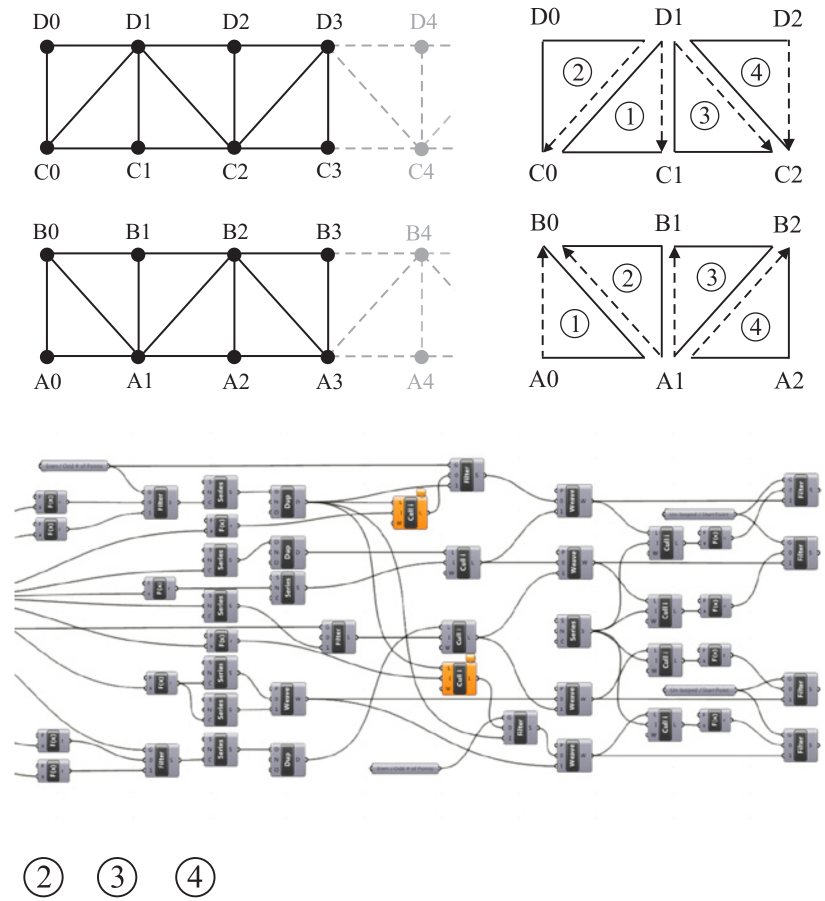


Figure 5-7 *Creating the triangulation patterns.*

Truss Members

The vertical truss members follow patterns similar to the panels, but the sequences are a bit simpler since many redundant lines can be eliminated and surface normals are not a factor. A sequence between rows A and B, for example, can simply be: {A0,B0,A0,B1,A1,B1,A2,B1,A2,B2,A2,B3...}. However, when only part of the façade is being viewed, the truss pattern changes depending on where the partial view starts and ends, so the point sequence generators have to query whether the sequence starts and ends on even or odd-numbered points and how long the sequence is in order to generate the correct patterns. Points are then culled from the sequences in order to eliminate extraneous geometry.

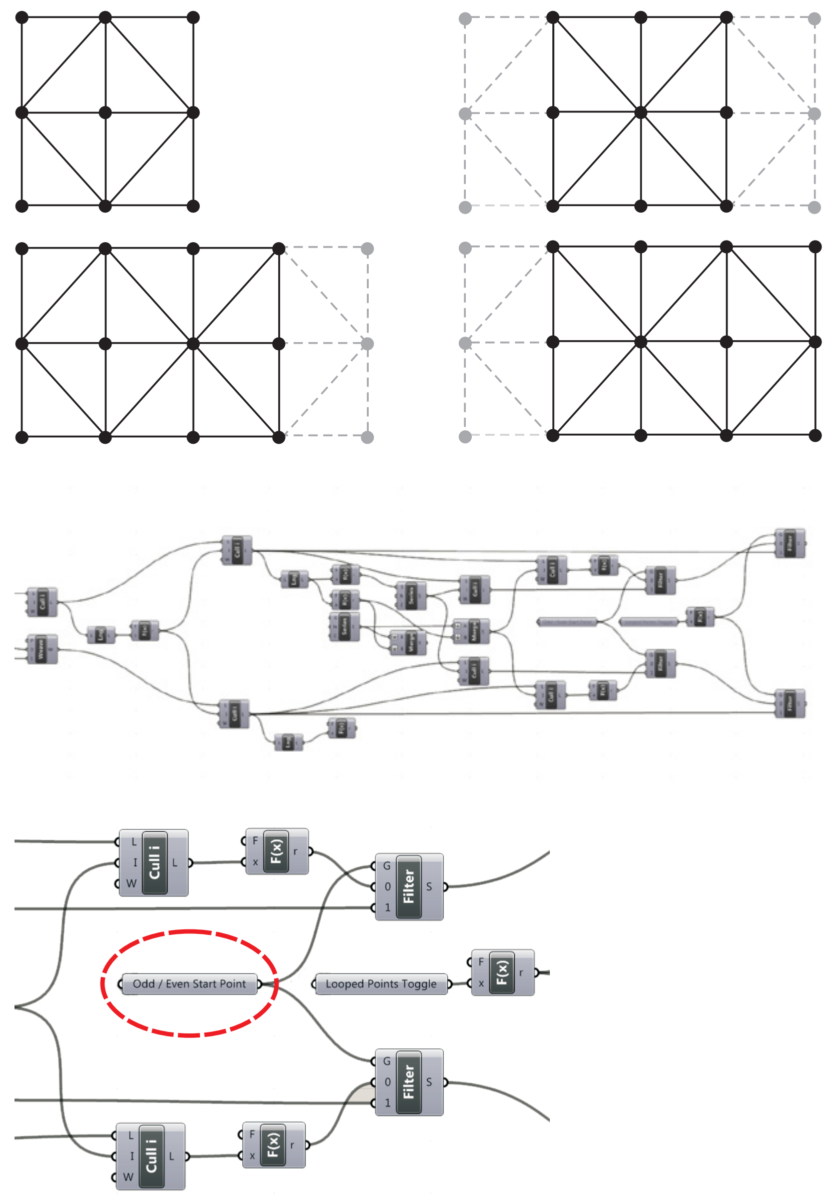


Figure 5-8 *Truss member culling for partial façade views.*

Offset Points

One of the challenges of working in Grasshopper at the time the model was created was finding a way to create proper offset points from an original set of Rhino input points. Normally, the original points could be used to create a line, the line could be offset, and then the desired points could be extracted from the offset line. This technique, however, was not possible at the time due to an quirk in the way Grasshopper created offset lines.

When a line was offset in Grasshopper, extraneous vertices would be created on the new line if the offset was on the same plane as the original line. These vertices could not be reliably culled because their number and locations depended on the shape of the original line, and the line would change shape whenever the façade's shape was manipulated. To work around this issue, an alternate method of creating offset points was developed:

1) The points in the original input point rows are sequenced into groups of three composed of each point and its two neighboring points. In some cases additional points need to be added at both ends of the point rows. For example, if only part of the façade is being viewed, points beyond the edges of the panels being viewed need to be added to each end of the point rows.

2) The groups of points are used to create open-ended polylines. The polylines are offset the distance and direction of the final offset points.

3) Vectors are created from the center points to the adjacent points. The two vectors are then added in order to create a third vector that bisects the angle between lines on the horizontal plane. This does not work in cases where the line is straight, so an algorithm checks to see if the angles is 180 degrees and, if that is the case, it creates a vector at 90-degrees.

4) Additional vectors are created in the Z-axis direction. Two sets of lines are then created along the Z-axis and in the direction of the bisecting vectors. Points at the end of each of these lines are created, and these points are used in conjunction with the center point to create vertical planes aligned with the bisecting vectors.

5) The planes are intersected with the offset lines created in step 2 in order to create the desired offset points. Because planes are infinite in their dimensions, each plane must only be intersected with its corresponding line or

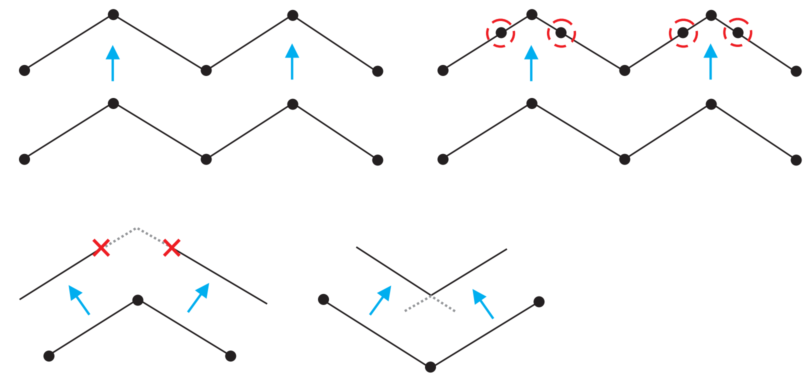


Figure 5-9 *The Grasshopper offset bug is most likely due to the program copying and moving individual segments of the polyline, extending or trimming the copies, and then failing to cull extra vertices created in cases where the lines were extended.*

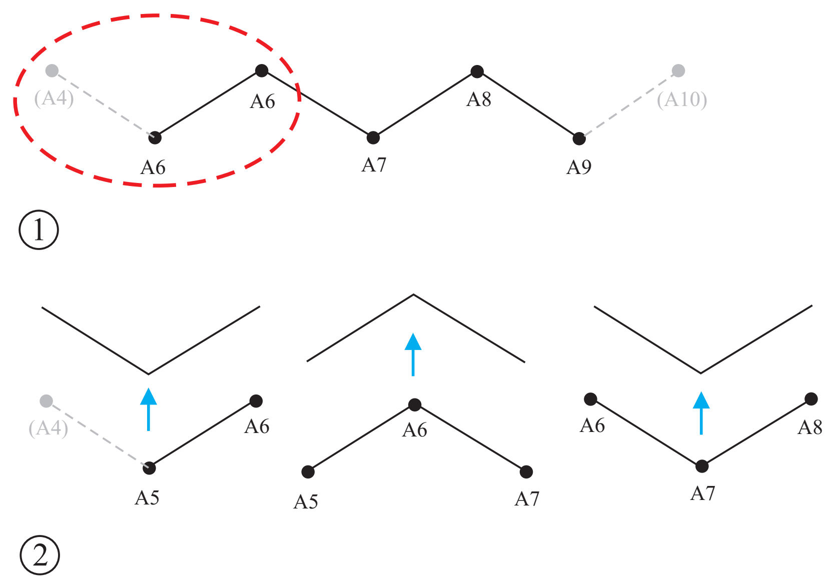


Figure 5-10 *Creating offset lines and intersection planes, steps 1 and 2.*

extraneous points can end up being created. Planes are used for the intersection operation because they are omnidirectional, whereas lines created from bisecting vectors point in the direction of the most acute angle by default, and this could be either direction depending on the façade's shape.

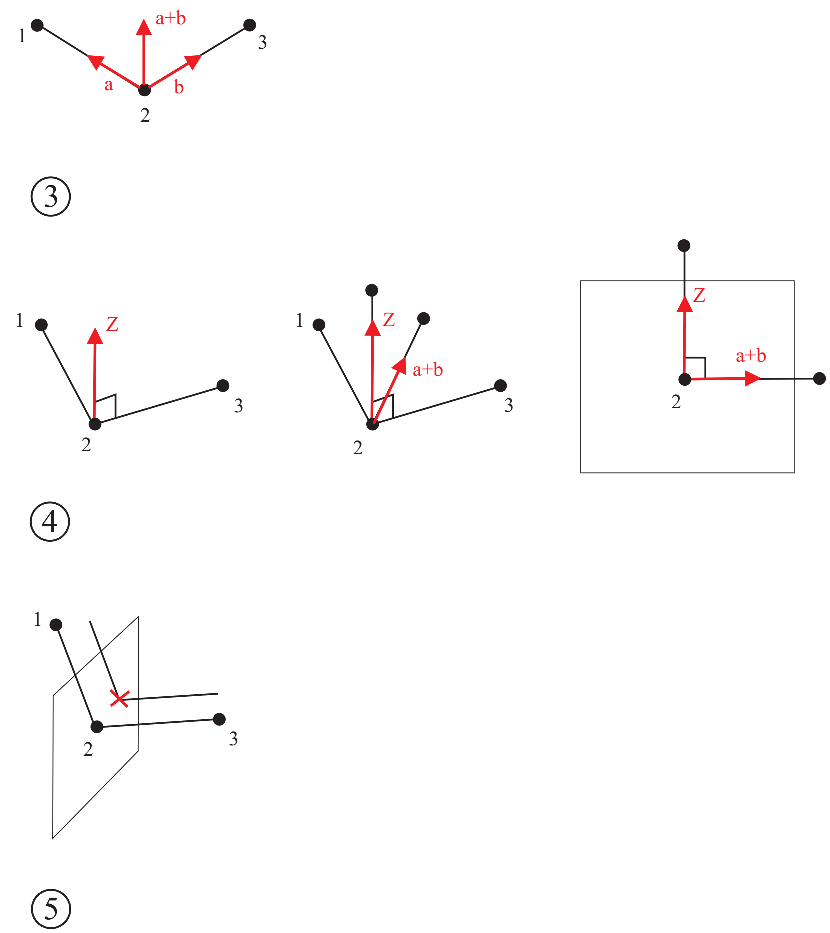


Figure 5-11 *Creating offset vectors and intersection planes, steps 3-5.*

Panel Components and Joint Offsets

In order to make the panel joints parallel to one another at all times, a somewhat elaborate offsetting mechanism is required. The process for creating the outer panes of the panels is described below.

1. A series of "vector lines" are created by connecting the original input points and a series of offset points (these lines bisect the angles between adjacent panels). The lines are then sequenced into groups of three corresponding corners of each triangular panel.

2. Triangular polylines are created for each panel and offset towards the center of the panels using planes parallel to the triangles for reference. This creates the joint space between panels on the outside.

3. The planes corresponding to each panel are offset in a direction perpendicular to the panels and intersected with the groups of vector lines, generating three new points per panel.

4. The new points are used to generate a new set of triangles. These these triangles are then offset inwards a distance equivalent to the joint width.

5. A loft object is created between the two triangles and then capped to form a complete pane.

Similar processes as the one described above are used to create the panel spacers and the inner panes. The inner edges of the spacers are perpendicular to the panel faces however.

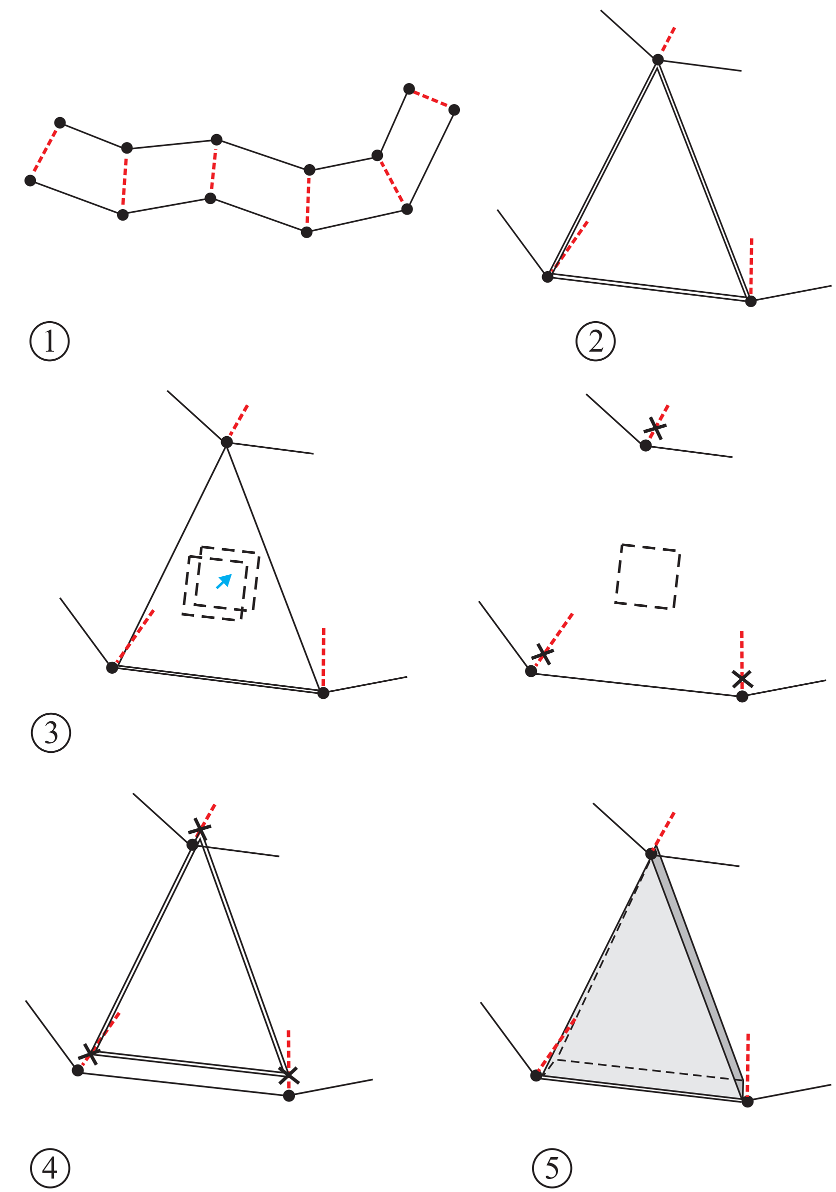


Figure 5-12 *Creating panel and joint offsets.*

Granular Control of Elements using a Spreadsheet

The ability to access external spreadsheet data in Grasshopper allows for some granular control over object parameters in a model. The façade model has been set up to allow granular control of panel types, empty bays, and light shelves through an OpenOffice spreadsheet. To import and export OpenOffice spreadsheet data, a third-party Grasshopper utility called gHowl is used. It's operation is very straight forward but also limited in some ways at this time. For example, only simple lists with single branches of data can be used.

This section describes how the spreadsheet data is used to generate geometry.

Panel Thicknesses

All panel types - glazed, opaque, translucent and solar - share the same basic topology, but each type has its own pane and spacer thicknesses. The way in which the spreadsheet data is used to create the different pane and spacer thicknesses is as follows:

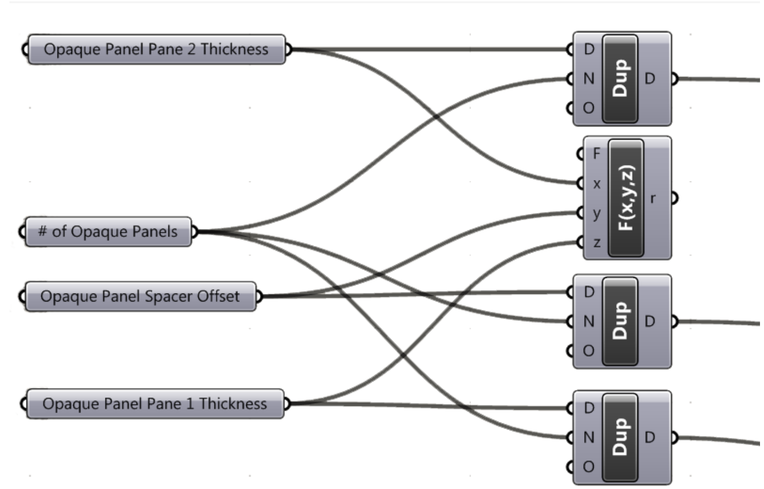
1) The number of instances of each type of panel is queried. If only part of the façade is being viewed, then only the appropriate subset of the panel list is considered.

2) The pane and spacer thickness parameters for each panel type (set by the user) are queried and then duplicated for each instance of that type of panel. For example:

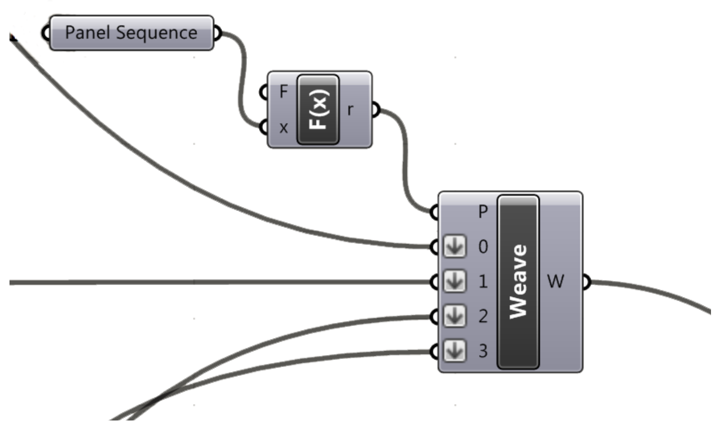
Solar Panel Spacer Thickness: 2
Number of Solar Panels: 3
Sequence: {2, 2, 2}

3) The sequences created in the previous step are merged together according to component type using the panel sequence in the spreadsheet as a weave pattern. The new sequence created represents the extrude parameters for the panes. For example:

Solar panel spacer thicknesses: {2, 2, 2}



① ②



③

Figure 5-13 *Panel thicknesses*

Glazed panel spacer thicknesses: {1, 1, ,1 ,1 ,1}
Panel sequence (weave pattern): {G, S, S, G, S, G, G, G}
Resulting sequence (extrude parameters): {1, 2, 2, 1, 2, 1, 1, 1}

4) The panel components are extruded according to the extrude parameter sequence.

5) The total panel thickness are calculated and passed on to the part of the model that instantiates the clip and hinge elements so that the assemblies can be properly positioned.

Light Shelves

Light shelves simply have an “on” or “off” state, indicated by a “1” or a “0” in the spreadsheet for any given location. If a light shelf location has a “0” associated with it, then no geometry is created at that location.

Empty Bays

Empty bays are listed by number in the spreadsheet. The implementation of empty bays was more complex than the other spreadsheet-based processes, in large part because the mechanism was added in an ad hoc fashion in the latter stages of the design process. In some cases the panel and truss geometry within the empty bays is not created, while in other cases the geometry is created and then either deleted or edited. Empty bays are sometimes tagged by inserting null values into the panel sequences.

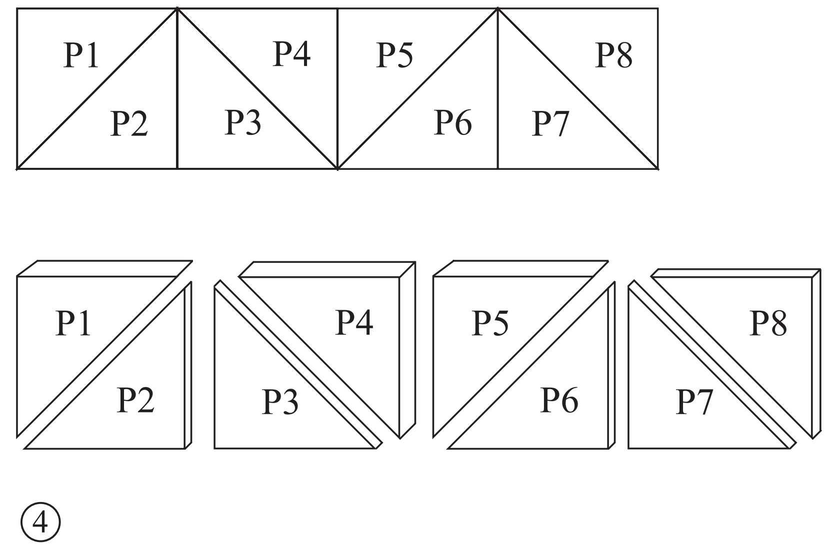


Figure 5.1-14 *Creating panel pane and spacer thicknesses, step 4.*

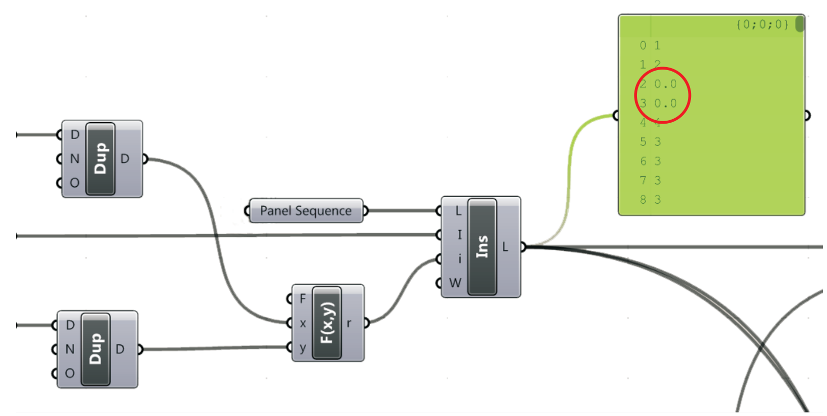


Figure 5-15 *For certain operations, empty bays are identified by inserting null values into the panel sequence.*

Panel Clip Assemblies

The clips for the panels are created as discreet objects and then instantiated onto the panels. This can be a useful approach for creating large arrays of complex yet identical objects because it allows the "master" object to be edited in isolation from the instantiated objects, greatly speeding up the editing process. Instancing objects also tends to be more efficient in terms of the processing power it uses because the objects are copied rather than recreated. The process of instantiating the panel hardware is described below.

1) The triangular lines that define the panel edges are divided into three segments, one for each side. Points are created along the lines according to the number of clips as well as the specified offset from the corners of the panels.

2) Three planes are created for each panel using the edge lines and the center point of the triangles.

3) The planes for each segment are copied to the points from step #1

4) Vectors are created by connecting the points from step #1 with points at an arbitrary distance along the X-axes of the planes (which are already aligned towards the center of the panels).

5) The clip and hinge assemblies are copied from their original construction planes to the new planes on the panel faces.

6) The clips are moved towards the center of the panel along the vectors created in step #4 according to the distance specified by the user in the *Panel Clip Edge Offset* parameter.

7) Clip assembly elements that need to be on the interior faces of the panels are moved along the normal panel normal vectors. Elements such as clip spacers and bolt shafts are extruded according to the respective panel thicknesses before being moved to the final destinations.

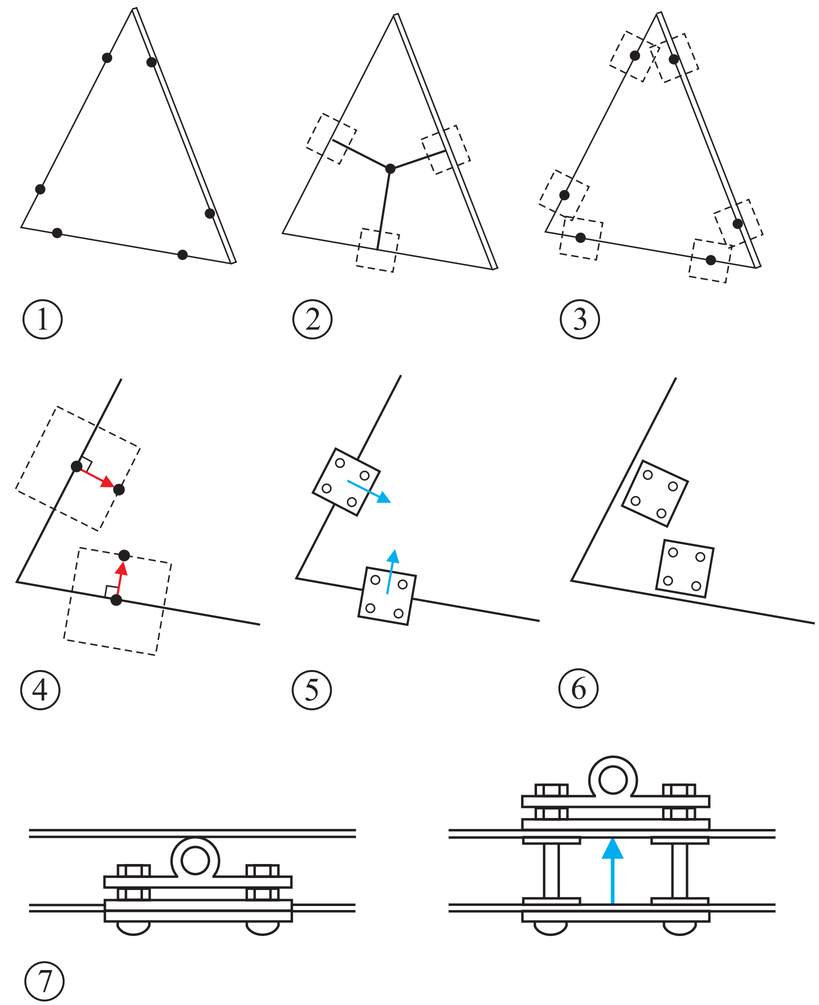


Figure 5-16 *Instantiating panel clip assemblies.*

<u>Panel Hinges</u>

Since the sizes and angles of the hinge pins change with the shape of the façade and the thicknesses of the panels, these elements can't simply populated onto the panels or the truss. The hinge pins are instead created through an interpolation process as follows:

1) The planes on the faces of the panels created previously for the clip assemblies are copied and moved to the inner panel face so that they can be used to build the hinge pin assemblies. The planes are then rotated selectively so the orientations of their X and Y axes will cause the hinge pins to always point upwards along vertical joints. Which planes are rotated depends on the panel row and the triangulation pattern used.

2) The corner points used to generate the original panel clip assemblies are copied to the planes.

3) Vectors are generated from the points, and the cross-product of the vectors is then used to generate a fifth point.

4) The fifth point is used to create a perpendicular plane.

5) The perpendicular planes are grouped according to the panel edge they are on and intersected with the truss center-lines to create a new set of points.

6) The hinge pin assemblies are created using the intersection points as guides for the lengths and rotation of the hinge plates.

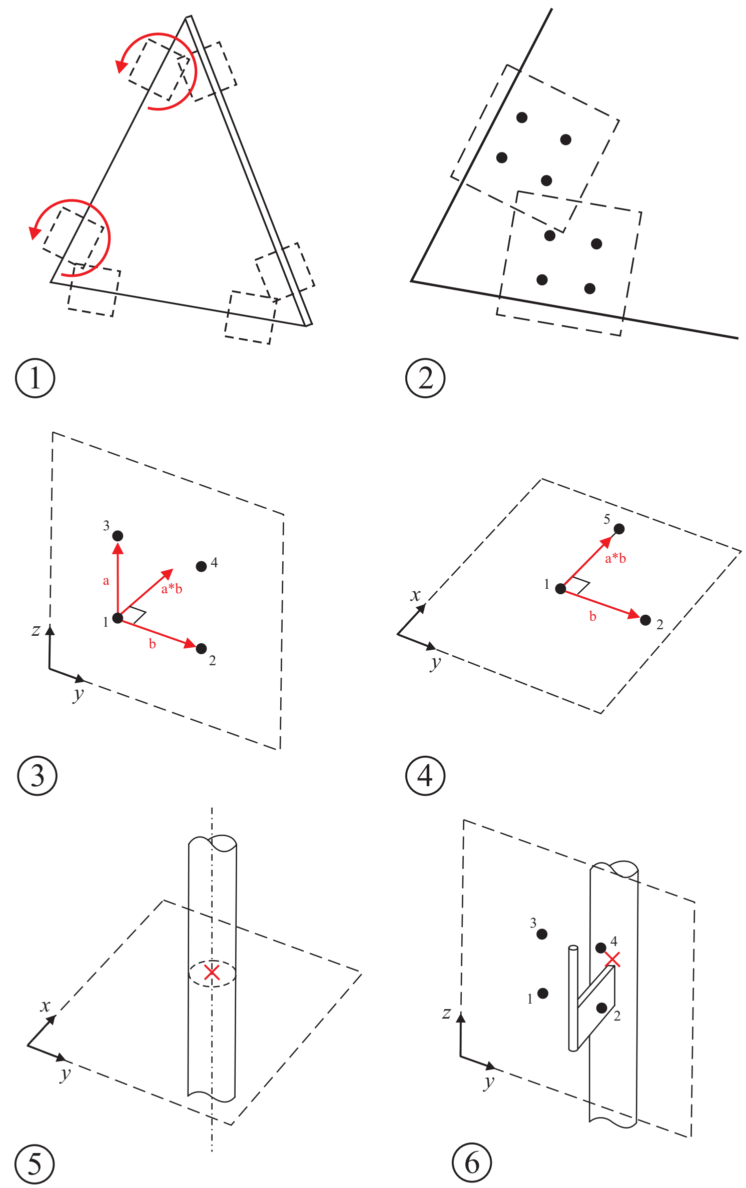


Figure 5-17 *Creating hinge pin assemblies.*

Truss Spokes

The "spokes" where the truss members meet the hubs present an interesting modeling challenge since they need to be parallel to the panel joints. The following method is used to create them:

1) Two sets of cylinders are created for each truss member. The ends of the larger cylinders coincide with the centroids of the hubs, while the ends of the smaller cylinders represent the joints between the truss members and the spokes.

2) The cylinders are exploded into their component parts, including circles at each end. Planes parallel to the ends of the cylinders are created using three points on the quadrants of each circle.

3) The truss center-lines are offset horizontally in both directions a distance slightly greater than the radii of the truss members.

4) The lines from step #3 are intersected with the planes to create new points.

5) The new points are used to create a new set of lines which can be offset and connected to create box shapes.

6) The box shapes at each end of a truss member are paired together and lofted to create spokes.

7) The spokes are trimmed by intersecting them with the hubs.

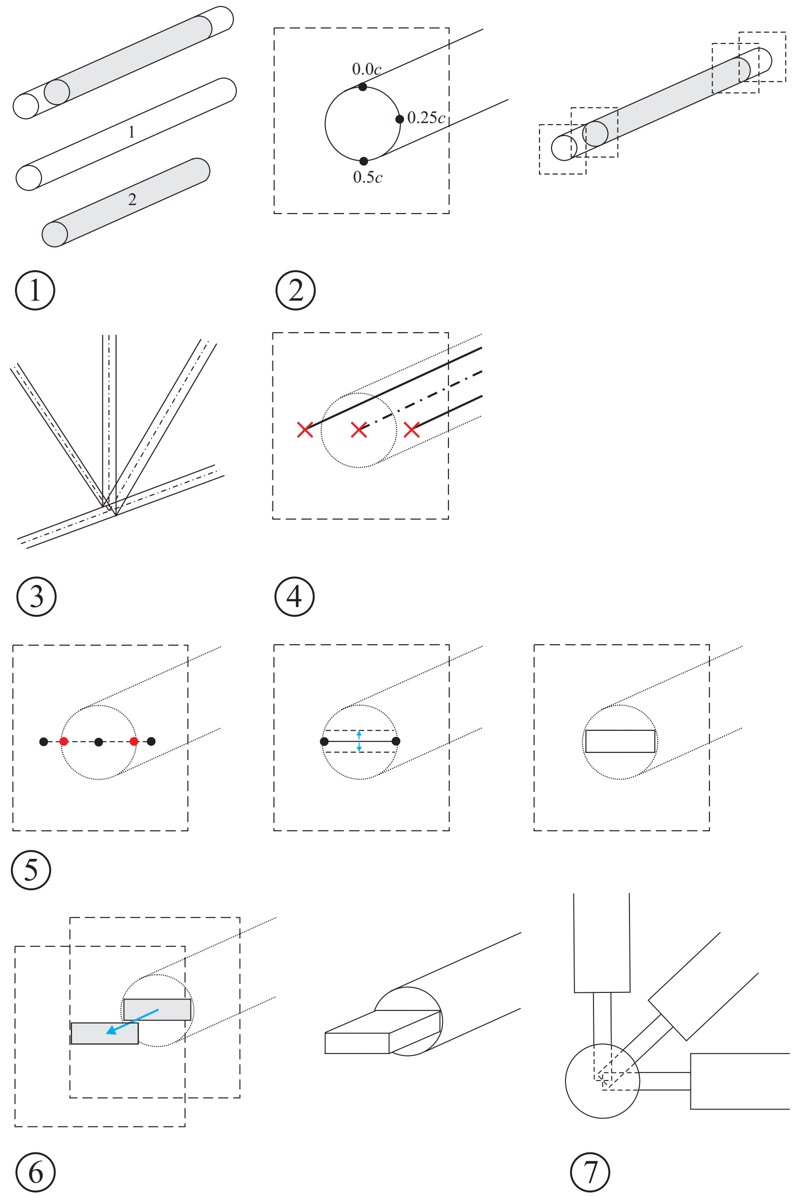


Figure 5-18 *Creating truss spokes.*

Vector Analysis

Several of the analysis modes are based on the comparison of panel normal vectors (vectors perpendicular to the centroids of the panel faces) to some reference vector, such as a vector representing due South, or a vector towards a point representing view. This explains how the vector analysis process works in more detail.

1. Simplified two-dimensional panels are created to speed up the analysis. The different panel types can be identified by symbols or the presence of holes in the case of glazed units.

2. A sorting routine removes any panels that the user doesn't want evaluated, such as opaque panels, and displays them as gray in the viewport.

3. Normal vectors for the remaining panels are created by generating vectors from the center of each panel to two of its corner points, and then finding the cross-product of those vectors.

4) If a view analysis is being performed, a vector is created from each panel centroid to the view object. If a panel orientation analysis is being performed, a reference vector is created instead (e.g. a Z-vector or South vector).

4. The angle between the two vectors is calculated and compared to the threshold angle that the user has specified.

5) The panels are filtered into two different groups depending on whether the angle is greater or less than the threshold angle. Where the angle exceeds the threshold, the panels are displayed as red, and where the angle is less than the threshold, the panels are either colored green, or colored somewhere on a spectrum that corresponds to the size of the angle.

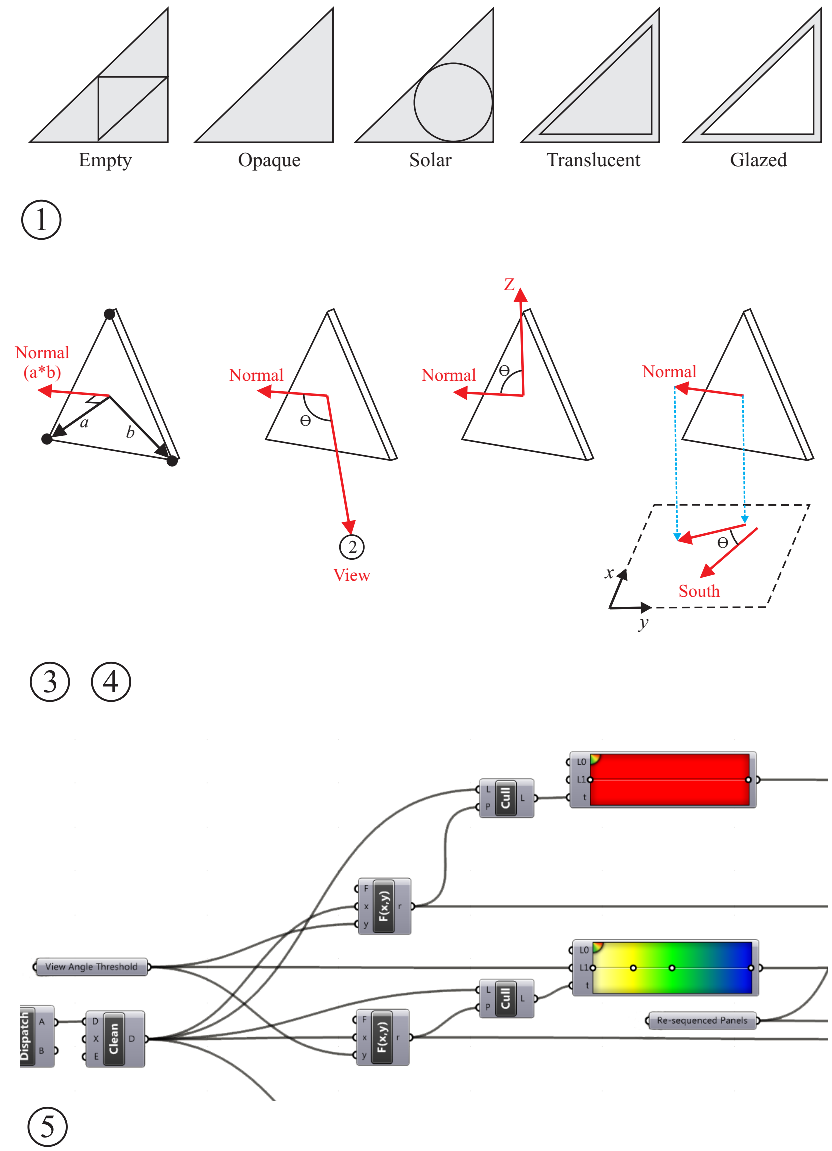


Figure 5-19 *Vector analysis*

Clash Detection - Clip Inaccessibility Warning

Clash detection mechanisms are extremely useful, if not essential at times, when parametrically modeling buildings or building assemblies. The Panel Clip Inaccessibility Warning tool in the model is a form of clash detection that determines whether or not truss members will block access to the panel clips while installing the panels. Here’s how it works:

1) Rectangles are the size of the panel clips created on the same planes on which the clips are populated.

2) The rectangles are extruded in the direction opposite to the normal vectors of each panel and then capped to make solid boxes.

3) The boxes are grouped according to the truss members they are closest to and the truss members are sequenced so that they can be grouped with the appropriate boxes.

4) Boolean “intersection” operations are attempted between each truss member and the boxes adjacent to it.

5) A “Null Item” component determines whether the boolean operation produced a result or not. If a boolean operation fails to yield a result, that means there is no conflict. If the boolean operation creates a new object, then that means there is an interference problem and a warning needs to be generated. The boxes that generated interference problems are displayed while the other boxes are hidden. Alternately, the boolean objects themselves can be displayed instead of the boxes in order to show the precise degree of interference.

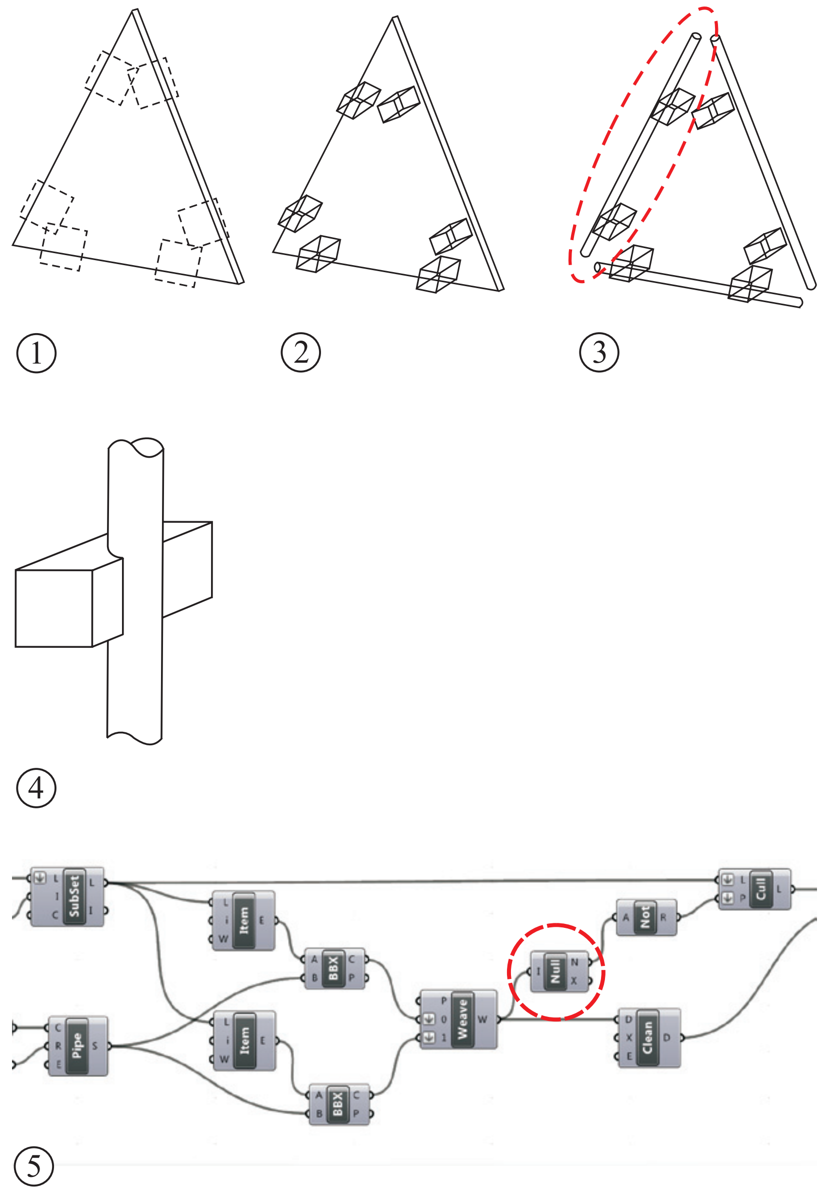


Figure 5-20 *Creating the clip inaccessibility warning mechanism.*

Datum Points and Dimensions

Because the points in each row of the model can be moved horizontally in any direction, dimensions such as floor-to-floor heights can't be calculated simply by measuring the distances between points above or below each other. The solution used here is to intersect a vertical line from one point with a horizontal plane centered on the point above it, and then to measure the distance between the first point and the new intersection point.

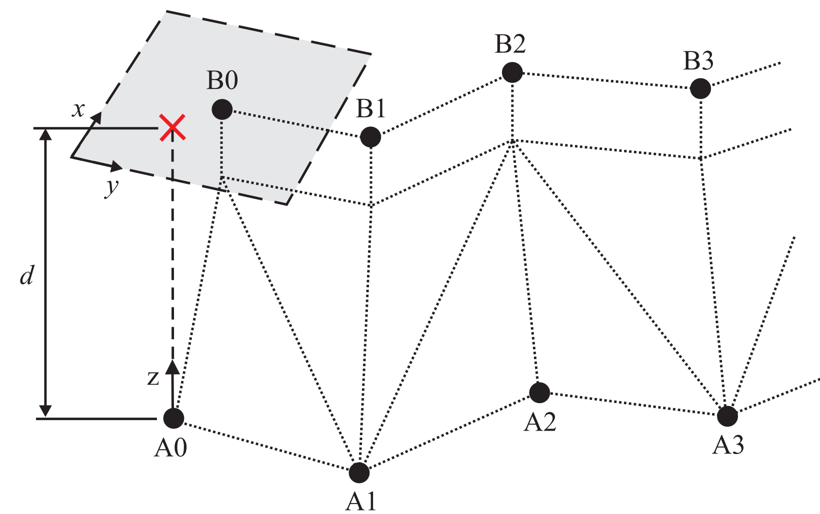


Figure 5 -21 *Evaluating floor heights.*

6. Conclusions

The process of developing the façade system and parametric model for this thesis provided a number of insights with respect to the role that parametric modeling can play as a customization and optimization tool. What follows is a discussion of some of the key issues that seem to be at stake right now for those who wish to pursue similar trajectories in design.

Opportunity Cost

Digital design and manufacturing tools have perhaps been overly-romanticized in the architectural press. These tools are indeed transforming architecture in very profound ways, but the process of transformation is slower than it may appear on the surface of things - it is more of an evolution than a revolution. The tools are still in their infancy at this point.

The potential for customization and optimization using parametric design remains great, but because modeling a building parametrically can be extremely time-consuming and requires specialized skills, the choice of whether or not to use these tools in a project is anything but a given. While most architects and architecture students by now have some familiarity with the basic concepts of parametric design and custom CNC fabrication, those who are actually engaged in these processes in the field are still effectively on the cutting edge of design and they are implicitly accepting the risk of negative pay-offs when using these tools.

Feedback Loops and Computer Processing Speeds

This thesis has emphasized the importance roll that rapid feedback loops play when using parametric modeling to optimizing designs. The reality right now, however, is that in a project of any great complexity, the designer will not receive anything close to instantaneous feedback unless the model is viewed in a simplified or partial state. Very complex models can take anywhere from seconds to minutes to fully regenerate after a change is made, assuming that they don't crash the host computer first. This is largely due to the fact that processing power in today's computers is inadequate for parametrically modeling the thousands of components in buildings, but it is also sometimes exacerbated by inefficient software design. For example, many software programs do not currently take advantage of multiple CPUs within a computer.

It's very difficult for a designer to explore the true limits of a parametric model or optimize a design when feedback loops are slow. While it is often possible to speed up the modeling workflow by creating alternate, simplified versions of the geometry (e.g. planes representing walls or panels), or by allowing the user to view only parts of the model at a time, these types of solutions reduce the amount of information the model provides, effectively dumbing-down the feedback the designer receives as design permutations are explored. Such strategies can also greatly increase the amount of work that must go into the model itself, as was discovered when creating the façade model.

One of the problems when using an array of points as an armature for an array of geometric elements is that it is difficult to maintain scalability of the system (i.e. allow for different amounts objects) without treating all points in the array as part of a single group. This means that a change to one point in the array will cause the software to recalculate all points in the array, regardless of whether or not the other points have changed position. This was the case with the Grasshopper model, and in order to eliminate the problem it was necessary to take a ground-up approach to making geometry "invisible". Instead of creating all geometry in the model and then removing some of it from view, a system was developed that allowed vertical slices of the façade to be created in isolation. This substantially reduced the long response times, but also required a lot of work to set up properly. And even with this mechanism in place, model regeneration speeds were still very slow.

Another option may have been to attempt to isolate horizontal rows of points or panels from one another so that the movement of points in one row would not affect other rows except adjacent rows. However, this was not considered an optimal approach because the geometry in every row must ultimately be processed by the same generative algorithms.

Perhaps in the future parametric software will do these types of things automatically or at least facilitate the process. For now, however, designers have to spend a lot of time finding creative ways to speed up model regeneration times, often just to make models manageable.

Different Modelers, Different Types of Flexibility

As discussed earlier, flexibility plays a key role in building customization and optimization because it allows building forms and materials to respond to the contingencies of context. But not all parametric modelers offer the

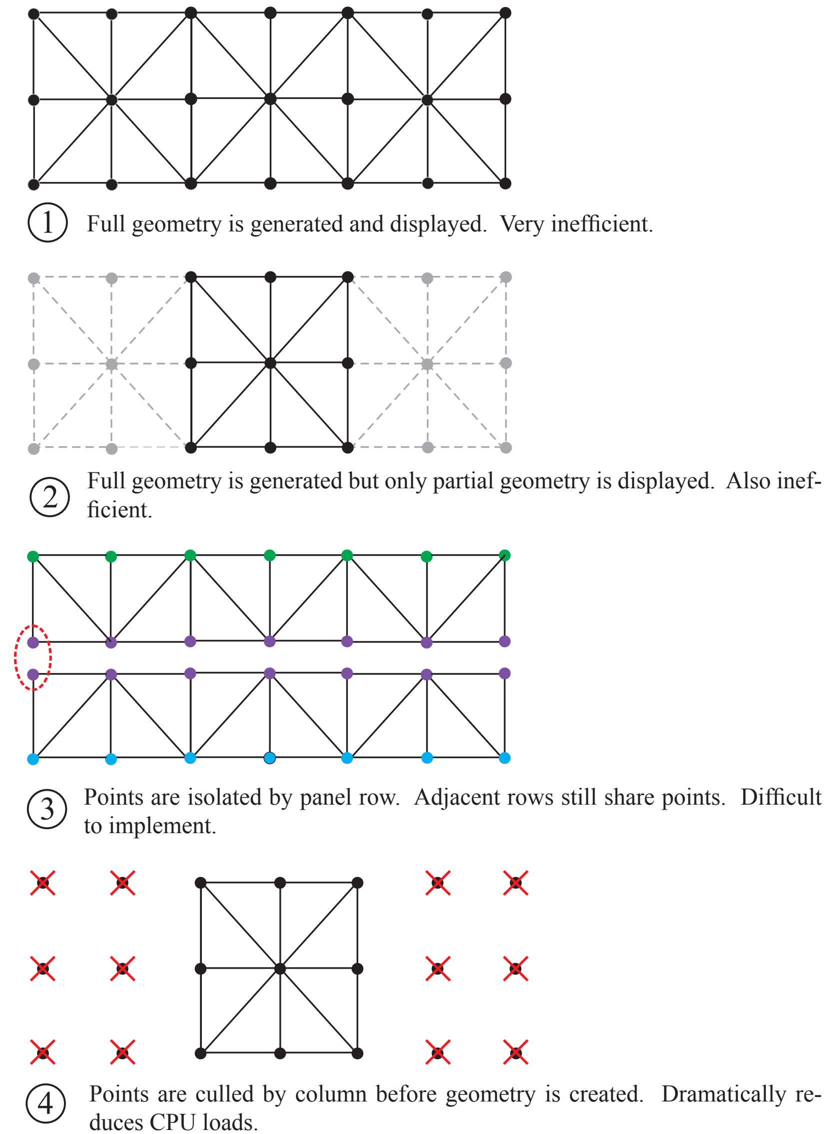


Figure 6-1 *Ways of displaying and generating partial geometry. Option 4 was chosen for the façade model.*

same types of flexibility, and the tools one chooses will greatly affect the formal possibilities and limitations of the final model.

There are only a handful of sophisticated parametric modeling programs are available for architects currently. Among these programs there two basic types: there are the algorithm-based generative modelers such as Grasshopper and Generative Components, and then there are the more hands-on modelers like Digital Project. As mentioned previously, all three of these modelers were tested as possible platforms for creating the façade model before Grasshopper was eventually chosen for the project.

Generative modelers such as Grasshopper are ideal for creating large arrays of topologically similar objects or for solving complex tiling and interpolation problems. Generative modeling generally requires a lot of preparation work or "programming" before the model can be used, but once the preparation is done, the number of components can often be scaled to any amount, within the limits of the software and hardware. One downside of a generative approach is that granular control of individual elements within an array tends to be limited and must be generally be performed within the bounds of clearly defined, pre-set rules.

At the other end of the spectrum there are programs like Digital Project which behaves much more like a conventional CAD modeler. One of the greatest strengths of Digital Project is the ability it gives users to directly modify individual components within an assembly, using either a free-form approach or using rules and constraints. Digital Project tends to shine in situations where there is a lot of very complex geometry but not necessarily universal rules - in other words, less repetition and more in the way of contingencies or spontaneity. The trade off when using Digital Project is that, while it is possible to create custom scripts within the program, it has relatively weak object propagation tools built into it. Nonetheless, the usefulness of not being constrained by the universal rules within a model can't be underestimated. No matter how flexible a building system is, there are always many instances where global rules can't be applied cleanly. Trying to create rules that account for all contingencies is, of course, impossible, and attempting to appeal to the lowest common denominator can very much work against the goals of customization and optimization.

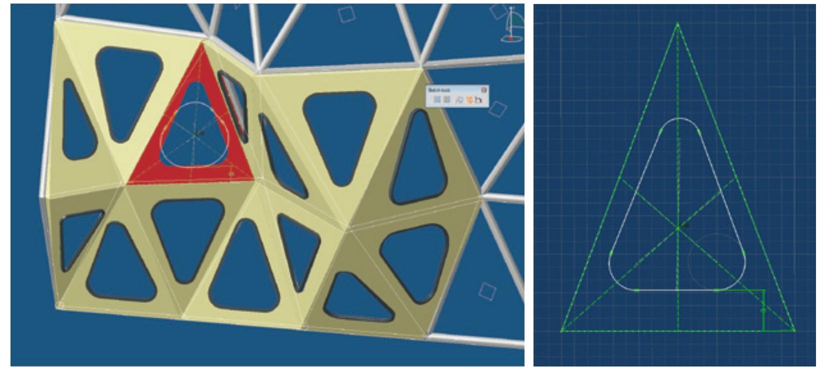


Figure 6-2 *Digital Project provides an unmatched degree of parametric control over individual objects. In the above model, drawings linked to the surface of each panel can be used to control the shape of windows individually at any time, as well as constrain the shapes according to rules.*

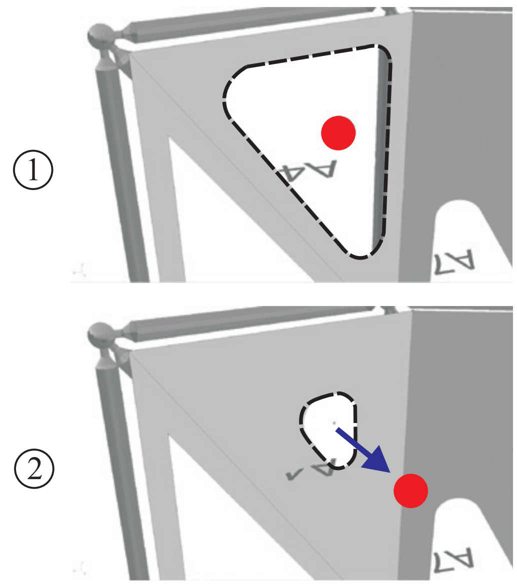


Figure 6-3 *In an early version of the Grasshopper model, some degree of hands-on granular control was added by using Rhino vertex points as "attractor" objects. The model was set up so that moving an attractor point away from a window would cause the aperture to shrink. While this approach seemed to show promise at first, difficulties were encountered when trying to use the strategy with large numbers of*

The Designer-Programmer

Modeling with Grasshopper or Generative Components can be a very painstaking and involved process. The person modeling must have some interest in - and facility with - abstract logical problem-solving, or they will lose patience with the process very quickly. Unlike programs such as Digital Project or Revit which have parametric functionality but more traditional CAD interfaces, programs like Grasshopper and Generative Components deal more directly with the software "code" that generates the geometry. Modeling in these programs is analogous to computer programming using a very high-level programming language. Each modeling task represents a series of logical problems that must be solved using a combination of mathematical and geometrical operators, variables, data sets, queries, conditional statements, and so on. Malfunctions or "bugs" will frequently occur when developing a generative model, causing a partial or complete loss of functionality, and these problems can become more and more difficult to diagnose and repair as a model increases in size and complexity due to the potentially vast interconnectedness of elements within a parametric model

A large generative model must in many ways be treated as if it were, in fact, a piece of software. First it must be made relatively "bug-free". This usually means testing every function of the model again and again, because changes in one area of a model very often have unintended consequences in another part of the model that may go undiagnosed for long periods of time. Secondly, care has to be taken so that the model runs as smoothly and efficiently as possible for whoever is going to be using it in the end. The interface must be practical and intuitive, the workflow must be smooth, and the characteristically long model regeneration times can't be exacerbated by inefficiencies in how the model is structured. And finally, because generative models are prone to bugs, great care must be taken to ensure that all the elements within the model are organized and labeled clearly and that the overall schematic structure of the model is as straightforward and legible as possible, in order to facilitate the quick diagnosis and resolution of any problems that arise.

The "programming" approach to creating geometry that generative modelers employ can be very powerful for creating large arrays of self-similar objects or for tackling complex geometrical problems, but this type of modeling either requires the aid of specialists, or requires that the designers themselves learn a completely new set of skills.

Barriers to Integration

Beyond the willingness of specialists within the building industry to collaborate with one another, the integration of design, analysis and fabrication

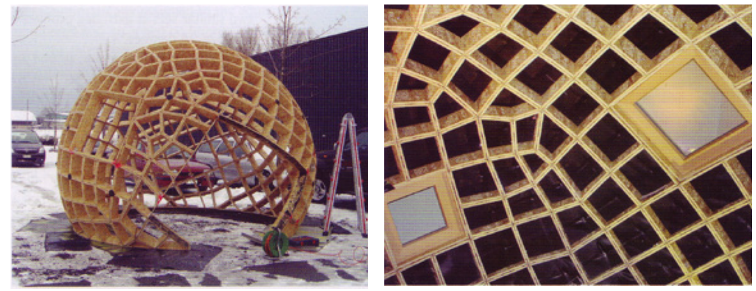


Figure 6-4 *Pavilion for the Swissbau 2005 exhibition by the Computer-aided Architectural Design group: custom generative software is used to solve a tiling problem where the sphere's structural lattice must adapt to accommodate window openings.*

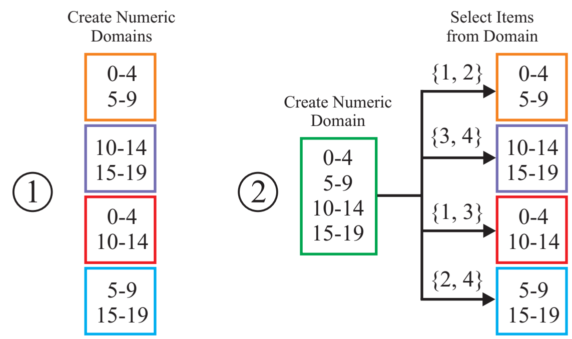


Figure 6-5 *One way to make a model more efficient is to find ways to create fewer redundant steps. For example, approach #2 above is typically more efficient than approach #1.*

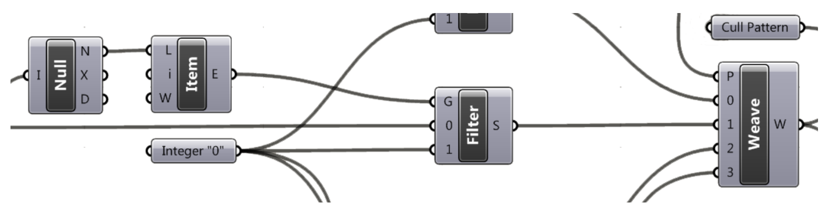


Figure 6-6 *"Programming" obstacles are encountered daily when working on a complex model. For example, the "weave" component in Grasshopper will not function if it does not receive data from one of its inputs, even if the input is not referenced in the weave pattern. A workaround is to check if inputs are producing data and to send the number "0" if they are not.*

processes ultimately relies most heavily on the effective interchange of information between specialists and the relative integration of different design and analysis software packages. Unfortunately the majority of architectural and engineering software programs available today do not exchange data effectively with one another - even packages sold by the same vendor can be notoriously poor at exchanging data with one another. The highly integrated parametric software suites that do exist, like Digital Project, currently come at a cost premium that puts them out of reach for many architectural firms.

There are some promising trends developing, however. The Rhino/Grasshopper software combination presents an affordable parametric solution for many smaller design and architecture firms. And while Grasshopper is still an esoteric program in some respects, its user interface represents a major improvement over the previous generation of generative software programs, which includes Generative Components. More and more plug-ins are now being developed that integrate directly with Grasshopper, including plug-ins for daylighting analysis, BIM, thermal analysis, and structural analysis. With the host of CAM tools already available for Rhino, there is the potential for the Rhino/Grasshopper combination to offer an affordable and fully functional suite of parametric design, analysis and fabrication tools in the not-too-distant future.

Most of the software tools used in the creation of the parametric model for this thesis were still in still development at the time they were being used. As of this writing, Grasshopper is still officially categorized as Work In Progress (WIP) software and, as such, it does not have any official technical support (though it does have excellent community and developer support through its online forums). Under these circumstances, the software generally performed well and had remarkably few bugs overall, but some of the bugs it did have posed very serious obstacles when developing the model for the thesis.

The Inherent Challenges of Customization

As software becomes more integrated this will naturally lead to more opportunities to integrate design, analysis and fabrication processes within the industry and open up the doors to more customization in architecture. But customization can still present considerable challenges in its own right. As we've seen with projects like Gehry Partners' MIT Stata center, digital tools can open up new avenues in design, but they don't automatically solve all the problems that come with the creation of novel and complex building forms.

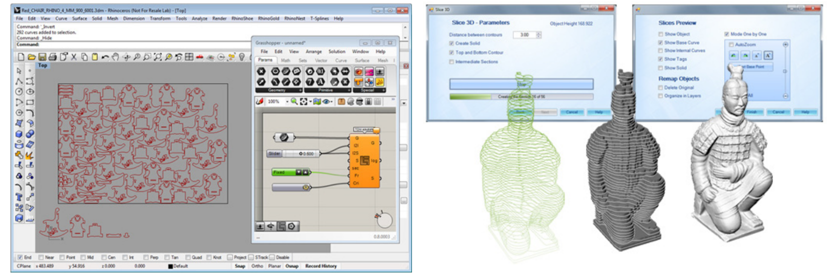


Figure 6-7 *Rhino already has many tools to assist with fabrication, including RhinoNest, a plugin that facilitates the nesting of irregular objects in order to use materials efficiently (now available for Grasshopper as well).*

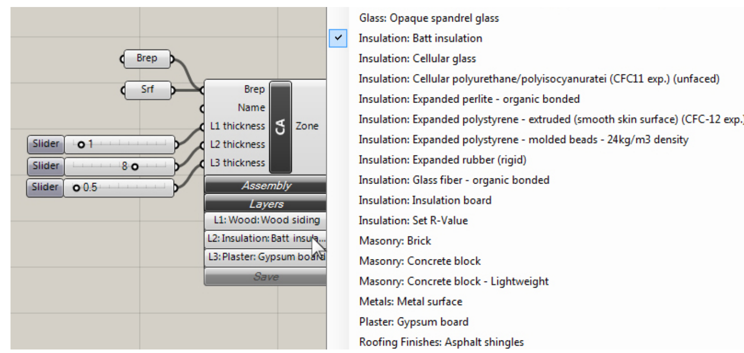


Figure 6-8 *Thermal analysis tools have recently been added to DIVA, the Rhino/Grasshopper plugin used for daylighting evaluation in the façade model,*

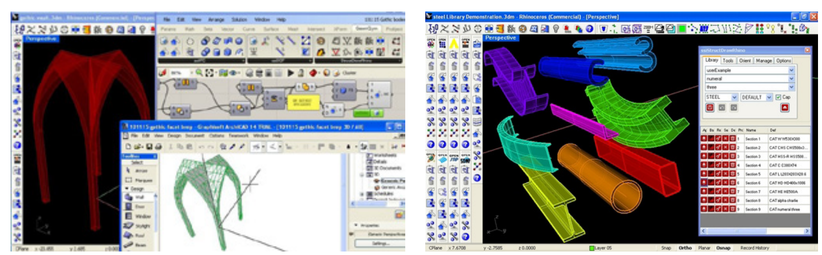


Figure 6-9 *Increasingly sophisticated analysis tools are being created for Grasshopper and Rhino, like Geometry Gym which currently offers generative structural analysis and will offer BIM in the future.*

Analysis tools are not yet sophisticated enough to automatically identify and resolve all issues relating to constructability, envelope performance, or serviceability within a project or within a custom building assembly. Careful inspection and evaluation of design details is still needed, and physical mock-ups still play an important role in design as well. In short, digital design tools do not remove the risks associated with customization – they merely give architects more opportunities to take on these types of risks.

Another point to consider is that while digital tools such as parametric design and CNC manufacturing have opened up great formal possibilities for architects, construction and assembly labor is by far the highest cost in the process of creating a building. Thus it is crucial when designing a custom building assembly that fabrication and construction processes are straightforward, repetitive, and well coordinated, or else the building will simply not be economical to construct.

Figure 6-10 *The doubly-curved glass panels Gehry Partners' Conde Nast Cafeteria required a custom glass-forming process to create. Numerous mock-ups and design revisions made by the fabricators C-Tek (glass) and TriPyramid (fasteners) were needed in order to find an elegant suspension solution that wouldn't cause the glass to fail.*[44]

The Future of Parametric Design in Architecture

Overall, it seems that parametric design will some day reach its full potential as a customization and optimization tool, but that potential will be achieved through a slow and sometimes painstaking evolution rather than a swift revolution. Fortunately, with platforms like Rhino/Grasshopper, the choice of whether or not to use of parametric tools in a given situation does not have to be an all-or-nothing one - designers can use a combination of conventional and parametric tools that's consistent with their comfort level and appropriate to the project at hand.

If current trends are any indication, parametric software will continue to become more accessible, more integrated, more streamlined, and more sophisticated over time. Improvements in computer processing power and software efficiency will ultimately allow better feedback loops for designers in the future. Over time, it is also likely that the range of flexible manufacturing processes will continue expand, which will in turn expand the possibilities for customization and optimization.

Appendix A: Model Inputs, Parameters and Statistics

1. Model Inputs

There is only one type of geometric input for the model: Rhino vertex points. These points are created by the user and can be located or moved anywhere within the Rhino model space as long as the topological relationships defined in Grasshopper (e.g. point order) remain intact.

Input points are used to control a variety of different objects, from the façade itself to building columns. Some of the sets of input points have number tags that can be enabled to allow the user to see the point order, as well as sphere markers that can be turned on to make the points more visible. The visibility of these elements is controlled under the “Component Visibility Parameters” section of the Grasshopper canvas.

<u>Input Points</u>

1. Façade Inputs Points
Rhino point rows that define the shape of the façade as well as the number of panels. The number of points per row must be consistent from row to row, and the number of rows should correspond to the number of rows of triangular panels, plus one. For example, if there are five storeys with one panel row per storey, then the number of rows would be six. If there are five stories with two rows of panels per storey, the number of rows would be eleven. Point row numbering starts with the bottom-most row.

2. View Input Points
Rhino points that specify the locations of important views in View Evaluation Mode.

3. Panel Hardware Plane Origin
A Rhino point that defines the center of the plane on which on which the panel clip and hinge objects will be defined before being instantiated onto the façade. This location is used to view and edit the shapes of the clips and hinges.

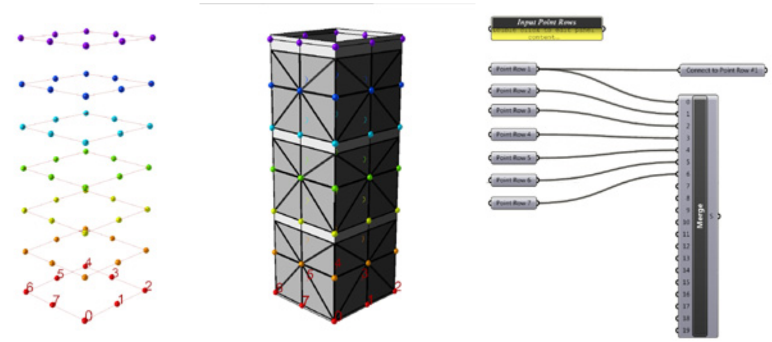


Figure A-1 *Façade input points.*

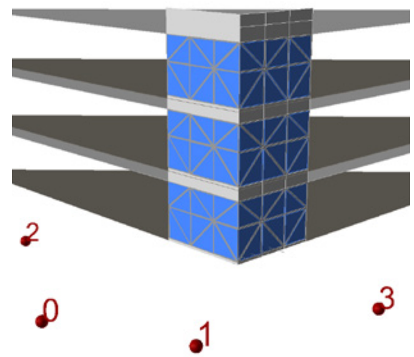


Figure A-2 *View object points.*

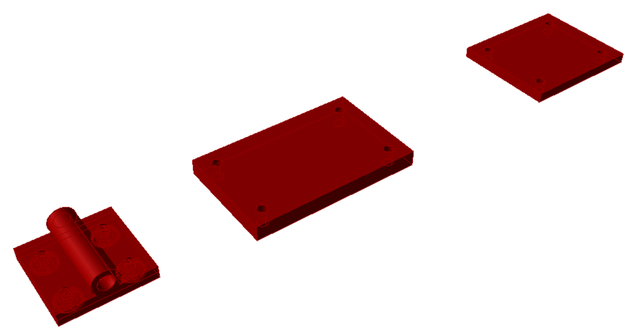


Figure A-3 *Each piece of hardware is generated on an independent plane.*

4. Panel Hardware Input Points
Rhino points that define the shapes and dimensions of both panel clips and panel hinges. The points should be symmetrical about the X and Y axes of the Panel Hardware Origin Plane.

5. Truss-to-Slab Attachment Plane Origin
A Rhino point that defines the center of the plane on which the Truss-to-Slab attachment is defined before being instantiated onto the slabs. This location is used to view and edit the attachment's properties.

6. Truss-to-Slab Attachment Input Points
Rhino points that define the shapes and dimensions of all Truss-to-Slab Attachments. The points should be symmetrical about the X and Y axes of the Truss-to-Slab Attachment Origin Plane.

7. Truss Base Plate Plane Origin
A Rhino point the defines the center point of the plane on which the Truss Hub Base Plate is defined before being instantiated onto the ground floor slab. This location is used to view and edit the plate's properties.

8. Panel Joint Drainage Trough Plane Origin
A Rhino point that indicates the center of the plane from which the panel joint drainage troughs will be instantiated.

9. Column and Core Input Points
Rhino points used to define the locations of columns and the corners of the building core. Any number of points can be chosen. These must be on the same plane as the façade input points.

10. Generic Façade Input Points
Rhino points that define the footprint of the Generic Façade used for Daylight Evaluation Mode. By default, the input points are the first row of points for the façade system. To change the input points, go to the "Input Points" section of the Grasshopper canvas, disconnect the "Generic Façade Input Points" component, then select points manually using the "Set Multiple Points" option.

11. Partition Wall Input Points
Rhino Input Points that define the shape of the partition wall, used to isolate

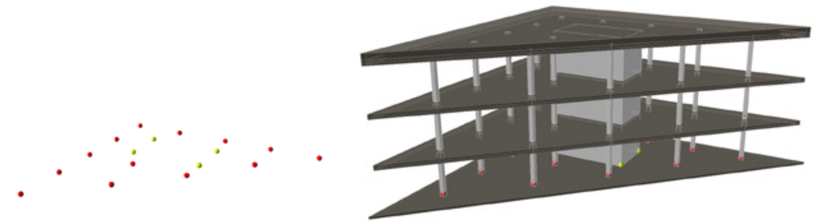


Figure A-4 *Building core and column input points*

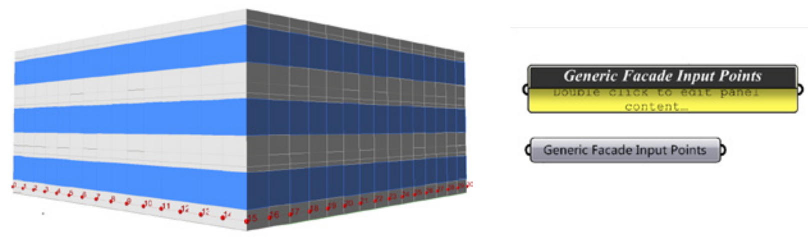


Figure A-5 *Generic Reference Façade input points*

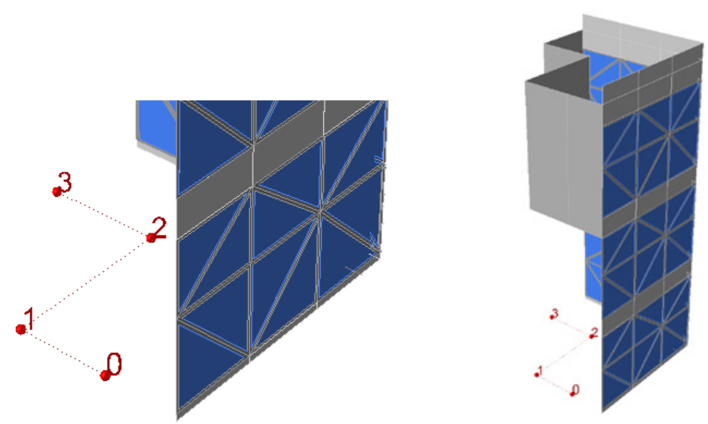


Figure A-6 *Partition wall input points* *sections of the building for analysis in Daylight Evaluation Mode when Full Façade View is turned off. The points must be on the same plane as façade input points, and the points should run in the same direction as the façade input points, with the first partition point adjacent to the first point of the partial façade view. To enable point number tags, go to the "Component Visibility Parameters section of the Grasshopper model.*

12. Ground Plane Input Points
Rhino Input Points that define the edges of the ground plane used in the Daylight Evaluation Mode.

13. North Arrow Size
Sets the length of the north arrow indicator.

2. Model Parameters

General Parameters

1. Evaluation / Modeling Mode
Turns geometry and analysis tools on or off according to the task being performed.

0 = Hardware Definitions Only
1 = All Detailed Geometry Enabled
2 = Quick Edit Mode
3 = Panel View Angle Evaluation Mode
4 = Panel Vertical Angle Evaluation Mode
5 = Solar Panel Angle Evaluation Mode
6 = Daylight Analysis Mode
7 = Panel Southern Exposure Evaluation Mode
8 = Panel Direction Evaluation Mode
9 = Daylight Evaluation Mode

2. Full Façade View On / Off
Determines whether the façade will be evaluated in its entirety or whether only a certain segment of the façade will be displayed.

2. Partial Façade View - Start Point
When working in Partial Façade View mode, this determines where the par-

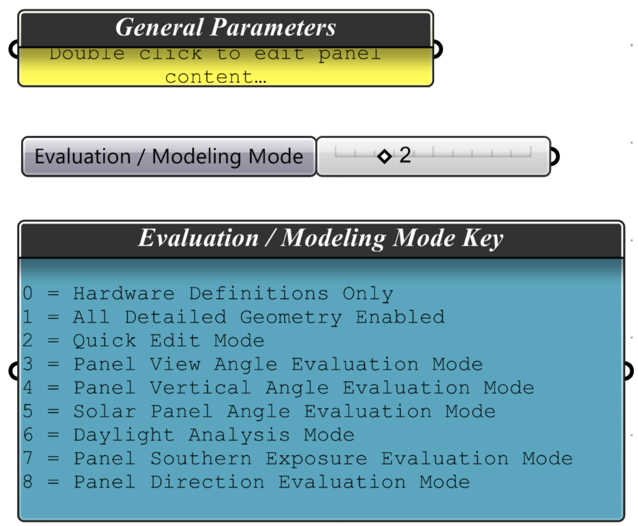


Figure A-7 *Evaluation / modeling modes* *tial view starts with respect to the point numbers on the first input point row.*

3. Partial Façade View - End Point
Sets where the Partial Façade View ends with respect to the point numbers on the first input point row.

4. Double Rows of Triangular Panels per Floor
Determines whether there are one or two rows of triangular panels per floor.

5. Triangulation Pattern
Sets the array pattern for the triangular panels.

<u>Input Point Parameters</u>

1. Number of Point Input Rows
Sets the number of rows of input points.

2. Input Point Number Text Size
Sets the size of the text displaying the index numbers for the first row of façade input points as well as the partition wall input points.

3. Façade Input Point Marker Size
Sets the radii of the spheres that mark each façade input point. These spheres can be turned on in order to make the Rhino points more visible.

4. Partition Input Point Marker Size
Sets the radii of the spheres that mark each façade input point.

<u>Empty Panel Bays</u>

1. Empty Bays
A set of numbers that indicates places on the façade where a gap in panelling is desired, such as at an entrance. The empty bay numbers must be listed in pairs corresponding to panel bay pairs (i.e. 1 and 2, 3 and 4, etc.).

<u>Panel and Bay Numbering</u>

1. Panel Number Text Size
Sets the size of the panel number text tags in the viewport. The panel num-

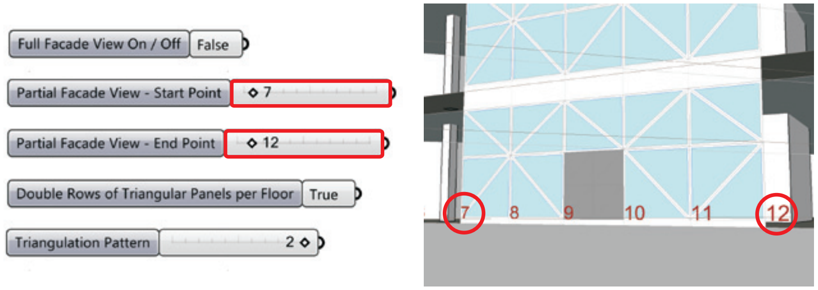


Figure A-8 *Partial Façade View and start/end points*

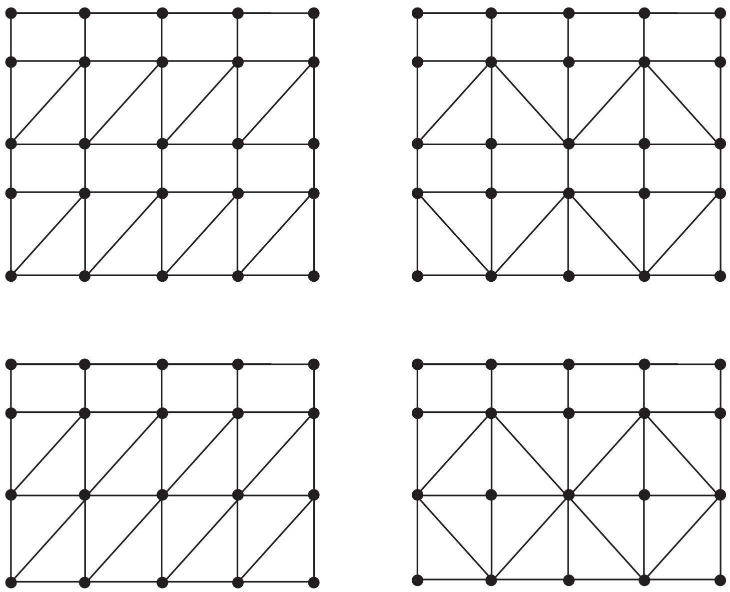


Figure A-9 *Panel triangulation pattern options* bers correspond to the actual panels and are independent of the panel bay numbers. Their positions will change depending on the number of empty panel bays.

2. Bay Number Text Size
Sets the size of the panel bay number text tags in the viewport. The panel bay positions are static.

File Information

The Grasshopper model refers to a spreadsheet file to determine the locations and types of panels and will also export model statistics to an spreadsheet file. The user can designate up to three different panel sequence files at this time.

1. Panel Sequence File Number
This slider determines which one Grasshopper will reference.

2. Panel Sequence File #1
Sets the name and location of the first panel sequence spreadsheet file.

3. Panel Sequence File #2
Sets the name and location of the second panel sequence spreadsheet file.
4. Panel Sequence File #3
Sets the name and location of the third panel sequence spreadsheet file.

5. Create Statistics File?
Outputs model statistics to a spreadsheet file. Keeping this switch on will export the data any time the model changes and lead to slow model regeneration times. Note: this feature will not work properly if any of the data being exported to the spreadsheet is "null".

6. Statistics File Number
This parameter determines which statistics file Grasshopper will output to. The spreadsheet file generated by Grasshopper ("Raw Model Statistics.ods") is not used to view the statistics directly but is referenced by another spreadsheet file called "Model Statistics Report.ods" which contains the proper labels and formatting. It is recommended that the name of the Grasshopper output file is not changed and that the file name/location input

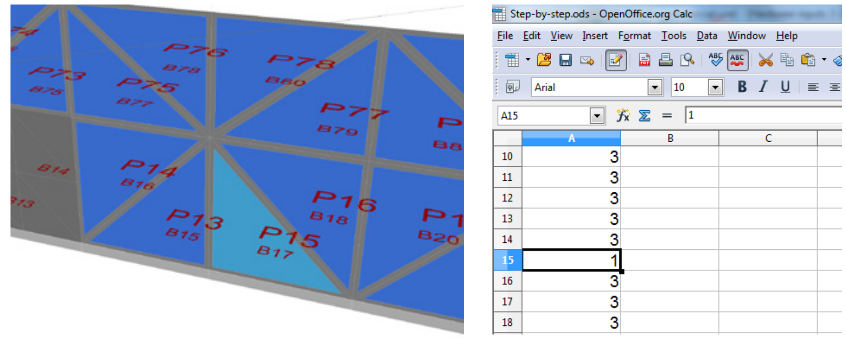


Figure A-10 *Panel and bay numbering*

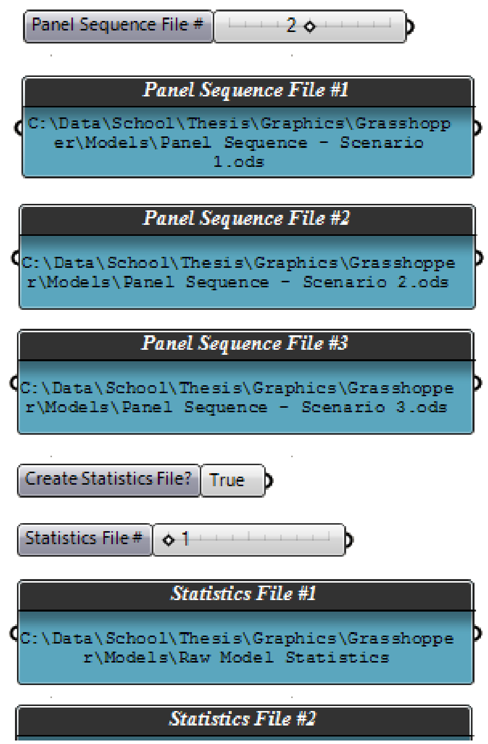


Figure A-11 *File Information*

fields below are used only to specify different file locations (e.g. different computers being used).

7. Statistics File #1
Sets the name and/or location of the first statistics spreadsheet file.

8. Statistics File #2
Sets the name and location of the second statistics spreadsheet file.

9. Statistics File #3
Sets the name and location of the third statistics spreadsheet file.

Analysis Parameters - General

1. Translucent Panels On / Off
Turns the translucent panels on or off for evaluation. On/Off toggles do not apply to Daylight Analysis mode. Turned-off panels display as gray.

2. Opaque Panels On / Off
Turns the opaque panels on or off for evaluation.

3. Glazed Panels On / Off
Turns the glazed panels on or off for evaluation.

4. Solar Panels On / Off
Turns the solar panels on or off for evaluation.

5. North Angle Correction
Sets the angle of true North relative to the model's Y-axis for vector analysis and can be used to reverse the direction of the sun for Panel Orientation Evaluation Mode, Southern Exposure Evaluation Mode, or Solar Panel Evaluation Mode for buildings in the Southern Hemisphere. The parameter can also be used to change the angle of the sun relative to the world coordinate system in Rhino, however it should be noted that Radiance evaluations assume that South corresponds to -X in the model's coordinate system by default.

Building Statistics – General	
Full Facade View (1) or Partial Facade View (0)	0
Partial Facade - Start Point	0
Partial Facade - End Point	33
Building Height	11.23
Number of Floors	2
Average Floor Height	5
Partitioned Floor Area	0
Total Floor Area	3231.5
Building Surface Area	357.14
Building Volume	0
Surface to Volume Ratio	0
Window to Wall Ratio	0.79
Window to Wall Ratio w/ Translucency	0.79
Translucency Factor	0.2

Panel Statistics – General	
Number of Translucent Panels	0
Number of Opaque Panels	0
Number of Glazed Panels	0
Number of Solar Panels	0
Total Number of Panels	526

Thermal Resistance Statistics	
Translucent Panel R-Value	3
Opaque Panel R-Value	5
Glazed Panel R-Value	2
Solar Panel R-Value	5
Spandrel Panel R- Value	5
Average R-Value of Building Envelope	2.62

Panel Costing Statistics	
Translucent Panel Cost / Sq.m	100
Opaque Panel Cost / Sq.m	100
Glazed Panel Cost / Sq.m	100
Solar Panel Cost / Sq.m	100
Spandrel Panel Cost / Sq.m	100
Total Translucent Panel Cost	0
Total Opaque Panel Cost	0
Total Glazed Panel Cost	68027
Total Solar Panel Cost	0
Total Spandrel Panel Cost	17007
Total Cost of Trangular Panels	68027
Average Cost per Triangular Panel	257.68
Total Panel Costs	92985

Figure A-12 *Model Statistics Report, Page 1*

View Angle Evaluation Parameters

1. Panel View Angle Threshold
Specifies the angle from a panel's normal (perpendicular) within which view objects are considered to be visible.

2. View Object to Evaluate
Specifies which view object point to use for the view evaluation.

3. Panel View Vector Line Visible
This function allows the user to see a line from the centroid of an individual panel to the selected view object. This can be used to fine-tune panel alignment or to verify that the view is unobstructed.

4. Panel Number for View Vector Evaluation
Specifies the panel from which the vector line will be drawn.

5. Disable Non-Glazed Panels During Evaluation
Specifies whether or not to automatically disable all panels other than glazed panels during view evaluation.

6. View Object Sphere Size
Sets the size of the spheres representing the view object. The centers of the spheres are the Rhino input points for the view objects.

7. View Vector Line Thickness
Sets the thickness of the vector line when viewing an individual panel vector.
8. View Object Text Size
Sets the size of the number tag for each view object.

Vertical Angle Evaluation Parameters

1. Panel Maximum Angle from Vertical
Sets the maximum angle relative to vertical for panels.

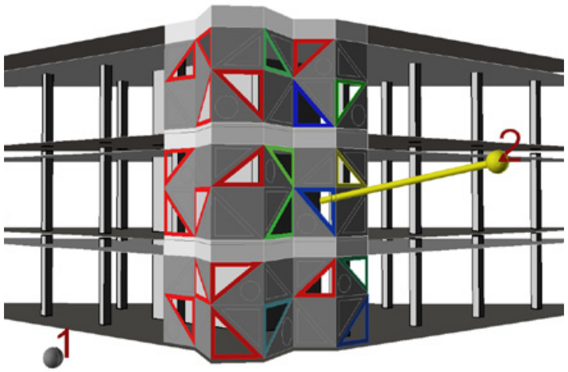


Figure A-13 *View Angle Evaluation Mode with visible view vector*

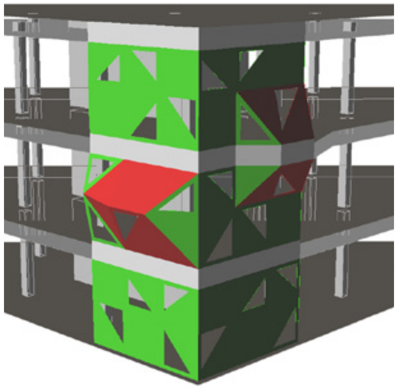


Figure A-14 *Vertical Angle Evaluation Mode*

Solar Panel Angle Evaluation Parameters

1. Solar Panel Optimal Vertical Angle
Sets the optimal vertical angle for solar panels

2. Solar Panel Vertical Angle Threshold
Sets the maximum angle deviation from the optimal vertical angle for solar panels

3. Solar Panel South-Facing Angle Threshold
Sets the maximum horizontal deviation from due south for solar panels.

4. Disable All Non-Solar Panels During Evaluation
Specifies whether or not to automatically disable all panels other than solar panels during solar panel angle evaluation.

Southern Exposure Evaluation Parameters

1. Southern Exposure Angle Threshold
Sets the acceptable deviation from due South for a panel

Panel Orientation Evaluation Parameters

1. Southern Exposure Angle Threshold
Sets the acceptable deviation from due South for a panel

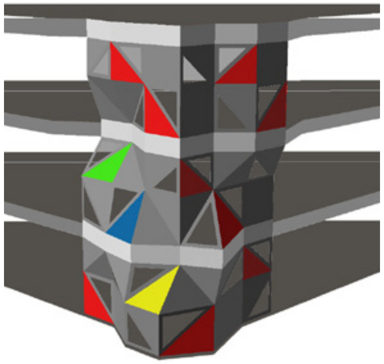


Figure A-15 *Solar Panel Angle Evaluation Mode*

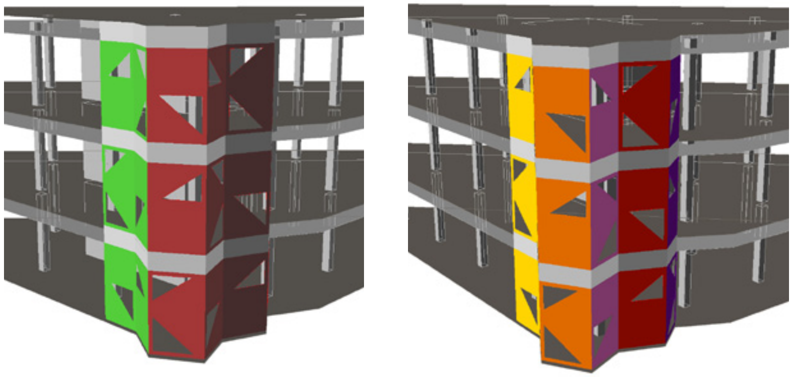


Figure A-16 *Southern Exposure Evaluation Mode and Panel Orientation Evaluation Mode*

Daylight Evaluation Parameters

1. Diva Evaluation On
Turning this on will begin the Diva Radiance / Daysim evaluation.

2. Show Nodes Only / Hide Geometry
Enabling this switch will hide all geometry other than the analysis nodes and the text (analysis results) attached to those nodes.

3. Floor to Evaluate
Specifies which floor of the building will be evaluated. All other geometry will be turned off when calculations are being done.

4. Generate Analysis Nodes
Turning this on generates a grid of nodes for daylight analysis at a specific distance from the floor slab.

5. Evaluation Metric
Specifies the type of evaluation Diva will do.

0 = Visualization
1 = Radiation Map
2 = Radiation Nodes
3 = Illuminance
4 = Daylight Factor
5 = Daylight Autonomy

6. Use Panel Centroids As Nodes
Instead of creating an analysis grid offset from the floor, this uses points at the centroids of each panel to create the node array. These points can be used in a radiation map analysis to see approximately how much sun each individual panel will receive.

7. Use Centroids of Solar Panels Only
When using panel centroids for the node array, this option allows only solar panel centroids to be used for the array to speed up calculations when evaluating solar panel orientations.

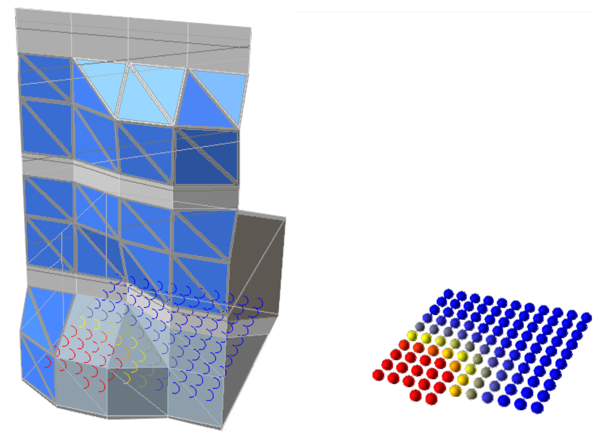


Figure A-17 *Daylight evaluation.*

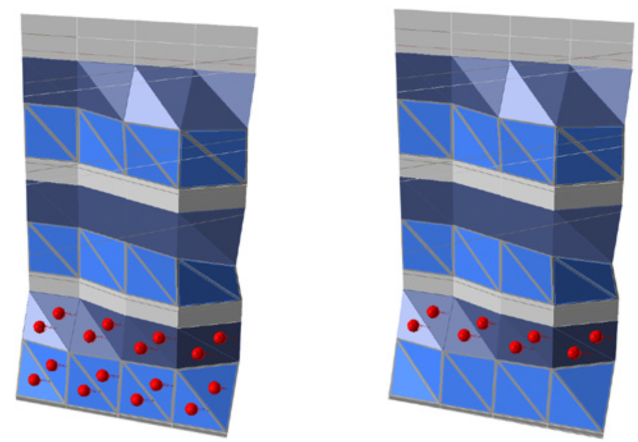


Figure A-18 *Panel centroids used for radiation map analysis*

8. Evaluation Grid Horizontal Offset
Specifies the minimum distance of analysis nodes from the walls and façade.

9. Evaluation Grid Vertical Offset
Specifies the distance between the floor slab and the analysis grid.

10. Evaluation Node Count (U)
Specifies the the amount of evaluation nodes in the U-direction.

11. Evaluation Node Count (V)
Specifies the the amount of evaluation nodes in the V-direction.

12. Evaluation Grid Node Marker Size
Sets the size of the spheres that mark the evaluation grid nodes.

13. Evaluation Grid Node Text Size
Sets the size of the text representing analysis results.

14. Evaluation Grid Density
Specifies the density of the Diva evaluation grid. Higher densities are more accurate but require more time to calculate.

15. Daylight Factor Threshold
Sets the minimum Daylight Factor target percentage. The percentage of evaluation nodes that exceed this number will be displayed in the statistics section of the model.

16. Daylight Autonomy Threshold
Sets the minimum Daylight Autonomy percentage target. The percentage of evaluation nodes that exceed this number will be displayed in the statistics section of the model.

17. Translucent Panel Material
Specifies the DIVA material type for all translucent panels.

18. Glazed Panel Material
Specifies the DIVA material type for all glazed panels.

19. Partition Material
Specifies the DIVA material type for the partition walls created by the partition input points.

20. Floor Material
Specifies the DIVA material type for the floors.

21. Ceiling Material
Specifies the DIVA material type for the ceilings.

22. Ground Material
Specifies the DIVA material type for the ground plane.

23. Light Shelf Material
Specifies the DIVA material type for the light shelves.

24. Generic Façade On
Replaces the façade with a generic façade which can be analyzed for comparison purposes.

25. Generic Façade Window Height
Specifies the window heights for the generic façade.

26. Generic Façade Window Sill Height
Specifies the window sill heights for the generic façade.

<u>WWR and Thermal Resistance Parameters</u>

1. Translucency Factor for WWR
Sets the translucency of the translucent panels for the purpose of Window to Wall Ratio calculations.

2. Translucent Panel R-Value
Sets the R-Value of the translucent panels for thermal resistance calculations.

3. Opaque Panel R-Value
Sets the R-Value of the opaque panels for thermal resistance calculations.

4. Glazed Panel R-Value
Sets the R-Value of the glazed panels for thermal resistance calculations.

5. Solar Panel R-Value
Sets the R-Value of the solar panels for thermal resistance calculations.

6. Spandrel Panel R-Value
Sets the R-Value of the spandrel panels for thermal resistance calculations.

<u>Panel Costing</u>

These parameters are used to estimate panel costs. The results of the costing calculations are displayed in the statistics section of the model under "Viewed Panel Areas and Costing".

1. Translucent Panel Cost per Square Meter
Estimate of the cost per square meter of translucent panels.

2. Opaque Panel Cost per Square Meter
Estimate of the cost per square meter of opaque panels.

3. Glazed Panel Cost per Square Meter
Estimate of the cost per square meter of glazed panels.

4. Solar Panel Cost per Square Meter
Estimate of the cost per square meter of solar panels.

5. Spandrel Panel Cost per Square Meter
Estimate of the cost per square meter of solar panels.

Panel Edge Length Warning

This function allows the user to see when a panel edge exceeds a particular dimension by generating a pipe object along the edge of the panel colored red by default.

1. Panel Edge Length Warning On / Off
Turns the warning system on or off. The system will work in any evaluation mode except "Hardware Definitions Only".

2. Maximum Panel Edge Length
Sets the maximum length a panel edge can be before a warning appears.

3. Edge Length Warning Line Thickness
Sets the thickness of the pipe object used to warn the user.

Panel Joint Angle Warning

This function allows the user to determine whether the angles between adjacent panels exceed a certain threshold.

1. Horizontal Warning On/Off
Turns the warning system for horizontal angles on or off.
2. Vertical Warning On/Off
Turns the warning system for vertical angles on or off.

3. Horizontal Angle Limit
Determines the angle threshold at which a warning indicator will appear for horizontal joints. The warning will occur when the joint angle is less than the set threshold.

4. Vertical Warning On/Off
Determines the angle threshold at which a warning indicator will appear for vertical joints. The warning will occur when the joint angle is less than the set threshold.

5. Joint Angle Warning Line Radius
Sets the radius of the pipe object that will appear at the panel joint when an angle threshold is crossed.

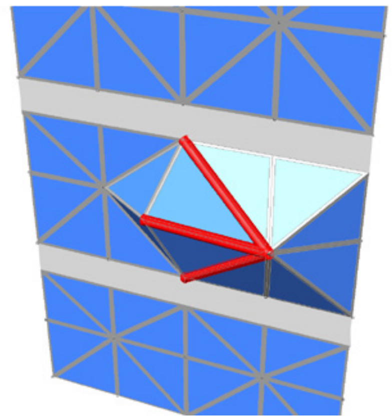


Figure A-19 *Panel Edge Length Warning Mode*

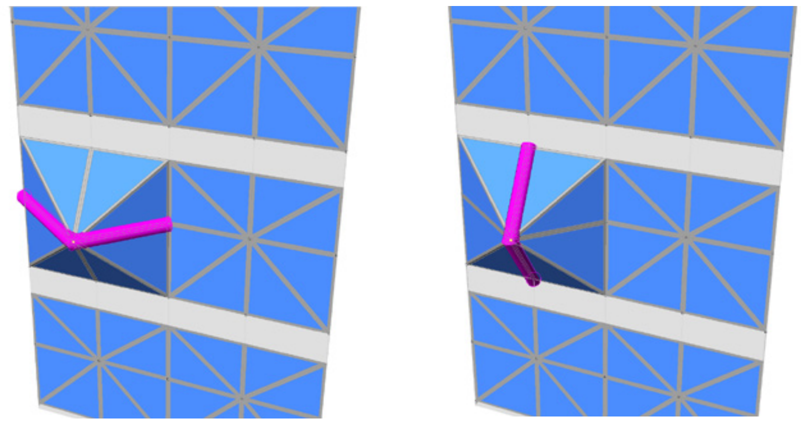


Figure A-20 *Panel Joint Angle Warning Modes*

Panel Clip Inaccessibility Warning

This function allows the user to determine whether truss members are obstructing access to panel clip bolts.

1. Panel Clip Inaccessibility Warning On / Off
Turns the warning system on or off. The system will only work in the "All Detailed Geometry Enabled" evaluation mode.

2. Panel Clip Inaccessibility Warning Mode
Chooses one of two monitoring modes:

1 = Use Boxes to Indicate Clips with Poor Accessibility
2 = Only Show Intersections between Boxes and Truss Members

3. Warning Box Length Factor
Sets the lengths of the warning boxes as a multiple of the truss offset plus the truss radius. Increasing the value will make the boxes more visible.

Component Visibility

These toggles allow the user to control the visibility of most geometry. Visibility of the following elements can be controlled:

1. Panel Visibility
2. Panel Clip / Hinge Visibility
3. Panel Clip Bolt / Clip Spacer Visibility
4. Spandrel Visibility
5. Spandrel Clip / Hinge Visibility
6. Spandrel Clip Bolt / Clip Spacer Visibility
7. Slab Visibility
8. Truss Visibility
9. Column Visibility
10. Floor / Façade-Slab Gap Cover Visibility
11. Parapet Cap / Façade Base Visibility
12. Ceiling Visibility
13. Light Shelf Visibility
14. Partition Wall Visibility
15. Ground Plane Visibility

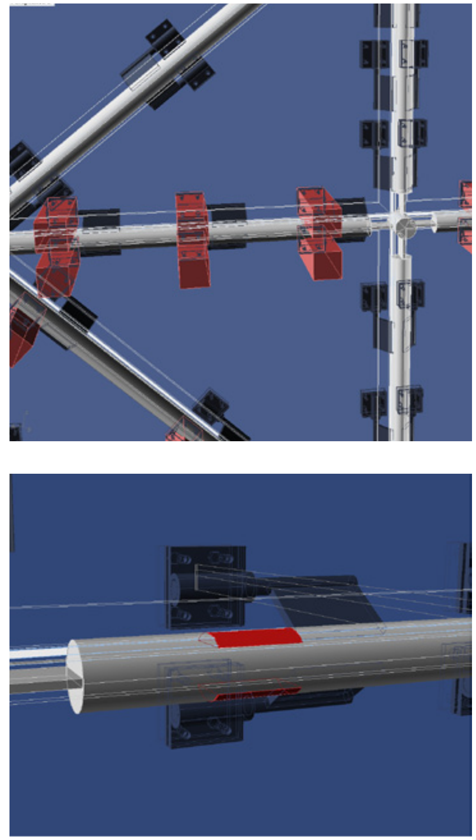


Figure A-21 *Panel Clip Inaccessibility Warning Mode*

Text / Object-Marker Visibility

Text and helper objects are available to assist with certain modeling tasks, including identifying panel numbers for assigning panel types.

1. Panel / Bay Number Visibility
2. Light Shelf Number Visibility
3. Input Point Number Text Visibility
4. Partition Wall Point Number Visibility
5. View Object Sphere Visibility
6. View Object Text Visibility
7. Façade Input Point Marker Visibility
8. Partition Input Point Marker Visibility
9. Ground Plane Visibility
10. Ground Plane Point Number Visibility
11. Ground Plane Point Marker Visibility
12. North Arrow Visibility

Component Enable / Disable

Detailed geometric elements such as truss members and panel hinges are automatically disabled in all evaluation modes except All Detailed Geometry Enabled mode. When mode this mode active, the following switches allow elements allow certain elements to be turned on and off independently:

1. Panels Enabled
2. Panel Clips / Hinges Enabled
3. Spandrels Enabled
4. Spandrel Clips / Hinges Enabled
5. Clip Spacers / Bolts Enabled
6. Slabs Enabled
7. Truss Enabled
8. Parapet / Façade Base Enabled
9. Columns / Building Core Enabled
10. Ceiling / Façade-Slab Gap Cover Enabled
11. Light Shelves Enabled
12. Disable Panel Clip Hinge Booleans

Disabling the boolean subtraction operations between hinge elements dra-

matically reduces the amount of time it takes to generate and view the panel clip assemblies.

General Panel Parameters

1. Outer Pane Spacing
Sets the width of the gaps between all panels, including spandrel panels.

2. Inner Pane Spacing
Sets the width of the gaps between all panels, including spandrel panels.

3. Horizontal Panel Joint Drain Angle
Controls the angle of all horizontal panel joints for drainage purposes.

4. Panel Clip Edge Offset
Sets the distance between the edges of the panels to the panel clips.

5. Panel Clip Corner Offset
Sets the distances from the panel corners to the nearest clip on each edge.

6. Number of Clips per Panel Edge
Sets the number of clips on each panel edge for the triangular panels.

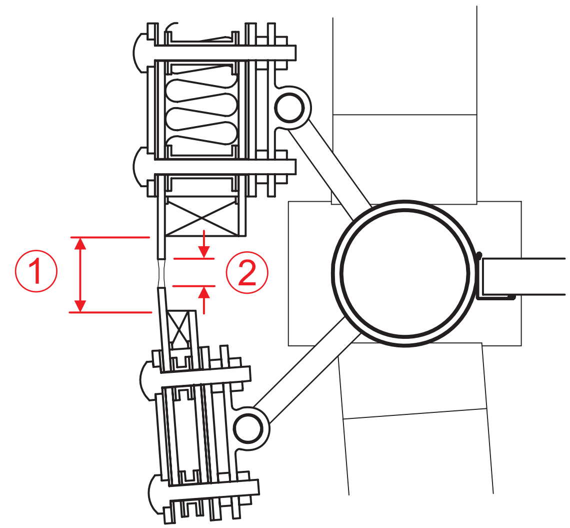
7. Panel Spacer Widths
Controls the widths of the panel spacers around the edges of the panels. This width should be less than the clip offset distance.

Spandrel Panel Parameters

1. Spandrel Panel Height
Sets the height of the spandrel panels as well as the depth of the ceiling cavities.

2. Spandrel Clip Edge Offset
Sets the distance from the edge of the panels to the panel clips.

3. Spandrel Clip Corner Offset
Sets the horizontal distance from the panel corners to the nearest clips.



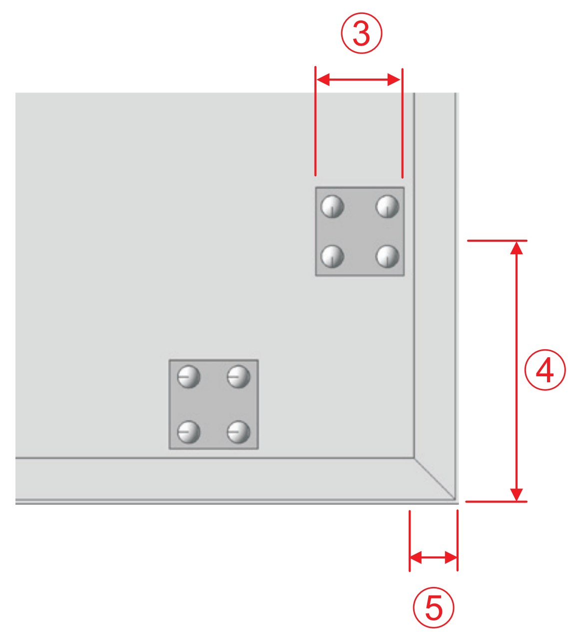


Figure A-22 *General panel parameters*

4. Number of Panel Clips per Edge
Sets the number of clips on each panel edge of the spandrel panels.

5. Panel Clip Corner Offset
Controls the distance between the corners of the panel and the centroids and the closest panel clips.

6. Spandrel Panel Spacer Thickness
Controls the thickness of the spacer and the gap between the inner and outer panes.

7. Spandrel Panel Pane 1 Thickness
Controls the outer pane's thickness.

8. Spandrel Panel Pane 2 Thickness
Controls the inner pane's thickness

9. Spandrel Panel Spacer Widths
Controls the widths of the panel spacers around the edges of the panels. This width should be less than the clip offset distance.

Panel Parameters by Type

These parameters control the thicknesses of the various triangular panel elements.

1. Translucent Panel Spacer Thickness
2. Translucent Panel Pane 1 Thickness
3. Translucent Panel Pane 2 Thickness
4. Opaque Panel Spacer Thickness
5. Opaque Panel Pane 1 Thickness
6. Opaque Panel Pane 2 Thickness
7. Glazed Panel Spacer Thickness
8. Glazed Panel Pane 1 Thickness
9. Glazed Panel Pane 2 Thickness
10. Solar Panel Spacer Thickness
11. Solar Panel Pane 1 Thickness
12. Solar Panel Pane 2 Thickness

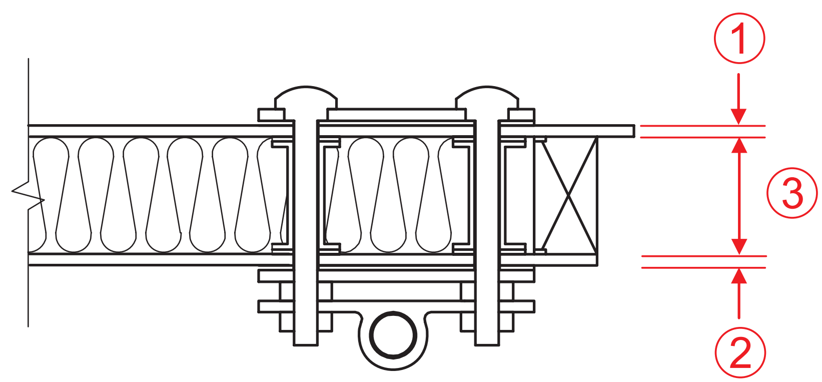


Figure A-23 *Panel parameters by type*

Panel Clip Parameters

1, Panel Clip Plate Thickness
Controls the thickness of the steel plate.

2. Panel Clip Plate Bolt Hole Radius
Controls the radii of the bolt holes.

3. Panel Clip Bolt Hole Offset 1
Controls the offset of the first set of bolt holes.

4. Panel Clip Bolt Hole Offset 2
Controls the offset of the second set of bolt holes.

5. Panel Clip Bolt Radius
Sets the radii of the clip bolt shafts.

6. Panel Clip Bolt Nut Thickness
Sets the thickness of the clip bolt nuts.
This dimension also determines the offset of the hinge plate.

7. Panel Clip Bolt Nut Radius
Controls the width of the clip bolt nuts.

8. Panel Clip Bolt Head Thickness
Controls the thickness of the bolt heads.

9. Panel Clip Bolt Head Radius
Controls the curvature of the bolt heads.

10. Panel Clip Bolt Head Nut Radius
Sets the radius of the nut integrated into the bolt head.

11. Panel Clip Bolt Protrusion Amount
Controls the excess length of the bolt.

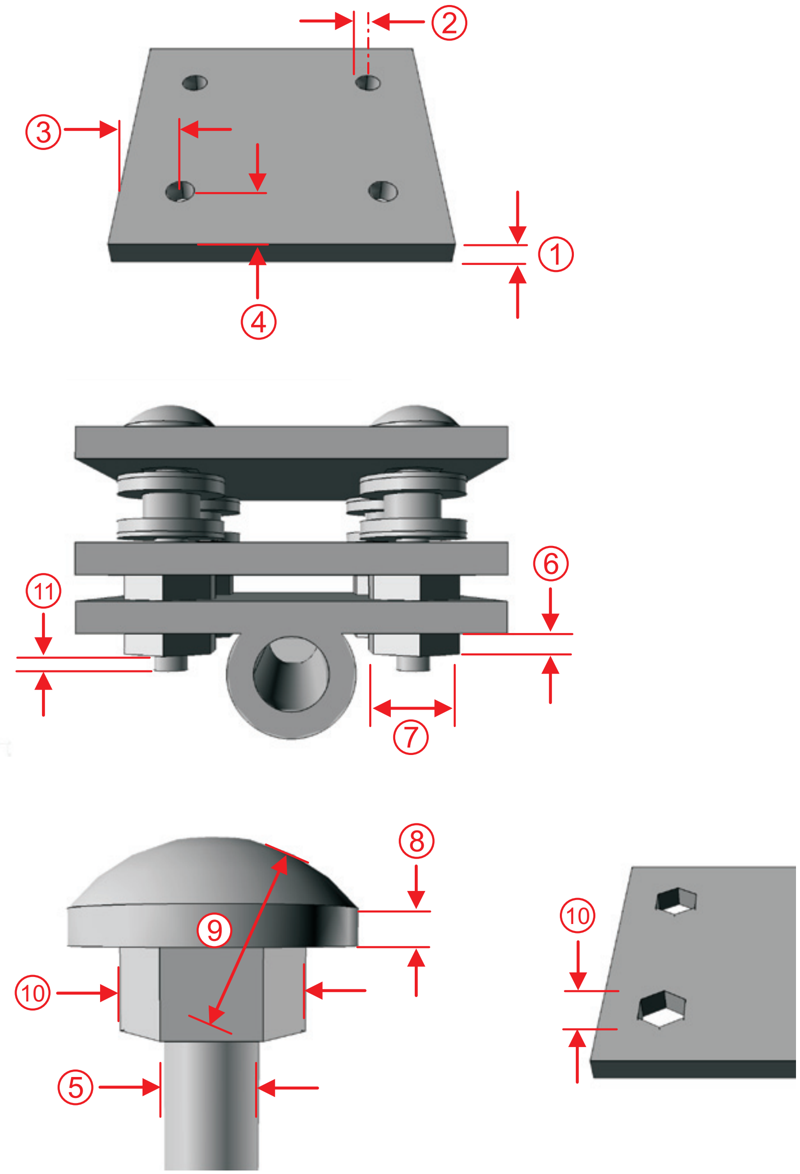


Figure A-2 *Panel clip parameters*

Panel Clip Spacer Parameters

1. Panel Clip Spacer Gasket Thickness
Controls the thickness of the gaskets between the panel clip spacers and the panel panes.

2. Panel Clip Spacer End Thickness
Controls the thickness of the ends of the panel clip spacer components.

3. Panel Clip Spacer End Radius
Controls the radii of the ends of the panel clip spacer components.

4. Panel Clip Spacer Tube Thickness
Controls the thickness of the tube sections of the panel clip spacer components.

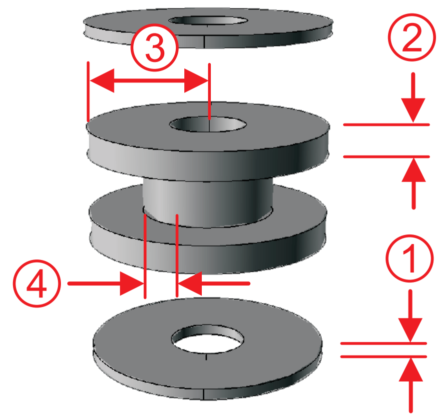


Figure A-25 *Panel clip spacer parameters*

Panel Hinge Parameters

1. Hinge Clip Thickness
Sets the thickness of the hinge clip that attaches to the panel clips.

2. Hinge Clip Bolt Hole Radius
Sets the radius of the bolt holes. Increased hole size is used to accommodates minor construction tolerances.

3. Hinge Pin Radius
Sets the radius of the hinge pin.

4. Hinge Cradle Radius
Sets the radius of the cradle the hinge pin fits into on the hinge clip.

5. Hinge Cradle Center Offset
Sets the distance from the hinge clip bottom to the center of the hinge cradle.

6. Hinge Cradle Wall Thickness
Sets the material thickness of the cradle.

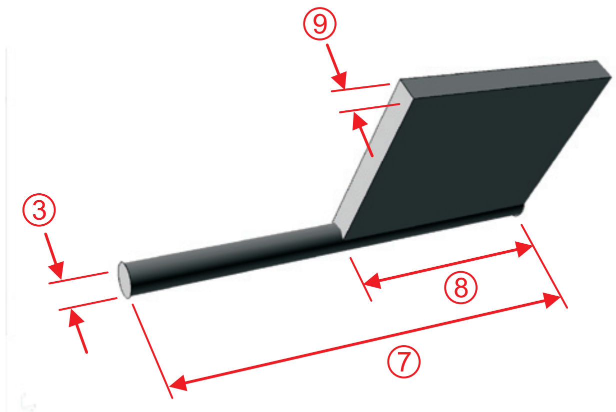
7. Hinge Pin Length - Panels
Controls the length of the hinge pin for triangular panels.

8. Hinge Plate Length - Panels
Controls the length of the plate connecting the hinge pin to the truss for triangular panels.

9. Hinge Plate Thickness - Panels
Controls the hinge plate's thickness for the triangular panels.

10. Hinge Pin Length - Spandrels
Controls the hinge pin's length for spandrel panels.

11. Panel Hinge Plate Length - Spandrels
Controls the length of the plates connecting the pins to the the truss for spandrel panels.



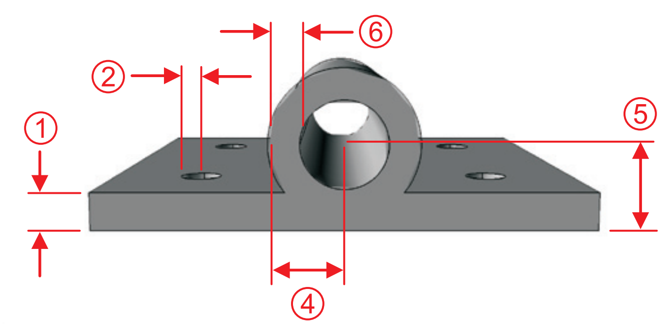


Figure A-26 *Panel hinge parameters*

12. Hinge Plate Thickness - Spandrels
Controls the hinge plate's thickness for spandrel panels.

<u>Slab Parameters</u>

1. Slab Offset From Façade
Controls the offset distance between the outer edge of the façade and the slab edge.

2. Slab Vertical Offset
Controls the heights of the floor slabs relative to the input point rows.

3. Slab Thickness
Controls the thickness of the slabs.

4. Slab Curb Height
Controls the heights of the curbs on all slabs except those of the ground floor and roof.

5. Slab Curb Width
Controls the widths of all slab curbs.

6. Slab Chamfer Width
Controls the widths of the chamfers where for the truss attachment plates.

7. Ground Slab Offset Margin
Allows for fine-tuning of the ground floor slab offset distance to accommodate truss hubs projecting out over the edge at bends in the façade.

8. Empty Bay Slab Curb Trim
Controls the width of the gaps in the ground floor slab curb where there are empty bays (for doors etc.)

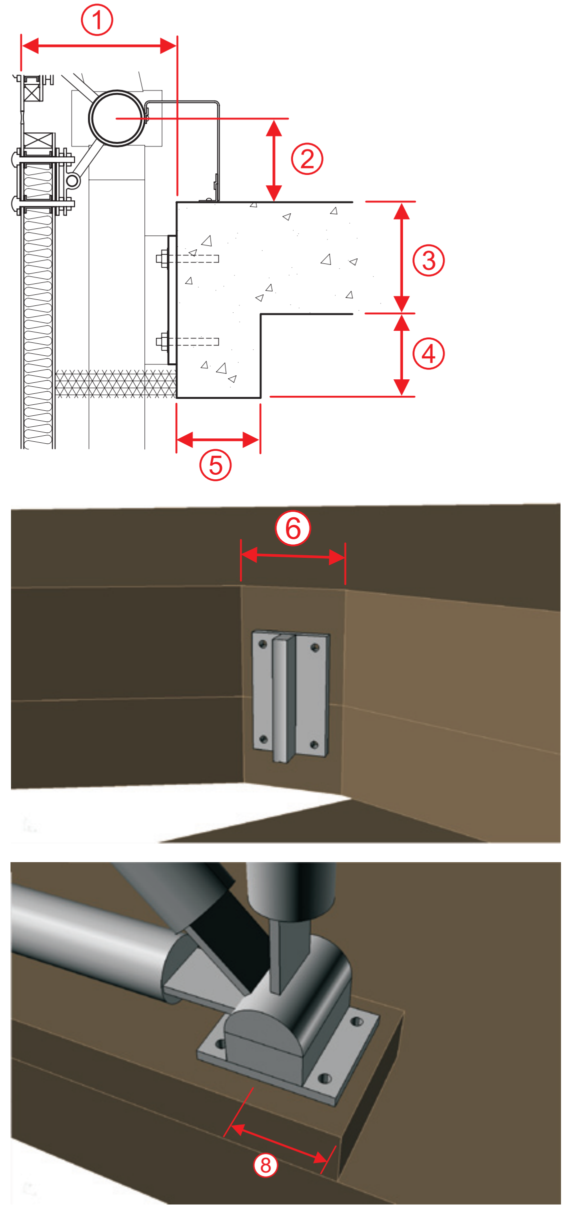


Figure A-27 *Slab parameters*

Ceiling / Column Parameters

1. Ceiling Thickness
Controls the thickness of the ceiling material.

2. Column Width
Controls size of all columns.

Parapet / Roof Parameters

1. Roofing Thickness
Controls the thickness of the roofing assembly above the roof slab.

2. Parapet Cap Thickness
Controls the thickness of the parapet cap.

3. Parapet Cap Angle
Controls the drainage angle of the parapet cap.

4. Parapet Inner Wall Thickness
Controls the thickness of the parapet's inner wall assembly

5. Parapet Height
Controls the height of the parapet not including the cap.

Façade Base Panel Parameters

1. Façade Base Width
Controls the offset between the building foundation and the base panel.

2. Façade Base Drip Edge Height
Controls the height of the base panel's drip edge.

3. Façade Base Thickness
Controls the base panel material's thickness.

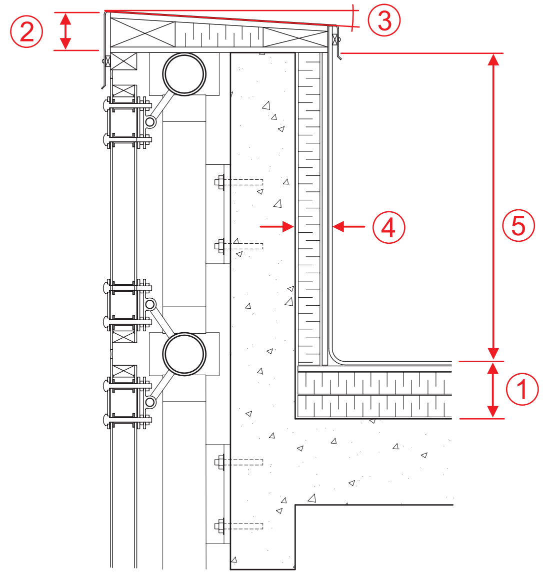


Figure A-28 *Parapet and roof parameters*

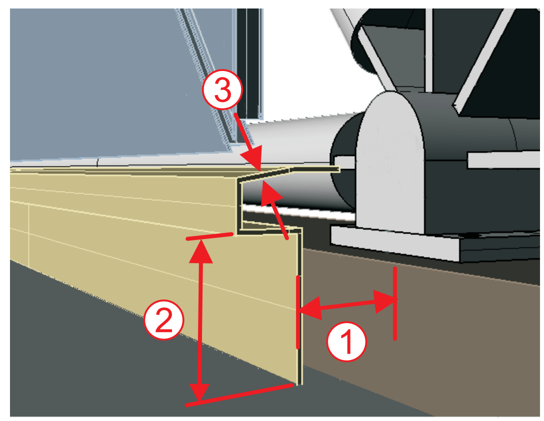


Figure A-29 *Base panel parameters*

Floor-Façade Gap Cover Parameters

1. Gap Cover Width
Controls the width of the floor-façade gap cover assembly. The height is linked to the panel sill height.

2. Gap Cover Thickness
Controls the thickness of the gap cover material.

3. Gap Cover Hub-Slot Clearance
Controls the clearance distance of the gap cover around truss hubs.

Truss Parameters

1. Truss Offset Distance
Sets the distance from the centroids of the truss members to the outer edge of the façade.

2. Truss Tube Radii
Sets the radii of the truss members.

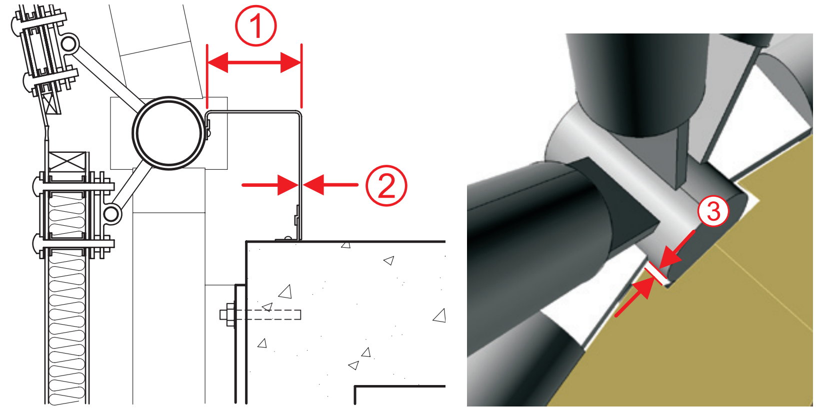


Figure A-30 *Façade-slab gap cover parameters*

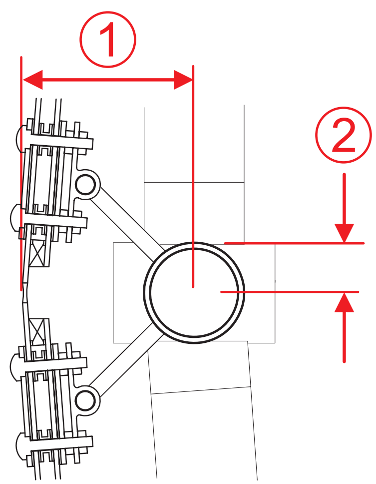


Figure A-31 *Truss parameters*

Truss Hub Parameters

1. Truss Hub Radii
Controls the radii of the truss hubs.

2. Truss Hub Lengths
Sets the truss hub lengths.

3. Truss Spoke Lengths
Sets the truss spoke lengths.

4. Truss Spoke Thickness
Controls the thickness of the truss spoke material.

5. Truss Spoke/Pipe Width Difference
Controls the dimensional difference between the radii of truss pipes and the widths of the spokes.

Truss / Slab Connection Parameters

1. Connection Plate Hole Radius
Controls the radii of the bolt holes.

2. Connection Plate Hole Offset 1
Controls the offset of the first set of bolt holes.

3. Connection Plate Hole Offset 2
Controls the offset of the second set of bolt holes.

4. Connection Plate Thickness
Controls the thickness of the steel plate.

5. Connection Plate Attachment Width
Controls the width of the attachment from the plate to the truss.

Truss Hub Base Plate Parameters

1. Base Plate Length
Specifies the base plate's length.

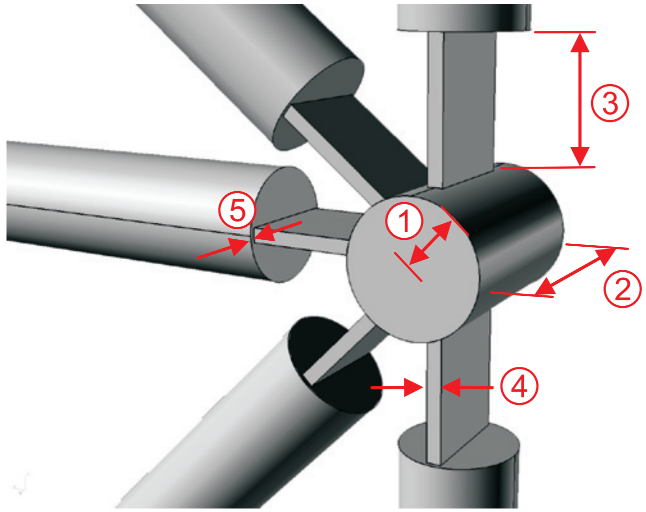


Figure A-32 *Truss hub parameters*

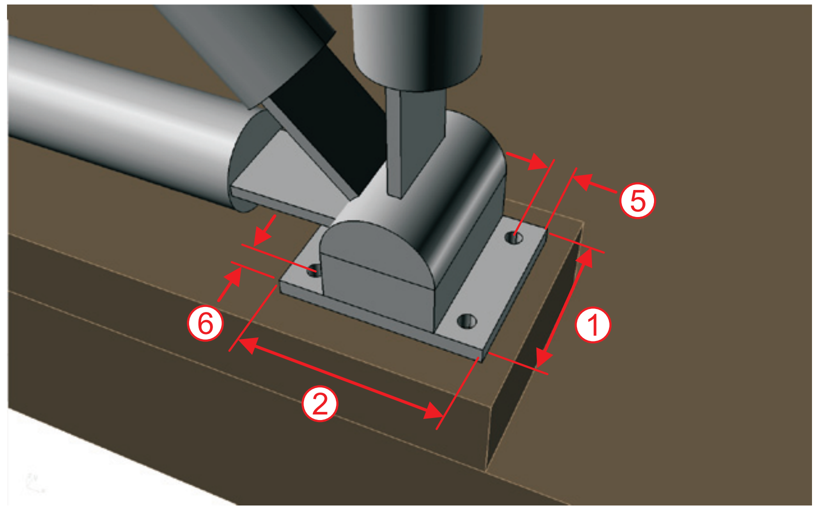


Figure A-33 *Truss hub base plate parameters*

2. Base Plate Width
Specifies the base plate's width.

3. Base Plate Thickness
Controls the thickness of the steel plate.

4. Base Plate Bolt Hole Radius
Controls base plate bolt hole radius.

5. Base Plate Bolt Hole Offset 1
Controls the offset of the first set of bolt holes.

6. Base Plate Bolt Hole Offset 2
Controls the offset of the second set of bolt holes.

Light Shelf Parameters

1. Light Shelf Depth
Controls the depth dimension of the light shelves.

2. Light Shelf Thickness
Controls the thickness of the light shelf assemblies.

3. Gaps Between Light Shelves
Specifies the dimension of the gap between adjacent light shelves.

4. Light Shelf Text Size
Specifies the size of the text that indicates light shelf numbers and locations.

Colors

The following is a list of user-definable colors in the model

1. Translucent Panel Color
2. Opaque Panel Color
3. Glazed Panel Color
4. Solar Panel Color
5. Spandrel Panel Color
6. Panel Spacer Color

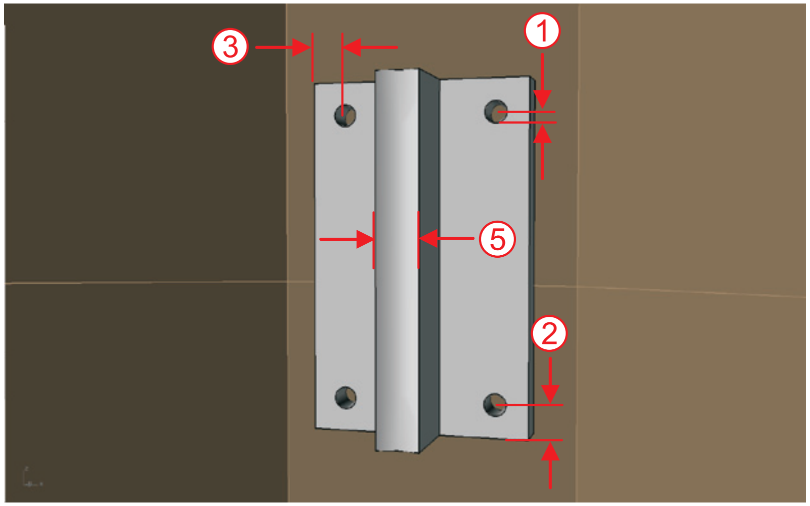


Figure A-34 *Truss-slab connection parameters*

7. Panel Hardware Color
8. Slab Color
9. Truss Color
10. Removed Panel Color
11. Column / Core / Ceiling / Finishing Color
12. Parapet / Façade Base Color
13. Light Shelf Color
14. Panel Edge Length Warning Color
15. Panel Clip Inaccessibility Warning Color
16. Panel Joint Angle Warning Color
17. Façade Input Point Marker Color
18. Partition Input Point Marker Color
19. View Object / Vector Color
20. Unselected View Object Color
21. Ground Plane Color
22. Ground Plane Point Marker Color

3. Model Statistics

Number of Panels

These statistics track the total number of panels as well as the number of each type of panel type of panel, as well as the number of panels currently being viewed if in Partial Façade View mode.

1. Total Number of Bays Viewed
2. Total Number of Panels Viewed
3. Number of Translucent Panels Viewed
4. Number of Opaque Panels Viewed
5. Number of Glazed Panels Viewed
6. Number of Solar Panels Viewed
7. Total Number of Panels
8. Total Number of Translucent Panels
9. Total Number of Opaque Panels
10. Total Number of Glazed Panels
11. Total Number of Solar Panels

View Evaluation

These statistics track the number and percentage of panels that fall within the set angle threshold for views when Panel View Angle Evaluation Mode is enabled. The numbers reflect only panels that are enabled.

1. Number of Panels Evaluated
2. Number of Panels Within Threshold
3. Percentage of Panels Within Threshold

Solar Panel Evaluation

These statistics track the number and percentage of panels that fall within the set angle thresholds set for solar panel placement while Solar Panel Angle Evaluation mode is enabled. The numbers reflect only panels that are enabled.

1. Number of Panels Evaluated
2. Number of Panels Within Threshold

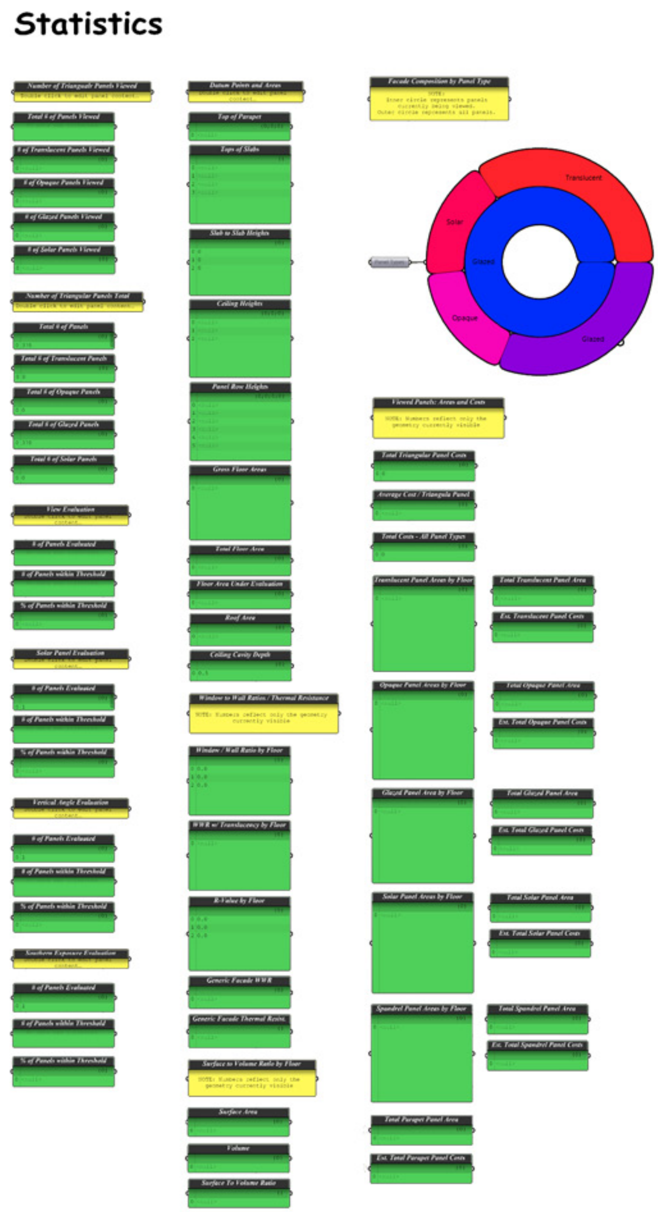


Figure A-35 *Model statistics*

3. Percentage of Panels Within Threshold

Vertical Angle Evaluation

These statistics track the number and percentage of panels that are within the set angle tolerance for Panel View Evaluation mode. The numbers reflect only panels that are enabled.

1. Number of Panels Evaluated
2. Number of Panels Within Threshold
3. Percentage of Panels Within Threshold

Southern Exposure Evaluation

These statistics track the number and percentage of panels that are within the set angle tolerance for Panel Southern Exposure Evaluation Mode. The numbers reflect only panels that are enabled.

1. Number of Panels Evaluated
2. Number of Panels Within Threshold
3. Percentage of Panels Within Threshold

Daylight Factor / Daylight Autonomy

These statistics track the number and percentage of nodes that are within the set Daylight Autonomy and Daylight Factor thresholds (minimums).

1. Number of Nodes Evaluated
2. Number of Nodes Within Threshold
3. Percentage of Nodes Within Threshold
4. Average Value of Nodes

Datum Points and Areas

1. Top of Parapet

Tracks the distance from the ground plane to the top of the parapet.

2. Top of Slabs

Tracks the distance from the ground plane to each floor slab including the *roof slab.*

3. Slab-to-Slab Heights
Tracks the distances between slabs.

4. Ceiling Heights
Tracks the distances between each set of slabs and ceilings.

5. Panel Row Heights
Tracks the heights of each individual row of triangular panels.

6. Panel Sill Heights
Measures the distance from the slab to the bottom of the first row of triangular panels, which is the same for each floor.
7. Panel Aperture Heights
Measures the vertical distance that the triangular panels span for each floor. If there are two rows of panels, the distance represents the sum of the two rows.

8. Panel Tops (from Slabs)
Measures the distances from the slabs to the top edges of the top row of triangular panels for each floor.

9. Gross Floor Areas
Tracks the Gross Floor Areas for each floor excluding the roof.

10. Partitioned Floor Area
Tracks the floor area of the partitioned space currently being evaluated in Daylight Evaluation Mode.

11. Total Floor Area
The sum of all Gross Floor Areas

12. Roof Area
Tracks the gross area of the roof

13. Ceiling Cavity Depth
Tracks the distance from slab bottom to ceiling. The distance is monitored because it is controlled indirectly by the spandrel height and the slab vertical

offset parameters.

Window to Wall Ratios and Thermal Resistances

1. Window to Wall Ratio
Gives the WWR by floor. The number will vary depending on the shape of the walls and the sequence of panels. For each floor the set of spandrel panels above the triangular panels is used in the calculation. When in Partial Façade View mode, only the visible panels are included in the calculation.

2. Window to Wall Ratio with Translucency
Performs the same calculation as above but counts the translucent panels as partial windows based on the coefficient defined by the "Translucency Factor" parameter.

3. R-Value
Gives an approximation of the thermal resistance of the panel assembly for each floor, including spandrel panels. In Partial Façade View mode, the numbers will reflect only the visible panels.

4. Window to Wall Ratio by Floor
WWR broken down by individual floor.

5. Window to Wall Ratio with Translucency by Floor
WWR with translucency broken down by individual floor.

6. R-Value by Floor
R-Value broken down by individual floor.

7. Generic Façade WWR
Gives the WWR for the Generic Façade generated by the user as a reference purposes.

8. Generic Façade Thermal Resistance
Gives the approximate thermal resistance for the Generic Façade created by the user for reference purposes.

Surface to Volume Ratio

1. Surface Area
The external surface area for the selected floor. The floor is selected in the Daylight Evaluation parameter section.

2. Volume
The volume for the selected floor. If only part of the façade is being viewed, the partition walls used in Daylight Evaluation Mode are used to calculate the volume.
3. Surface to Volume Ratio
The Surface Volume divided by the Surface Area for the part of the model being evaluated.

Façade Composition by Panel Type

This pie chart give s a quick visual indication of the relative proportions of each panel type used in the façade. The inner circle represents the current view and the outer circle represents the entire building.
Viewed Panel Areas and Costs

These statistics measure the total areas of panels by type and also use cost per square meter parameters set by the user to estimate panelling costs. When viewing only part of the façade, the areas and costs reflect only the panels currently being viewed.

1. Total Triangular Panel Costs
This calculates the total cost of all triangular panels currently being viewed.

2. Average Cost per Triangular Panel
Calculates the average cost per triangular panel for the triangular panels currently being viewed.

3. Total Costs - All Panel Types
This calculates the total cost of all the panels including spandrel and parapet panels currently being viewed.

The areas and costs are broken down by panel type as well:

4. Translucent Panel Areas by Floor
5. Total Translucent Panel Area
6. Estimated Translucent Panel Costs
7. Opaque Panel Areas by Floor
8. Total Opaque Panel Area
9. Estimated Opaque Panel Costs
10. Glazed Panel Areas by Floor
11. Total Glazed Panel Area
12. Estimated Glazed Panel Costs
13. Solar Panel Areas by Floor
14. Total Solar Panel Area
15. Estimated Solar Panel Costs
16. Spandrel Panel Areas by Floor
17. Total Spandrel Panel Area
18. Estimated Spandrel Panel Costs
19. Total Parapet Panel Area
20. Estimated Parapet Panel Area

Appendix B: Explorations in Generative Components and Digital Project

Before Grasshopper was chosen as the software to develop the façade model in, a number of exploratory models were created in Generative Components and Digital Project. In addition to being informative, these experiments served as an integral part of the development of the façade system concept.

<u>Generative Components</u>

Most of the experiments with Generative Components were done in the very early stages of the design process when the façade system concept was only vaguely defined. Initially it was thought that the façade system would involve singly or doubly-curved shapes, so the focus was on finding a highly flexible curvilinear system that could be modeled efficiently by the software. Overall, the experience of using Generative Components was not a very positive one. Its interface was found to be clumsy and it was difficult to find comprehensive documentation for the product.

1) A doubly-curved control surface is used as a means of modifying the shape of a series of columns and beams.

2) An egg-crate support structure is used for a doubly-curved surface.

3) A space-frame type of component is populated onto a curved surface. An inner offset surface controls the depth of the truss.

4) Iso-curves are used to trim a surface in order to create fragments to be used for panels. Attempts to turn the two-dimensional trimmed surface facets into three-dimensional panels were unsuccessful.

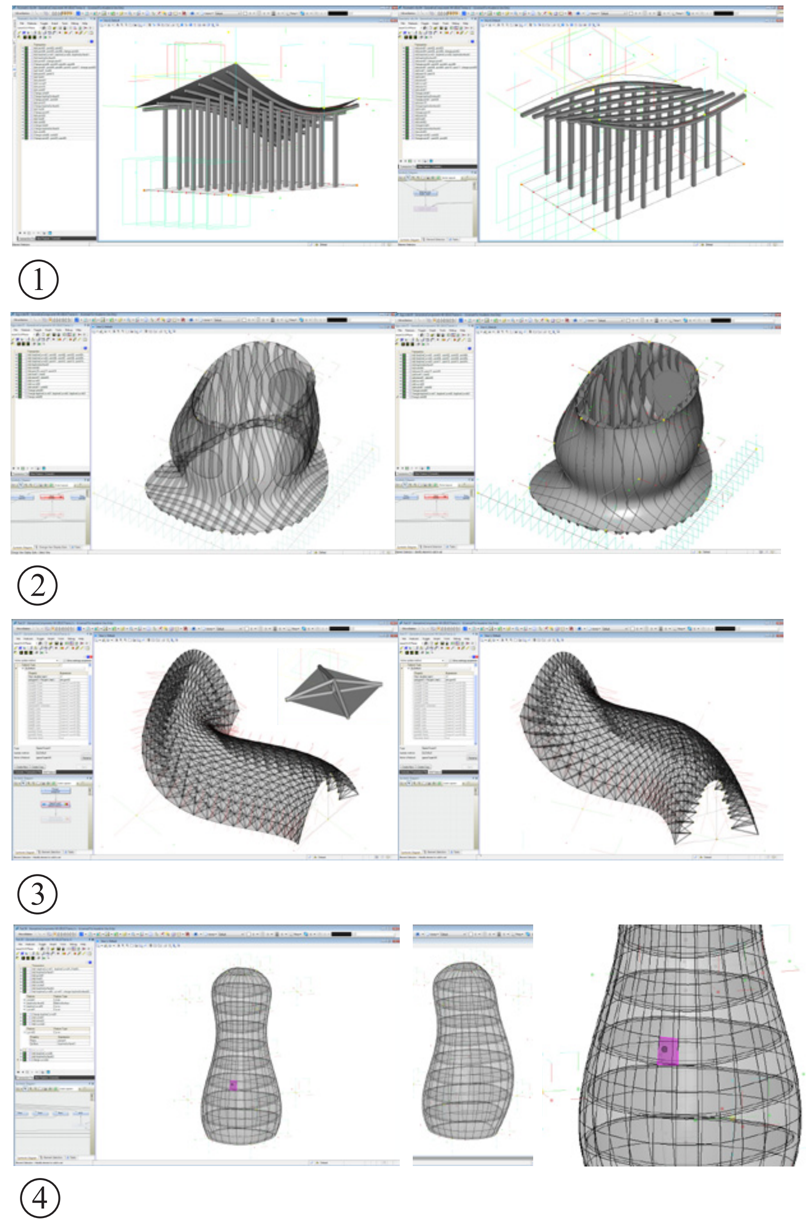


Figure B-1 *Generative Components models*

Digital Project

While the explorations in Generative Components were somewhat abstract, the experiments in Digital Project were much more focused and proved to be instrumental in the evolution of the façade concept.

1) A façade surface is formed by interpolating between four splines - one at the level of each slab. The shapes and positions of windows are controlled via a two-dimensional drawing parallel to the façade. The window shapes are punched out of the façade using boolean operations.

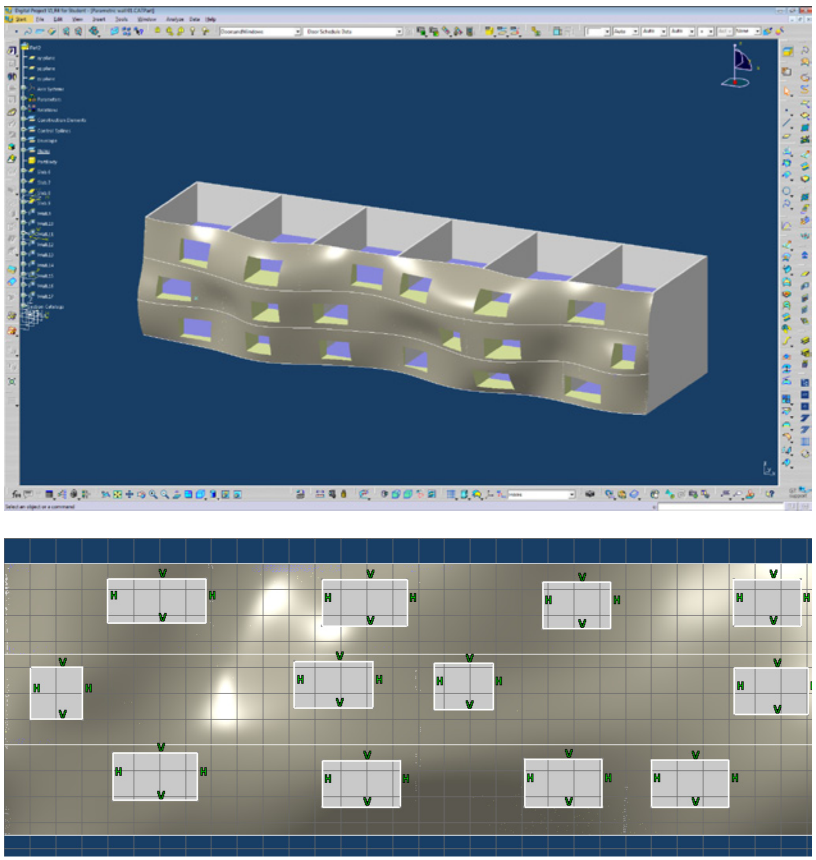


Figure B-2 *Generative Components model #1.*

2) A triangulated façade is constructed here in a similar fashion to the one in the final Grasshopper model. Vertices at the corners of the triangular facets can be moved freely in the horizontal plane to modulate the shape of the façade.

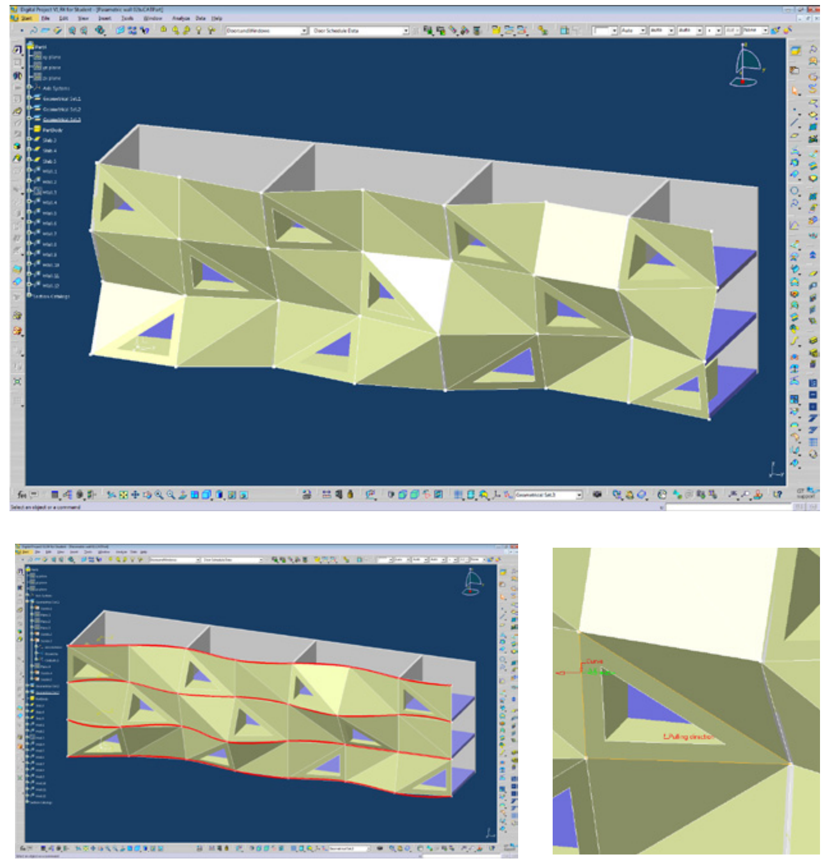


Figure B-3 *Generative Components model #2.*

3) The idea of facets is now taken a step further by using the PowerCopy tool to populate multiple instances of a flexible module onto an array of points. The points are connected in rows by splines, and can once again be freely moved in the horizontal plane to modulate the façade. This time, however, the shapes and sizes of the window apertures can adjusted individually using two-dimensional drawings linked to the face of each module.

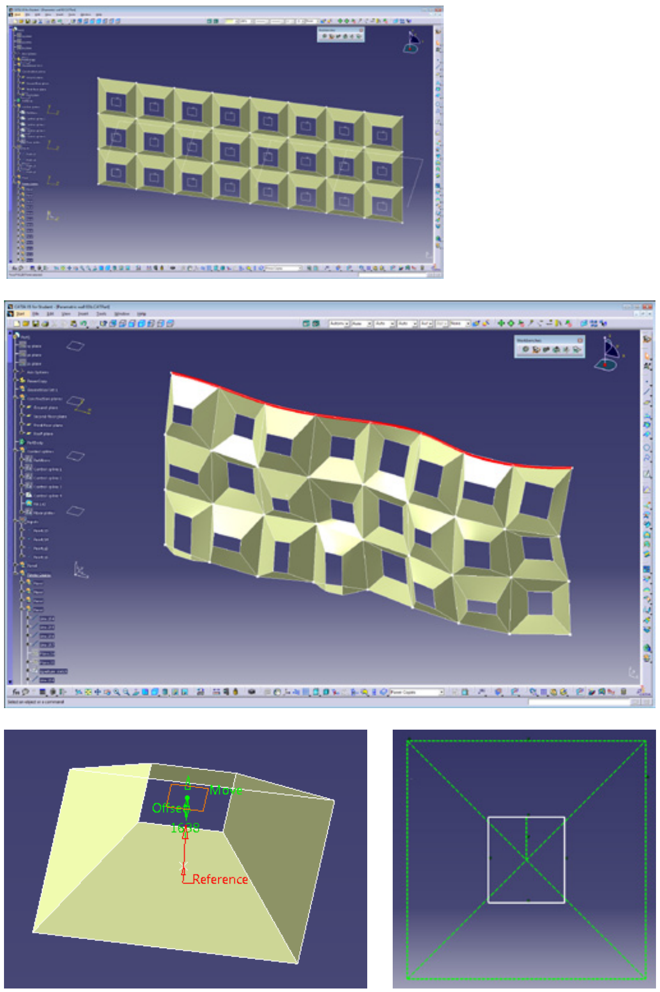


Figure B-4 *Generative Components model #3*

4) Here, the panels start to take on much more organic forms. The curvature and depth of the façade is determined by a double-layered spline grid, while the shapes of the individual windows are still controlled by drawings linked to the face of each panel. The system was very expressive but somewhat unpredictable in terms of topology. For example, because the panel shapes could twist in three dimensions, the panels would tend to become non-planar. This meant that window shapes would change unpredictably when the overall surface shape was altered, and that window frames were not guaranteed to seat properly within the openings.

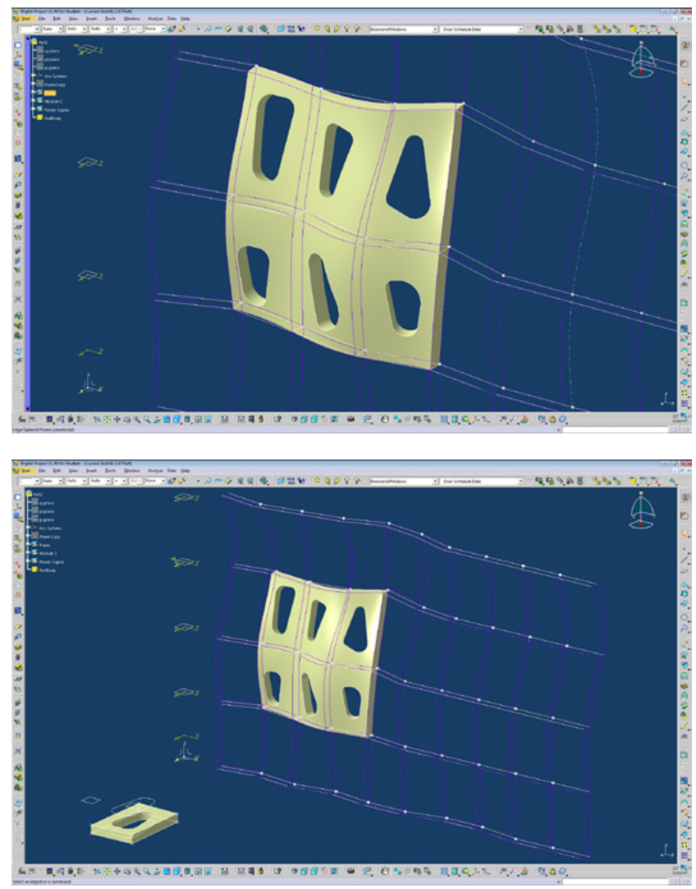
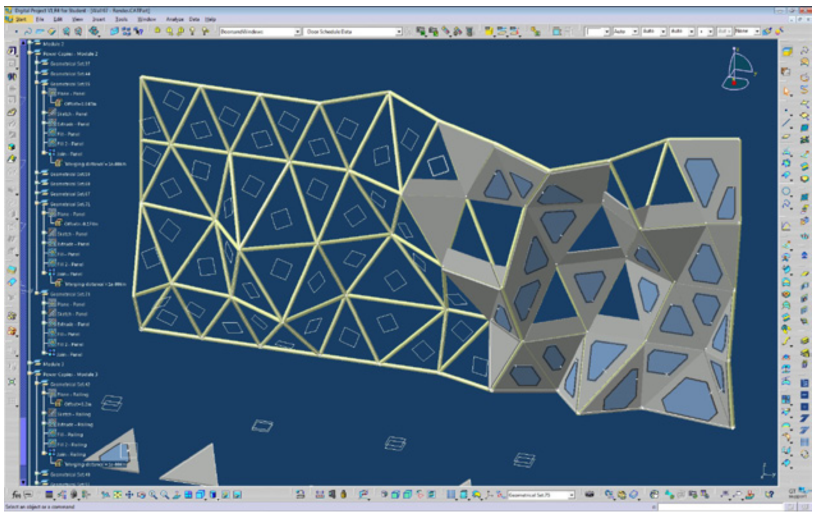
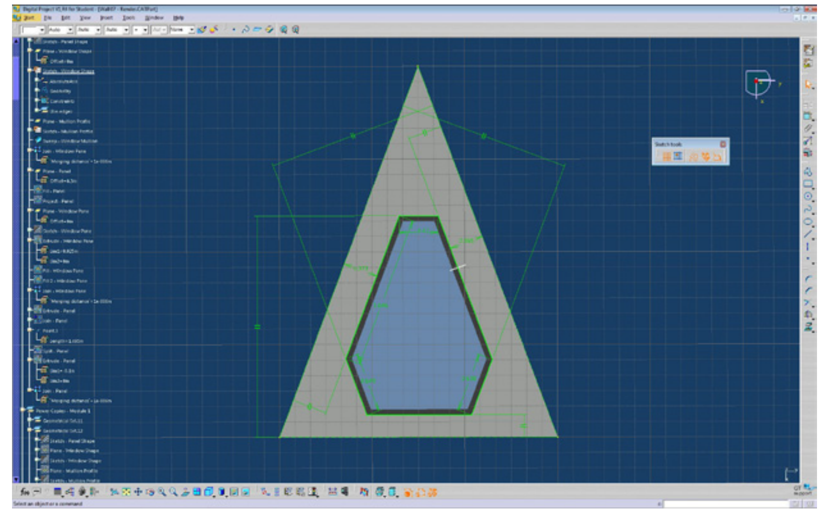


Figure B-5 *Generative Components model #4*

5) This experiment marked a critical turning point in the design. It used a triangular grid, which proved to be flexible as well as consistent topologically (e.g. surfaces are always planar). The idea of using different, interchangeable panel types also emerged here. There were three basic types of panels initially: partially glazed panels with mullions, opaque panels, and two types of partial panels that were designed to delineate balcony areas.





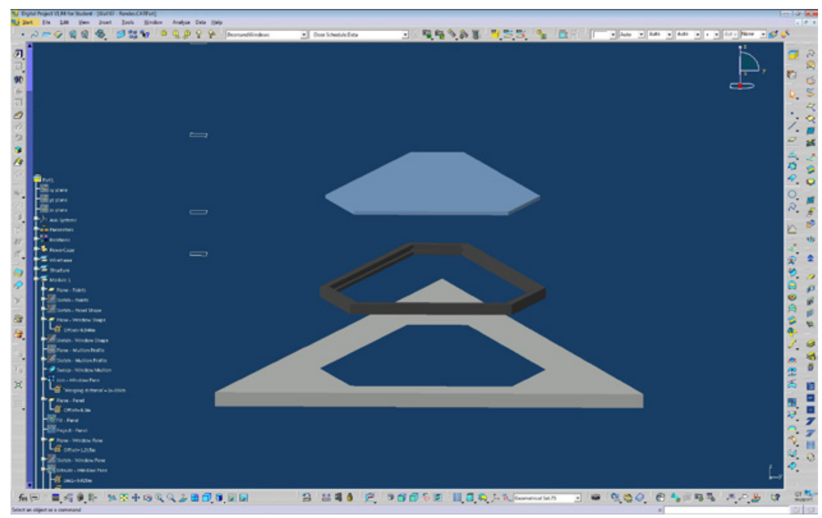


Figure B-6 *Generative Components model #5*

6) The triangular topology was used again here, only with slightly different geometry and constraints for the windows.

While the ability to sculpt window shapes individually in Digital Project was an asset, it was becoming clear at this point that the instantiation process for populating panels was both cumbersome and unpredictable, and that the ability to either scale the total number of panels (or truss members) quickly, or substitute panel types on the fly, was very limited.

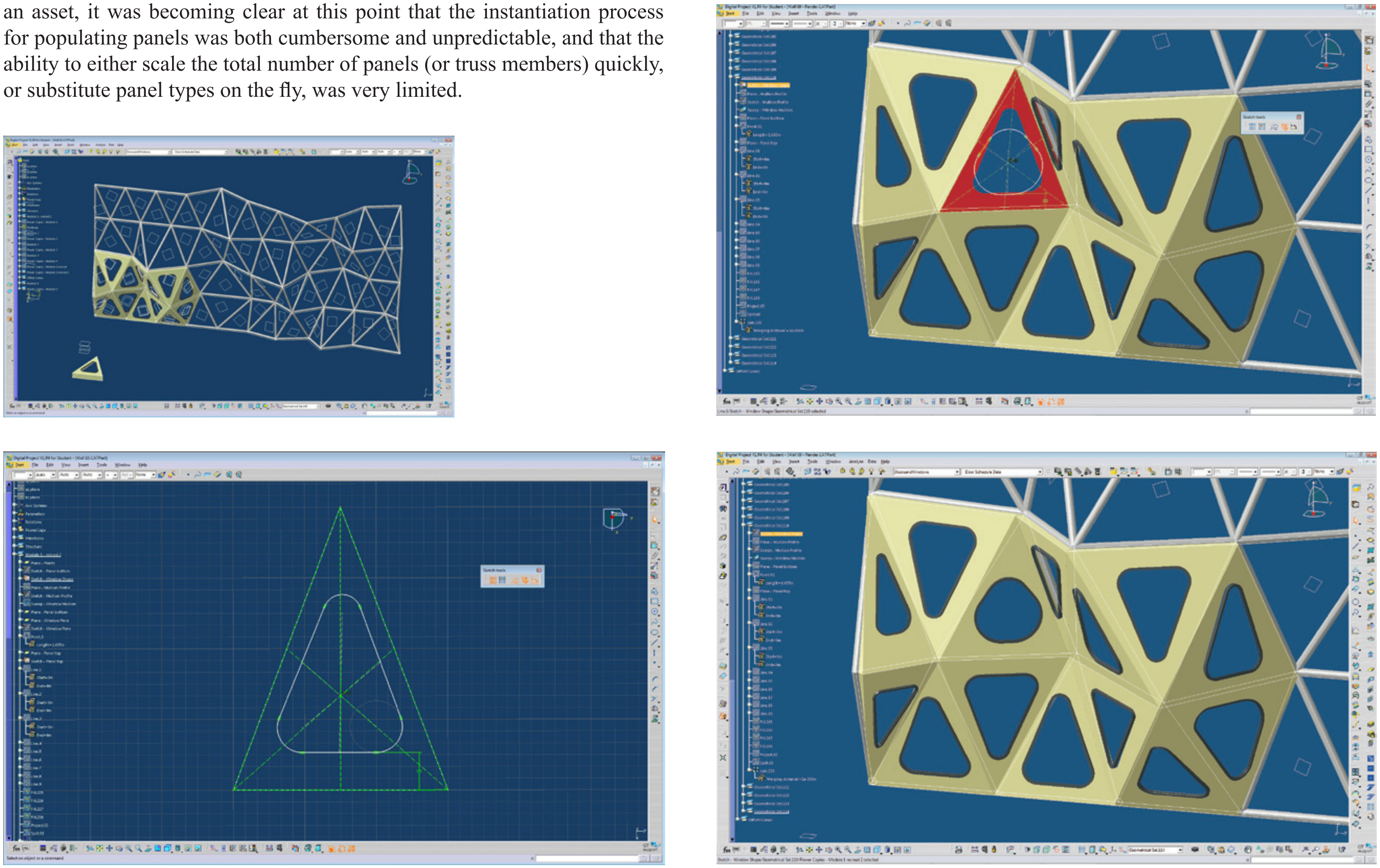


Figure B-7 *Generative Components model #6*

Appendix C: Bugs and Idiosyncrasies in the Model

This section outlines some of the known bugs and quirks in both the software and the model itself.

Grasshopper / Rhino Bugs

1) As outlined in section 3.11, there is a bug in Grasshopper that causes extraneous vertices to be created when a line is offset in a parallel direction to its construction plane. As of this writing, the bug remains, though some other work-arounds have been proposed by members of the Grasshopper community.

2) Boolean and loft operations in Grasshopper can be very unreliable at this time. Consequently there are many combinations of parameters for the clips and hinges that will either not produce the correct geometry, or simply fail to produce any geometry. Other objects in the model like the base panels can occasionally fail to generate properly as well.

3) Due to a memory management bug in either Rhino or Grasshopper, there are fairly severe limitations as to how much geometry can be displayed in the model at once. During the creation of the façade model, crashes would tend to occur if the amount of memory being used by Rhino exceeded three Gigabytes. This limit can easily be reached when displaying fifty panels or less if all the clip and hinge geometry is enabled. A series of successive changes to the model's parameters or shape can also cause crashes even with much less geometry enabled.

4) Converting or "baking" Grasshopper geometry into Rhino geometry can have unpredictable results at times.

DIVA bugs

The daylight evaluation plugin for Rhino, DIVA, is still in beta development has many bugs currently. Some of the bugs were brought to the attention of the software developers and were resolved, but several remain unresolved at this time, including:

1) Visualization-based evaluations, such as glare analyses, are not working at all in the model.

2) Analyses won't work if any of the objects are using the translucent material.

3) The material list will not update properly at times causing the analysis to fail unless a material is changed or the material component is disconnected and then reconnected.

Façade Model Bugs and Quirks

1) Truss spokes occasionally disappear when manipulating the shape of the model. This appears to be caused by failed boolean operations, but it is uncertain whether this is a software problem or a problem with how the objects are grouped for the boolean operations.

2) The workaround created to deal with the offset bug in Grasshopper is fairly reliable on the whole, however it appears to be afflicted occasionally by another bug. Certain façade configurations cause an irregular sequence of offset points to be generated in spite of the fact that the intersection lines and planes appear to be aligned perfectly. Consequently, certain façade configurations will cause the truss to resemble spaghetti. It is unknown if this is a software bug or an error in the way the geometry was created.

3) When viewing only part of the façade, the view can't use a start point that with a greater number than the end point. For example, a start point of "30" and an end point of "5 will fail to produce any results due to the fact that all geometry is created in sequence starting from lower-numbered input points and moving to higher-numbered input points.

4) Not specifying any empty bays in the spreadsheet can cause panels to disappear when in Partial Façade View mode.

Endnotes

[1] Branko Kolarevic, *Architecture in the Digital Age : Design and Manufacturing* (New York ; London: Spon Press, 2003), 52.

[2] Stephen Kieran and James Timberlake, *Refabricating Architecture : How Manufacturing Methodologies are Poised to Transform Building Construction* (New York ; London: McGraw-Hill, 2004), 13.

[3] Kenneth Frampton, *Modern Architecture : A Critical History*. 3rd ed., rev. and enl ed. (London: Thames and Hudson, 1992), 33-35.

[4] Ivan Pasini, *Daylighting Guide for Canadian Commercial Buildings* (Public Works and Government Services Canada, 2002), 6-9.

[5] Frampton, 33.

[6] "Henry Ford: Quotes", Wikiquote, accessed December 30, 2011, http://en.wikiquote.org/wiki/Henry_Ford

[7] Daniel L. Schodek, *Digital Design and Manufacturing : CAD* (Hoboken: John Wiley & Sons, 2005) p.155.

[8] A study of Frampton's *Modern Architecture : A Critical History* or Alan Colquhoun's *Modern Architecture* provides a good introduction to the interesting and inextricable relationships between mass-production, artistic expression, and social ideologies in modern architecture.

[9] Schodek, 20.

[10] Kieran and Timberlake, 12.

[11] Branko Kolarevic and Kevin R. Klinger, *Manufacturing Material Effects : Rethinking Design and Making in Architecture* (New York: Routledge, 2008) 65.

[12] Kolarevic, 53.

[13] Schodek, 226-228

[14] Ibid., 124-125.

[15] Shiro Matsushima, "Technology-mediated process: MIT Stata Center case study" (from the proceedings of the AIA/ACADIA Fabrication Conference, Cambridge, Ont., 2004), 205.

[16] Kolarevic and Klinger, 62.

[17] David Littlefield, *Space Craft : Developments in Architectural Computing* (London: RIBA Pub., 2008), 182.

[18] Kolarevic and Klinger, 120.

[19] Kieren and Timberlake, xi-xii.

[20] Kolarevic and Klinger, 28.

[21] Alec Leggatt, *GRP and Buildings : Design Guide for Architects and Engineers* (London ; Toronto: Butterworths, 1984), 62-69.

[22] Kieran and Timberlake, 59.

[23] Schodek, 316-317.

[24] Kieran and Timberlake, 129.

[25] Kolarevic and Klinger, 59.

[26] Matsushima, 202-219

[27] Shelley Murphy, "MIT sues Gehry, citing leaks in $300M complex", *Boston Globe*, November 6, 2007, accessed December 28, 2011, http://www.boston.com/news/local/articles/2007/11/06/mit_sues_gehry_citing_leaks_in_300m_complex/

[28] Francis Strauven, *Aldo Van Eyck : The Shape of Relativilty*, Rev. and updated English ed. (Amsterdam: Architectura & Natura, 1998), 342.

[29] Kolorevic and Klinger, 123.

[30] Kolarevic and Klinger, 121.

[31] Robert Woodbury, *Elements of Parametric Design* (London ; New York: Routledge, 2010), 12.

[32] Per Eriksson et al., "Static Eigenvalue Analysis as an aid in furniture design" (from the proceedings of the AIA/ACADIA Fabrication Conference, Cambridge, Ont., 2004), 126-137.

[33] C. Rother & W. Rudolph, "The canopy at the Smithsonian institutions Kogod courtyard in Washington, D.C". *Detail (English Ed.)*, 2, 182-186, 217 (2009). Retrieved from http://search.proquest.com/docview/55470793 ?accountid=14906

[34] Littlefield, 30-33.

[35] Ibid., 27-30.

[36] As a general rule, machining or material removal processes, such as cutting, drilling or milling, are the simplest, cheapest and most common operations in which CNC is used. Fabrication processes that involve deformation or moulding, such as extruding, stamping, forging, casting, etc., tend to be more cost and labor intensive even when employing CNC tools. For a good overview of CNC tools and fabrication processes, see Shodek's *Digital Design and Manufacturing*, 255-311.

[37] "Top 10 Most Frequently Asked Questions," Research Frontiers Inc., accessed December 30, 2011, http://www.smartglass.com/faqs.asp#100011

[38] "Environmentally Friendly Smart Glass Windows," SmartGlass International Ltd., accessed December 30, 2011, http://www.smartglassinternational.com/smart-glass-windows/

[39] "Building Integrated Photovoltaics (BIPV)", National Institute of Building Sciences: Whole Building Design Guide, accessed December 30, 2011, http://www.wbdg.org/resources/bipv.php

[40] During the development of the thesis project, the author visited a fabrication facility run by Walters Inc., a Hamilton, Ontario based steel contractor and fabricator. Walters specializes in both pre-fabricated and custom steel truss work for buildings and infrastructure, and has been involved in high profile projects such as the Michael Lee-Chin Crystal, Daniel Leibskind's addition to the Royal Ontario Museum. While details of the visit to Walters are subject to a non-disclosure agreement, the visit provided the author with valuable information about the very sophisticated capabilities of steel fabricators currently with respect to, cutting, bending, rigging and welding steel truss components. The visit helped inform the design for the truss system as well as ideas about its fabrication and construction.

[41] For a good overview of the myriad of issues involved in daylighting, see Norbert Lechner, *Heating, Cooling, Lighting : Design Methods for Architects*, 2nd ed ed. (New York ; Chichester [England]: John Wiley, 2001), 325-406.

[42] Rea, Mark Stanley, *IESNA Lighting Handbook : Reference & Application*, 9th ed ed. (New York, N.Y.: Illuminating Engineering Society of North America, 2000), 467.

[43] Pasini, 25.

[44] Schodek, 82.

[45] André Chaszar, *Blurring the Lines. Architecture in Practice* (Chichester, England ; Hoboken, NJ: John Wiley, 2006), 175.

Bibliography

ACADIA (Conference) (28th : 2008 : Minneapolis,Minn.), Andrew Kudless, Neri Oxman, and Marc Swackhamer. *ACADIA 08 : Silicon + Skin : Biological Processes and Computation : Proceedings of the 28th Annual Conference of the Association for Computer Aided D*. [United States]: Association for Computer-Aided Design in Architecture, 2008.

ACADIA (Conference) (29th : 2009 : Chicago,Ill.), Tristan d'Estrée Sterk, Russell Loveridge, and Douglas Pancoast. *ACADIA 09 : ReForm() : Building a Better Tomorrow : Proceedings of the 29th Annual Conference of the Association for Computer Aided Design in Archite*. [United States]: Association for Computer-Aided Design in Architecture, 2009.

AIA/ACADIA Fabrication Conference (2004 : University of Waterloo), Philip Beesley, Nancy Yen-Wen Cheng, and R. Shane Williamson. *Fabrication : Examining the Digital Practice of Architecture : Proceedings of the 2004 AIA*. [Cambridge, Ont.]: University of Waterloo School of Architecture Press, 2004.

Allen, Edward and Joseph Iano. *Fundamentals of Building Construction : Materials and Methods*. 4th ed. Hoboken, N.J.: J. Wiley & Sons, 2004.

Boake, Terri Meyer. *CISC Guide for Specifying Architecturally Exposed Structural Steel*. 2011. Canadian Institute of Steel Construction. www.cisc-icca.ca

Chaszar, André. *Blurring the Lines. Architecture in Practice*. Chichester, England ; Hoboken, NJ: John Wiley, 2006.

Colquhoun, Alan. *Modern Architecture*. Oxford History of Ar. Oxford ; New York: Oxford University Press, 2002.

Eiffert, Patrina and Gregory J. Kiss. *Building-Integrated Photovoltaic Designs for Commercial and Institutional Structures: A Sourcebook for Architects*. National Renewable Energy Laboratory, 2000.

Frampton, Kenneth. *Modern Architecture : A Critical History*. 3rd ed., rev. and enl ed. London: Thames and Hudson, 1992.

Gross, Markus and Doo Young Kwon. *A Framework for 3D Spatial Gesture Design and Modeling using a Wearable Input Device*. Swiss Fereral Institute of Technology, Zürich. https://www.rdb.ethz.ch/projects/project.php?type=&proj_id=12888&z_detailed=1, accessed December 30, 2011.

Guzowski, Mary. *Daylighting for Sustainable Design*. McGraw-Hill Professional Architectur. New York ; London: McGraw-Hill, 2000.

Harvey, David. *Condition of Postmodernity : An Enquiry into the Origins of Cultural Change*. Oxford [England]: Basil Blackwell, 1990.

Herzog, Thomas, Roland Krippner, and Werner Lang. *Facade Construction Manual.* 1st ed. Basel; Munich: Berkhäuser ; Edition Detail, 2008.

Issa, Rajaa and Andrew Payne. *Grasshopper Primer*. 2nd ed. www.lift-architects.com, 2009.

Khabazi, Zubin. *Generative Algorithms using Grasshopper*. www.morphogenesis.com, 2010.

Kieran, Stephen and James Timberlake. *Refabricating Architecture : How Manufacturing Methodologies are Poised to Transform Building Construction*. New York ; London: McGraw-Hill, 2004.

Knaack, Ulrich. *Façades Principles of Construction*. Basel: Birkhäuser, 2007.

Kolarevic, Branko. *Architecture in the Digital Age : Design and Manufacturing*. New York ; London: Spon Press, 2003.

Kolarevic, Branko and Kevin R. Klinger. *Manufacturing Material Effects : Rethinking Design and Making in Architecture*. New York: Routledge, 2008.

Lechner, Norbert. *Heating, Cooling, Lighting : Design Methods for Architects*. 2nd ed ed. New York ; Chichester [England]: John Wiley, 2001.

Leggatt, Alec. *GRP and Buildings : Design Guide for Architects and Engineers*. London ; Toronto: Butterworths, 1984.

Littlefield, David. *Space Craft : Developments in Architectural Computing*. London: RIBA Pub., 2008.
Moore, Fuller. *Concepts and Practice of Architectural Daylighting*. New York: Van Nostrand Reinhold, 1985.

Norberg-Schulz, Christian. *Principles of Modern Architecture*. London: Andreas Papadakis, 2000.

Pasini, Ivan, project manager. *Daylighting Guide for Canadian Commercial Buildings*. Public Works and Government Services Canada, 2002.

Rea, Mark Stanley. *IESNA Lighting Handbook : Reference & Application*. 9th ed ed. New York, N.Y.: Illuminating Engineering Society of North America, 2000.

Robbins, Claude L. *Daylighting : Design and Analysis*. New York: Van Nostrand Reinhold, 1986.

Rother, C., & Rudolph, W. (2009). "The canopy at the Smithsonian institutions Kogod courtyard in Washington, D.C". *Detail (English Ed.)*, 2, 182-186,217. Retrieved from http://search.proquest.com/docview/55470793?accountid=14906

Schodek, Daniel L. *Digital Design and Manufacturing : CAD*. Hoboken: John Wiley & Sons, 2005.

Strauven, Francis. *Aldo Van Eyck : The Shape of Relativilty*. Rev. and updated English ed ed. Amsterdam: Architectura & Natura, 1998.

Woodbury, Robert (Robert Francis). *Elements of Parametric Design*. London ; New York: Routledge, 2010.

Web Sites

Bentley. http://www.bentley.com/en-US

Digital Toolbox. http://digitaltoolbox.info/

Diva for Rhino. http://www.diva-for-rhino.com/index.html

Gehry Technologies. http://www.gehrytechnologies.com/

Generative Design Computing. http://www.generativedesigncomputing.net/

Grasshopper 3d. http://www.grasshopper3d.com/

Morphcode. http://morphocode.com/work/research

National Institute of Building Sciences: Whole Building Design Guide. http://www.wbdg.org

Notion Parallax http://www.notionparallax.co.uk/blog/index.php

ParaMod.net. http://paramod2.blogspot.com/

Research Frontiers Inc. http://www.smartglass.com

Rhino 3D.TV. http://www.rhino3d.tv/

Rhinoceros. http://www.rhino3d.com/

SmartGlass International Ltd. http://www.smartglassinternational.com