SPEEDING UP THE PROCESS OF MODELING TEMPORARY STRUCTURES IN A BUILDING INFORMATION MODEL USING PREDEFINED FAMILIES

A Thesis

by

PARSA SABAHI

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

December 2010

Major Subject: Construction Management

Speeding Up the Process of Modeling Temporary Structures in a

Building Information Model Using Predefined Families

SPEEDING UP THE PROCESS OF MODELING TEMPORARY STRUCTURES IN A BUILDING INFORMATION MODEL USING PREDEFINED FAMILIES

A Thesis

by

PARSA SABAHI

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	Julian Kang
Committee Members,	Boong Yeol Ryoo
	Wei Yan
Head of Department,	Joe Horlen

December 2010

Major Subject: Construction Management

ABSTRACT

Speeding Up the Process of Modeling Temporary Structures in a Building Information Model Using Predefined Families. (December 2010)

Parsa Sabahi, B.S., Iran University of Science and Technology

Chair of Advisory Committee: Dr. Julian Kang

It has been less than a decade that Building Information Modeling (BIM) has been used in construction industries. During this short period of time the application of this new modeling approach has increased significantly, but still the main users of such models are architects (for design purposes) and general contractors (for coordination purposes). Most of the BIM applications are developed to meet design and coordination requirements; yet sub-contractors face hassles when using this new technology to model their products, equipment and services in detail. The literature reveals that one of the reasons is they do not have access to the tools and objects they need to model their work within the BIM environment. Temporary structures, such as shoring systems and formworks, are good examples. Although these structures play a significant role in the logistics of the jobsite, there are neither a special tools nor predefined objects in BIM applications to help model these elements.

In this case study a real building project has been used to model these temporary structures by two different methods: 1. Using ordinary tools in BIM application (Revit), and, 2. Using predefined parametric families of objects developed and customized for this project. During the modeling process, time has been recorded as well as other observations describing obstacles, advantages, and disadvantages of both methods. The results show that the usage of predefined parametric families speeds up the process of modeling and also helps to create a model which is more understandable and informative than 2D drawings that are currently being used by most of the sub contractors.

ACKNOWLEDGEMENTS

I would like to thank my committee chair, Dr. Kang, and my committee members, Dr. Ryoo and Dr. Yan, for their guidance and support throughout the course of this research.

I also would like to thank Skanska USA Building Inc. for providing the Structural Revit Model and Baker Concrete Construction Contractors for providing the shoring and formwork layout and details.

Thanks also go to my friends and colleagues, and the department faculty and staff, for making my time at Texas A&M University a great experience.

TABLE OF CONTENTS

Page

Page

LIST OF FIGURES

Page

Page

LIST OF TABLES

Page

1. INTRODUCTION

The concept of Building Information Model has been around since late 1970’s. This concept then came into reality in late 1980’s by computer applications such as “Graphisoft” and “ArchiCAD”, nevertheless BIM as everybody knows and uses today, is less than 10 years old. BIM is known as an object-oriented model of the building in virtual reality contains every detail, as well as any imaginable data related to a building such as cost, schedule, etc.

Construction visualization is one of the greatest benefits of BIM. In this process one or more building elements shape a work package. In a separate file a schedule defines sequence and duration of different work packages, then in another software (e.g. Autodesk Navis works) the work packages and schedule will be linked together and construction process is visualized as it is done in reality (4D modeling). Visualization will help project participant to make more reliable decisions during both planning and construction stages (Chau et al, 2004).

Construction visualization could also be used for communication among project participants, construction safety management, and site logistic planning which is the scope of this research.

As it is mentioned above, construction visualization resembles the work flow in the job site. The more accurate this presentation is, the more reliable the decisions based on the model would be. Since the logistics are an important part of the work flow in the jobsite, considering them as a part of 4D model is inevitable. For example in a reinforced concrete building project, the formworks (which are one of the main focuses of current

---

This thesis follows the style of *Construction Engineering and Management*.

study) play an important role in construction work flow. Their relevant issues from storage to installation and schedule of use could have dramatic effects on cash flow, number of required formworks, sequence of tasks and safety. By modeling formworks and their related equipments such as shoring, all these issues could be visualized before the actual project takes place.

This is the point that problem arises. When it comes to graphical part of the Building information model, BIM applications focus on the buildings elements itself and they neglect temporary equipments and logistics. For example; in REVIT there is a variety of objects for each budding element category from structural to architectural and MEP but there is no predefined family or object specifically developed to model formworks. In such a case modelers will have to make these equipments either by existing entities (e.g. making formworks using structural elements like beams and columns as support and bracing combined with some solid geometries to model the formwork's panel), or using features designed for making solid geometries in the Revit model. This process seems to be cumbersome and tedious; besides, the process of placing formworks would not be straight forward. The reason is that such an approach will not utilize intelligent capabilities of BIM application. For example REVIT users can not benefit from object snap[1] or host feature[2] like smart families do.

The alternative solution would be utilizing families' capabilities by creating certain families for formworks and related equipments such as scaffolding and shoring. In following sections this alternative solution will be discussed.

---

[1] A feature in the drawing area of the application that enables the user to select and grab commonly used points of an object like end point, center point, midpoint etc.

[2] In REVIT some objects are defined as a host for some other objects. For example walls are host for doors and windows. This makes it easier to place these objects in the model in right place and reduces the corrections required after placing them.

2. LITERATURE REVIEW

2.1. BIM in Practice

In practice, BIM could be something from “a glorified 3D CAD” to “workflow management and decision support tool” depending upon the user (Klemens, 1999).

The major benefit of using BIM is integration of all data produced by different project participants. This fact results in many other benefits. For example, having all geometric data, BIM is a good source for making quantity take-off of the project (Klemens 2009).

By encapsulating any building related data in a single model, BIM facilitates data sharing in a project, its benefits could be dramatic in an “Integrated Project Delivery System” or “Design Build Delivery system”, where there is no ownership issue for the drawings and data sharing is the key to success (Klemens, 1999) .

In practical situation, BIM Could save designers time in various ways. As an instance; after producing computer model required drawings for each task could be generated using that single model. On the other hand a BIM provides details in a 3D environment, and when it is used for communication between designer and contractor it can save time by reducing number of requests for information (RFI). In 2005 U.S. General Services Administration decided to make a Building Information Model of a developed design for a federal courthouse to generate 2D contract documents. The model was also given to general contractor to be used as reference during construction. In the middle of model development process, because of lack of powerful hardware they decided to eliminate some details from Building Information Model and model those in CAD, those details later became a source of RFIs (Post, 2009).

Saving time by using BIM is not limited to construction phase. In the project mentioned above, it also helped new employees to have a better understanding of Project standards by reviewing the library of families (Post, 2009).

Utilization of BIM in industry is not limited to design and construction phase. BIM is being used to assess inventories of existing buildings and identify the renovations which provide the highest return on investment. For example Building Information Model of old buildings (buildings built before 1980) in collaboration with load-calculation softwares are being used to determine energy consumption in different parts of the building and the amount of energy could be saved by upgrading each part of the facility. This process helps owners to prioritize their renovation projects and add to the sustainability of built environment (Middlebrooks and Behrens, 2009).

When it comes to real projects adopting BIM has its own difficulties which Copyright issues appear to be one of them. When BIM is utilized in projects, the question is that who owns the model and is responsible for the materials in the model. On designer side it also alters “professional liabilities, Project Communications and even changes how designers are compensated” (Sullivan, 2005).

In addition existing insurance policies do not cover problems caused by sharing a digital model (Van Hampton et al., 2010).

“Transition from CAD to BIM”, in addition to new softwares, requires hardware upgrade, powerful telecommunication system, software customization, and training skillful staff (Bratton 2009).

2.2. BIM and Contractors

On contractor’s side, one of the most important challenges is that they do not have their component in BIM compatible format. Meadati (2009) believes that lack of 3D as-built models and in cases which there is such a model, lack of information required for construction process in those models is the reason that BIM is less used after preconstruction stage.

Some contractors such as MEP contractors already use their own softwares for calculation purposes and that is why they may not find it reasonable to adopt BIM in their projects. Moreover, other than generic components, MEP firms are usually not provided with a BIM compatible format of components they need by manufacturers (Ireland, 2009).

For MEP contractors BIM needs to be customized to suit their needs. One of the most required customizations is modeling building electrical system components. For example EMCO's Dynalectric has started modeling these components since 2002 and over a 6 year period of time they modeled 4000 components (Bratton 2009).

2.3. BIM and Constructability Analysis

Visualization could be the first contribution BIM makes to the constructability of a project. Since BIM builds up a virtual reality through 3-D environment, any project participant can better understand the complications of the project by looking at the model especially in congested and complicated areas in a building like mechanical rooms. On the other hand, other parameters such as material or functionality could be seen viewed, this sort of data is usually unavailable in traditional reviewing tools (Klemens, 2009).

In addition, BIM applications are capable of finding building elements which may unintentionally interfere with other elements. This process, known as clash detection, could be done either for the entire model or between certain parts of the model which later approach is more time efficient. More over the clash detection feature could also recognize unsatisfied access requirements such as clearance for deferent elements such as pipes or ladders (Klemens, 2009).

Some efforts have been made by Hijazi et al (2009) to evaluate the constructability based on computer model and 4D visualization. In this process peers review the model and visualization and evaluate the whole construction process on a quantitative basis in 18 different areas. The score for each area will be given a different weight, and the total constructability score will be summation of these weighted scores. This idea forms based on the fact that BIM helps decision makers to look at a construction project (which is fragmented by its nature and different participants play role in that process) as a whole to better evaluate the constructability.

Utilizing site facilities is a part of construction process. Looking at these temporary structures as a resource contractors have to have a reasonable schedule for using them as well. Modeling site facilities in a computer 3D model and linking this model with schedule in a 4D visualization makes site planning and management easier and helps to predict any site related problem (Chau et al, 2004). Modeling the use of resources such as scaffolding is essential to evaluate a 4D visualization (McKinney et al., 1998).

3. MOTIVATION

As the literature reveals the most important challenge BIM arises for sub contractors is the necessity of having components they use in construction process in the form of 3D entities which could be placed in Building Information Models. These components could be either materials and building elements or temporary facilities used in jobsite such as shoring system and formworks which are the focus of this research. Customization process could be done by either BIM application developers or contractors themselves. But no matter who takes the responsibility, having each of those two categories above customized for a Building Information Model requires investment in terms of money and time. So contractors need to be convinced that if they have a library of required components in hand it saves them a reasonable amount of time to make their own model. Once contractors are encouraged to adopt such an approach, software vendors will have motivation to invest on such a library.

MEP contractors have started creating their own library, but when it comes to facilities which concrete contractors usually use in the jobsite such an effort could not be seen. It seems either those companies or software vendors are not convinced whether or not creating and using such a library of 3D components (families) representing formworks, scaffolding, shoring systems etc in Building Information Model really makes modeling process faster and more efficient. Finding the answer to this question is the motivation of this study.

4. PROBLEM STATEMENT

Regarding the necessity of having jobsite facilities in the model, and the concept of using customized families, the question is that “does creating certain families for formwork, shoring and scaffolding systems makes the process of modeling these temporary equipments faster, easier, and more efficient?

5. HYPOTHESIS

The following hypothesis will be tested for this study:

Creating certain families for formworks and shoring systems speeds up the process of modeling these temporary equipments in construction visualization process.

6. DELIMITATIONS AND ASSUMPTIONS

In this study following limitations and assumptions will be considered:

1) Autodesk REVIT is the BIM application used for this study so BIM application means REVIT.
2) This is a case study in which a real medical office building project with reinforced concrete structure is used to model mentioned temporary structure.
3) The outcome is based on the time recorded during the process and other observations from a modeler's stand point.
4) All temporary structures created in this study are based on 2D drawings given by concrete sub contractor, and they have been modeled based on modeler's understanding of those 2D drawings.
5) In some cases the configurations presented in 2D drawings have been simplified and some details beyond the scope of the research have been ignored.

7. DEFINITIONS

Following definitions will be used in this study:

1) Family:

   A group of objects that share the same definition and general characteristics in the model, with a common set of parameters (properties), identical use, and similar graphical representation. For example all doors will fall in the doors family and all formworks will shape formworks family.

2) Type:

   A group of objects within a family that have the same geometry. Different types in a family actually represent different kinds or different sizes of the object classified in that family. For example; single flush door and double flush door could be two different types of doors in the door family.

3) Instance:

   Every time users place a certain type of a family somewhere in the model they make an instance of that type.

4) In-place Mass Object:

   Is a feature in Revit allows the modeler to make a compound object by putting existing elements or some simpler solids together and create a complex object out of them.

5) Parameters:

   A setting that determines the appearance or behavior of an element, type, or view. In a REVIT project, Parameters define the relationships between elements of the building model. These relationships are created either automatically by

REVIT or by users as they create a new parametric object (parametric family). The REVIT Parametric Change Engine automatically coordinates changes in all model views, drawing sheets, schedules, sections, and plans.

8. PROPOSED SOLUTION AND IMPLEMENTATION

As it was mentioned earlier, the alternative solution for modeling formworks and related equipments is to create different families containing different types and sizes of formworks and shoring systems. What is thought that will benefit us by creating such families is the capability of handling parameters as well as specific hosts in REVIT families. For each object made in REVIT family editor, different parameters could be defined. Each parameter controls a part of the object's geometry or material (e.g. a formwork height or column material). The good thing about parameters is that user will be able to define certain parameters as a function of the others. In this case when independent parameter changes, REVIT change engine changes the other related parameters automatically.

On the other hand, by making a family starting from specific template user can define that the object is placed only on the surface of another object which will be so useful especially for formworks and shoring systems.

8.1. Vertical Formwork

As an example of how to create a family using a family template, in this section, the procedure of creating wooden formwork for vertical surfaces (such as walls, Columns and beam sides) is illustrated.

In REVIT under file menu by choosing new family the application asks us to choose a family template to start from. Related window is shown in figure 1.

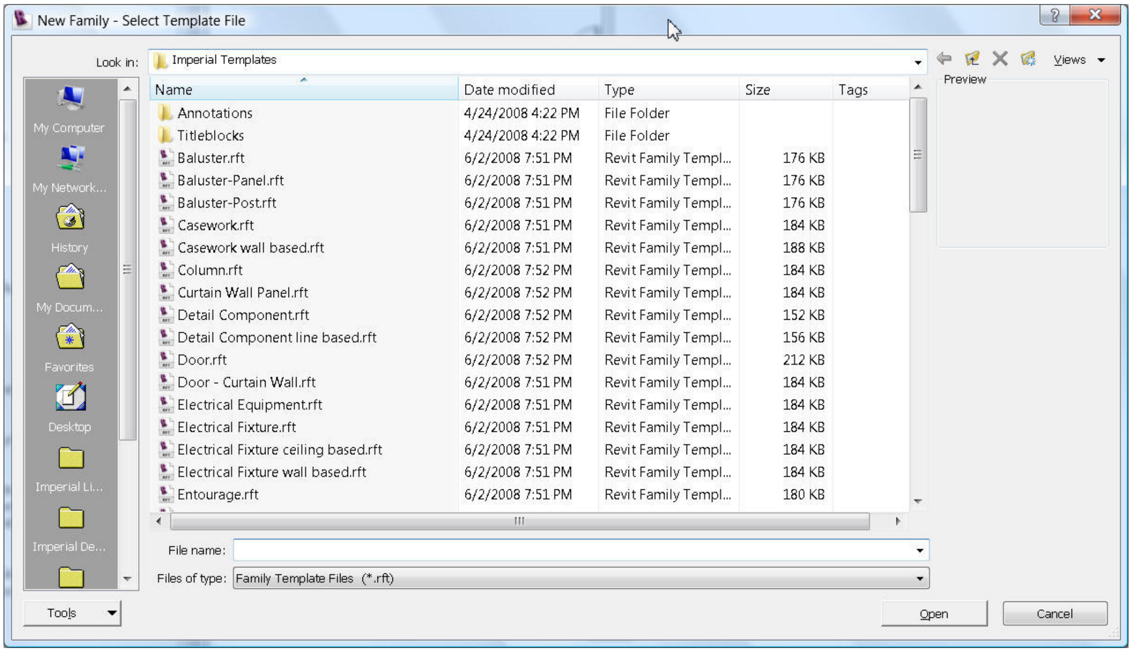


Figure 1 Family template window after opening a new family

The best template that suits our need is “generic model - wall based”. This makes the formwork stick to the surface of any wall in the model. By choosing this template application will takes us to family editor environment (figure 2).

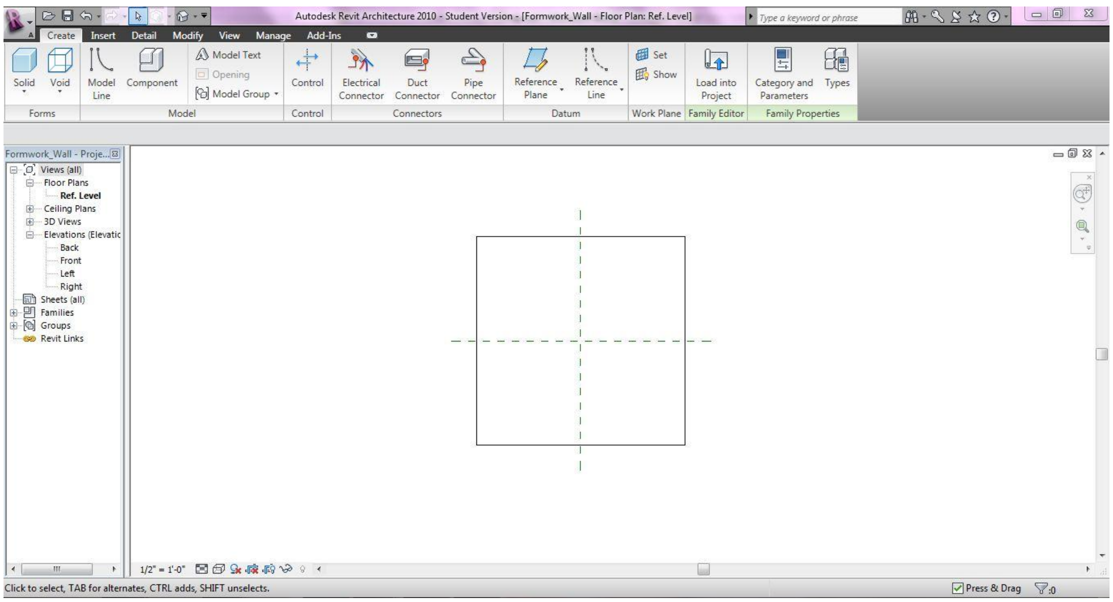


Figure 2 Family template window which pops up after opening a new family

In the next stage major parameters have to be planed. Parameters are what control any variable values (sizes, materials, informational elements, etc. of the family). In addition to deciding what the parameters need to be, user will also need to consider whether they will be type or instance parameters[3]. For this formwork there will be parameters defining

[3] Type parameters control a variable for all instances of a type, which means by changing its value that variable will change on all instances belonging to that certain type while an instance parameter controls related variable on a single object in the model.

height and width of the formwork and the depth of supports and their spacing as shown in figure 3.

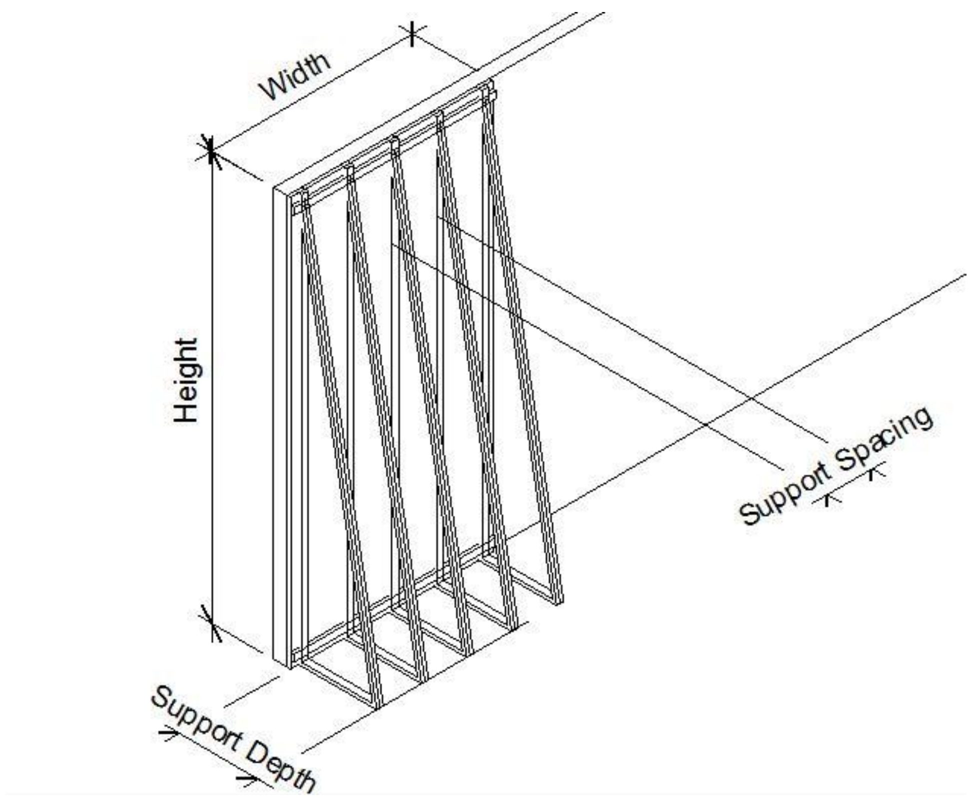


Figure 3 Vertical formwork's parameters

Table 1 shows the name, formula and other properties of each parameter.

Table 1 Parameters and formulas in vertical formwork family

Parameter	Name	Parameter type	Formula
Height	Height	Length	
Width	Width	Length	
Support Depth	Support_Depth	Length	
Support Spacing	Support_Spacing	Length	
Number of Supports	Support_num	Integer	Width/Support_Spacing + 1

There are several ways to create these parameters in REVIT. User can simply use "Family Types" window by pressing "Types" button under "Family Properties" menu.

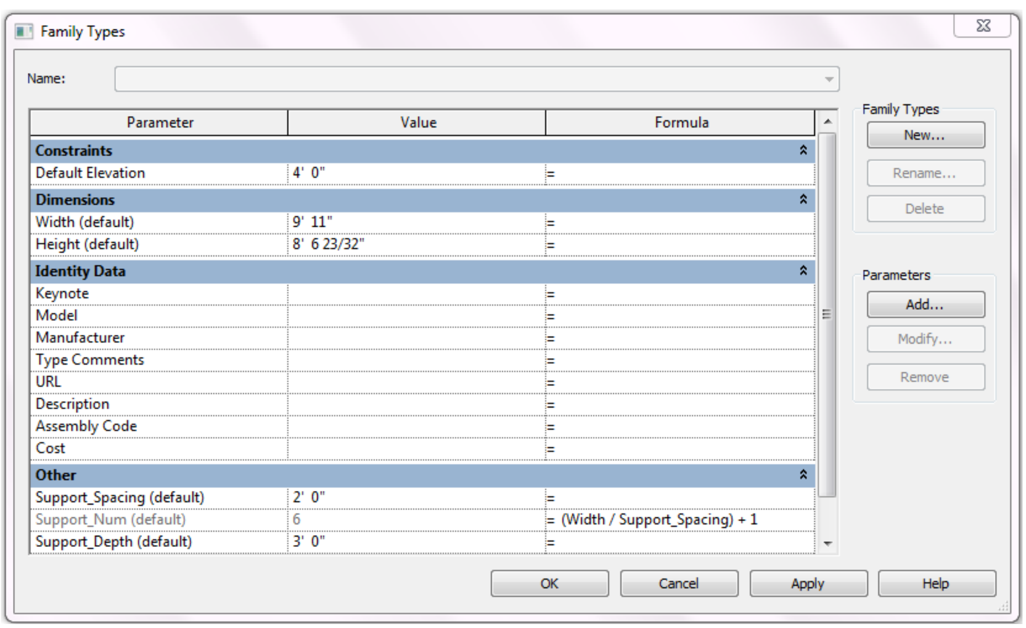


Figure 4 Family types window

Parameters, their default value and their formula are shown in figure 4. Pressing "Add" button will take the user to the "Parameter Properties" window (figure 5) where the user can define a new parameter.

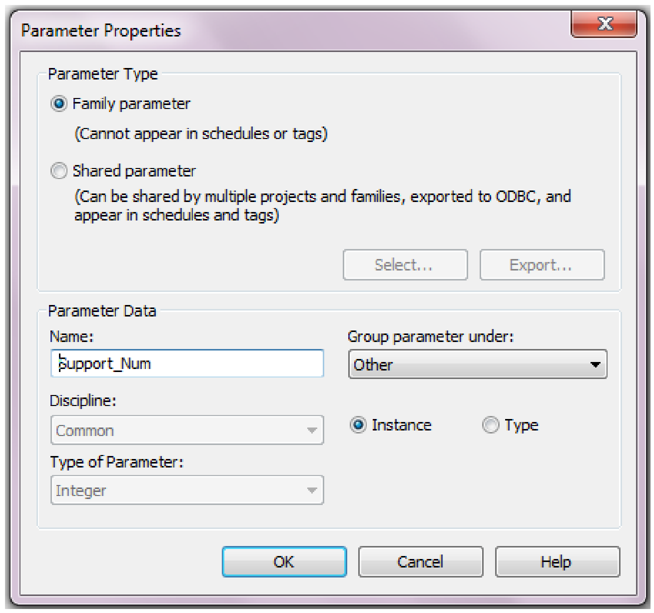


Figure 5 Parameter properties window to define a new parameter

After creating all required parameters, next step is to define the structure of the component using reference lines. Reference lines layout the geometry of the component. Wherever the user needs to control a size (e.g. height, width etc) in the component it has to be defined as a dimension between two reference lines, and then the dimension will be linked to desired parameter. In other word the dimension between two reference lines is where the parameter and the geometry are linked. Based on this convention in REVIT the reference lines would be as in figure 6. In order to be more visible each reference line is presented by a bounding box around it.

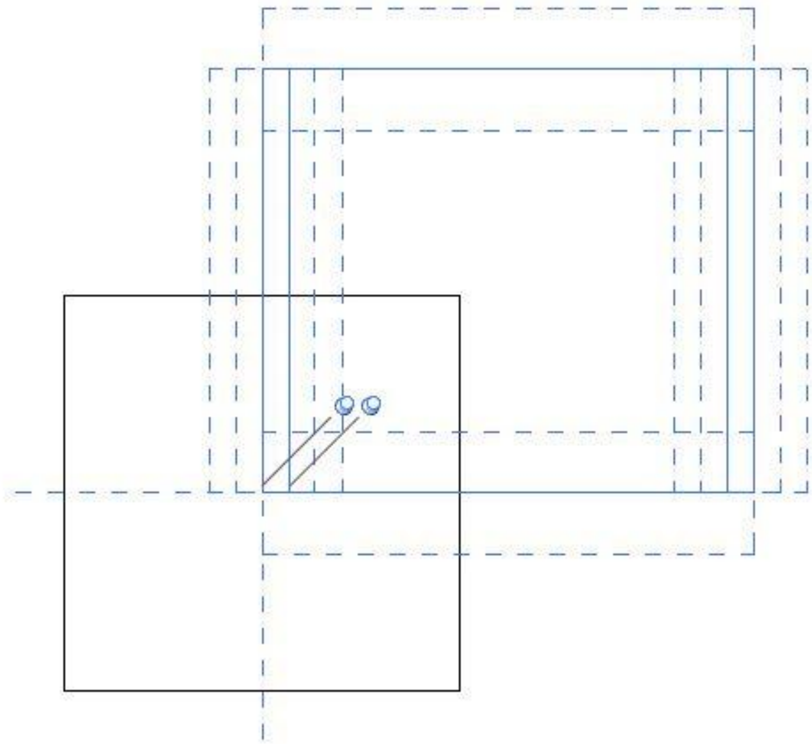


Figure 6 Form work structure created by reference lines

Surrounding reference lines define height and width, while two vertical reference lines will define the location of supports, and two horizontal lines will define the location of horizontal braces.

Now user can create required dimensions. Later they can link each dimension to the appropraite parameter. For height and width two dimention line will be drawn between two vertical and two horizental surounding lines respectively (figure 7).

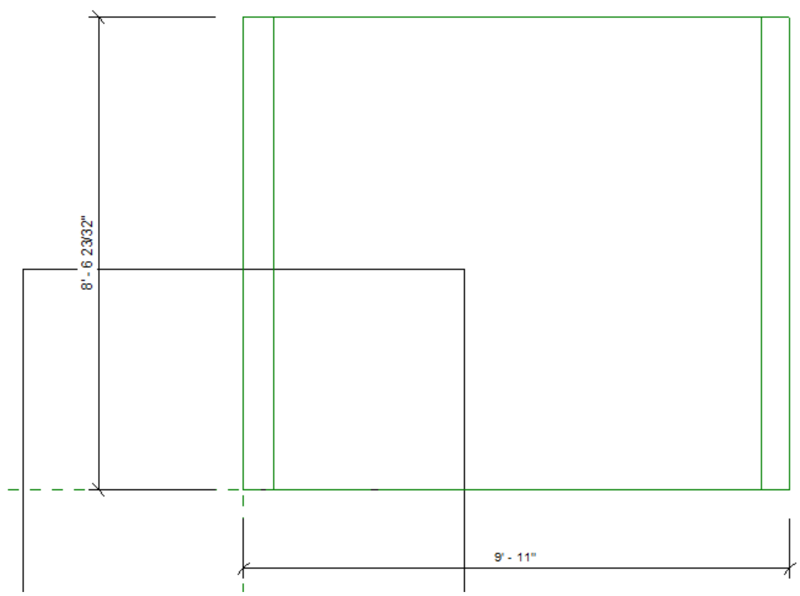


Figure 7 Dimension lines specify what needs to be linked to a parameter

Now in order to specify the related parameter, as shown in figure 8 user can right click on each dimension text and from the drop down menu choose the appropriate parameter created before.

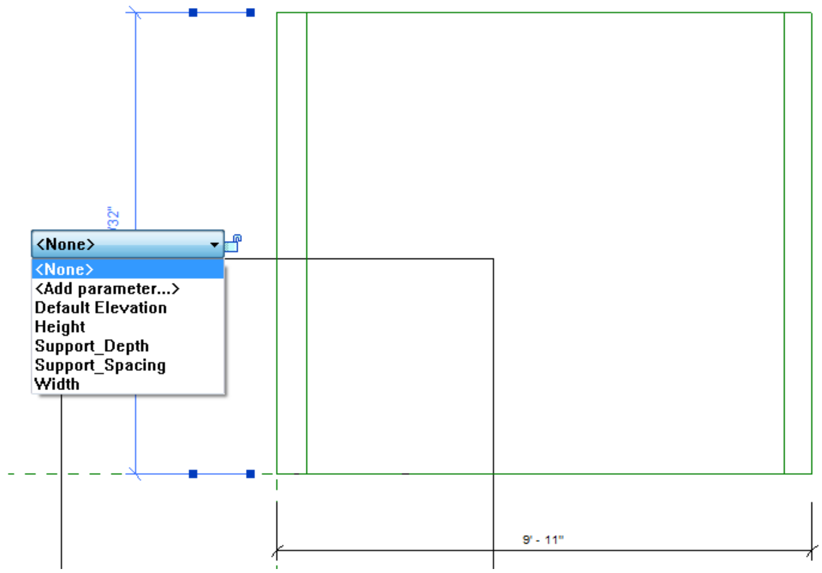


Figure 8 Linking dimensions to appropriate parameters

The triangular support, shown in figure 9 has been made in a different family. The same procedure has been done for the support in its own family and “Support_Depth” has been assigned to the corresponding dimension.

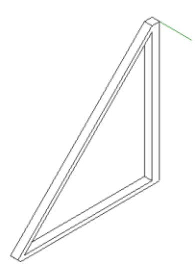


Figure 9 A separate family has been created to model supports

For support spacing the procedure is more complicated and will be discussed when the supports are placed.

After creating the reference lines and the dimensions, different parts of the component will be drawn in place, and connected and locked to one of the reference lines. To make it easier to handle the entire component the support has been made in a separate family and will be loaded to the formwork family later as a nested family. The following figures show how to create the rest of the component step by step.

A panel is made of extrusion of a rectangle which presents the formwork's face (figure 10). The panel is locked to two reference lines on the sides and two other reference lines at the bottom and top of the panel, to define width and height respectively.

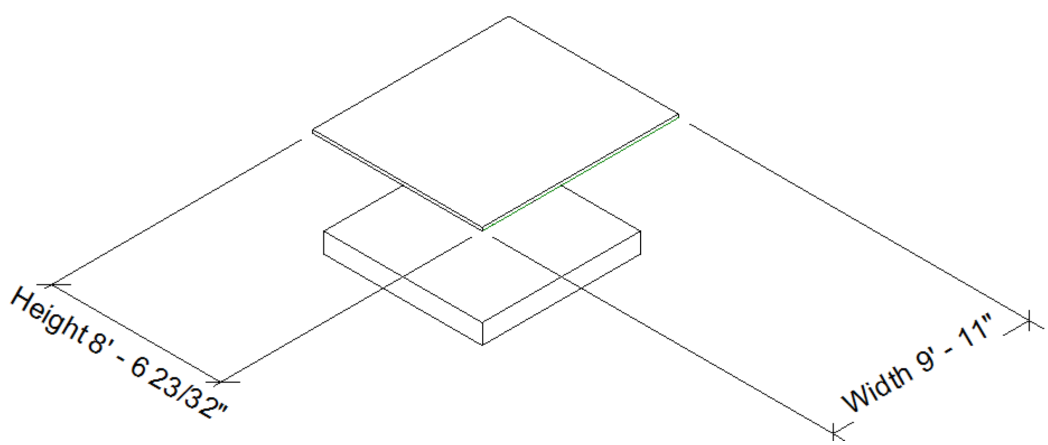


Figure 10 An extrusion has been used to model the face of the formwork

Horizontal braces are made by extruding a rectangle. The two horizontal reference lines are aligned with the center line of the rectangles and locked to them, to define the location of these two braces (figure 11).

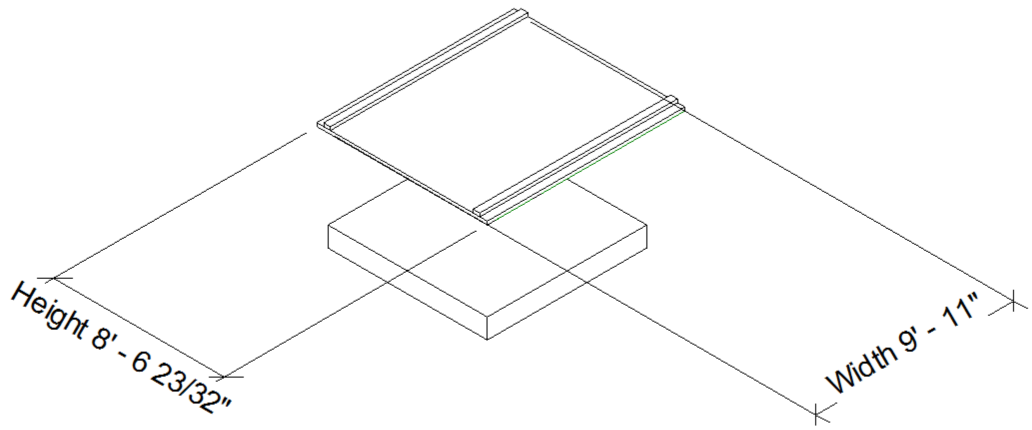


Figure 11 Lumbers as horizontal stiffeners

Now the support is loaded in to the family as a nested family and is aligned with one of the vertical reference lines on its center line (figure12).

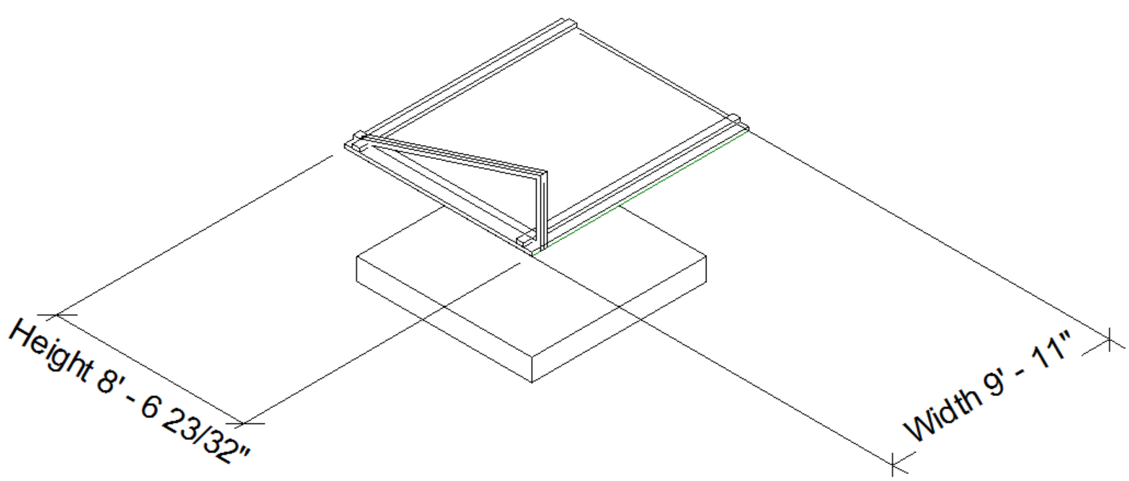


Figure 12 Support family has been nested into main family

Now it is time to specify the supports spacing. In order to do that, first the support is selected, then by using array command (under modify menu) and choosing the "move to last" option the other vertical reference line is introduced as the centerline of the last support and it will be locked to the reference line. At this point there is an array of two supports each one located on one of the vertical reference lines (figure 13). Now the support spacing should be defined.

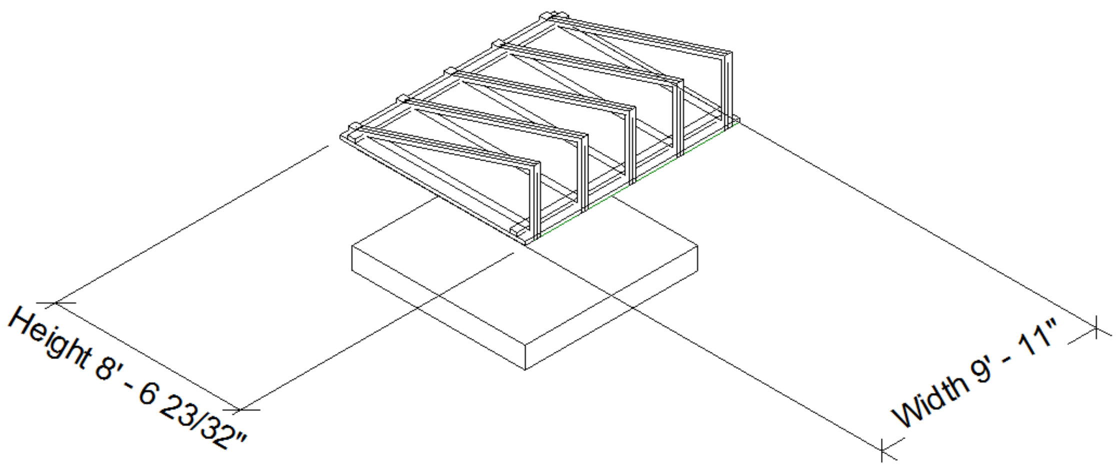


Figure 13 Creating an array of supports

In order to do that, users can create a new parameter named "Support_Spacing" which defines the distance between two consequent supports and will be specified by user. As you can see in figure 14 the default value for this parameter is 2.0 ft. Assuming the distance between first and last support equals the width, the formula in figure 14 will define total number of supports.

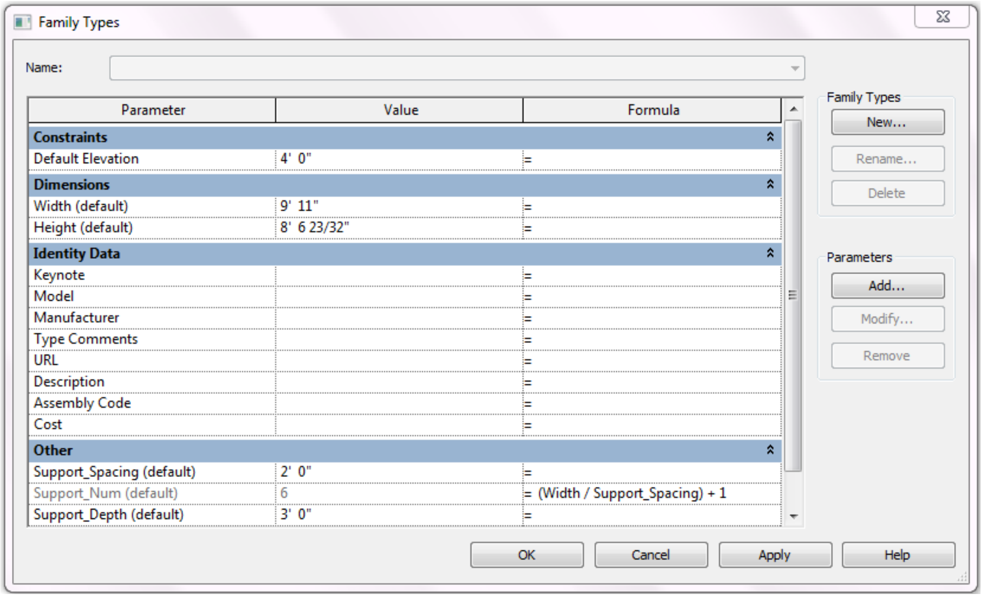


Figure 14 Family types window shows parameters and their formulas

On the other hand, by creating the array, a label specifying number of elements in the array will be created. User will follow the same procedure that has been taken for dimension texts to link this label to the parameter "Support_Num" (figure 15). Now by changing the width of the formwork the distance between two consequent supports will remain approximately constant.

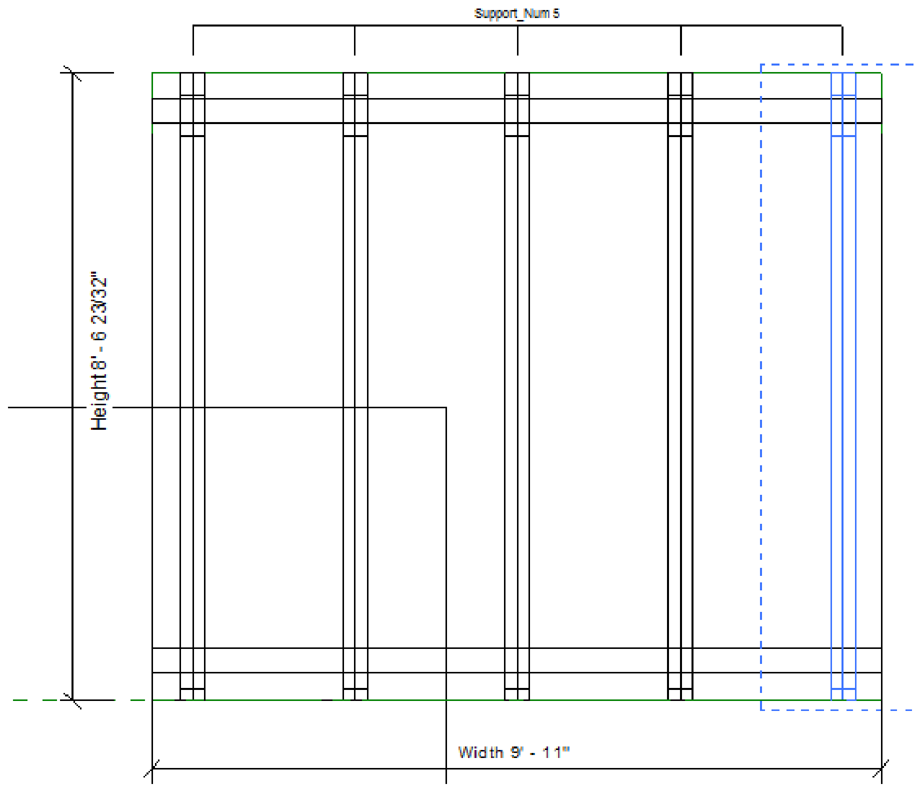


Figure 15 Number of supports is linked to its parameter

To place this object against a wall user needs to go to component toolbox and load this family. Since this family is wall-based once it is load it will be stuck to the surface of the wall or any vertical surface such as side of a beam automatically. The insertion point is located on one of the corners and object snaps could assist users to find the exact insertion point. For placing a formwork system (a set of formworks) Array command could be used. In this case spacing between array elements must be compatible with the length of the formwork to avoid overlapping. Elevation view or section view would be the best view for placing this family.

Following is a brief description of other families created for this research and their parameters.

8.2. Vertical Formwork (Corner)

Taking a look at the formwork system just described shows there will be a clash on the corners, where the supports from two adjacent formworks get in the way of each other. To solve this problem another family is built to cover the corners (figure 16). Like wall formwork this family is also wall-based and the same considerations apply.

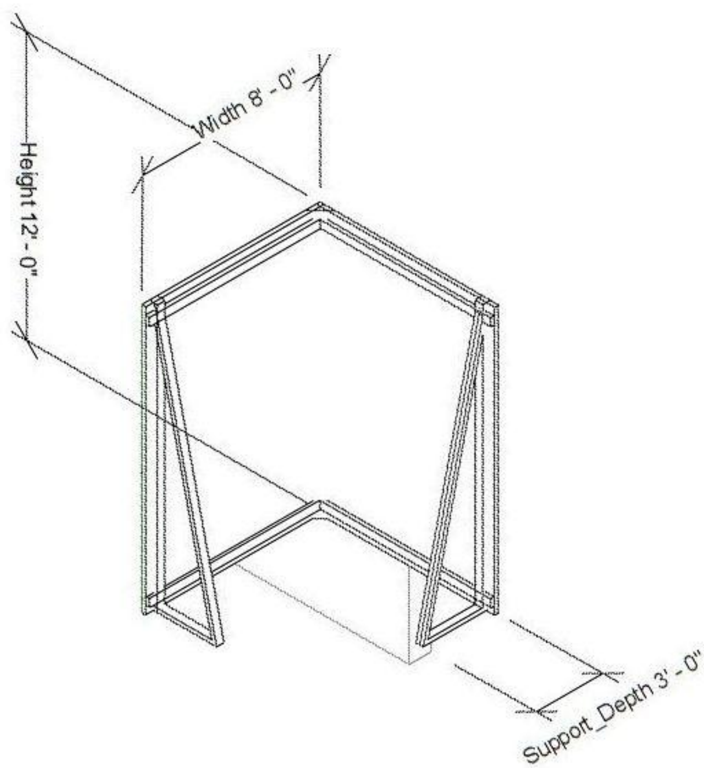


Figure 16 Vertical form for corners

Table 2 Parameters and formulas in vertical formwork-corner family

Variables	*Parameter Name*	*Description*
Height	*Height*	
Width		
Support Depth	*Support Depth*	*Controls the lower side of the triangular support*

Parameters of this family are shown in table 2.

8.3. Post Shore

This family presents a post shore (also called Prop) which is used to support slab formworks. Shore post consists of two pipe-like elements from which one slides into the other one and its height could be adjusted using a pin (which keeps the two pieces together) and an expansion bolt.

Figure 17 shows the post shore and its only parameter, Height. An image of a real post shore is also presented.

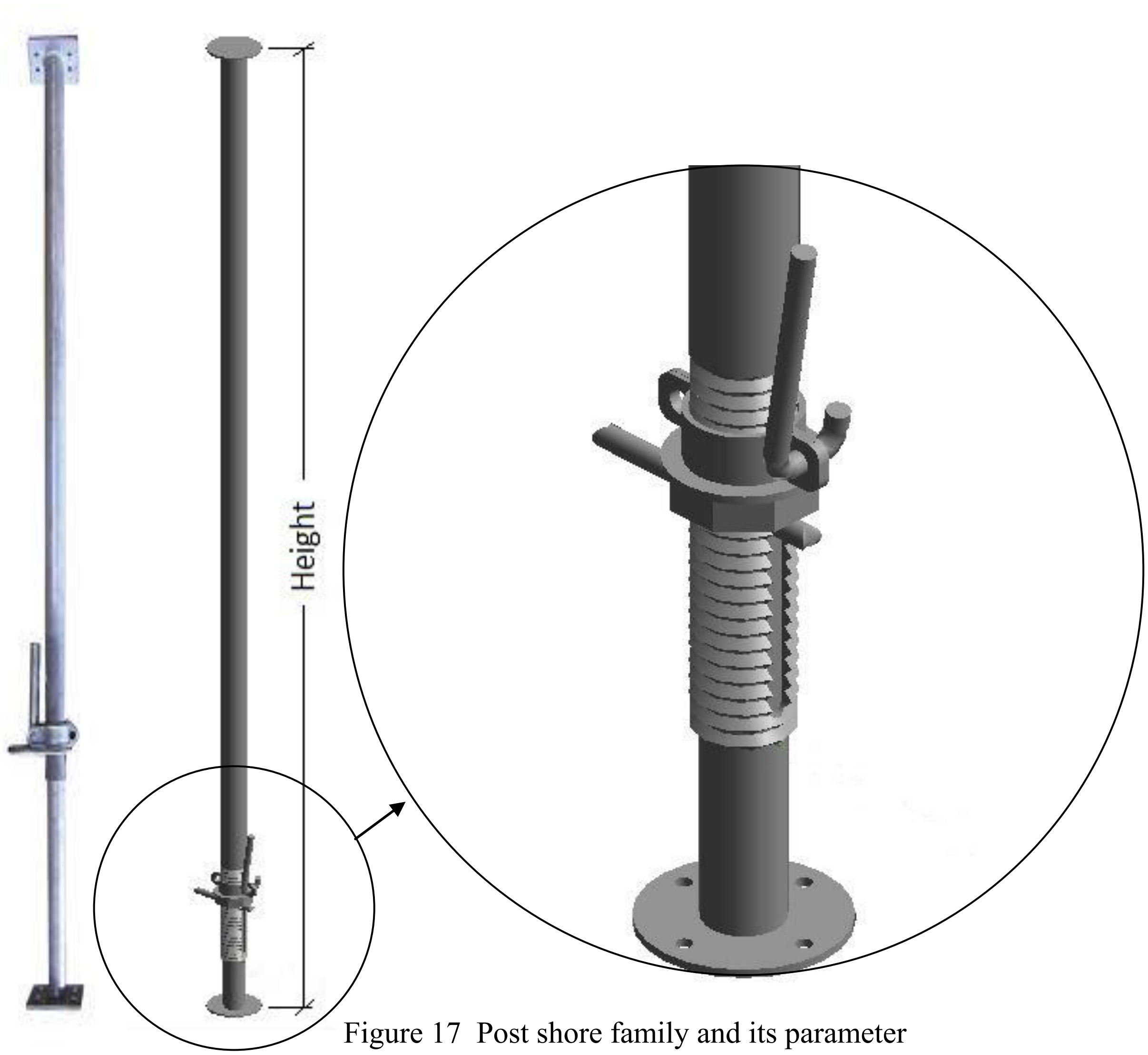


Figure 17 Post shore family and its parameter

This family has been developed on a “generic model – floor based” template, so for placing it component feature must be used and once it is placed in the model it will automatically be located on the floor surface. The best view for placing post shore is “plan view” where user can see the arrangement of wall and other vertical elements clearly. The insertion point is located on the center and Object snaps could assist user to put the component exactly in place.

Array command could be used to place a set of scaffolding components.

8.4. Steel Shoring System

This family is used to Model the shoring system which supports concrete beams formwork or heavy flooring systems. As it is shown in figure 18, this shoring system comprises of steel frames bonded together by cross bracings and an I-beam on top. There are two pairs of expansion bolts one at the top and one at the bottom of each frame which sets the height.

This shoring family is a nested family with several layers; the first layer is the steel frame and the I-beams, but the steel frame itself consists of other parts each of which is a family.

This shoring is again floor based and will be used as a component in the model. All parameters in this family have been defined as “type parameter” except the length of the I-beams which are instance parameters and are controlled by shape handles[4]. Type parameters help users define new types of the same family and change that parameter in all instances of that type at the same time. On the other hand instance parameters, allow users to set the geometry in any instance independently. The insertion point is located on the center of the post on the corners. The most convenient view for placing this object is

[4] Shape handles are grips that appear nest to the objects while selected and users can change the value of instance parameter attached to them by dragging those handles.

appropriate plan view and array feature could be used to generate a series of shorings. The number of spans is set automatically based on the total length and the span length. Following figure shows the structure of steel shoring system and its parameters. These parameters and their characteristics are also tabulated in table 3.

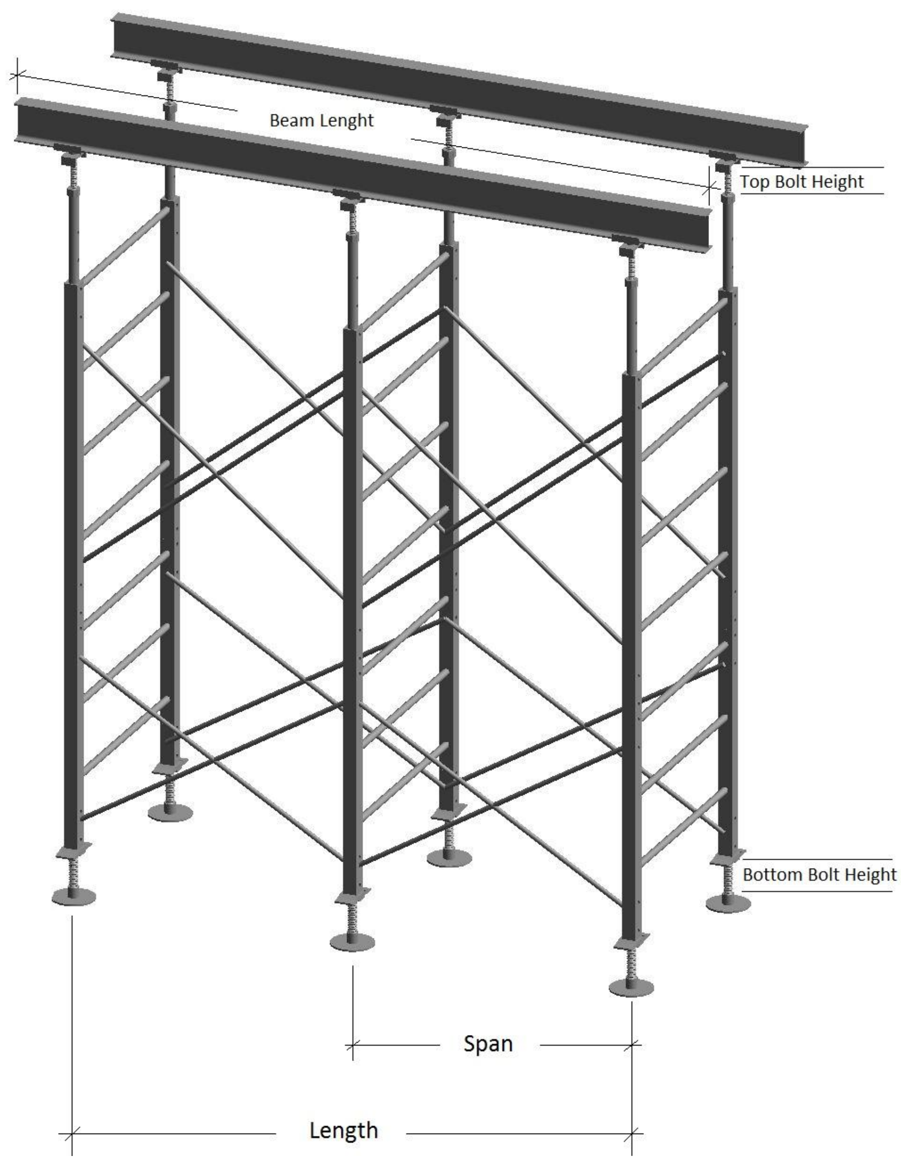


Figure 18 Steel shoring system

Table 3 Parameters and formulas in steel shoring system family

Variable	Parameter Name	Parameter Type	Description
Height	Height	Type	cannot be manipulated, just reports the total height
Length	Length	Type	
Bottom explanation bolt height	Bottom Bolt Height	Type	The total height is controlled by this parameter and "Top Bolt Height"
Top expansion bolt height	Top Bolt Height	Type	
Distance between to consequent frame	Span	Type	
I-Beam length	N/A	Instance	Controlled be shape handles

8.5. Aluminum Shoring System

This shoring system supports heavy concrete placement and by its wide platform provides a large area for complex formwork arrangements. In this project it has been used in areas in which there are a number of openings in flooring system. The structure of this shoring system includes an aluminum truss which lays on four feet each of which has an expansion bolt for height adjustment (figure 19). On top of the truss there is a set of aluminum stringers that their length is an instance parameter and could be set by shape handles for each of them independently. Other parameters are defined as type parameter. The insertion point is located on the corner of the system. The most convenient view for placing this object is appropriate plan view and array feature could be used to generate a series of this shoring. Parameters and their characteristics for this family are tabulated in table 4. Figure 20 shows the shape handles that can be used to change the length of each stringer separately.

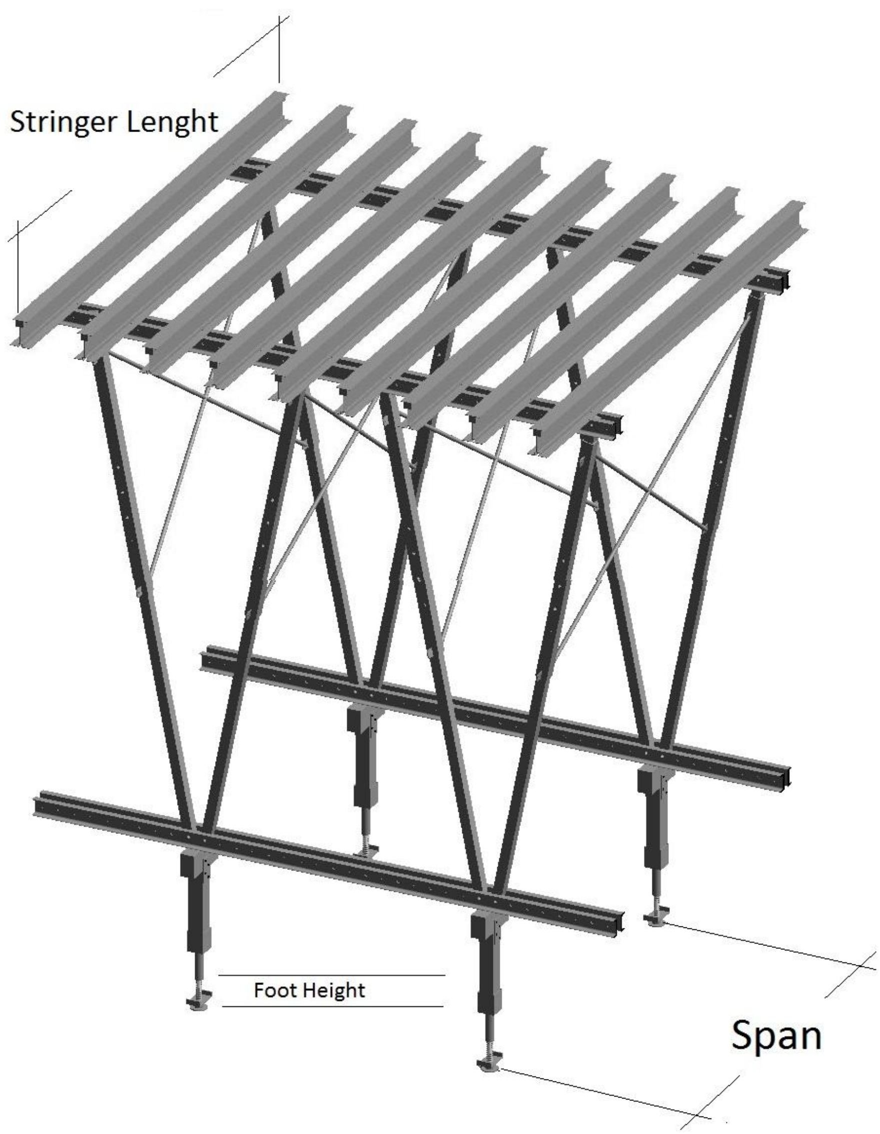


Figure 19 Aluminum shoring system

Table 4 Parameters and formulas in aluminum shoring system family

Variable	Parameter Name	Parameter Type	Description
Height	Height	Type	cannot be manipulated, just reports the total height
Bottom Expansion bolt height	Foot Height	Type	The total height is controlled by this parameter
Span	Span	Type	
Stringer length	N/A	Instance	Controlled be shape handles

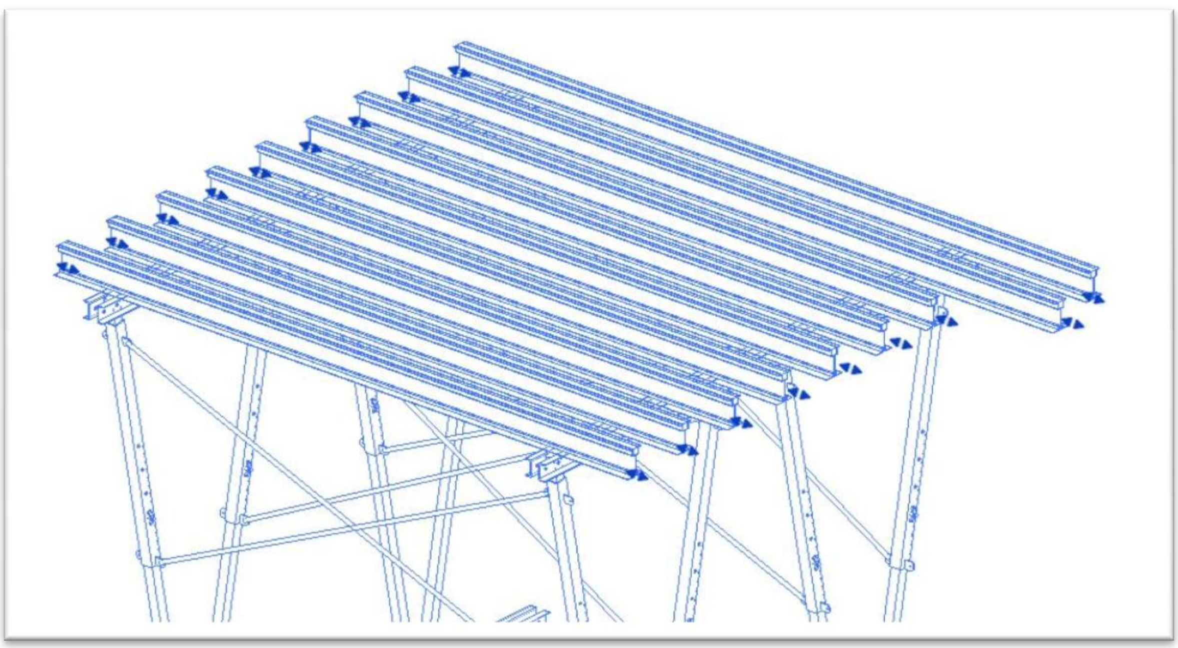


Figure 20 The length of each stringer on aluminum shoring system can be manipulated independently

8.6. Pan Slab Formwork

This steel formwork unit is used to from one-way ribbed slabs (slab and joist). A set of these forms covers the entire slab and the whole system rests on post shores. Two parameters control the geometry of this form: 1) width and 2) span. Both of these parameters are instance parameters and can be controlled by shape handles. Having all parameters as instance, the best way to place this family in the model is to place it in plan view and set its elevation to desired value then in elevation or section view set it in contact with the joists with the help of shape handles and snap feature. Figure 21 shows the structure of this steel shoring system and its parameters. These parameters and their characteristics are also tabulated in table 5.

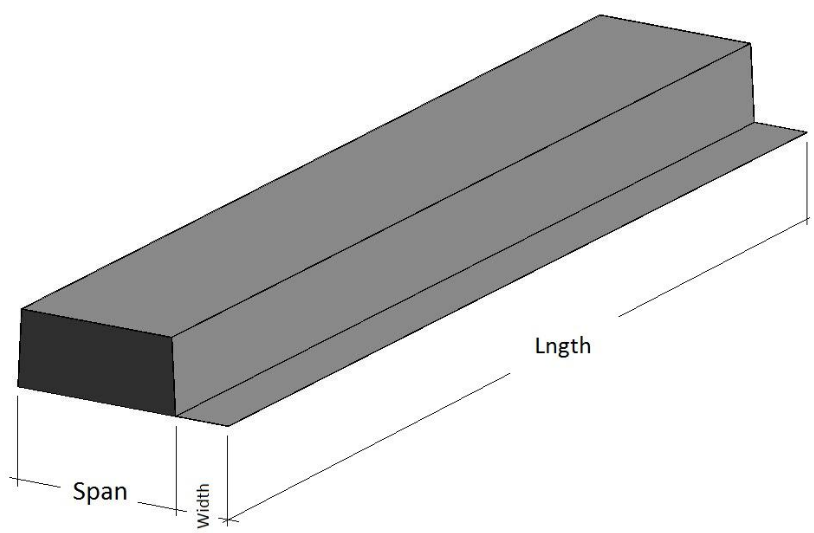


Figure 21 Pan slab formwork

Table 5 Parameters and formulas in pan slab formwork family

Variable	Parameter Name	Parameter Type	Description
Span	Span	Instance	Can also be controlled by shape handles
Width of bottom part of the joist	Width	Instance	Can also be controlled by shape handles
Length	Length	Instance	Can also be controlled by shape handles

As you can see in figure 22, sides are inclined and the reason is that the joists have a trapezoidal cross section. Another family with square sides has been created to model forms located at the end of the slab where formwork is suppose to form the beam with a rectangular cross section. Following figure shows how a system of formworks shapes the trapezoidal joists

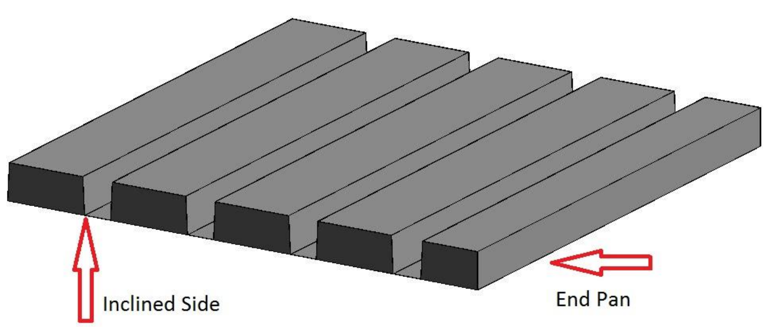


Figure 22 A set of formworks forms the trapezoidal joists

9. METHODOLOGY OF TESTING HYPOTHESIS

Having all required families modeled, and all temporary structures' building blocks in hand, know shoring systems and formworks can be placed in a Revit model using two different methods: 1) using predefined families and 2) by in-place mass feature. While placing the objects, modeling time will be recorded for both methods and at the end these two times will be compared. An MS Excel spread will be used to log the time spent over each session of the work.

In this case study a medical office building project has been used to place the shoring systems and formworks. At the time that the research is being done the project is under construction on Highway 47, south west of College Station as a part of Texas A&M University campus. The structural Revit model of the building is provided by Skanska which acts as the general contractor, and Formworks and shoring systems layout is provided by Baker Concrete Construction which is the concrete sub-contractor of the project in 2D format.

This four story building has a reinforced concrete structure whit an area of 30500 sqft on each level. The structure consists of square columns and beams and trapezoidal joists. A five inch slab covers the trapezoidal joist system which in turn carries the load to beams and columns. Different views of the structural model are shown in figures 23, 24, and 25.

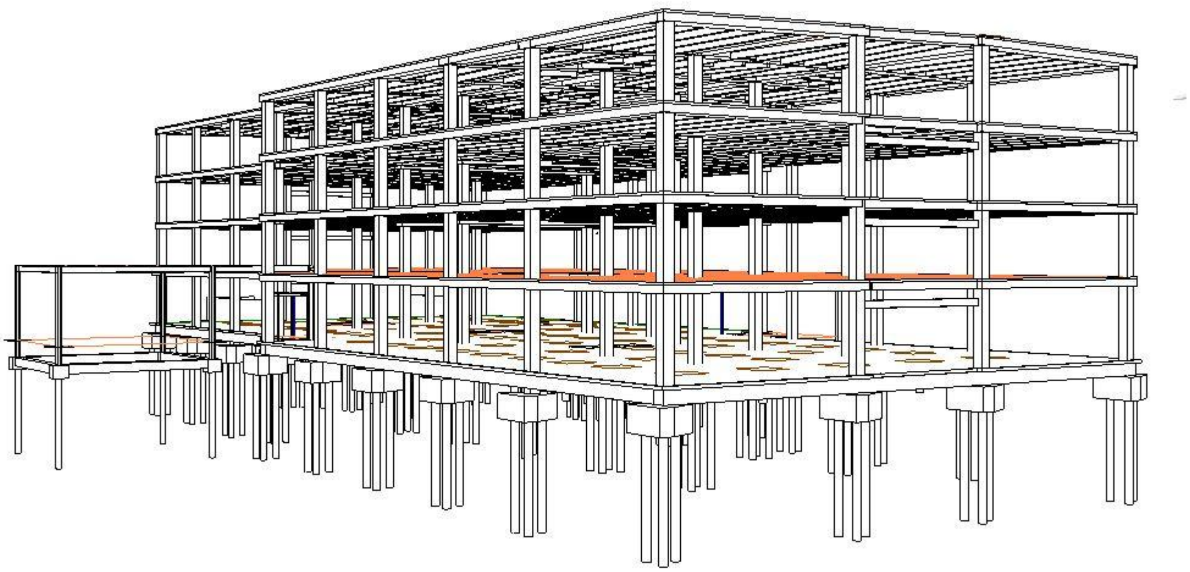


Figure 23 Structural model, created in Revit

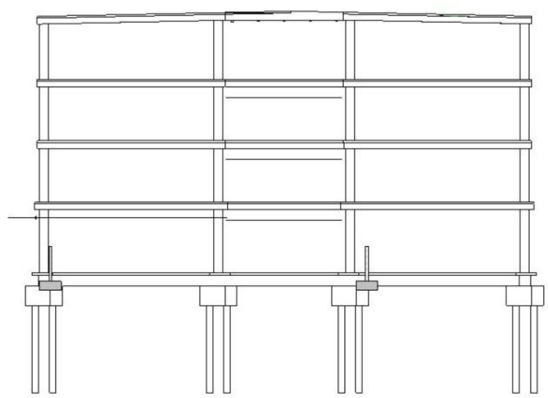


Figure 24 East elevation view of the model

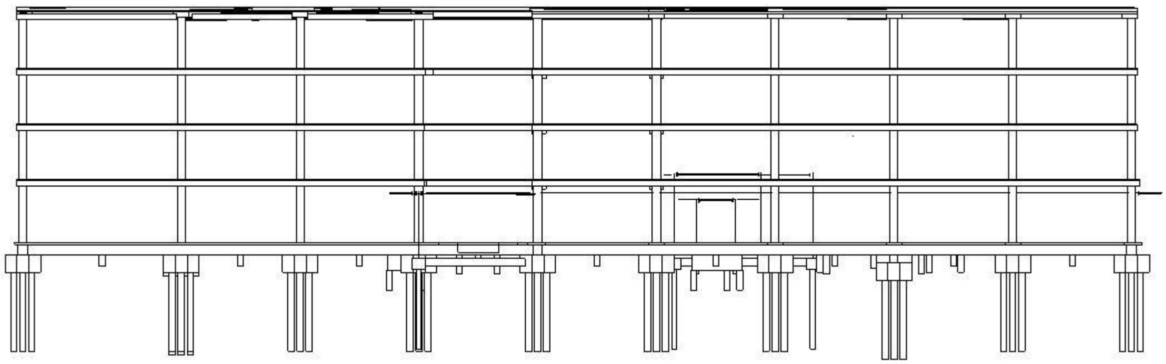


Figure 25 North elevation view of the model

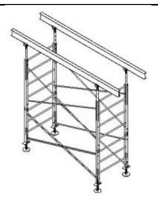
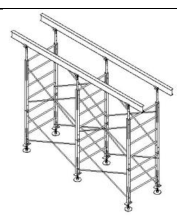
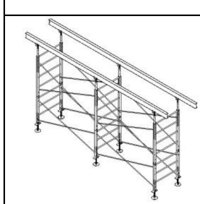
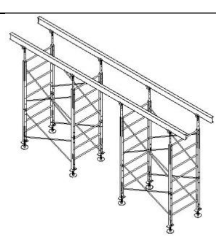
To place the temporary structure consisting of shoring system and formworks, the east part of the first level (which provides support for the second level) is used. To place the temporary structure faster and more accurately the shoring and formworks layout is (which is a DWG file) imported into the Revit model using “Link Cad” feature. Since “Snap” feature in Revit recognizes different entities in Linked DWG file, the exact location for each family could be accurately identified while placing them in the model. In other word each family is placed exactly on its 2D projection presented by linked DWG file.

9.1. Placing Predefined Families

As mentioned before the shoring system includes steel shoring, aluminum truss shoring and steel post shores. The process of placing shoring starts with steel shoring.

The steel shoring system family is defined in a way that could generate all different types of this shoring system in the model by changing the parameters. As mentioned before, these parameters are all defined as “type parameter” (except the length of the I-beam on top of the frame) and once they are set to represent a certain configuration they can be saved as a different type of the same family. Table 6 shows the different types generated using this family.

Table 6 Specification and quantity take off for different types of steel shoring system

Preview	Name	Length	Height	Span	Number of spans	Quantity
	1 X 10 ft	20 ft	13’ 7 ¼”	10’	1	19
	2 X 7ft	14ft	13’ 7 ¼”	7’	2	3
	2 X 7ft short	14ft	12’ 10 ¼”	7’	2	8
	2 X 10ft	20ft	13’ 7 ¼”	10’	2	5
	3 X 7ft	21ft	13’ 7 ¼”	7,	3	6

There were only five instances that couldn't match with developed family. In these cases, subfamilies which have been used as building blocks of steel shoring system and have been nested into it, used to model these elements. These building blocks are "I-Beam", "shoring system-no beam" and "steel shoring frame". Figure 26 shows the model with steel shoring system placed in it. It also shows the locations in which family building blocks have been used separately instead of the steel shoring system family. Since these building blocks are families themselves, their parameters could be set to meet the requirements in the model.

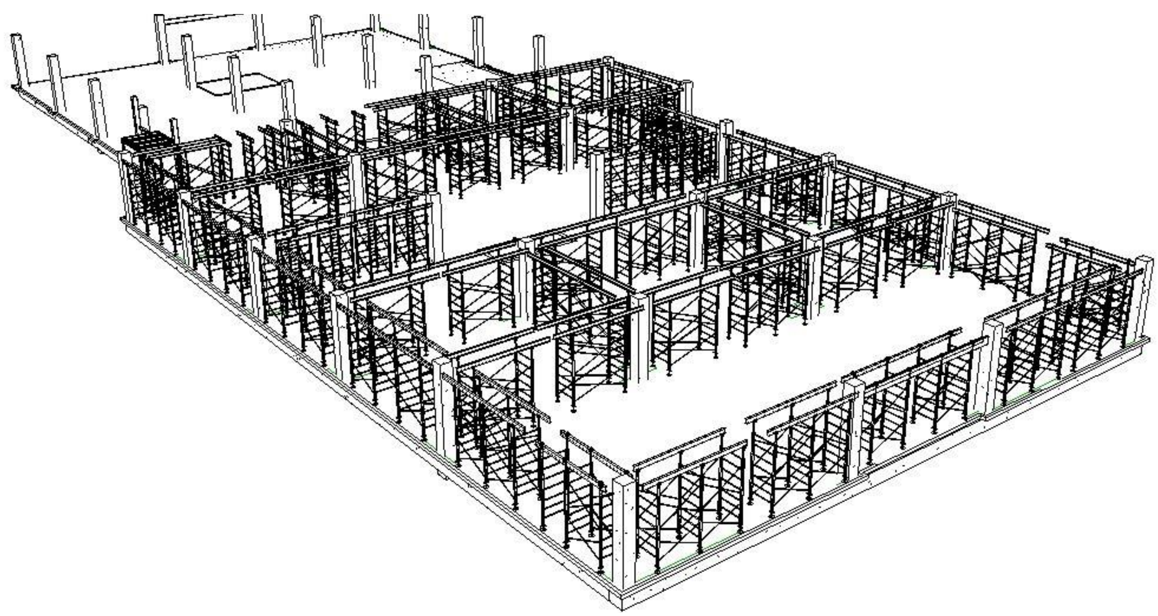
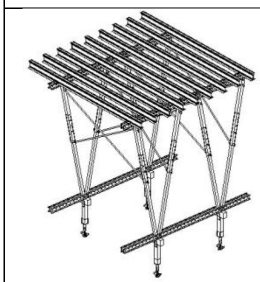


Figure 26 Revit model with steel shorings

In the next step aluminum truss shores are placed. According to shoring layout, two different types of this family could be recognized which vary by their width. Table 7 shows the different types generated using the same family and their quantity take off. Figure 27 shows the model after placement of these types in the model.

Table 7 Specification and quantity take off for different types of aluminum shoring system

Preview	Name	Length	Height	Span	Quantity
	7ft-7in	12ft	14’ 2 ¼”	10’ 10”	16
	10ft-10in	12ft	14’ 2 ¼”	7’ 7”	8

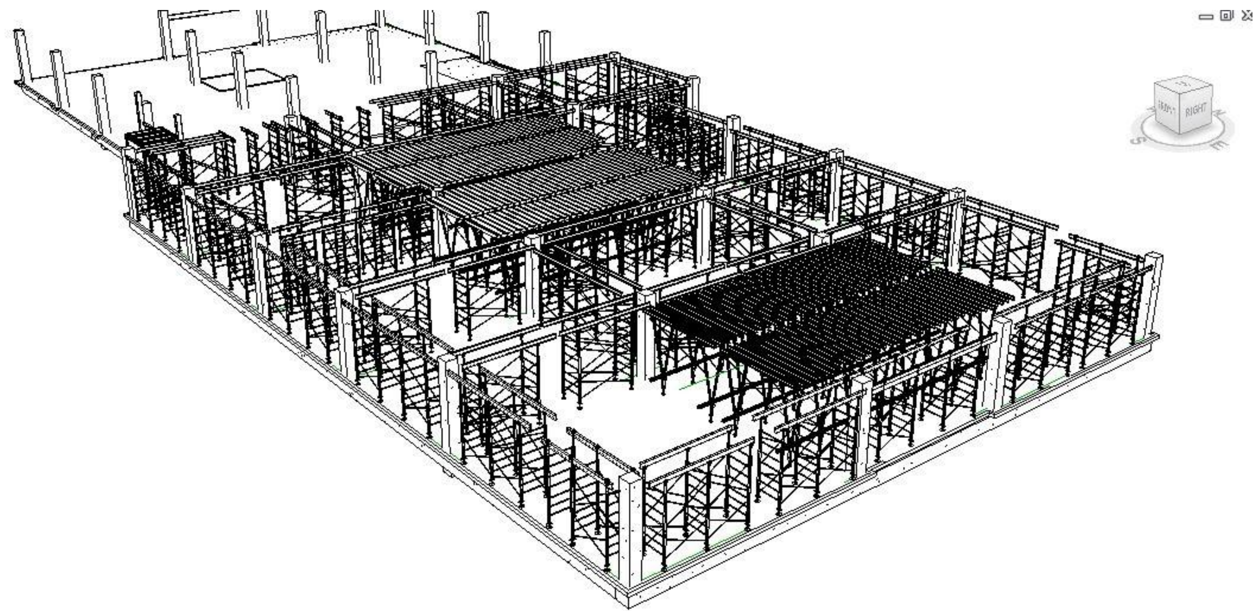


Figure 27 Revit model with steel and aluminum shoring

Now the post shores will be placed in the model as the last part of shoring layout. According to the layout a group of these shores support pan formworks in each span. Their spacing is consistent in each group. That is way array feature is used to place them at a time and together in each span. They all have the same height and they are of the same type. Quantity take off for this family is shown in table 8. Figure 28 shows the model with the props.

Table 8 Specification and quantity take off for post shore

Preview	Name	Height	Quantity
	Prop	14’ 3”	181

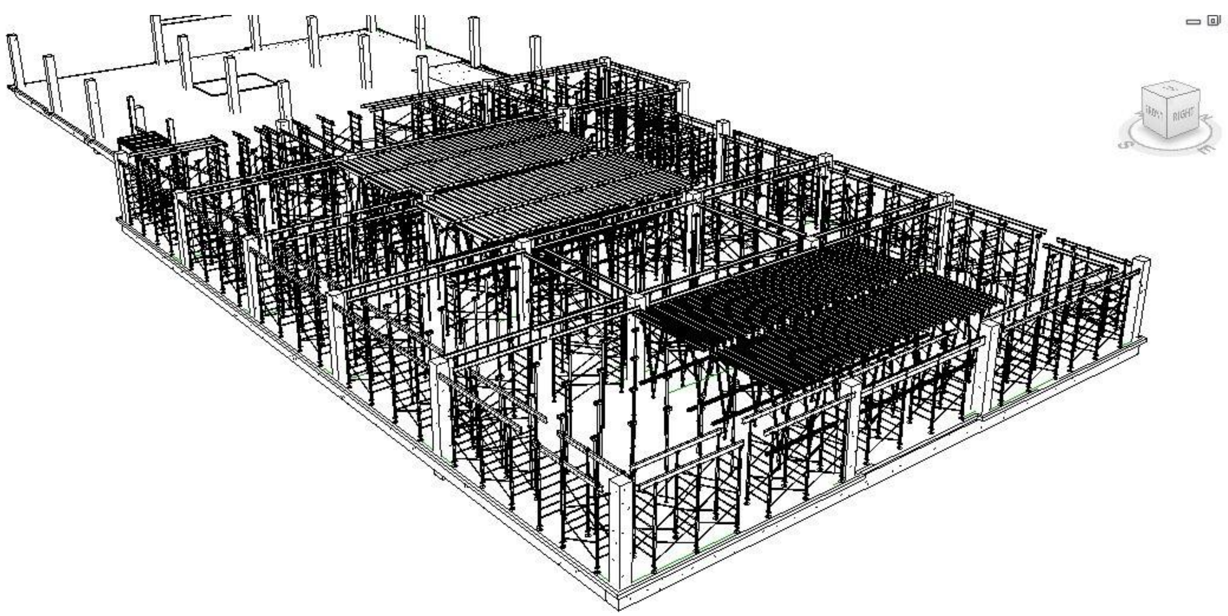


Figure 28 Revit model with all different types of shorings

As the modeling process goes on the obstacles, suggestions for improvement, advantages and disadvantages are written down as “lessons learned” during the procedure.

As the layout shows a wooden platform on top of the shoring frames shapes the bottom part of the beams formwork, while the rest of the floor system (slabs and joists) are placed into a steel pan laid on group of post shores. The outer side of the beams is also formed with vertical wooden formworks out of ply and lumber.

Investigating different formworks used in these projects shows that it is not reasonable to model all of them by customizable families. In other word, the level of complexity and variation of their geometry from one location to another makes it unreasonable and probably impossible to encapsulate all different formworks in a few parametric families.

In these research parametric families created to model the pan and the vertical formwork modules. The rest of the formwork system is modeled using existing elements in Revit.

According to plan and elevation view in shoring and formworks layout, the wooden platform consists of a series of transverse elements which are 2X4 or 4X6 lumbers at the bottom with a series of longitudinal 2X6 lumbers on top of it. A ¾” plywood lies on top of the lumbers and forms the surface of the formwork. To model the lumbers “Wall” element is used. In this case height and thickness of the wall is set to represent the dimension of the lumbers. The question might be “why are we using walls?”. By using wall element it is possible to pick the lines representing lumbers in DWG background and create the wall based on that line. This makes the process of modeling faster and more accurate. Besides, controlling the orientation of wall elements is easier and their connections is more like reality compared to other elements can be used such as beam element. In figure 29 the final horizontal wooden form is shown. In figure 30 the ply is removed and the lumbers can be seen underneath.

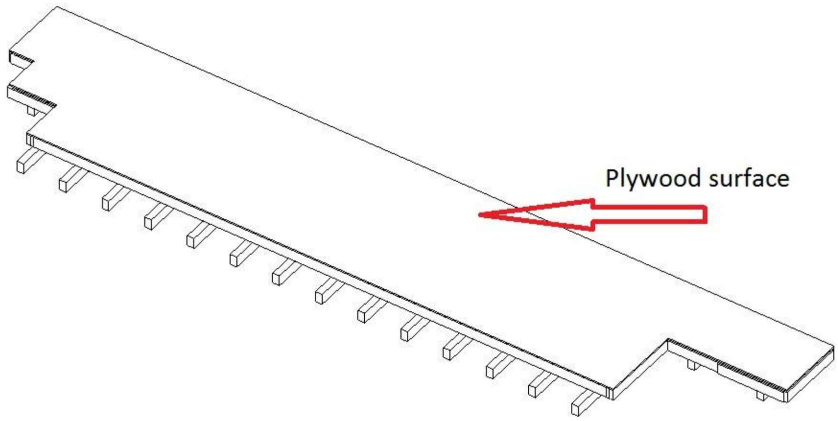


Figure 29 Wooden formwork used for bottom face of the beams

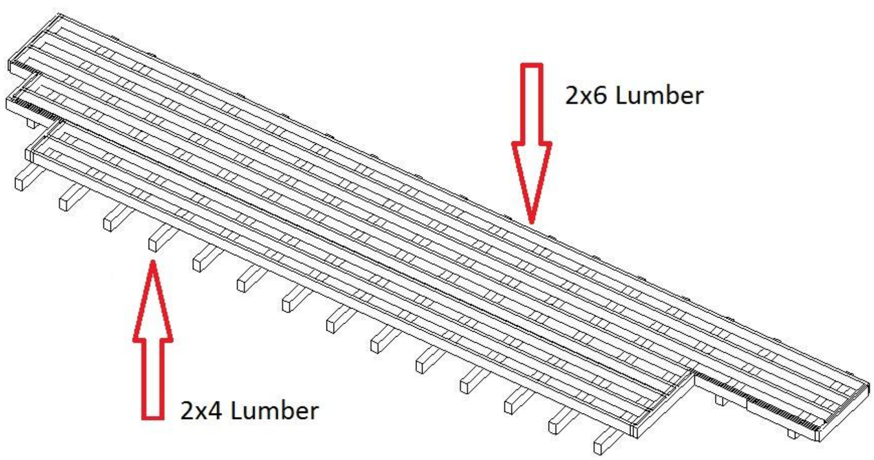


Figure 30 Lumbers under the surface of the wooden formwork

As was mentioned before, all parameters in “Pan” family are defined as instance parameter. So in order to place the pans in the model they are placed in the area between two consequent joist then in plan view their length is set by dragging the length gripes to the beams at two ends and in section view the width and span of the pan is set again by using grips. This procedure is shown in figures 31 and 32.

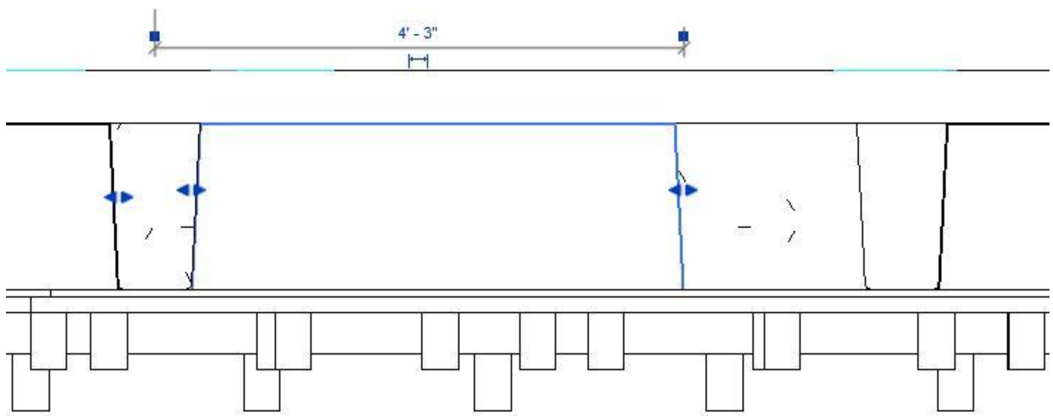


Figure 31 On section view shape handles makes it easier to set the pans span

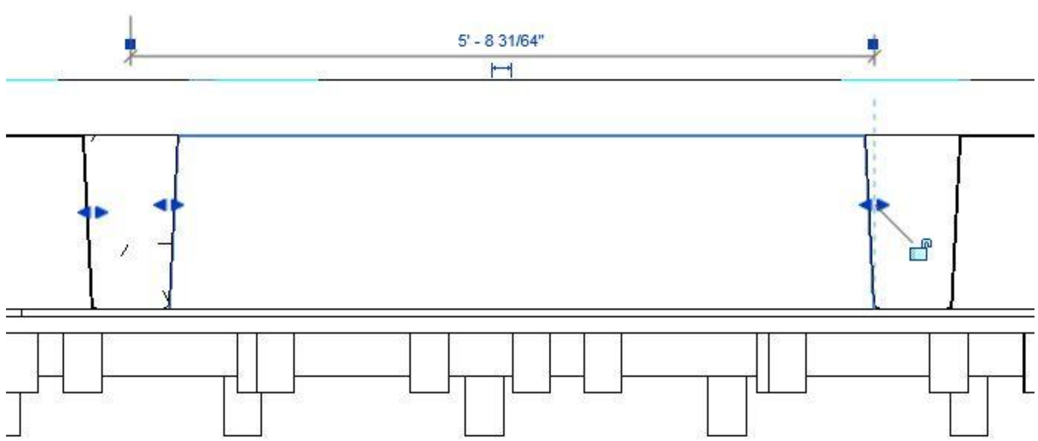


Figure 32 Object snap feature helps the surface of the form to touch the side of the joist

The last stage is to put vertical wooden formworks which support the outer side of the beams on perimeter of the floor system. Since this family is a wall-based family, it automatically gets stuck to the side face of the beam. On plan view they are placed by the beam and their length is set using length grips. Corner formwork is used to cover the corners.

The quantity take-off taken from Revit model shows that the model includes 466 ft of wooden vertical formwork with a height of 1’ 9”.

Also the model has a total area of 15,313 sqft covered by pan forms.Figure 33 shows the shoring systems and forms together in the model. The details can be seen in figure 34.

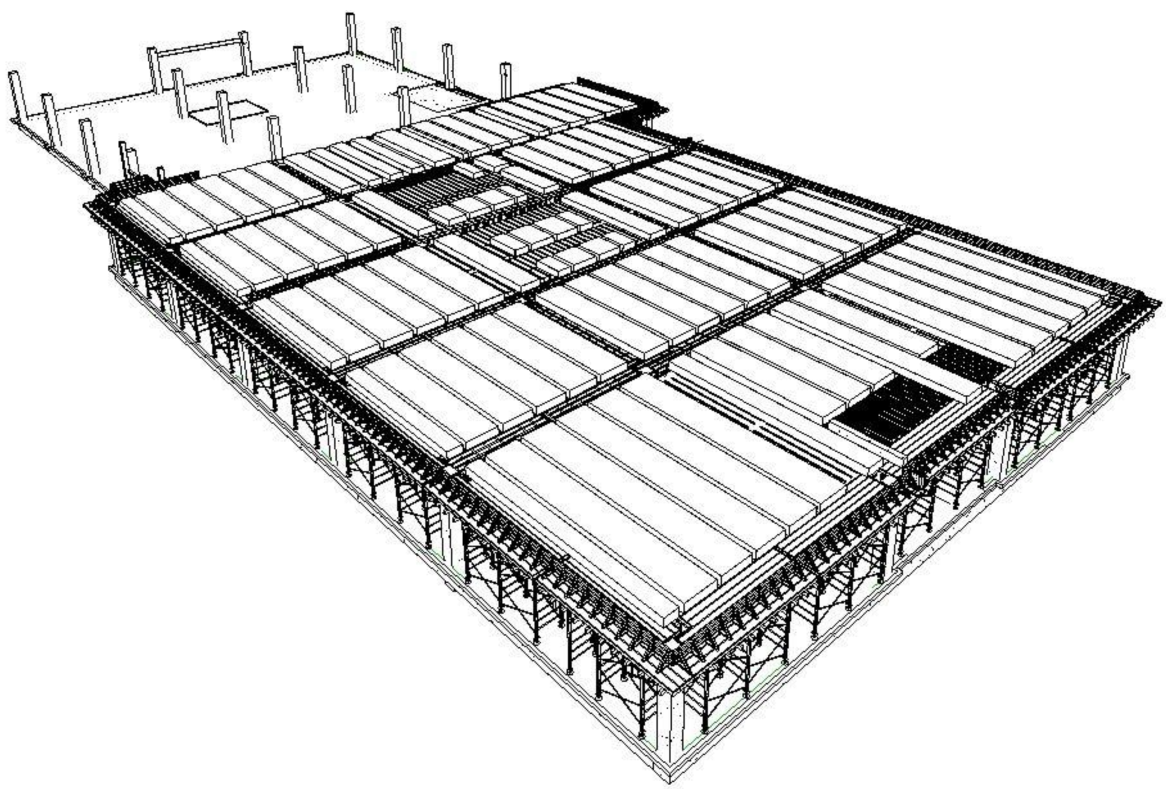


Figure 33 Revit model with shorings and forms

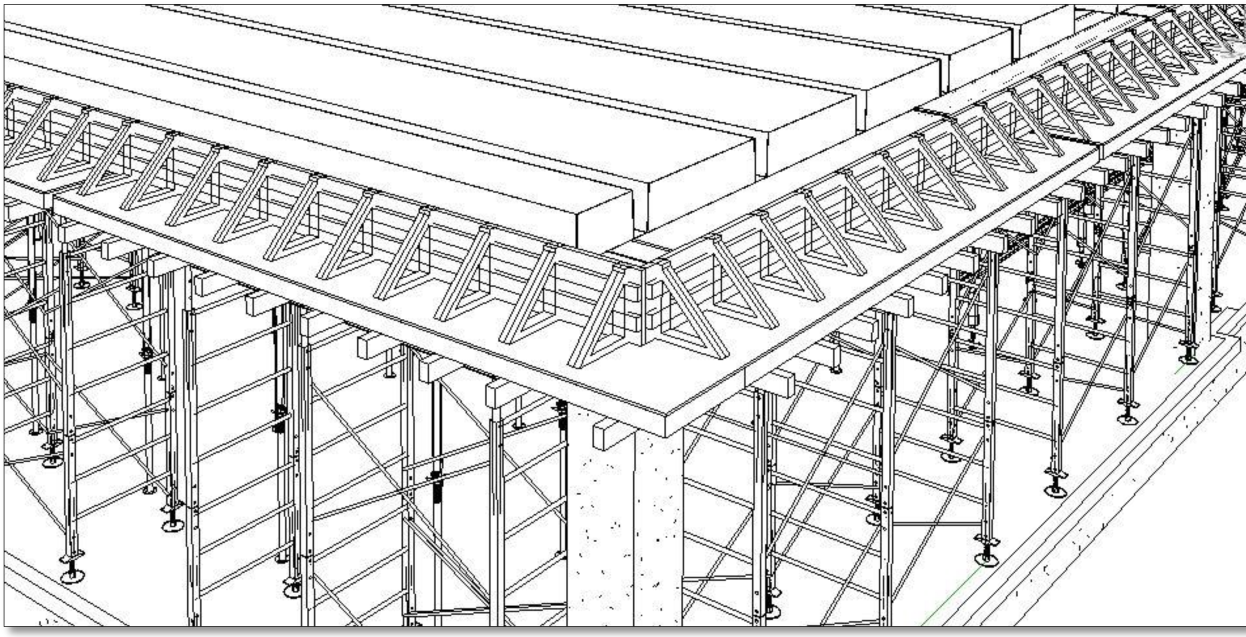


Figure 34 Revit model contains every detail for each element in the model

9.2. Placing In-Place Masses

In second part of this study the same temporary structures are placed in the model using "In-place Mass" feature in Revit. Massing feature enables the user to model objects which are customized for the project and cannot be found among ordinary elements offered by Revit. If these objects are not supposed to be parametric and they will not be used in other projects, in-place mass can be used. This feature provides a modeling environment within Revit environment in which the user can make solid objects with complicated geometry. The modeling tools in this environment resemble those of family editor environment. Since the building blocks have been created in the first part of the study, there is no need to create them again. In place massing feature allows us to load those elements into the mass as a component and put them together. This environment is used to make changes, such as manipulating the length, width, and configuration etc of the objects as the user proceeds in modeling. Figure 35 shows the building blocks of a shore frame.

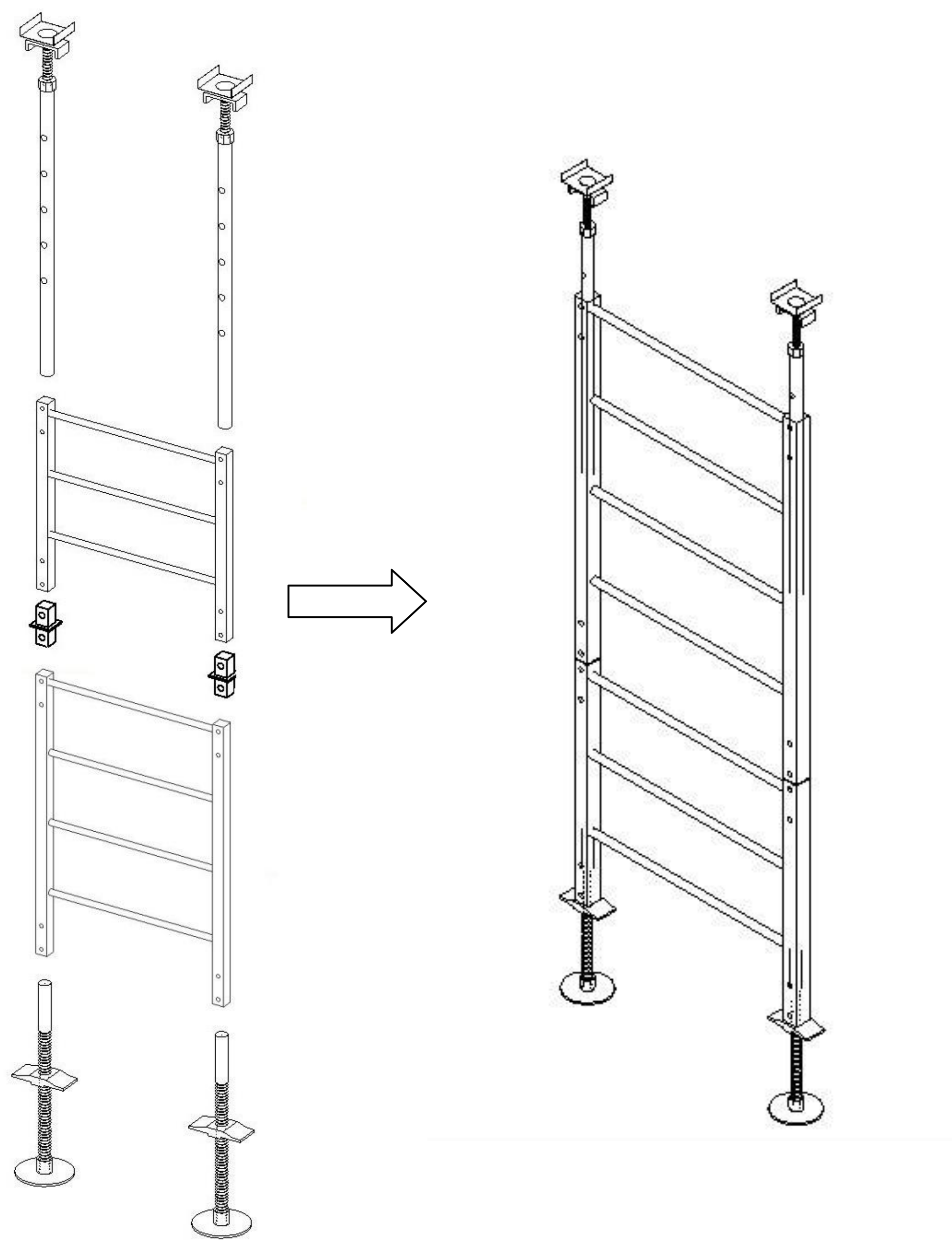


Figure 35 Building blocks of a shore frame together shape one solid object

Similar process has been done for any other temporary structure in the model. The fact that this objects are not parametric and flexible, requires the modeler to get back to the massing environment for even any small change and save the manipulated object as another mass. In this case the only feature that speeds up the process of modeling is array and copy and paste to make other instances of an existing mass.

By this approach the time spent for modeling the object itself is eliminated for both methods and comparison will be made based on the time spent for placing and manipulating the objects in the model.

In this part again DWG background is used for faster and more accurate placement of objects, and time is logged in a spread sheet.

10. OBSERVATIONS AND CONCLUSION

10.1. Time Comparison

Figure 36 compares the time spent on placing each temporary structure in both methods.

Temporary Structure	Using Predefined Families	Using Mass	Time Saved (%)
Steel Shoring System	5:52	12:17	51.9
Aluminum Shoring System	0:43	1:25	49.4
Post Shore	0:35	0:49	28.6
Horizontal Formworks	4:41	4:27	-5.24
Vertical Formworks	0:36	1:12	50.0
Pan	2:22	4:46	61.2

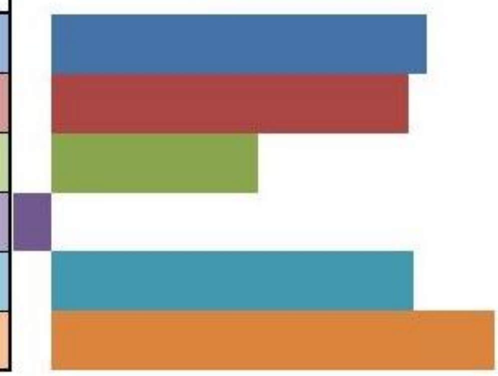


Figure 36 The time spent on placing each structure for both methods

This figure shows that using predefined parametric families has saved more or less 50 percent in time. For horizontal formworks this percentage is negative, the reason is that for modeling this particular temporary structure in both methods existing Revit elements have been used (wall and slab). The reduction in time shows the effect of training on the modeling time. In second method modeling with the same tools has been done 5.24% faster.

Among other structures, Pan has the most saved time. The reason is that the variation in dimension for this structure is more than others, so more manipulation in mass is required, on the other hand while using parametric families, as it was mentioned before; the object could easily be set up by dragging shape handles.

For post shores the difference in time spent is relatively low, this is because there was just one type of this element, and in both cases array feature used to generate several instances of the same type. Although there was no type variation, using predefined family facilitated correction of errors. Post shore family was defined as a floor-based object so its vertical alignment automatically was set to the floor surface while for the post shores in the form of mass, the work plane needed to be set before placing them.

10.2. Lessons Learned

Observations are not limited to time deference between those two methods. As I was proceeding to the modeling made notes of problems to which I encountered and kept a track of advantages and disadvantages of each of those two methods. Following presents what I learned during the process of modeling in both methods.

10.2.1. Hardware Issue

Using predefined families because of its parametric nature utilizes the hardware on a higher level, so it is more hardware depended. Especially when it comes to complicated families whit larger number of parameters and formulas, the more powerful the hardware is the more efficient parametric families work. For example in cases user decides to make some changes in the value of a type parameter, this value will change in all instances of that type. If there are other parameters which are functions of this one and a large number of instances of that type exist in the model, updating all those instances may take awhile.

10.2.2. Making Changes in the Model

In situation above I described the case in which all instances of a type get updated, but there is another situation in which the modeler tries to make a change in the entire family. For example if braces in steel shoring system have been modeled using elements with circular cross section and modeler decides to change them to square cross section (which happened in this study) while using predefined families, the only thing that the modeler needs to do is to update the family in family editor and reload it into the project. No matter what the value for different type and instance parameters are, for all types of that particular family the cross section of braces will change to square. If mass is being used, for such a change modeler has to define each mass again (because they are in-place masses). In other word making a change in one single instance of a mass will not make the same change in geometry of other instances of that mass. This implies that correcting mistakes or male changes could be done easily by utilizing families while it is a tedious task while using masses.

10.2.3. How to Structure a Family

Making a decision about which parameter should be defined as instance and which should be defined as type is an important part of structuring families. The variation in value of each parameter form one instance to another in the model is a good indicator to make this decision.. If a parameter with a particular value occurs a few times in the model it is better to be modeled as an instance parameter, but another parameter that takes a few different values in entire project should be defined as a type parameter. Doing so, enables the modeler to change that parameter in a certain type and save it as a new type. For example in this study the length of each stringer in Aluminum Shoring System family has been defined as independent instance family. This is because the configuration of these stringers varies from one instance to another. As conclusion, encapsulating an entity into a single family with different types is all about categorizing all its defining parameters into instance and type parameters.

10.2.4. One Family/Too Many Types vs. a Few Families/a Few Types

At a certain point and in a higher level modeler has to decide about how many types should be included in a single family. Creating a family capable of generating too many different types adds to the number of parameters and makes the family too complicated and large. This in turn may create a bunch of useless information in the model. In this study, I created a single family for steel shoring system, which covers all different configurations in the model except a few odd instances. But this family is 11MB itself and handling such a large family in the process of modeling is to some extent cumbersome. In this particular case I have provided this family with a parameter which indicates that the bracing is either on all spans or on every other one. This parameter caused the family to be so large. By making two different families each one satisfying one the mentioned configurations, the final product could be much smaller. Keeping the balance between number of families and number of their types is an important issue in handling families in the model efficiently.

10.2.5. Choosing Family Template

Each family is defined based on a specific template which is directly related to the function of that family. Templates are actually a short cut during the process of defining a new family. But the more important advantage of using templates is that they specify how and where to place families in the model. In this study all shoring systems have been defined based on a "floor-based generic model" template. This means users have to use component feature and its interface to place them in the model and they can only be placed on a floor system. This makes the placement process much more real , for example if modeler by mistake tries to put the shoring on an opening in the floor, the application does not allow him or her to do so, and modeler has to make required provisions (for example cover the opening with another object) and then place the object. In conclusion without choosing the right template utilizing full capacity of a parametric family is not possible.

Moreover this feature justifies the use of predefined families even for objects which are not very complex.

10.2.6. Using Mass and Size of the Model

The model created using mass object is 50% larger than the model created with parametric families. The reason could be that for each instance of a mass there is an independent data base embedded in the model no matter if they are completely identical or not, while each instance of a family shares the common properties with other instances in the embedded data base created for that specific family.

10.2.7. Using Mass and Making Changes in the Model

As it was mentioned before by using mass modeler will have a hard time making changes in the model. This change does not need to be phenomenal to get the modeler in trouble. Even if modeler needs to make a small change in all instances of a particular mass the same procedure must be taken. The reason is that changing one instance of a certain mass, does not change other instances of that. In this case modeler will have to make the same change on every instance or easily delete them all and create them again. In other word different instances of a mass are as independent as two totally different mass.

11. FUTURE RESEARCH

In order to have a useful visualization of construction process, a spotless schedule and efficient jobsite facility management it seems to be vital to have logistics in the model. When it comes to reinforced concrete structures modeling formworks and relevant equipments plays an important role in work flow and without this part, access and work space for the labors, safety issues, sequence of placing formworks (and consequently concrete) could not be seen in the model. Since contractors usually provide formworks for a small part of concrete elements and use the same formworks in sequences as the project progresses, providing accurate visualization of this temporary equipment could give contractors a more accurate estimation of the number of required formworks in the job site. But the problem is that usually because of lack of predefined objects in BIM applications they are not able to model them in a reasonable time so it makes them ignore this part of the model. This effort may convince software vendors that it is really helpful to have such objects and eventually provide contractors with a more complete set of site logistic objects in their computer models.

Once it is approved that using predefined families for temporary equipment expedites the process of modeling such equipments in BIM, next stage could be developing these families as a tool-based family in the application. In this case rather than placing families and setting their parameters, users can have a more dynamic and more user friendly environment to customize and place the objects in the model, exactly like what they do when they add ordinary objects like walls and floors to the model. Apparently such a feature requires programming within REVIT environment.

As another effort different equipments with more real-like appearance and parameters could be made, which are more representative of the existing equipments in the industry. By doing so, users can download these objects from online libraries all around the world and use them in their models.

REFERENCES

Bratton, J. (2009). “Making the transition from CAD to BIM.” *Electr. Constr. Maint.* 108(3), C26-C31.

Chau, K. W., Anson M., and Zhang, J. P. (2004). “Four-dimensional visualization of construction scheduling and site utilization.” *Journal of Construction Engineering and Management* , 130(4), 598-606.

Hijazi, W., Alkass, S., and Zayed, T. (2009). “Constructability assessment using BIM/4D *CAD* simulation model.” *AACE International Transactions,* 4, 1-14.

Ireland, B. (2009). “Barriers to BIM”. *Electr. Constr. Maint.* 108(3), 22-26.

Klemens, T. (1999). “The truth about BIM.” *Concrete Construction Magazine,* 53(12), 31-2, 34, 36, 38.

Klemens, T. (2009). “Beyond constructability.” *Public Works Magazine* 140(12), 30-33.

McKinney, K., Fischer, M., and Kunz, J. (1998). “Visualization of construction planning information.” *Int. Conf. on Intelligence User Interface,* ACM, San Francisco, CA, 135-138.

Meadati, P. (2009). “BIM extension into later stages of project life cycle.” *Associated Schools of Construction 45th Annual International Conference*, Gainesville FL, 121-129.

Middlebrooks, R., and Behrens, L. L. (2009). “A new collaborative workflow: building information modeling.” *HPAC Engineering Magazine,* 81(11), 12-15.

Post, N. M. (2009). “3D modeling spurs architect to reorganize divisions of labor.” *ENR: Engineering News-Record*, 262(14), 30-31.

Sullivan, C. C. (2005). “Brace for BIM.” *Architecture*, 94(4), 77-80.

Van Hampton, T. (2010). “Hospital owner's digital mandate gives builders a dose of reality.” *ENR: Engineering News-Record*, 264(18), 80-80.

VITA

Name: Parsa Sabahi

Address: Research Unit of Computer Integrated Construction,
J.K. Williams Building, Room 010, Texas A&M
Univarsity
College Station, TX 77843

Email Address: paarsab@yahoo.com

Education: B.S., Civil Engineering, Iran University of Science and Technology, 2003
M.S., Construction Management, Texas A&M University, 2010