DESIGN DIGITAL RECIPES

Facilitating collaboration for façade design

Trevor Stephen Lewis, PhD, PE
Walter P Moore
slewis@walterpmoore.com

Lukasz Blonski
Walter P Moore
lblonski@walterpmoore.com

Jessica Chen, PE
Walter P Moore
jchen@walterpmoore.com

ABSTRACT

Precision in digital workflow is necessary to deliver facade projects where there is a high design aesthetic or structural performance requirement. This process was essential in the design of the structural skin of a stadium proposed for Minnesota United MLS team. A philosophy of “digital recipes” was used for documentation, collaboration, and scalability of digital workflow among the design team. Efficient and effective digital communication between team members allowed the project to constantly evolve based on design pressure from the architect.

For this project, the overall digital workflow was condensed into four main recipes. Each recipe contained information on fundamental steps to generate either engineering or design results. This paper will go in depth into each of the recipes and its relationship to the project, describing the value it brought to the project. It will also present connections between a recipe and the team member working on it. This paper will describe a generation process of two types of models: one for analysis and one for design. Each of these models played a key role in overall development of the project. Creation of these models depended on the digital recipes designed by the team.

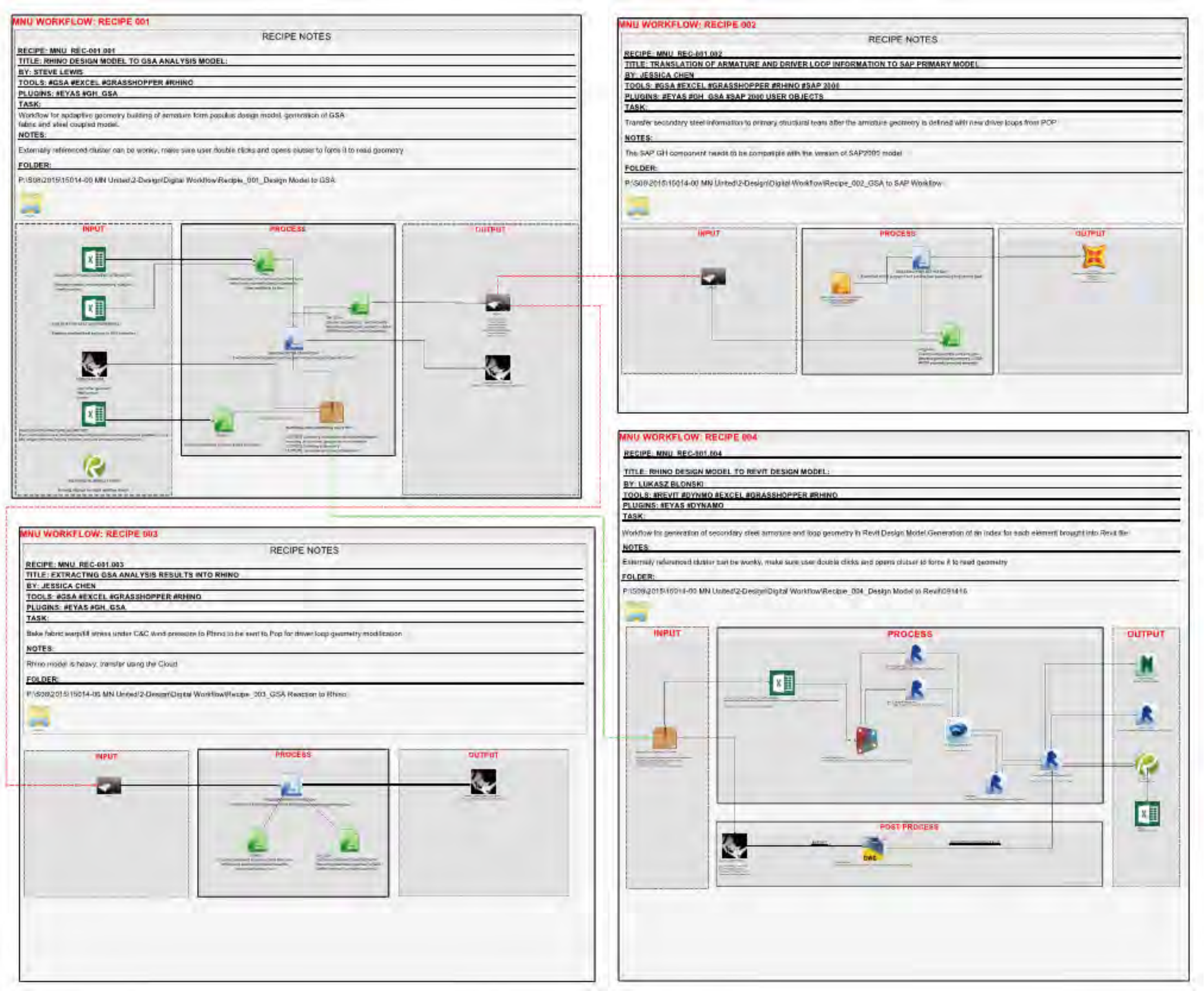
The digital approach also provided great value downstream to the fabrication process. The level of precision of the information embedded within the “master menu” of digital recipes made the production of a digital fabrication model much quicker, and lead to successful collaboration between the design and engineering team at Walter P Moore and the architect Populous through the use of transparent digital tools and their distribution among the team members.

KEYWORDS

parametric workflow, computational design, design processes, design optimization, collaboration

INTRODUCTION

Digital workflow has been assisting designers and engineers to conquer intricate design and bring stunning architecture to urban skylines. This process should be more of a team collaboration than an individual effort, and it is important to document the workflows for repetitive use and understanding each team member's work in delivering the results for project management. 'yEd Graphic Editor' is a powerful desktop application that can be used to quickly and effectively generate high-quality diagrams. During this project case study, "digital recipes" generated using yEd documented the workflow input, process, and output, and which software platforms and tools were used. The recipes also stated clearly the team member working on it, the task that the workflow accomplished, and the folder hyperlink where the work was saved. Four of the core recipes are discussed here (Figure 1).



MNU WORKFLOW: RECIPE 001

RECIPE NOTES

RECIPE: MNU_REC-001.001
TITLE: RHINO DESIGN MODEL TO GSA ANALYSIS MODEL:
BY: STEVE LEWIS
TOOLS: #GSA #EXCEL #GRASSHOPPER #RHINO
PLUGINS: #EYAS #GH_GSA
TASK:
NOTES:
FOLDER:

MNU WORKFLOW: RECIPE 002

RECIPE NOTES

RECIPE: MNU_REC-001.002
TITLE: TRANSLATION OF ARMATURE AND DRIVER LOOP INFORMATION TO SAP PRIMARY MODEL
BY: JESSICA CHEN
TOOLS: #GSA #EXCEL #GRASSHOPPER #RHINO #SAP 2000
PLUGINS: #EYAS #GH_GSA #SAP 2000 USER OBJECTS
TASK:
NOTES:
FOLDER:

MNU WORKFLOW: RECIPE 003

RECIPE NOTES

RECIPE: MNU_REC-001.003
TITLE: EXTRACTING GSA ANALYSIS RESULTS INTO RHINO
BY: JESSICA CHEN
TOOLS: #GSA #EXCEL #GRASSHOPPER #RHINO
PLUGINS: #EYAS #GH_GSA
TASK:
NOTES:
Rhino model is heavy; transfer using the Cloud
FOLDER:

MNU WORKFLOW: RECIPE 004

RECIPE: MNU_REC-001.004
TITLE: RHINO DESIGN MODEL TO REVIT DESIGN MODEL:
BY: LUKASZ BLONSKI
TOOLS: #REVIT #DYNMO #EXCEL #GRASSHOPPER #RHINO
PLUGINS: #EYAS #DYNAMO
TASK:
NOTES:
FOLDER:

Figure 1: Master menu for digital recipes.

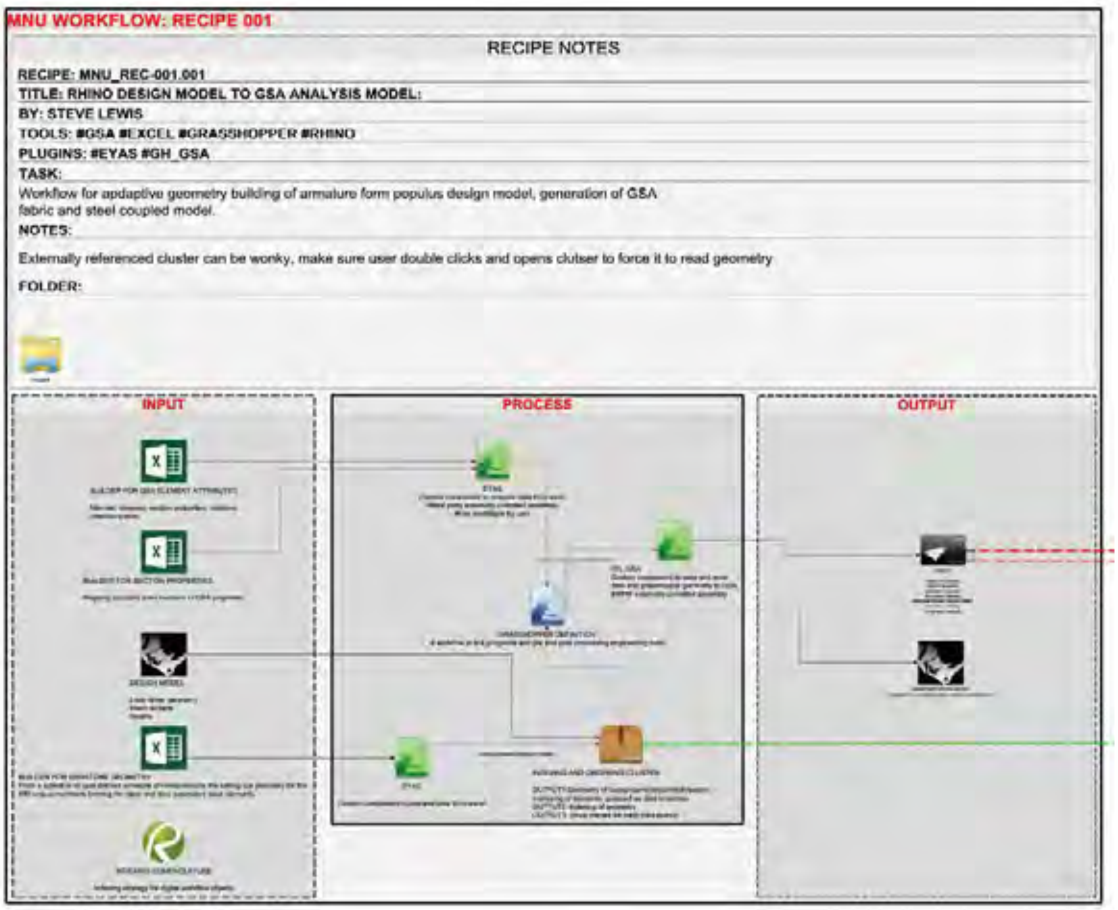
DIGITAL RECIPES

A digital recipe is a set of instructions needed to perform a specific task for a project. It can also be described as a visual history of steps needed to produce design or analysis results. Each recipe maps the software types used to conduct and finish a specific scope of digital work. The idea of having a documented recipe was not derived in the initial phase of the project, but rather the need for it became more evident due to the high level of collaboration and a desire for increased efficiency among the design and engineering team. Once the recipe is finalized, the output becomes an input for the next recipe in the workflow. Thus the process of design iteration becomes almost automated.

The “master menu” (Figure 1), which consists of the four recipes, becomes the digital “engine” of the Minnesota United project. Each recipe is responsible for a specific scope of the digital work and generation of results which affect the subsequent steps in the workflow. The set of tasks varies from geometry generation, indexing, attribute assignment, and engineering analysis to documentation. Recipe 001 starts the secondary steel and fabric design process by generating a design and analysis model with assigned indexes. Recipe 002 translates Rhino geometry into SAP and GSA structural analysis software, and assigns attributes for each element. Recipe 003 becomes a platform to then visualize engineering results from SAP and GSA in Rhino. Recipe 004 produces a documentation package.

The outline of the recipes creates clear understanding of the process for each member of the project team and increases efficiency in collaboration. This digital set-up also allows for efficient management and distribution of critical files amongst the team.

RECIPE 1 - SETTING UP SECONDARY STRUCTURE GEOMETRY PARAMETRICALLY



MNU WORKFLOW: RECIPE 001

RECIPE NOTES

RECIPE: MNU_REC-001.001
TITLE: RHINO DESIGN MODEL TO GSA ANALYSIS MODEL:
BY: STEVE LEWIS
TOOLS: #GSA #EXCEL #GRASSHOPPER #RHINO
PLUGINS: #EYAS #GH_GSA
TASK:
Workflow for apdaptive geometry building of armature form populus design model, generation of GSA fabric and steel coupled model.
NOTES:
Externally referenced cluster can be wonky, make sure user double clicks and opens clutser to force it to read geometry
FOLDER:

Figure 2: Recipe 1.

Façade design was the key architectural aspect for the Minnesota United Stadium. The secondary support structure behind the façade is a system made up of HSS rolled steel loops undulating around the primary stadium structure. PTFE fabric spans between the loops creating the main architectural vision of the façade design. The steel loops are attached to the primary columns by a system of struts and rakers depending on their position in space. Layout of the primary columns follows a set of 48 grids which became a major rule for the digital data distribution and the parametric set-up. Each grid line indicates a branch of information and an attribute associated with the structural elements within that zone.

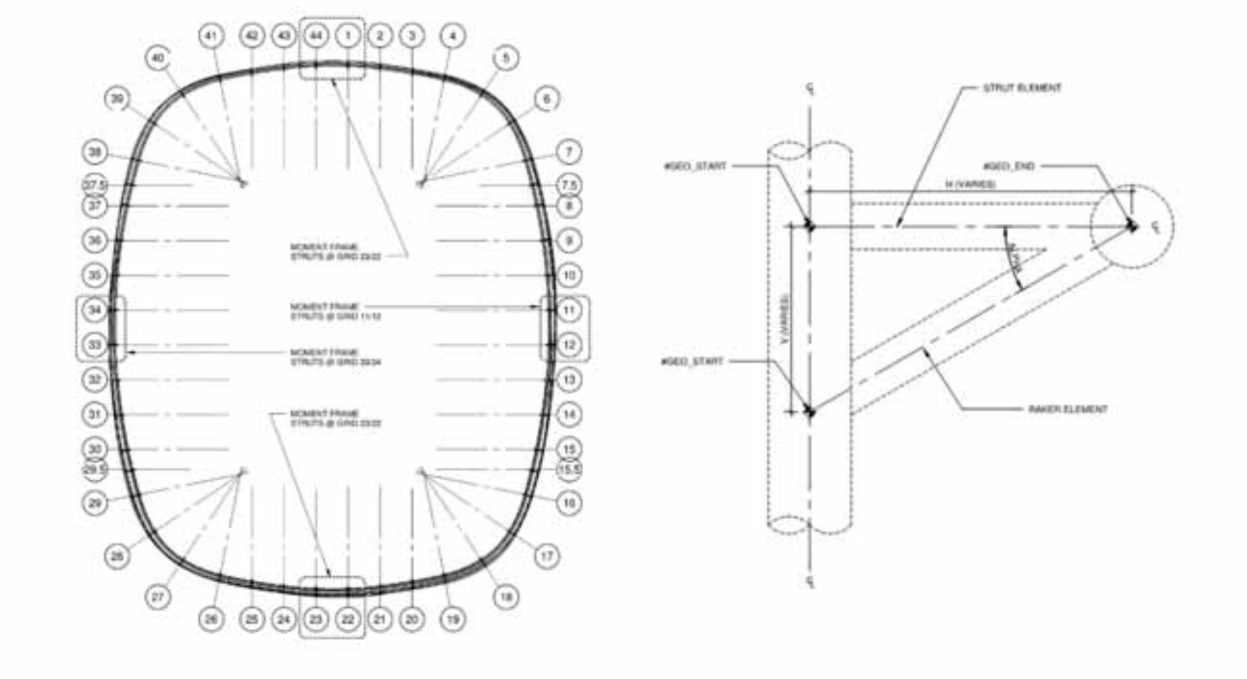


Figure 3: 48 grids layout – strut / Raker connection to the primary column.

Once the inputs were provided by the architect (Populous), the generation of the design and analysis model could be started. The analysis model was rationalized to the degree in which it was compatible with the analysis software (SAP/GSA). The three-degree loop geometry curves (splines) were translated into sets of straight segments. The design model translation was rationalized such that three degree curves become arcs (two degree curves). This translation allowed for extraction of end and mid points' coordinates (X, Y, Z) as well as the radius of each arc, which is the information required for steel fabrication. A work schedule was generated in the documentation process, based on which the façade geometry could be recreated in any software platform. During the fabrication process this schedule was referenced.

Recipe 001 started off the project's digital process by setting up a Grasshopper (GH) definition controlling geometry parameters and element attributes. The GH definition can be broken down into three main zones: wireframe geometry, generation of indices, and exchange of data between excel spread sheet and the geometry (Figure 4, Left). Zone one generates a rationalized wireframe for loops as well as the set up and arrangement of the armature geometry. The dimension embedded within each strut and raker was controlled by a spreadsheet which was plugged in through a custom component which allows for live exchange of data. The spreadsheet could be accessed and updated by any member of the team. Any changes made to the spread sheet would affect the geometry generation in the Rhinoceros/Grasshopper environment.

Once the geometry was set and ready to be sent for engineering analysis, the Grasshopper definition got condensed into a cluster with outputs including geometry wireframe, element index, and group names carrying the type of the structural members (Figure 4, Right). Assigning a name to the group helped sort the information more effectively. Data embedded within the cluster was live and available to access. If there were any changes to the input or the definition within, the cluster would be updated simultaneously from the parametric Grasshopper definition.

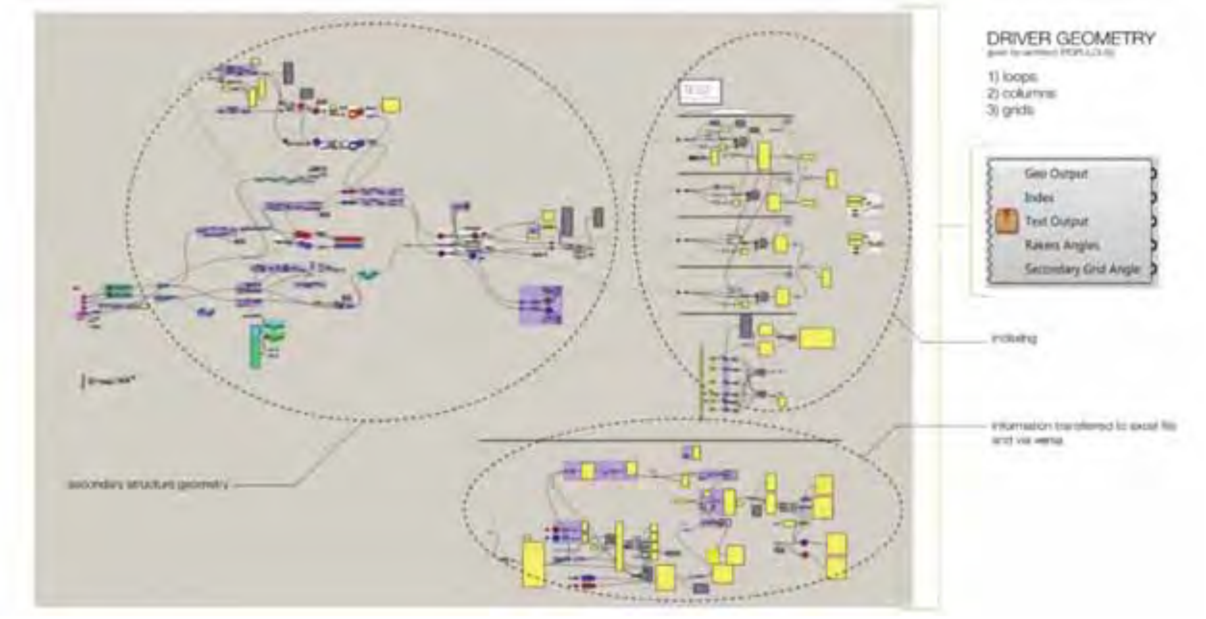


Figure 4: Grasshopper definition set up for secondary backup steel and indices.

Due to the asymmetrical nature of the loop geometry, the back-up steel configuration varied to avoid clashing with the driver loop as it converged and diverged. A central spreadsheet was used to store the parameter data of the armature geometry at each grid line. The armature data got translated into a wireframe by the algorithm embedded within the Grasshopper definition (Figure 5, Middle). The computation of this process to generate the armature geometry began by extracting work points and their elevation in space. The work points were obtained from the intersection of the gridline plane and the loop curves, and then sorted based on their elevation and associated grid number. After the data sorting was completed, each work point was projected onto the column closest to its position. The line generated between these two sets of points became the horizontal members, i.e. the struts.

Generation of the raker geometry was created from the shared intersection point of the strut and a second point projected onto the corresponding column based on direction of a defined angle vector. During the design iterations, definition of a consistent angle was suggested by the contractor instead of a constant vertical dimension in order to facilitate the steel fabrication process. The initial design included double armatures, where left and right side had a symmetrical relationship (Figure 5, Middle). The design later evolved to a single armature configuration to achieve fewer connection points and a cleaner aesthetic (Figure 5, Right). The parametric GH definition was updated correspondingly while the geometry generation philosophy remained similar.

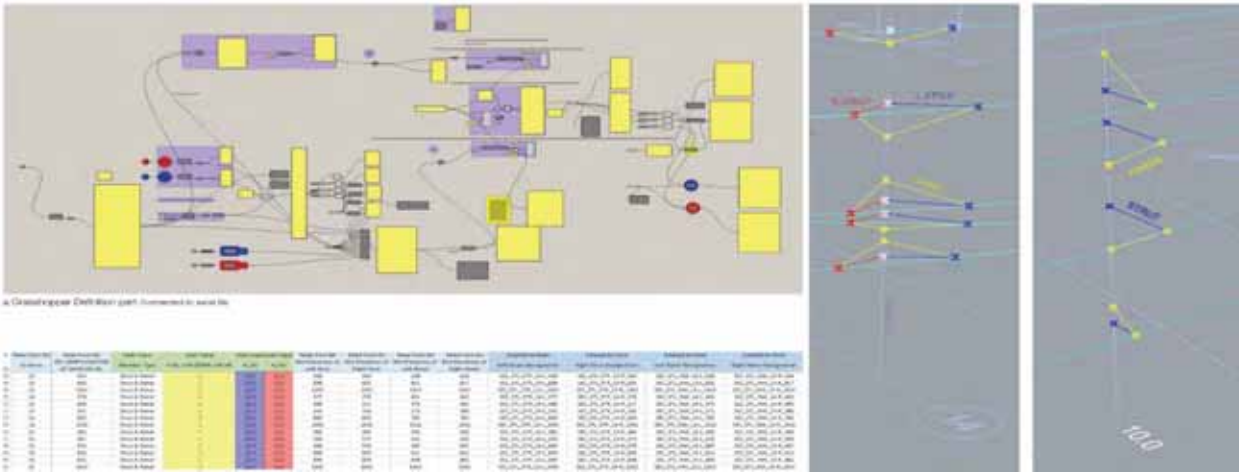


Figure 5: Grasshopper definition set up for secondary backup steel and indices.

EMBEDDING INDEX IN THE MODEL

A high level of geometric complexity and large number of structural members required an additional layer of information associated with each model element. Indexing as a type of naming convention brought organization to the overall project and communication. Once the index structure was designed, it was distributed among the teams working on the project. Distribution of the naming convention also increased accuracy of any process conducted over the duration of the project.

Common understating of the index structure greatly impacted the analysis and design process, which resulted in efficiency of the overall project development. An assigned index to each wireframe element set up a common language between the software platforms as well as the team members, and adaptation to any sort of changes became very manageable. Identification, location, or sorting of the elements became an evident and efficient process. Each component's index indicated its properties and attributes, including its discipline (primary or secondary), material, type, and elevation in the space (Figure 6). Analysis wireframe model elements include an additional number which allows for high resolution of identification and the breakdown of the larger component. Elevation number was extracted from the Z-coordinate of the element's centroid.

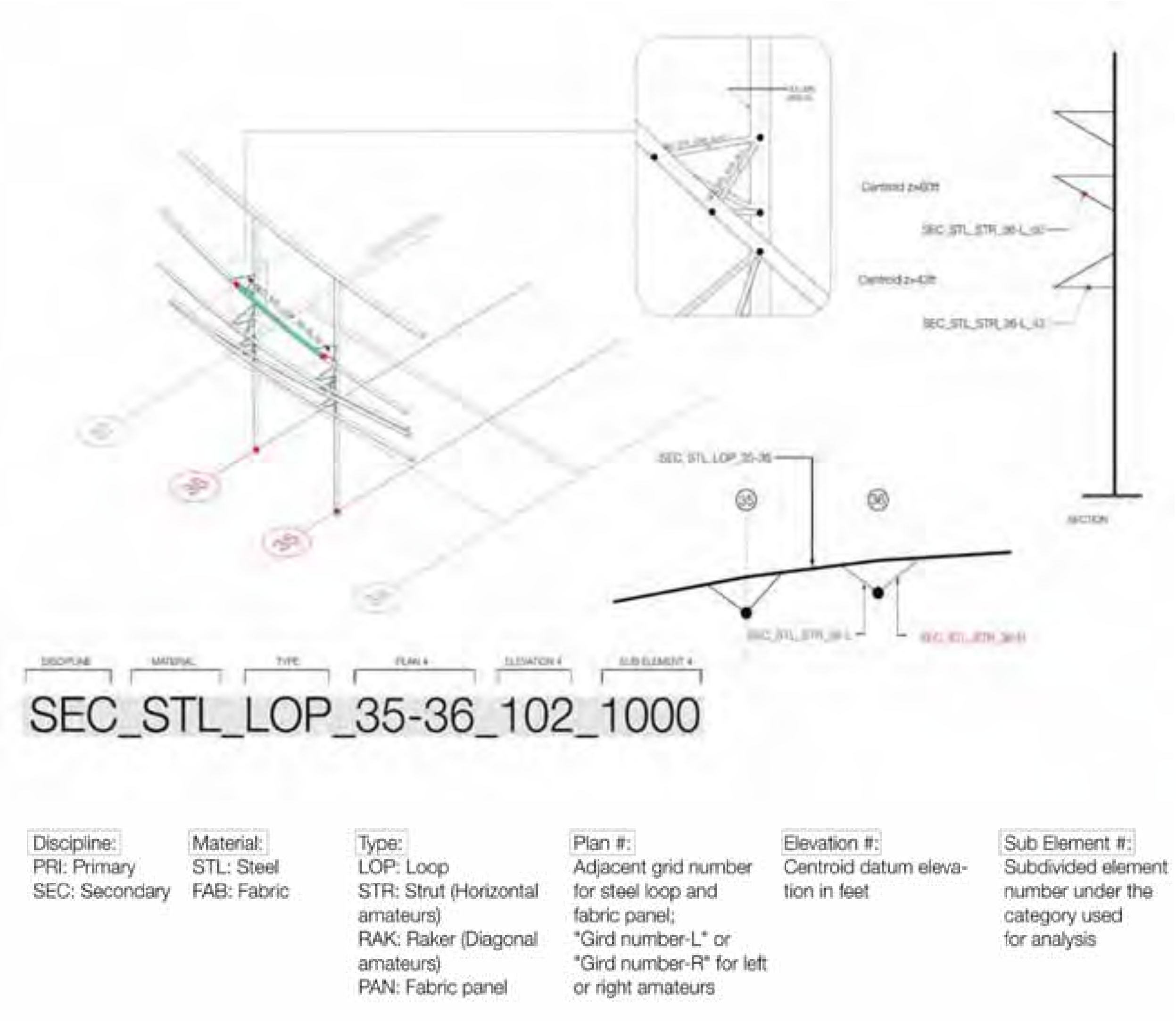


Figure 6: Index diagram break down.

CUSTOM GRASSSHOPPER COMPONENT

Besides general components, custom GH components were created to assist in establishing the design recipes specifically for this project. These included components to read or write data in GSA/SAP2000 and ‘Eyas’, which is a Grasshopper suite of tools to provide Object Relational Mapping (ORM) tightly linked with Excel.

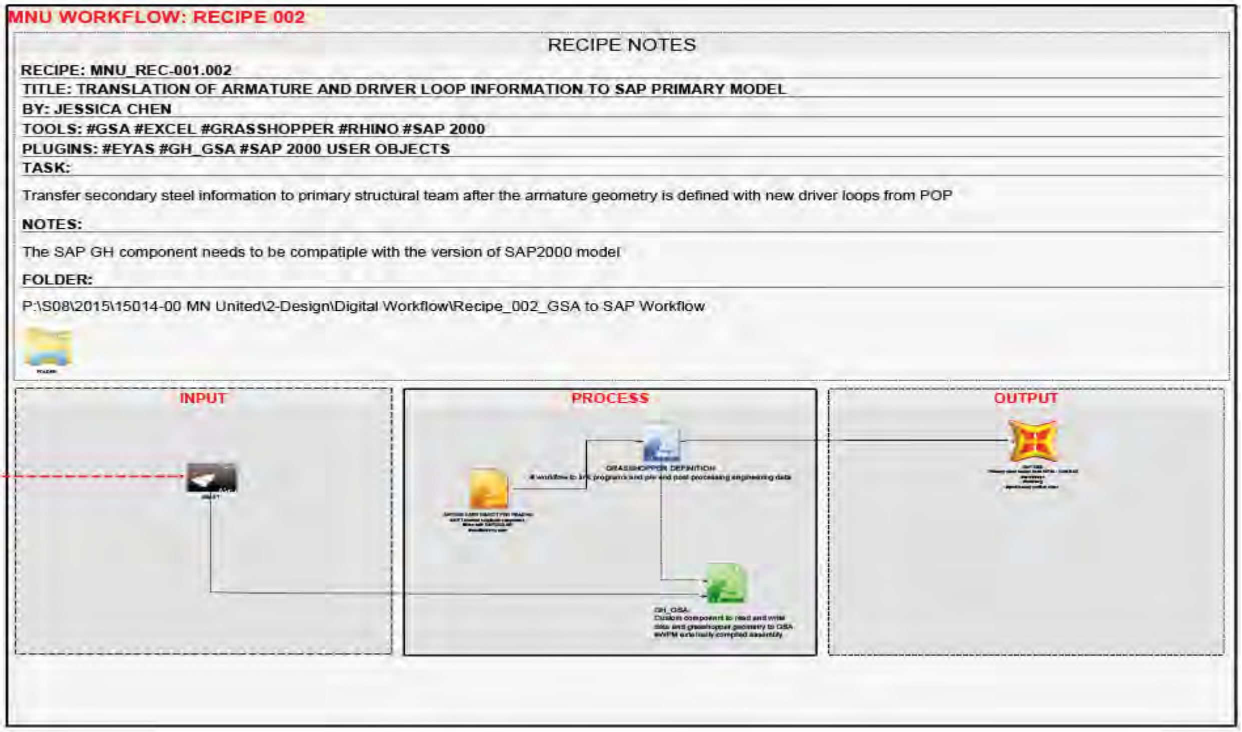
The GSA/SAP2000 GH components access the analysis model’s information through the software’s application programing interface (API). They were intensely utilized during the creation of the analysis model as well as for interchanging information between the two different software for collaboration between the primary and secondary structural teams. Eyas provided a fast real-time interoperable link with Excel, allowing for content management and documentation of Grasshopper data. The information link was a two-way system, both retrieving data stored in the spreadsheet (e.g. creation of armature geometry) and exporting data from analysis software to Excel sheets (e.g. extracting reactions for connection design). Eyas also provided options to sort, match, and visualize data using principles of object attributes, data tables and dictionaries.

DESIGN MODEL TO ANALYSIS MODEL

With the secondary structural geometry and indexing information summarized in the Grasshopper cluster, the information could be passed down to set up the engineering analysis model in GSA (structural analysis software by Oasys). This was the second half of Recipe 001. Among common practice, Grasshopper would usually be treated as tools for setting up geometry, and digital workflow would terminate from the point when the geometry was baked into Rhino and exported as DXF (Drawing Interchange Format) files. Processes like this provided a certain control over the analysis model geometry-wise, but structural attributes would need to be re-assigned over and over again each time a new DXF was imported.

With the API Grasshopper component, the information contained within the cluster could be translated directly into GSA and SAP. Elements (loops, armatures, meshes, etc.) were sorted into different groups based on their indices. Each group of elements was then paired with corresponding attributes, including section properties, orientation angles (for unsymmetrical sections) and element releases. The recipe proved to be time-efficient through the numerous rounds of driver loop geometry evolutions as the project developed.

RECIPE 2 – COMBINING SECONDARY STEEL INFORMATION INTO SAP



MNU WORKFLOW: RECIPE 002

RECIPE NOTES

RECIPE: MNU_REC-001.002
TITLE: TRANSLATION OF ARMATURE AND DRIVER LOOP INFORMATION TO SAP PRIMARY MODEL
BY: JESSICA CHEN
TOOLS: #GSA #EXCEL #GRASSHOPPER #RHINO #SAP 2000
PLUGINS: #EYAS #GH_GSA #SAP 2000 USER OBJECTS
TASK:

Transfer secondary steel information to primary structural team after the armature geometry is defined with new driver loops from POP

NOTES:

The SAP GH component needs to be compatiple with the version of SAP2000 model

FOLDER:

P:\S08\2015\15014-00 MN United\2-Design\Digital Workflow\Recipe_002_GSA to SAP Workflow

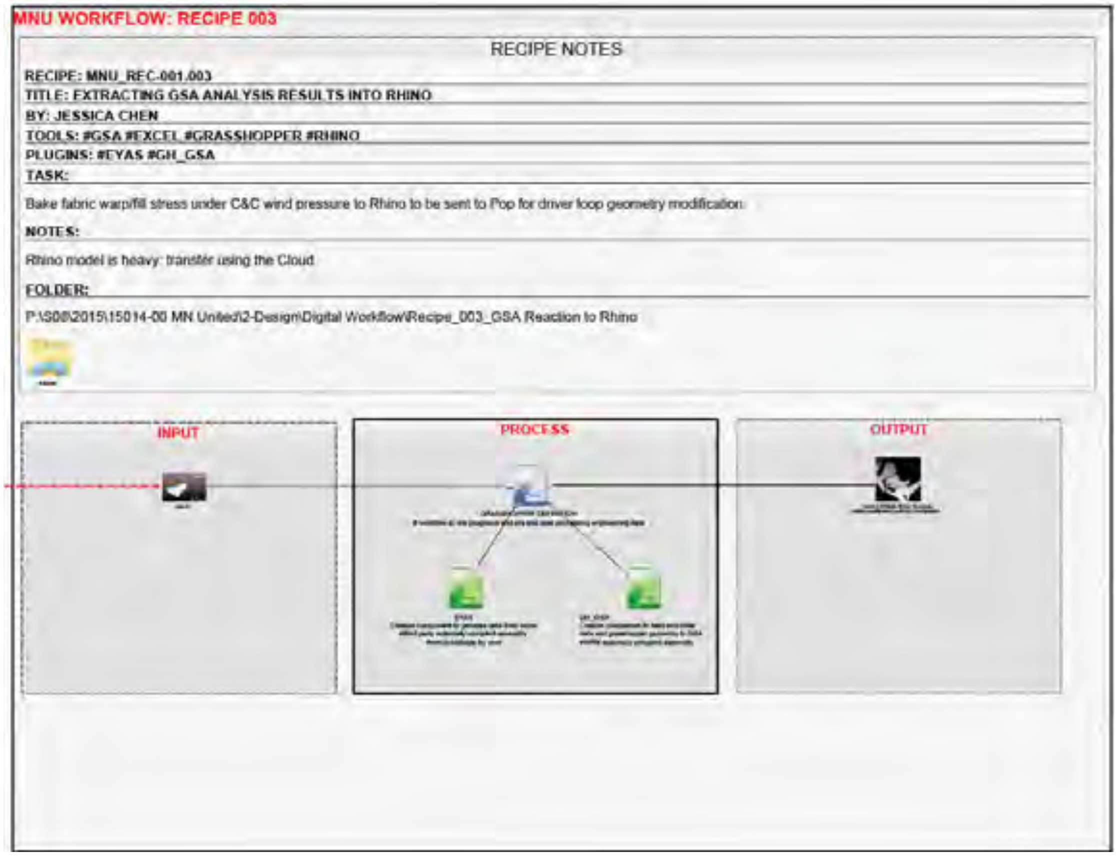
Figure 7: Recipe 2.

As WPM-Los Angeles Office was fully in control of the secondary structural geometry (except for the driver tube, which was issued by Populous), the armature's information needed to be sent to WPM-Kansas City Office for primary columns' design each time the secondary system was redefined. Recipe 002 using API Grasshopper components was performed, carrying the secondary system's indexing and structural attributes to transfer the model from GSA (secondary structure's analysis software) to SAP2000 (primary structure's analysis software) (Figure 8).

Figure 8-1: Indexing information Embedded in GSA.

Figure 8-2: Indexing information Embedded in SAP2000.

RECIPE 3 - VISUALIZING ANALYSIS RESULTS



MNU WORKFLOW: RECIPE 003

RECIPE NOTES

RECIPE: MNU_REC-001.003
TITLE: EXTRACTING GSA ANALYSIS RESULTS INTO RHINO
BY: JESSICA CHEN
TOOLS: #GSA #EXCEL #GRASSHOPPER #RHINO
PLUGINS: #EYAS #GH_GSA
TASK:

Bake fabric warp/fill stress under C&C wind pressure to Rhino to be sent to Pop for driver loop geometry modification.

NOTES:

Rhino model is heavy: transfer using the Cloud

FOLDER:

P:\S08\2015\15014-00 MN United\2-Design\Digital Workflow\Recipe_003_GSA Reaction to Rhino

Figure 9: Recipe 3.

Demand-capacity ratio (DCR) is one of the most important indicators to understand if a structural element has enough capacity under the prescribed loading. However, the traditional approach of presenting the results in spreadsheets packed with numbers is difficult to perceive and hard to diagnose in complex projects. Visualization, on the contrary, is an intuitive way to help everyone on the team to understand the data implication. Using modeling software Rhino to visualize analysis results also tackled the challenge of differential usage of software by different parties, especially useful for presenting results to the architect. Recipe 003 was set-up to visualize the fabric and steel DCR from GSA in Rhino.

Both of the visualization models used intuitive color gradient from green to red, where green indicates a lower DCR and red indicates a member or fabric under higher stress. Besides the color code, the mesh and polylines contained the exact numbers of their demand-capacity ratio. The fabric model was sent to the architect to analyze where the geometry should be massaged to lower PTFE's stresses (Figure 10-1). The steel model was used for optimizing steel section sizes. Based on the visualized model, the driver tube worked the hardest at the four stadium corners, while the straight bays generally had lower DCRs (Figure 10-2). When all the driver tubes were using a uniform HSS section, steel tubes with DCRs smaller than 0.5 were filtered out and downsized to a thinner wall thickness while maintaining the consistency of outer diameter around the stadium. The tonnage for secondary steel was able to be reduced by 15% due to this optimization.

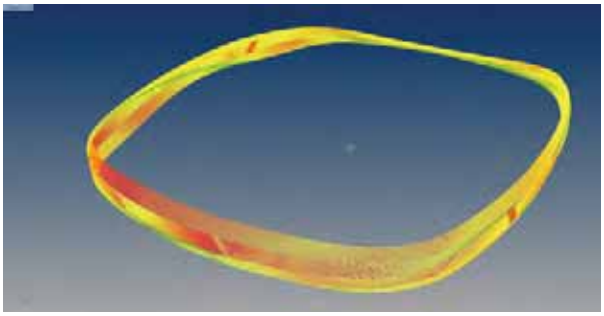


Figure 10-1: Fabric meshes' DCR in Rhino.

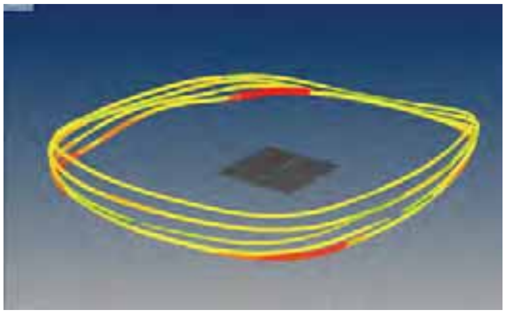
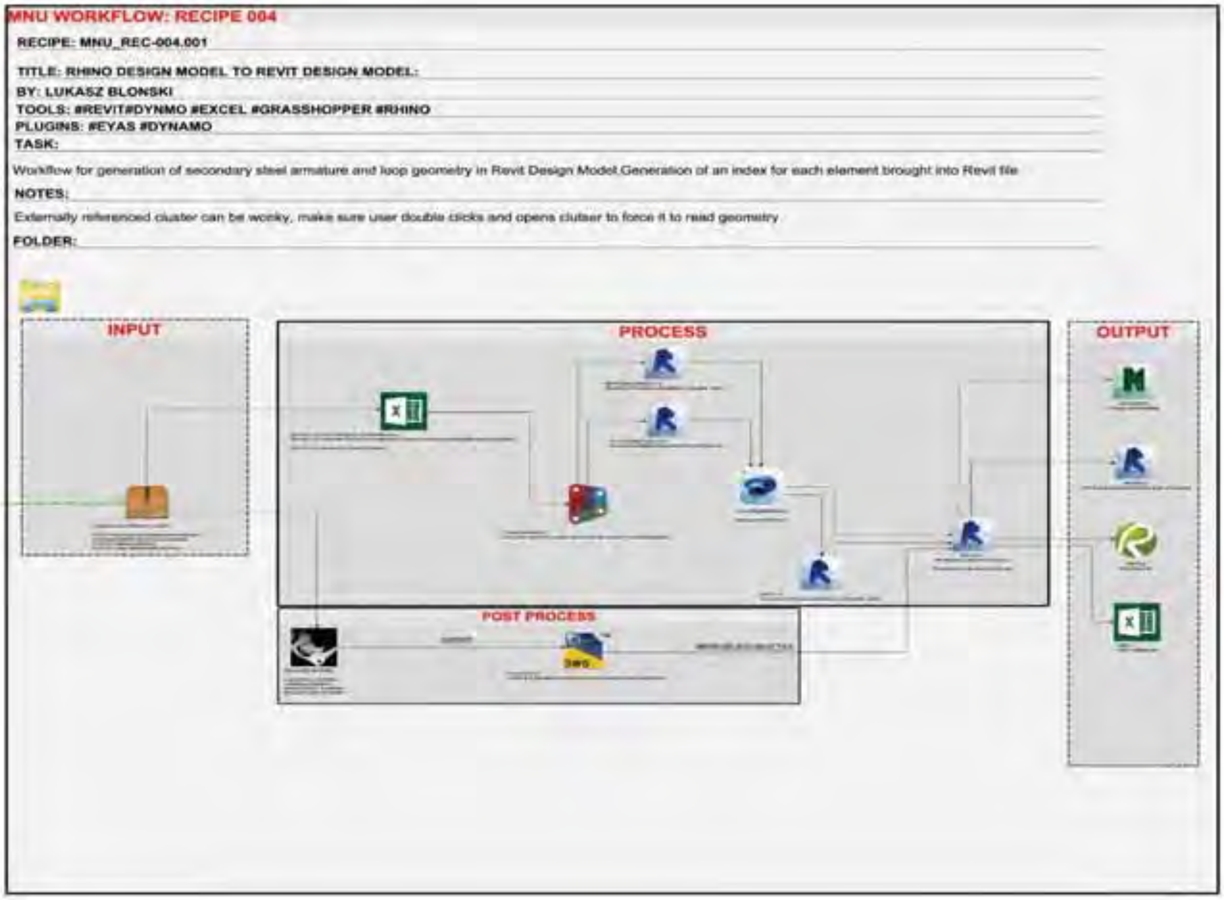


Figure 10-2: Steel driver loop's DCR in Rhino.

RECIPE 4 - DOCUMENTATION IN REVIT



MNU WORKFLOW: RECIPE 004

RECIPE: MNU_REC-004.001

TITLE: RHINO DESIGN MODEL TO REVIT DESIGN MODEL:

BY: LUKASZ BLONSKI

TOOLS: #REVIT#DYNMO #EXCEL #GRASSHOPPER #RHINO

PLUGINS: #EYAS #DYNAMO

TASK:

Workflow for generation of secondary steel armature and loop geometry in Revit Design Model.Generation of an index for each element brought into Revit file

NOTES:

Externally referenced cluster can be wonky, make sure user double clicks and opens cluster to force it to read geometry

FOLDER:

Figure 11: Recipe 4.

Recipe 004 is a documentation process which combined all the information gathered throughout the analysis and design phase. Revit performed as the main documentation tool. In addition, the project used A360 collaboration cloud service which could be described as an external software plug-in. This plug-in allowed for live updates, making documentation process more efficient. This type of digital collaboration enabled engineers and designers to be up to date with any changes generated by any of the team member.

Documentation started off by instantiation of the design model 3D geometry generated during the final analysis process. Dynamo, a parametric tool which runs with Revit, was used to transfer Rhino/GH geometry into the Revit environment (Figure 12). The work point schedule from the design model was a crucial integrant for this process. The WPM team developed a

Dynamo graph/definition that would recreate the wireframe geometry based on the (X, Y, Z) coordinates of the end points of the line segments (rakers/struts) as well as the mid-points and radii for the arc segments (loops). Assigning member size and type was also part of the dynamo definition; assignment and regeneration of the member types was computed simultaneously each time the Dynamo definition ran.

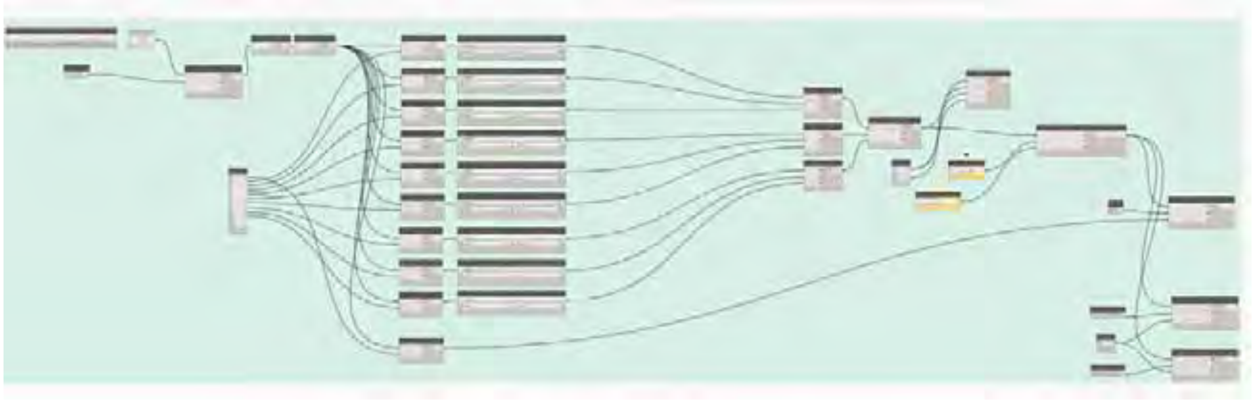


Figure 12: Part of Dynamo definition for loop geometry regeneration.

In later stages of the documentation process there was a need to use alternative software platforms in order to document drawings to achieve a higher level of accuracy. Based on irregular façade geometry and complexity of the project, WPM was required to deliver documents with highest level of clarity. The documentation package was geared toward showcasing intent of the design and a minimal level of ambiguity and interpretation.

Rhinoceros was one of main alternative tools to generate the drawings. After customization of the drawings was done, it could be easily exported out of the Rhino environment. Custom drawings generated in Rhino were carried over to the Revit platform using DWG file format. WPM's documentation package also included 3D rendered views generated using V-ray rendering engine for Rhinoceros, which is very common in the architectural field, but not common in structural engineering companies (Figure 13). Choice of a finial graphical representation was a crucial part of project delivery. Clear graphics led toward better understanding of the project.

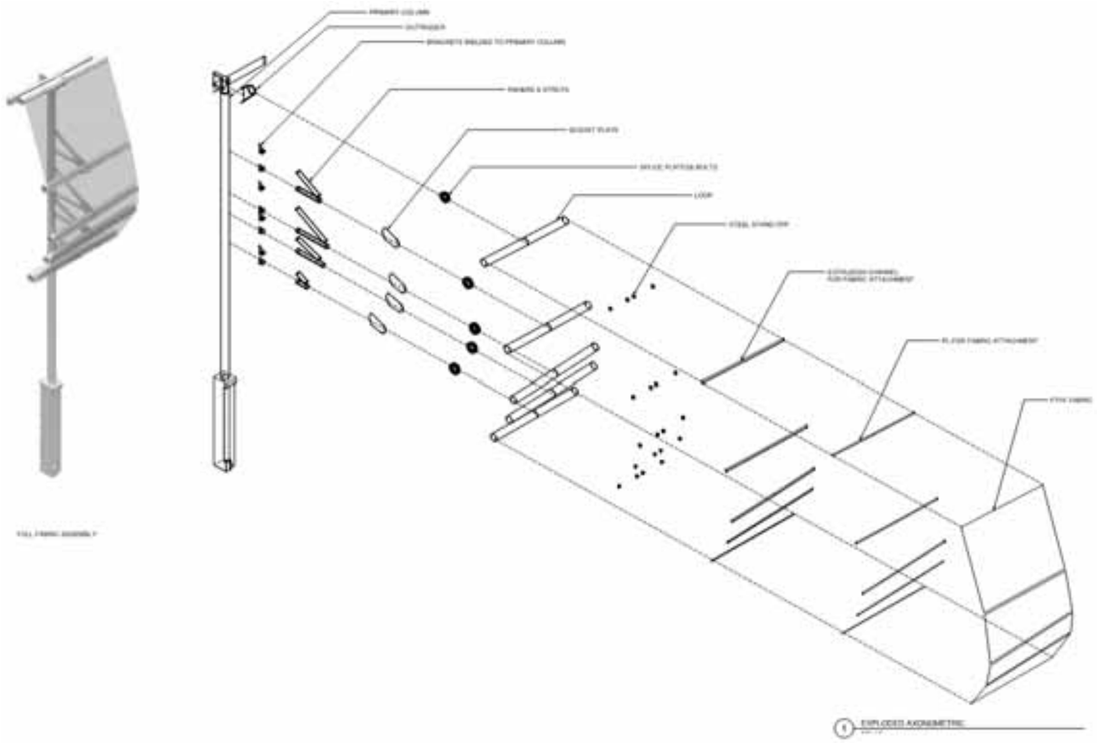


Figure 13: Sheet example contains an axonometric followed by section/elevations to showcase main assembly.

GENERATION OF TEKLA MODEL

The undulating driver pipes created the backbone for the dynamic façade at Minnesota United Stadium. Each of the driver pipes had a varying radius, length, and different start-end coordinates in all 3 directions (x, y, z). Rather than having the steel fabricator create a model of the driver pipes from the paper drawings, the design team generated a Tekla model directly from the digital documentation model developed by Walter P Moore and Populous. This brought all the geometry, member size, and index documentation into the fabricator model to ensure accuracy and speed in the overall fabrication process. The Tekla model also included information of the erection sequence. Model elements were color coded to allow for strong graphical representation and clear understating from the fabrication team (Figure 14).

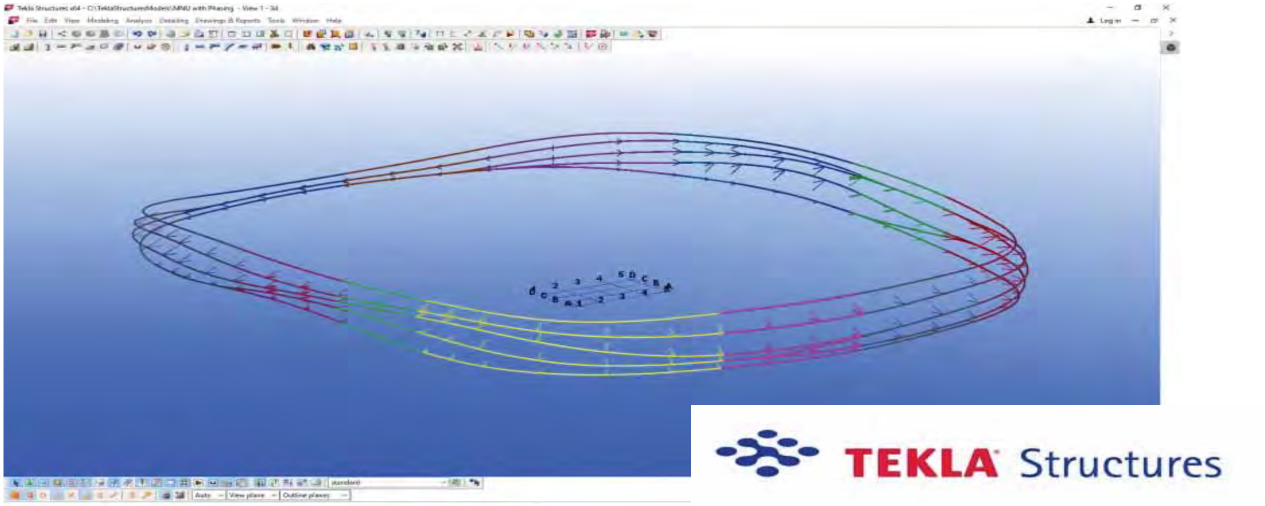


Figure 14: 3D screenshot of the color-coded Tekla model.

CONCLUSION

A number of highlights in the project's design process are:

- Digital workflow was documented in a bespoke menu, which consisted of various design recipes.
- Attributes assigned according to native object geometry allowed for effective and efficient way of position tracking.
- Digital workflow was used early on where performative engineering data was used to modify architectural design.
- Data was translated to a graphical medium for use by architect in an agnostic model format – color coded Rhino model.
- Indexing and ordering of geometry allowed for seamless communication of utilization ratios between primary and secondary structure.
- High fidelity digital model was constructed for the fabrication process

ACKNOWLEDGEMENTS

The authors would like to thank the project team for their collaboration on this project:

Phil Kolbol – Populous (Kansans City, MO)

Justin Barton, David Landis, Aditya Potdar – Walter P Moore (Kansas City, MO), Structural Engineering

Marty Augustyniak - Walter P Moore (Buffalo, NY), Facade Engineering

REFERENCES

"EYAS." Food4Rhino. October 25, 2016. Accessed January 12, 2018. http://www.food4rhino.com/app/eyas.