

AS502169

AUTOMATING THE NEW GOVERNMENT QUARTER IN NORWAY

Morten M. Ræder and Håvard Vasshaug

LEARNING OBJECTIVES

After consuming this content, participants should learn:

1. Strategies for automated data flow between Revit models, IFC deliverables and different stakeholders in large projects
2. How to set up centralized template models that contains and synchronizes all model settings, types and families
3. Methods for setting up and executing automation infrastructure that helps design and construction teams
4. How to implement project purpose custom-built design tools under pressure of constant project deadlines

DESCRIPTION

The New Government Quarter is an ambitious project with 7 buildings of more than 125.000 sqm / 1.3 million sq. ft highly attractive office space. But how to keep dozens of Revit models consistent across different teams, starting with the same templates, but not ending up with developing different model “dialects” over time? How can we avoid spending massive man-hours, and still keep data quality and data management easy to maintain?

To solve these issues, the design team is developing a set of automation processes for Revit. These are built as scalable add-ins that anyone can use and configure, and they are implemented using structured support, training and documentation strategies. The heaviest processes are being run at night using a large set of external resources managed by the IT dept. and configured by the BIM team.

The result is a staggering reduction of non-value adding activities by the design team, resulting in much greater return of investment in the form of design quality.

SPEAKERS



Morten is working as an architect and BIM coordinator on large-scale multi-disciplinary projects. He is also working with standardization and implementation of BIM on industry level.

Morten is allergic to buzzwords and marketing hype, but energetic about what technology can do to automate repetitive tasks and reduce complexity to free up time for creative work.



Håvard is passionate about improving digital tools, processes and workflows in the AEC industry, and convert that passion as value as founder of Reope.

Structural engineer by education and design technology expert at heart.

Just the right guy for this project.

Table of Contents

0 INTRODUCTION	2
1 PROJECT BACKGROUND	3
2 REQUIREMENTS	5
3 CHALLENGES	7
4 SOLUTIONS	9
<i>Prototype</i>	<i>11</i>
<i>Development Challenges</i>	<i>12</i>
<i>Development Prerequisites</i>	<i>13</i>
<i>Workflows</i>	<i>13</i>
5 SUMMARY	18
6 APPENDIX	19

0 INTRODUCTION

The idea for automation in the New Government Quarters in Oslo, Norway, sprang out of Morten's experience as BIM manager for the expansion of Oslo Airport Gardermoen, aka. the T2-project, which also was presented at AU back in 2017. The project was a huge success and was delivered on time and under budget. It pioneered several BIM workflows and processes, like IFC-linking, rebar automation and model-based manufacturing. Still, it identified several flaws and rooms for improvement in BIM workflows, particularly in the development of multiple Revit models and reliance upon a huge number of constantly updated IFC links. Further, it showed the importance of BIM standards and guidelines for setting up metadata flow, to allow for consistent data across all models, useful for all project stakeholders, without non-value adding activities.

This case study presents the project background for the New Government Quarters, explaining the project organization, digital strategy and requirements that has formed the framework for automating Revit's handling of families, settings and information exchange across a huge number of models.

All illustrations and photographs are copyright of Statsbygg / Team Urbis, unless otherwise stated.

1 PROJECT BACKGROUND

On July 22 2011, a right-wing terrorist detonated a car bomb in the Government Quarter in Oslo, Norway, killing 8 people and causing extensive material damage. Later the same day, the terrorist began a massacre at the Labour Party's Youth summer camp at Utøya, killing 69 people, before he finally was arrested.

This is the tragic backdrop for the New Government Quarter. It's not only a building project, but also about repairing the urban scar after the bomb and rebuilding the democratic institutions the terrorist wanted to destroy. From the original quarter, only two historic buildings remain in the new project. It's also about adapting to the original urban context, reopening passages and sightlines blocked after decades of badly planned development.



22 July 2011

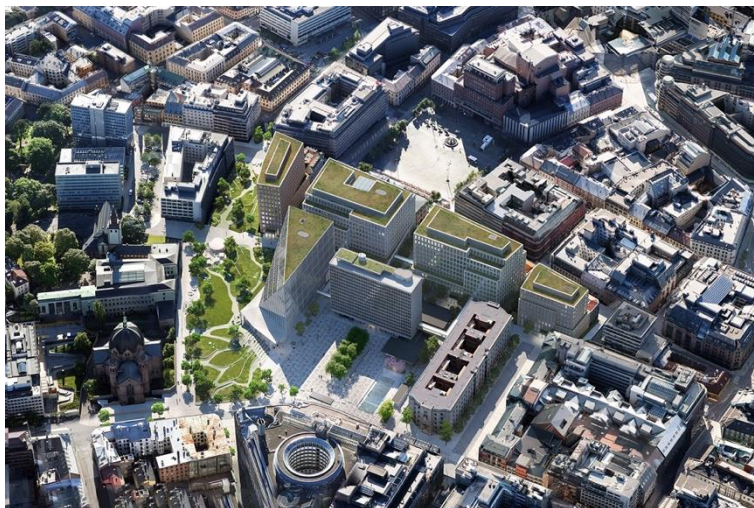


© NTB Scanpix / Thomas Winje Øijord

Vision

The New Government Quarter will contribute to a better urban environment — it will be efficient, flexible and secure for the ministries it will house in the years to come. It will be an area of representation for the government, yet open and available to all. Four words guide its development: noble, enduring, beautiful, and friendly.

Adapting to the city



New Government Quarter adapts to existing city spaces

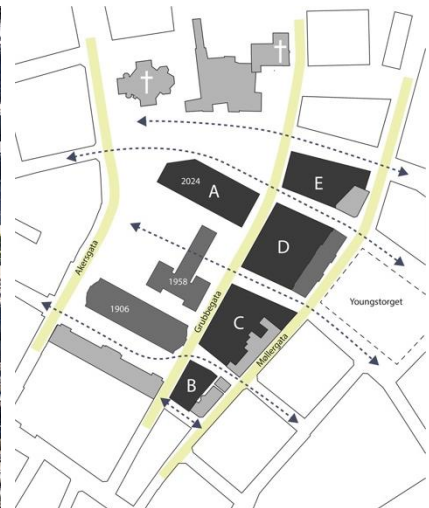


Figure 1 Opening up blocked passages and sightlines

The concept adapts to the city; when finished, the New Government Quarter will convey Norwegian values: transparency, accessibility and a close affinity to nature. It will be a highly functional and attractive place to work, and it will feel like a natural extension of the city — a place where the residents of the city will feel at home.

Past meets present

Today the Government Quarter is characterized mainly by two historical buildings: the G-block and Høyblokken. The G-block was completed in 1906, the year after Norway was made independent from the union with Sweden, and the H-block from 1958 stands as a symbol of the rebuilding of the city after World War 2. The new A-block, together with other connected new buildings will represent modern Norway and hereby bring a historical continuity to the district.

G-block

The G-block has a unique historical value, as the first government building constructed in Norway. The exterior and interior is listed, except from the fifth floor, which was rebuilt in 1976. This floor will be upgraded to modern standards while the rest of the building will undergo upgrades to technical systems and adjustments for inclusion in the overall concept. There will be about 400 workplaces in this building.

Høyblokken

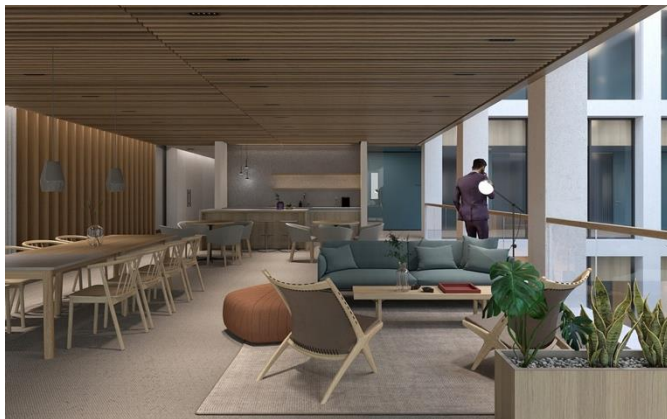


© Nordic – Office of Architecture

Høyblokken will be upgraded to become a well-functioning office building for about 500 people. The preservation of Høyblokken ensures historical continuity and emphasizes the concepts' sustainable approach. The building will go through a considerable transformation to contemporary standards. Our goal is to preserve cultural, historical, artistic and architectural values, restoring lost qualities and adding new.

Flexible workspaces

Interaction and communication between ministries are encouraged by a “ribbon” connecting all the buildings on level 2, providing easy access to all common areas and entrances. Office spaces will be flexible and adaptable to future needs, designed with an open mind towards new working methods and implementation of technology in the years to come. The concept for the office spaces is clear and simple; workstations and social zones are located towards the facades taking advantage of the daylight, while meeting rooms and break out spaces are places further into the buildings.



Social zone in office area



Interaction zone connecting all blocks

Project structure and BIM-organization

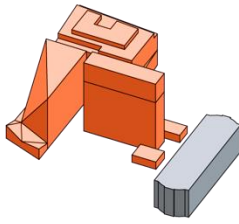


Figure 2 Building Phase 1

A-Block
D-Block
Høyblokken
Basement
Urban Spaces

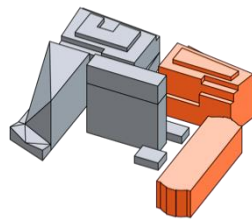


Figure 3 Building Phase 2

C-Block
G-Block

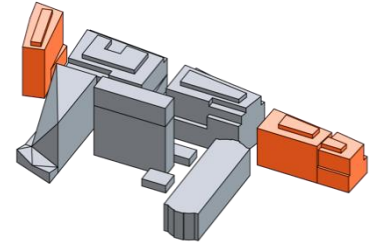


Figure 4 Building Phase 3

B-Block
E-block

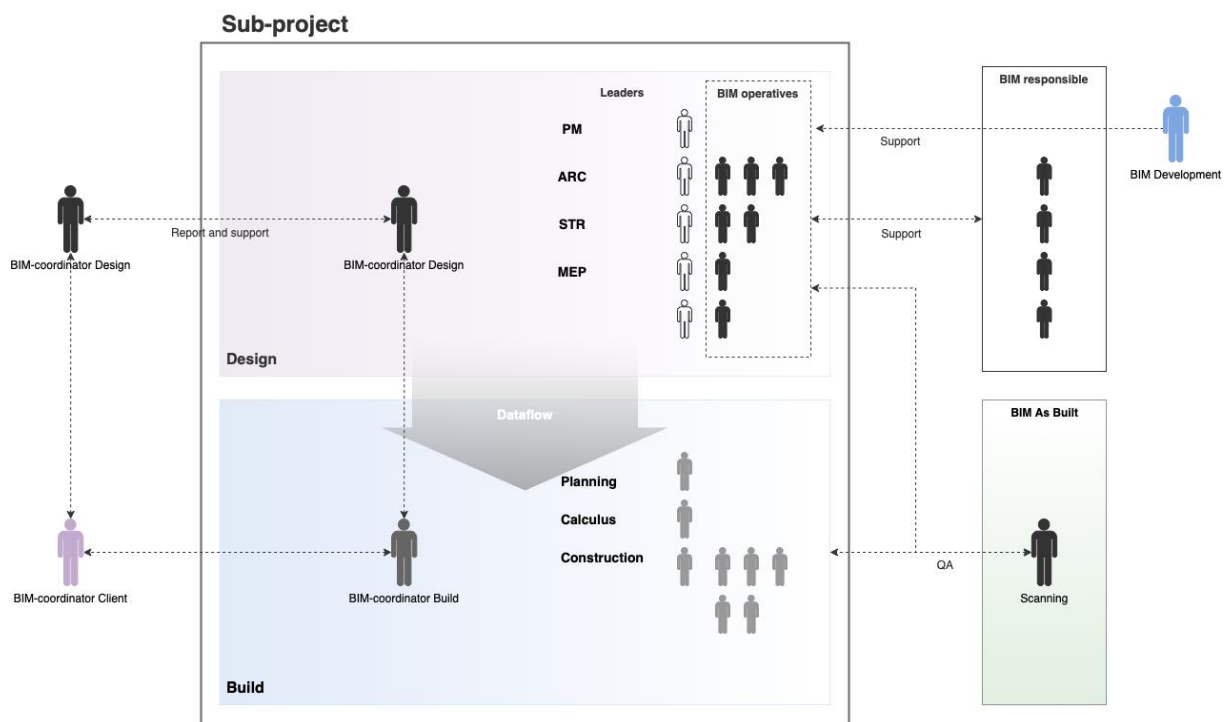


Figure 5 - BIM-organization within sub-project.

All sub-projects have dedicated BIM-coordinators for both the design and build teams, responsible for model collaboration within the sub-project. They report directly to the main project BIM-coordinators, who also offer support on implementation of BIM requirements etc.

For team support, there's a dedicated group of BIM responsible for each discipline, as BIM resources within the sub-project may be limited in some cases. Additionally, we have a BIM-development role, responsible for development and implementation of automation procedures.

2 REQUIREMENTS

Security

Due to the delicate nature of the project, there is a strict emphasis on digital workflows as the best way to control information. Designing and building of government ministries require a high level of security, where information is provided on a need-to-know basis.

As a result, the project has been divided into several, largely autonomous projects, that when finished will end up as an integrated building complex

Digital Strategy

The digital strategy for the projects is based on four main points:

- **Design and Planning**
BIM tools and methodology is at the core of all planning and design processes
- **Interaction and Coordination**
Collaboration between all stakeholders using efficient digital tools
- **Construction**
No physical drawings on site, BIM is to be used as much as possible for construction
- **Digital Twin**
A digital representation of the finished complex to be used in the operations and management phase

Standards and information requirements

Now, this is where BIM leaves buzzword territory and disappears into a fog of three-letter abbreviations. Generally, a lot of projects either break their neck in trying to comply with vaguely defined BIM requirements (which practically every commissioner or contractor pride themselves with developing, in lack of common industry standards), or fail to deliver on comprehensive and detailed requirements due to a lack of proper tools or failure to recognize the huge amount of non-value adding activities currently needed to deliver according to such specifications.

We were very lucky in this project to have as our client the most experienced public building commissioner, property manager and developer in Norway, Statsbygg. They have pioneered the use of open BIM and IFC as the main model collaboration format for all their projects since 2010. Being a public agency, they also have the mandate to promote technological innovation for the common good of the entire industry. When starting out the project they mainly had two requirements for metadata:

SIMBA¹

No, we're not talking about the Lion King here, but the short name for our Exchange Information Requirements (EIR). It relies heavily on openBIM and the IFC standard (2x3), specifying in detail specific property sets and names for building elements.

Over the last 10-12 years, the largest public building owners in Norway has adopted this EIR either directly or indirectly, and as a result open BIM and IFC are the industry standard for all model exports and collaboration.

TFM²

Abbreviation for (in English) Cross-disciplinary Marking System. This is a classification manual for all building elements, and the use of TFM-tagging really boils down to the need for a code that easily will identify the type and location of a specific building element and tell which system the element belongs to (if any). The TFM code is the main identifier needed for product lookup from the digital twin, for use in future operation and management of the building complex.

It's crucial to avoid creating too many TFM-codes that need documentation due to slight variations in product descriptions or unnecessary element types due to slight differences in model development.

¹ See appendix 6.1

² See appendix 6.2

The TFM-codes will live on for the lifespan of the building, and a systematic and structured database is needed for efficient operations and management.



The new A-block as seen from the new Government Park

3 CHALLENGES

The project requirements establish the framework for BIM workflows in the project and constitute an important part of the automation strategy. Still, they also generate a few challenges, which also happened to overlap with previous experience.

The major partners in the design group, Team Urbis, had just finished the expansion of Oslo Airport Gardermoen before starting the New Government Quarters. This was another large-scale project, who pioneered many BIM workflows (see AU presentation 2017!), like linking of IFC in Revit, automated rebar, model-based construction etc.

Working as BIM-manager for this project prior to the New Government Quarters, Morten saw a lot of time being spent by the team on complicated workflows and painstakingly boring tasks. To improve on these lessons learned, well-planned automation was without doubt the holy grail that would solve a lot of issues. It would dramatically reduce time spent on BIM QA and manual processes related to model imports/exports, reducing needless manhours and cost for the public, while freeing up time for all architects and engineers to focus on project development instead.

Four main challenges were initially identified:

Challenge #1 – Model “dialects”

- Due to security requirements, only a minimum of the architects and engineers in the project are allowed to access models outside their sub-project, making cross-referencing difficult
- How to co-ordinate development across sub-projects when human interaction is limited?
- How to avoid model “dialects” emerging?

Challenge #2 – Properties

- Employer’s Information Requirement (EIR) refers primarily to open standards (IFC 2x3) and properties geared towards building operations and management. The IFC standards doesn’t relate well to professional terminology or everyday language
- No industry standard information requirements available for design, building, planning or procurement
- How could we establish a hierarchy of relevant metadata using professional terminology, for consistent naming and easy identification in BIM QA processes and still be able to deliver on open data format standards?

Challenge #3 – Revit itself

- Basically, all Revit models are their own self-contained database with no built-in functionality for data consistency across several models
- Model «dialects» and data inconsistencies are incompatible with the final delivery of a digital twin for the entire complex
- But how to keep these consistent over dozens of Revit models, with minimum manpower and human interaction?

Challenge #4 – Security

- Model collaboration is based on a huge set of general and custom model exports, both as native Revit and IFC exports, as the only way to provide information for different stakeholders on a need-to-know basis
- IFC exports and links from other design software (Tekla, ArhiCAD etc.) needs custom Revit models or central processing
- How to make sure we can efficiently process IFC-links and create custom Revit models for different purposes?



A-block, main entrance

4 SOLUTIONS

- Strategies for automated data flow between Revit models, IFC deliverables and different stakeholders in large projects

It's important to keep in mind the background for the project because you need to understand the reasons for the way we are organized. For a regular office complex, it's overly complex, but for our project it was necessary due to security requirements.

We have a ton of Revit models, and with complicated and rigid interfaces between the teams building the different blocks in the Quarter, there really is no way around automation. We certainly don't want to waste the taxpayers' money on highly skilled architects and engineers copying and pasting data most of the day!

This isn't just about saving public money, but about reducing tedious tasks. When too many people in a design team needs to spend too much time on the technical aspects of BIM, we risk losing the creative possibilities that lies within the technology. This means fewer people finds BIM interesting and it's reduced to a technical necessity handled by geeks like us.

Preparation and analysis

We came up with an initial set of proposals and guidelines for the solutions we needed to answer the challenges we were facing:

Challenge 1 and 3 – Revit Master Settings model

To battle the inevitable model dialects emerging, we needed a Revit master settings model for storing family types, project properties, view templates, browser organization etc.

Challenge 2 - Metadata

All project properties must be named according to professional terminology and be of actual use and provide value to the project. All property names and sets must be represented equally regardless of software platform while still compliant with IFC / EIR requirements. Data shall be entered once and referenced everywhere needed.

Challenge 3 and 4 – Automation hardware framework

We need a framework of virtual servers to automate synchronizing between Revit settings and project models, to handle all exports for model referencing, together with automatic regeneration and copying of links across all sub-projects.

Automation prerequisites

One basic point when working with databases, is that data should reference everywhere needed from one single entry. This may be possible to handle within a single Revit model, but there are very few tools available that could provide consistent data flow between several Revit models.

To find a solution to this, we needed to have a basic idea for the data flow between models:

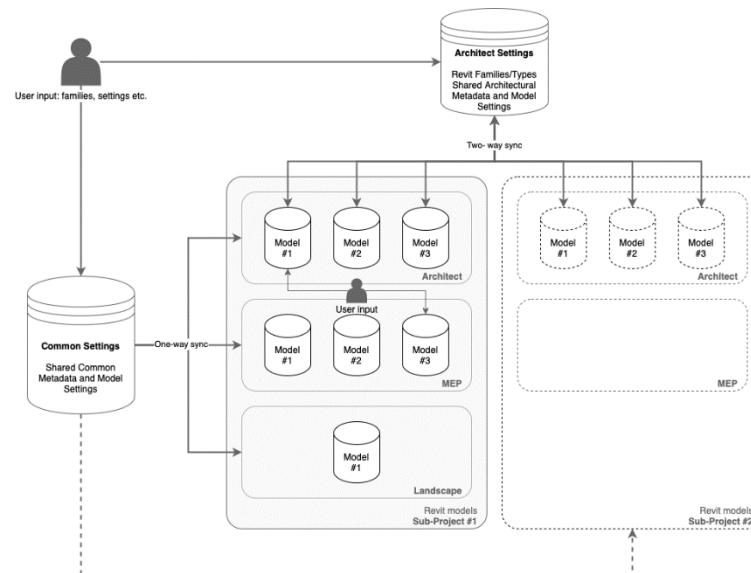


Figure 6 Prototype for basic data flow across project models

This was sufficient for testing a prototype solution, typically done as a Dynamo script. But to make this work in a project spanning several years and over several Revit versions, we needed something robust and scalable. We'd need an exact answer to *what* data should be referenced, *who* originally owns it (i.e., enters it in Revit) and *where* it should be accessible

Metadata - Property naming and property sets

To use established technical standards / API for metadata is great for automating data flow. They're great on a technical level, as they're robust and scalable, because the data is found always in the same place regardless of models or platforms. On the other hand, though, standards tend to need translation into everyday language to work in real-life. For communicating data with such a diverse group of people as in a building project where we want the models to be used by anyone from architects to plumbers, my experience is that translations into common language or established professional terminology provides the greatest common understanding of the importance of metadata values.

Even though SIMBA (see above) is a very comprehensive and well documented EIR, it focuses primarily on the final model handover to the client. It relies heavily on standardized open BIM/IFC properties for long-term data access, tuned to the needs for building operations and management. Additionally, IFC property names as project parameters in Revit, is not communicating well with end-users, and was never considered an option.

Apart from SIMBA, there were no other information requirements openly available in the industry that we saw fit. As a result, we started the process of defining the information needs for the necessary BIM tasks to spring out of the digital strategy for the project, like design verification, project scheduling, interface control, procurement, and building. A major advantage of having a huge amount of project stakeholders, is that we fairly quickly were able to map out properties for these tasks, not only tuned for the project but the entire industry.

Another important aspect is how to present the information to the end-users. People are usually happy to find info grouped together in property sets for quick access, and that the property names are consistent regardless of opening the models in either design or viewer software. For most projects, there's usually a need to expand native Revit properties by adding Project Properties. In our case, where using industry terminology as property names, we set up all required information as Revit Shared Properties, mapped to different categories.

The project parameters were implemented in an ongoing project, so we needed tools to automatically update all relevant project files. This works well on paper, until we realized that Revit by reasons beyond imagining insist on sorting project parameters in the properties panel *chronologically*! Mostly though, we were lucky to get it right the first time, and the result would be correctly sorted and grouped parameters in Revit as well, but every time we needed to add or do an update to existing parameters, they ended up at the bottom the properties panel. Aaargh!!

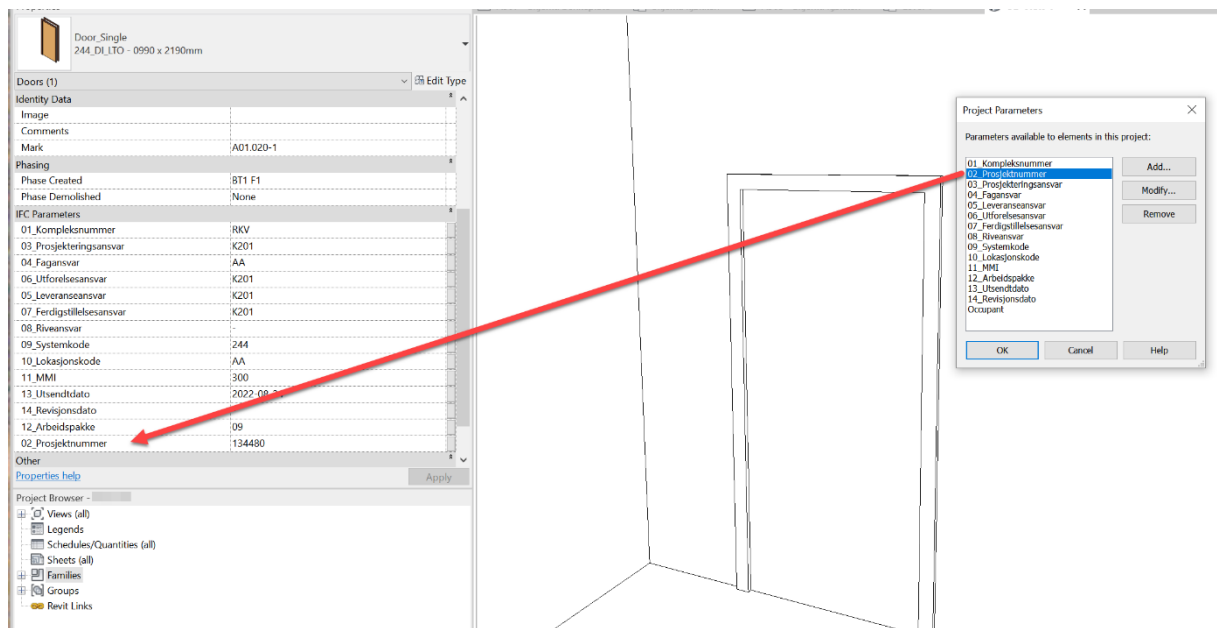


Figure 7 Revit properties not sorting alphabetically in Project Browser

To achieve consistent naming in both Revit and IFC exports, we created comprehensive user defined property sets in Revit's IFC Exporter UI, exporting to both localized naming and IFC standard properties. If needed on site, we'd potentially be able to publish models in different languages like English, Polish etc., just by translating terminology in the text-file. Still, we really wish for a better GUI and support for this in Revit...

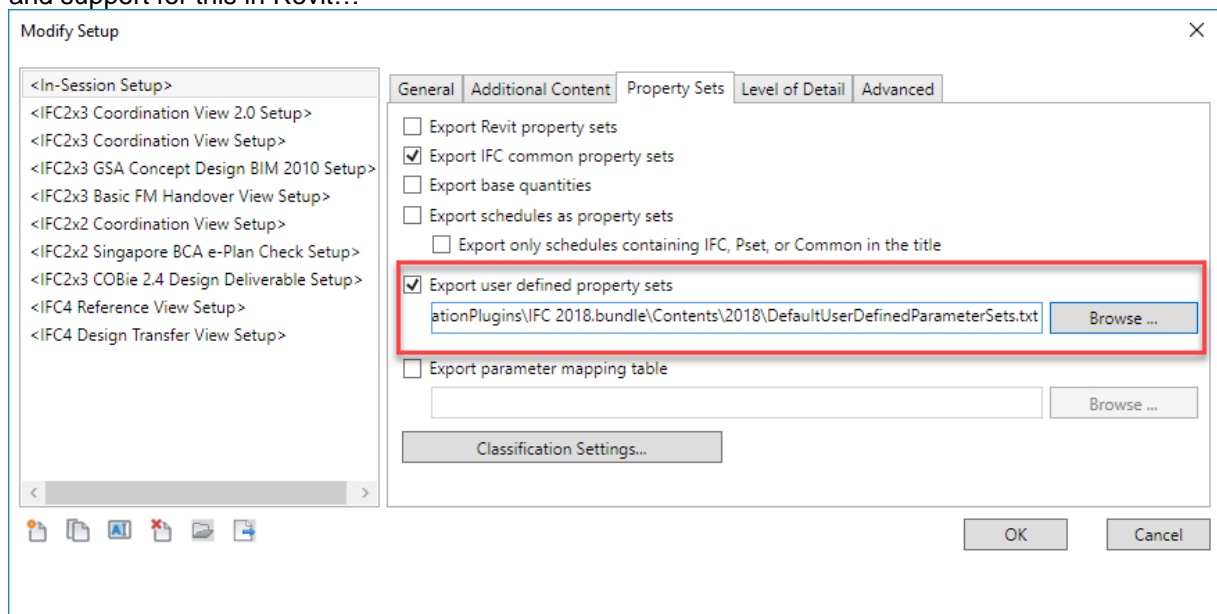


Figure 8 Revit IFC Exporter UI - User defined property sets

Prototype

Based on the analysis of metadata and initial solutions, we had enough for a working prototype. Alongside metadata, we also analyzed geometry definitions and model maturity levels, for a holistic approach to our BIM Execution Plan (BEP), which guided the development of what went into the central Revit Master settings-file that contains all official project families and types, properties, settings etc. From this model, all relevant families and settings syncs and overwrites across all connected Revit models. This way, we can edit data in one place, and it then updates everywhere! It also opens up for edits being done in one project to be uploaded to the master settings model, so that solutions developed in one sub-project easily can benefit the entire project.

Imports/Exports

In economy, the relationship between imports and exports is crucial to control if you want a healthy trade balance. In BIM, this is analogous to the time spent doing imports/export in a project and your mental health. Not much help in Revit out-of-the-box, but there are several great export add-ins, like RTV Xporter Pro. Imports on the other hand, and especially regeneration of IFC links, is where it can get tricky, and people start behaving erratically. What if we could slice all hours spent on this, and let the machines take over these tasks all together?

Now, enough theory! We've come to the hands-on technical part explaining our solutions developed for the project by Håvard Vasshaug and Reope. This is quite clever, and something I've wanted to come true ever since the Oslo Airport project.

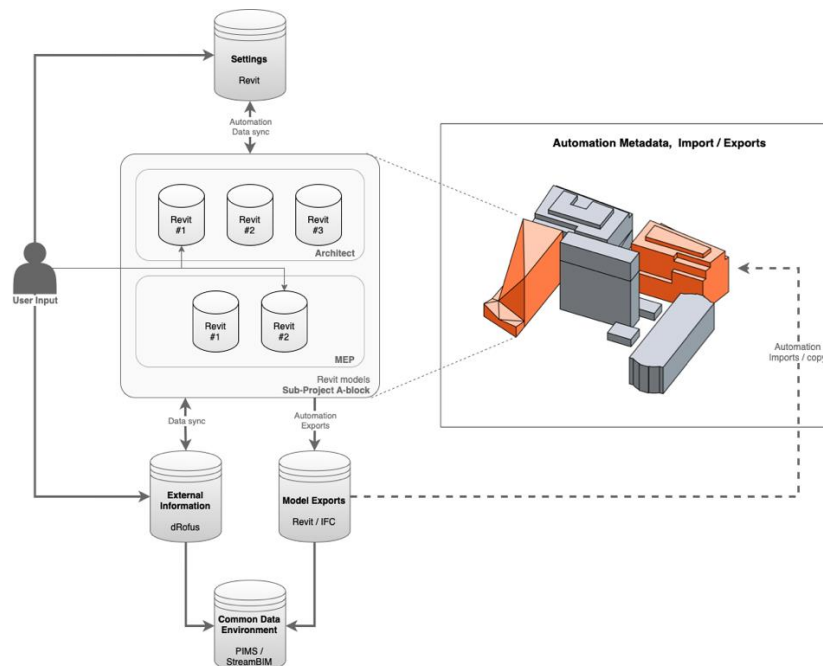


Figure 9 Data flow from user input to common data environment, with automated model exports and linking

Development Challenges

– What were the difficulties with driving a software development project inside a building project with its own constraints?

Security

The security strategy for the project resulted in a few challenges with driving what basically became a software development project inside the building project.

No:

- Foreign nationals
- Remote work
- Internet

Difficulties:

- Installation process
- Reporting and understanding bugs
- Recreating problems online

The process of explaining to the developer what a tool needed to do without any access to model data, required an efficient and clear line of communication. It also required developers who understood, in detail, the challenges a project team faces with design, deadlines, BIM, Revit and so on. It also required an implementation expert who understood and were able to foresee potential problems with features before they were being implemented. The installation process could potentially take weeks, requiring resources from an IT team that had quite a lot to do, so rapid iterations and quick bug fixing weren't really that easy.

The staffing constraints meant that only Norwegian nationals could work onsite on the project. They would also have no access to online tools. If cloud-based applications were needed, they had to be recreated offline, which always involved a big cost. This was of course inconvenient, as most skilled Revit developers aren't Norwegian, and a lot of Revit automation tools are cloud based.

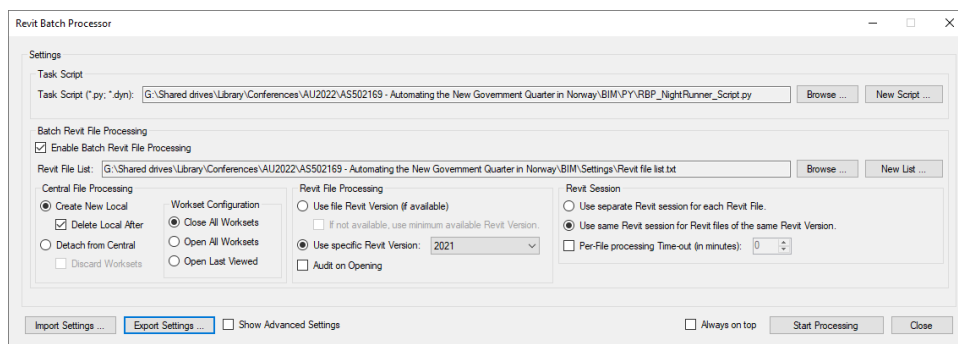
Deliveries

Driving product development aligned to project deliveries was creating some situations that caused a little stress. Project deadlines follow their own logic and while they are also not so much fun, they have nothing to do with the difficulties of developing an application. At the same time, project models that were the main deliverable needed the functionality to be developed before delivery, hence it was crucial to make the two work together.

Development Prerequisites

– Which resources did we need to succeed?

Our main application for driving Revit on an automated schedule quickly became the free and open-source Revit Batch Processor (RBP). We also applied RTV Tools for some export processes. The setup we used for RBP and Revit, was that RBP opened a list of Revit files, and started a Python script that pointed to a settings file (json) that opened our add-in with the specified settings being applied.



For testing workflows onsite, we mostly used Dynamo and Revit Python Shell. Dynamo was an important prototyping tool for the implementation specialist who could test Revit API related issues on the real project models. Without being able to do that, our development process would have been much slower.

Last, with little access to customize anything, we were totally dependent on a close collaboration with the excellent IT team at Statsbygg. After some trials and errors, we were able to develop a system based on Remote Desktop and a spreadsheet specifying the different automation processes that the remote machines would execute and when. In the end, all we had to do when implementing a new process was opening a central Excel file, add a line with information about the script and when it should run, save and close. Easy peasy lemon squeezy.

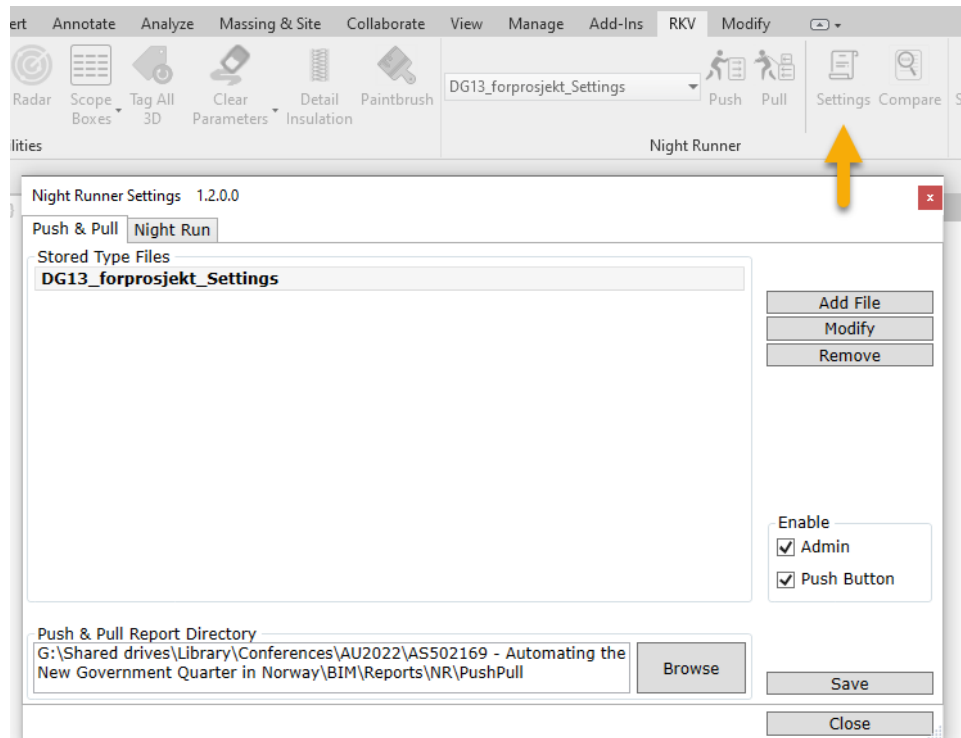
Workflows

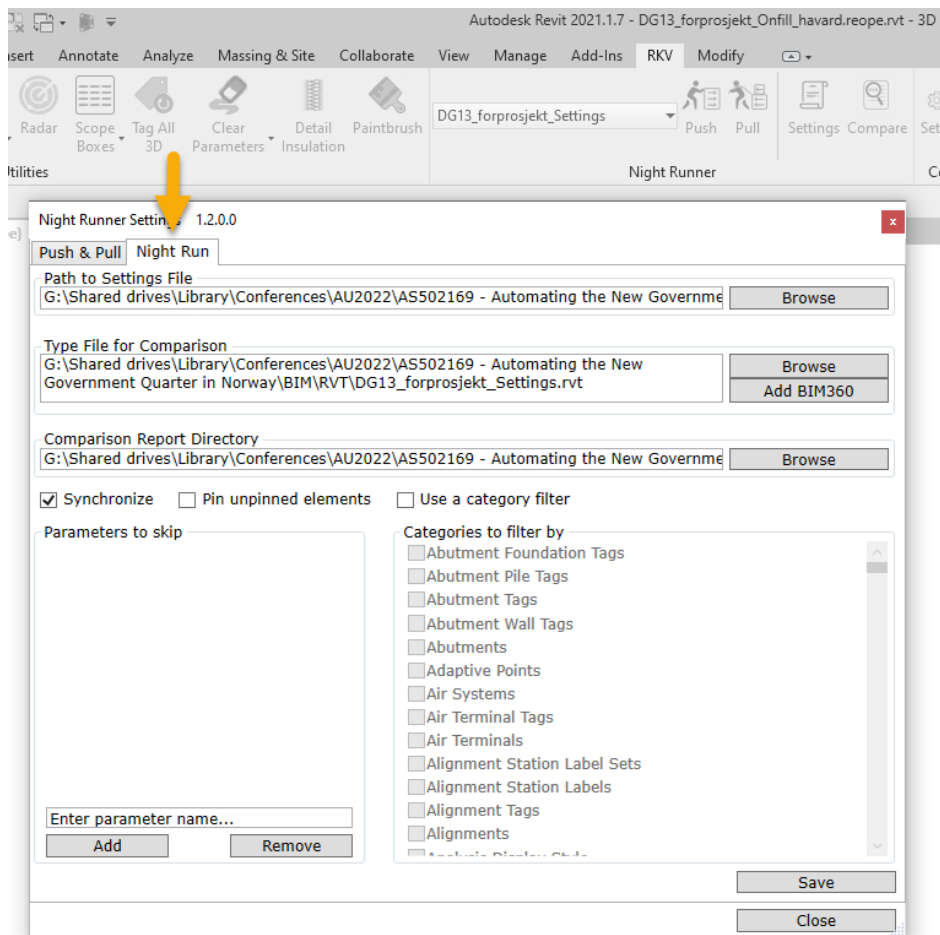
While we created (and still are creating) several automation workflows, these are the most valuable processes to date.

Model Standardization

Revit's biggest problem since day one, and Morten's main headache from previous projects, were the disconnectedness of different Revit files that on creation were the same, but after time started deviating greatly. The only way to handle performance issues on large projects is to divide the projects in separate files. This is also Autodesk's number one advice. And once you do, they start living separate lives, with Families, Types and Settings having differences. This in turn means that it gets difficult for design teams to deliver consistent models, graphics and data.

The process we built was based on the use of a Settings Revit file. The Settings file contained all project wide standard Families, Types and Settings. Every night at 01:00 Revit Batch Processor opened all the Revit models of a discipline and compared their content to what was currently in the Settings file. If the add-in found a deviation within Type Parameters, Settings or item counts (in families), it would use different methods (based on the element type) for updating the design file content. For instance, if someone changed the text type of a dimension system family in the Settings file, and the comparison process identified the deviation, it would change the setting in the design file. If it found a deviating item count in a family, it would load that family. An example of this could be that you opened a Door Family in the Settings File and added a Sweep or a Symbolic Line. When the Comparison process started at night, it would detect a deviation in the count of Sweeps or Symbolic Lines between the Settings File and the Project Files, and hence load the updated Family onto all Project Files automatically. This way we avoided all the duplication problems of using Copy and Paste with Clipboard and Transfer Project Standards.





This workflow of course meant that anything in the Settings file would overwrite anything deviating in the design files. That was the point. But how can anyone on the project then make that change? Does everyone have to work in the Settings file?

To make it easier for the entire team to participate in the buildup of the project standards, and to invite for involvement and autonomy, we built two tools in the add-in; Push and Pull. Push would send any element based on selection or a list to the Settings file. So basically, anyone wanting to make a change to the standards, could make it in their design file, push it to Settings, go home to make dinner, and sleep knowing their standard would spread like a virus in all the other design files, overwriting anything that was before.

The same way anyone could Pull anything from the Settings file at any time in the UI of Revit without having to find and open it and mess around with the various ways of moving content from one file to another.

In the beginning of the project, we were generally rather scared that the process we built would get out of control and change many things that we didn't control. We were also worried that someone would make a mistake and spread their wrong data in all the Revit files across the project. Over time, however, we learnt to trust the process, control and master it, and we also started understanding that we're basically applying the same principle as Revit has at its core, only on a macro level: anyone can change anything. And with trust, people learnt the tool, and very few mistakes were made.

Exports and imports

Of all tasks that architects, engineers and BIM managers perform, exporting and importing must be one of the most meaningless manual tasks. Even cleaning the coffee machine at the office has more value than staring at a progress bar processing IFC elements. Yet, you'd be surprised by how much time people spend doing just that. (I guess one can always clean the coffee machine *while* waiting for progress bars, but just one person at the time.)

We set up configurable IFC, DWG and PDF exports from all main Revit models with RTV Tools. The result was that everyone in the project knew where all the latest information was. All IFC files were always updated from the latest design changes from yesterday. I cannot underestimate the positive impact of this, as it removed all the questions about what the latest exports were and where people could find the current state of design.

We also built custom IFC imports (or, rather Links) for Revit, for eliminating the waiting time people experience when they reload an IFC Link. IFC link regeneration isn't just triggered by file date/size, but several variables like Revit version, IFC template file, colour of your desktop background etc. Being a large project with well over 100 architects and engineers, this means someone, somewhere, always were regenerating the IFC links... and each time it did, a designer would have to wait for the process to finish anywhere between 15 and 90 minutes. A lot of coffee machines can be cleaned in 90 minutes.

The process we built used the converted ifc.RVT Revit files as links, replacing linking of the IFC files directly. This worked well because as mentioned above, all the IFC files from the non-Revit disciplines are all auto exported every night.

The Import process started with opening an empty Revit file and linking all new IFC-files from a list of folders, hence creating all the ifc.RVT files. We also built in a post process that changed the Appearance Assets of all the IFC generated materials based on a Revit template, making it much easier for the designers to render with Enscape directly from Revit.



Figure 10 D-block office space

Data flow/model enriching

The need for the third and last major automation workflow was identified when the project was running, not prior, like the two previous workflows. The need for adding metadata to model elements, as mentioned in the Preparations and Analysis section above, became somewhat of a manual process at first, and scripted workflow after a while. We developed a lot of Dynamo scripts to copy and write Text Parameter Values from elements in one place to elements in other places. When we realized how important these values were for the delivery of quality data for model-based construction and later facilities, operations and maintenance, we decided to rebuild the scripts as a scalable and robust add-in.

We mainly used this process to achieve two different goals. One was automating the enriching of technical data in the “main” design models. For the architects this could mean getting Fire Rating

information about a certain set of Doors from the Fire Consultants, without anyone having to do anything but model a Fire Data Container in a Fire Model in Revit. We did the same for Acoustics, Security and Technical Door Properties. Without almost no effort, the architects auto-produced information rich IFC models every night.

After this process had run successfully for a while, we extended it to also produce Level based section models from the Revit files. The reason we needed this was to export smaller sections of the building for construction and onsite access. This also proves easier for the workers to navigate the model, as they will only see where they're supposed to actually work. This was done by establishing volume models with Mass in a separate Revit file and copying location data from the different Mass Elements to any Element of any Category whose center of gravity was inside a Mass. This resulted in every element in the entire project having information about where it was, as a unified parameter. (Try using View Filters for this. Doesn't work. Same with Section Boxes.)

Image

The result of all of the above was, and still is, the automated production of consistent, coordinated, information resulting in rich, targeted model outputs. Better data quality, fewer resources. And in the end, better design, because the designers didn't have to do any of this.

5 SUMMARY

Summing up the main learning goals from this session:

Strategies for automated data flow between Revit models, IFC deliverables and different stakeholders in large projects

- Start out analyzing what purposes the models actually are going to be used for
- Identify stakeholder requirements and project resources, always using industry standards when possible
- To find possible automation scenarios, use the analysis and identified requirements to set up data flows

How to set up centralized template models that contains and synchronizes all model settings, types and families

- Figure what metadata, family types and settings you need to keep common within the project
- Call Håvard
- Make sure people can easy upload changes from live models, so you don't risk pissing off the team, when the Night Runner automatically overwrites model inconsistencies

Methods for setting up and executing automation infrastructure that helps design and construction teams

- Revit Batch Processor together with Python scripting is a very robust combination
- Dynamo is great for prototyping
- Use server-side scripting based on input from Excel, for easier access to team members who can revise script values without much scripting knowledge

How to implement project purpose custom-built design tools under pressure of constant project deadlines

- Make sure you've got a development team available that is connected to the project, preferably directly but as a minimum should have knowledge of its organization and requirements
- Create lines of command that allows for quick decisions based on evaluation of return of investment in the project
- Deliver on time and on budget in order to get funding for the next step!

6 APPENDIX

6.1 TFM

Cross-disciplinary tagging format or TFM, is a Norwegian standard ([NS 3457-7:2021](#)) for coding building elements for referencing product documentation etc. in external databases. This way, you can reduce the amount of data within the model or the need to alter the model for building management operations. The standard is currently only available in Norwegian.

6.2 SIMBA

Statsbygg has developed a comprehensive guide to its BIM requirements, SIMBA, available in English:

<https://sites.google.com/view/simba-bim-krav/simba-2-1/simba-2-1-veiledning>

This is the latest version of requirements, which describes object classes and their properties according to the IFC4 standard. The New Government Quarters uses a previous version, which relates to the IFC 2x3 standard.

6.3 Geometry levels

We found the LOD-specification from BIMForum too complex to navigate and not necessarily relevant in a Norwegian context. In our experience, we don't need as many iterations as described in the LOD specification, as the design process moves forward so quickly there isn't time to verify each level.

Based on work previously done for buildingSMART Norway, we elaborated on existing text descriptions for geometry levels for all disciplines, sorted according to the well-established standard for building element codes in Norway ([NS 3451:2009](#)). We've ended up with the need for three geometry levels for the design team's deliveries, as manufacturing models are produced by the manufacturers themselves according to their needs.

This would be one of our basic "building blocks" for defining which family types from the central library settings file were relevant for the different project models at different stages of project development.

6.4 Model Maturity Index

The Model Maturity Index – MMI – is a local Norwegian industry standard describing how "mature" an object within a BIM is in relation to project development and quality assurance processes, defining what purpose it can be used for.

Mapping the MMI levels to the project schedule, we used MMI to validate both geometry and information level requirements.

The MMI guide is currently under revision, and the new version will prove a valuable expansion of the different levels that much better adapts to the typical design and build process
([link will be updated](#))