buildingSMART Danmark

IFC GUIDE

EXPORT - IFC2x3

ENGLISH VERSION

CONTENTS

INTRODUCTION: WHY AN IFC GUIDE?

IFC is the common format that enables knowledge sharing and ensures that construction data can be communicated between disciplines and across software platforms — today and for many years to come.

In Denmark, IFC has become a common file format for exchange of models between disciplines, coordination between design consultants, quality assurance, quantity surveyors and coordination of handover to building owners and operators. But for regular users it can still be difficult to keep track of the many options for managing geometry and data when exporting building models to IFC.

With this guide, we hope to make IFC more accessible to ordinary users, so that the use of IFC doesn't require help from the office BIM manager or the construction site ICT manager.

The IFC format was developed by buildingSMART, which in Denmark is represented by Molio and building SMART Denmark.

This guide is created by Molio in collaboration with Nicklas Verdier Østergaard (SHL) and Thomas Graabæk (GRAPHISOFT Center Danmark), with additional help from Peter Tranberg (NTI), Aidin Irandoust (Municipality of Hillerød) and Erik Falck Jørgensen (DTU Byg).

For this IFC guide a building model has been created, 'Molio House', as a sample file in both Revit and ARCHICAD and as IFC files from both programs. This will make it easier to follow the guide's descriptions, by allowing you to simultaneously see the results of the IFC exchange. Revit and ARCHICAD are different programs so the IFC files they create are not exactly the same. For more advanced users it might also be interesting to dig deeper into the differences in the IFC files from the two programs.

In order to limit the scope of this guide, the project team initially chose only to describe how to export IFC files from Revit and ARCHICAD for project coordination and delivery purposes. Therefore, the IFC standard IFC 2x3 Coordination View 2.0 is used. Based on the project group's experience, it has also been decided to describe how CCS CCS (Cuneco Classification System) codes are added to the building components of the models and included in the IFC exports.

This IFC guide is the first of what hopefully will be a series of guides on IFC from Molio and building SMART Denmark. This guide focuses on how to ensure high quality IFC exports from Revit and ARCHICAD. Future guides will focus on other aspects.

Software versions used in the illustrations:

- Revit 2018 and 2019
- ARCHICAD 22
- Solibri Model Checker v. 9.8
- Navisworks 2019

Imprint
IFC Guide to ARCHICAD and Revit
Released April 2020
Revision 2

Publisher
Molio - Construction Information Centre
Lyskær 1, 2730 Herlev
Phone 70 12 06 00
info@molio.dk
molio.dk

The guide is prepared by the working group listed below with the backing of a number of people in Molios building smart networks:

- Thomas Graabæk, GRAPHISOFT Center Danmark
- Nicklas Verdier Østergaard, Schmidt Hammer Lassen Architects
- Peter Tranberg, NTI
- Brian Rasmussen, Contiga Tinglev

With this guide we hope to make IFC more accessible so that the use of IFC does not require assistance from the design studio's BIM manager or the construction site ICT manager.

WHAT IS IFC?

IFC (Industry Foundation Classes) is a common file format developed to exchange information between AEC professionals across software platforms. An IFC file is an object-based data model whose purpose is to meet the construction industry's need to share knowledge and information. IFC is a neutral and open file format and is not controlled by a single company but is developed by the international not-for-profit organization buildingSMART. In Denmark, buildingSMART is a part of the not-for-profit organization Molio.

You can compare the IFC format to other open file formats such as PDF for print files, JPEG for image files, MP3 for audio files and MPEG4 for video files. IFC is the open standard for BIM. Using open file formats, such as IFC, allows users to access data from a wide range of software platforms (including free viewers) running on different operating systems (e.g., Mac, Windows, Linux, iOS and Android) and various devices (traditional computers, tablets, smartphones, etc.). This flexibility ensures maximum data availability throughout the whole life cycle of a building — from concept to operation and maintenance.

Due to buildingSMART's focus on standardization and collaboration across various software platforms, the Danish government has required use of the IFC format on all major, publicly supported construction projects. This is done through the Danish ICT Regulations.

Increasingly public and private developers have begun to focus on the opportunities embodied in the IFC format and are increasingly requiring and use IFC. While some BIM programs are highly specialized, others span multiple professions, project types and services. IFC allows the team on a construction project to use various BIM programs, giving them freedom to choose the software that best supports their specific needs, their company set up and allocated budget. Since various software developers have implemented IFC in different ways, there are differences in how IFC is imported and exported, depending on the specific program you use. Therefore, a good understanding of how to import and export IFC files with the least possible data loss is important.

Examples of using IFC

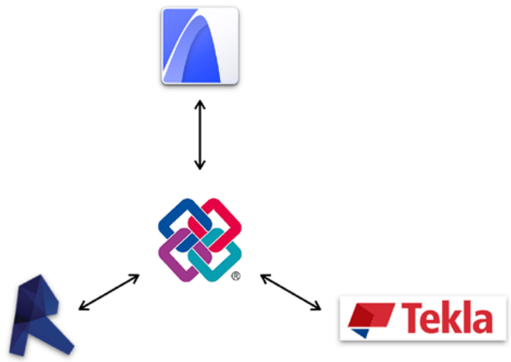
You can see examples of the use of IFC, where the AEC professionals on a given building project use different BIM programs. For example, the architectural advisors might use ARCHICAD, the structural engineers do their work in Tekla and the installation engineer uses Revit MEP. Each disciplines consultant can export an IFC file from their specific model, which is then shared with the other consultants and the contractors. Also, the consultants can perform subject-specific calculations/simulations directly on other consultants IFC files or import/link the IFC files into a federated model, and in this way verify that their solutions are compatible with the rest of the project teams.

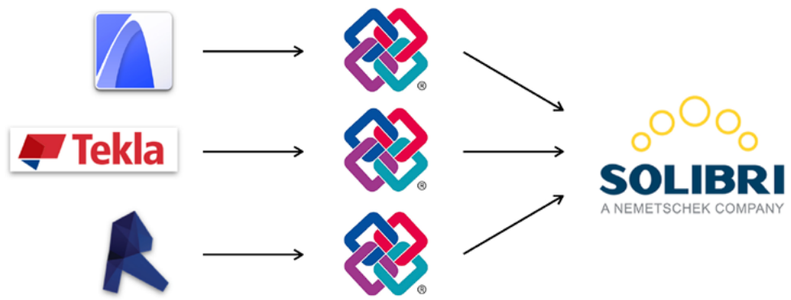
Importing a series of IFC files into your own project file is not always the best solution, since it can make the project heavy and difficult to view. The imported models may contain too much data that you don't need. Therefore, it might be a sensible solution to combine all discipline models into a common federated model in an independent BIM program, which is created for coordination and quality assurance. This could be Navisworks, Solibri Model Checker or Tekla BIMsight. Since each discipline model might already be heavy to work with, it will facilitate the exchange on large or complex projects. This makes it possible to automate quality assurance in order to identify issues between two discipline models and communicate these to the project team via eg. BCF files or "issue management" programs such as BIMcollab.

Various versions of IFC

The IFC format is available in several versions. The latest version is IFC4, while IFC version 2x3 is still the most widely used. Which version of the IFC format that is used in a specific project is most often defined in the project's ICT specifications. Learn more about the IFC format, buildingSMART and other buildingSMART standards here:

http://anvisninger.molio.dk/





PROJECT AGREEMENTS

Most problems with coordination arise from lack of agreement before BIM models are initiated and are mainly related to matters such as a common coordinate system, rotation from north and project levels.

In a Danish context, Molio's ICT specifications (A102) are frequently used as a basis for ICT specifications. BIM-based collaboration works best with clear agreements between the actors. Therefore, it is always recommended that an ICT specification and/or ICT process manual be prepared specifying what, when and how information is to be exchanged. buildingSMART and Molio both have good tools for these types of agreements.

Naming of files

A consistent and coherent file name for BIM models (as well as the model's objects) ensures that IFC files are easy to understand for internal as well as external project participants.

Information Delivery Manual (IDM)

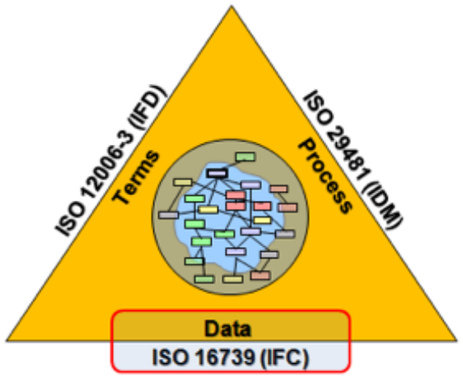
IDM is a method of specifying which data is to be exchanged between actors but also how and when data is to be exchanged. IDM is therefore an important part of the ICT specification.

During any given construction project, a massive amount of information changes hands. Work on areas, spaces, quantities, energy calculations, climate simulations and much more is conducted on a daily basis. There is a need for digital deliveries to take place in a way that ensures that the individual actor knows what is to be delivered and what they can expect to receive at any given time.

An IDM describes the process and which data is to be exchanged, by whom and when. The two Danish IDMs from buildingSMART on "Areas" and "Collision Control" address the process part, while the IFC file contains data.

Molio has made 3 IDM templates available that can be used as inspiration:

https://anvisninger.molio.dk/Gratis-vaerktojer/Udveksling



Model View Definition (MVD)

Since an IFC file usually contains a very large amount of data, buildingSMART has defined a number of specific data sets, called Model View Definitions (MVD). MVDs makes it easy to ensure that the relevant data comes with the extracted IFC file. Coordination View 2.0 (CV 2.0) is one of the most used MVD's and is specifically designed for coordination and quality assurance of building models.

CV 2.0 specification

IFC 2x3 Coordination View Version 2.0 is the first and most widely used MVD and was developed by buildingSMART International. CV 2.0 is primarily developed to support the coordination work required during the detail design stage between separate disciplines such as architects, engineers and contractors.

This is precisely why CV 2.0 is an obvious format for conducting an overall quality control, such as checking for collisions, classification and coordination of quantities between the discipline models. In addition, CV 2.0 is frequently used for conducting consistency checks, design validation, void coordination and more. CV 2.0 contains a hierarchical structure for buildings, their constructions and installations with their associated graphic representation, feature sets, materials and information.

As mentioned earlier, CV 2.0 is designed for the exchange of discipline models and therefore takes into account that the subset of information is different for specific disciplines. This means that each consultant only exports the relevant information. For example, the discipline model from the plumbing engineer will not include the export of columns as it is not relevant to this discipline.

There is a scheme for certification of software in relation to CV 2.0. You should make sure that the software application you are using are certified, which is most often shown with one of the labels below.

Structuring IFC Models

As shown in the figure below, the IFC file's building model is hierarchically constructed. Thus, the following conditions should be respected:

Project: Models must only contain one IfcProject and IfcProject.Name must be completed.

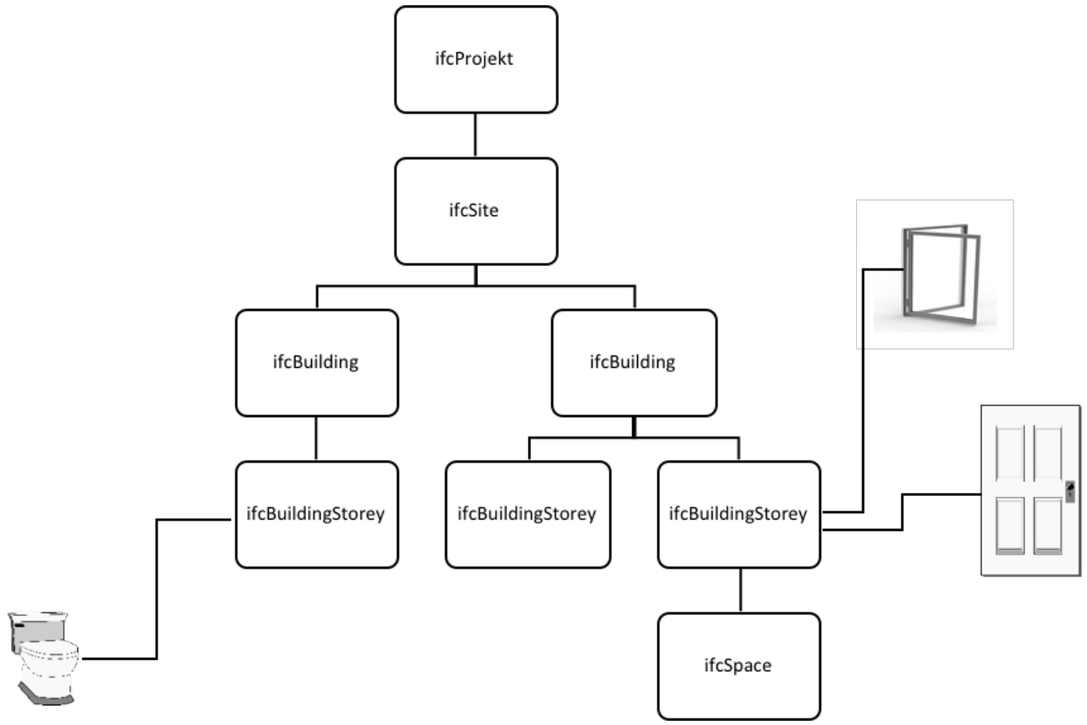
Site: Unless otherwise agreed, the building model should contain only one ifcSite for each project. In addition, the landmark number is specified on IfcSite. LandTitleNumber.

Buildings: One or more buildings must appear on a plot. Please note that buildings must be defined independently as IfcBuilding.

Levels: Building models must contain one or more levels (IfcBuildingStorey)*.

Spaces: Space in building models must be modeled as IfcSpace objects and must belong to the level on which they are located.

* IFC uses the British English spelling 'storey' for the levels of a building



Development of the model

Various methods have been developed in order to easily communicate the level of model development at the model parts level and building level. Often the level of development is used as a term for the whole building model, but it is equally relevant to define the individual building parts. Sometimes the windows should be modeled to one level, while the outer wall should be modeled at another.

The development levels of the model are specified in the ICT specification.

The European standardization organization CEN is currently working on developing a common European standard for a method for specifying the level of BIM information to be exchanged.

Below are the most widely currently used methods in Denmark and they will be used until the result of the European standardization work is available.

CCS Information Levels are described in the CCS Information Levels sheet

LOD (Level of Development):
http://bimforum.org/

DiKon / BIM7AAs's Specification of Building Parts:
https://www.dikon.info/en/publications/
http://www.bim7aa.dk/DIKON_BIM7AA_Bygningsdels-specifikationer_UK.html

BCF

buildingSMART also develops an open file format for the exchange of error reports, for example in the event of a clash between discipline models. BIM Collaboration Format (BCF) files can be imported and exported from many of the most common BIM applications. This way, BCF files form the basis for the process of eliminating errors in the project material.

BEFORE EXPORTING TO IFC

Overall structure

With the amount of data that a building model contains, the structure of the content is important in order to make the relevant data easily accessible to collaborators as well as to the person in charge of preparing the material. The more people that are working on a given project, the greater is the need for a clear structure.

Modeling tips

In general, there are a number of tips on how to obtain a model suitable for exchange:

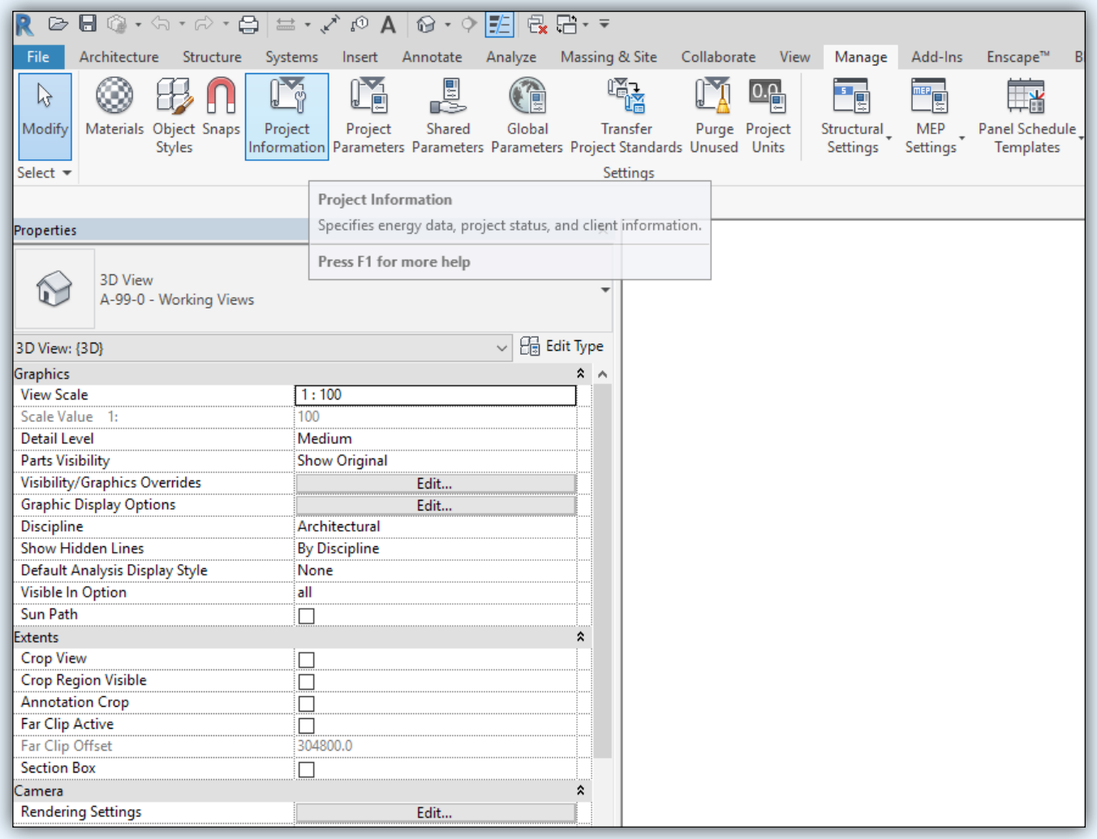
- Make sure models from the same user are modeled consistently.
- Coordinate that all discipline models use the same base point (0,0,0).
- Make sure levels are consistent in number and placed equally in each discipline model ("vertical sectioning").
- There may be advantages in splitting objects at levels and adding the level name as an object property.
- Use the building modeling software's tools for their intended purpose, (eg windows with the window tool) or reclassify the objects accordingly.
- Check that room objects have correct room heights.
- Avoid unnecessary objects in the model.
- Check for duplicates (two identical objects located in exactly the same place) and unintentionally intersecting geometry (two building parts overlap).
- Export only the necessary parts of the model.
- Model items at the level to which they belong.
- Divide objects into as few sub-elements as possible (with the exception of splitting by level).
- Export the model with the lowest level of detail possible according to the recipient's needs.

Project info

Basic information about the project can be entered in the project file, which makes it easier for external parties to review the contents of the model. It can be project name, landmark, contact information, etc.

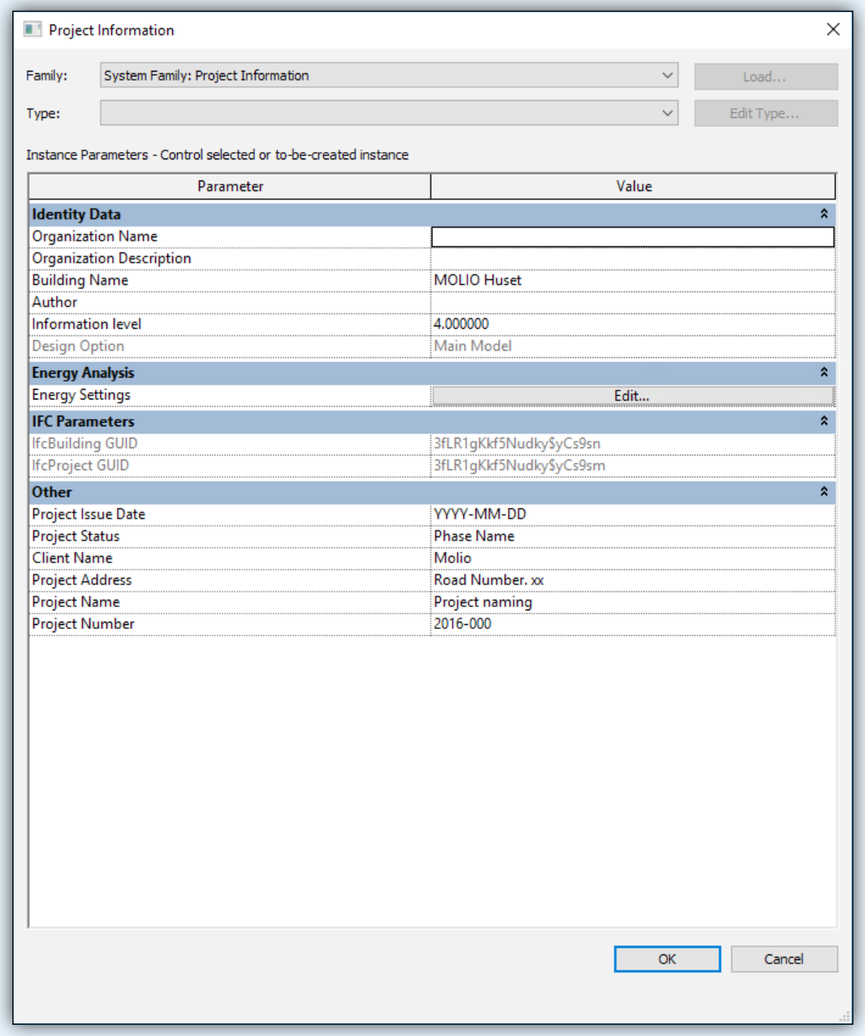
The most important information that should be entered is:

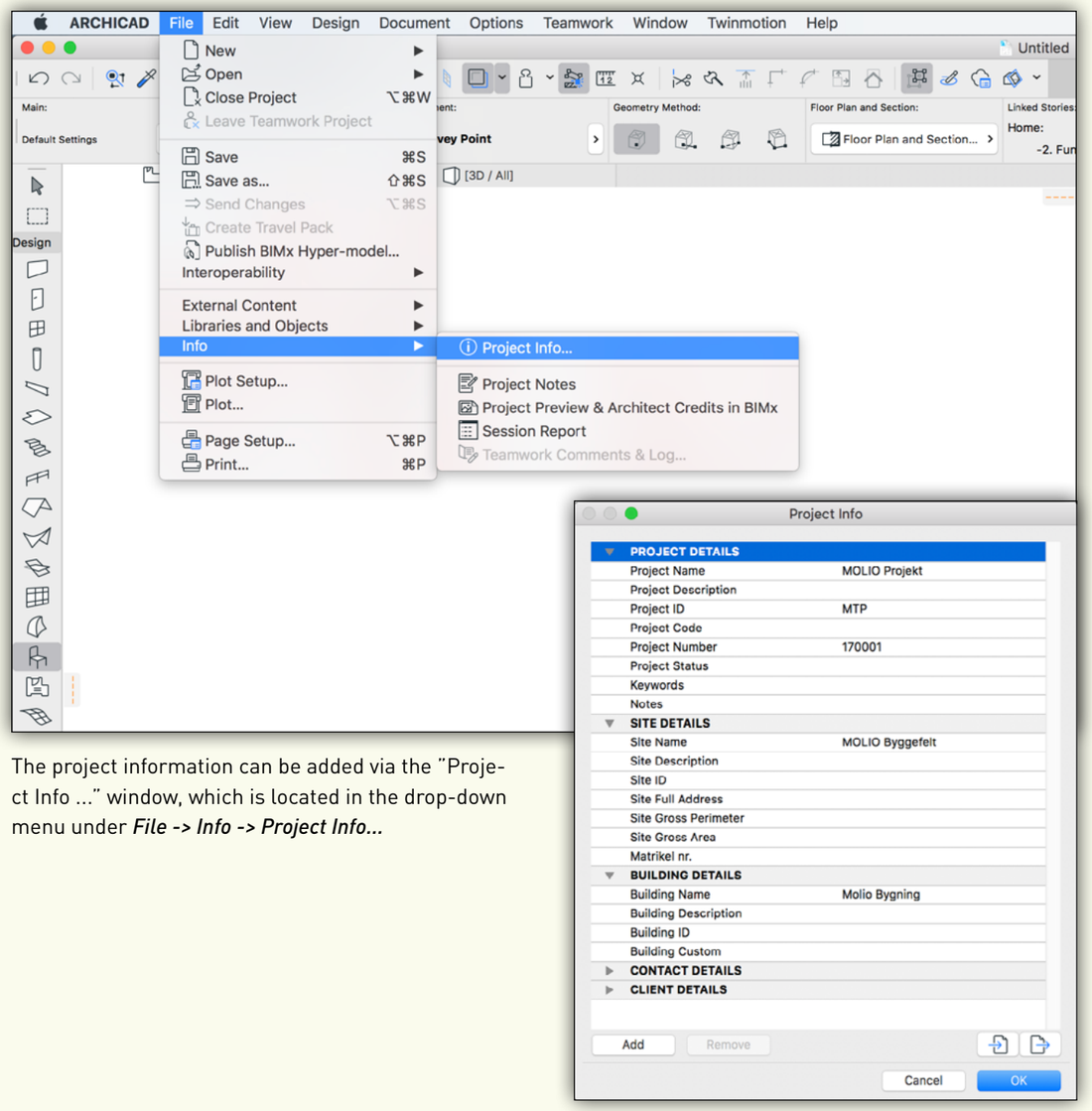
Project name
Project number
Building name
Plot number



Projektoplysningerne kan tilføjes via vinduet projektoplysninger, der kan findes under fanen Manage -> Project Information.

This window appears:





The project information can be added via the "Project Info ..." window, which is located in the drop-down menu under ***File -> Info -> Project Info...***

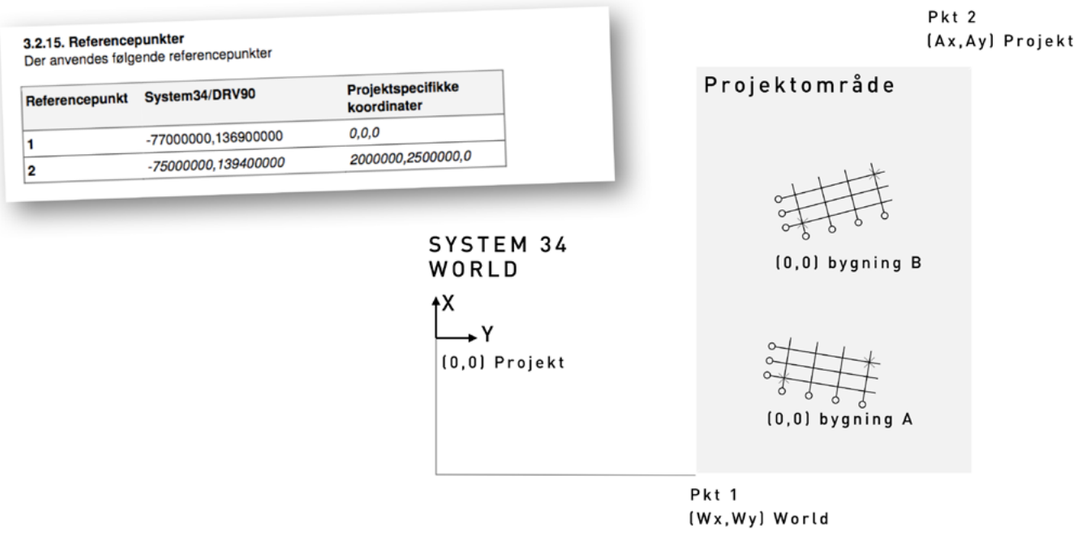
PROJECT COORDINATES AND SURVEYORS' COORDINATES

In order to position the project correctly in relation to the real environment, and to ensure that the discipline models are placed so that they automatically fit together, they must be placed correctly in the building modeling program. This is done by means of a common zero point. Whether the zero point is the surveyors' coordinates, or the bottom left grid intersection is not important as long as all parties agree - and preferably has documented this in an ICT specification and/or ICT process manual.

There are differences in how the various building modeling programs handle a zero point that is placed far away from the building, which is often the case with surveyors coordinates in their DWG drawings, where the "World Coordinate" in AutoCADis located many thousands of miles from the building.

In ARCHICAD, this works well, while in Revit you have establish a relationship between the project and the country coordinator using a "Survey Point" object. If this solution is desired in ARCHICAD, a "Survey Point" object is also used.

It is recommended that, regardless of the solution, a single common local zero point which is close to the building, is defined in the project's ICT specification or ICT process manual, and describes the location of the location (s) of the local zero (s), as seen in the example below.



Revit operates with three types of coordination points, of which only two are visible:

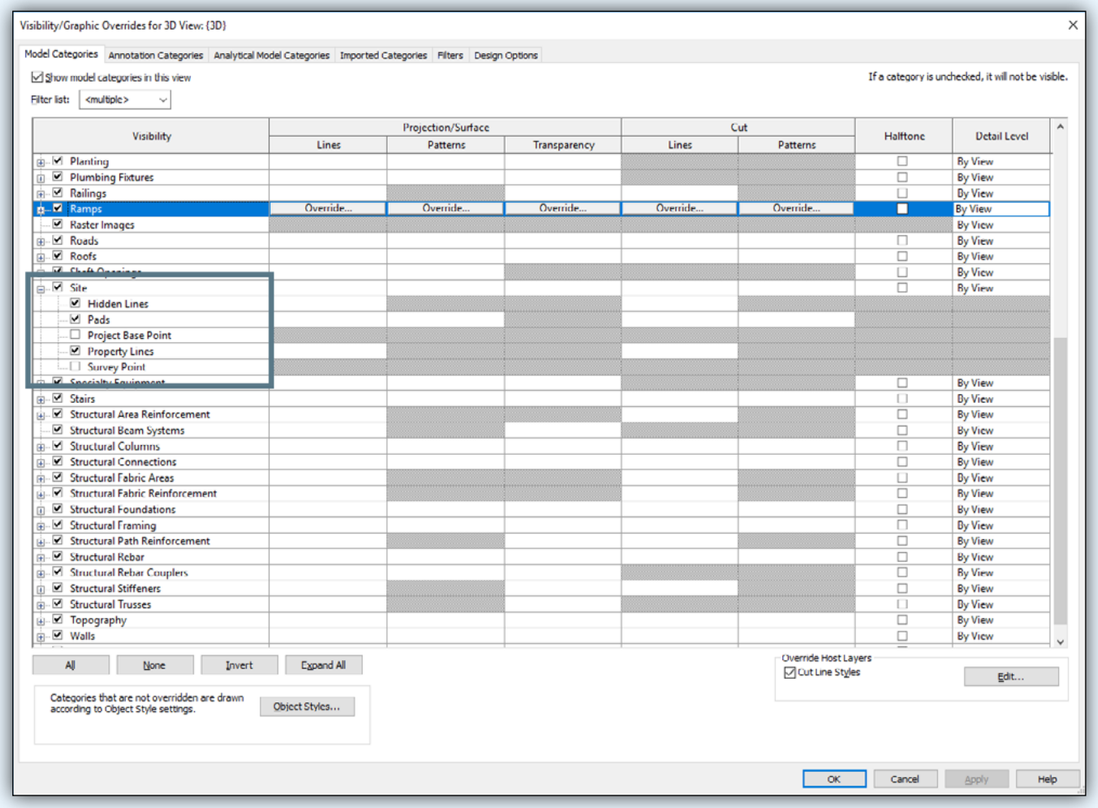
1. Project Base Point - used as 0,0,0 for the project
2. Survey Point - surveyors' coordinates that tell where in the world the building is located
3. Origin Point - Inner Base Point in Revit. This point is not visible in the project.

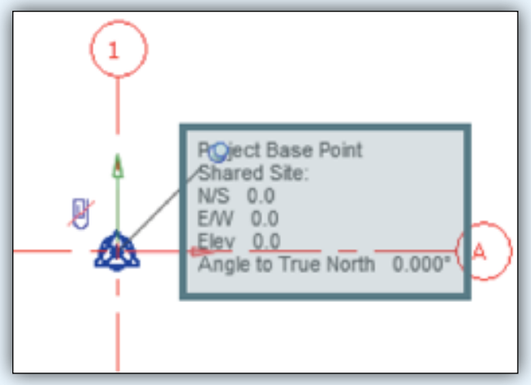
It is advantageous to place all three coordination points in the same place and mark the point with two module lines (x, y).

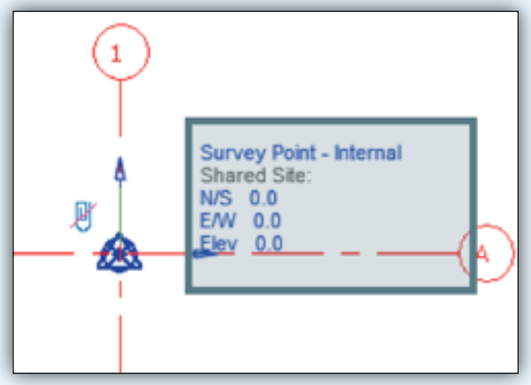
The location of the building is then coordinated between disciplines and should be located close to Project Base Point.

Link:
https://knowledge.autodesk.com/support/revitproducts/learn-explore/caas/simplecontent/content/coordinates-systems-20-miles-limit-the-maxdistance-input-part-i.html

In Revit Coordination points can be turned on via the Visibility window. Under the Site category, Project Base Point and Survey Point can be displayed by checking the box.

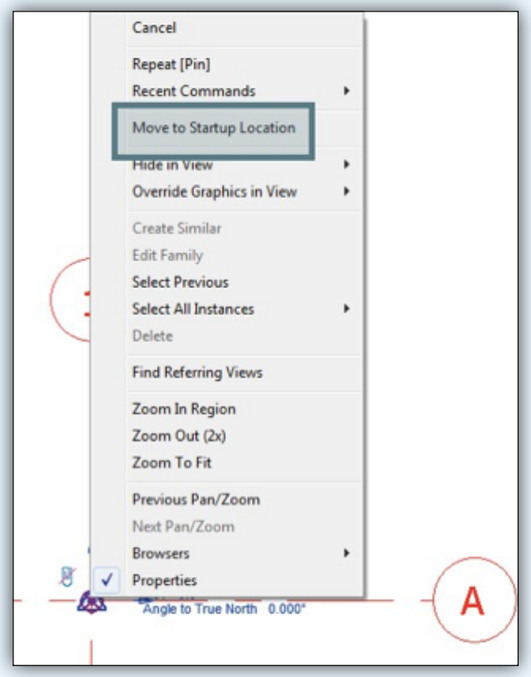






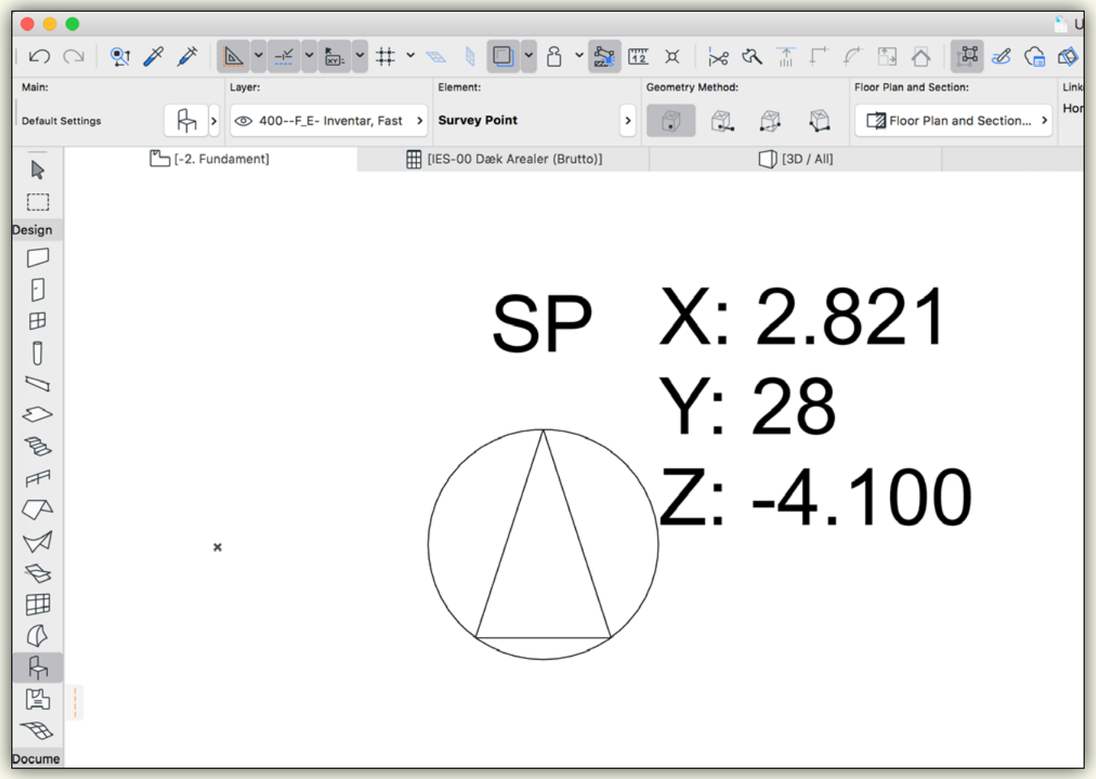
Two icons appear in the workspace. The circle is Project Base Point, the triangle is Survey Point. All values out of the box are zero. To check if Startup Point is the same point as the other two, it is possible to move Project Base Point to the starting location (original). This can be done by selecting Project Base Point, right-clicking and selecting Move to Startup Location.

When this setting is dimmed, Project Base Point is already at the starting position.



The ARCHICAD Project Origin is shown with a black cross in plan view. Here, the surveyors coordinate, or a local, project-specific origin can be located.

If you don't want the IFC file to use ARCHICAD's Project Origin, you can instead place a "Survey Point" object that defines a new origin point for IFC exports:



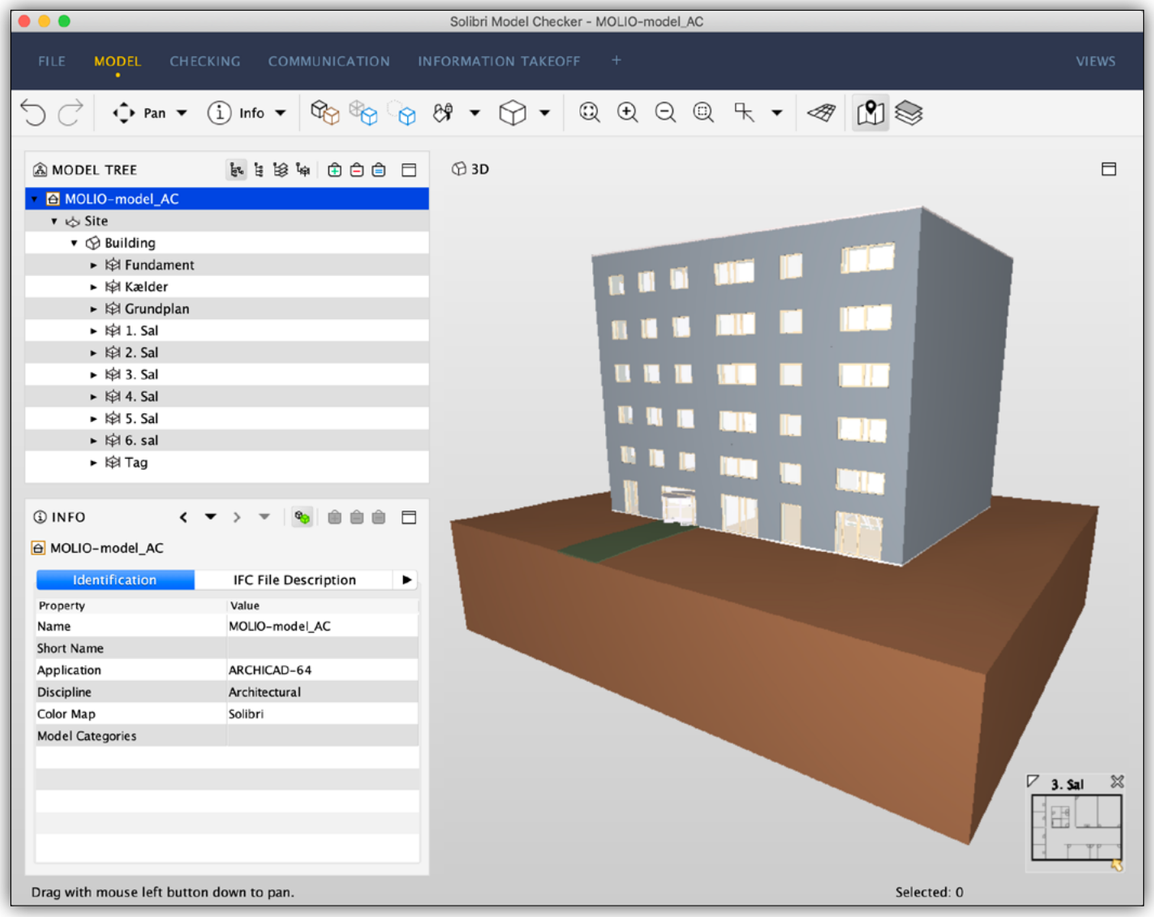
VERTICAL SECTIONING

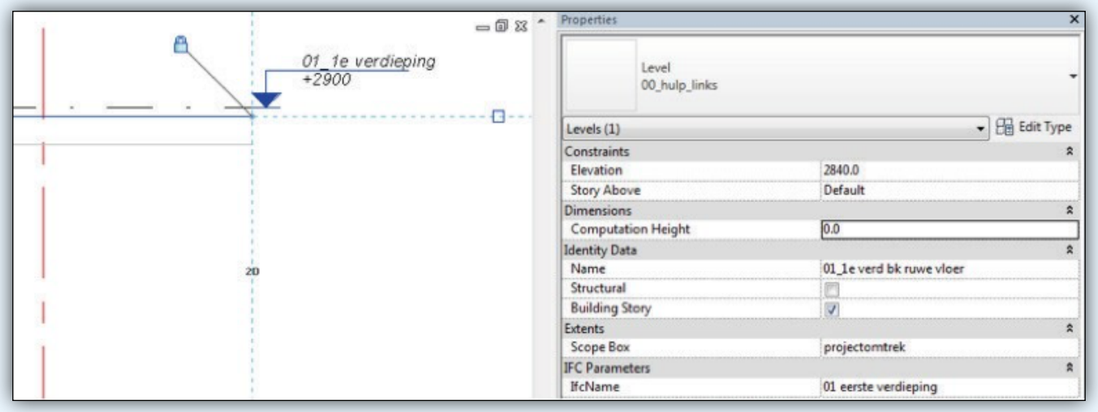
It is important to have a "vertical sectioning" defined in the ICT specification, so that all actors define the project levels in the same way. It is a question of naming the levels (referred to as the ground floor, eg "GF" or "00") and the elevation of each level. It is also important to define whether, for example, the level lies on the upper surface of a finished floor or the upper surface of a concrete deck.

It is recommended that the official levels of the project are described in the project's ICT specification and/or ICT process manual:

- Level number (IfcName)
- Level name (IfcLongName)
- If appl. roof description (IfcDescription)
- Level elevation

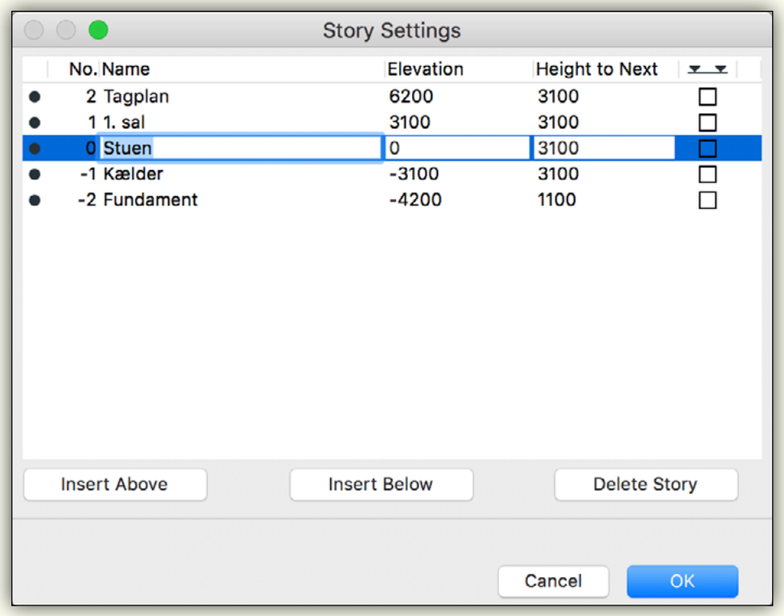
After export the IFC model's data hierarchy looks like this:





In Revit levels have the "Building Story" property. Only levels with Building Story selected will be exported as levels when the model is exported to IFC. Only the levels that are agreed for exchange and which are named correctly should be exported.

It is not unusual for non-story levels to be modeled in Revit. For example, the architect might have created a level for placing suspended ceilings. The number of secondary levels must always be kept to an absolute minimum, as this can lead to unnecessary errors in exports (not only in IFC exports).



In ARCHICAD, the levels are created under "Story Settings". Only the chosen levels are created.

CLASSIFICATION AND IDENTIFICATION

As building models contain large amounts of geometry and data, it is important to structure the content. Building modeling software automatically categorize the content through the tools used to model. For example walls, floors, doors, windows, stairs, roofs, columns and beams are automatically categorized in their own specific categories in the building model.

The IFC format also has built-in categorization, where the building parts are divided into "IfcWall", "IfcWindow" and so on.

In order to be able to divide data into several categories and structure data for use, for example, tender lists and FM systems, a number of classification systems exist that allow the model's objects to be placed in classes based on different systems. The most commonly used classification systems in Denmark are probably CCS and FM-classification (forvaltningsklassifikation).

However, earlier Danish classification systems, such as Sfb and DBK, are still used in many places. In addition, there are a number of foreign classification systems, such as UniClass and OmniClass, which are widely used in many parts of the world. The BIM7AA system, developed by a number of architectural firms in Aarhus, is also widely used. BIM7AA is a system for specifying building component types based on a predefined list of numbers.

Using one system or another doesn't make a big difference in terms of software, but it is important to define at least one classification system used by all the project's consultants, for example in order to extract quantities for use in calculations.

In this guide we have chosen to use Molio's CCS as a classification system.

For Revit, Molio has created an official Shared Parameter file (https:// ccs.molio.dk/News/ImplementationInRevit), which everyone should use for CCS classification and identification.

Almost all parameters are available in two versions. One is for "Type" and the other is for "Instance". You should only use either Type or Instance but choosing one or the other depends on your work processes. The architect and the structural engineer will most often be able to settle for the version for Type, while the installation disciplines will often need the version for Instance.

This is because, for example, pipes in Revit are created as one type regardless of their dimensions. This does not correspond with the typical organization of, for example, a tender list, where each dimension typically defines a new type.

In this example we focus only on CCS classification and CCSTypeID as identification.

CCS classification

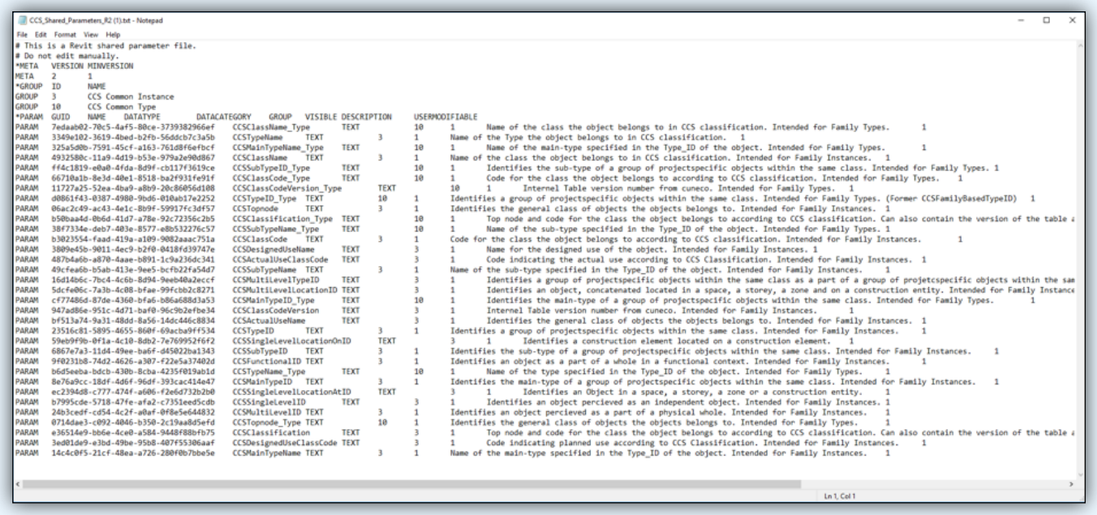
CCSClassification is composed of ***<CCSTopnode ([L])><CCSClassCode (QQA)><(;)><(CCSClass Name)>***

CCS identifikation

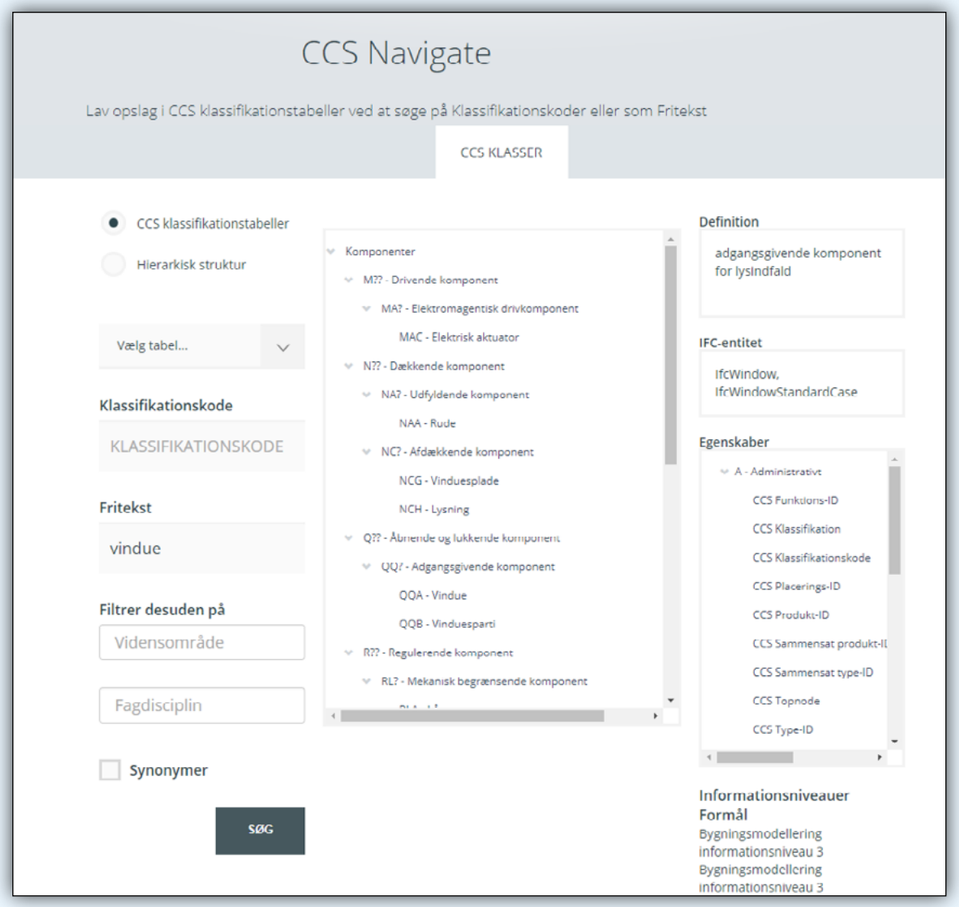
CCSTypeID is composed of ***<CCSTopnode ([L])><præfix (%)><CCSClassCode (QQA)><Løbenummer (01)>.***

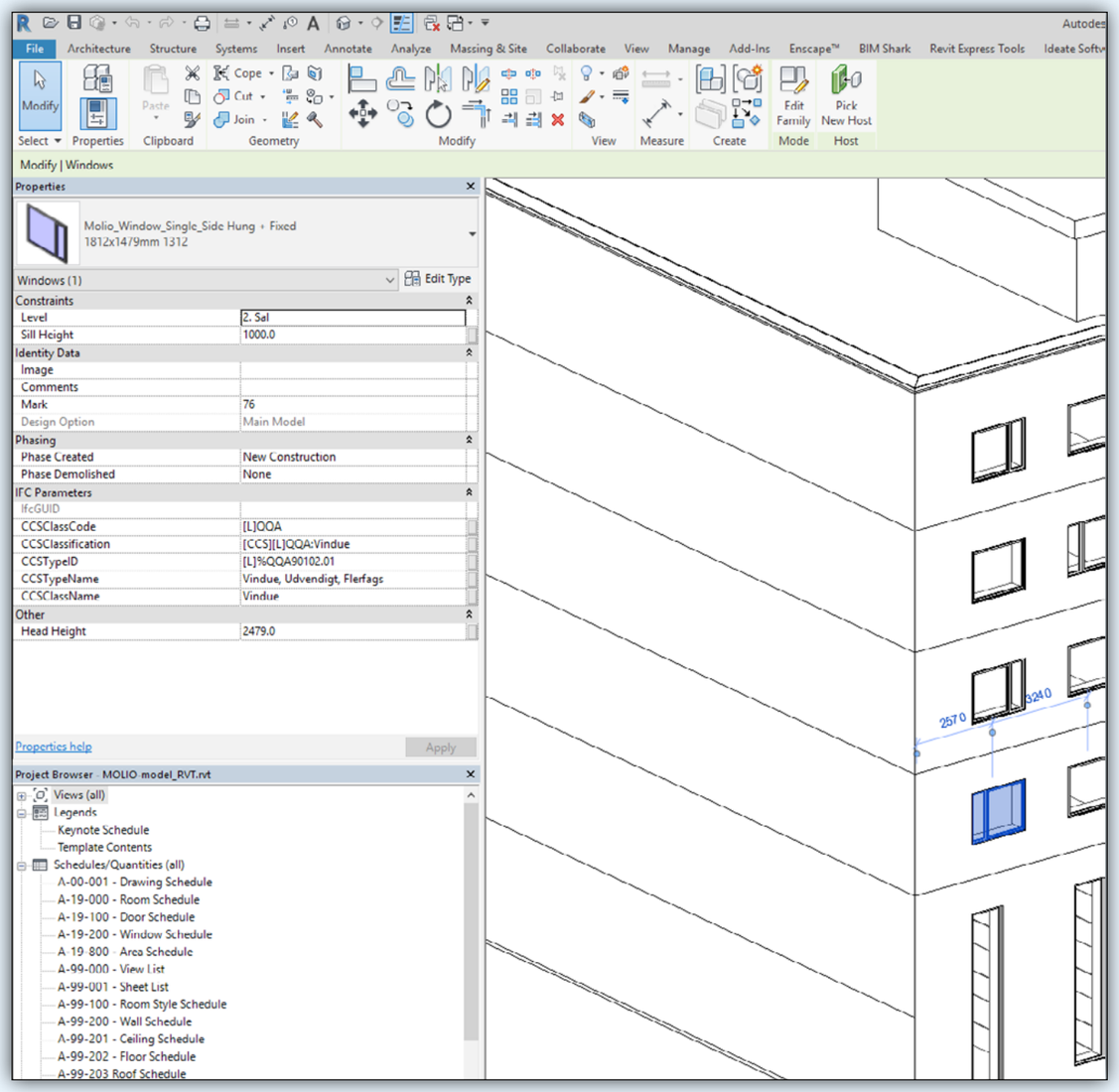
The use of top node is optional in CCS. In Revit as a minimum you need to create CCSClassCode, CCSClassName and CCSTypeID as project parameters based on Molio's Shared Parameter file. In CCSClassification, ";" is inserted to get Revit's IFC export to correctly split the parameter into "Reference" and "Name". See the example on page 14.

In the example, values are entered manually, but there are a number of software applications for automatic handling of CCS codes.




```
# This is a Revit shared parameter file.
# Do not edit manually.
*META   VERSION MINVERSION
META    2       1
*GROUP  ID      NAME
GROUP   3       CCS Common Instance
GROUP   10      CCS Common Type
*PARAM  GUID    NAME    DATATYPE        DATACATEGORY    GROUP   VISIBLE DESCRIPTION     USERMODIFIABLE
PARAM   7edaab02-70c5-4af5-B0ce-373938296 6ef   CCSClassName_Type       TEXT            10      1       Name of the class the object belongs to in CCS classifications. Intended for Family Types.      1
PARAM   3369a102-3619-4bed-b2fb-56ddcb7c3a5b    CCSTypeName     TEXT            3       1       Name of the Type the object belongs to in CCS classification.   1
PARAM   325a5d0b-7591-45cf-a163-761d8f6efbcf    CCSMainTypeName_Type    TEXT            10      1       Name of the main-type specified in the Type_ID of the object. Intended for Family Types.       1
PARAM   4932580c-11a9-4d19-b53e-979a2e90d867    CCSClassName    TEXT            3       1       Name of the class the object belongs to in CCS classification. Intended for Family Instances.   1
PARAM   ff4c1819-e0a0-4fda-8d9f-cb117f3619ce    CCSSubTypeID_Type       TEXT            10      1       Identifies the sub-type of a group of projectspecific objects within the same class. Intended for Family Types. 1
PARAM   66710a1b-8e3d-40e1-8518-ba2f931fe91f    CCSClassCode_Type       TEXT            10      1       Code for the class the object belongs to according to CCS classification. Intended for Family Types.    1
PARAM   11727a25-52ea-4ba9-a8b9-20c86056d108    CCSClassCodeVersion_Type        TEXT            10      1       Internel Table version number from cuneco. Intended for Family Types.   1
PARAM   d0861f43-0387-4980-9bd6-010ab17e2252    CCSTypeID_Type  TEXT            10      1       Identifies a group of projectspecific objects within the same class. Intended for Family Types. (Former CCSFamilyBasedTypeID)   1
PARAM   06ac2c49-ac43-4e1c-8b9f-59917fc3df57    CCSTopnode      TEXT            3       1       Identifies the general class of objects the objects belongs to. Intended for Family Instances.  1
PARAM   b50baa4d-0b6d-41d7-a78e-92c72356c2b5    CCSClassification_Type  TEXT            10      1       Top node and code for the class the object belongs to according to CCS classification. Can also contain the version of the table a
PARAM   38f7334e-deb7-403e-8577-e8b532276c57    CCSSubTypeName_Type     TEXT            10      1       Name of the sub-type specified in the Type_ID of the object. Intended for Family Types. 1
PARAM   b3023554-faad-419a-a109-9082aaac751a    CCSClassCode    TEXT            3       1       Code for the class the object belongs to according to CCS classification. Intended for Family Instances.       1
PARAM   3809e45b-9011-4ec9-b2f0-8418fd39747e    CCSDesignedUseName      TEXT            3       1       Name for the designed use of the object. Intended for Family Instances. 1
PARAM   487b4a6b-a070-4aae-b891-1c9a236dc341    CCSActualUseClassCode   TEXT            3       1       Code indicating the actual use according to CCS Classification. Intended for Family Instances.  1
PARAM   49cfea6b-b5ab-413e-9ee5-bcfb22fa54d7    CCSSubTypeName  TEXT            3       1       Name of the sub-type specified in the Type_ID of the object. Intended for Family Instances.    1
PARAM   16d14b6c-7bc4-4c6b-8d94-9eeb40a2eccf    CCSMultiLevelTypeID     TEXT            3       1       Identifies a group of projectspecific objects within the same class as a part of a group of projetcspecific objects within the sam
PARAM   5dcfe06c-7a1b-4c08-bfae-99fc2cb2c827    CCSMultiLevelLocationID TEXT            3       1       Identifies an object, concatenated located in a space, a storey, a zone and on a construction entity. Intended for Family Instance
PARAM   cf774886d-87de-4360-bfa6-b86a688d3a53   CCSMainTypeID_Type      TEXT            10      1       Identifies the main-type of a group of projectspecific objects within the same class. Intended for Family Types.       1
PARAM   947ad86e-951c-4d71-baf0-96c9b2efbe34    CCSClassCodeVersion     TEXT            3       1       Internel Table version number from cuneco. Intended for Family Instances.       1
PARAM   bf513a74-9a31-48dd-8a56-14dc446c8834    CCSActualUseName        TEXT            3       1       Identifies the general class of objects the objects belongs to. Intended for Family Instances.  1
PARAM   23516c81-5895-4655-860f-69acba9ff534    CCSTypeID       TEXT            3       1       Identifies a group of projectspecific objects within the same class. Intended for Family Instances.    1
PARAM   59eb9f9b-0f1a-4c10-8db2-7e769952f6f2    CCSSingleLevelLocationOnID      TEXT            3       1       Identifies a construction element located on a construction element.    1
PARAM   6867e7a3-11d4-49ee-ba6f-d45022ba1343    CCSSubTypeID    TEXT            3       1       Identifies the sub-type of a group of projectspecific objects within the same class. Intended for Family Instances.    1
PARAM   9f0231b8-74d2-4626-a307-f22e5a37402d    CCSFunctionalID TEXT            3       1       Identifies an object as a part of a whole in a functional context. Intended for Family Instances.       1
PARAM   b6d5eeba-bdcb-430b-8cc6-4235f019ab1d    CCSTypeName_Type        TEXT            10      1       Name of the type specified in the Type_ID of the object. Intended for Family Types.     1
PARAM   8e76a9cc-18df-4d6f-96df-393cac414e47    CCSMainTypeID   TEXT            3       1       Identifies the main-type of a group of projectspecific objects within the same class. Intended for Family Instances.    1
PARAM   ec2394d8-c777-474f-a606-f2e6d732b2b0    CCSSingleLevelLocationAtID      TEXT            3       1       Identifies an Object in a space, a storey, a zone or a construction entity.     1
PARAM   b7995cde-5718-47fe-afa2-c7351eed5cdb    CCSSingleLevelID        TEXT            3       1       Identifies an object percieved as an independent object. Intended for Family Instances. 1
PARAM   24b3cedf-cd54-4c2f-a0af-0f8e5e644832    CCSMultiLevelID TEXT            3       1       Identifies an object percieved as a part of a physical whole. Intended for Family Instances.    1
PARAM   0714dae3-c092-4046-b350-2c19aa8d5efd    CCSTopnode_Type TEXT            10      1       Identifies the general class of objects the objects belongs to. Intended for Family Types.      1
PARAM   e36514e9-bb6e-4ce0-a584-944bf88bfb75    CCSClassification       TEXT            3       1       Top node and code for the class the object belongs to according to CCS classification. Can also contain the version of the table a
PARAM   3ed01de9-e3bd-49be-95b8-407f55306aaf    CCSDesignedUseClassCode TEXT            3       1       Code indicating planned use according to CCS Classification. Intended for Family Instances.       1
PARAM   14c4c0f5-21cf-48ea-a726-280f0b7bbe5e    CCSMainTypeName TEXT            3       1       Name of the main-type specified in the Type_ID of the object. Intended for Family Instances.    1
```





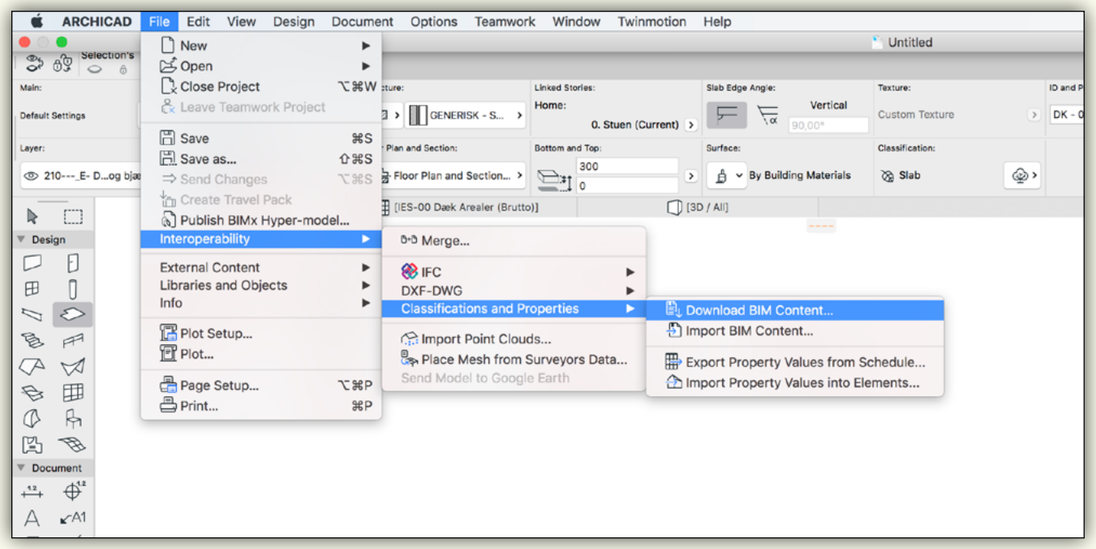
To get an overview of which CCS codes are to be assigned to the individual elements in Revit, you can use CCS Navigate (https://ccs.molio.dk/Navigate/CodeCracker). The above example shows the search for window classification.

CCS classification

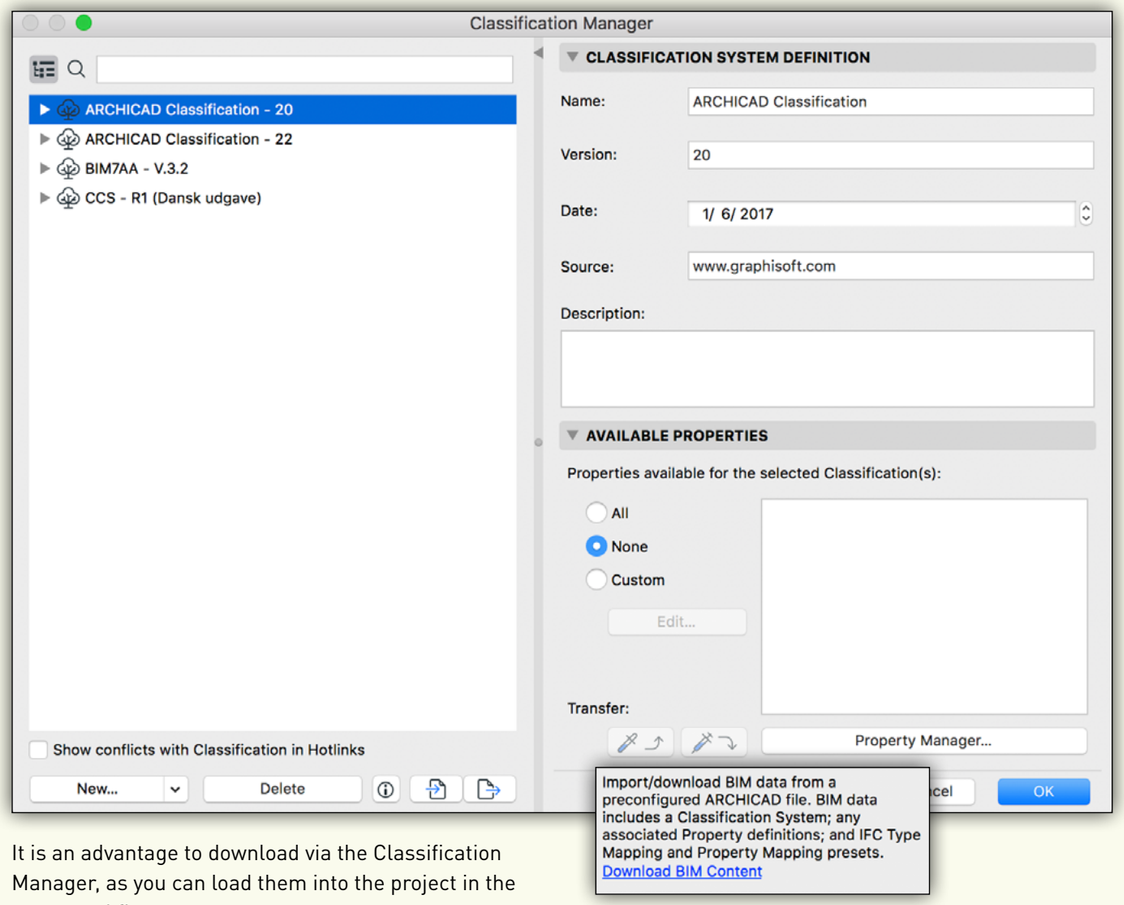
CCSClassCode has the value [CCS] [L] QQA: Window.

CCS identification

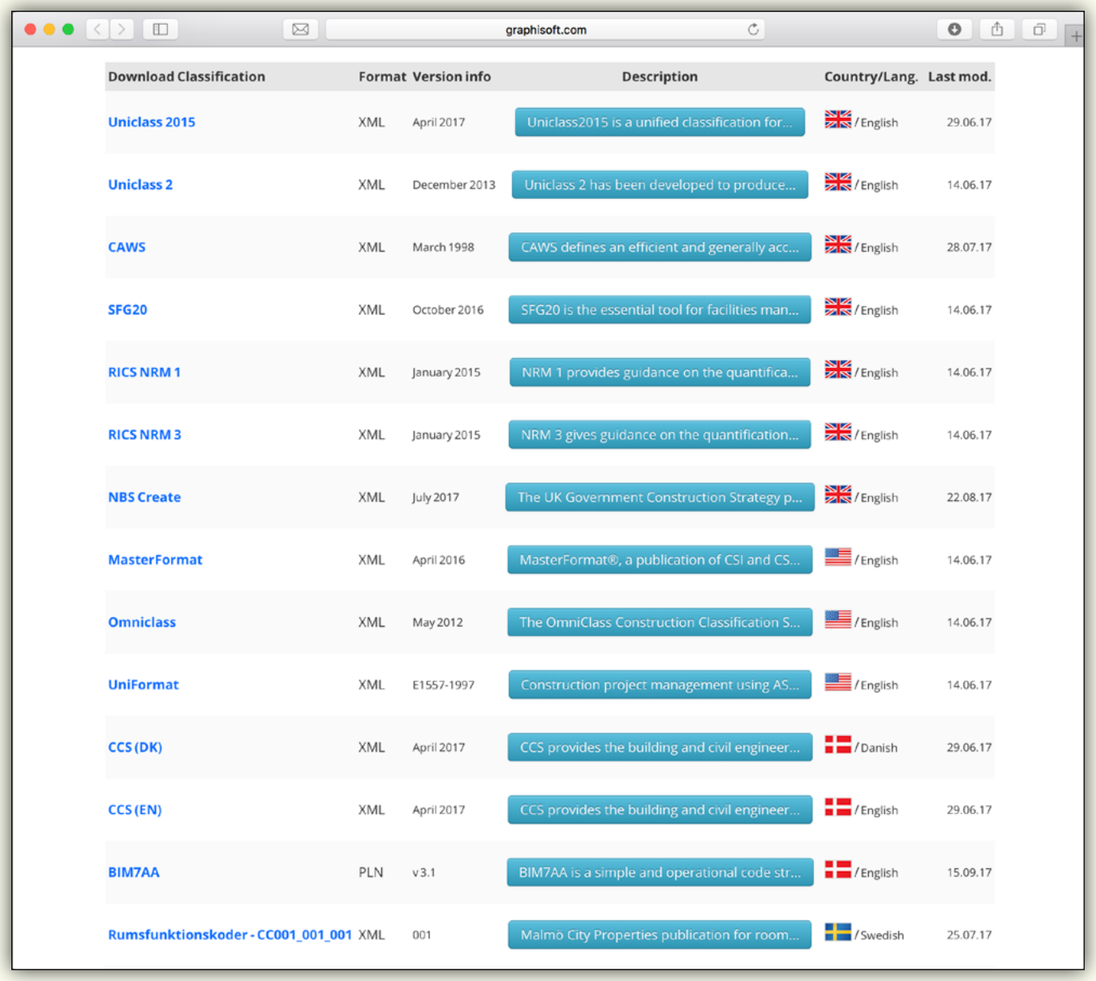
CCSTypeID has the value [L]% QQA90102.01. The parameter value is composed of CCSTopnode ([L]), CCSClassCode (QQA)> with a prefix (%) and serial number (90102.01). The serial number must be unique for the specific type. In the example, Molio has chosen to use a standard division of building part types. 90102 thus refers to an "exterior multidisciplinary window", type 01.



Classification tables for ARCHICAD can be downloaded via "Download BIM Content..." either via "Interoperability" in the File menu or via "Classification Manager" in the Option menu.

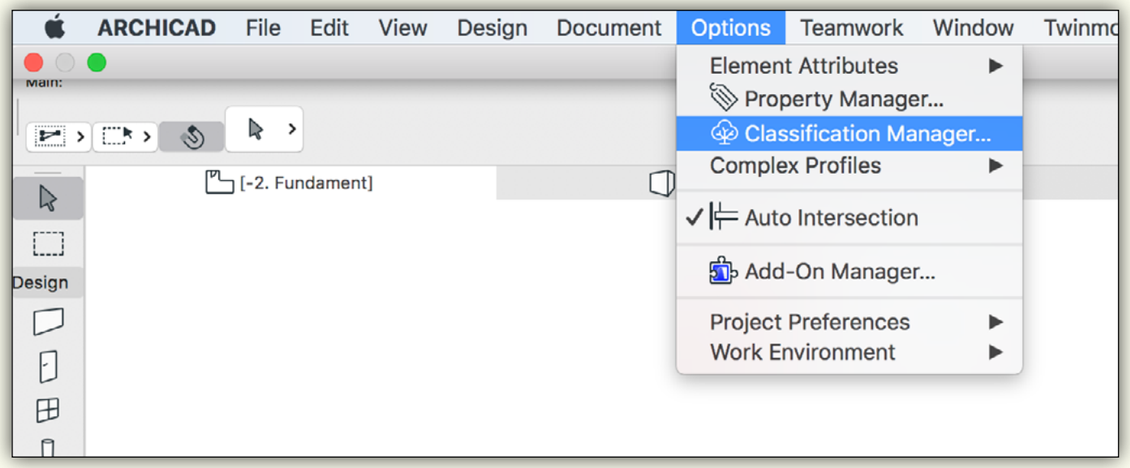


It is an advantage to download via the Classification Manager, as you can load them into the project in the same workflow.

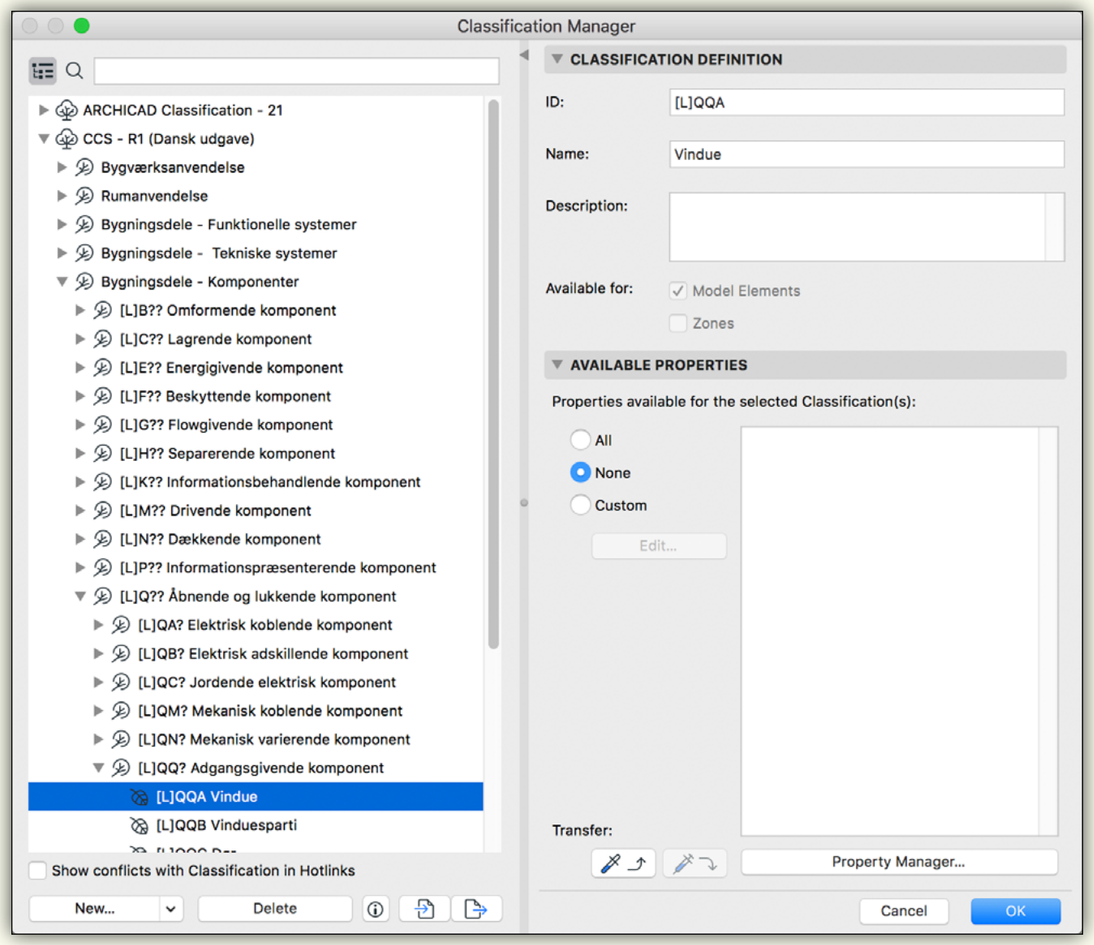


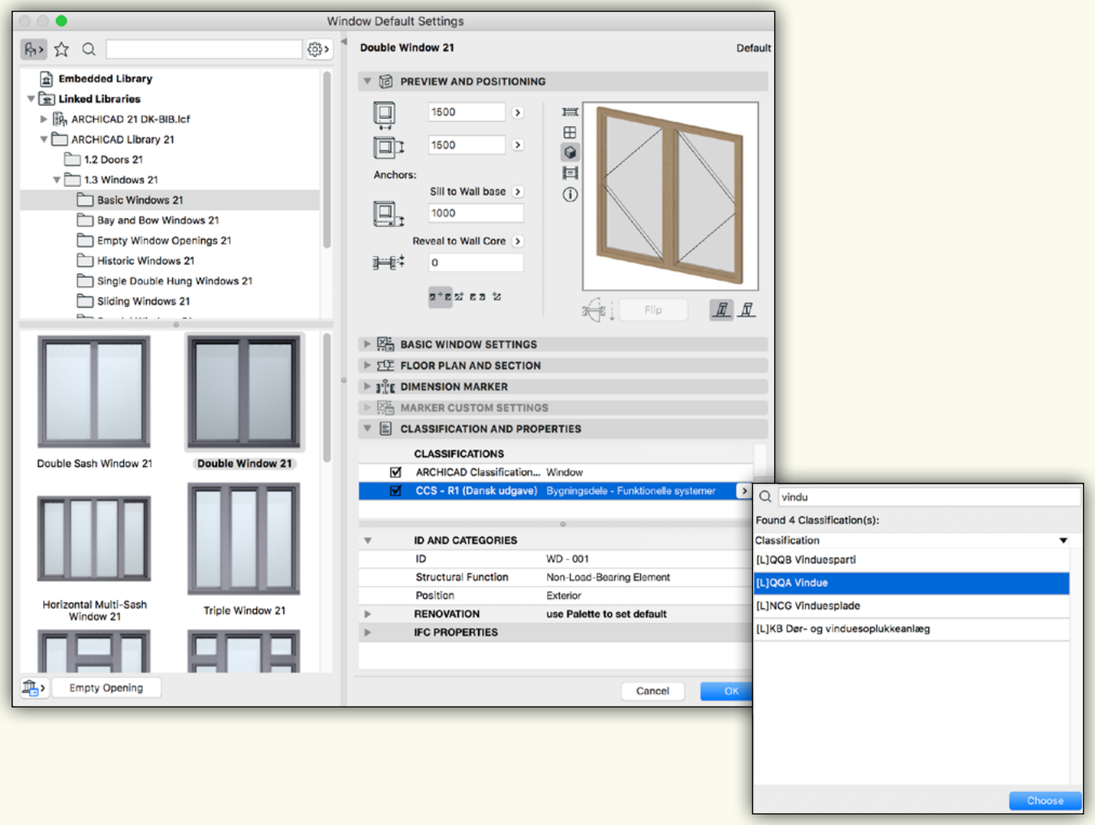
Download Classification	Format	Version info	Description	Country/Lang.	Last mod.
Uniclass 2015	XML	April 2017	Uniclass2015 is a unified classification for...	/ English	29.06.17
Uniclass 2	XML	December 2013	Uniclass 2 has been developed to produce...	/ English	14.06.17
CAWS	XML	March 1998	CAWS defines an efficient and generally acc...	/ English	28.07.17
SFG20	XML	October 2016	SFG20 is the essential tool for facilities man...	/ English	14.06.17
RICS NRM 1	XML	January 2015	NRM 1 provides guidance on the quantifica...	/ English	14.06.17
RICS NRM 3	XML	January 2015	NRM 3 gives guidance on the quantification...	/ English	14.06.17
NBS Create	XML	July 2017	The UK Government Construction Strategy p...	/ English	22.08.17
MasterFormat	XML	April 2016	MasterFormat®, a publication of CSI and CS...	/ English	14.06.17
Omniclass	XML	May 2012	The OmniClass Construction Classification S...	/ English	14.06.17
UniFormat	XML	E1557-1997	Construction project management using AS...	/ English	14.06.17
CCS (DK)	XML	April 2017	CCS provides the building and civil engineer...	/ Danish	29.06.17
CCS (EN)	XML	April 2017	CCS provides the building and civil engineer...	/ English	29.06.17
BIM7AA	PLN	v 3.1	BIM7AA is a simple and operational code str...	/ English	15.09.17
Rumsfunktionskoder - CC001_001_001	XML	001	Malmö City Properties publication for room...	/ Swedish	25.07.17

Download BIM Content leads to the website where classification tables can be downloaded for ARCHICAD. The list is long, but it's easy to get an overview if you look at the countries' flags. So far, there are two Danish systems for download: CCS in Danish and English, as well as BIM7AA.

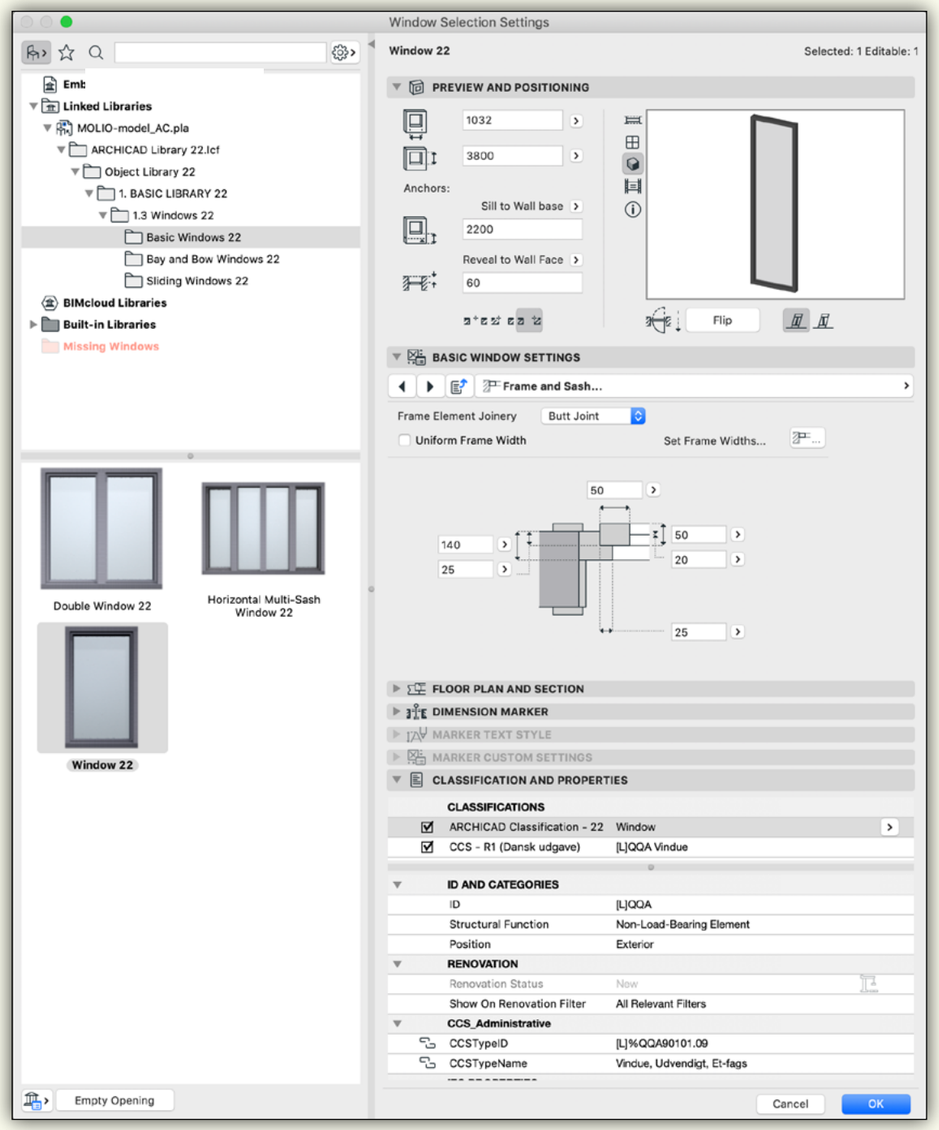


If the tables are downloaded via file / interoperability, or the workplace has the tables located locally, they are imported into the project via the "Classification Manager".





Here is the list of CCS classification codes (which in this example define doors and windows). Each object in the project can be classified with the classification systems that appear in the "Classifications Manager", and you can classify with several different systems on the same object.



Screenshots from the Molio house as it appears in ARCHICAD. To the left is the entire model with a vertical section and a window selected. On the right you will see the settings for the selected window.

CLASSES

Classes are displayed in the IFC file as "Classes" or "Entities". It is possible to assign properties to objects in the building models. The properties overwrite the building part name with which the object was created.

This is especially beneficial if you have modeled a building part using another building part tool. An example might be that you have used the wall tool to make a handrail, after which you overwrite the class to be "Railing" rather than "Wall".

IFC contains more object categories than Revit. In Revit, many objects will be correctly translated into IFC out of the box. For example, walls will be exported as "IfcWall" and floors as "IfcSlab". But especially for "Loadable Families" it may be necessary to specify a translation for IFC's Entities. As a starting point, Revit uses the values from respectively: "IFC Export Mapping" and "IFC Import Mapping". A more detailed mapping it is also possible.

This can be done in three ways:

1. Customizing the IFC Export Mapping table
2. Adding the IfcExportAs parameter and defining its value (recommended)
3. Customizing the IFC Export Mapping table combined with the use of subcategories.

Method 1: Mapping categories and subcategories (SUBCATEGORIES) (Do not do this)

A "Loadable Family" in Revit has a category with any number of subcategories. Subcategories can be used to subdivide an overall Revit (Object Styles) category. Subcategories can be part of a Revit standard setup or defined by the individual user/company.

Revit subcategories can be used for the following purposes:

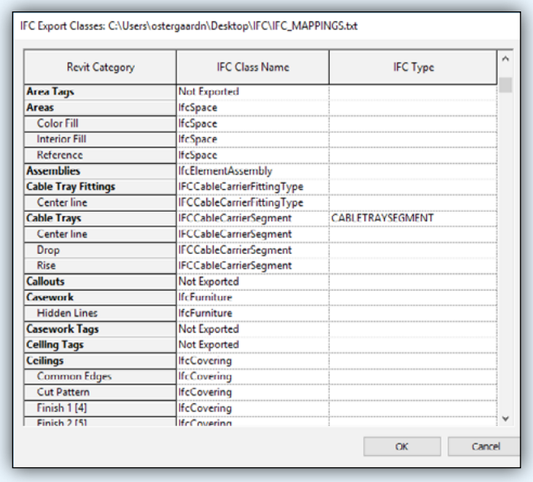
- Configuration of objects' visual properties (or object parts).
- Determining import and export of exchange file types such as .dwf and .ifc in advance. A sub-category must be created when creating a family. By default, Autodesk Revit has some predefined subcategories, but they can also be created by the user.

When loading a family into a project, it is important to customize Revit's IFC mapping tables. Objects in a subcategory will not automatically be exported. This is because new subcategories are set to "Not Exported" and therefore will not be included in an IFC export.

Mapping Table

See Appendix 1.

Revit mapping tables cover only a small percentage of scenarios. For example, mechanical equipment has more categories in IFC than in Revit.

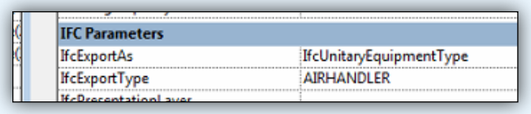

IFC Export Classes: C:\Users\ostergaardn\Desktop\IFC\IFC_MAPPINGS.txt

Revit Category	IFC Class Name	IFC Type
Area Tags	Not Exported	
Areas	IfcSpace	
Color Fill	IfcSpace	
Interior Fill	IfcSpace	
Reference	IfcSpace	
Assemblies	IfcElementAssembly	
Cable Tray Fittings	IFCCableCarrierFittingType	
Center line	IFCCableCarrierFittingType	
Cable Trays	IFCCableCarrierSegment	CABLETRAYSEGMENT
Center line	IFCCableCarrierSegment	
Drop	IFCCableCarrierSegment	
Rise	IFCCableCarrierSegment	
Callouts	Not Exported	
Casework	IfcFurniture	
Hidden Lines	IfcFurniture	
Casework Tags	Not Exported	
Ceiling Tags	Not Exported	
Ceilings	IfcCovering	
Common Edges	IfcCovering	
Cut Pattern	IfcCovering	
Finish 1 [4]	IfcCovering	
Finish 2 [5]	IfcCovering	

OK Cancel

Method 2: Export IFC as (IfcExportAs)

An alternative way to export to IFC is by using the "IfcExportAs" parameter. The parameter will overwrite the association table and can therefore be used alone or in conjunction with Revit's mapping table. The "IfcExportAs" parameter can be used for "System Families" and "Loadable Families".



IFC Parameters	
IfcExportAs	IfcUnitaryEquipmentType
IfcExportType	AIRHANDLER

Method 3: IFC layers

Revit does not use layers for organizing objects. Revit uses the class of the IFC object (Class).

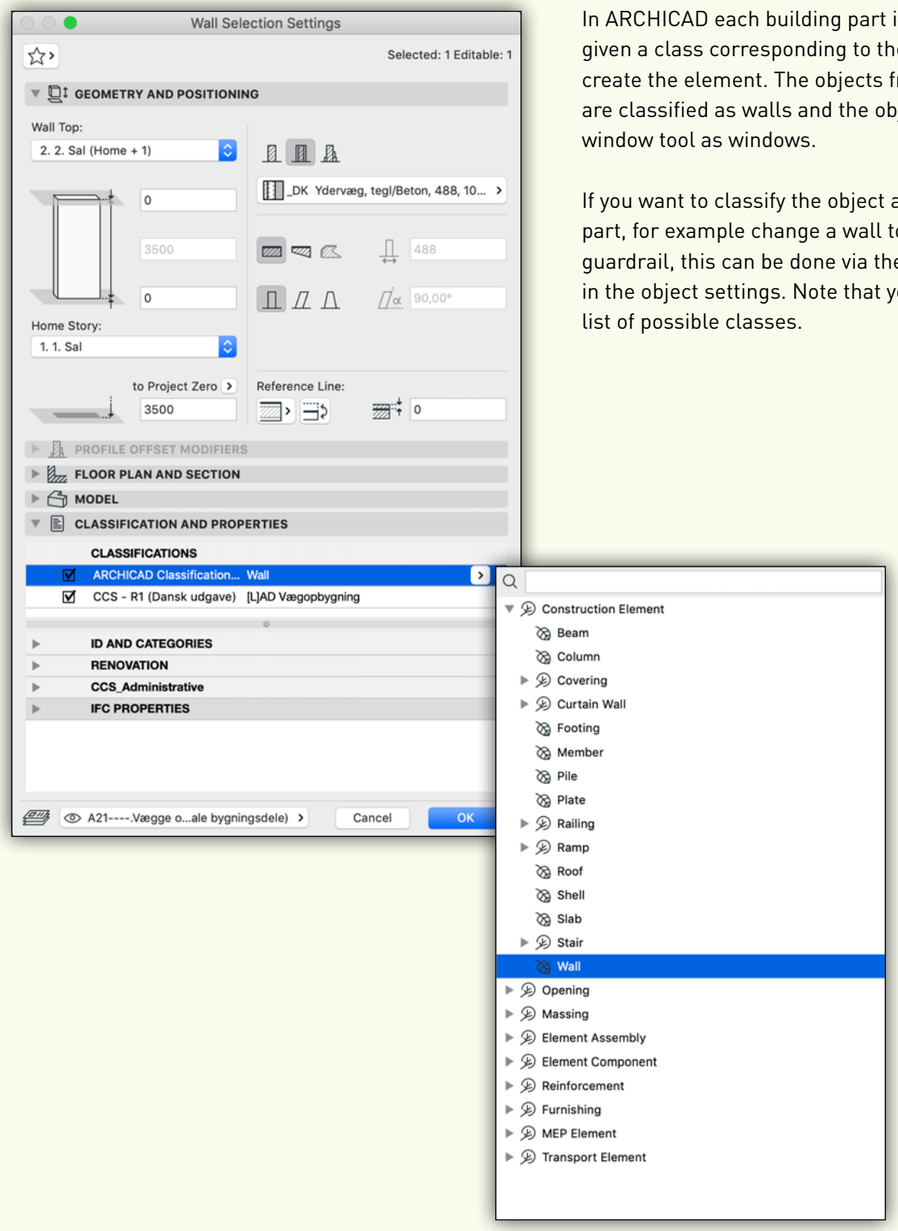
Layers are required for effective collaboration with consultants using tools like ARCHICAD and Tekla.

Layers are required for effective import mapping in accordance with project standards. Revit will associate objects with layers by default. The standard template is adjusted according to BS1192: 1997 and CI / SfB.
Use Uniclass 2015 instead.

Examples of layer mapping:
Curtain Wall
A-214-M_CURT_WALL
A-Ss_25_60_35-M_GlazingSystems
StructuralFrame
S-280-M_STRUCT_FRAME
S-EF_20_10-M_Frames
Duct
M-287-M_DUCT
E-EF_65-M_VentAndAitCon

The mapping table is defined in Revit via a Revit.ini variable.
ExportLayersNameDGN = "P: \ Autodesk \ Revit \ 2017 \ <Company> -exportlayers-dwg-Uniclass2015. txtP:\Autodesk\Revit\2017\ <Company> -exportlayers-dwg-Uniclass2015.txt ".

Note: It is best to use quotes for file paths in case of blank spaces in the file path.



In ARCHICAD each building part is automatically given a class corresponding to the tool used to create the element. The objects from the wall tool are classified as walls and the objects from the window tool as windows.

If you want to classify the object as another building part, for example change a wall to a parapet or guardrail, this can be done via the drop-down menu in the object settings. Note that you can search the list of possible classes.

LAYERS

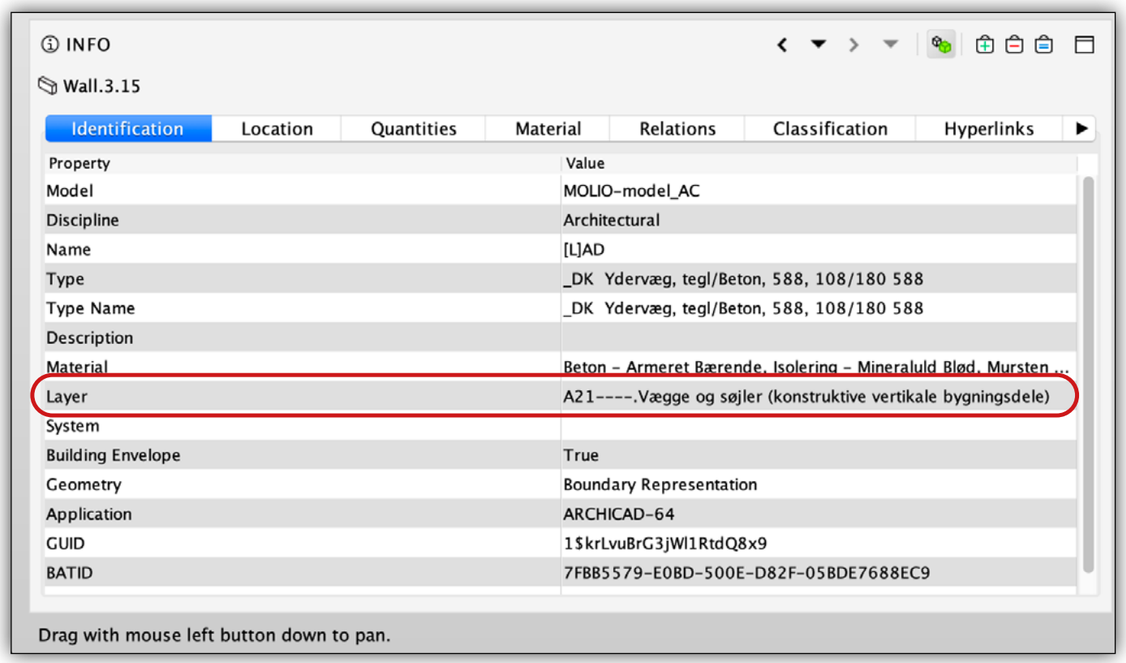
Another opportunity to create structure in the building models is by using layers. Almost all building modeling programs (and many other graphical programs, as well as CAD programs) use layers to structure project content, with the exception of Revit, which structures with other types of property data.

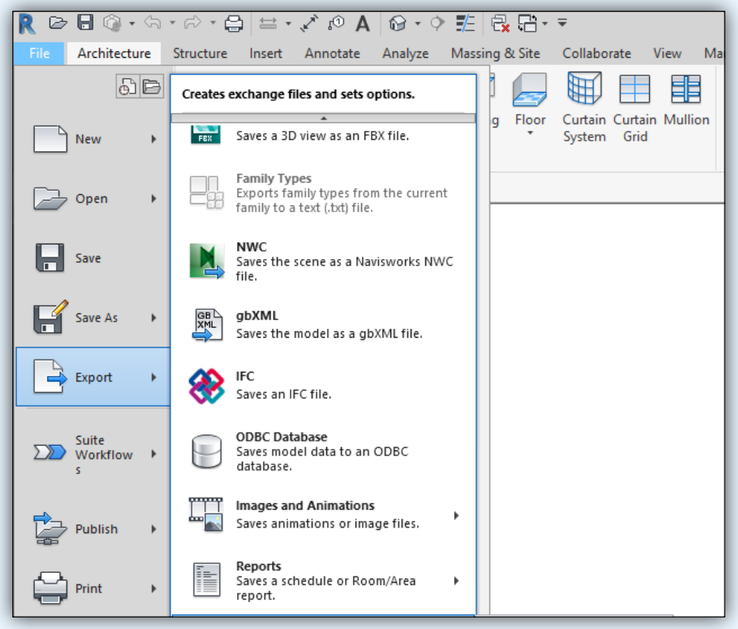
The IFC project can contain information about the layers of the various objects, which is a simple and intuitive way to structure the project if one's building modeling program supports layers. Molio publishes revised versions of their layer structure that can be used to define layer names.

If the content of the IFC file is structured in different layers, it is easy for the recipient to structure, display, search and filter the content. This also makes it possible to separate the imported layers from an IFC file with the layers on which your own professional model is structured.

In Revit, you can automatically create layers in the IFC export based on the classes of the objects (see the Classes section). If one's building modeling program supports layers, the layer name will automatically be exported as part of the object's property data.

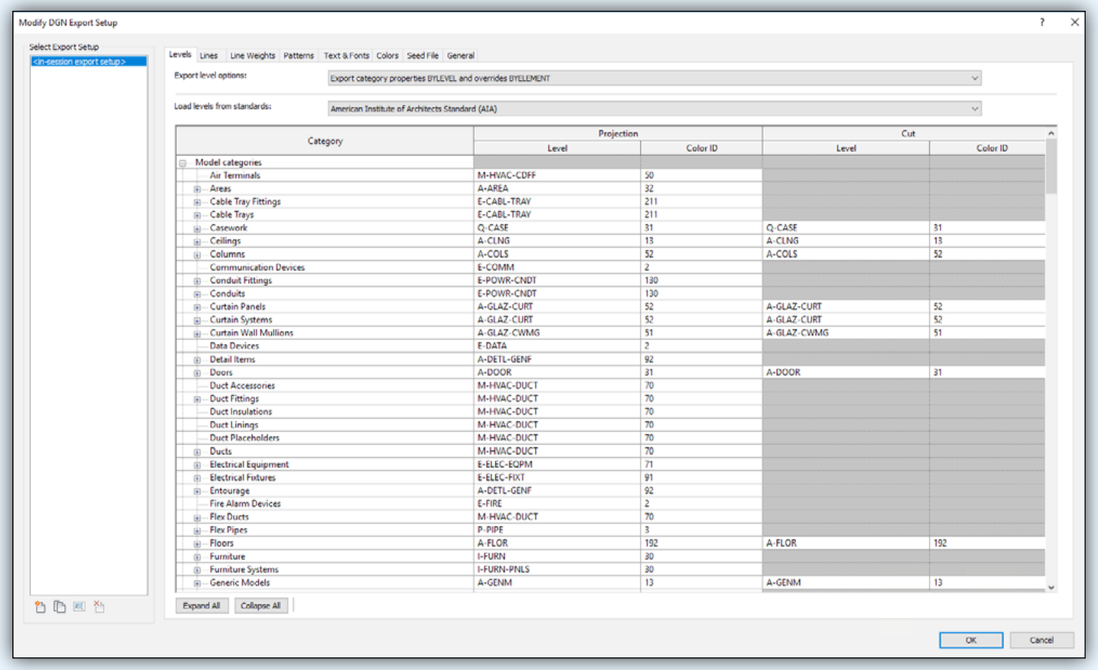
In the illustration below information from a wall object in an IFC file is shown in Solibri Office, where the layer information is assigned according to Molio's Layer Structure 2015:





Layer can be used to secure a "good" import across programs. The setup should be described in the project's ICT process manual. Revit does not use layers for organizing objects. Revit uses the class of the IFC object. When you export to IFC, Revit will use the export layer structure associated with your region by default.

This is done by using the DWG mapping file that comes with Revit. You can change which file is to be used by, for example, correcting your revit.ini file (look for the [Directories] section). Under this section, add or change: ExportLayersNameDGN = "C: \ ProgramData \ Autodesk \ <RVT Version> \ exportlayers-dwg-AIA.txt".



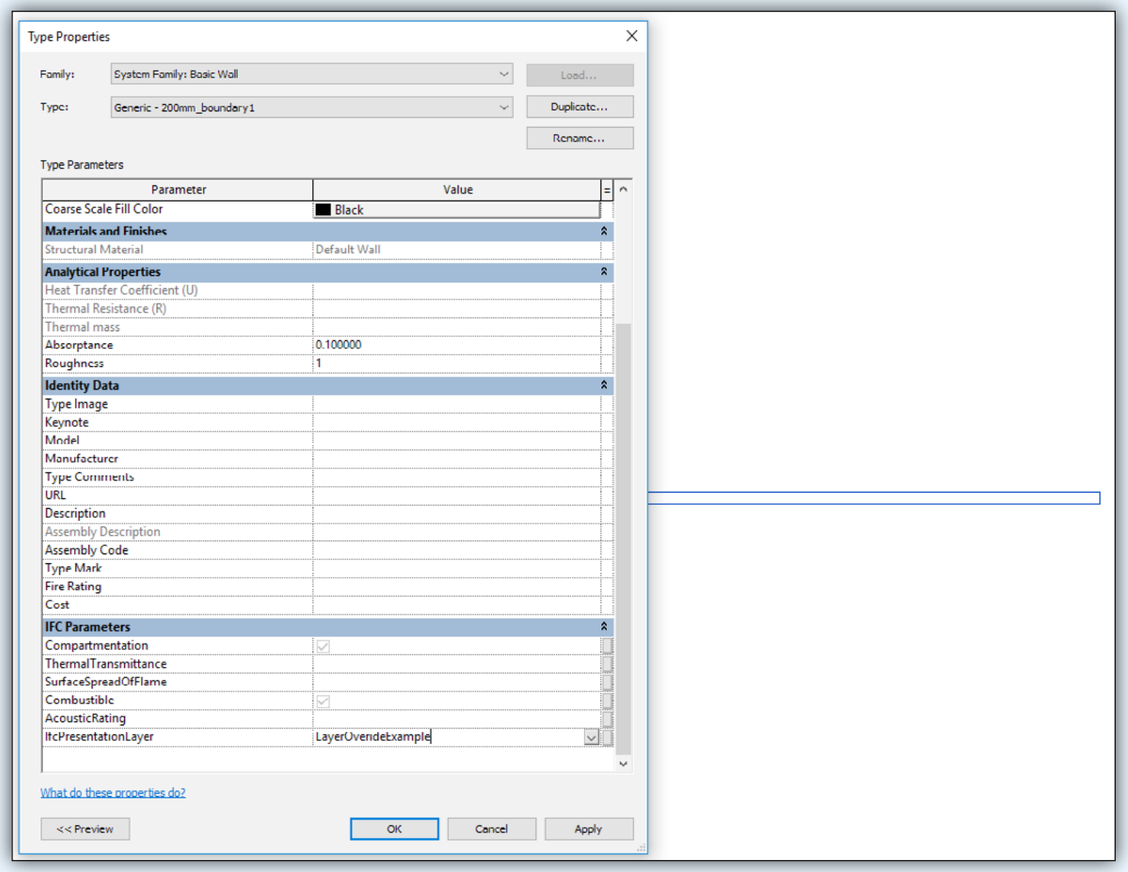
Method 1: Predefined Table

Examples of Layer Mapping:

Walls<tab><tab>A-WALL-____-OTLN<tab>113<tab>A-WALL-____-MCUT<tab>113

Doors<tab><tab>A-DOOR-____-OTLN<tab>31<tab>A-DOOR-____-MCUT<tab>31

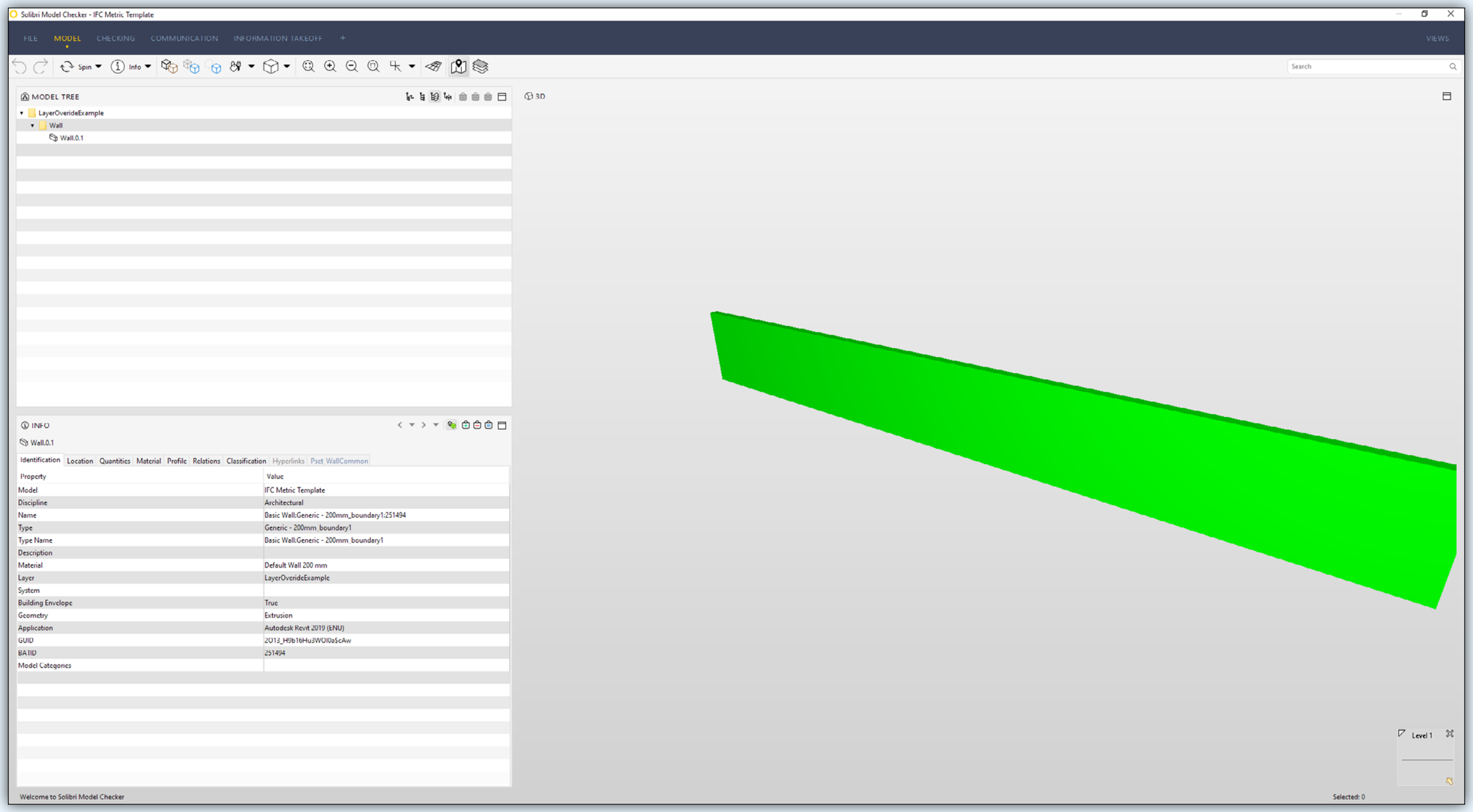
Windows<tab><tab>A-GLAZ-____-OTLN<tab>151<tab>A-GLAZ-____-MCUT<tab>151

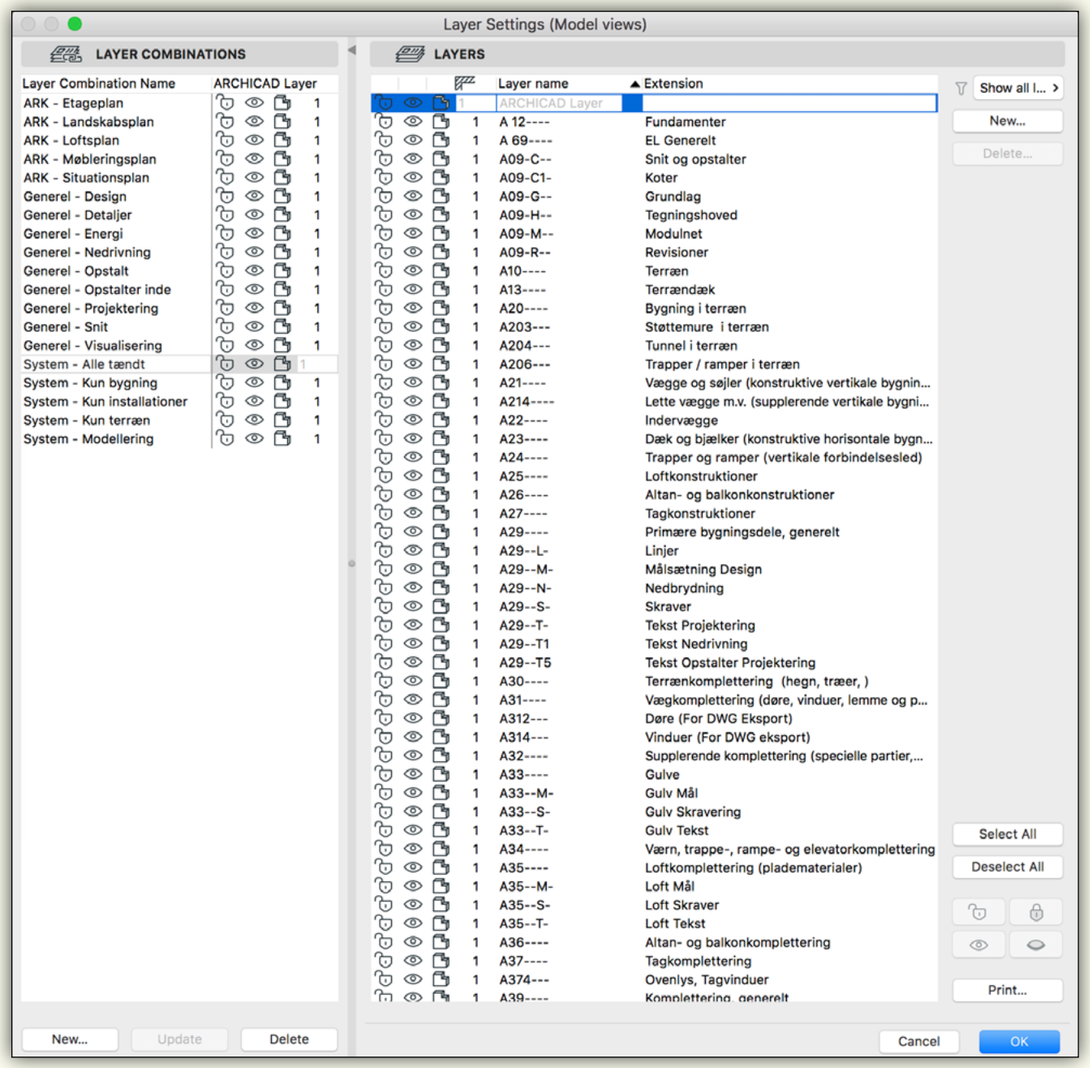


Method 2: Shared Parameters

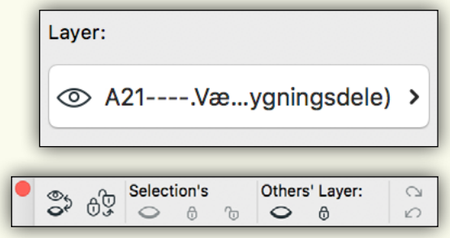
It is possible to create a Shared Parameter named "IfcPresentationLayer" and, by using the value of the parameter, determine the object's layer in the IFC file. This method overwrites all other layer settings (recommended).

Result:


Solibri Model Checker - IFC Metric Template
FILE
MODEL
CHECKING
COMMUNICATION
INFORMATION TAKEOFF
VIEWS
Spin
Info
Search
MODEL TREE
3D
LayerOverideExample
Wall
Wall.0.1
INFO
Wall.0.1
Identification
Location
Quantities
Material
Profile
Relations
Classification
Hyperlinks
Pset_WallCommon
Property
Value
Model
IFC Metric Template
Discipline
Architectural
Name
Basic Wall:Generic - 200mm_boundary1:251494
Type
Generic - 200mm_boundary1
Type Name
Basic Wall:Generic - 200mm_boundary1
Description
Material
Default Wall 200 mm
Layer
LayerOverideExample
System
Building Envelope
True
Geometry
Extrusion
Application
Autodesk Revit 2019 (ENU)
GUID
2O13_H9b16Hu3WOI0a$cAw
BATID
251494
Model Categories
Level 1
Welcome to Solibri Model Checker
Selected: 0



In ARCHICAD, the layer system is used to structure the model content. For example, you can use Molio's Layer structure, which is part of the Danish localization for ARCHICAD.



The left side of the dialog box shows layer combinations that can switch between groups of layers. Each object or group of objects can be assigned a layer.

Layers can easily be turned on and off via the "Quick Layers" palette.

DUPLICATES ACROSS DISCIPLINES

As a starting point, it is ideal that each discipline model contains only those parts of the building for which the specific discipline is responsible.

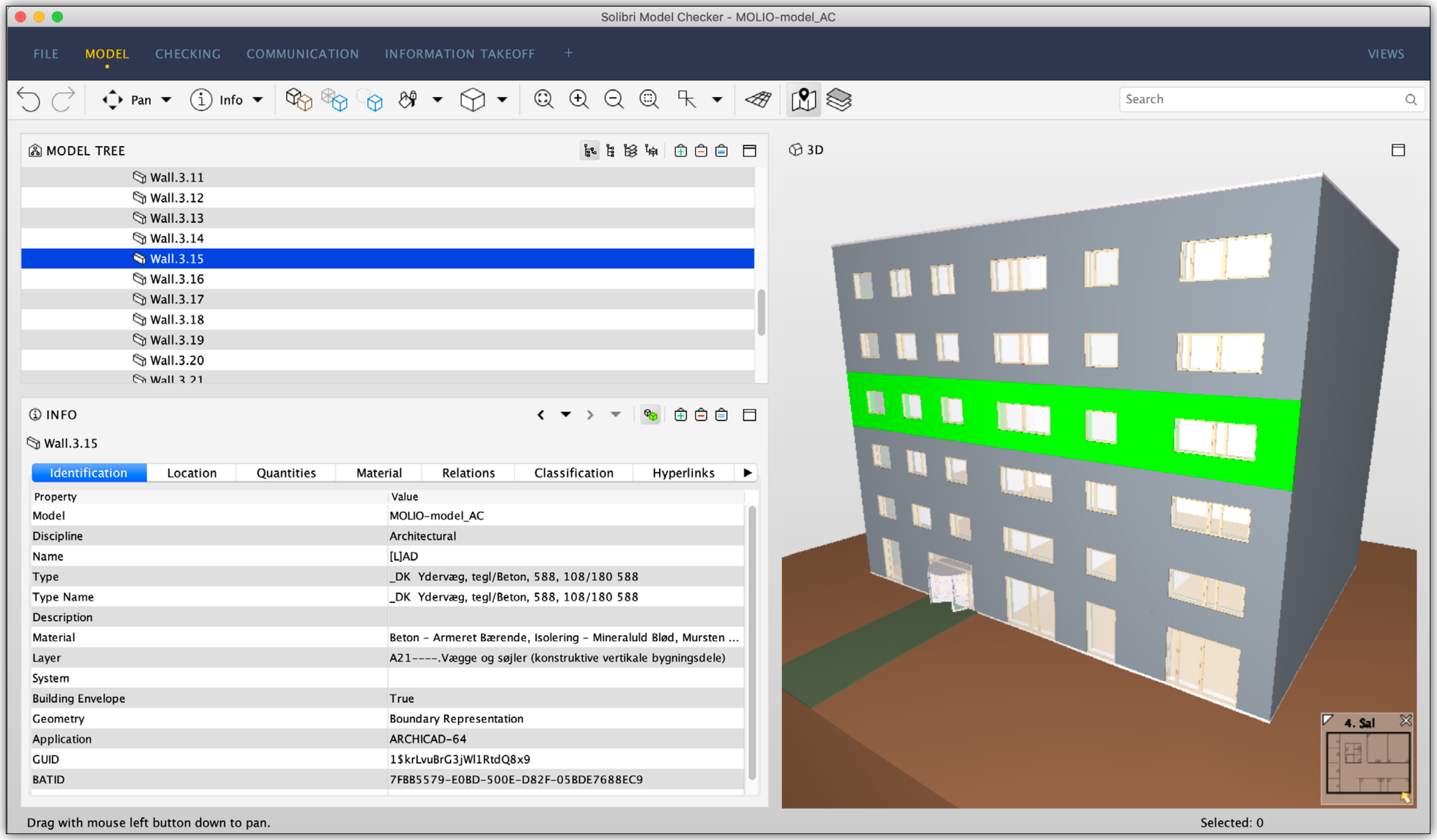
In practice this can be difficult. For example the architect is responsible for insulating and lining the outer walls, while the structural engineer is responsible for the load-bearing part of the walls, the architect is also responsible for the window holes and windows that are fixed in the load-bearing part.

Often, some disciplines will detail parts of their models at a level far too specific to be show in other disciplines models. The architect may want to model single object including the upper floor structure to the lower suspended ceiling, while the structural engineer wants to divide the concrete part of the deck into elements of extruded decks with recesses etc.

Often, each disciplines model will therefore also contain duplicates from other disciplines models, but for the final delivery the federated model should not contain duplicates. See BIM7AA's publications on "BIM Detailing and Responsibility (BDA)" for further information:

http://www.bim7aa.dk/BIM7AA_BDA.html

Software such as Solibri Office can be used to check discipline models, even if they contain duplicates. This is done, for example, by ensuring that two consultants models have concrete wall elements and all window openings are positioned identically.



Screenshot from the Molio house, as the IFC file from ARCHICAD appears in Solibri Office. The green wall is marked in both the content list (MODEL TREE) and the 3D view, and the object's information is shown in the lower left under "Info".

IFC EXPORT

IFC 2x3

The IFC format is constantly updated by buildingSMART. The currently used version is called IFC 2x3, for which many BIM programs are certified for both import and export. See the list of certified BIM programs here:

https://www.buildingsmart.org/compliance/certified-software/

The next version of the IFC format, IFC4, has already been launched by buildingSMART and implemented in several BIM applications. IFC4 handles increased types of data, such as map data in GIS format. However, there are no BIM programs that are certified for IFC4 yet, which means that it is difficult to ensure that the quality of imports and exports is satisfactory. Therefore, it is still recommended to use IFC 2x3 (and not IFC4) for projects where the IFC file has legal significance.

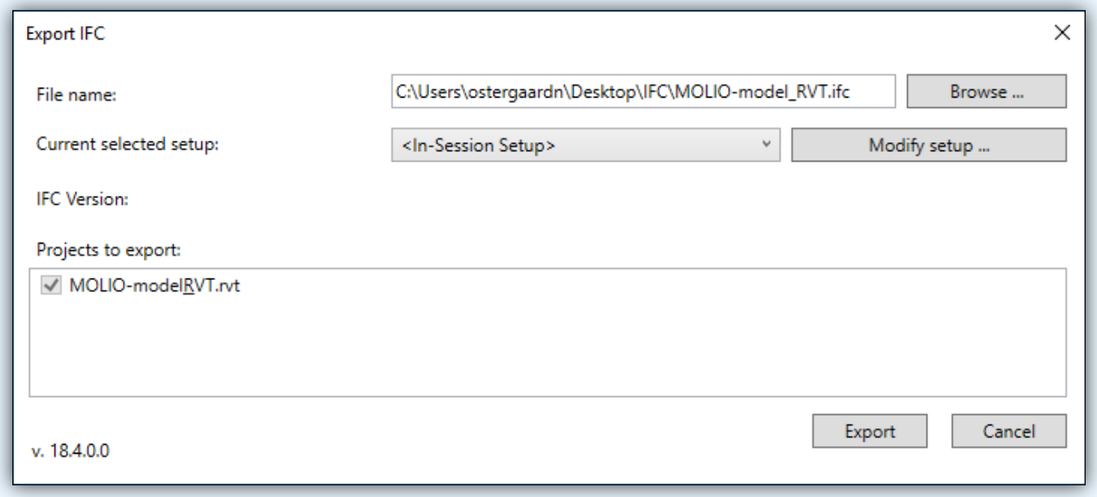
Coordination View 2.0 (MVD)

Since an IFC file can contain a very large amount of data, buildingSMART has defined a number of specific data extracts that make it easy to ensure that the relevant data comes with the IFC file. Coordination View 2.0 is one of the most widely used and has been developed for coordination and quality assurance of building models. Coordination View 2.0 locks the geometry (also called "Boundary Representation" or "Brep") and a wide range of properties are stored at the object level.

In the following, we will use IFC 2x3 Coordination View 2.0 to export the IFC file.

What model parts to export?

You should always only export what is needed, for example, furnishing or terrain will often not be necessary, and would simply make the file slower and unnecessarily complex for the intended use, which could be clash detection or quantity takeoff.



Exporting IFC

To get a proper IFC export, it is often necessary to use Revits Open Source IFC exports. Compared to Revit's built-in "normal" IFC exports, the Open Source IFC exporter offers many more options and greater flexibility. Since Revit 2017, the Open Source exporter has been used as Revit's standard IFC exporter. The Open Source IFC exporter is constantly updated, whereas the version that comes with Revit is "locked" with the release of the specific version of Revit. Therefore, you should remember to update the IFC exporter on an ongoing basis and make sure the ICT specification or ICT process manual describes which specific version the project uses, as functionality and errors are corrected continuously. If this isn't done it could result in two otherwise similar IFC exports looking different.

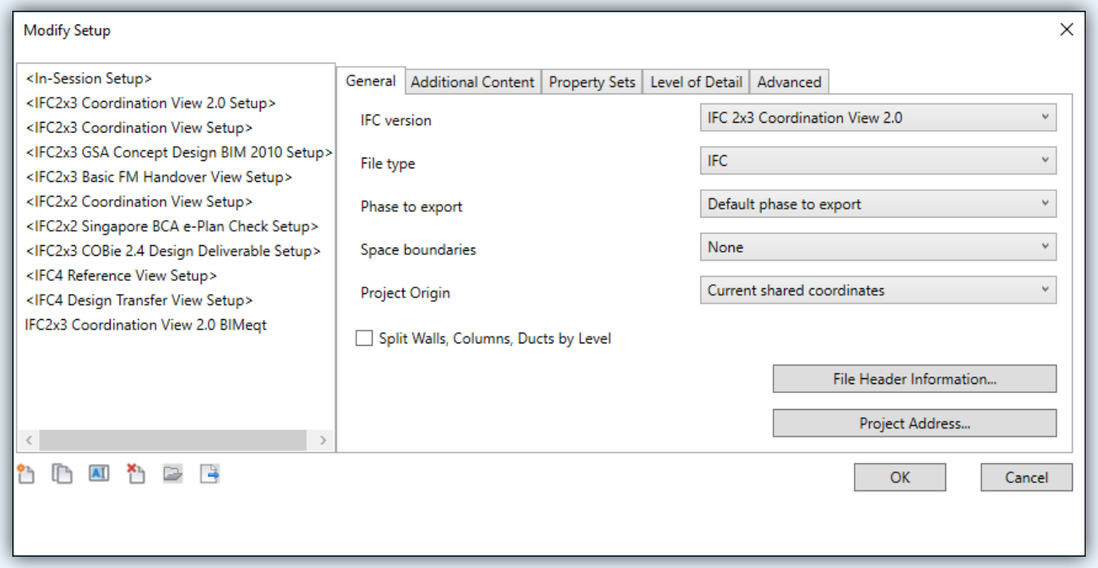
Revits Open Source IFC exports can be downloaded via the "Autodesk Exchange Appstore" (https://apps.autodesk.com).

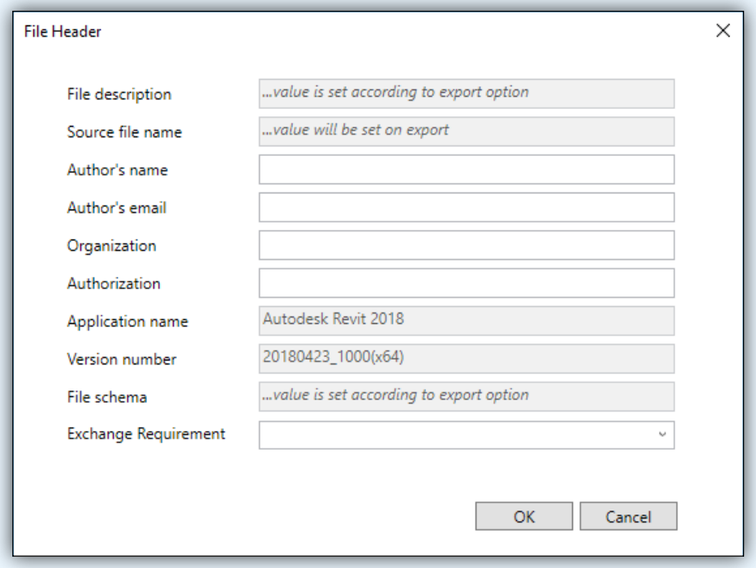
The website https://sourceforge.net/projects/ifcexporter/ contains a user forum including the necessary instructions for proper setup and use of the Open Source IFC exporter. The site is currently moving its code and activities to Github (https://github.com/Autodesk/revit-ifc).

Before exporting to IFC, there are a number of settings that need to be checked to ensure a good and usable IFC file.

The setup can be changed when you click the "Modify setup" button.

The various settings in the main window are grouped into five tabs.





The settings on each of the five tabs are explained in detail below.

General

1. IFC Version: Name of the selected configuration. The most common configuration is "IFC2x3 Coordination View 2.0". The choice should be described in the project's ICT specification and/or ICT process manual.

2. File Type: File type selection. Choose either IFC or IFCzip. IFC is the most common.

3. Phase to export: Select the phase you want to export. By default, this is set to "Default phase to export", which is most often "New Construction". Unfortunately, multiple project phases cannot be exported at once.

4. Space Boundaries: "None" does not export perimeter information. Only the reference room and references to adjacent rooms and components are exported. This setting is often used.

"Level 1" exports spatial boundaries for volume and volume studies, including room/space perimeters. Openings created with the "Edit profile" tool and surfaces edited with the "Split Faces" tool are not included in the export.

"Level 2" exports spatial boundaries along with all data required for energy or thermal calculations. In level 2 the boundaries are affected by the adjacent surfaces and their properties, such as their material properties. For energy calculations, the interfaces are associated with the geometry of the building. If the project uses phases, it is important to ensure that the appropriate phase is selected in the project's energy settings.

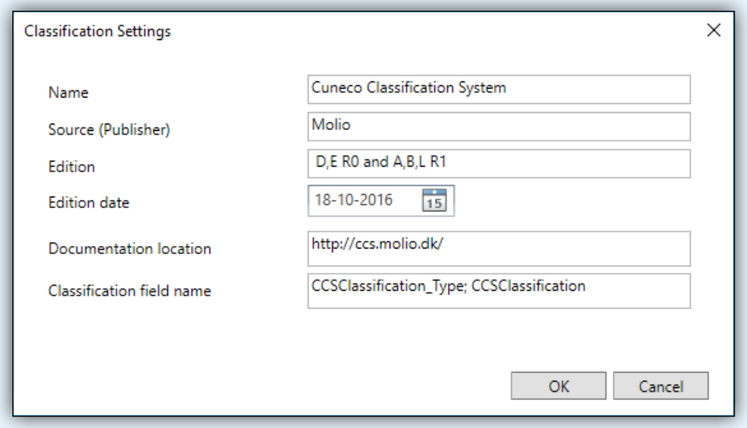
5. Project Origin: Select which of Revit's origin points is used for export. There are four options: "Current shared coordinates" - Origin of current shared coordinates, "Internal Revit coordinates" - Internal Revit origin, "Project Base Point" - Project origin. "Site Survey Point" - World / national surveyors' coordinates.

6. Split Walls, Columns, Ducts by level: Divides the elements by level.

Do not use this feature. All objects should generally be modeled according to how they are to be built or modeled on an appropriate level if desired.

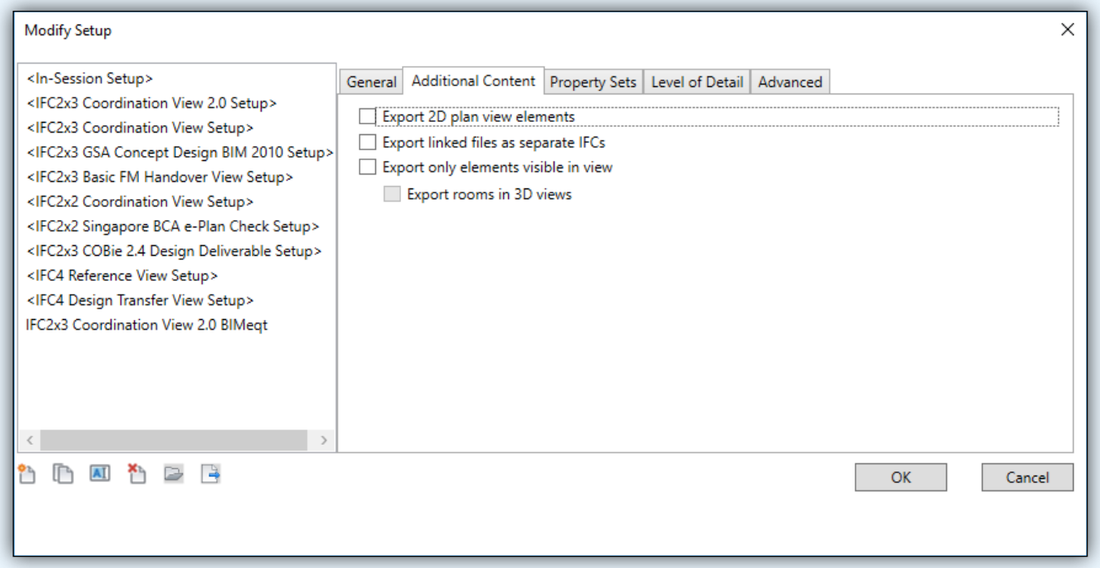
7. File Header Information: Here it is possible to save general file and project information, such as author and company.

8. Project Address: Here you can save information about the project address.



9. Classification Settings: Here the classification system used in the IFC file can be specified. For more information about this setting, see Sourceforge Wiki, Section 6: https://sourceforge.net/p/ifcexporter/wiki/New%20features%20in%20v2.8/

See how CCS is set up correctly in the CLASSIFICATION AND IDENTIFICATION section.



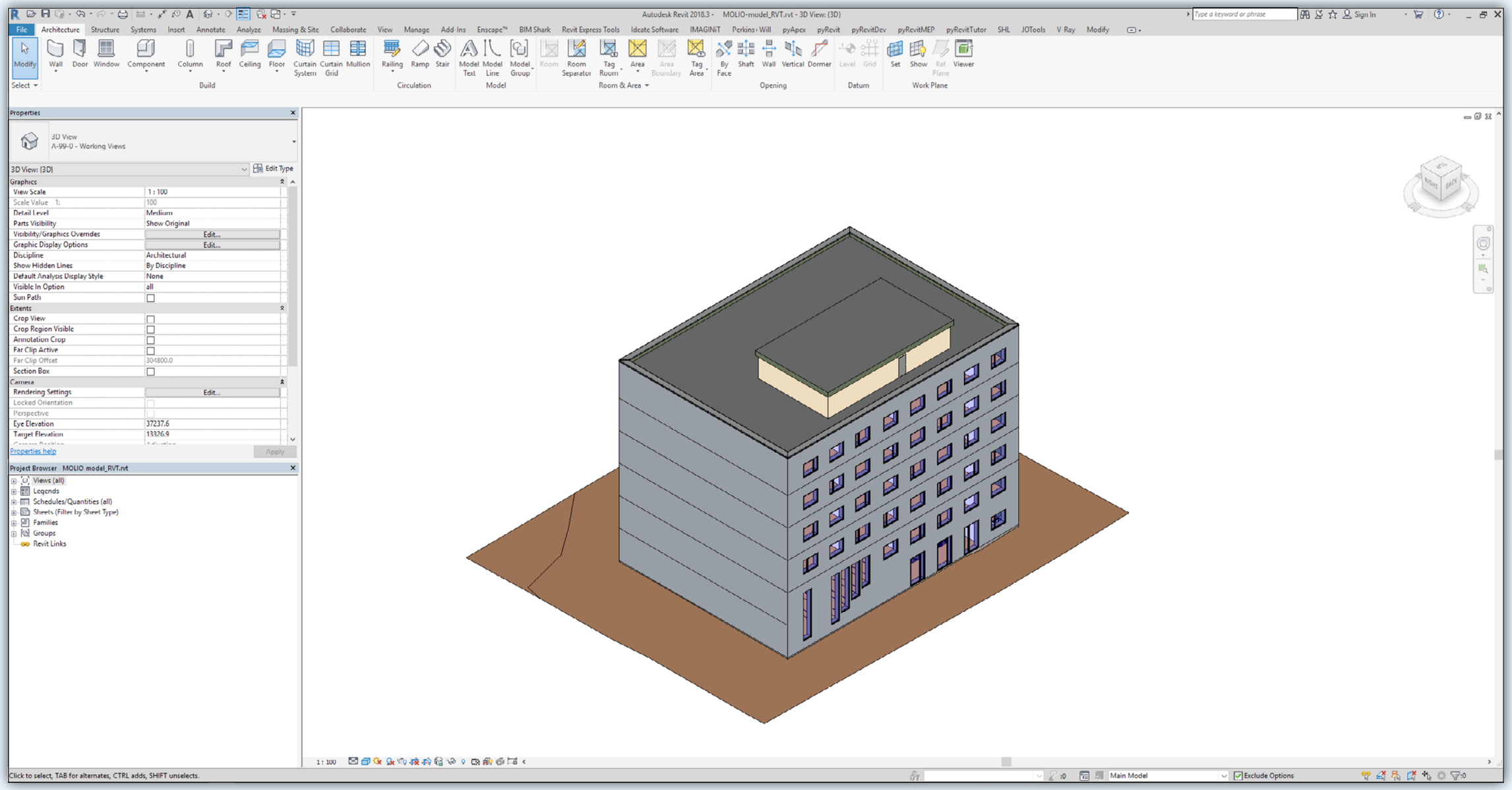
Additional Content

1. Export 2D Plan View Elements: Some 2D symbols and lines can be exported to IFC (module lines, doorway symbols, etc.). Depending on the type of model, it may be advantageous to export the 2D information (e.g., for architects to display opening symbols in the IFC file).

2. Export Linked Files as separate IFCs: Exports files associated with the project as separate IFCs. This feature is very useful when the base points of the various linked files are not coordinated or when a linked file is placed multiple times in the host file. This option will place all links correctly and create an IFC file for all separate linked files (and all instances of a linked file).

3. Export only elements visible in view: Exports only elements visible in the current view. This feature is extremely useful as it gives you 100% control over what is exported.

4. Export Rooms in 3D views: E xports spaces in the IFC file.



Screenshot from the Molio house as it appears in Revit. To the right is the model and to the left are some of the project settings.

As with all other export situations in ARCHICAD, a Layer Combination and a View is created containing the geometry to be exported. In the Danish Localization, this is preset: "General - IFC".

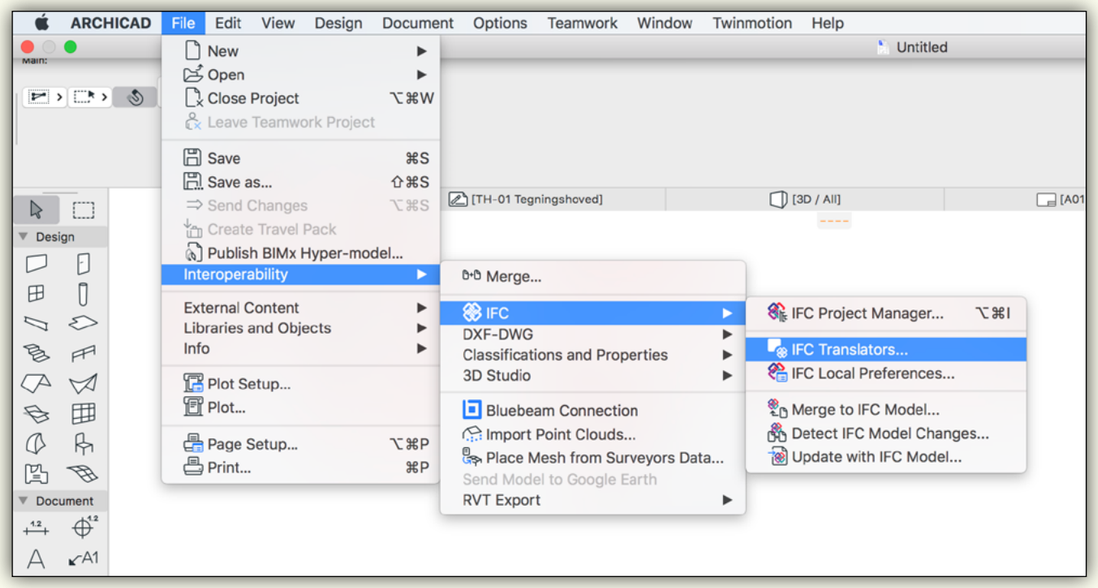
Please note that it is also possible to export 2D elements such as Grids, Lines, Text, Labels, Fills and 2D Views (to plan) of windows and doors. Elements to be exported must be part of the current View.

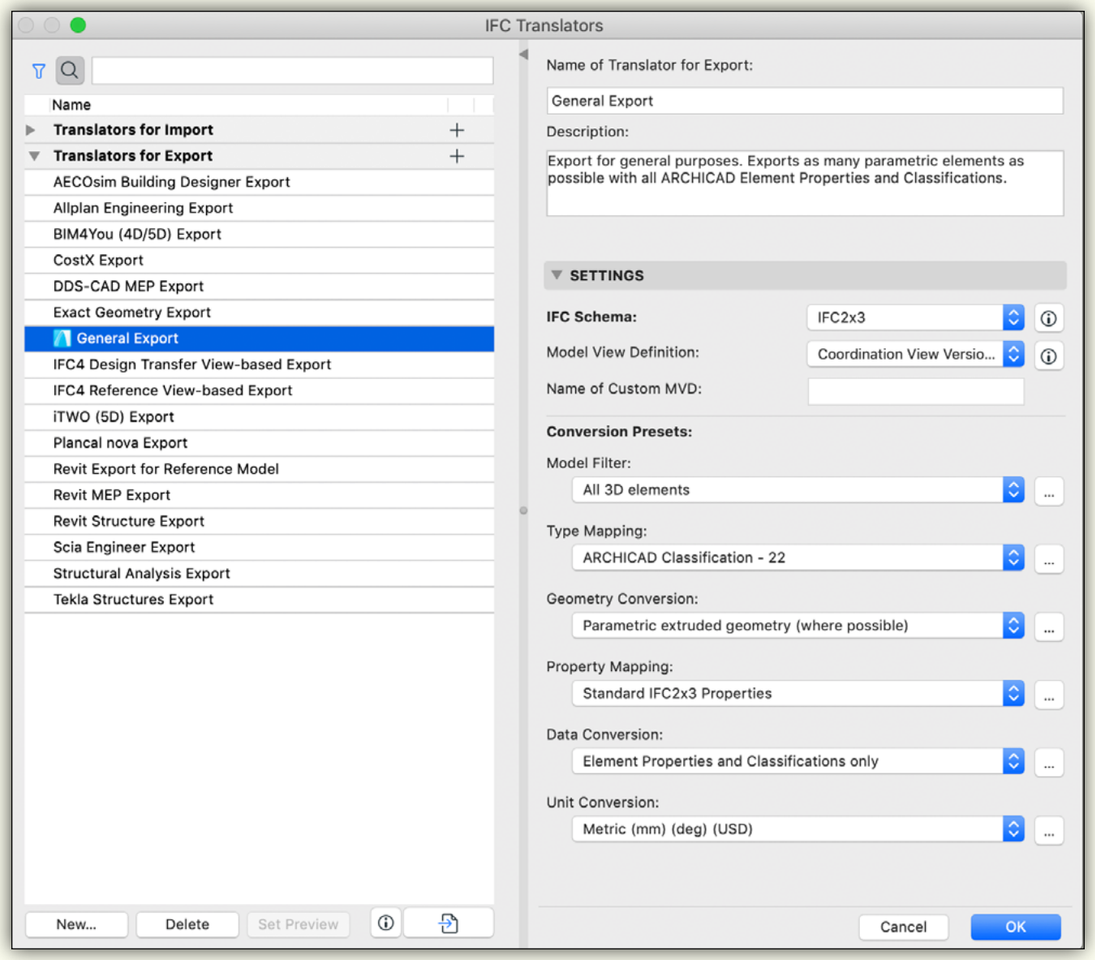
IFC Translators

Together with the correct View, the IFC Translator ensures that there is full control over what and how data is exported and that it is done consistently every time. It is also possible to exchange Translators with consultants or import them from other projects.

Often, the IFC Translator will be part of the company's template. Keep in mind that it is easier to keep track of your configuration if it is the View that controls what is exported while the IFC Translator controls how. Conversly if geometry is turned off in the IFC Translator, but visible in the view, it can be difficult to discover why it is not exported.

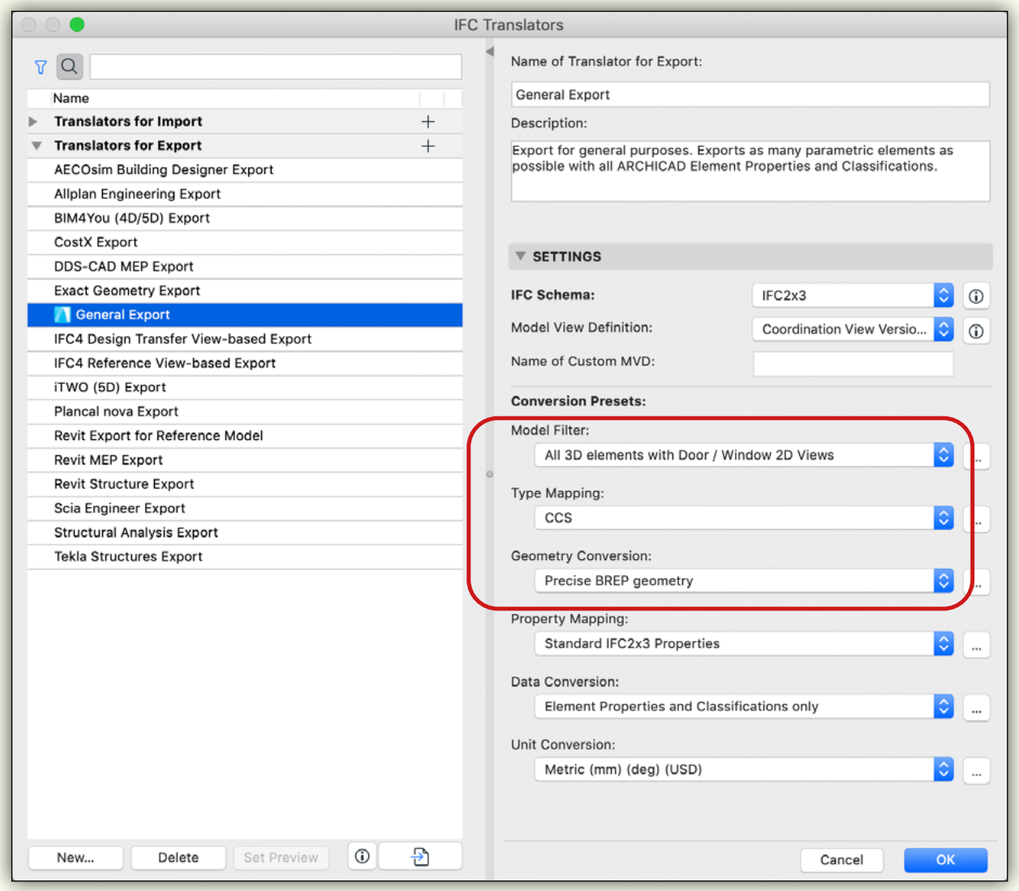
Select the IFC Translator under the menu:
File / Interoperability / IFC / IFC Translators:





On the left side of the IFC dialog box you will see various Translators for import and export, and on the right side you will find the settings. Translators are named accordingly to their intended use. There are Translators for export to Autodesk Revit MEP / Structure, Trimble Tekla, RIB iTWO and others.

When you select a Translator, you will see all the settings on the right. The Translators are preconfigured, so in most cases "out of the box" export will work well.



Translator: General Export
This Translator is set up to export IFC 2x3 in Coordination View 2.0.

To get a precise IFC export regardless of geometry, the settings must be changed so that "Geometry Convertion" is set to "Precise BREP Geometry" so that, for example, Trim, Solid Element Operations and Edge settings of, for example, floor slabs, also are updated in the IFC.

A number of other settings are also available:

"Model Filter" is set to "All 3D Elements with Door / Window 2D Views".
"Type Mapping" is set to "CCS".

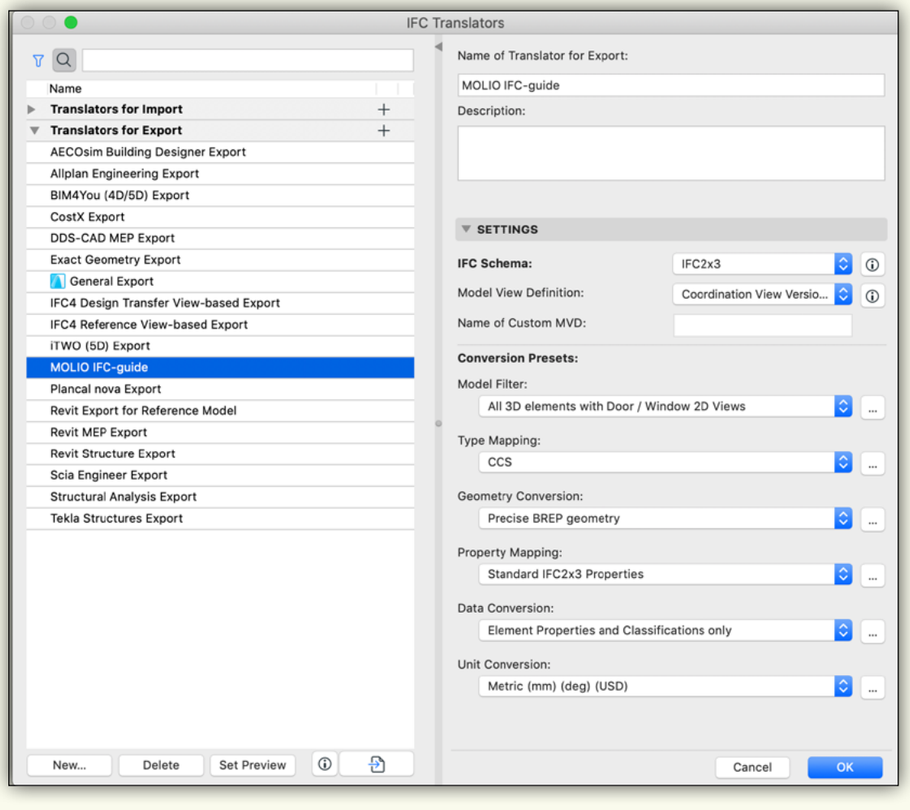
Specific Translators for IFC Export:

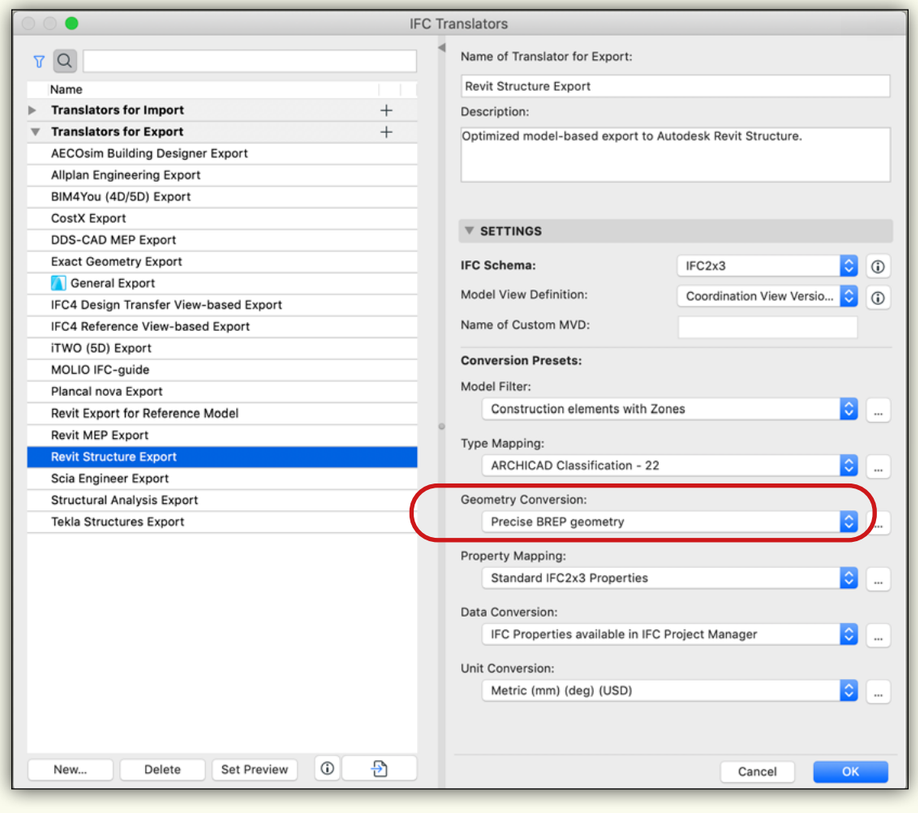
Revit Export to Reference Model
This Translator has been set up to use the model as a reference. Included are grid lines and 2D representations of windows and doors, but without any data.

Revit MEP Export
This Translator is set for subsequent import into Revit MEP, and exports 3D geometry and Grid lines.

Revit Structure Export
This Translator is set to only export items that are defined as load bearing. In addition, the 3D geometry is set to appear as "Extruded", which means that, for example, Trim and Solid Element Operations are not included. If you want a more precise geometric representation, you have to change the geometry conversion setting:

Select Brep as shown below:





ARCHICAD IFC import in Revit

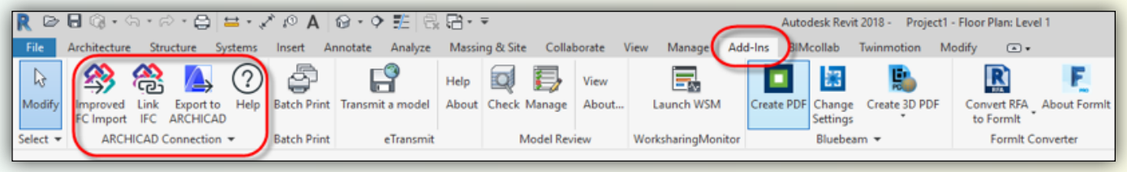
To optimize the import of ARCHICAD's IFC models in Revit, the Revit user should install the "GRAPHISOFT Improved IFC Import" add-in to Revit. It improves Revit's import of the IFC model – both when importing and also when linking the IFC file. In addition, exports of the Revit model to IFC are improved for ARCHICAD imports.

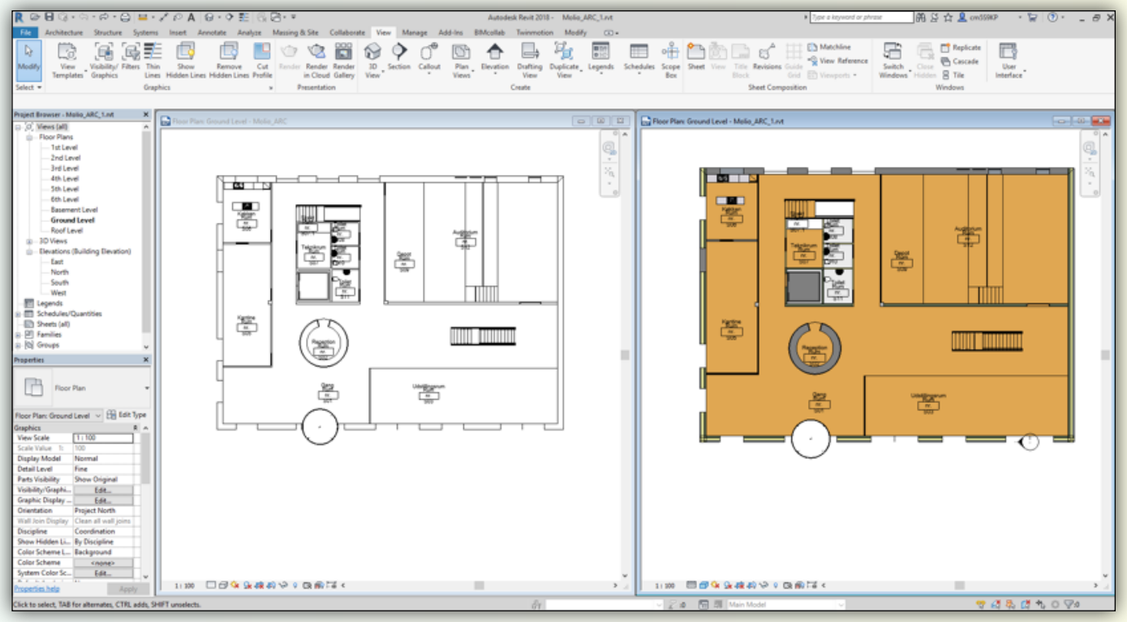
Notify your Revit partners that they should use ARCHICAD's add-in for import and export:

IFC Model Exchange with ARCHICAD for Revit 2019:
http://www.graphisoft.com/downloads/interoperability.html

NOTE! The "GRAPHISOFT Improved IFC Import" add-in to Revit has a conflict with MagiCAD, which means Revit will be incredibly slow if both programs / add-ins are installed. Therefore, always check if this will be the case before installation.

The image at the bottom right shows the import with Revit Open / IFC in the left window and with GRAPHISOFT's Improved IFC import in the right window.





EXPORTING PROPERTY DATA

Object Properties

A building part's characteristics are often described in the BIM world as a "Properties", where a Property Set (Pset) is a collection of properties, typically within a specific domain, such as product information (manufacturer name, date of production, country of production, etc.).

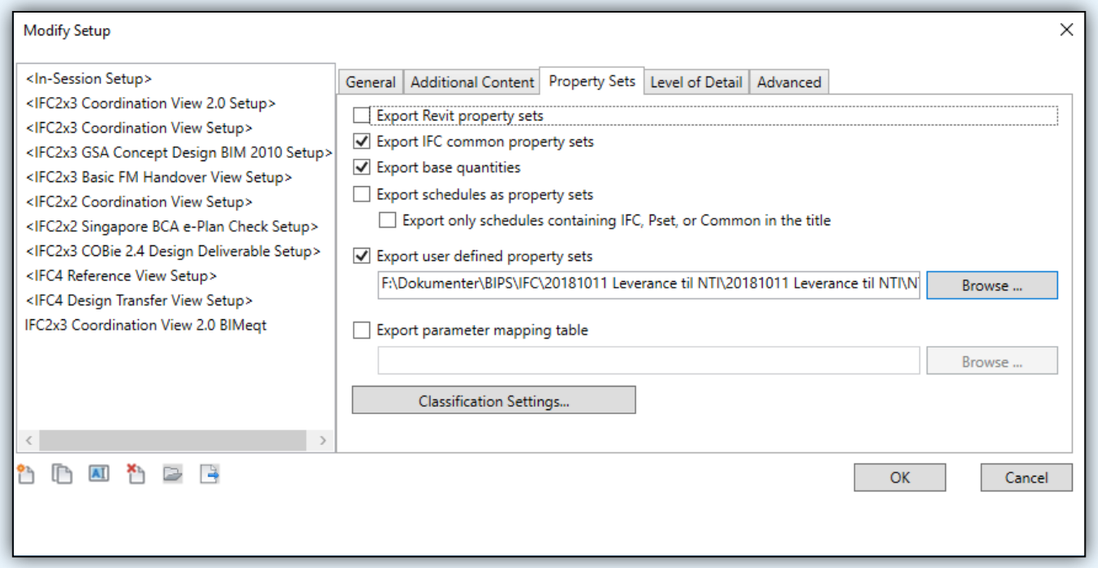
All objects in the building model may have properties assigned. Some properties define the geometry of the object and are also called parameters (e.g., height and width), while others are data-only associated with the object (e.g., U-value).

Molio and others are currently working on identifying which properties add value in relation to different project stages and levels of detail. At the same time, they are working on defining a method so that one can uniformly define sizes (e.g. opening dimensions or product dimensions), colors (e.g. RAL, NCS or Pantone), fire class (e.g. EI 30 A2-s1, d0) and units (e.g. meters or millimeters)).

Property Sets

If it is agreed in the ICT specification that additional properties must be provided on some or all objects, it will be practical to provide a list of these properties as a "Property Set" ("Pset") that can be imported into the building modeling program. This ensures that all data fields are automatically created with fewer chances for error as a result. This is called a "Shared Parameter" file in Revit or a "Property Manager" file in ARCHICAD.

```
#
# User Defined PropertySet Definition File
#
# Format:
#    PropertySet:       <Pset Name>     I[nstance]/T[ype]       <element list separated by ','>
#       <Property Name 1>       <Data type>     <[opt] Revit parameter name, if different from IFC>
#       <Property Name 2>       <Data type>     <[opt] Revit parameter name, if different from IFC>
#       ...
#
# Data types supported: Area, Boolean, ClassificationReference, ColorTemperature, Count, Currency,
#       ElectricalCurrent, ElectricalEfficacy, ElectricalVoltage, Force, Frequency, Identifier,
#       Illuminance, Integer, Label, Length, Logical, LuminousFlux, LuminousIntensity,
#       NormalisedRatio, MassDensity, PlaneAngle, PositiveLength, PositivePlaneAngle, PositiveRatio,
#       Power, Pressure, Ratio, Real, Text, ThermalTransmittance, ThermodynamicTemperature, Volume,
#       VolumetricFlowRate
#
# Example property set definition for the NLRSv2.5.2:
#
PropertySet:    NLRSv252_Test   I       IfcElement
        IFC_ContentCreator      Real    Length
```



Property Sets

1. Export Revit property sets: Exports all Revit properties to the IFC file. This feature should not be used for "phase delivery" as the IFC file will most likely contain both non-validated and validated data, without the recipient being able to determine what is what. Instead, use "Export User Defined Property Sets" to select which properties are exported.

2. Export IFC common property sets: IFC contains a set of "default properties" for each object type. If this feature is enabled, the properties will be exported / created.

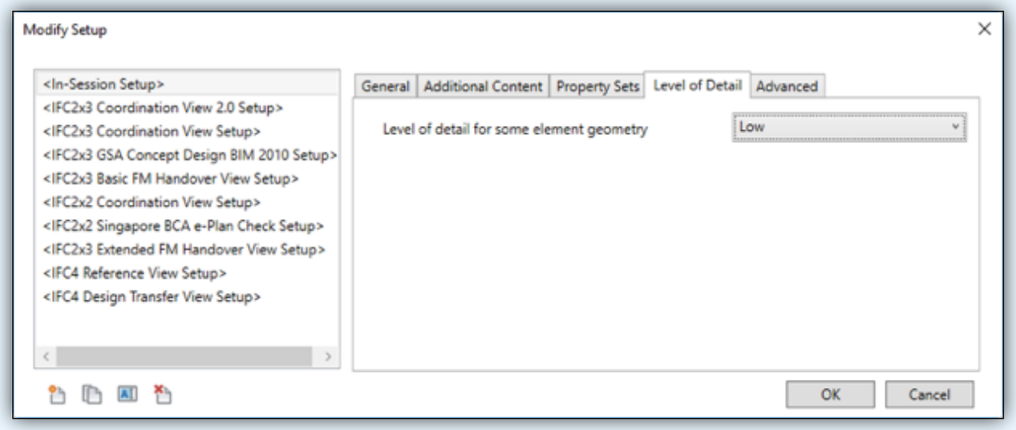
3. Export base quantities: Calculates the IFC quantities based on Revit's geometry. The values are transferred as "locked" values in the IFC file. (Recommended)

4. Export schedules as property sets: All Revit schedules will be converted and exported to custom IFC Property sets. See Appendix 2.

5. Export only schedules containing IFC, Pset or Common in the title: This will only export the schemas to IFC custom property sets when the Revit schemas contain "IFC", "Pset" or "Common" in the title. Should only be used if this is a conscious choice and configured correctly in the project. See Appendix 2.

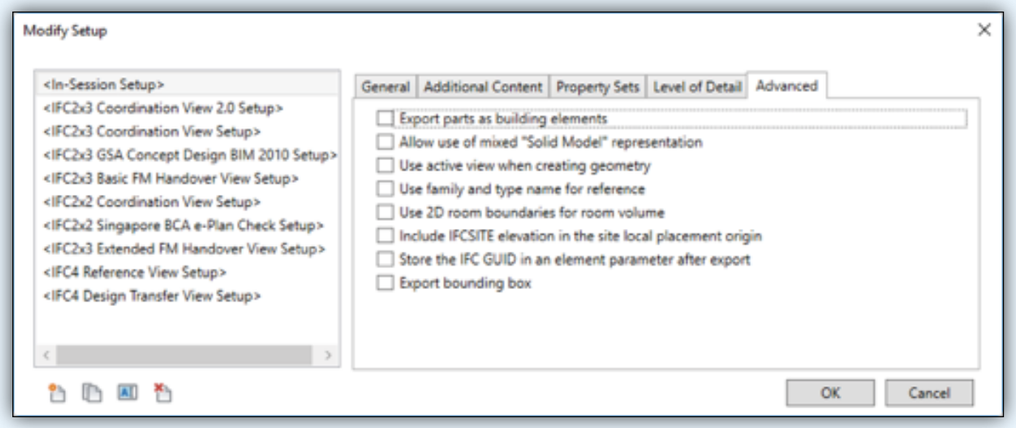
6. Export User Defined Property Sets: Exports Revit parameters in a custom property set. This is the preferred method of exporting custom properties to IFC. See Appendix 3.

7. Export parameter mapping table: Exports Revit parameters to IfcCommon Propertysets. This is not recommended.



Level of Detail

1. Level of detail for some element geometry: Reduces the amount of meshes when exporting to IFC using "BoundaryRepresentation". Set this to "Low" to reduce the IFC file size.



Advanced

1. Export parts as building elements: Exports Revit's "parts" as separate objects.

2. Allow use of mixed "Solid Model" representation: Several solids created per object in case of complex geometry. This results in lighter IFC export for complex geometry.

3. Use active view when generating geometry: Use the current view to create geometric representation of the objects in the IFC. This setting is recommended.

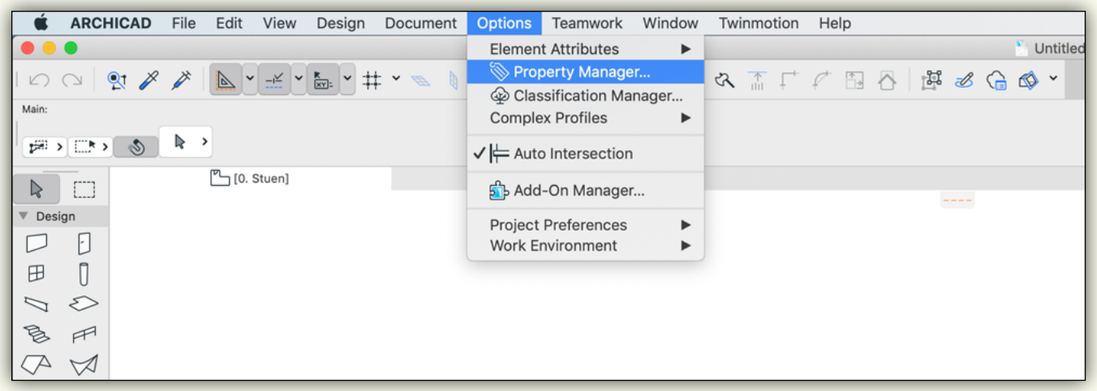
4. Use family and type name for reference: The family name and type name are used as "Reference" in IfcCommon Property Sets.

5. Use 2D room boundaries for room volume: Exports simplified space geometry (based on 2D plane instead of the actual 3D shape). If only the area of the room is needed, this feature may be good, otherwise it should not be used.

6. Include IFCSITE elevation in the site local placement origin: Exports the height of IfcSite's local placement. It should be agreed in the project's ICT specification or ICT process manual whether this feature is used or not.

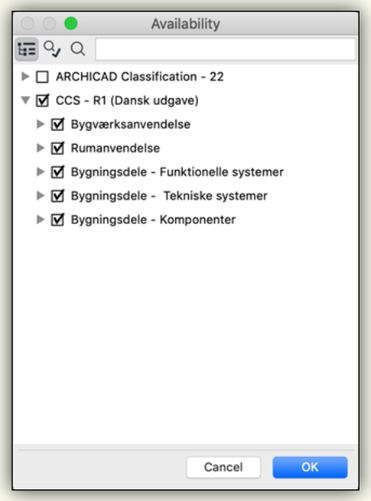
7. Store the IFC GUID in an element parameter after export: Save the IFC GUID for each object. This allows for for checking whether all items were exported to IFC. It also ensures that the same IFC GUID will be used for the next export. The people behind this guide have in some cases experienced failures with this feature.

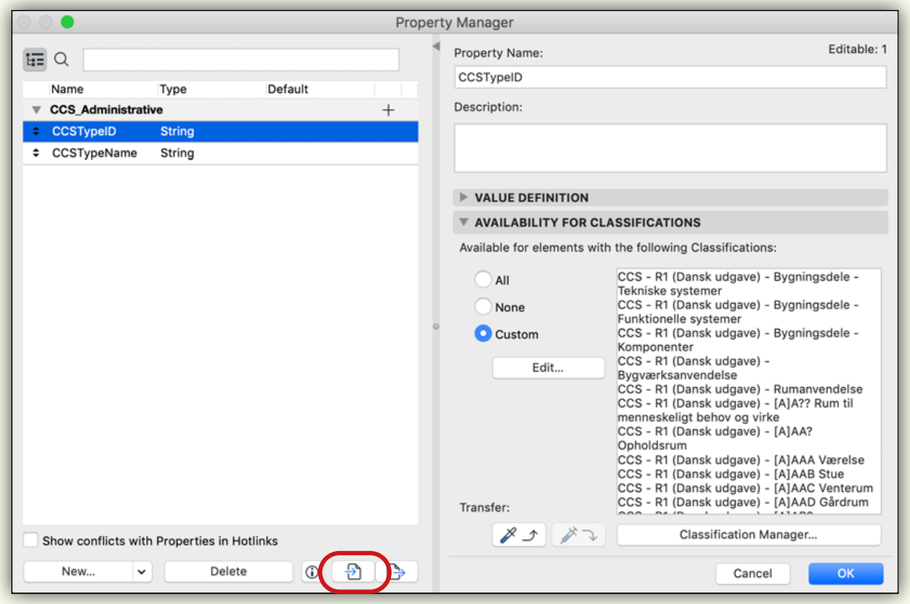
8. Export bounding box: Create a virtual IFC element that represents the smallest right-angled box the object can be in.

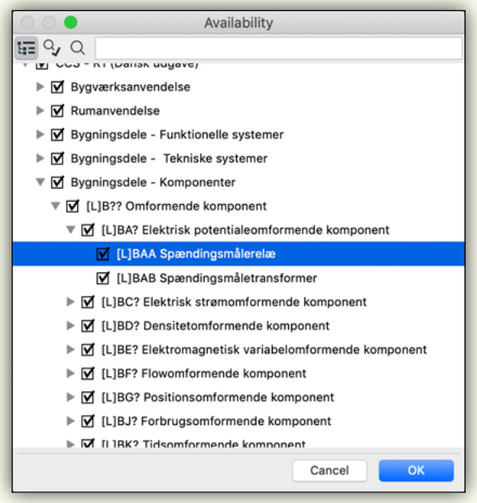


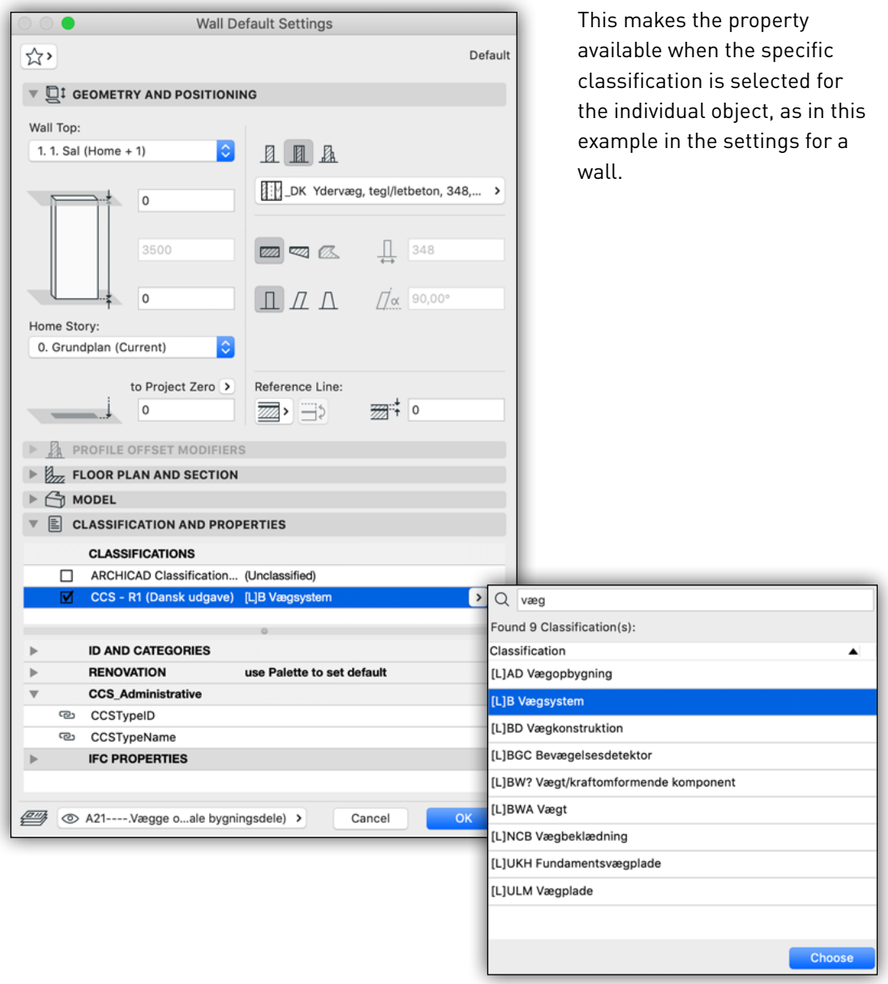
In the Property Manager, preset properties can be loaded, and new properties can be created to add to the model. By starting a new group, you can structure your data into relevant "PropertySet".

Each property can be associated with a specific classification. In the "Edit Availability" for "Classifications" checkbox select the current classification. Thus, a given property must be available when classifying with "CCS Building Components - Components". Or even more specifically CCS Building Components, [L] BA? Electrical potential converting components, [L] BAA Volta









This makes the property available when the specific classification is selected for the individual object, as in this example in the settings for a wall.

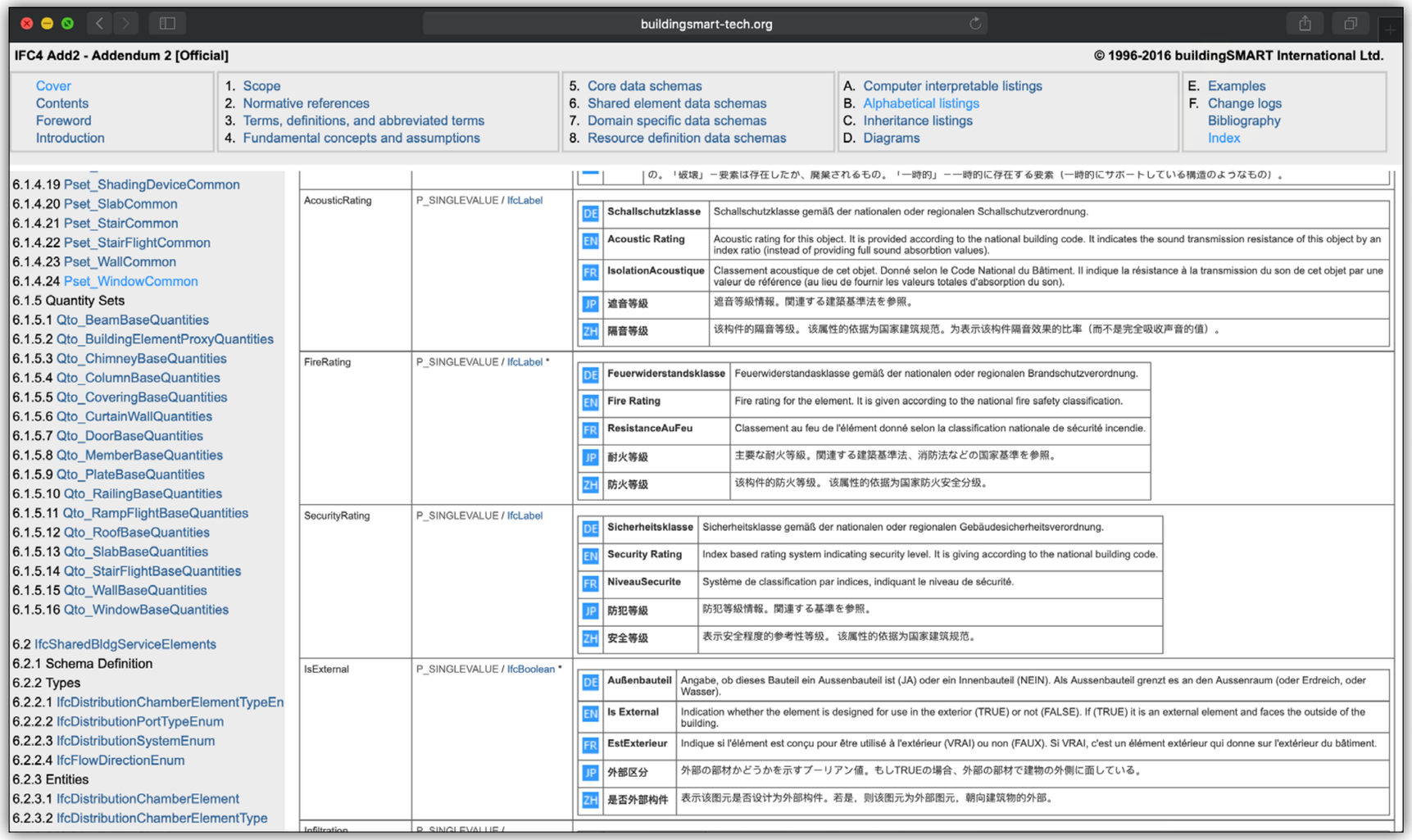
IFC STANDARD PROPERTIES

In addition to classification systems, the BIM objects can be assigned several properties that further describe the objects and their function. This could for example be:

- Structural building parts - whether a building part is part of the structural system
- Internal or external building part - whether a building part is part of the building envelope (exterior / interior)
- Renovation status / Phases
- Fire rating / class

These characteristics are universal across disciplines, software platforms, and projects.

IFC properties for all IFC objects can be found on the buildingSMART technical website: http://www.buildingsmart-tech.org/ifc/IFC4/Add2/html/



buildingsmart-tech.org

IFC4 Add2 - Addendum 2 [Official] © 1996-2016 buildingSMART International Ltd.

Cover
Contents
Foreword
Introduction

1. Scope
2. Normative references
3. Terms, definitions, and abbreviated terms
4. Fundamental concepts and assumptions

5. Core data schemas
6. Shared element data schemas
7. Domain specific data schemas
8. Resource definition data schemas

A. Computer interpretable listings
B. Alphabetical listings
C. Inheritance listings
D. Diagrams

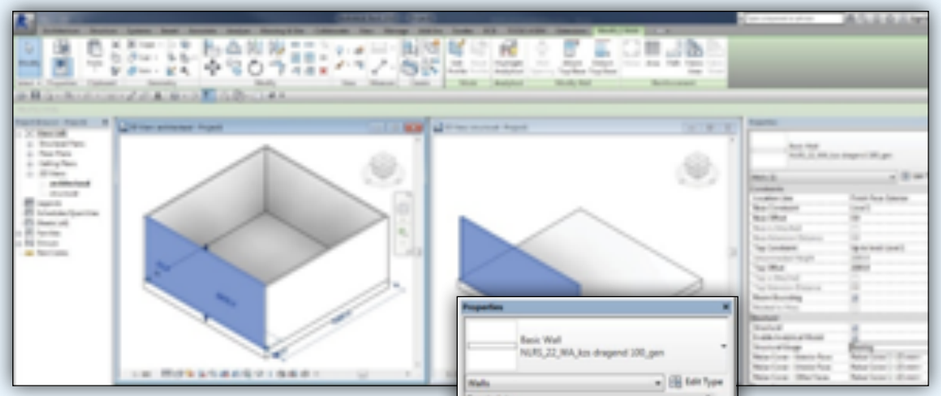
E. Examples
F. Change logs
Bibliography
Index

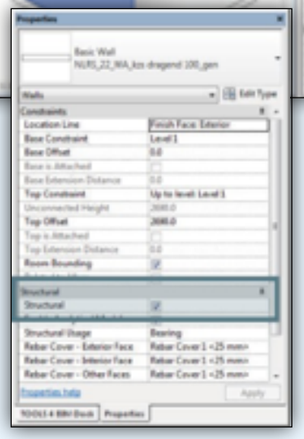
6.1.4.19 Pset_ShadingDeviceCommon
6.1.4.20 Pset_SlabCommon
6.1.4.21 Pset_StairCommon
6.1.4.22 Pset_StairFlightCommon
6.1.4.23 Pset_WallCommon
6.1.4.24 Pset_WindowCommon
6.1.5 Quantity Sets
6.1.5.1 Qto_BeamBaseQuantities
6.1.5.2 Qto_BuildingElementProxyQuantities
6.1.5.3 Qto_ChimneyBaseQuantities
6.1.5.4 Qto_ColumnBaseQuantities
6.1.5.5 Qto_CoveringBaseQuantities
6.1.5.6 Qto_CurtainWallQuantities
6.1.5.7 Qto_DoorBaseQuantities
6.1.5.8 Qto_MemberBaseQuantities
6.1.5.9 Qto_PlateBaseQuantities
6.1.5.10 Qto_RailingBaseQuantities
6.1.5.11 Qto_RampFlightBaseQuantities
6.1.5.12 Qto_RoofBaseQuantities
6.1.5.13 Qto_SlabBaseQuantities
6.1.5.14 Qto_StairFlightBaseQuantities
6.1.5.15 Qto_WallBaseQuantities
6.1.5.16 Qto_WindowBaseQuantities

6.2 IfcSharedBldgServiceElements
6.2.1 Schema Definition
6.2.2 Types
6.2.2.1 IfcDistributionChamberElementTypeEn
6.2.2.2 IfcDistributionPortTypeEnum
6.2.2.3 IfcDistributionSystemEnum
6.2.2.4 IfcFlowDirectionEnum
6.2.3 Entities
6.2.3.1 IfcDistributionChamberElement
6.2.3.2 IfcDistributionChamberElementType

AcousticRating	P_SINGLEVALUE / IfcLabel	DE	Schallschutzklasse	Schallschutzklasse gemäß der nationalen oder regionalen Schallschutzverordnung.
		EN	Acoustic Rating	Acoustic rating for this object. It is provided according to the national building code. It indicates the sound transmission resistance of this object by an index ratio (instead of providing full sound absorbtion values).
		FR	IsolationAcoustique	Classement acoustique de cet objet. Donné selon le Code National du Bâtiment. Il indique la résistance à la transmission du son de cet objet par une valeur de référence (au lieu de fournir les valeurs totales d'absorption du son).
		JP	遮音等級	遮音等級情報。関連する建築基準法を参照。
		ZH	隔音等級	该构件的隔音等级。 该属性的依据为国家建筑规范。为表示该构件隔音效果的比率（而不是完全吸收声音的值）。
FireRating	P_SINGLEVALUE / IfcLabel *	DE	Feuerwiderstandsklasse	Feuerwiderstandasklasse gemäß der nationalen oder regionalen Brandschutzverordnung.
		EN	Fire Rating	Fire rating for the element. It is given according to the national fire safety classification.
		FR	ResistanceAuFeu	Classement au feu de l'élément donné selon la classification nationale de sécurité incendie.
		JP	耐火等級	主要な耐火等級。関連する建築基準法、消防法などの国家基準を参照。
		ZH	防火等级	该构件的防火等级。 该属性的依据为国家防火安全分级。
SecurityRating	P_SINGLEVALUE / IfcLabel	DE	Sicherheitsklasse	Sicherheitsklasse gemäß der nationalen oder regionalen Gebäudesicherheitsverordnung.
		EN	Security Rating	Index based rating system indicating security level. It is giving according to the national building code.
		FR	NiveauSecurite	Système de classification par indices, indiquant le niveau de sécurité.
		JP	防犯等級	防犯等級情報。関連する基準を参照。
		ZH	安全等级	表示安全程度的参考性等级。 该属性的依据为国家建筑规范。
IsExternal	P_SINGLEVALUE / IfcBoolean *	DE	Außenbauteil	Angabe, ob dieses Bauteil ein Aussenbauteil ist (JA) oder ein Innenbauteil (NEIN). Als Aussenbauteil grenzt es an den Aussenraum (oder Erdreich, oder Wasser).
		EN	Is External	Indication whether the element is designed for use in the exterior (TRUE) or not (FALSE). If (TRUE) it is an external element and faces the outside of the building.
		FR	EstExterieur	Indique si l'élément est conçu pour être utilisé à l'extérieur (VRAI) ou non (FAUX). Si VRAI, c'est un élément extérieur qui donne sur l'extérieur du bâtiment.
		JP	外部区分	外部の部材かどうかを示すブーリアン値。もしTRUEの場合、外部の部材で建物の外側に面している。
		ZH	是否外部构件	表示该图元是否设计为外部构件。若是，则该图元为外部图元，朝向建筑物的外部。

Section from the Propertyset IfcWindowCommon, which contains commonly used properties for windows.

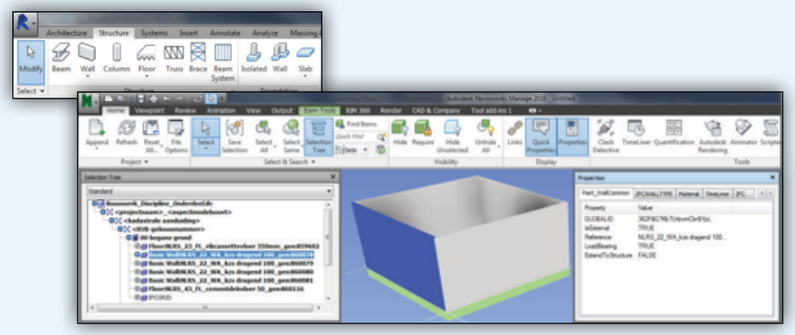


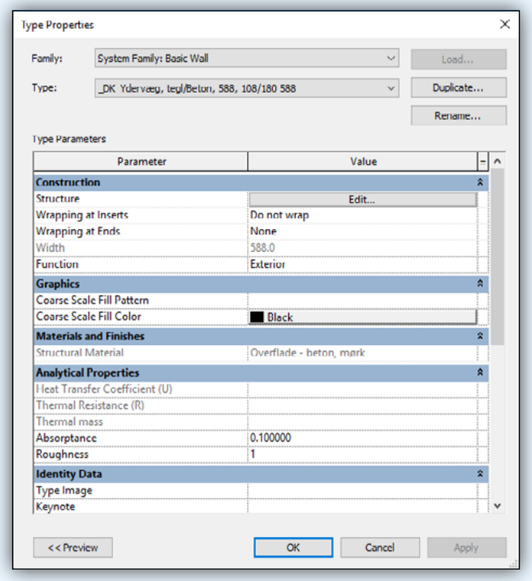


Structural / Non-Structural

If this property is set to "structural", the element is load bearing. In all other cases, it is non loadbearing.

When modeling walls via the "Structure" tab, Revit will automatically define a wall as a structural wall. Objects that have the "Structural" property checked in Revit will also have this property in IFC. The property can be found in the property "Pset_xxx-Common" (ie Pset_ WallCommon for walls), where the "LoadBearing" property is set to "True".





Exterior / Interior

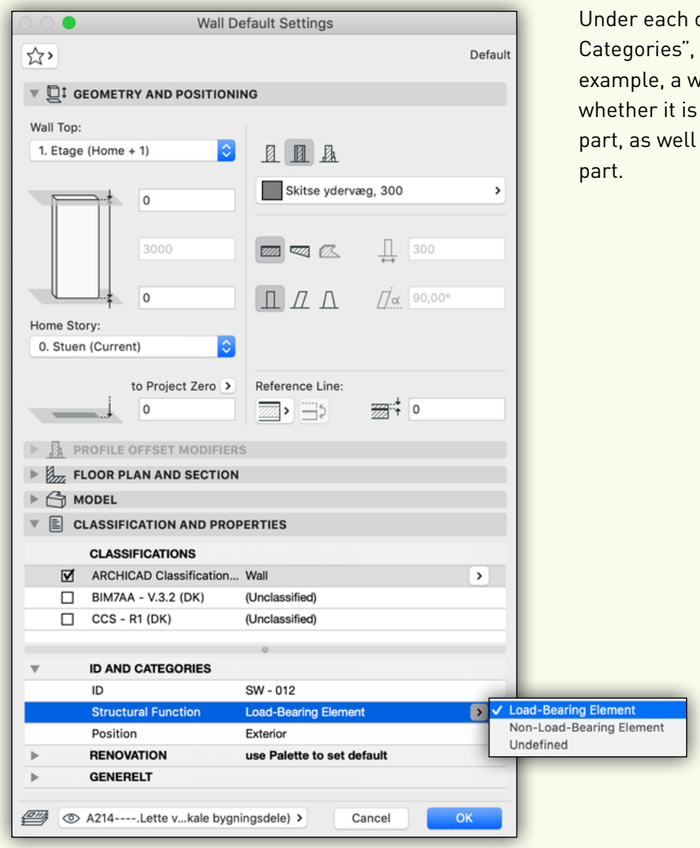
Walls, floors, ceilings, roofs and columns in the IFC file can have the "IsExternal" property. This property defines whether an object is interior or exterior. For system families such as walls and floors, this can be specified using the built-in "Function" parameter found in the properties of the type.

For "Loadable Families", such as columns, a project parameter can be used to describe whether the object is interior or exterior.

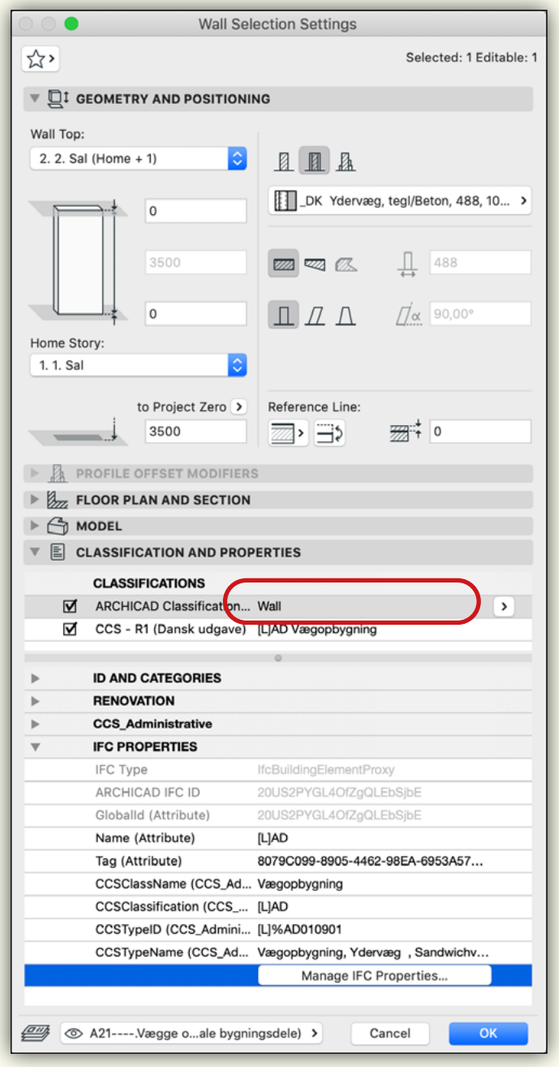
Fire Rating

As with the "IsExternal" parameter, the "FireRating" parameter must be added manually for those families, which don't have this as a default property. Note that the property built into Revit called "Fire Rating" is a Type parameter. Sometimes this may be desirable, sometimes it is better to make this an Instance parameter.

In the case of the latter, the parameter "Fire Rating" can be added as an Instance parameter. When the value for the parameter is not filled, Revit will use the parameter for the type when exporting to IFC. For "Loadable Families", the parameter "Fire Rating" can be created manually, as with "External / Internal".

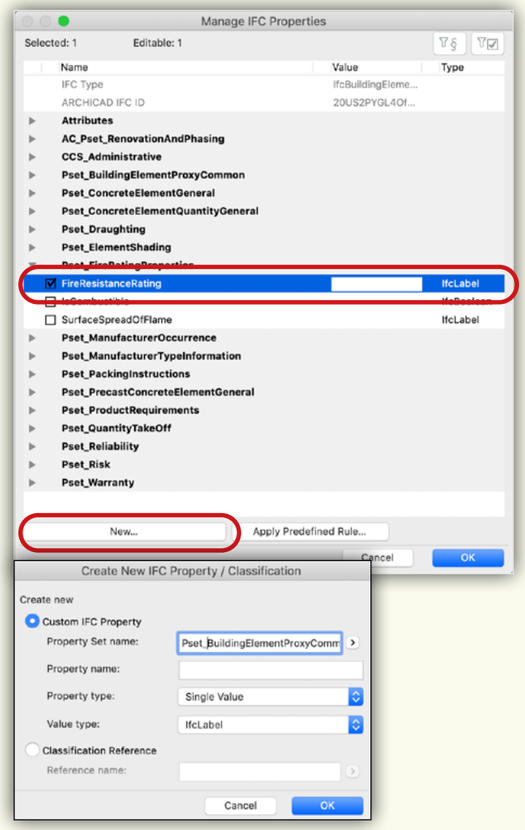


Under each object's settings, under the tab "ID and Categories", you can set the object's category (for example, a wall can be set to "foundation") and whether it is a load bearing, inner or outer building part, as well as the renovation status of the building part.

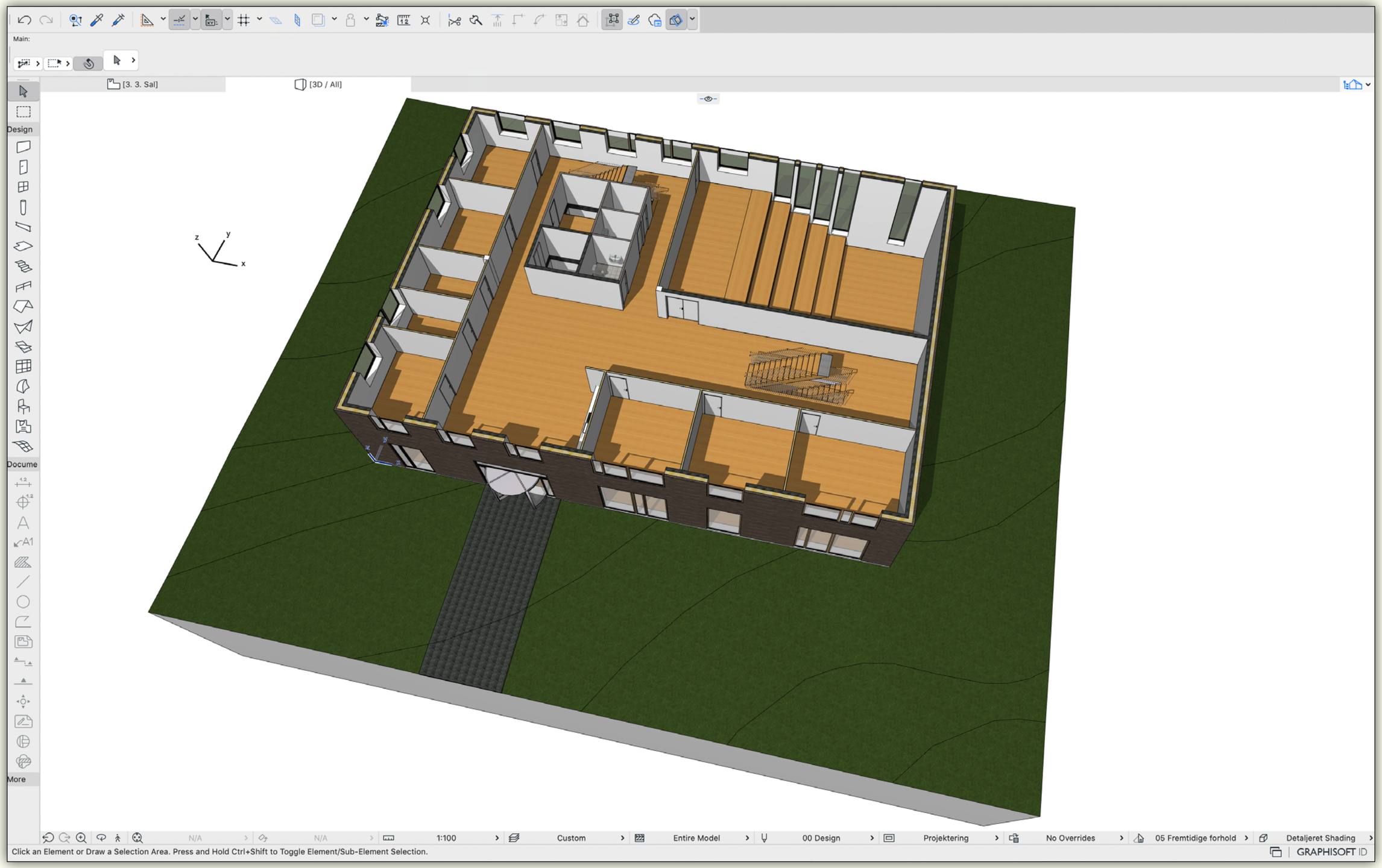


If you want to add parameters that are not already listed in the object's settings, they can be added manually by selecting the relevant data fields under IFC parameters.

If you want to create data fields that are not common IFC parameters, they are added by clicking "New".



Screenshot from the Molio house as it appears in ARCHICAD, with a horizontal section through the first floor.



VISUAL CHECK AND COMPRESSSING IFC FILES

Before sending your IFC file to your collaborators it is inherently important to check it for errors. This is done in a different program than the one you are exporting from.

Which software is best?

It is recommended to use an IFC viewer developed by a third party, which means that if the IFC file is exported from Revit do not use Autodesk software to check, and if the file is exported from ARCHICAD do not use GRAPHISOFT software.

Free IFC viewers

There are a number of good free IFC viewers on the market that are optimized for visual review:

Solibri Anywhere

One of the most used and proven IFC viewers is the Solibri Anywhere (previously called Solibri Model Viewer), which is known for the best translation of the IFC format on the market. The free program is a "light" version of Solibri Office (previously called Solibri Model Checker), and therefore lacks a number of features, such as the ability to assemble discipline models into federated models and to do automated checks of the models. Solibri Anywhere is an intuitive program that works for both Mac and Windows and can handle even very large, complex projects.

The Solibri Anywhere is owned by the company Nemetschek, which also owns the company GRAPHISOFT, which is the developer of ARCHICAD.

Download: https://www.solibri.com/solibri-anywhere

Tekla BIMsight

An IFC viewer with slightly more features is Tekla BIMsight, which, in addition to visual examination of discipline models, can also assemble several files into a federated model and perform low level collision control. Tekla BIMsight is only available for Windows.

Tekla BIMsight is owned by the company Trimble.

Download: https://www.tekla.com/tekla-bimsight/

KUBUS BIMcollab Zoom

One of the latest IFC viewers on the market is BIMcollab Zoom, developed by the Dutch company Kubus. The program is very intuitive and is visually more polished than the two above competitors with for example filtering features called "smart views". Like Tekla BIMsight, BIMcollab Zoom lets you collect discipline models into federated models but does not have automated check of models. BIMcollab Zoom runs on both Mac and Windows.

Download: https://www.bimcollab.com/en/Products/zoom

What to look for?

When looking through your exported file, you should pay attention to the following:

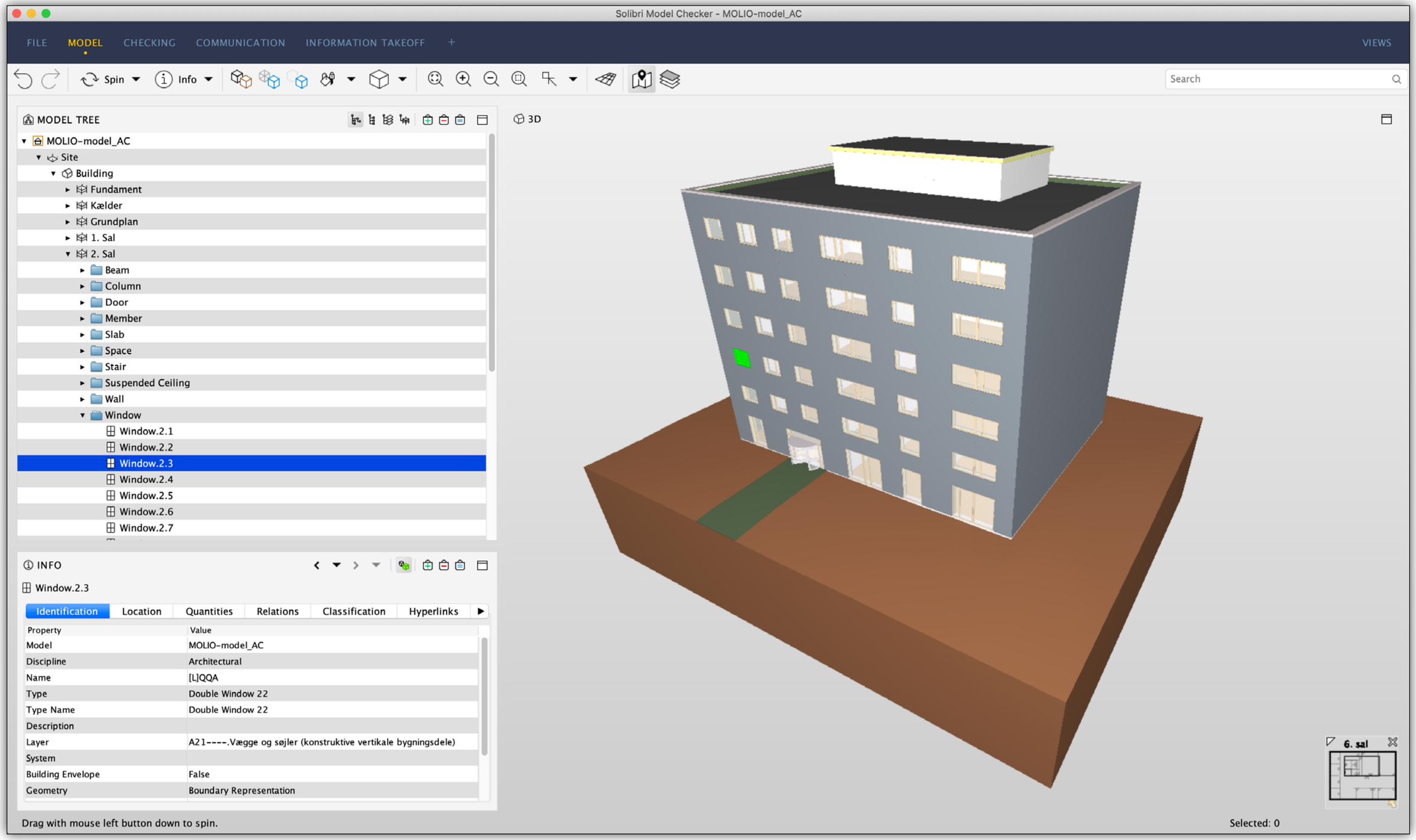
- Is the IFC file structured by building, levels and building components (including spaces) and does it contain the appropriate layers and property data for the project?
- Are parts of the model missing?
- Are all building elements correctly aligned with each other and are they on the correct levels?
- Does each building part have the relevant property data (do a random check)?
- Is the model placed correctly at the agreed coordination point (loading other models will quickly check this)?

Compressing the IFC File

An IFC file can contain a lot of repetition of data (redundant data), which is easily removed with the small and free Solibri IFC Optimizer program that removes redundant data, reducing the file size without reducing quality. This means that the file will be faster to send, upload and download, as well as import into other programs faster. Solibri IFC Optimizer removes approx. 10-30% of file size.

Download Solibri IFC Optimizer: https://www.solibri.com/solibri-ifc-optimizer

It also makes sense to compress the IFC file into a ZIP file (can also be done with Solibri IFC Optimizer). There are several BIM programs that can import ZIP compressed IFC files directly, but alternatively, the ZIP file must be extracted before import. ZIP compression removes up to 90% of file size, but also requires extra time for both compression and extraction.



Screenshot from the Molio house as it appears in the Solibri Office "Model Tree" (object hierarchy) unfolded and a window selected. At the bottom left is a number of window properties.