Revit IFC manual

Detailed instructions for handling IFC files

CONTENTS

1. INTRODUCTION

BIM (Building Information Modeling) is an intelligent 3D model-based process that gives architecture, engineering, and construction (AEC) professionals the insight and tools to more efficiently plan, design, construct, and manage buildings and infrastructure. At the heart of BIM is a smart building data model that incorporates not only 3D geometry but also all the relevant data relating to the building and its components. This kind of building data model can only be created using complex, BIM-enabled software such as Autodesk Revit®.

Provided that all of those involved in the planning are working with the same software, data exchange is loss-free. The native BIM format also facilitates the coordination of all planning stages and stakeholders.

In building projects, it can happen that those involved in the planning process are using different BIM software from different providers. The buildingSMART initiative (www. buildingSMART.org), in which Autodesk has

been actively involved from the very outset, has developed the IFC format to support such openBIM workflows. IFC allows the exchange of a specific subset of the native model.

Since the IFC4 release, the IFC format has met a recognized ISO standard (ISO 16739:2013). In its current version, buildingSMART maintains a list of all applications with certified IFC support: www.buildingsmart.org/compliance/certified-software/

IFC as the standard for exchanging BIM information

The Industry Foundation Classes (IFC) are an open standard for the exchange of building data models used in building design and construction across different software. They are used to exchange information within a project team and between software applications used in design, construction, procurement, maintenance, and operation. Current IFC Model View Definitions primarily support 3D geometry and property data. If exchange of 2D information, such as plan views and annotations, is important, Revit and coordination tools such as Autodesk BIM 360 which support the native file format should be preferred.

With IFC, the standard workflow is as per this model:

| Native format | > | IFC | > | Coordination/ No editing |

Find out more on the buildingSMART website:
https://www.buildingsmart.org/users/international-user-group-faqs/

Using IFC in practice

In an ideal scenario, the IFC file should be used for coordination purposes in an IFC viewer or as a reference within the editor software. For instance, the architect would receive an IFC file from the building equipment engineer, allowing them to see where the installation is placed. This workflow is referred to as the coordination workflow and is covered with the Coordination Model Views in IFC.

In some cases they may be the need for a design transfer workflow, for example if the architect had created the design in another software, but needs to carry on planning the building in Revit. This is the more difficult workflow and often requires manual adjustments to deal with the differences in software.

The Revit IFC manual

This document is intended to serve as a guide for Revit users handling IFC data and providing a better understanding of the settings available in Revit, discussing the way they can influence the quality and the content of the IFC file. The

Revit IFC manual therefore sets out the basics of IFC and explains in detail how to export, link and open IFC files in Revit.

2. THE BASICS

The key considerations when using an IFC file are the file format, IFC version, model view definitions, and file structure. We explain all these on the following pages.

2.1 IFC file formats

.ifc	Standard format, based on STEP (STEP: Standard for the Exchange of Product Model Data).
.ifcZIP	Compressed IFC files with much smaller file size; can be read by most software applications that support IFC. It can be unpacked to make the uncompressed IFC file visible.
.ifcXML	XML-based representation of IFC data, required by certain calculation software.

2.2 IFC versions (schema)

The IFC definitions are regularly updated and developed by buildingSMART. It is recommended that you use the latest versions wherever possible. Among other things, the IFC4 format allows better representation of complex geometries.

Current versions:
· IFC4 (still in beta, certification process in progress, offers certain advanced possibilities but is not widely supported yet);
· IFC2x3 (currently the most supported and stable format, certified in Revit and recommended for production);
· IFC2x2 (recommended if the recipient of the file does not have software with IFC2x3 or IFC4 support).

Revit no longer supports the following older versions: IFC2.0 (import still supported), IFC1.5.1, IFC1.5, and IFC1.0
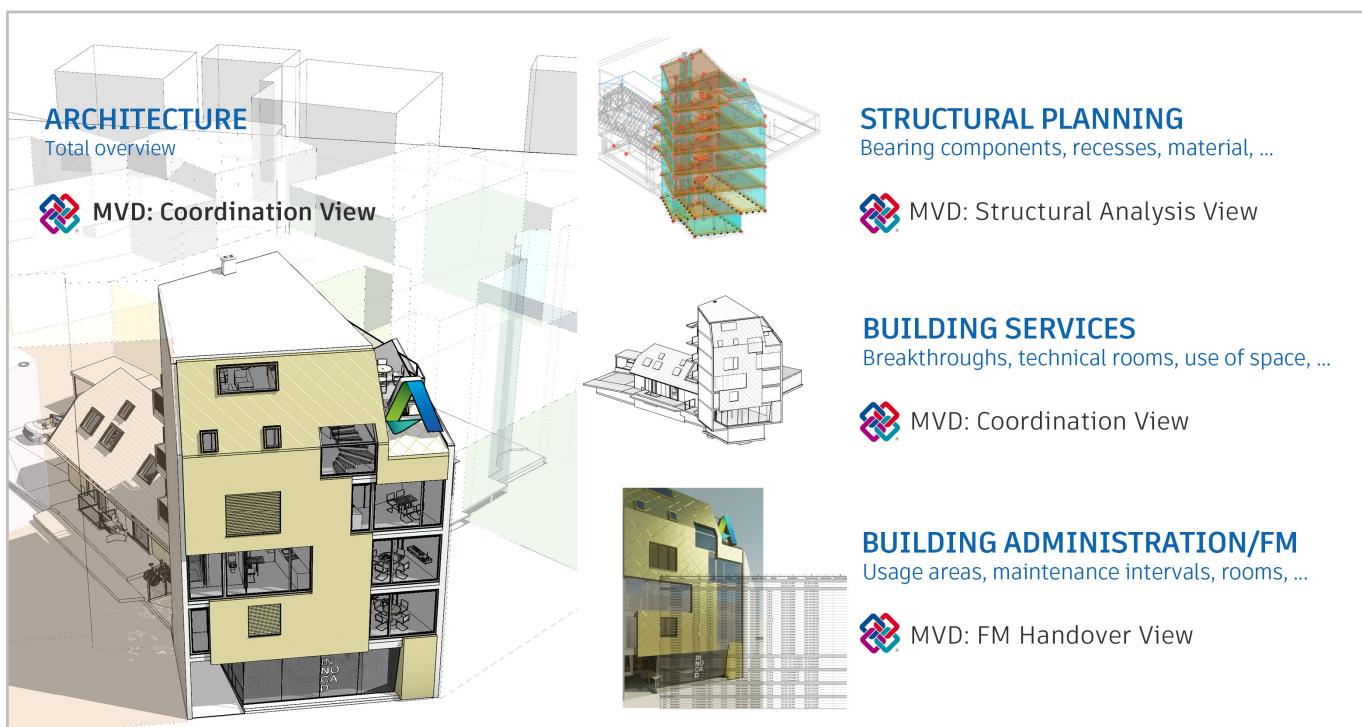
2.3 Model view definition (MVD)

In addition to the file format and versions, the model view definition determines how you use an IFC file, because it enables a specific data exchange scenario.

MVDs are used for the targeted exchange of specialized models, taking account of the graphic and content-related information that the planner needs.
For instance, thermal simulations require information about lighting areas in a wall and a room. Conversely, specialized IFC models require only the basic geometric information to be transferred to an FM system, and focus instead on spatial information and specific component features (such as system information, fire protection features, and usable areas) in relation to the MVDs. Moreover, a specialized model for structure planning requires specific information about the supporting building elements and apertures.

ARCHITECTURE
Total overview

MVD: Coordination View

STRUCTURAL PLANNING
Bearing components, recesses, material, ...

MVD: Structural Analysis View

BUILDING SERVICES
Breakthroughs, technical rooms, use of space, ...

MVD: Coordination View

BUILDING ADMINISTRATION/FM
Usage areas, maintenance intervals, rooms, ...

MVD: FM Handover View

The official, buildingSMART-defined MVDs available in Revit are listed below.

IFC4: Model Reference View

The model reference view was designed for the standard delivery of a reference model for specialist planners in IFC4. In the first instance, it provides an IFC model for coordination and model-based quantity determination, as referred to in the modeling software. A model exported as a Model Reference View is not suitable for importing with the intention of conducting further work on the geometry, as it contains only the most essential geometric definitions.

The model is not necessarily heavily simplified in graphic terms; it serves merely as a reference that may be quite detailed but cannot be edited.

IFC4: Design Transfer View (beta)

Introduced for the first time with IFC4, this is used to transfer IFC models for the purpose of importing and editing them in BIM-enabled software. As previously mentioned, the ability to transmit parametric design and complex contexts is restricted in the IFC format, so some manual adjustments will be needed to deal with differences in software, and the data should always be manually verified.

IFC2x3 Coordination View Version 2.0

Optimized for the coordinated exchange of BIM models between the main disciplines in the building industry. Coordination View 2.0—also known as CV 2.0—is currently the most widely used and supported Model View Definition. CV 2.0 supports the rudimentary parametric derivation of building components when importing into planning tools.

This MVD is primarily used for exchanging architectural, building technology, and construction engineering models.

IFC2x3 COBie 2.4 Design Deliverable

IFC format equivalent to the COBie (Construction Operations Building Information Exchange) output required by the UK government for their 2016 Level 2 BIM mandate for collaboration on public-sector work.

For exporting in COBie format, the relevant add-on from http://www.biminteroperabilitytools.com can also be installed.
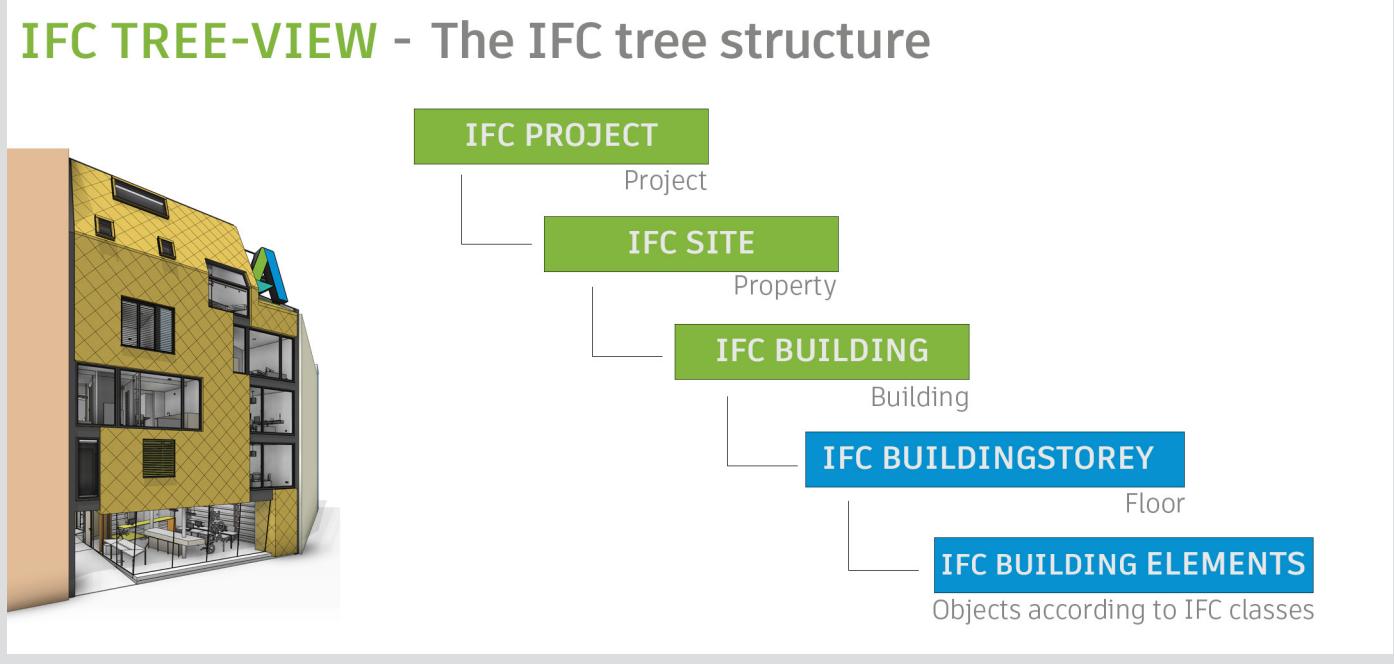
IFC2x2 Coordination View

Used only in isolated cases, such as when exporting MVDs for software that does not support IFC2x3.

Each of these model view definitions (MVDs) can naturally be adapted to the needs of the workflows—more information on this can be found in the following chapter, "Exporting IFC files."

To identify which MVD is used by an existing IFC file, you can open the file in a text editor of your choice. The header contains all the information about the MVD, the exact IFC exporter version and the software that it has been exported from:

```
FILE_DESCRIPTION(('ViewDefinition [ReferenceView_V1.0]'),'2;1');
FILE_NAME('Project Number','2016-12-14T17:37:10',(,'),(,'),'The EXPRESS Data Manager Version 5.02.0100.07:
28 Aug 2013','20161006_0315(x64) - Exporter 17.2.0.0 - Alternate UI 17.2.0.0','');
FILE_SCHEMA(('IFC4'));
ENDSEC;

DATA;
#1= IFCORGANIZATION($,'Autodesk Revit 2017 (ENU)',$,$,$);
#5= IFCAPPLICATION(#1,'2017','Autodesk Revit 2017 (ENU)','Revit');
```

2.4   IFC structure

IFC files create a building model based on a pre-defined structure that builds the model in a logical way. When it is saved, the IFC file format orders the IFC units hierarchically according to their type, as follows.

IFC TREE-VIEW - The IFC tree structure

IFC PROJECT
Project

IFC SITE
Property

IFC BUILDING
Building

IFC BUILDINGSTOREY
Floor

IFC BUILDING ELEMENTS
Objects according to IFC classes

For a list of all buildingSMART-defined classes, see
https://autode.sk/IFClinks.

This page also allows you to install all Revit-supported entities that can be used for export.

2.4.1 IFC classes and types

An IFC entity is a uniquely defined object in the IFC data model. Depending on the entity assignment and type definition, the object is allocated certain default attributes and dependencies within the IFC schema.
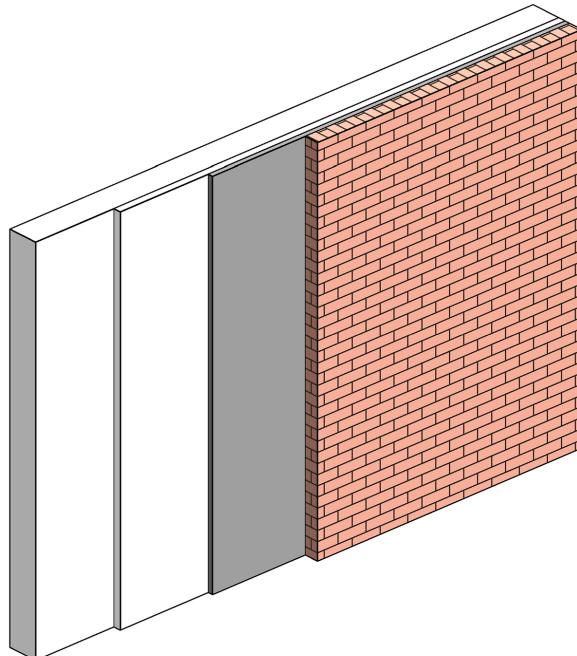
Choosing the right entity is crucial when exporting IFCs: if a wall is not assigned to the entity IfcWall, it is not allocated all the attributes that it requires to be described clearly. This means that it is not interpreted correctly by other coordination or evaluation programs in turn.

Distinctions are not only made between major categories, so components can also be assigned as entities to reproduce them more accurately within the IFC data model. This classification can roughly be compared to the sub-categories in Revit. A foundation of the IfcFooting entity can also be shown, depending on the type of component and its intended use, for instance by specifying the Ifc type as a sleeve foundation, "PILE_CAP".

Complex structures emerge on the basis of this system, enabling the generation of a data model in which every element can be geometrically and alphanumerically represented and clearly identified.


CLASS ALLOCATION
What am I?

TYPE DEFINITION
More accurate allocation

INTERACTION
Which objects can I affect?

GEOMETRY
What are my measures?

RELATIONSHIPS
What objects are superior or inferior to me?

STANDARD ATTRIBUTES
What information do I generally provide?

2.4.2 Geometric representation of IFC objects

There are three basic possibilities for geometrically representing a three-dimensional IFC object:
- extrusions
- solid body representation using a sweep, and
- representation using B-reps.

Extrusions
Are the most common and simple graphical method and are used for most of the cases when the shape can be described by a simple profile.

Swept solids
As the name implies, an element is created with the swept solid method using a sweep. In this case, a defined profile is led along a path (direction vector) to generate the solid. This profile may change due to rotation or distortion along the path. Revit uses this method for describing rebar and other shapes which cannot be described with extrusions.
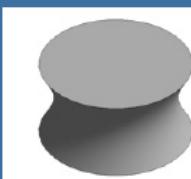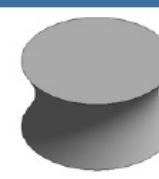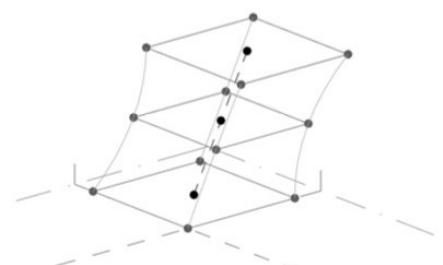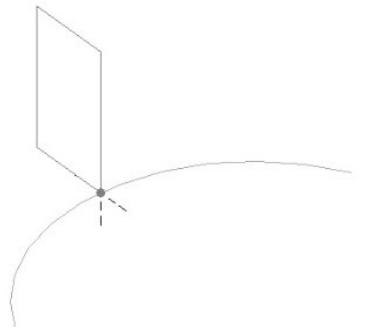
B-rep
The method known as boundary representation (B-rep) can also be described as a boundary surface model. The surfaces of a component are represented using coordinates and together form the actual solid, allowing even complex forms to be represented.
B-rep objects use complex calculations to represent individual surfaces in detail, and thus use more data memory.

NURBS and other smooth surfaces
In the IFC4 schema, it is possible to generate B-rep objects as advanced B-reps using NURBS (non-uniform rational B-splines) surfaces. The memory space required is thus drastically reduced, while the bodies are represented more accurately.

Revit	IFC2x3 CV 2.0	IFC4 DTV
	1442 facets 227 KB	4 facets 16 KB
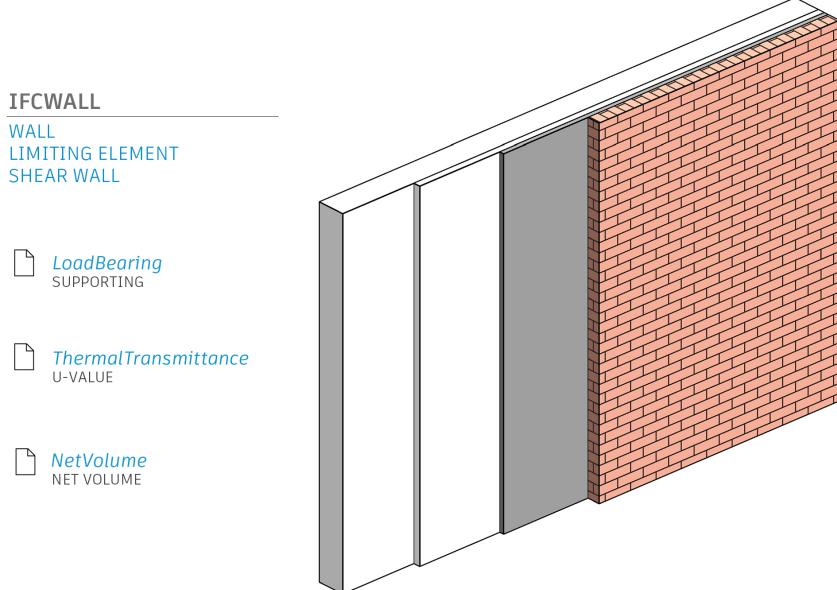
2.4.3 Default attributes and parameters

One of the key considerations when transferring IFC data models is the provision of information that can be correctly interpreted and evaluated by specialist planners and their planning and calculation tools, regardless of the internal attribute structure and descriptions within the respective applications.

IFC properties can be formulated across the board using default attributes. These attributes are stored in the IFC definition and have English names.

Some BIM applications can automatically assign internal attributes to IFC-compliant default attributes. This ensures that the necessary information is provided to represent an object.

When exporting an object out of Revit, not only is all the necessary information relating to classification, global location, and geometric representation transferred, but also default attributes. For instance, for a wall the value from the instance parameter 'supporting' is automatically assigned to the IFC attribute LoadBearing.

IFCWALL

WALL
LIMITING ELEMENT
SHEAR WALL

LoadBearing
SUPPORTING

ThermalTransmittance
U-VALUE

NetVolume
NET VOLUME

Please note that the IFC exporter only transmits valid property values, i.e. those that are not empty. If the parameter is missing from your IFC file, this is likely because the Revit parameter does not have a value. This optimizes the file size, as empty data fields are not exported.

An overview of all default parameters defined in IFC format is provided by buildingSMART in the form of parameter sets (P-sets).
As an example, these are the default parameters for a wall:

Pset_WallCommon
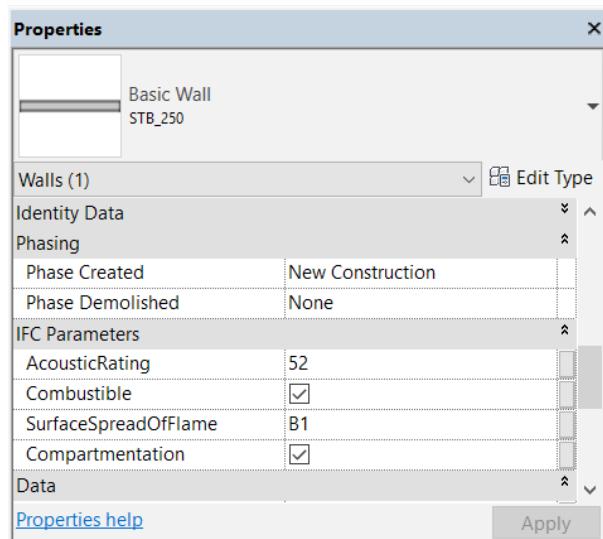
Default parameters in Revit:

Reference	Component type (type name)
FireRating	Fire-resistance class (type parameter)
ThermalTransmittance	U-value (type parameter)
IsExternal	Exterior component (type parameter, given as yes/no)
LoadBearing	Load-bearing (instance parameter)
ExtendToStructure	Fixed on top (behavior)

The following parameters are also part of the Pset_WallCommon, but are either unavailable or not assigned by default in Revit:

AcousticRating	Sound insulation class
Combustible	Combustible material
SurfaceSpreadOfFlame	Fire behavior
Compartmentation	Fire compartment-defining component

In order to create these parameters in Revit, they have to be created using the exact name, the correct type (text/number/yes/no, visible in the buildingSMART documentation):

Additional IFC parameters in Revit

IFC properties after exporting in FZK Viewer

As soon as these parameters are available and have a value, they are taken into account in the export. The advantage of this standardization is that the parameters of other programs are automatically recognized and assigned correctly. Since version v18.4.0 the Autodesk IFC Export supports all common property set defined in the IFC schema.

Thanks to the extensive settings of the Revit IFC exporter, other parameters that do not appear on this list can be exported. This is covered in greater detail in Chapter 4, "Exporting IFC files."

2.4.4 Reference structure within an IFC file

An IFC file can be opened in a text editor. This can be very helpful for analysis or troubleshooting. There are two parts to the underlying structure of an IFC file: the header and the body. While the header contains general information about the building model, the IFC version and software used, the schema, and the MVD, the body contains the information about the geometry and attributes of the building itself.
In IFC format, the description of an element begins with a line that classifies, uniquely identifies, and names the object. Our wall example might look like this:

#177= IFCWALLSTANDARDCASE(‚1sfW$3YQj9jBEISmjkeABP‘,#41,‘Basiswand:STB 20.0:388701‘,$,‘Basiswand:STB 20.0:3895‘,#146,#173,‘388701‘);

Within this definition line, the wall object refers to other lines in the file structure, distinguishable by the # that precedes them. These describe it in greater detail and refer in turn to other lines:

#146= IFCLOCALPLACEMENT(#128,#145); ➜ Reference to the lines that define the global position
#173= IFCPRODUCTDEFINITIONSHAPE($,$,(#152,#170)); ➜ Reference to lines that provide a geometric description of the wall

This reference structure continues until a logical data model has been generated that provides a clear description of each object.

The advantage of this method is that particular attributes are stored only once and can be used by other components by means of referencing. This significantly reduces the file size and means, for instance, that components with the same material refer to the same material definition lines within the data model.

2.5   Open-source IFCs

Revit comes with an integrated IFC interface whose functionality can be extended using the Open Source extension. Another advantage of this plug-in is that it is constantly being developed and is regularly updated by Autodesk, regardless of Revit update cycles.

Developers can access the complete source code and customize the exporter as required. This is especially useful if specific workflows require this kind of customization for a construction project.

If you are working with IFC files in Revit, you should first install the latest version of the Open Source extension. This can be found in the Autodesk AppStore:

http://apps.autodesk.com

After installation, you will not see a new icon in Revit. Instead, the plug-in overwrites the standard dialog fields. If you are a developer and want to work with the source code, you can find additional information on SourceForge: **https://sourceforge.net/projects/ifcexporter**:

If working with IFC files, it is important that you are not only informed about their structure (MVD) and version, but also that you are aware of the possibilities and importance of the individual export and import options. You will only get an IFC file with all the information that you need if you use the right settings. The following chapters show you how to do this.
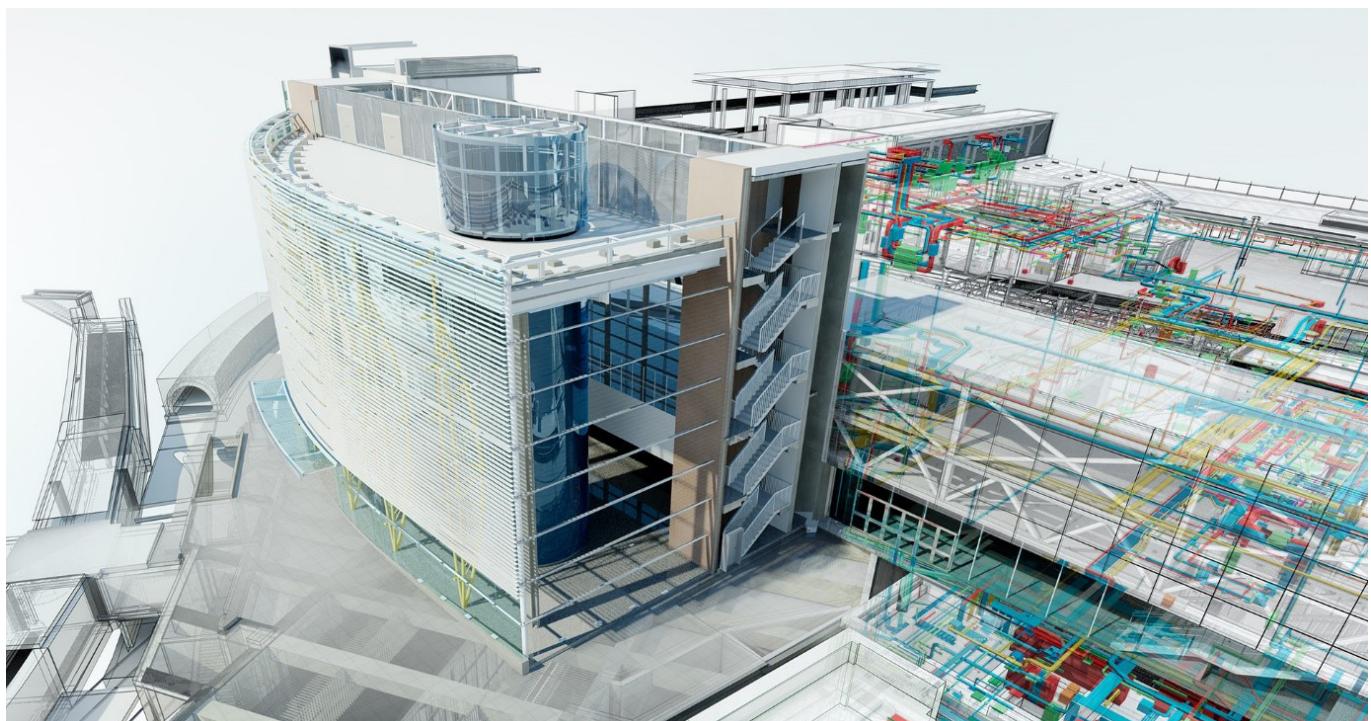
2.6 IFC viewers

Before you pass an IFC file on to planning partners or use it yourself in Revit, you are advised to test it in an IFC viewer and check the export result.

There are a range of IFC viewers on the market. These viewers support certain IFC functions with different levels of effectiveness, depending on their focus.

As Autodesk customers, you have access to Navisworks as part of the AEC Collection. Navisworks can be used not only to view IFC files, but also to carry out collision testing, create construction workflow simulations and determine quantities.

As part of the Autodesk BIM 360 services, you can also view and share IFC files (and many other formats) directly in your browser.
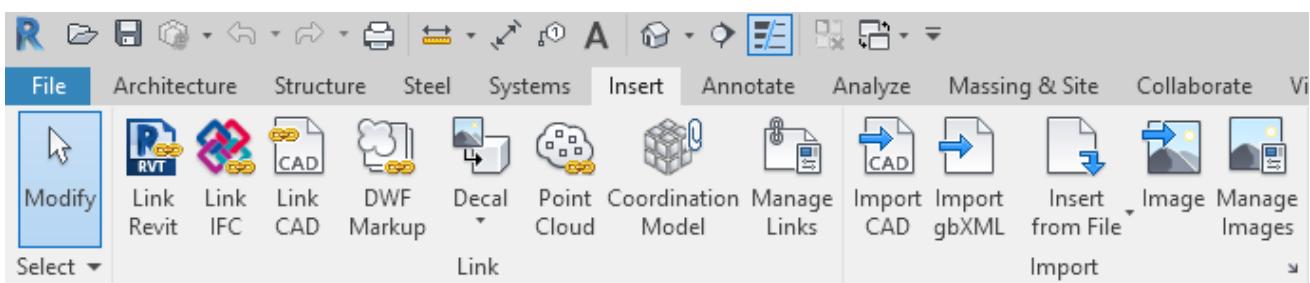
(c) London Blackfriars station, courtesy of Network Rail and Jacobs(r)

The FZK Viewer, created by the Karlsruhe Institute of Technology (KIT), has established itself as an open-source, independent viewer. It is highly streamlined and manageable, and is therefore well suited to checking small to medium-sized models quickly. It was used for some of the screenshots that appear in this manual.

The current version of the FZK Viewer can be downloaded from the KIT website:
www.iai.kit.edu

3.   LINKING IFC FILES IN REVIT

Besides Revit models and CAD data (2D/3D), you can also link IFC models in a Revit project:

This option links the IFC file within the Revit project, so that updates are possible at a later date. This process is similar to the linking of other Revit or CAD files with Revit. You can find the linked IFC file in the project browser:

The IFC file is automatically updated at the start of the project and can be manually updated at any time during editing. To do this, select the IFC file in the project browser and reload it via the context menu (right-click). Revit automatically creates a Revit file in the same folder, which contains the IFC file:

You should not move, change, or open this file.

Linking IFC files is the preferred option for coordination purposes and gives the best results because it does not generate native Revit elements.

There are no detailed settings for the linking of IFC files, as the system automatically generates the best result.

The maker's export settings play a far greater role in the quality of the linked file.
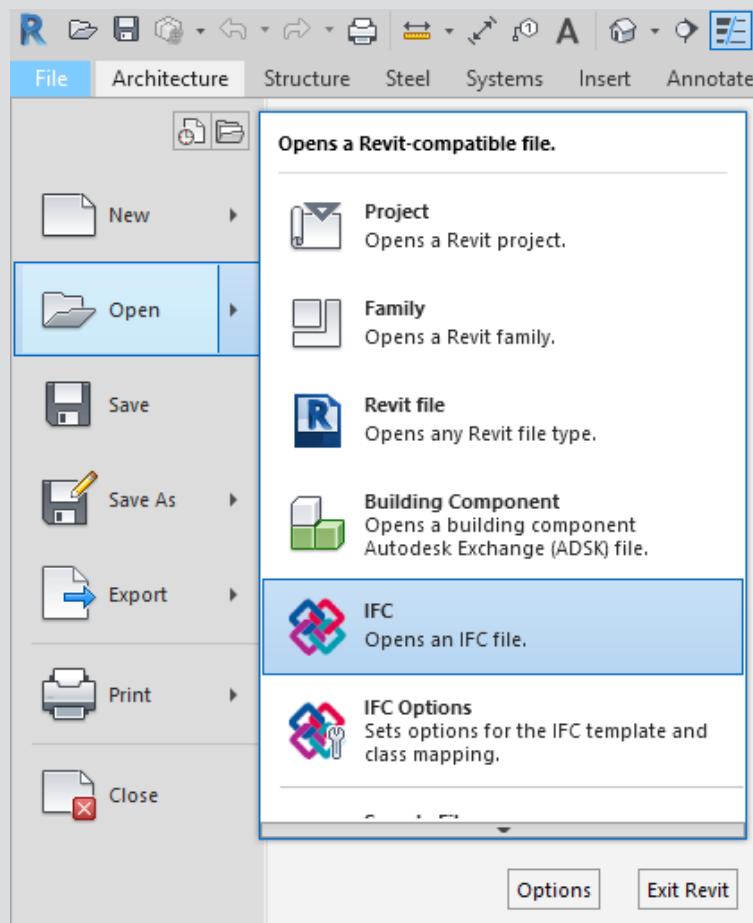
When linking an IFC specialist model, a "shared parameters" file is created at the location of the IFC file. This can be used to create filters that allow the selection, graphic overwriting, or hiding of components from the linked specialist model. Specific examples of applications can be found in the final chapter of this manual.

4. OPENING IFC FILES

In some cases, it is necessary to open the IFC file in Revit to continue editing it. This may be the case if the architect has created the design using different software but now intends to carry on planning the building in Revit.
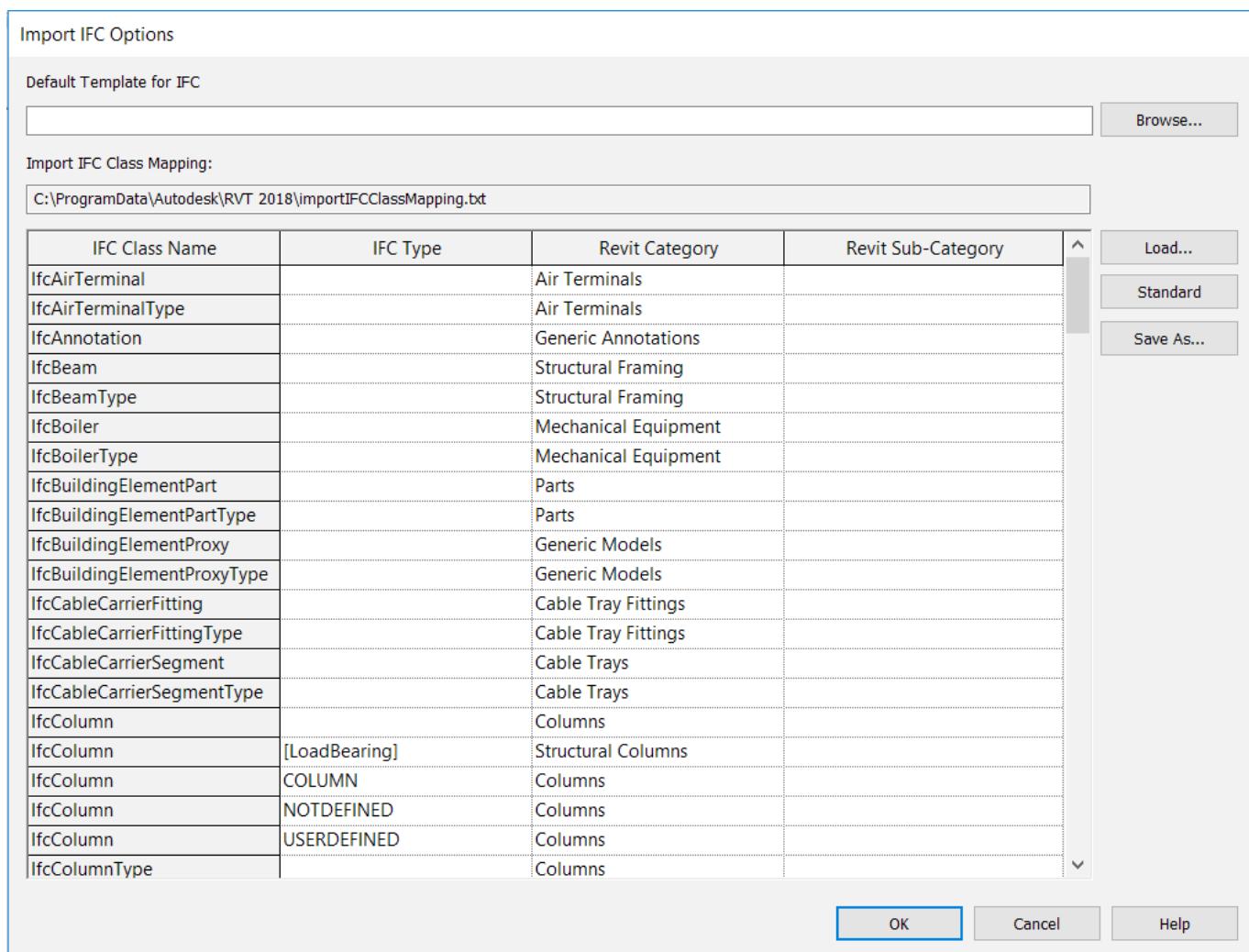
As already mentioned in the Introduction, this workflow does not come fully recommended as the model loses some of its intelligence and parametrics when it is exported to the IFC format. In some cases, however, an imported IFC model can prove a good foundation for further planning. When importing, Revit transforms each element contained in the IFC file into a native Revit object. For this reason, importing large models can also be quite time-consuming. Ultimately, the quality of the import depends very much on the quality (export settings) and the content (IFC version, MVD).
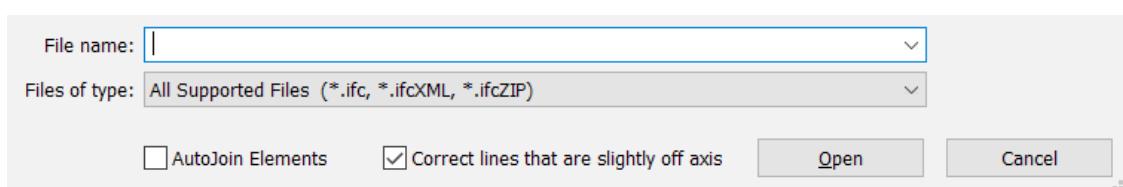
4.1 Mapping table

The mapping table for IFC imports is structured in a similar way to the export mapping table and can be called up via Revit > Open > IFC options:

IFC Class Name	IFC Type	Revit Category	Revit Sub-Category
IfcAirTerminal		Air Terminals	
IfcAirTerminalType		Air Terminals	
IfcAnnotation		Generic Annotations	
IfcBeam		Structural Framing	
IfcBeamType		Structural Framing	
IfcBoiler		Mechanical Equipment	
IfcBoilerType		Mechanical Equipment	
IfcBuildingElementPart		Parts	
IfcBuildingElementPartType		Parts	
IfcBuildingElementProxy		Generic Models	
IfcBuildingElementProxyType		Generic Models	
IfcCableCarrierFitting		Cable Tray Fittings	
IfcCableCarrierFittingType		Cable Tray Fittings	
IfcCableCarrierSegment		Cable Trays	
IfcCableCarrierSegmentType		Cable Trays	
IfcColumn		Columns	
IfcColumn	[LoadBearing]	Structural Columns	
IfcColumn	COLUMN	Columns	
IfcColumn	NOTDEFINED	Columns	
IfcColumn	USERDEFINED	Columns	
IfcColumnType		Columns	

Import IFC Options

Default Template for IFC

Browse...

Import IFC Class Mapping:

C:\ProgramData\Autodesk\RVT 2018\importIFCClassMapping.txt

Load...

Standard

Save As...

OK   Cancel   Help

4.2 Import options

The Open dialog window offers a few options that can help you handle IFC files in Revit.

File name:

Files of type: All Supported Files (*.ifc, *.ifcXML, *.ifcZIP)

☐ AutoJoin Elements   ☑ Correct lines that are slightly off axis   Open   Cancel

AutoJoin elements

connects walls, supports and other elements automatically using the same approach as that applied in the Revit modeling. This can lead to undesirable results for complex structures or long export times, so this option can be disabled if necessary.

Correct lines that are slightly off axis

is an option that will be familiar from the link/import CAD function, and seeks to correct elements that deviate slightly from the main axis. This option can be disabled if necessary, as it can lead to problems with elements which are meant to be off axis, like site boundaries for example.

The quality of the imported data depends not only on the import settings, but also to a large degree on the export settings and the modeling method applied in the original software.

The Revit–IFC import interface is regularly updated together with the export module, but is also subject to the limits of technical feasibility and the IFC format. It is not possible for a standardized format to transfer complex parametrics and contexts.

Essentially, in most cases this workflow represents a good working basis. If IFC data have to be processed further, one should always expect the loss of data in comparison with the original software.

For example:
To create a floor slab in Autodesk Revit, a profile is sketched and is generated (or extruded) at a right angle to the profile, taking into account the component types and the defined layer thickness. Autodesk Revit generates standard floor slabs on the basis of this principle.

This approach is very similar to that seen in the IFC schema, and typically gives optimum results, generating clean standard floor slabs in Revit.

As planning proceeds, the slab is given an opening and a gradient, with the underside remaining naturally flat. Within the IFC schema, representing this component as a swept solid, in other words generating a solid using a defined profile and pathway, is no longer possible.

The element is thus described as a B-rep object in the data model, with all the necessary geometry points. When importing, the slab is recognized on the basis of its object class (IfcSlab) and assigned to the correct category. However, the geometric description no longer corresponds to the basic principle for creating a floor slab, so a project family is generated.

This is not a fundamental problem, especially in terms of coordination, as the slab is correctly represented. Nonetheless, if you want to edit the imported slab, certain limitations arise, as you cannot work on it using the normal tools.

Recommendations on construction
Edit profile and Edit Footprint are very useful tools when it comes to modifying the geometry of wall and ceiling elements. However, these functions can make the planner's life particularly difficult when it comes to exchanging specialist models by importing or exporting IFC files, as geometries may be incorrectly interpreted or represented.
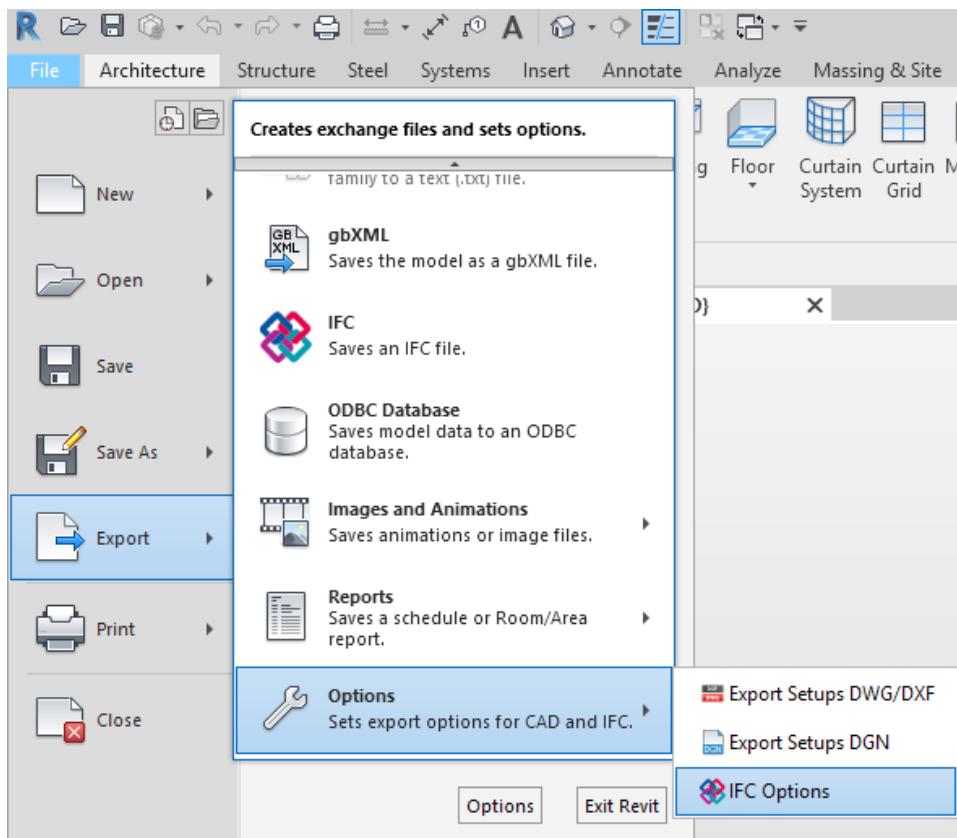
Likewise, an opening sketched using editing tools will not always result in the generation of an "Opening element," as it would in the case of a void form.

5. EXPORTING IFC FILES

When choosing the right settings for exporting an IFC file, it is important to first think about what the file is to be used for: Will it be used only for the purposes of coordination, or does it have to be processed using a different editor software? Below you can see the impact of the respective mapping and export settings, together with the available options.

5.1   Mapping tables

Before exporting an IFC file, it is important to check the settings. You can find these at: Revit > Export > Options > IFC options.

Revit categories are assigned to IFC classes using a mapping table. This table is stored as a text file (*.txt) and can be customized directly in Revit or by using a text editor.

The first column, the Revit category, is unchangeable and automatically lists all the categories and sub-categories available in the Revit project.

The IFC class name column contains the IFC class to which the sub-category or category should be assigned. If the category is not to be exported, you can enter Not Exported. This can vary depending on the workflow and requirements

of the contents and structure of the IFC model, and therefore cannot be specified across the board. However, Revit is delivered with basic settings that meet a certain basic standard.

The IFC classes and types must be entered manually, with the correct spelling. This is how footings from the IFC class become assigned to the IfcFooting. The list of classes supported in Revit is regularly updated and can be called up

here for version 2019: https://autode.sk/IFClinks.

An **IFC type** can also be assigned which, like the sub-categories in Revit, allows more accurate differentiation within a category. For a pile cap, the type "PILE_CAP" can thus be specified in addition to the category IfcFooting.
The types available in IFC format can be checked for their respective IFC release on the buildingSMART page. Current links and lists can be found at https://autode.sk/IFClinks.
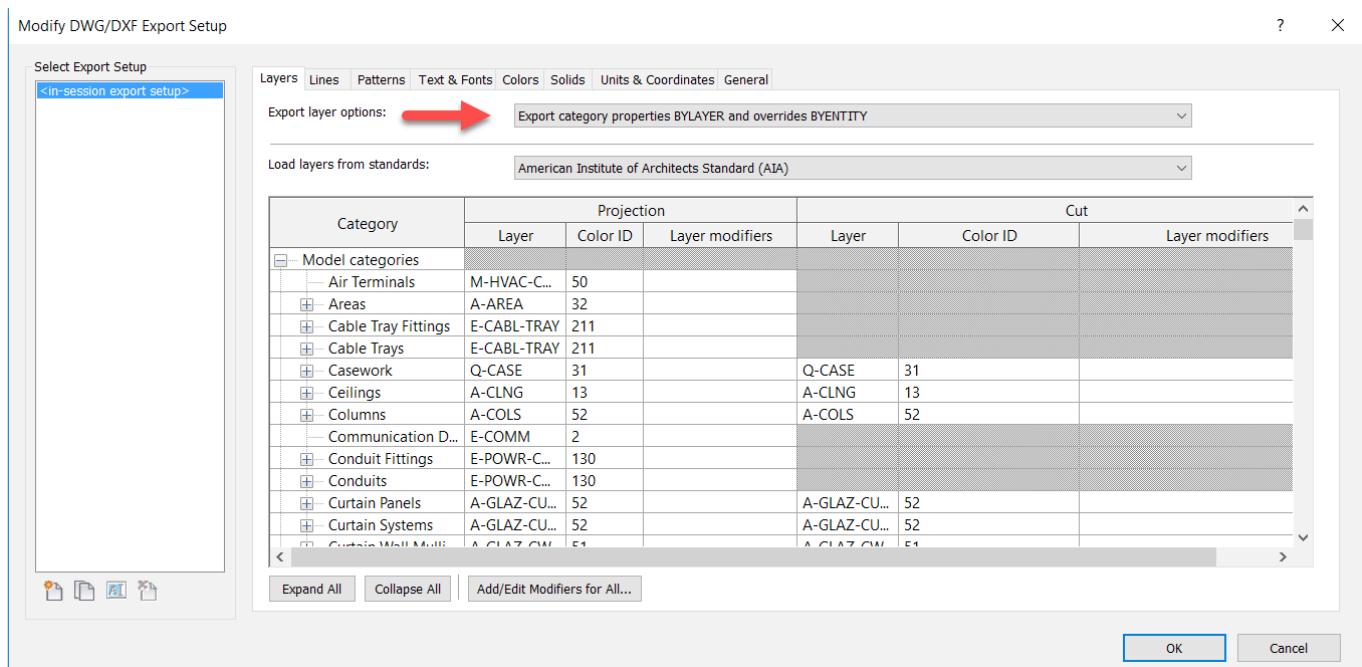
The pre-configured mapping table is stored by default via the system path C:\ProgramData\ Autodesk\RVT**(Version)**\exportlayers-ifc-IAI. txt. A standardized export can be performed across different companies using a centrally stored mapping table.

It is worth noting that unlike Revit, certain BIM programs do not only work with categories, but also with the layers used in CAD operations.

When exporting IFCs from Revit, the settings file that was also used when exporting the CAD (.dwg or .dgn) is accessed. This file is defined in the Revit.ini and can be changed there if required. The default configuration file can also either be customized or another configuration specified.

The default configuration file for Revit 2018 can be found at C:\ProgramData\Autodesk\RVT 2018\exportlayers-dwg-AIA.txt

The configuration can also be customized and exported via the Revit dialog, which can be found at > Export > Options > Export settings DWG/DXF:

To refer to a different configuration file, change the following path in the Revit.ini file using a text editor: ExportLayersNameDGN=C:\ ProgramData\Autodesk\RVT 2018\exportlayers-dwg-AIA.txt

The Revit.ini for the 2018 version is located at C:\Users\<username>\AppData\Roaming\ Autodesk\Revit\RVT 2018

As this is a hidden folder by default in Windows, the display of hidden folders has to be activated beforehand.
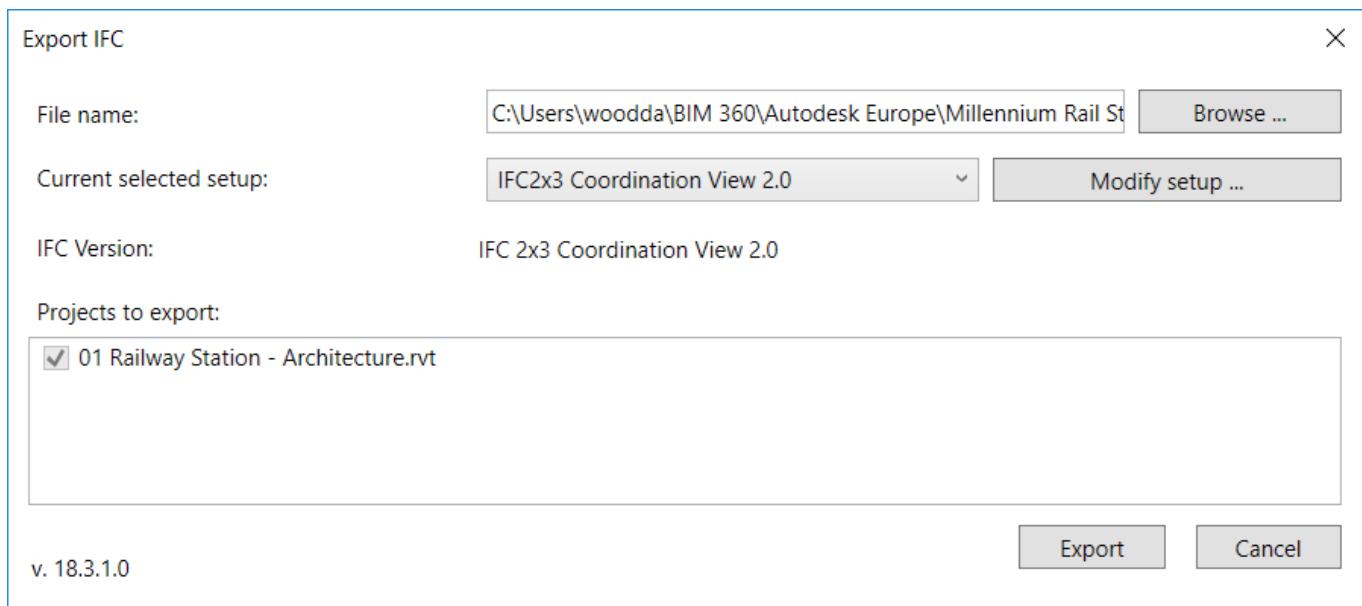
To completely reset the Revit.ini, the file can be deleted under the aforementioned path. It will be recreated after restarting Revit.

5.2 Revit IFC exporter settings

A project opened in Revit can be exported via File > Export > IFC. Extensive settings are possible in the subsequent dialog, explained in greater detail below.

Main dialog
First, this main window opens:

The desired name and file location of the IFC file to be exported are specified under File name.

Current selected setup: allows export according to preset configurations. The selection of the schema and MVD to be used is crucial in defining the content and structure of the IFC file and should therefore be coordinated and chosen according to their intended use.

Modify setup: these settings can be adapted if required and custom definitions can be created, which are stored with the Revit project.

Projects to export: Here you can choose projects that are currently open in Revit and are to be exported. When exporting several projects, the same settings are used for all files, and saved to an IFC file.
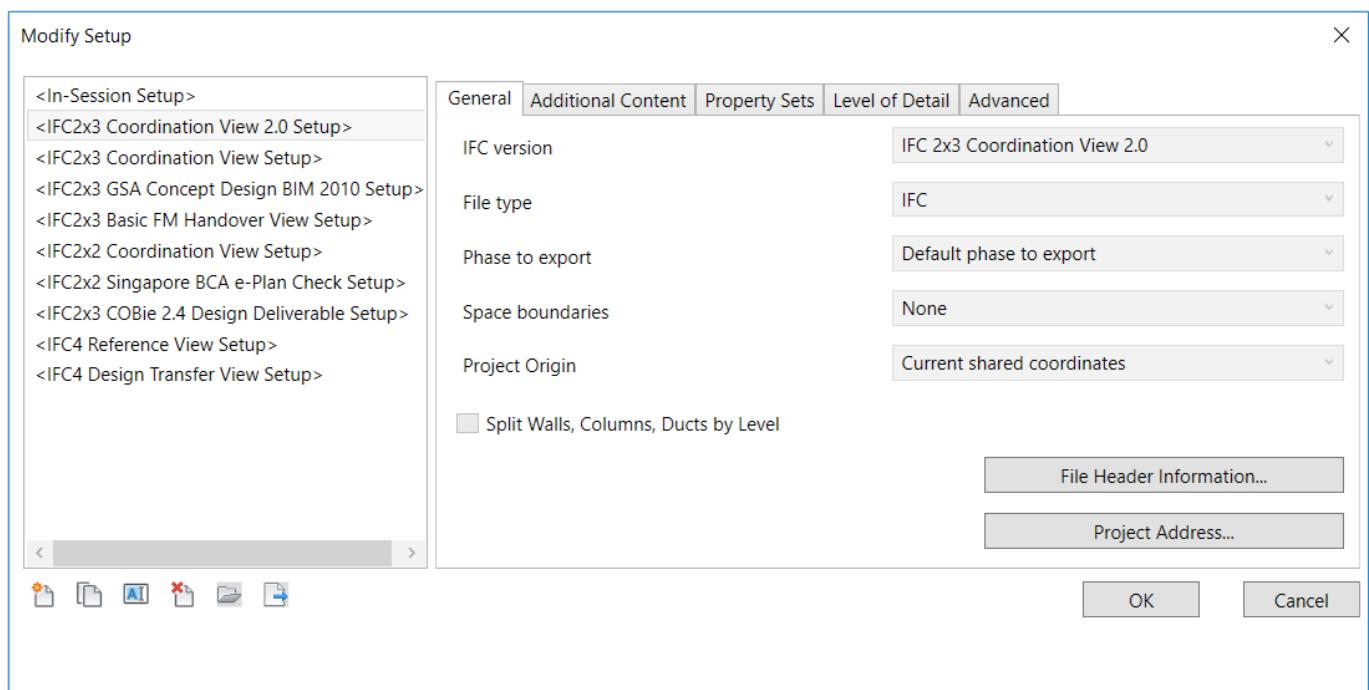
Modify setup

Individual settings for IFC export can be made and saved in the Modify setup window.

All the pre-installed configurations are displayed on the left-hand side. The default configurations are marked with brackets and cannot be changed, renamed or deleted. However, they can be copied and thus used as the basis for custom configurations. In addition, you can export or import previously created settings, which are stored outside of the Revit project.

5.2.1 General settings

The following general settings can be found in the general tab for Advanced IFC export settings:

The IFC version allows you to select the IFC schema and the MVD, as explained in detail earlier in this manual. The most commonly used schema is IFC 2x3 Coordination View 2.0, as it is supported by most programs. For complex geometries, you can try to use IFC4 because of improvements to the geometry translation, but keep in mind that is still in beta and double check the result for any errors or missing elements.

The file type determines the file format in which the exported file will be saved. For big projects, the compressed *.ifczip format can be used, which is also supported by most IFC viewers. If required, the *.ifczip file can be unzipped to get the uncompressed *.ifc file.

⊛ rac_advanced_sample_project.ifc	53.005 KB
⊛ rac_advanced_sample_project.ifczip	9.895 KB

The spatial boundaries option determines how room boundaries that are required for various energy calculations and quantity and material statements are exported. These boundaries are classified in levels, according to their purpose and the information that they contain:

None does not export boundary surface information. Only the reference scope and reference to adjoining rooms and components are stored.

Level 1 exports boundary surfaces for quantity and mass evaluation, taking room bounding elements into account. Openings created with the "Edit Profile" tool and surfaces edited with the "Split Faces" tool are not considered.
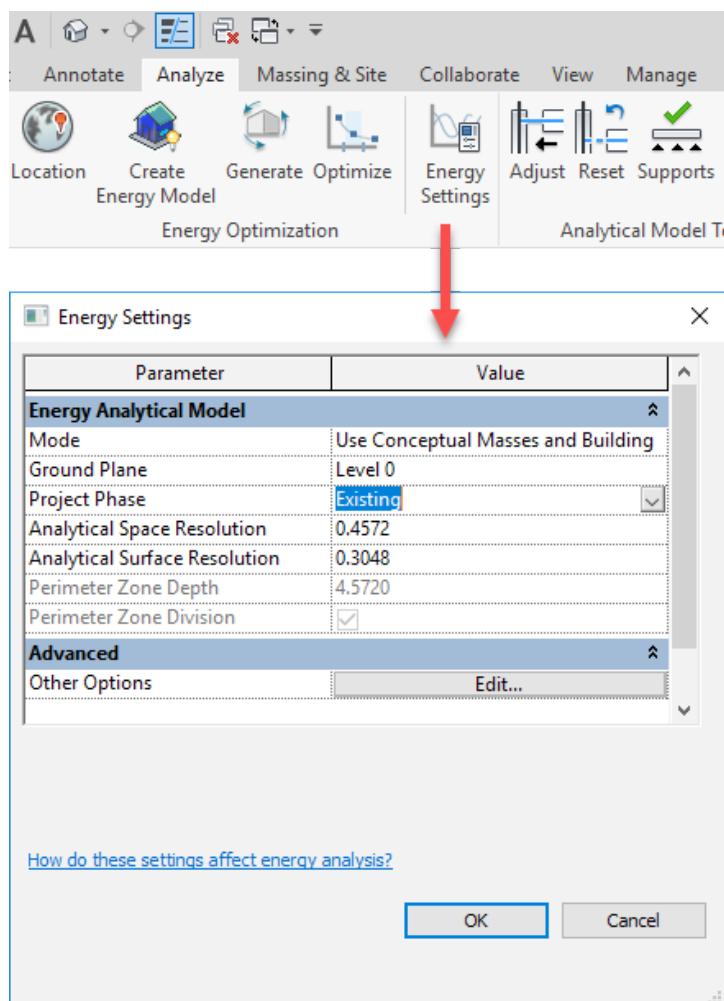
Name	Value	Description
⊟ **IfcRelAggregates**		
IfcBuildingStorey	Level 1 (#124)	

Name	Value	Description
⊟ **IfcRelAggregates**		
IfcBuildingStorey 🔳	Level 1 (#124)	
⊟ **IfcRelSpaceBoundary**		
⊞ #440 -> #916 🖳		
⊟ #440 -> #962 🖳		
Name	1stLevel	
Description	?	
OID	#1092	
GUID	2mAPm7hMr6GRNKX5Hz...	
Related Buildingele...	IfcSlab[Floor] (#962)	
Physical/Virtual	PHYSICAL	
Internal/External	INTERNAL	
⊞ ConnectionGeometry		
⊟ #440 -> #715 🔳		
Name	1stLevel	
Description	?	
OID	#1094	
GUID	3aw218fWL2chWCvclyrFNT	
Related Buildingele...	IfcWallStandardCase (#715)	
Physical/Virtual	PHYSICAL	
Internal/External	EXTERNAL	
⊞ ConnectionGeometry		
⊞ #440 -> #681 🔳		

Space boundaries: None

Space boundaries: Level 1

Level 2 exports boundary surfaces together with all data required for energy or thermal calculations. In this case, the boundary surfaces are influenced by the adjacent surfaces and their attributes, such as materiality. For energy calculations, boundary surfaces are associated with the building's geometry. If the project is set up with phases, the desired phase for the export must be selected in the energy settings for the project:
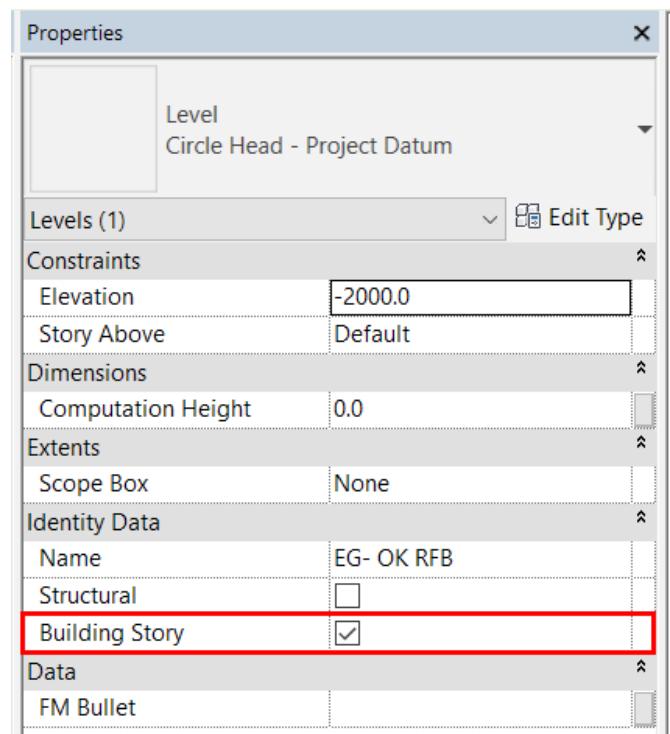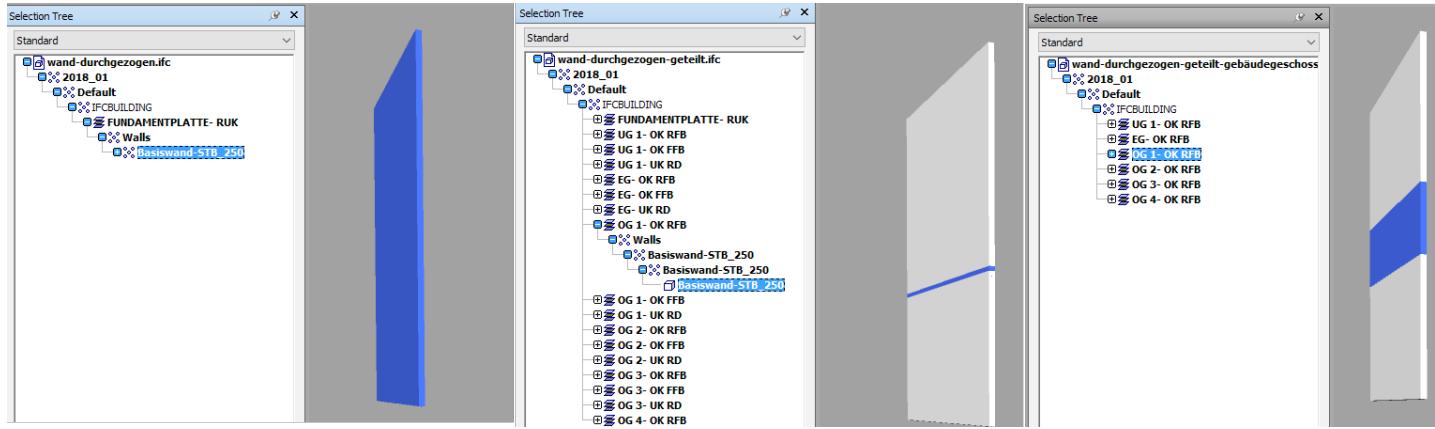
Project Origin
Chose the origin of the exported file. There are 4 options:
- Current shared coordinates origin
- Internal Revit coordinates
- Project Base Point
- Site Surve Point

Split walls, supports, and air ducts by level divides these elements if they have been modeled over several floors. They are divided according to the floors of the building. This setting can be determined for each floor in the Revit properties.

This view shows how important it is to define the right levels as floors, as otherwise the IFC structure becomes too confusing and the elements are not separated in an optimal way. Ideally, a project will have one level for each floor. There are certain cases when the next level is not the desired building story, like in split level houses. For such cases the level parameter "Story above" can be used to explicitly specify the next story.

File Header information/project address allows general project information delivered with the IFC file to be customized.

File information can be viewed with a text editor and, in addition to the optional information, automatically provides information about the original software, the IFC exporter, and the IFC schema.

This information is mainly relevant to IFC exports for a CAFM platform in COBie format. We recommend using the COBie extension for Revit for this purpose. It is available at http://www.biminteroperabilitytools.com/.
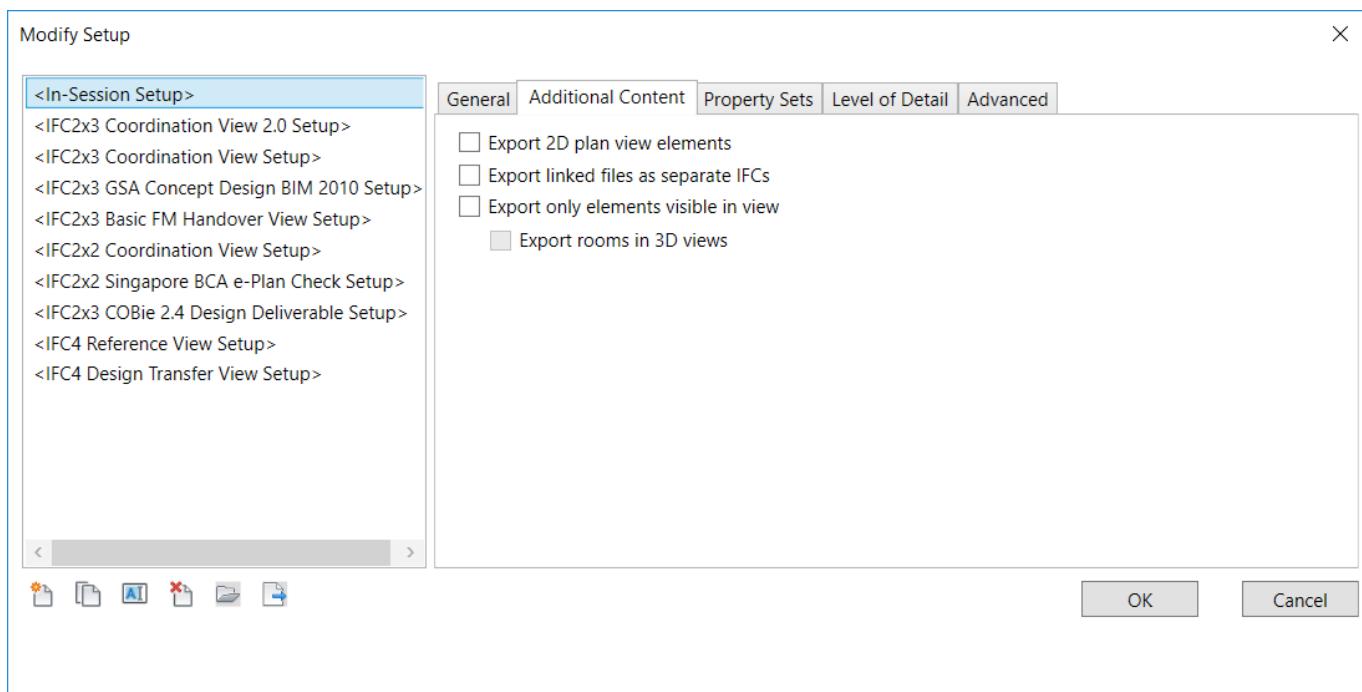
The project information is partly based on the location of the building (via address details).

The information in the IFC export dialog can be used to supplement or overwrite these data.

5.2.2 Additional content

The following additional settings can be found in the tab for Advanced IFC export settings:

Export plan view elements enables the export of some 2D elements such as grids, text, and lines. It is vital to ensure that the correct classes are used, such as IfcAnnotation or IfcGrid for grids. However, not all IFC viewers support the display of these classes. 2D support is restricted because the IFC format was designed to export BIM data, i.e. the 3D geometry, including the associated information. As such, it is not possible to export plan views.
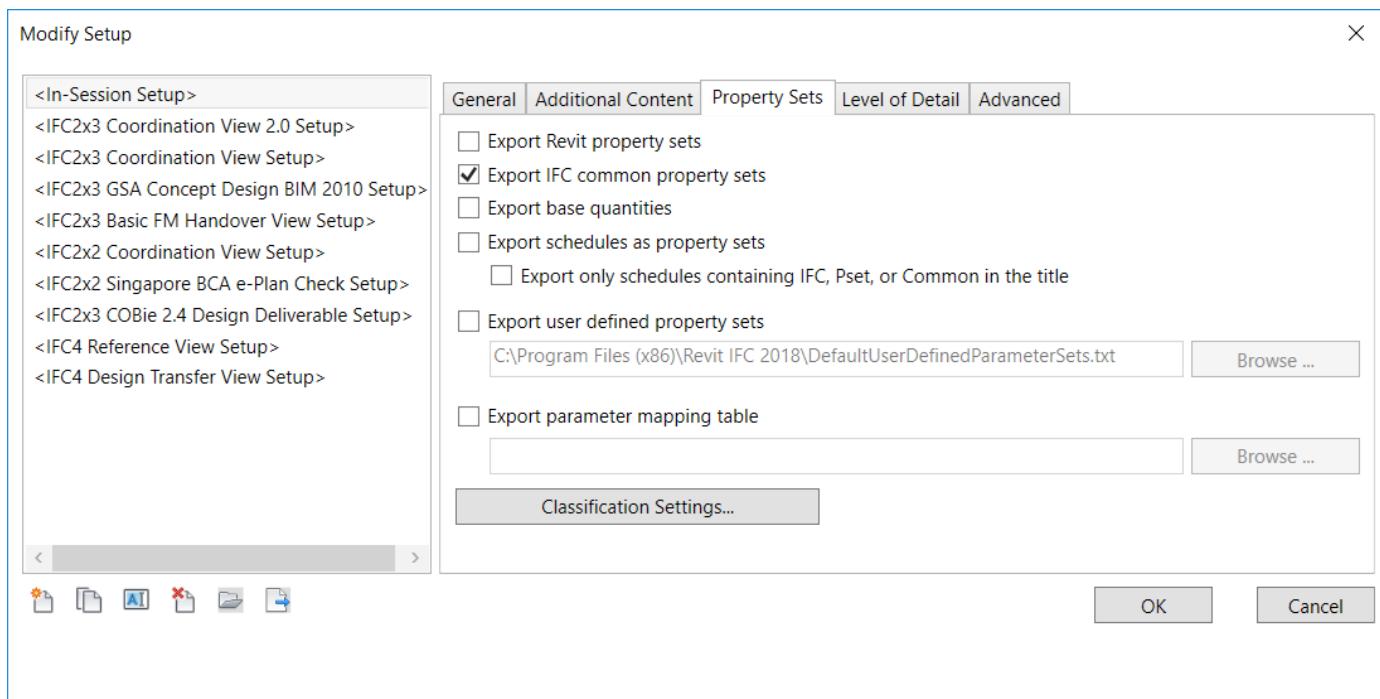
Export linked files as separate IFCs exports the Revit files that are currently linked within the project as separate IFC files. If this function remains deactivated, the Revit links are not exported.

Export only elements visible in view includes only elements that are visible in the current view due to visibility settings, filters, and phases.

Export spaces in 3D views generates IFC spaces and 3D volumes that can then be selected in an IFC viewer.

5.2.3 Property sets

The **property sets** tab in the advanced IFC export settings can be used to access various other key settings:

Export Revit property sets allows you to export all the properties of a component. Although it might seem desirable at first glance, this function is not recommended for exchanging specialist IFC models. It augments the data model with lots of unnecessary information and thus has a negative impact on the file size. A data model exported via this option can be up to 70% bigger than one exported via another method.

Export IFC common property sets includes the default properties defined in the IFC schema. This option should always be activated.

Export base quantities provides base quantities as a basis for determining quantities and creating simulations. During export, all elements are assigned as "base quantities" (fixed property sets defined by buildingSMART). For instance, a wall might look like this:

Export schedules as property sets enables the targeted export of properties that are defined in the schedules. As a Revit project usually contains lots of schedules, this option may be restricted to certain component lists by using only schedules with "IFC", "Pset" or "default"' in their name:

Revit schedule and the resulting properties in the IFC file

Export user defined property sets is another way of exporting specific selected properties. The parameters to be exported can also be specified in a text file. When Revit is installed, the default file is stored at this location:

C:\ProgramData\Autodesk\ApplicationPlugins\IFC2018.bundle\Contents\2018\
DefaultUserDefinedParameterSets.txt

It serves as the basis for the individual data sheet and is structured as follows:

```
# 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>
#       ...
```

As a comparison, a completely configured data sheet might look like this:

```
#
PropertySet:     Autodesk Parameter      I               IfcWall
                 Phase                   Text            Phase Created
                 Space Boundary          Boolean
                 Structural              Boolean
```

Please note:

· A data sheet begins with a hash key. This allows several data sheets to be specified in a text file.

· Parameters are separated using the tab key.

· The name of the required data sheet follows a tab after the name PropertySet:; in the example above, this is specified as the "Autodesk parameter".

· Specifying "I" for instance or "T" for type determines whether this is a type or instance parameter.

· The IFC classes of the elements to which these properties are assigned are then specified. This example applies only to walls, so "IfcWall" is used.

· In the list that follows, the Revit parameters are shown on the left, followed by the IFC data type and the desired name of the IFC attribute, each separated with a tab. Note that calculated values such as the unlinked height of a wall cannot be transferred to the data sheet.

In this example, the following instance properties of a wall are transferred: the phase, the space boundaries, and the supporting structure. It is important to specify the correct data type—normally "text," "Boolean" for yes/no parameters, "area" or "length." If the IFC parameter does not have the same name as the Revit parameter, the latter can be specified at the end of the line.

Export parameter mapping table allows the overwriting or extended assignment of certain parameters that are already defined in the IFC scheme, which are set out in the chapter on default attributes.

The following example shows how custom properties can be mapped to standard parameters "Compartmentation" or "Combustible".

These can be defined in the mapping file as follows:

| Pset_WallCommon | Compartmentation | CustomParameter1 |
| Pset_WallCommon | Combustible | CustomParameter2 |
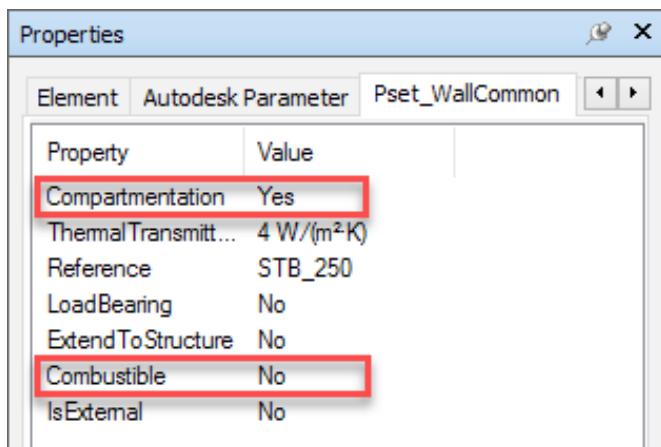
The definition follows this principle:
**IFC_Common_PropertySet_Name<tab>IFC_Property_Name<tab>Revit_Property_Name**
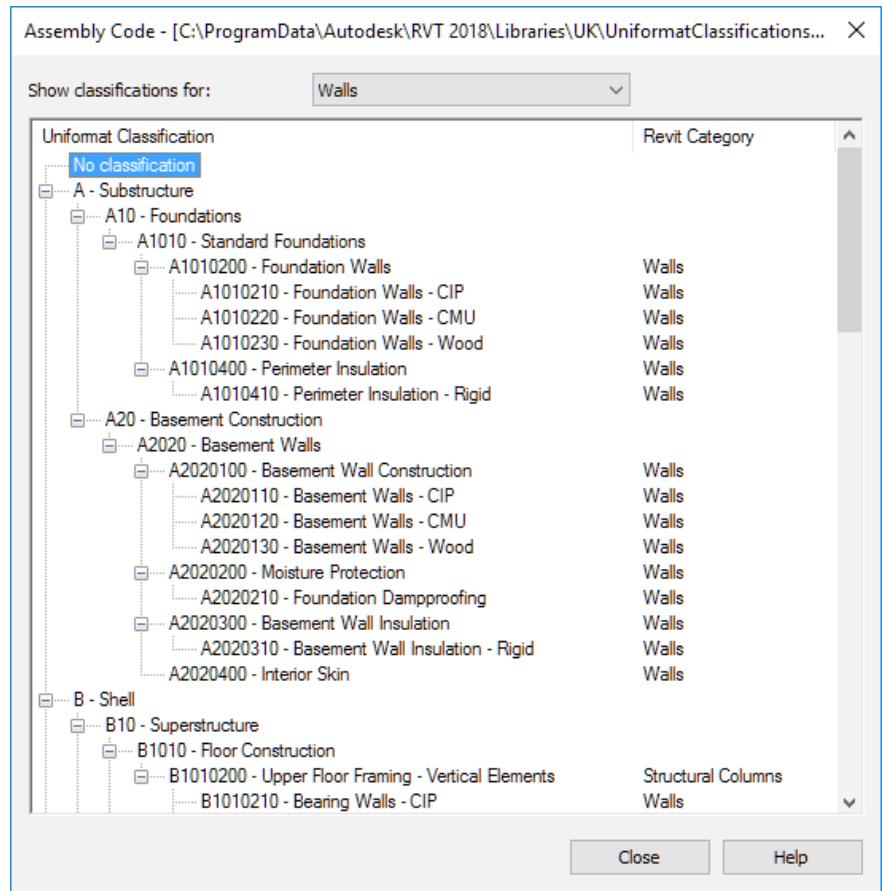
The supported property sets are defined in detail in the buildingSMART documentation online. A current list can be found at https://autode.sk/IFClinks.

Classification settings allow you to specify the uniformity classification used in the project according to a country-specific system. In the UK, for instance, the Uniclass system is established as the means of classification, and
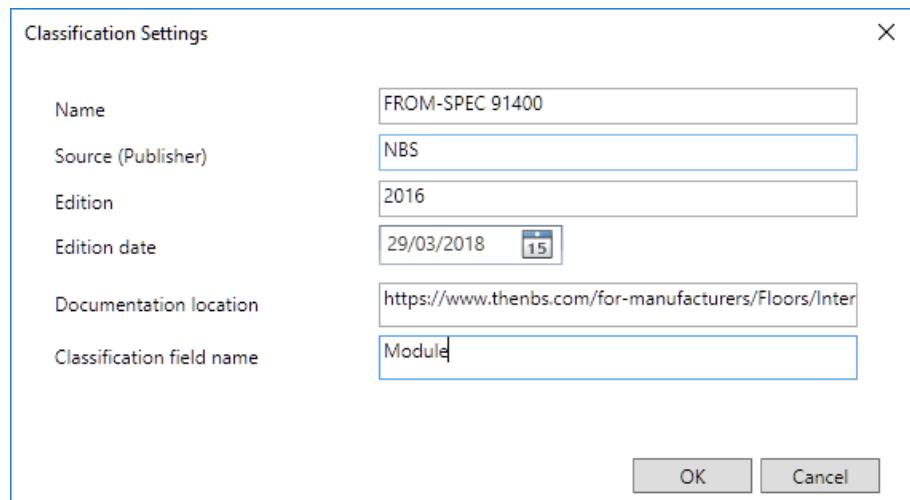
is delivered with Revit. In this case, the building information modeling comes with unique key numbers for component properties, allowing machine processability and data linking.

Revit affords the opportunity to use the standardized uniformity classification of components or a custom classification file. This is usually done by assigning the type property "assembly identifier". This field allows you to select a pre-defined value from the classification file, which can be found in text format here: C:\ProgramData\Autodesk\Libraries\<your country>\UniformatClassifications.txt
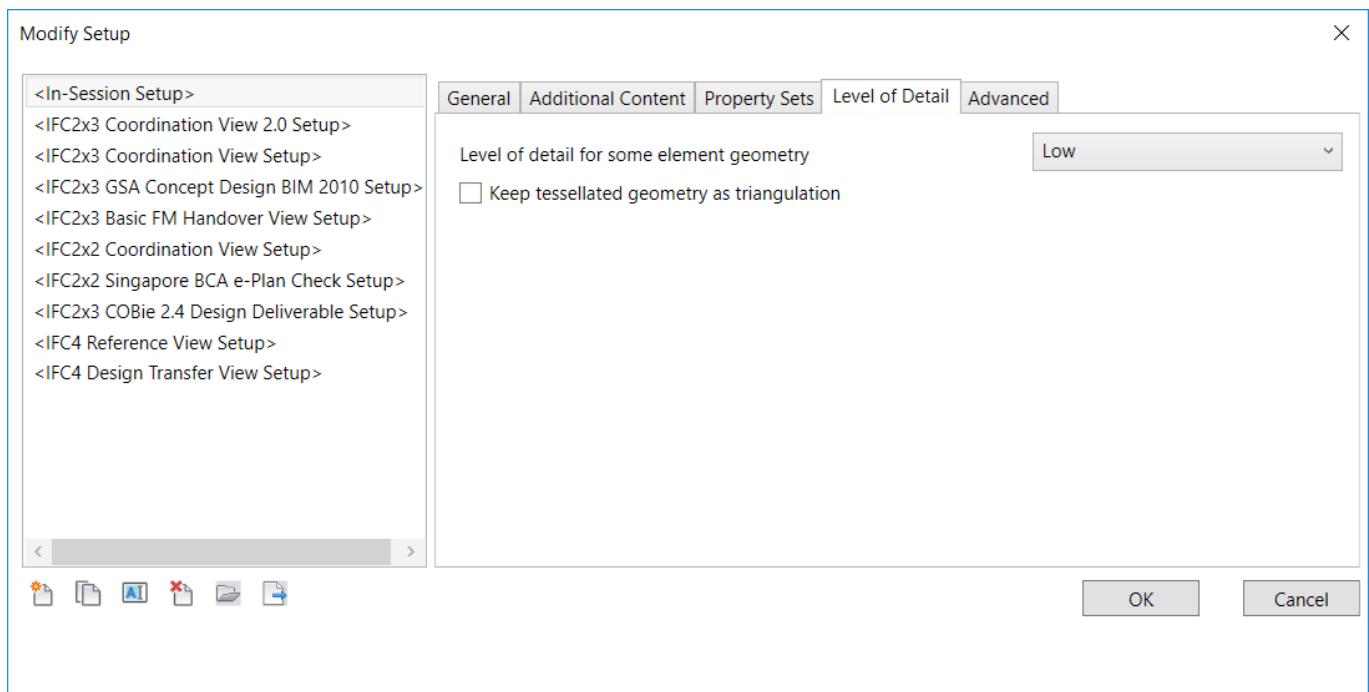
This file can be adapted in line with the aforementioned local classification system. Information about Autodesk's current modifications to these files can be found on the BIM blog: https://autode.sk/IFClinks.

The specification in the IFC export dialog is merely information about which classification system is being used, and does not affect the actual content of the model:

5.2.4 Level of detail

The **Level of detail** tab in the IFC export settings can be used to select the following options:
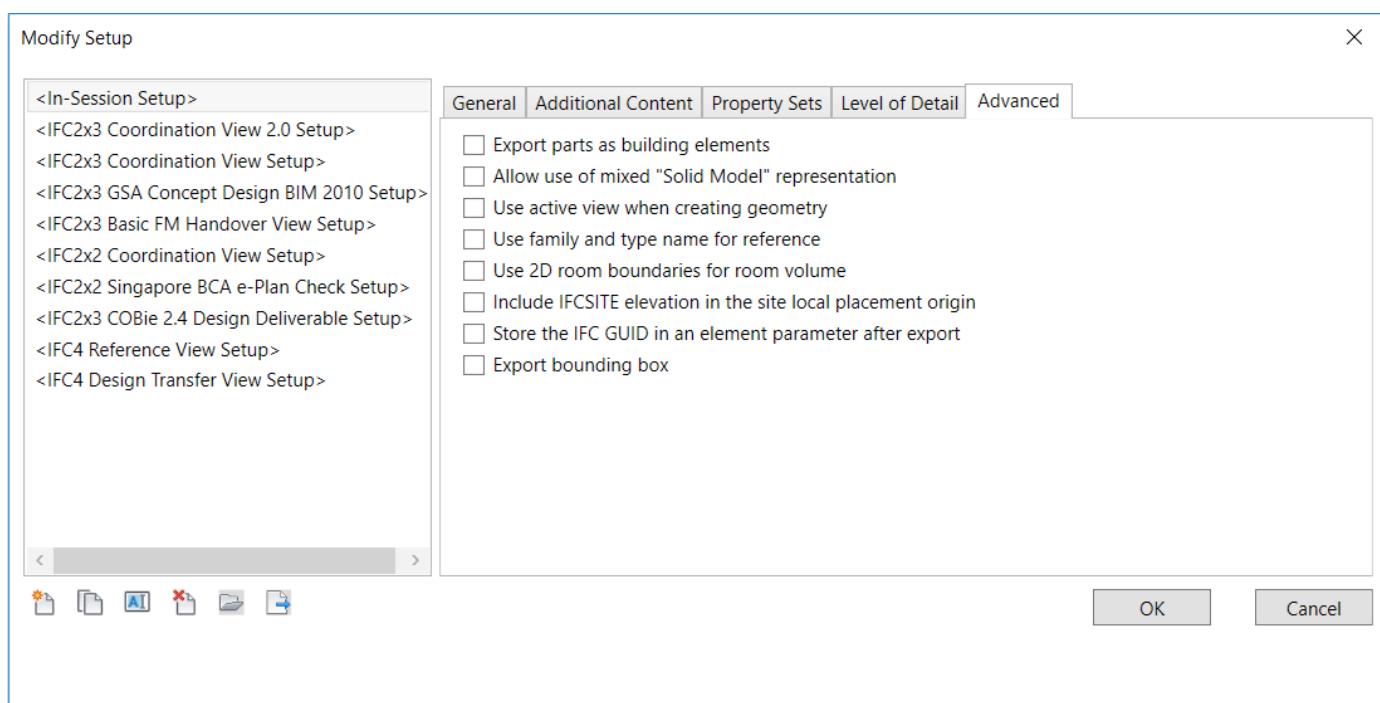
Level of detail for certain geometric elements enables you to set the level of detail. This has a significant impact on the file size and correct interpretation.

Components should be exported with a high level of geometric detail only if needed, as it can cause data bloat. The Detail level "low" is usually sufficient.
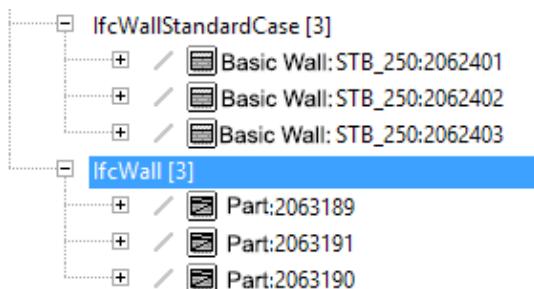
5.2.5 Advanced settings

The last tab, Advanced, allows you to select the following additional options:

Export parts as building elements is relevant to IFC data exchange when working with partial elements in the construction of walls or floor slabs. Partial elements are exported as

"IfcBuildingElementPart" by default. This allows individual parts to be assigned to a higher-level element within the IFC data model:
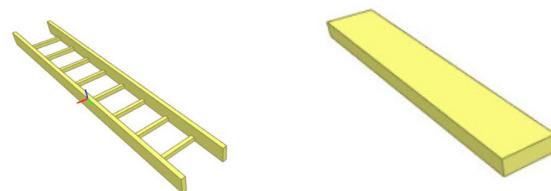

Example of a three-layered wall, exported as partial elements in the FZK viewer

However, some BIM applications cannot interpret these special elements correctly, so that these are displayed in the IFC data model as separate wall elements without being assigned to a higher level.

Allow use of mixed solid model representation enables the export of combined swept solid and B-rep models. A geometric object in an IFC data model is normally generated from either one or several swept solid objects, or from B-rep objects alone. The combination of these two types of representation is not enabled by default in the IFC schema. For more complex components, in particular, this leads either to a larger file size or incorrect presentation, as elements are wholly represented B-rep objects. Solid model representation combines the two types of representation within a single class,

which can mean better geometric results at a smaller file size for complex models. It should be noted, however, that the IFC file exported using this setting no longer complies with the default IFC schema and must therefore be accepted as such by all those involved in the project. For certain areas of use, it may be necessary to have an unaltered default schema for export.

Use active view when creating geometry incorporates the display settings for the current view into the IFC export and was specially developed for building equipment elements such as cable routes and built-in parts whose model geometry differs from the represented geometry:

Fine representation vs. intermediate representation

Use family and type names for reference enables referencing based on Revit family and type. The default setting is for referencing of a component based on the types used.
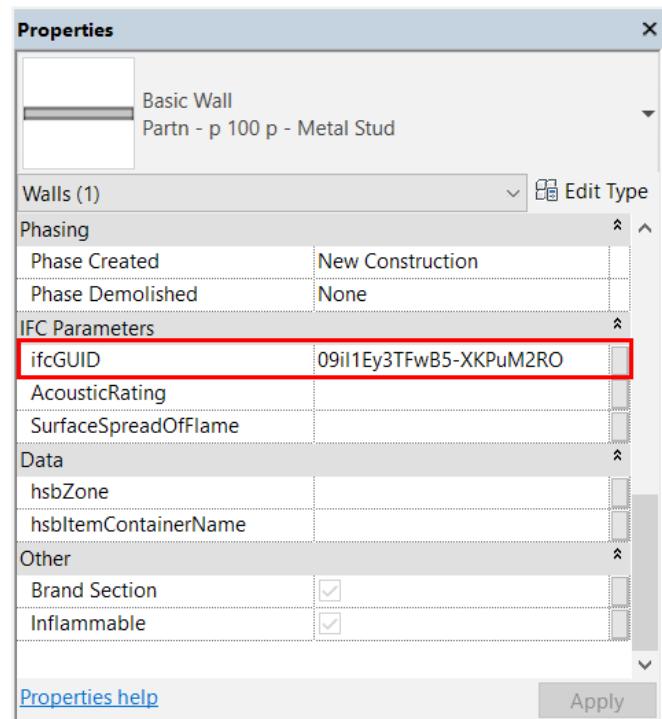
Use 2D room boundaries for room volume simplifies the calculation of the room volume based on two-dimensional spatial boundaries. The space geometry from Revit is used to determine the volume in the IFC scheme by default.

Include IFCSITE elevation in the site local placement origin: when exporting information about the site, the area (Ifc site) has a height value for the project. In IFC2x3 CV2.0, this value is set to "0" by default, which may not be interpreted correctly by older applications. The corresponding value is also provided with the help of this export setting.

Store the IFC-GUID in an element parameter after export stores the generated IFC-GUIDs in the "IfcGUID" parameter after they have been successfully exported. This simplifies subsequent specialist model coordination, as the components are clearly identifiable.

Export bounding box. Every geometrical element can also be represented in a simplified manner using a bounding box. If an object cannot be exported due to its complex geometry or needs to be simplified to better determine clearances, the bounding box can serve as an alternative to the representation itself or allow the object to be represented altogether.

5.3 Other settings

The class assignments made in the IFC export settings are formulated as defaults and form the basis for IFC export, with one IFC class assigned to each Revit category.

In some cases, however, a finer division may be necessary, so that components are assigned to different IFC classes within a Revit category, often using the "Generic model" category.

By using export parameters, these components can be assigned to specific IFC classes and types regardless of the default settings in the mapping table.
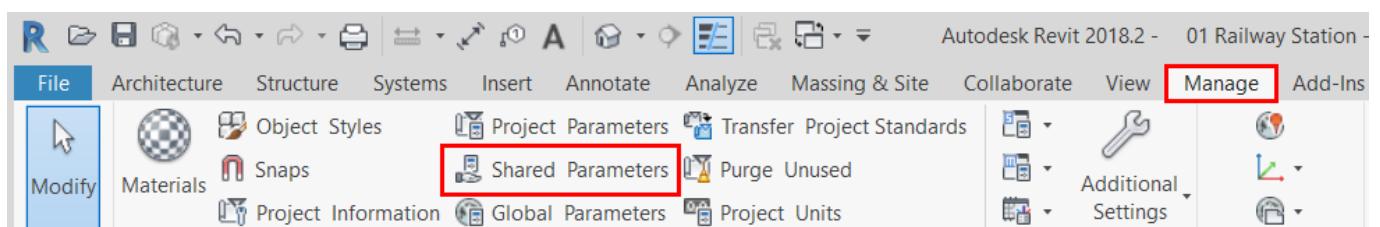
IFC export parameters

The following export parameters may be created:
- IfcExportAs
- IfcExportType
- IfcName
- IfcDescription
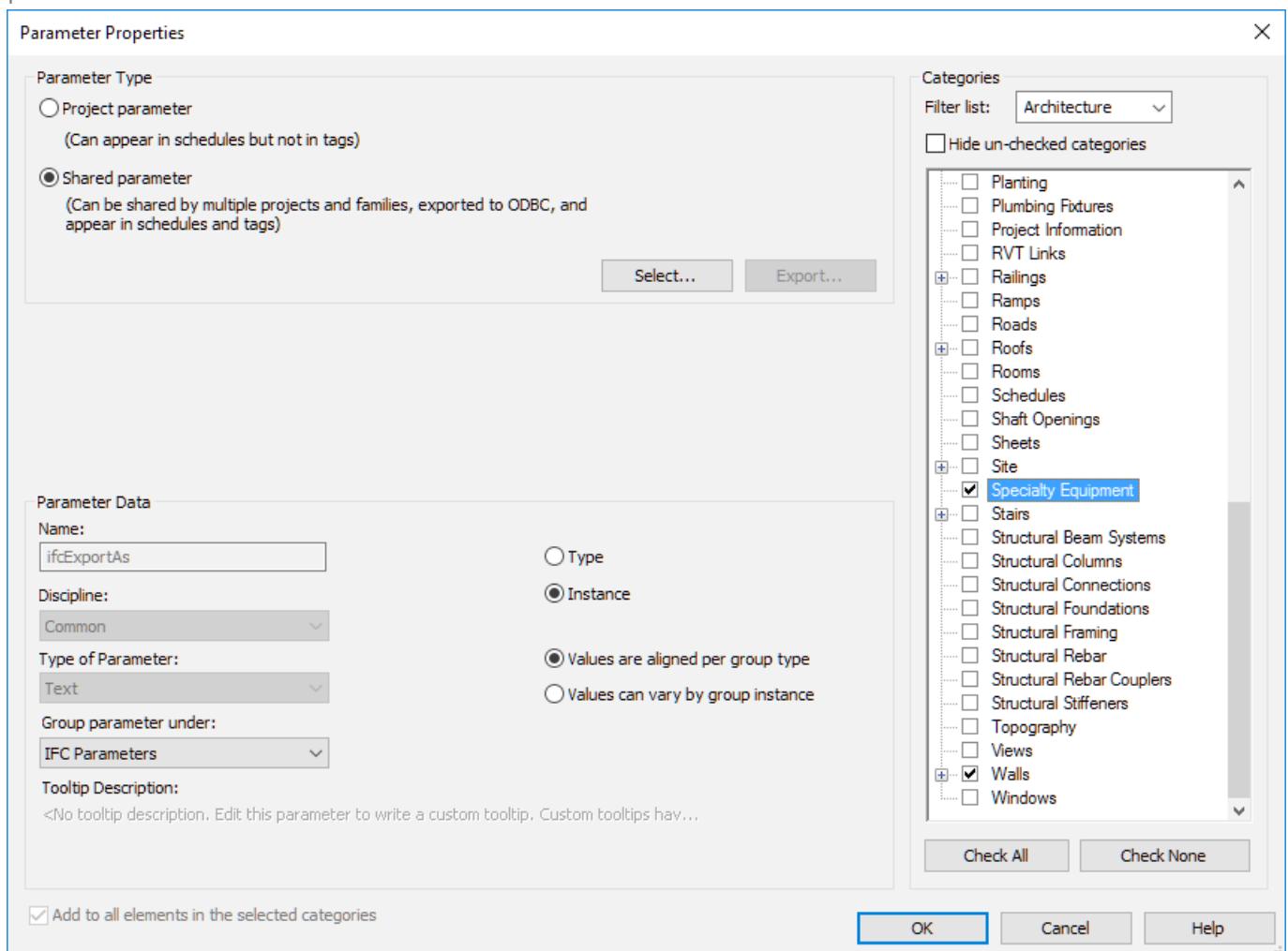
- IfcObjectType
- IfcLongName

It is best to set up these parameters as "shared parameters" for project and family files. Autodesk provides a central IFC shared parameters text file, which can be downloaded here: https://autode.sk/IFClinks.

According to the hierarchy, export parameters are above the default settings of the mapping table in the IFC export settings and thus overwrite the default settings.

The parameter names are defined in the IFC format and are only taken into account if the spelling is correct. IFC export parameters should be assigned when they are integrated into the group "IFC parameters."

The parameters can be created as type or instance parameters as required, depending on the office standard. In general, it is recommended that you use type parameters, as these data rarely vary between individual instances of the same type.
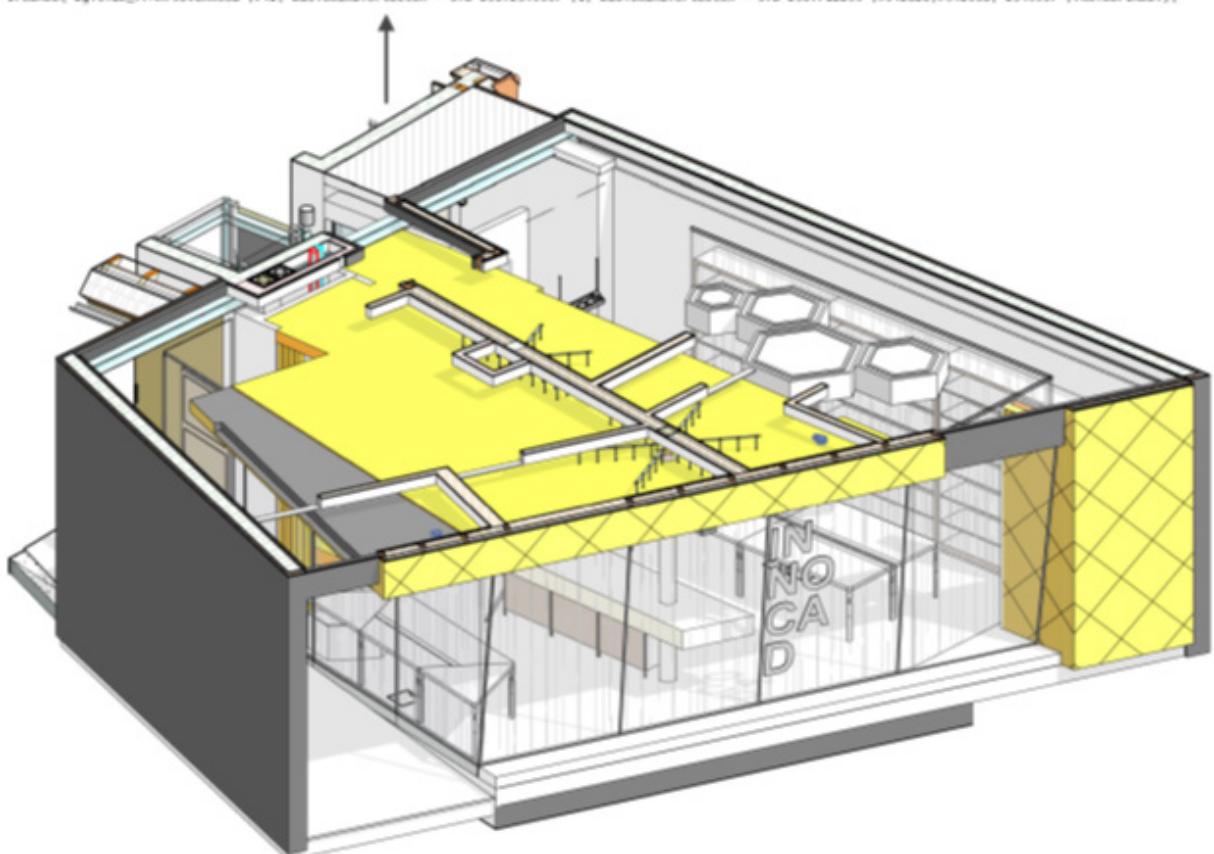
The "IfcExportAs" parameter overwrites the default IFC class of a Revit component for the IFC export, e.g. IfcSlab for slabs. The parameter value "DontExport," on the other hand, prevents the creation of a component type when exporting the data model.

The "IfcExportType" parameter overwrites the default type for the IFC export. This parameter is not used often because the type can also be

defined using the IfcExportAs parameter, where the type is separated after the class with a colon, e.g.. "IfcSlab.ROOF" for a slab of the roof type.

In certain cases it can be useful to override certain information during the export, such as IfcName, IfcDescription, IfcObjectType, IfcLongName. While these are the preferred names for the overrides, Revit currently also supports NameOverride, DescriptionOverride, ObjectTypeOverride, LongNameOverride. In terms of its actual function these parameters should not be considered as export parameters, but they have the ability to overwrite the default values such as type and generate the associated property data sheets.

Revit K Category	Default IFC class	Predefined Type	Alternative classes (IfcExportAs)	Alternative types (IfcExportType)
Wall	*IfcWallStandardCase*	UNDEFINED	*IfcFooting*	PAD_FOOTING PILE_CAP STRIP_FOOTING FOOTING_BEAM
	IfcWall	STANDARD	*IfcFooting*	
Slab	*IfcSlab*	FLOOR		FLOOR ROOF LANDING BASESLAB
			IfcFooting	PAD_FOOTING PILE_CAP STRIP_FOOTING FOOTING_BEAM
			IfcCovering	CEILING FLOORING CLADDING ROOFING
			IfcRamp	
Ceiling	*IfcCovering*	-	-	CEILING FLOORING CLADDING ROOFING
Ramp	*IfcRamp*	-	-	-
Stairs	*IfcStair*	-	-	-

Behavior of project families

A family created within the project can be assigned to any class officially supported by Autodesk using the IfcExportAs parameter.

Behavior of loadable families

Essentially, loadable families behave like project families and can be assigned to any supported class. It is also possible to assign nested families to different classes and types of IFC export. It is important to ensure that the "shared" option is activated in the properties of the individual families.
Voids are also automatically assigned to the OpeningElement class.
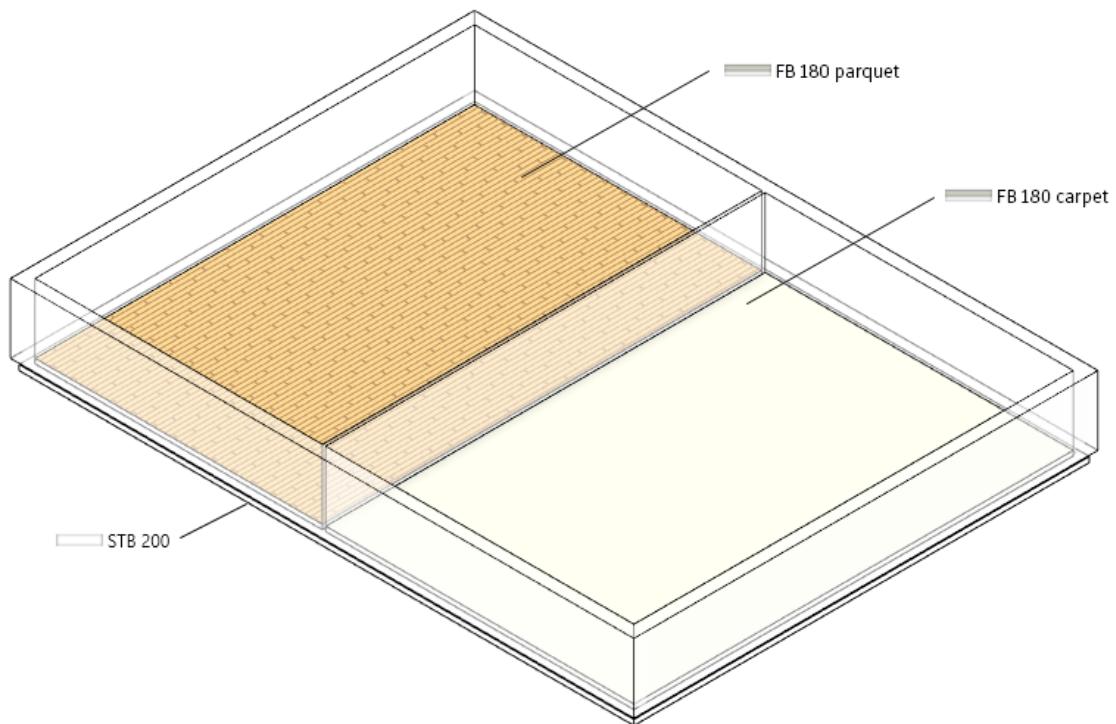
When creating a project family as a void, this is also automatically assigned to the corresponding OpeningElement class.

6 USECASES

6.1 Floor slab construction
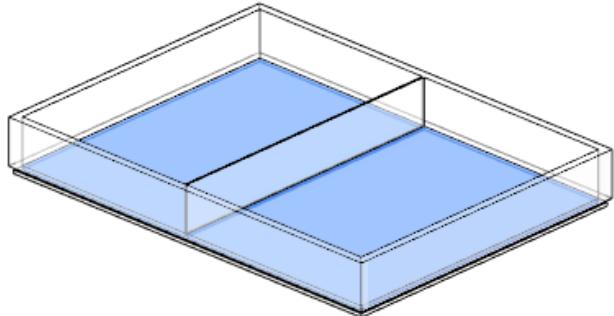
Floor slabs are often modeled using two separate elements: the supporting floor slab is modeled before the floor structure is added as a multi-layer floor for each room:

For IFC export, all floor slabs are assigned to the IfcSlab class. This may prove a hindrance to further planning or assignment in AVA or calculation software. Instead, it makes sense to export the floor not as IfcSlab, but as IfcCovering, which also assigns it the appropriate attributes, such as combustibility or surface finish.

As such, for both floor constructions, the IfcExportAs parameter is specified as "IfcCovering. FLOORING," which assigns the elements to the IfcCovering class and the FLOORING type on export:

This assignment gives the floor construction the right class/type and the properties defined in the Pset_CoveringCommon, which facilitates further evaluations:
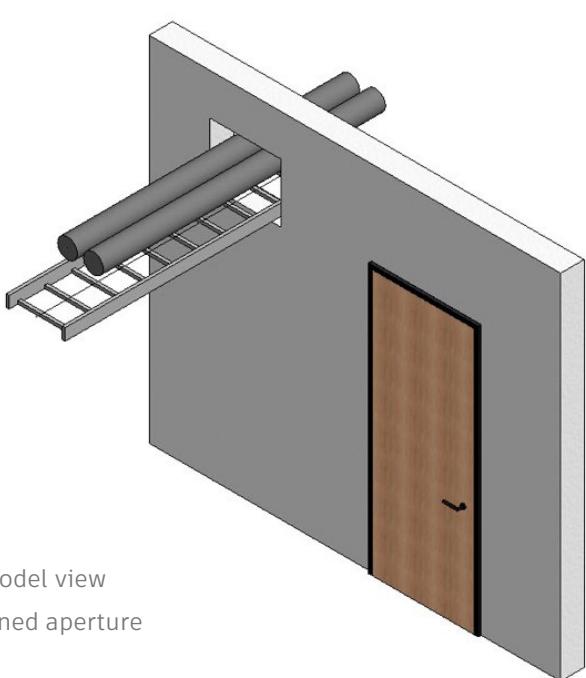

6.2 Cut openings

The use of placeholders has largely been established in the planning and coordination of cut openings in the BIM process, as "provision for void" objects. These can be exchanged between specialist models, including all the necessary information and dimensions, and can also be used for approval processes and the eventual creation of the aperture itself.
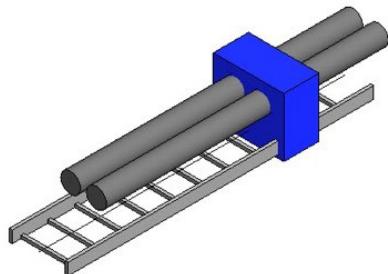
These may be based either on aperture elements from the Revit library or a simple family with a void.

Optionally, the family may also include another solid form extrusion, which is controlled using a visibility parameter and has the same dimensions as the void. With the help of this second solid, a specialist model may be created separate from the overall model as an IFC file, specifically for the purposes of aperture planning.

Using a component filter and a coordinated 3D export view allows you to visualize and export the provision for void solid as a specialist model.

Overall model view
with planned aperture

3D view with view filter and superimposed
aperture solids

In addition, 2D breakthrough symbols for sectional and plan views can also be added to this family.
For IFC export, the aperture family is given the following data:

Revit component	IfcExportAs	IfcObjectTypeOverride
Void	IfcBuildingElementProxy	PROVISIONFORVOID
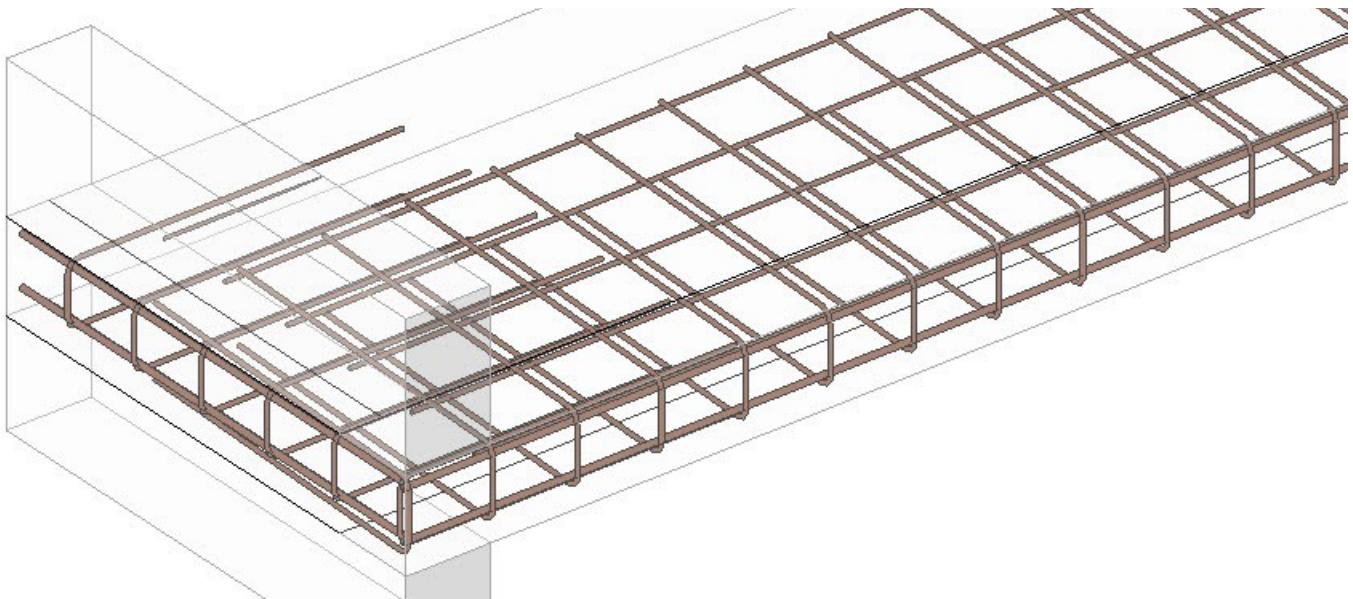
By assigning the type, the aperture is provided with all the necessary information:

Pset_ProvisionForVoid	
Depth	0,3
Height	0,5
Shape	Rectangle
System	
Width	0,6

6.3 Assigning assemblies

Assemblies are important for the higher-level grouping of components and are often used for truss fields, beam grids and reinforcement cages. Unlike the grouped elements in Revit, the assemblies are also transferred during IFC export and can have higher-level properties.

The reinforced bars and brackets of a floor slab are assigned to an assembly in this example:

The IFC parameters are overwritten in the process:

Revit component	IfcExportAs	IfcObjectTypeOverride
Reinforcement	IfcElementAssembly	REINFORCEMENT_UNIT

In the IFC data model, assigning this class and applying the instance parameter "IfcObjectTypeOverride'" defines the predefined type as "REINFORCEMENT_UNIT" and aggregates the higher-level assembly as "IfcElementAssembly."
The individual assembly elements can still be selected separately. This assignment provides a better structure within the IFC model for the evaluation and classification of components.

6.4 Assigning default attributes

BuildingSMART provides appropriate information about default attributes as part of its online documentation. Under the term "Pset_CoveringCommon," for instance, you can find all the default attributes of the entity class "IfcCovering."

For a list of the property data sheets for architectural components of the IFC4 schema with multilingual explanations, see
http://www.buildingsmart-tech.org/ifc/IFC2x4/rc3/html/schema/ifcsharedbldgelements/pset/

The attributes made available are determined by the choice of class or type.
As an example, "Pset_CoveringCommon" has been specified here, so that all elements of the IfcCovering class are automatically assigned:

| IFC-Attribute	Revit parameter
Reference	Component type
Status	Construction phase for the element
AcousticRating	Sound insulation class
FireRating	Fire-resistance class
Combustible	Combustible material
SurfaceSpreadOfFlame	Fire behavior
ThermalTransmittance	U-value
IsExternal	Exterior component
LoadBearing	Load-bearing component
Compartmentation	Fire compartment-defining component
FlammabilityRating	Flammability rating
FragilityRating	Fragility class
Finish	Surface finish
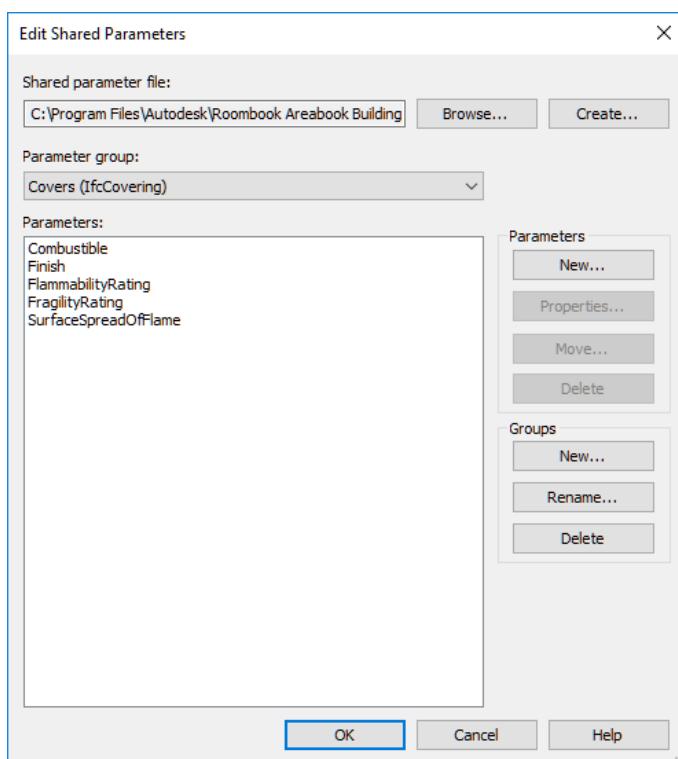
When exporting, please note that only Revit parameters that exist in the project and show the correct unit and a valid value are included. "Empty" parameters are not exported.

Creating selected attributes in Revit projects

Due to the amount of data, it does not make sense to provide all the available attributes in a single template. Adding the necessary attributes as required is more practical.

Autodesk provides its own IFC shared parameters file, which already contains all the available parameters. For the link to this file, see
https://autode.sk/IFClinks.
The correct units for the parameters are predefined in this file. The parameters can be defined either as type or as instance parameters. Ultimately, this depends on the specific company standard and way of working.

As a result, the newly created attributes are assigned to the corresponding "Pset" during export. The following comparison shows the basic contents and the contents after the corresponding attributes have been assigned:

Property	Value
Reference	Floor 10.0 - Tiles 25 x 25
TotalThickness	0.1

Property	Value
Reference	Floor 10.0 - Tiles 25 x 25
FireRating	F60
FlammabilityRating	B1
Combustible	False
Finish	R 13
TotalThickness	0.1

Basic contents Pset_CoveringCommon

Contents after assignment Pset_CoveringCommon

6.5 Structuring the IFC data model

When exchanging models, not all the information generating in the course of the planning process is relevant. Often, many reference levels are used, for instance, that are ultimately irrelevant to the model transfer. As such, only certain levels tend to be identified and exported as story levels.

The export of story levels is influenced by the corresponding "Story level" parameter in the Revit properties for a level. All components on other levels are assigned to the nearest story level. Ideally, only one defined story level should exist for each story of the building.

In the next example, the shared parameter "IfcName" is used to rename the levels when they are exported. This also overwrites the predefined name for most of the other Revit elements.

The "IfcName" parameter is created as a text parameter and can be assigned to the desired Revit categories as a project parameter

(instance or type parameter). The parameter "Ifc-Description" can also be created in the same way. This attribute can be used to append additional information to an element.

Such customization results in the renaming and additional description of the corresponding level:

Property	Value
Model	Golden Nugget
Prefix	
Name	Level 1
Description	
Story Number	14
GUID	23fwaNhKz4ZgpqN_uBUq4m

Default name after IFC export

Property	Value
Model	Level Description
Prefix	
Name	Level 1
Description	Further Information about this level
Story Number	18
GUID	23fwaNhKz4ZgpqN_uBUq55

Level name with additional notes

If required, the building floor can be assigned other attributes in a similar way to other components. The basis for the assignment of default attributes is the "Pset_BuildingStoreyCommon."

6.6 Usage groups in the IFC data model
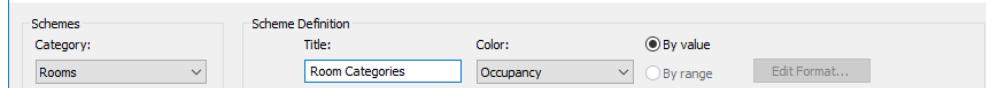
Usage groups can also be stored within the IFC data model. Based on an export schema generated in Autodesk Revit, these groupings can be passed on to those involved in the project for further use.

The export of usage groups and zones is based on the parameter "ZoneName." This is assigned as a shared parameter (type: text) of the Revit category "Rooms." Only one instance parameter can be used for this category.

Using this parameter, you can then generate a color scheme for rooms/zones and create the necessary categories:

Schema definition using the parameter "ZoneName"

In a floor plan, the corresponding categories of the color scheme can now be assigned to the generated rooms:

Schema assignment for the room properties in Autodesk Revit

During the subsequent IFC export, the respective rooms are assigned to the corresponding usage groups.

It is also possible to assign several usage groups to a room. To do so, it is necessary to create additional "ZoneName" parameters and to number these in ascending order ("ZoneName 2, ZoneName 3...").

Stored usage groups in the IFC model

6. SUMMARY

Building Information Modeling (BIM) provides everyone involved in a building design and construction project with new opportunities to make well-informed project decisions, communicate better, optimize workflows, and improve documentation. The data exchange functions provided by openBIM and IFC allow all of those participating in a project to work together, even if they use different software products.

More information about BIM, openBIM, and IFC is available via these links:

https://www.autodesk.com/solutions/bim

https://www.autodesk.com/solutions/bim/hub/bim-interoperability

http://buildingsmart.org/

Autodesk, Inc.
111 McInnis Parkway
San Rafael, CA 94903

https://www.autodesk.com/solutions/bim/
hub/bim-interoperability