

# Revit IFC 手册 2.0



## 简介

BIM（建筑信息模型）是一种基于模型的流程，可帮助建筑师、工程师、客户和承包商采购、设计、建造和管理建筑及基础设施。BIM 本质上是资产的物理特性和功能特性的数字表示。用于创建和修改 BIM 数据的两大关键 Autodesk 工具分别为 Revit（用于建筑）和 Civil 3D（用于大型基础设施）。

如果所有参与设计的人员使用相同的软件工具，那么就可以轻松协作并直接交换数据。用于检查数据质量的协作工具（如 Revit 的互操作性工具）集成到创作软件中，便于随时以用户定义的质量生成报告。此流程称为**原生 BIM 工作流**。

对于大型项目和复杂的团队结构而言，原生 BIM 可能是一项挑战，因为不同的设计任务会使用来自不同供应商的各种创作软件工具。为了便于相关人员跨软件平台实现集成式协作，Autodesk 在 1996 年与其他 11 家行业领军企业携手，共同成立了行业协同联盟 (IAI)。<sup>1</sup> 此联盟开发的一个主要概念便是 IFC（工业基础类）。2005 年，IAI 更名为 buildingSMART。

如今，Autodesk 是 buildingSMART 战略咨询委员会 (SAC) 的成员，SAC 主要面向那些认为全面实施和采用开放式 BIM 对建筑环境领域具有重要战略意义并将 IFC 作为数据互操作性通用标准的领先跨国企业。<sup>2</sup>

此外，Autodesk 于 2020 年加入了开放设计联盟 (ODA)，以期走在互操作性改进的最前沿。<sup>3</sup>

IFC 是通过 **openBIM** 工作流在不同应用之间交换数据的基础，可交换建筑设计、施工、采购、维护和运营等数据，适用于项目团队内部的跨软件应用。buildingSMART 认为，IFC 是对建筑环境（包括建筑物和民用基础设施）的标准化数字描述。这是一项开放的国际标准，独立于供应商或不受其影响，可在各种不同用例中用于各种硬件设备、软件平台和接口。<sup>4</sup>

自 2005 年起，IFC 2x3 被采纳为一项 ISO（国际标准化组织）标准 (ISO 16739:2005)。自 ISO 16730:2017 起，IFC 标准被 CEN（欧洲标准委员会，法语：Comité Européen de Normalisation）采纳，成为一项欧洲标准。由于促进协作是 IFC 的核心宗旨，因此 buildingSMART 为软件产品制定了一个认证计划。<sup>5</sup>

鉴于 BIM 项目的复杂性、各种不同的项目设计交付要求，以及不同软件平台和供应商提供的不同功能，工程建设从业人员和项目团队必须了解 openBIM 工作流的基本原则，这也是本手册的重点讨论内容。本手册主要围绕 **Revit** 的 IFC 功能。其中一个章节还介绍了 AutoCAD 产品的 IFC 功能，并涵盖了适用于基础设施项目的现有和全新 openBIM 标准及功能。

要获取本文所含参考链接的最新清单，请访问 [Autodesk IFC 资源](#)。

1) <https://en.wikipedia.org/wiki/BuildingSMART>

2) <https://www.buildingsmart.org/community/members/strategic/>

3) <https://adsknews.autodesk.com/news/open-design-alliance-membership>

4) <https://technical.buildingsmart.org/standards/ifc/>

5) <https://www.buildingsmart.org/compliance/software-certification/certified-software/>

<b>简介</b>	<b>2</b>	<b>IFC 文件导出选项</b>	<b>18</b>	<b>其他用例与提示</b>	<b>37</b>
IFC 概述	4	IFC 基本结构	18	将楼板导出为 IFC	37
IFC 文件格式	4	IFCPROJECT	18	对底板建模以导出为 IFC	37
IFC 模式版本	4	带 IFCSITE 的 IFCPROJECT	19	剪切洞口	38
模型视图定义 (MVD)	5	IFCBUILDING	20	嵌套族	38
IFC 中的几何表示	8	IFCBUILDINGSTOREY	21	指定部件	38
IFC 查看器	9	使用 IFC 共享参数	21	分区	39
REVIT IFC 开源软件	10	基于图层的软件导出	23		
		IFC 导出设置对话框	23		
<b>在 REVIT 中使用 IFC 文件</b>	<b>11</b>	常规设置	24	<b>附录 40</b>	
常规设置	11	其他内容	27	DYNAMO 和 IFC	40
链接 IFC	11	属性集	27	向 REVIT 添加分类	40
打开 IFC	13	详细程度	31	从基于 AUTOCAD 的产品导出为 IFC	41
				创建和指定 IFC 类	41
				属性、属性数据格式和属性集	42
<b>从 REVIT 导出为 IFC</b>	<b>14</b>	<b>在 REVIT 中使用分类</b>	<b>34</b>		
默认映射	14	分类基础知识	34	<b>IFC 项目的数字质量管理 - TÜV 南德意志集团 TOBIAS SCHMIDT</b>	<b>44</b>
单独映射	15	UNICLASS 2015	34		
AUTODESK CLASSIFICATION MANAGER FOR REVIT	17	OMNICLASS®	35	<b>EIR 与 BEP - PETER KOMPOLSCHEK</b>	<b>50</b>
		使用 AUTODESK CLASSIFICATION MANAGER FOR REVIT 进行分类	35		
		高级分类/多个分类	36		

## IFC 概述

IFC（工业基础类）是一个面向对象的数据模型，用于描述建筑物、制造产品、机械/电气系统的物理构件，以及更为抽象的结构或能源分析模型、成本明细、工作和维护明细表等。

buildingSMART 的官方文档涵盖了上述所有方面，其中还包括软件供应商实施指南，因此对于只需使用 IFC 进行数据交换的工程师和设计师而言，这往往有些难以理解。

使用 IFC 进行数据交换时，必须考虑要使用的版本、模型视图定义 (MVD) 以及文件格式。

为了在 BIM 项目中顺利地进行数据交换，必须遵循由客户/BIM 经理制定的要求。需要注意的是，无法为所有用例创建一个通用的 IFC 文件，但是可以根据具体要求单独创建。这些要求通常在雇主的信息要求 (EIR) 中指定。

IFC 的定义由 buildingSMART 国际组织定期更新和制定。建议设计团队成员在每次协作之初确定所有相关方可使用的 IFC 最新版本。然而，在实际情况允许时，最好使用最新版本。当前的 IFC4 格式具有一系列优势，对复杂几何图形的渲染效果极佳。本手册附录中提供了由 BIM 专业人士撰写的文章，其中介绍了他们对 openBIM 项目的质量管理工作的见解。

## IFC 文件格式

IFC 数据模式以字母数字格式表示，可存储为不同的文件格式。下面列出了 Revit 常用且支持的文件格式：

### .IFC

基于 STEP（产品模型数据交换标准）的标准格式 [EN ISO 10303]。

### .IFCZIP

压缩的 (zip) IFC 文件，文件大小较小，是适用于大多数支持 IFC 的软件应用的导入格式。此格式可解压缩为原始的 .IFC 文件，也可通过压缩来手动创建。

### .IFCXML

基于 XML 的 IFC 数据表示，是某些计算软件的必需格式。

### .IFCXMLZIP

与 .IFCZIP 一样为压缩文件。<sup>6</sup>

## IFC 模式版本

目前（2021 年 4 月）正在使用以下 IFC 模式版本：

**IFC4:** 最新开发的版本，具有以下特点：

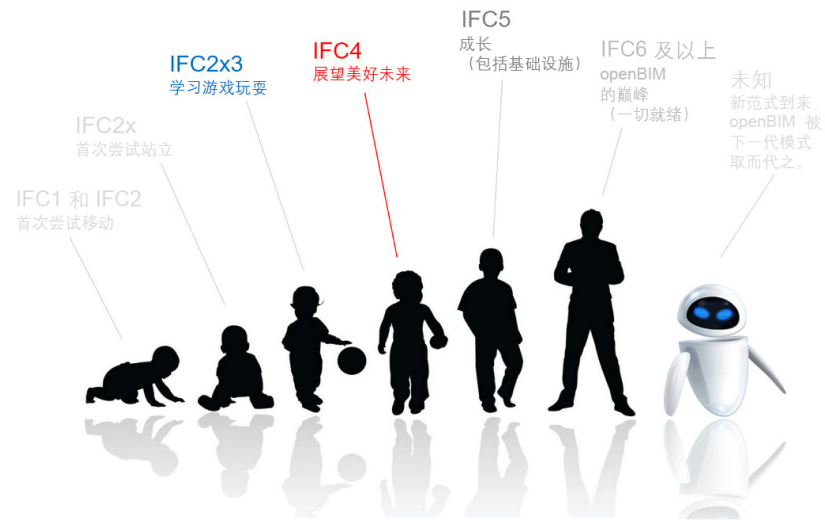
- 效率和模式一致性大幅提升，而文件大小显著减小
- 扩展了对建筑服务元素、结构和分析模型的定义
- 实现了 GIS 坐标系转换
- 支持属性集样板、多语言参照以及与 buildingSMART 数据字典的集成
- 提供了常规几何图形增强功能（拉伸倾斜、任意扫掠、非平面表面、优化镶嵌、纹理和灯光）
- 在设计传递视图中支持非均匀有理 B 样条 (NURBS) 表示
- 单点发布 (4.x) 已在进行中，其中包括基础设施（桥梁、铁路、道路、港口和水路）的增强功能和新类

注意：Revit 已通过 IFC4 认证，但并非所有软件工具均完全支持 **IFC4**。**IFC2x3** 目前仍是支持最广、最稳定的格式。

6. EN ISO 是欧洲 ISO 标准的缩写，表示某一 ISO 标准被 CEN 采纳为欧洲标准。

有关所有版本的全面概述以及官方文档的直接链接，请访问以下网址：

<https://technical.buildingsmart.org/standards/ifc/ifc-schema-specifications/>

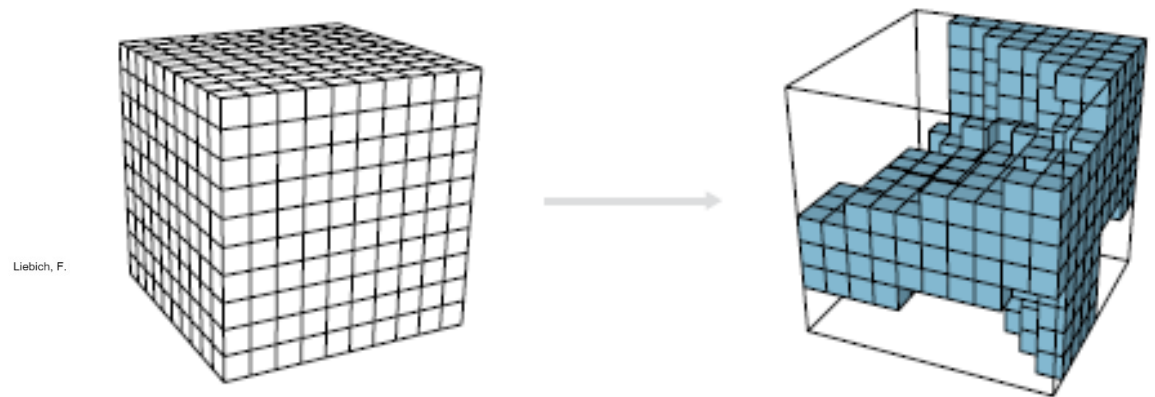


IFC 的演变 - (c) Keentlside/Liebich/Grobler

## 模型视图定义 (MVD)

模型视图定义 (MVD) 是 IFC 数据交换中的一个基本概念。模型视图定义是数据过滤器，可精确定义必须包含在数据交换中的图形和字母数字信息。因此，MVD 是完整 IFC 模式的一个子集。

例如，热仿真需要关于墙体洞口及其材质的信息，结构分析需要关于分析模型的信息，而 FM 系统只需要基本的几何图形，侧重于空间信息和具体的构件特征，如机电系统信息、防火特征以及可用面积。



IFC 模式 (左侧) 与 MVD 子集 (右侧) 的对比 - (c) BIM 经理 Mark Baldwin

buildingSMART 协会目前正在开发上述的 MVD 和 IFC 模式。<sup>7</sup>

MVD 用于检查传入的 IFC 文件是否符合 EIR 和 BEP 中定义的数据要求。它还可以检查要导出为 IFC 的 Revit 文件是否符合质量规范。

“由于范围较大，IFC 未在软件中实施。IFC 是一套广泛的协议；而 MVD 使用 IFC 实体来定义特定用例或工作流的交换标准。目前，软件供应商正在执行这一交换标准 (MVD)。由于软件供应商执行 MVD，因此 MVD 是进行软件认证的基础。软件实施情况将根据 MVD 的要求进行检查。”<sup>8</sup>

以下 MVD 已经过 buildingSMART 认证，广泛用于各种协调工作流：

模式	MVD	说明	Revit 认证
IFC4	参照视图	空间和物理构件的简化几何和关系表示，以参照模型信息，用于在建筑、结构和施工服务（机电）领域之间进行设计协调	建筑参照交换 - 导出 结构参照交换 - 导出  进行中： 机电参照交换 - 导出  建筑参照交换 - 导入
IFC 2x3	协调视图 2.0	空间和物理构件，以用于在建筑、结构和施工服务（机电）领域之间进行设计协调	建筑、结构、机电 - 导出  建筑、结构、机电 - 导入

7. 有关 buildingSMART 开发的 MVD 的完整清单和状态，请访问：<https://technical.buildingsmart.org/standards/ifc/mvd/mvd-database/>

8. <https://technical.buildingsmart.org/standards/ifc/mvd/>

需要注意的是，目前的 IFC 模型视图定义主要支持三维几何图形和属性数据。对于平面图和注释等二维信息的交换，必须使用传统格式，如 DWG 或 PDF 等。

此外，预期用例仅涵盖 BIM 协调软件和查看器中的协调，或作为 Revit 等 BIM 建模软件的参照。不建议导入 IFC 文件进行编辑，这不仅是出于责任考虑，而且存在一定的数据丢失风险。IFC 模式基于 STEP 格式，目前（还）无法有效应对 BIM 建模软件的复杂性和内部相关性。

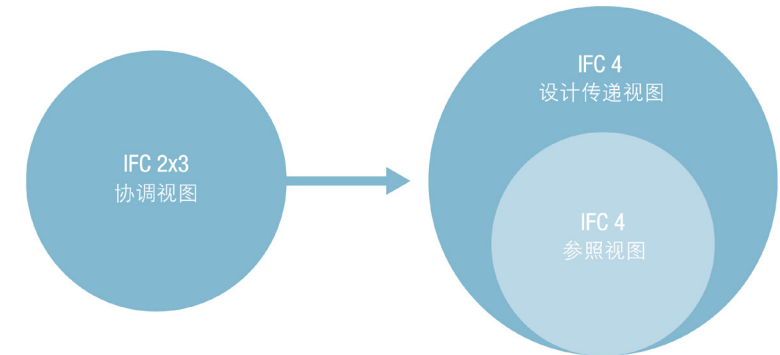
随着 IFC 4 的推出，buildingSMART 已开始向这一方向迈进，目前正在开发一个专用的设计传递视图，旨在实现更好的单向传递，以解决这些问题：

#### IFC4 设计传递视图

空间和物理构件的高级几何和关系表示，使模型信息能够从一个工具传递到另一个工具。这不是双向传递，而是具有更高保真度的单向数据和责任传递。

#### 仍在开发中 -

不属于认证过程的一部分



IFC2x3 协调视图与 IFC4 参照视图的范围对比 - (c) BIM 经理 Mark Baldwin (基于 AEC3 可视化)

使用 buildingSMART 官方文档时，不建议使用主模式文档，而应使用专用的 MVD 文档，可通过以下链接访问此类文档：<https://technical.buildingsmart.org/standards/ifc/mvd/mvd-database/>

这样一来，您可以仅访问所使用的 MVD 中可用的功能，而完整的文档可能包括所使用的 MVD 中不提供的类和属性。

## IFC 中的几何表示

尽管 BIM 和 IFC 涉及大量数据和信息，但几何图形也经常发挥重要作用。因此，了解几何图形的描述方式至关重要，因为这会显著影响 IFC 文件的大小和整体性能。IFC 格式基于 STEP 和实体几何图形，通过以下方法生成：

### 拉伸

最常用、最简单的图形方法，当形状可使用简单的轮廓描述时，大多数情况下都会采用这种方法。

### 扫掠实体

顾名思义，通过扫掠实体的方法使用扫掠来创建图元。使用这一方法时，将沿着一个路径（方向矢量）扫掠轮廓，从而生成实体。由于沿途的旋转或扭曲，轮廓可能会发生变化。Revit 使用这一方法来描述各种无法使用拉伸（钢筋）描述的形状。

### Brep

边界表示 (B-rep) 方法，也称为边界曲面模型。构件的各个曲面由坐标形成，共同代表了实际的实体。因此，即使是最复杂的形状，也可通过几何方式妥善重现。由于 B-rep 对象需要进行复杂的计算才能显示各个曲面，因此需要较大的内存。

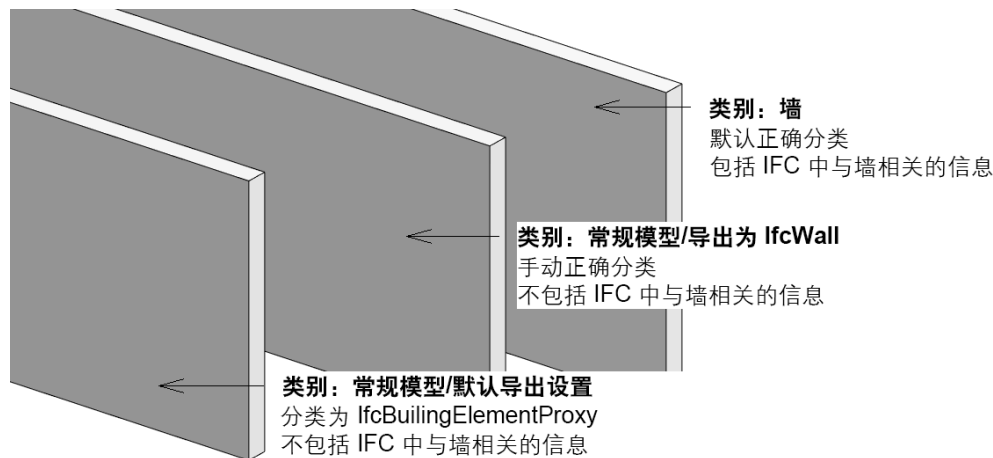
## NURBS (IFC4 的新增功能)

IFC4 可使用 NURBS（非均匀有理 B 样条）曲面来描述复杂曲面。这显著降低了对可用内存的要求，并大幅提高了不规则曲面的质量。

注意：NURBS 在 IFC4 参照视图中不受支持，将成为 IFC4 设计传递视图的一部分。

## IFC 类

每个面向对象的数据模式均基于类（实体）。IFC 模式包含建筑（以及越来越多的基础设施）项目中大多数物理对象的定义，同时还包含整个生命周期中抽象概念（如任务或资源）的定义。



本手册主要围绕 IFC 模式中 with Revit 用户最相关的部分，即物理对象。

对于物理对象而言，IFC 类与 Revit 类别非常相似，两者都定义了每个图元的关系和属性。如果一个建筑图元使用错误的 Revit 类别创建和/或使用错误的 IFC 类导出，它将缺失重要信息。每个图元均定义了与其他元素的关系，并根据所使用的模型视图预定义了属性集，具体视分类而定。

Revit 支持软件中表示的所有主要 IFC 类。当前的支持清单可通过 AKN 上的 Revit 帮助来获取。<sup>9</sup>

9. 支持的 IFC 类 | Revit 产品 2022 | Autodesk Knowledge Network 8.

<https://knowledge.autodesk.com/zh-hans/support/revit/learn-explore/caas/CloudHelp/cloudhelp/2022/CHS/Revit-DocumentPresent/files/GUID-EE6C0CF8-7671-4DCC-B0C7-EEA7513C90A9-htm.html>



除了类，IFC 模式还区分类型（类似于 Revit 中的子类别），并提供进一步的分类。类型记录在 buildingSMART 文档中的“类型枚举”下，并以大写字母书写。例如，IFC4 RV 中的 IfcWall 可以有以下类型：MOVABLE、PARAPET、PARTITIONING、PLUMBINGWALL、SHEAR、STANDARD、ELEMENTEDWALL、USERDEFINED、NOTDEFINED。

IFC4\_ADD2\_TC1 - 4.0.2.1 [Official] © 1996-2020 buildingSMART International Ltd.

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

B. Alphabetical listings

- lfcVertex
- lfcVertexLoop
- lfcVertexPoint
- lfcVibrationIsolator
- lfcVibrationIsolatorType
- lfcVirtualElement
- lfcVirtualGridIntersection
- lfcVoidingFeature
- lfcWall
- lfcWallElementedCase
- lfcWallStandardCase
- lfcWallType
- lfcWasteTerminal
- lfcWasteTerminalType
- lfcWindow
- lfcWindowLiningProperties
- lfcWindowPanelProperties
- lfcWindowStandardCase
- lfcWindowStyle
- lfcWindowType
- lfcWorkCalendar
- lfcWorkControl
- lfcWorkPlan

B.1.1 Defined types  
B.1.2 Enumeration types  
B.1.3 Select types  
B.1.4 **Entities**  
B.1.5 Functions  
B.1.6 Rules  
B.1.7 Property sets  
B.1.8 Quantity sets  
B.1.9 Individual properties

B.2 DE [German]  
B.2.1 Defined types  
B.2.2 Enumeration types  
B.2.3 Select types  
B.2.4 Entities  
B.2.5 Functions  
B.2.6 Rules

B.3 EN [English]  
B.3.1 Defined types  
B.3.2 Enumeration types

▼ Attribute definitions

#	Attribute	Type	Cardinality	Description
1	PredefinedType	EnumTypeDefinition	1	Predefined generic type for a wall that is specified in an enumeration. There may be a property set given specifically for the predefined types. NOTE: The PredefinedType shall only be used, if no InstanceType is assigned, providing to own InstanceType.PredefinedType. IFC4 CHANGE: The attribute has been added at the end of the entity definition.

▼ Enumeration definition

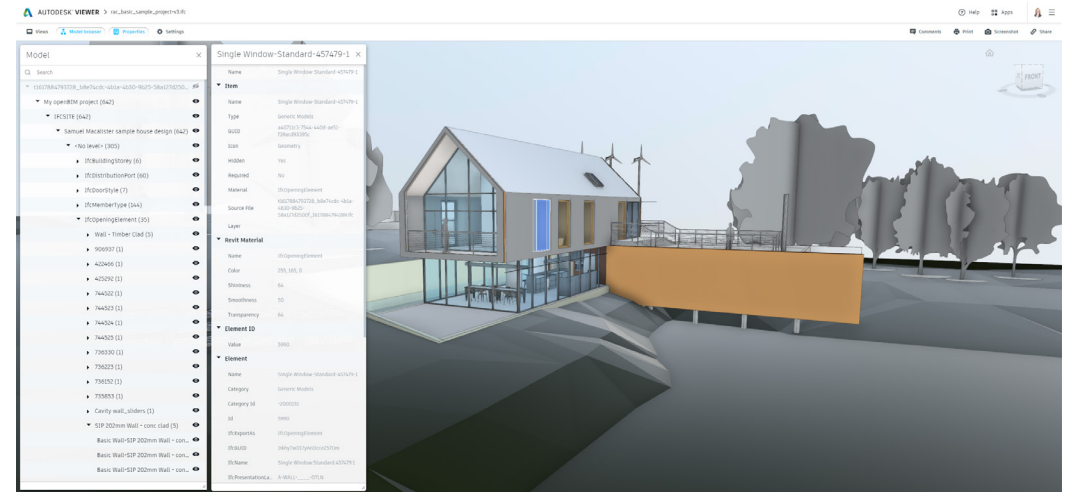
Constant	Description
MOVABLE	A movable wall that is either movable, such as folding wall or a sliding wall, or can be easily removed as a removable partitioning or mounting wall. Movable walls do normally not define space boundaries and often belong to the surrounding system.
PARAPET	A wall-like barrier to protect human occupants from falling, or to prevent the spread of fires. Often designed at the edge of balconies, terraces or roofs.
PARTITIONING	A wall designed to partition spaces that often has a light-weight, sandwich-like construction (e.g. using gypsum board). Partitioning walls are normally non load bearing.
PLUMBINGWALL	A pipe, or enclosure, or enclosure, normally used to enclose plumbing in sanitary rooms. Such walls often do not extend to the ceiling.
SHEAR	A wall designed to resist shear loads. Such shear walls are often designed having a non-rectangular cross section along the wall path. Also called retaining walls or supporting walls they are used to protect against soil pressure behind.
SOLIDWALL	A massive wall construction for the wall core being the single layer or having multiple layers attached. Such walls are often masonry or concrete walls (both cast-in-situ or precast) that are load bearing and fire protecting.
STANDARD	A standard wall, constructed vertically with a constant thickness along the wall path.
POLYGONAL	A polygonal wall, constructed vertically, where the wall thickness varies along the wall path. IFC4 DEPRECATION: The enumeration POLYGONAL is deprecated and shall no longer be used.
ELEMENTEDWALL	A flat wall formed with studs and faced with sheatings, slings, wallboard or plasterwork.
USERDEFINED	User-defined wall element.
NOTDEFINED	Undefined wall element.

## IFC 查看器

共享 IFC 文件之前，验证它是否已正确导出至关重要。此操作通常在 IFC 查看器中完成，不建议为进行验证而在执行导出操作的软件中链接或打开 IFC 文件。有许多免费的 IFC 查看器可供选择：

Autodesk 解决方案：

[viewer.autodesk.com](http://viewer.autodesk.com) (免费的 Autodesk 查看器) 支持 50 多种文件格式，并支持共享和添加注释



**Autodesk Docs**（包含在工程建设软件集中）基于与 Autodesk 查看器相同的技术，但提供了一些扩展的文档和项目管理功能。

**Autodesk Navisworks**（包含在工程建设软件集中）是 Autodesk 的桌面协调解决方案，具有四维/五维仿真和冲突管理等扩展功能。Navisworks 使用 Revit IFC 引擎，后者随 Revit IFC 插件一起更新。

精选第三方 IFC 查看器：

**OpenIfcViewer**：由开放设计联盟 (ODA) 开发，是一款高效、先进的 IFC 查看器，支持包括 IFC 4.3 在内的最新 IFC 版本。

**FZKViewer**：由卡尔斯鲁厄理工学院 (KIT) 开发，支持多种 IFC 版本，包括 IFC 4.3、mvdXML、GML、LandXML、gbXML、e57 等。

**BIMvision**：由 Datacomp 开发，支持包括 IFC 4 在内的 IFC 版本，可通过商业插件进行扩展。

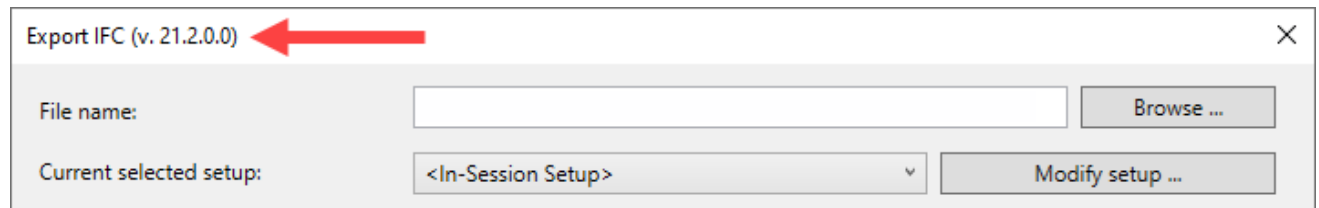
**BIMcollab Zoom**：由 BIMcollab 开发，支持包括 IFC 4 在内的 IFC 版本，是一款功能更为强大的商业软件。

### Revit IFC 开源软件

Revit 带有一个集成式 IFC 解析器，用于读写 IFC 文件；这是开源项目的一部分，因此独立于 Revit 进行更新。新版本会在两个位置发布：

- Github（安装文件和源代码）：  
<https://github.com/Autodesk/revit-IFC>
- Autodesk AppStore（安装文件，发布时间通常比 Github 晚 1-2 周）：  
<https://apps.autodesk.com/>

当前安装的版本显示在导出对话框中（Revit > “导出” > “IFC”）：



不显示版本即表示当前使用的是 Revit 附带的原始版本。

重要提示：每个版本的 Revit 均有一个单独的安装程序，安装时也会更新 Navisworks 中的解析器。

安装会更新当前版本的 Revit IFC，还会提供附加资产。其中最相关的是 IFC 共享参数文件，用于在 Revit 中添加 IFC 属性。这些参数文件存储在：C:\ProgramData\Autodesk\ApplicationPlugins\IFC 20xx.bundle

## 在 Revit 中使用 IFC 文件

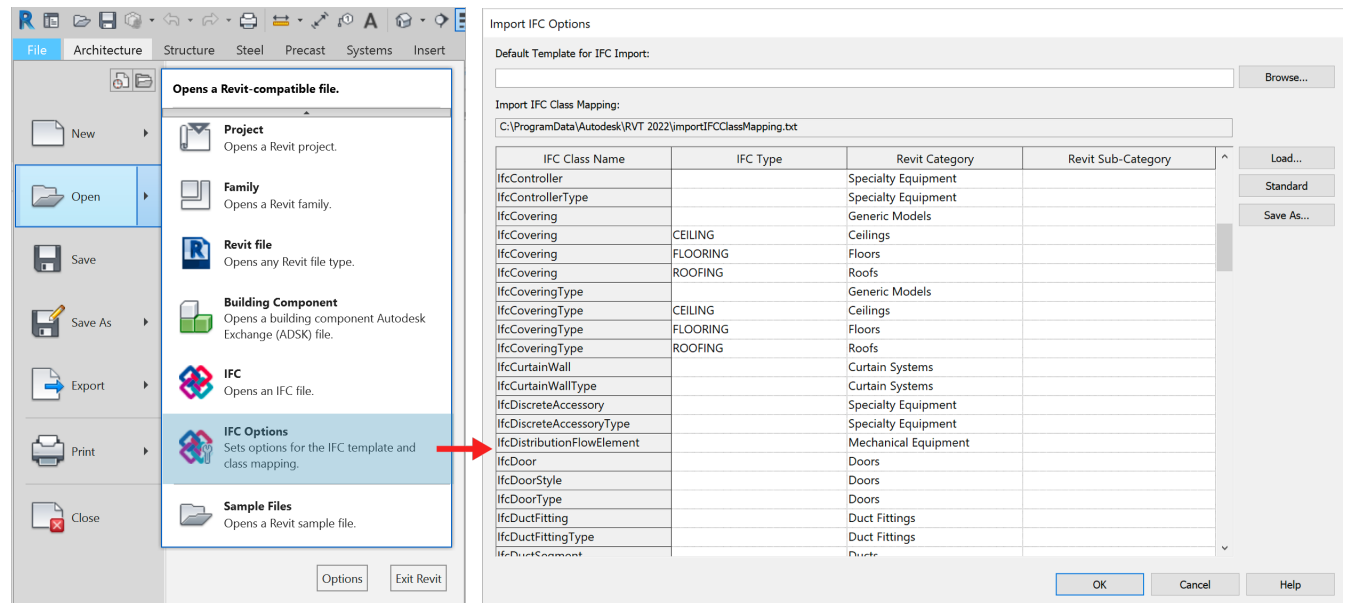
要在 Revit 中使用 IFC 文件，可以将其链接为参照（推荐做法）或直接打开。

### 常规设置

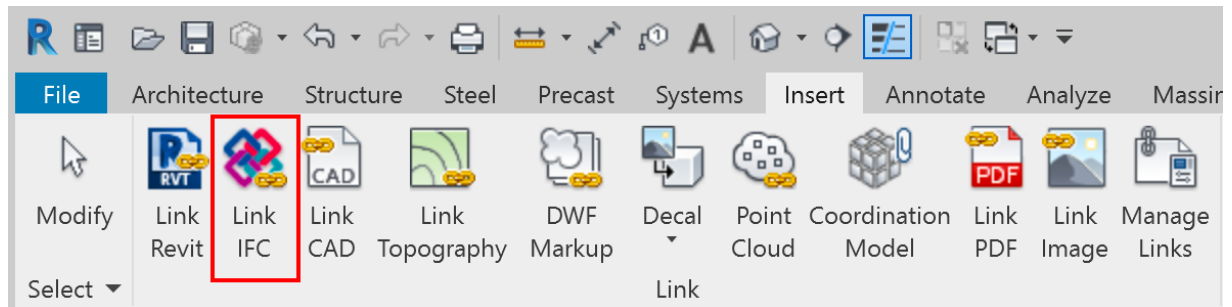
在 Revit 中选择“文件”>“打开”>“IFC 选项”，可以打开和链接 IFC 文件：

**导入（链接）IFC 的默认样板：**将使用在常规 Revit 选项中定义的列表中的第一个样板，此样板在创建新项目文件时也会出现。建议为导入/链接 IFC 选择一个最简单的样板，以免文件中充满视图或族等不必要的信息。选择“新建”>“项目”>“样板：<无>”，即可从头创建一个最简单的样板，然后将其保存为新的 IFC 样板。

**导入 IFC 类映射：**一个与导出映射表非常相似的映射表。它可以直接在对话框中编辑，也可以通过打开和编辑参照的文本文件来进行编辑。当默认的映射表不包含特定的 IFC 类和类型时，这一映射表尤为有用。可以通过输入不导入将类排除在外（而非 Revit 类别）。建议排除那些与 Revit 无关的类，以获得最佳性能。

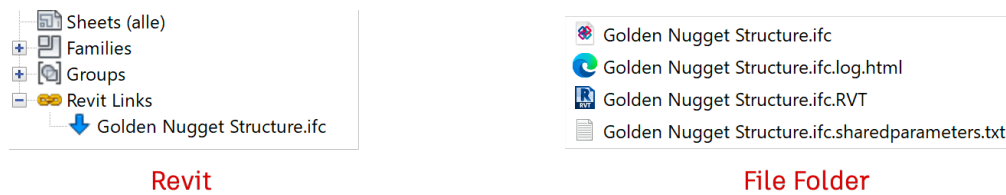


### 链接 IFC



要在 Revit 中使用 IFC 数据，最可靠的首选方法是在 Revit 中链接或参照 IFC 文件。此方法将在后台处理 IFC 文件，并将其作为参照显示。链接的 IFC 文件如有更新，在下次打开项目时，系统将自动在 Revit 中重新加载并进行更新。或者，也可以手动更新，方法是在项目浏览器中选择更新文件并单击“重新加载”。

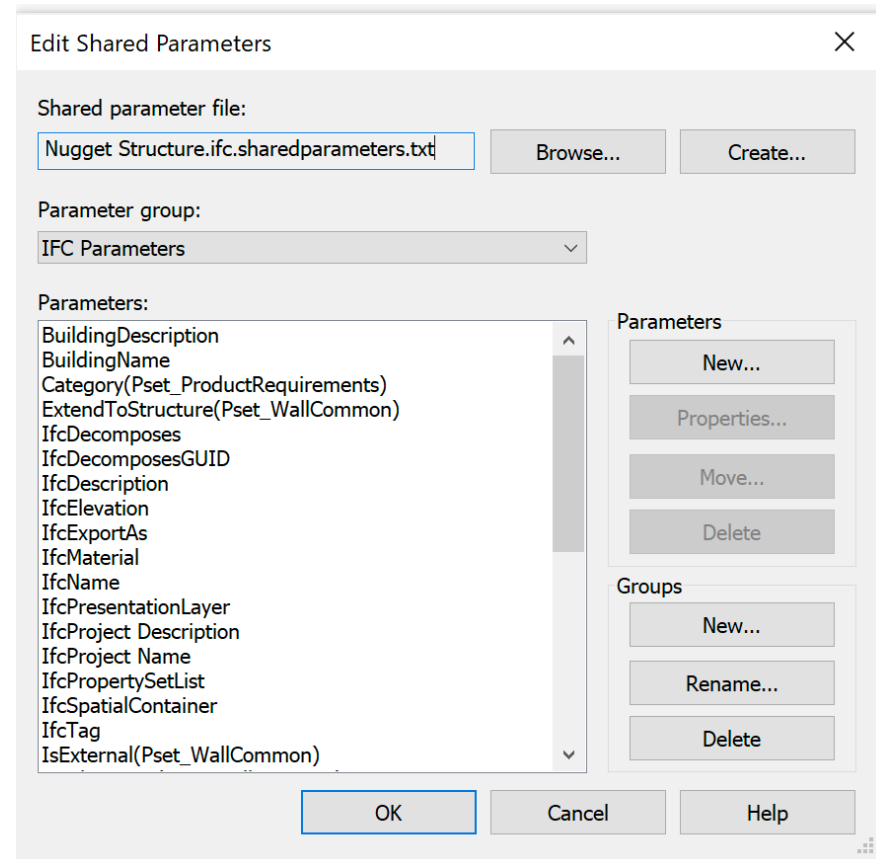
在 Revit 中链接 IFC 文件时，会自动在同一目录下创建三个文件：



\* ifc.RVT：在 Revit 内部使用，无法移动或编辑，旨在保持 Revit 项目与 IFC 文件之间的关系。

\* ifc.log.html：这实际上是转换过程的日志文件，其中包含关于链接图元的报告以及错误消息和提示，有助于进行故障排除

\* ifc.sharedparameters.txt：包含 IFC 中的共享 IFC 参数。为了能够调度链接的 IFC 文件中包含的某些参数，可通过此文件将这些参数添加到项目中。



## 打开 IFC

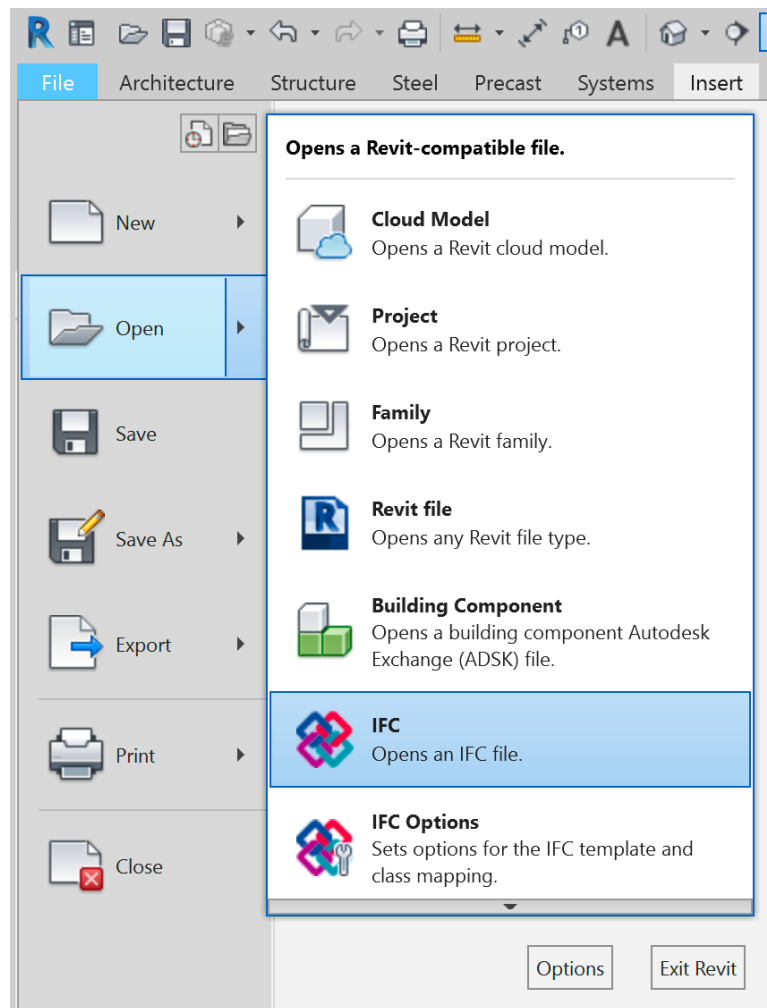
IFC 文件也可在 Revit 中打开，这会将所有的 IFC 几何图形转换为原生的 Revit 族，使其可供编辑。正如本手册开头所述，IFC 作为一种协调格式而开发，在转换和编辑方面的功能仍然有限。IFC4 设计传递视图等新概念正在解决这一问题，但 buildingSMART 尚未开发完成。

此外，更改 IFC 数据会导致责任问题。

在某些情况下，由于创作软件的变化，必须导入 IFC 文件。需要注意的是，目前这一过程可能会导致数据丢失，因此需要检查导入的模型是否有误或缺失图元。然而，最重要的是 IFC 文件的实际内容和质量，这取决于导出设置。

在 Revit 中导入 IFC 文件时，最好遵循以下最佳实践：

- 在查看器中检查 IFC 文件，确保所有图元均已正确分类；如果分类有误，则重新获取一个正确分类的 IFC 文件。
- 在文本编辑器中打开 IFC 文件，检查标题中关于 IFC 模式和 MVD 的信息。目前建议在 Revit 中打开 IFC2x3 协调视图 2.0 以获得最佳效果。
- 在“IFC 选项”下的映射表中插入“不导入”，排除所有 Revit 中不需要的 IFC 类。
- 在打开对话框中禁用自动连接图元和更正稍微偏离轴的线以加快导入速度。

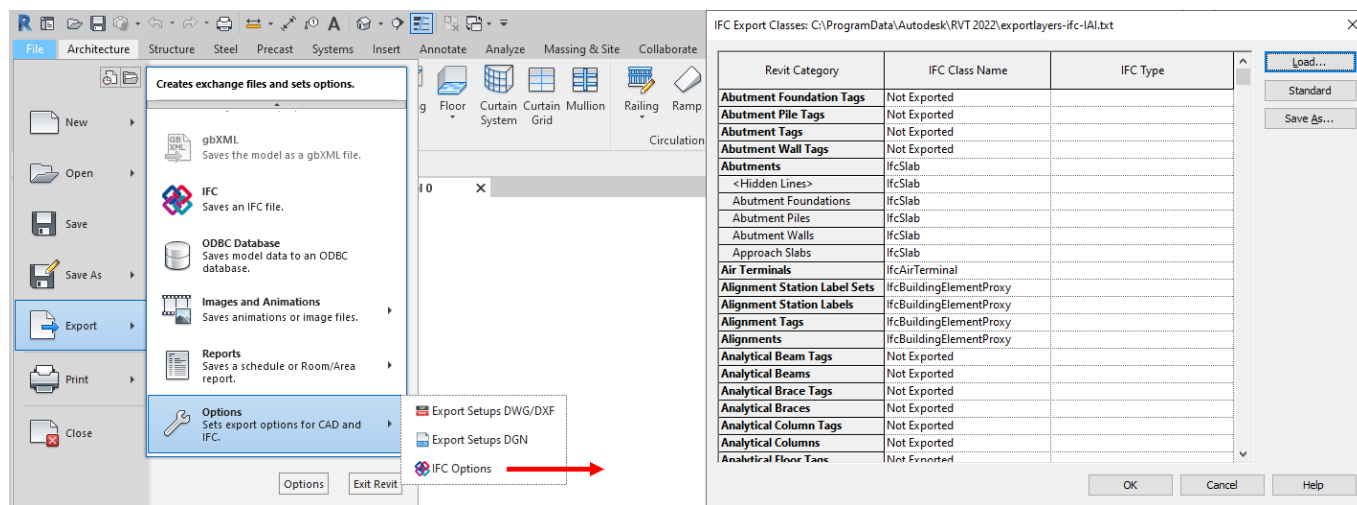


## 从 Revit 导出为 IFC

### 默认映射

最重要的导出设置是将 Revit 类别正确映射至 IFC 类。  
 此操作可通过位于“C:\ProgramData\Autodesk\RVT20xx<sup>10</sup>”目录下的“exportlayers-ifc-IAI.txt”通用映射表完成。要编辑/更改此映射表，从 Revit 用户界面中选择菜单项“文件”->“导出”->“选项”->“IFC 选项”：

注意：无法在这一层级替代 Revit 子类别和 IFC 类型，只有主 Revit 类别应映射至 IFC 类。对于更精细的映射，可以单独映射图元。使用“不导出”替换 IFC 类名称时，会从导出中完全排除 Revit 类别。



如果使用不同语言版本的 Revit，将根据对话框启动时所用的第一种语言生成“exportlayers-ifc-IAI.txt”。要将映射表重置为默认设置和/或当前语言，请删除文本文件（路径见标题），然后在上述对话框中单击“标准”。此操作将以硬编码设置重新创建映射文件。

建议将自定义设置保存在一个单独的文件中。

10. 20xx 代表所使用的 Revit 版本。

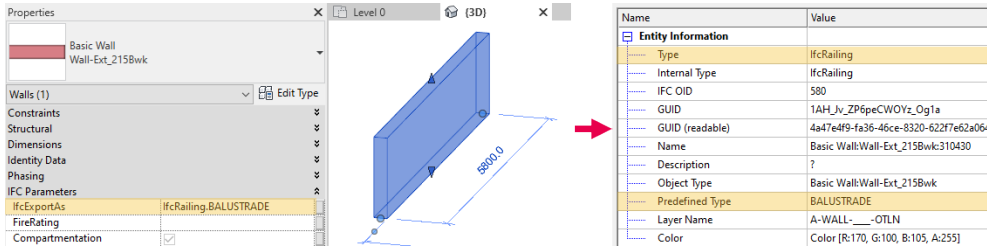
### 单独映射

在很多情况下，需要基于图元替代之前讨论的全局映射，因为 IFC 类通常比 Revit 类别更为精细，也具有自己的预定义类型。

基于图元的映射可通过为 IfcExportAs 参数指定值来实现。强烈建议使用 Revit IFC 附带的共享参数文件将该参数作为共享参数添加到项目中。

此参数的值必须为 IfcClass.TYPE - 两者均在 IFC 模式中定义。与主映射表类似，“不导出”可用于从导出中排除某个特定图元。

也可以将 Revit 类别映射至它们原本不属于的类，但请注意，只能导出 Revit 中的信息。在我们的示例中，这意味着可以将墙体映射至预定义类型为 BALUSTRADE 的 IfcRailing。



但是，与普通的栏杆相比，并非所有从 Revit 导出时自动映射的自定义属性均可用于被替代的栏杆，它们可能需要手动添加：

### Revit Railing exported as IfcRailing

Pset_RailingCommon	
Height	900 [mm]
IsExternal	FALSE
Reference	900mm

### Revit Wall exported as IfcRailing

Pset_RailingCommon	
IsExternal	TRUE
Reference	Wall-Ext_215Bwk

注意：对于将幕墙等更为复杂的系统族映射到其他 IFC 类，存在一些限制。有关所有限制和可能映射的概述，请参阅 [Autodesk IFC 资源](#) 中的链接。

IFC 模式还允许使用 USERDEFINED 类型。将 USERDEFINED 添加为类型，然后使用 IfcObjectType 参数指定类型，即可正确使用这些类型。下面是 IFC 4 文档中为 IfcRailing 定义的类型概述：

常量	说明
HANDRAIL	一种栏杆类型，可作为一种支撑结构，支撑使用者所施加的负荷（在手的高度）。一般位于坡道和楼梯附近，采用落地式或壁挂式设计。
GUARDRAIL	一种栏杆类型，旨在防止使用者从边缘具有垂直落差的楼梯、坡道或楼梯平台上跌落。
BALUSTRADE	与 GUARDRAIL 的定义类似，只不过其位置在楼板边缘，而非楼梯或坡道边缘。例如，屋顶或阳台的栏杆。
USERDEFINED	用户定义的栏杆图元，此术语用于表示用户类型由 IfcRailing.ObjectType 属性给出。
NOTDEFINED	未定义的栏杆图元，无可用的类型信息。

在 Revit 中，用户定义的类型定义如下所示：

The screenshot displays the Revit Properties panel on the left and the Entity Information table on the right. The Properties panel shows a railing element with the following settings:

- Railings (1)**: Edit Type
- Constraints**: (collapsed)
- Dimensions**: (collapsed)
- Identity Data**: (collapsed)
- Phasing**: (collapsed)
- IFC Parameters**:
  - IfcExportAs: IfcRailing.USERDEFINED
  - IfcObjectType: My special railing type

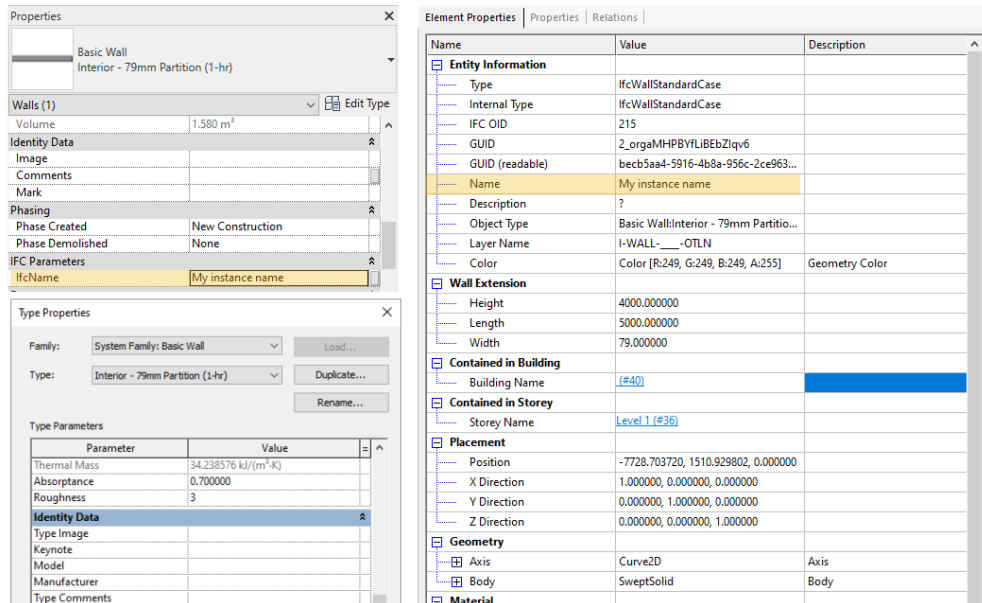
The Entity Information table on the right provides the following details:

Entity Information	
Type	IfcRailing
Internal Type	IfcRailing
IFC OID	1059
GUID	1AH_Jv_ZP6peCWOYz_Ojv\$
GUID (readable)	4a47e4f9-fa36-46ce-8320-622f7e62de7f
Name	Railing:900mm:311941
Description	?
Object Type	My special railing type
Predefined Type	USERDEFINED

A red arrow points from the 'My special railing type' value in the 'IfcObjectType' property to the 'Object Type' row in the Entity Information table.

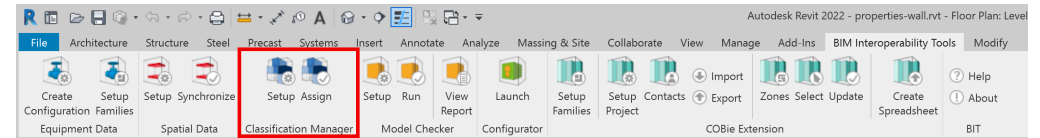


此外，还可以为 IFC 实体导出用户定义的类型名称。Revit 使用一个特殊的类型参数“NameOverride”来更改 Revit 图元类型的名称。此参数与实例参数“IfcName”相结合，可根据项目或公司标准制定任何命名约定。



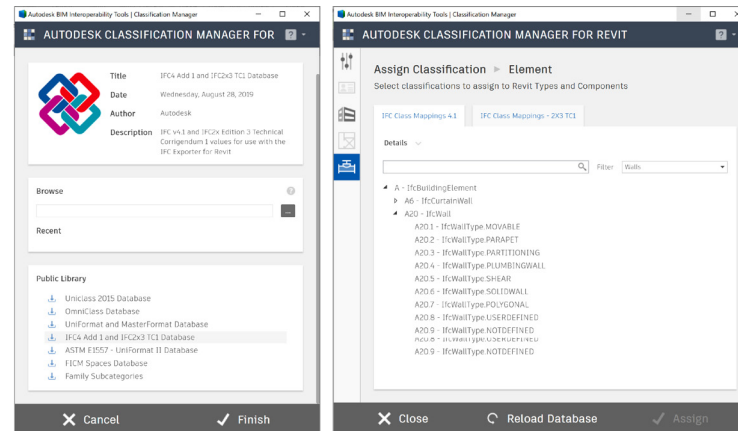
### Autodesk Classification Manager for Revit

Autodesk 互操作性工具是一个免费的插件集，可在 <https://interoperability.autodesk.com> 获取。



Classification Manager 带有一套预定义的分类表，其中包括 IFC2x3 和 IFC4 分类表。Classification Manager 的对话框提供了一个选择列表，可简化类的单独映射，同时还支持对实例和类型参数的图元和类别进行多重选择。

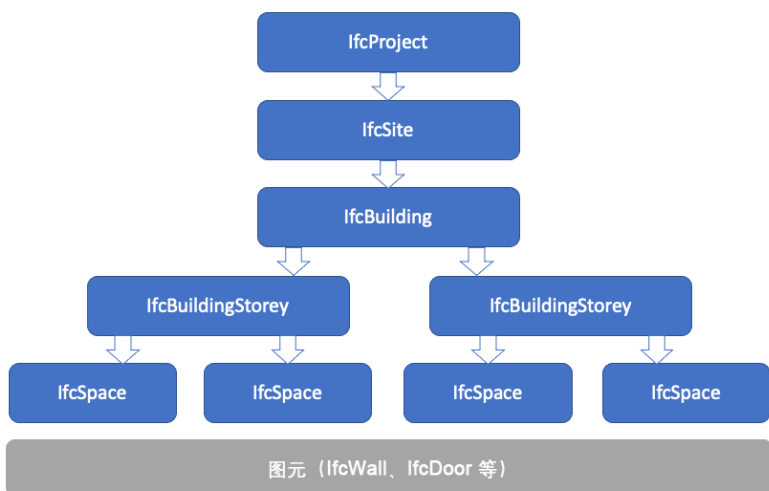
如果项目中不存在 IfcExportAs 参数，则预定义的配置还会将其创建一个类型参数。配置文件可使用 Excel 格式下载，其中还包括说明，可根据需要进行调整。



## IFC 文件导出选项

### IFC 基本结构

IFC 模式的结构颇为复杂，包含许多对最终用户不可见的抽象图层。对于可在 IFC 查看器中使用的可见结构，我们可以注意到以下层次结构：

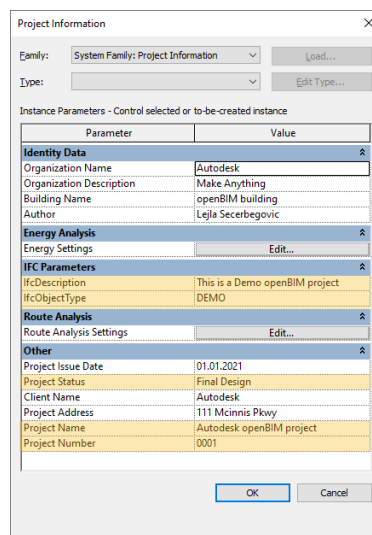


前 3 个实体。IFC 模式允许每个场地存在多个建筑物，但它并不支持一个 Revit 项目中存在多个建筑物，因此 Revit 只能导出一个建筑物。

这些实体的处理方式与 Revit 中的其他实体不同，因为它们在 Revit 中没有物理表示，而是衍生自项目信息。

### IfcProject

顶级实体通常是 IFC 查看器的树状结构中的主容器。它未定义任何 Pset，也无法在这一层级附加自定义的 PSet，但项目中有一些可填充的属性：



Result in IFC:

Name	Value
<b>Entity Information</b>	
Type	IfcProject
Internal Type	IfcProject
IFC OID	127
GUID	2317quDJ511e3GjRdswfvcv
GUID (readable)	83bc7d38-3531-4106-80d0-b5b9f6ea99b9
Name	0001
Description	This is a Demo openBIM project
Object Type	DEMO
Layer Name	
Color	---
LongName	Autodesk openBIM project
Phase	Final Design
<b>Placement</b>	
Position	0.000000, 0.000000, 0.000000
X Direction	1.000000, 0.000000, 0.000000
Y Direction	0.000000, 1.000000, 0.000000
Z Direction	0.000000, 0.000000, 1.000000

注意：“IFC 参数”下的参数已手动添加，并作为实例参数指定给“项目信息”类别。

“图层名称” / “颜色”只与表示物理对象的实体相关，而 IfcProject 只是一个容器，因此在 CAD 软件中没有物理表示。

## 带 IfcSite 的 IfcProject

第二级表示场地，从项目来看更复杂一些，因为它可以与 Revit 中的地形对象相关联。在没有地形的情况下，也可以通过共享参数文件将主属性添加到“项目信息”中（只需搜索所有以“Site”开头的属性即可）：

The screenshot displays the 'Project Information' dialog box with various tabs like Identity Data, Energy Analysis, IFC Parameters, Route Analysis, and Other. The 'IFC Parameters' tab is active, showing fields for IfcDescription, IfcObjectType, SiteName, SiteDescription, SiteLandTitleNumber, SiteCoverageRatio, SiteLongName, and SiteObjectType. Below the dialog, the 'Result in IFC' tree view shows a hierarchy starting with '0001' and 'opensite'. The 'Property Toolbar' table shows the resulting IFC properties for the 'opensite' object.

Name	Value
Type	IfcSite
Internal Type	IfcSite
IFC OID	189
GUID	2317quDJ511e3GjRdswfcx
GUID (readable)	83bc7d38-3531-4106-80d0-b5b9f6ea99bb
Name	opensite
Description	opensite description
Object Type	opensite demo
Layer Name	
Color	---
LongName	long opensite
CompositionType	Element
RefLatitude	N 42° 21' 31.1819"
RefLongitude	W 71° 3' -24.-263"
RefElevation	0.000000
LandTitleNumber	2022
North Direction (GeometricRep...	0.00
MapConversion (GeometricRep...	

如果项目中包含地形对象，则也可以在这一级指定 IFC 属性，这将覆盖之前在“项目信息”中指定的属性。

The screenshot shows the 'Properties' dialog box for a 'Topography (1)' object, with the 'IFC Parameters' tab selected. The 'IFC Parameters' section shows 'IfcDescription' set to 'New Site Description', 'IfcObjectType' set to 'New Site Type', and 'IfcName' set to 'New Site Name'. To the right, the 'Result in IFC' tree view shows the hierarchy for 'New Site Name' under '0001'.

Name	Value
Type	IfcSite
Internal Type	IfcSite
IFC OID	228
GUID	2317quDJ511e3GjRdswfcx
GUID (readable)	83bc7d38-3531-4106-80d0-b5b9f6ea99bb
Name	New Site Name
Description	New Site Description
Object Type	New Site Type
Layer Name	C-TOPO-___-OTLN
Color	---
LongName	long opensite
CompositionType	Element
RefLatitude	N 42° 21' 31.1819"
RefLongitude	W 71° 3' -24.-263"
RefElevation	0.000000
LandTitleNumber	2022
North Direction (GeometricRep...	0.00

此操作也可以通过其他可用属性（如 LongName 和 LandTitleNumber）来实现。根据 IFC 4 RV 文档，IfcSite 有两个预定义的 Pset：Pset\_SiteCommon 和 Pset\_LandRegistration，两者均受支持并包含在共享参数文件中。只需添加属性（至“项目信息”或“地形”类别）并进行填充。

RefLatitude 和 RefLongitude 衍生自 Revit 的“管理”选项卡中设置的“位置”。

## IfcBuilding

第三个容器也是第一个空间容器，代表建筑物。它也在“项目信息”中定义。您可以通过共享参数文件添加更多支持的属性，方法为搜索以“Building”开头的属性并将其添加到“项目信息”类别。

The image displays four screenshots related to IFC export configuration in Revit:

- Project Information Dialog:** Shows the 'Instance Parameters' section with a table of parameters for the 'openBIM building' instance.
- Result in IFC:** A tree view showing the hierarchy: 0001 -> New Site Name -> openBIM building -> Level 1.
- Element Properties Table:** A table showing the mapping of Revit parameters to IFC properties for the 'openBIM building' element.
- Shared Parameters Dialog:** A dialog box for selecting parameter groups and parameters. The 'IFC Properties' group is selected, and a list of parameters is shown, with 'BuildingDescription' highlighted.

Parameter	Value
<b>Identity Data</b>	
Organization Name	Autodesk
Organization Description	Make Anything
Building Name	openBIM building
Author	Lejla Secerbegovic
<b>Energy Analysis</b>	
Energy Settings	Edit...
<b>IFC Parameters</b>	
IfcDescription	This is a Demo openBIM project
IfcObjectType	DEMO
SiteName	opensite
SiteDescription	opensite description
SiteLandTitleNumber	2022
SiteLongName	long opensite
SiteObjectType	opensite demo
BuildingDescription	This is the demo building for openBIM
BuildingLongName	openBIM building
BuildingObjectType	commercial

Name	Value
<b>Entity Information</b>	
Type	IfcBuilding
Internal Type	IfcBuilding
IFC OID	142
GUID	2317quDJ511e3GjRdswfcu
GUID (readable)	83bc7d38-3531-4106-80d0-b5b9f6ea99b8
Name	openBIM building
Description	This is the demo building for openBIM
Object Type	commercial
Layer Name	
Color	---
LongName	openBIM building
Composition Type	Element
ElevationOfRefHeight	0.000000
ElevationOfTerrain	0.000000

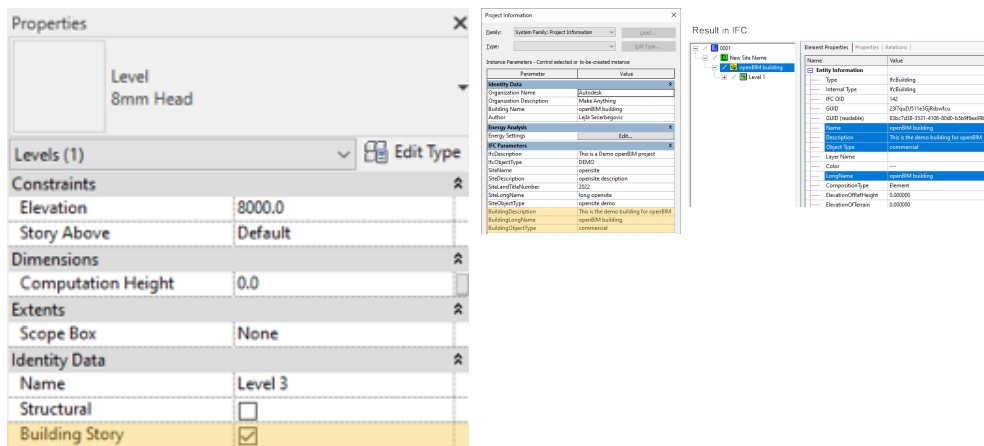
如果通过共享参数文件添加并填充属性，则会自动导出在 IFC 模式中定义的 Pset。

如前所述，IFC 模式支持多个建筑物，但由于 Revit 内部结构的原因，每个项目只能导出一个建筑物。

## IfcBuildingStorey

第四个容器相当于实际的建筑楼层，其中包含墙体或家具等各种建筑图元。由于 Revit 中经常有很多不代表建筑结构的参照标高，因此每个标高的属性中均有“**建筑楼层**”选项，用于定义是否导出该标高。

启用此选项后，系统会将标高导出到 IFC，否则则会将其忽略。对于在 Revit 中指定给非建筑楼层的图元，系统会自动将其指定给下一个较低的建筑楼层；如果没有较低的建筑楼层，则会将其指定给下一个较高的建筑楼层。每个项目至少应有一个建筑楼层。



## 使用 IFC 共享参数

默认情况下，并非所有在 IFC 模式中定义的属性都是 Revit 的一部分，否则会导致项目过载。建议仅添加特定项目中需要的参数。常用的参数可以添加到项目样板中。

Revit IFC 开源软件带有两个共享参数文件，安装后存储在以下文件夹中：

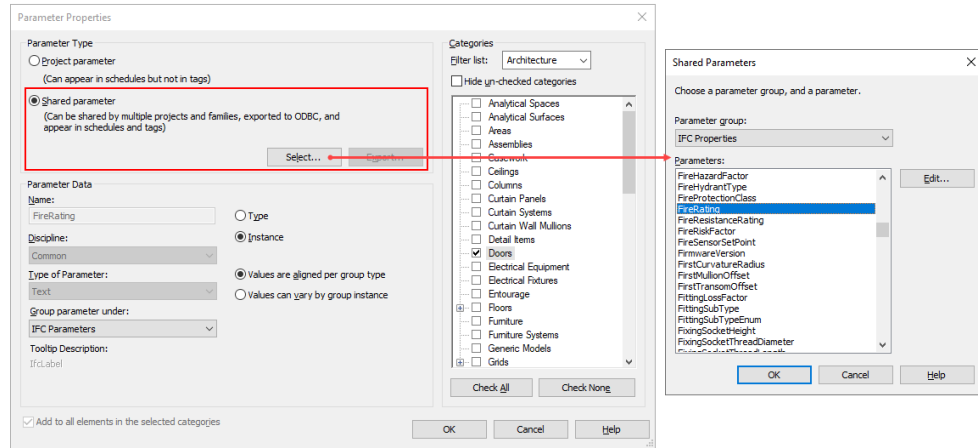
C:\ProgramData\Autodesk\ApplicationPlugins\IFC <Version>.bundle\Contents\

或者，您也可以从上一章提到的 Github 储存库进行下载。

这两个文件分别为：

- IFC Shared Parameters-RevitIFCBuiltIn\_ALL.txt
- IFC Shared Parameters-RevitIFCBuiltIn-Type\_ALL.txt

共享参数通过“管理”>“项目参数”中的对话框添加到 Revit 中，建议使用第一个文件添加实例参数，使用第二个文件添加类型参数。



提供两个文件的原因如下：与 Revit 一样，IFC 模式也基于类型和实例。然而，在 IFC 中，同一个参数可以同时附加到实例和类型（也可以赋予不同的值），而 Revit 要求用户在分配参数时在类型与实例之间选择其一，不可同时选择。

根据项目要求，您可能需要将某些属性同时附加到 IFC 实例和 IFC 类型两个层级。为此，您可以通过第一个文件添加实例属性，而通过第二个文件添加类型属性。第二个文件的属性在 Revit 的名称中包含 [Type]，这将在导出时删除。

为了说明这一点，我们假设您需要使用具有不同的 FireRating 类型和实例参数值的 Pset\_DoorCommon 来交付门。步骤：

- 根据之前的截图，通过 IFC Shared Parameters-RevitIFCBuiltIn\_ALL.txt 添加实例属性，并将其指定给“门”类别，最好将其分组在“IFC 参数”下（不是强制性的，但可以改进概述）。
- 通过 IFC Shared Parameters-RevitIFCBuiltIn-Type\_ALL.txt 添加类型属性，并将其指定给“类型”（默认设置为“实例”），然后选择“门”类别并将其分组在“IFC 参数”下。

结果应与下图类似：

PropertySets from entity	
Pset_DoorCommon	
FireRating	90
IsExternal	FALSE
Reference	0915 x 2134mm
ThermalTransmittance	3.7021
Pset_EnvironmentalImpactIndic...	
Qto_DoorBaseQuantities	
PropertySets from entity type	
Pset_DoorCommon	
FireRating	0
IsExternal	FALSE
ThermalTransmittance	3.7021

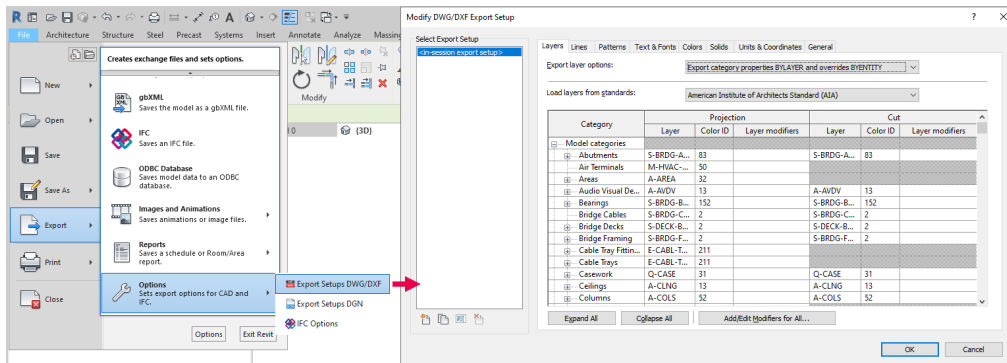
这一做法是否适用于您的项目在很大程度上取决于您的要求，但这很可能非常有帮助。

## 基于图层的软件导出

某些软件产品可能需要在 IFC 分类中加入一个图层结构。Revit 根据默认的 CAD (.dwg/.dgn) 映射文件自动指定图层值。默认的配置文件为：C:\ProgramData\Autodesk\RVT 20xx\exportlayers-dwg-AIA.txt

此文件中的配置可以按需调整，方法为在 Revit 用户界面中选择“导出”>“选项”>“导出设置 DWG/DXF”，或使用以下语法手动调整：

<Revit Category Name><tab><tab><Layer Name>



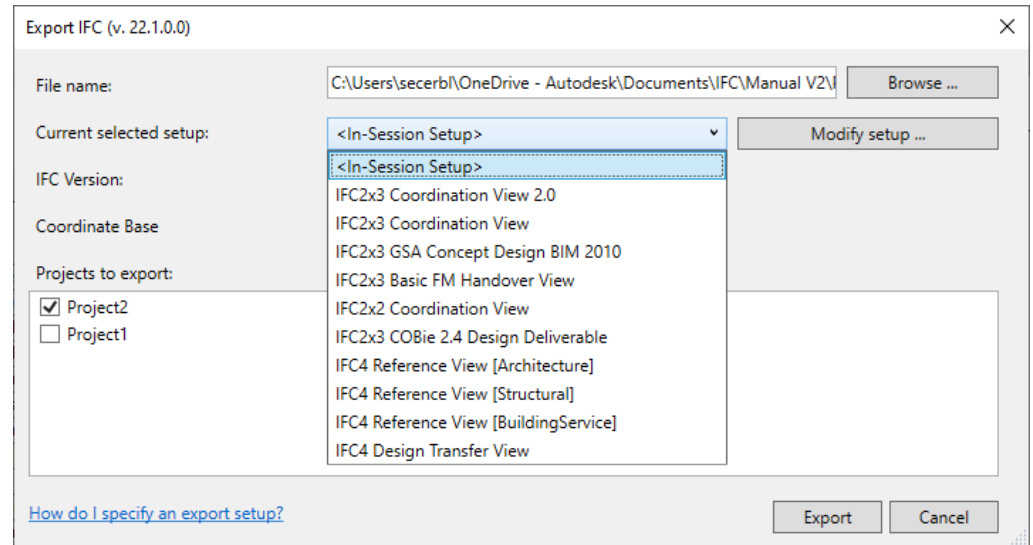
自定义图层参照文件的参照需要添加到位于以下位置的 Revit.ini 文件中：C:\Users\<USER>\AppData\Roaming\Autodesk\Revit\Autodesk Revit 20xx

图层参照文件的完整路径将添加到以 ExportLayersNameDGN= 开头的行中

例如：ExportLayersNameDGN=C:\Users\<USER>\Documents\RevitLayers.txt  
与类映射一样，有时需要在图元级别指定图层值。为此，您可以使用官方共享参数文件中包含的共享参数 IfcPresentationLayer。

## IFC 导出设置对话框

在 Revit 中单击“文件”>“导出”>“IFC”即可打开 IFC 导出对话框，您可以在其中直接选择所有内置的模型视图定义 (MVD)，并导出所有打开项目，而不仅仅是活动项目：

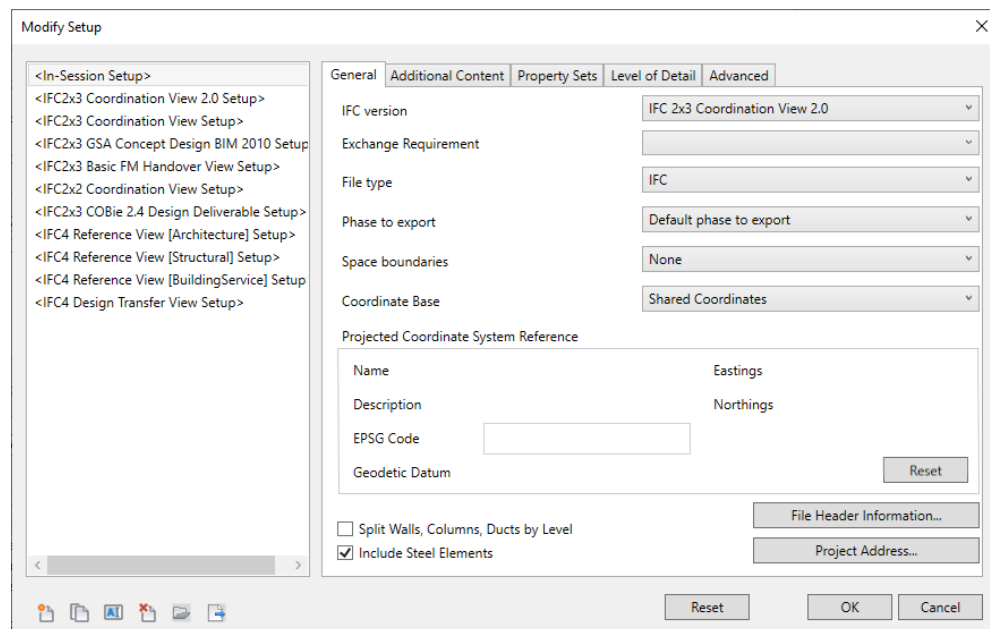


如本手册之前所述，选择合适的 IFC 版本和 MVD 对于导出为 IFC 的内容质量至关重要。最常用的 MVD 为 IFC2x3 协调视图 2.0 和 IFC4 参照视图。

此外，这些设置可通过选择“修改设置”进行修改。下面几页提供了有关这些设置的详细信息。

## 常规设置

本部分介绍了如何更改“在任务中设置”的设置，以及如何通过复制现有设置创建新的设置。左侧 <> 中所列的预定义设置无法更改：



**IFC 版本：**用于选择 IFC 规范和 MVD，通常为 IFC2x3 协调视图 2.0 或 IFC4 参照视图。要了解更多信息，请参阅本手册第一章。

**交换要求：**仅在使用 IFC4 时有效，buildingSMART 借此为建筑、结构和机电交换的认证定义了不同的用例。

**文件类型：**用于选择备选类型，如 .IFCXML 或压缩版的 .IFC/.IFCXML。导出 .IFC 并进行压缩可以获得相同的结果，但 .IFCXML 仅用于特定应用。大多数情况下，默认设置 .IFC 应是首选。

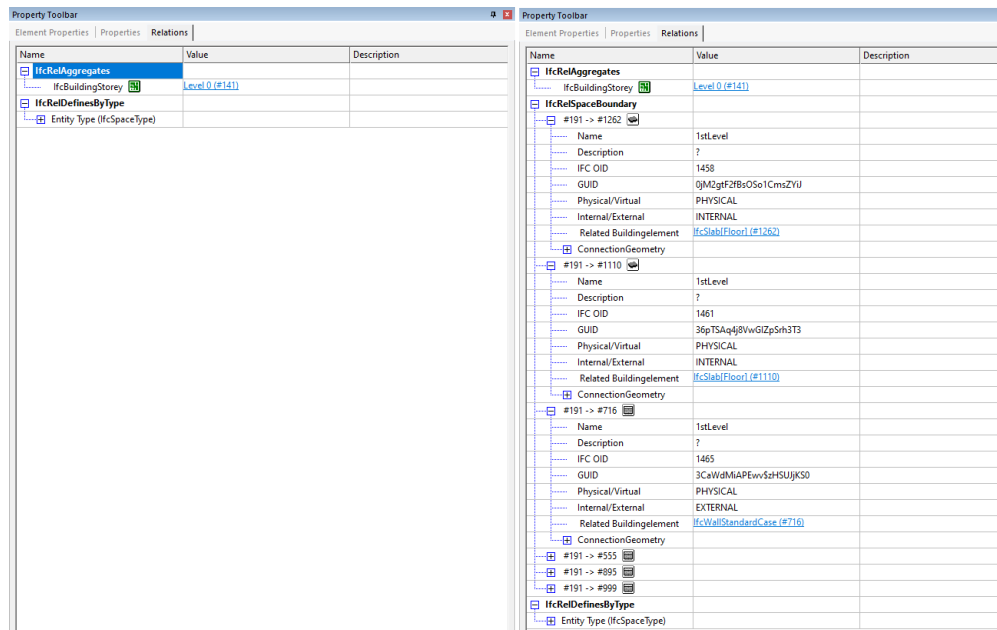
**要导出的阶段：**用于选择项目中要导出的特定阶段。默认情况下，要导出的阶段为项目的最后一个阶段。“仅导出视图中可见图元”处于选中状态时，将使用视图阶段，此选项灰显。

**空间边界：**用于定义导出的房间/空间边界的级别：

- 无：不会导出房间/空间边界。
- 第一级：包括房间/空间边界但不优化以相对于边界另一侧空间拆分图元。
- 第二级：包括房间/空间边界且相对于边界另一侧空间进行拆分。第二级空间边界会考虑建筑图元的材质以及其后相邻的空间，为进一步分析提供热属性。

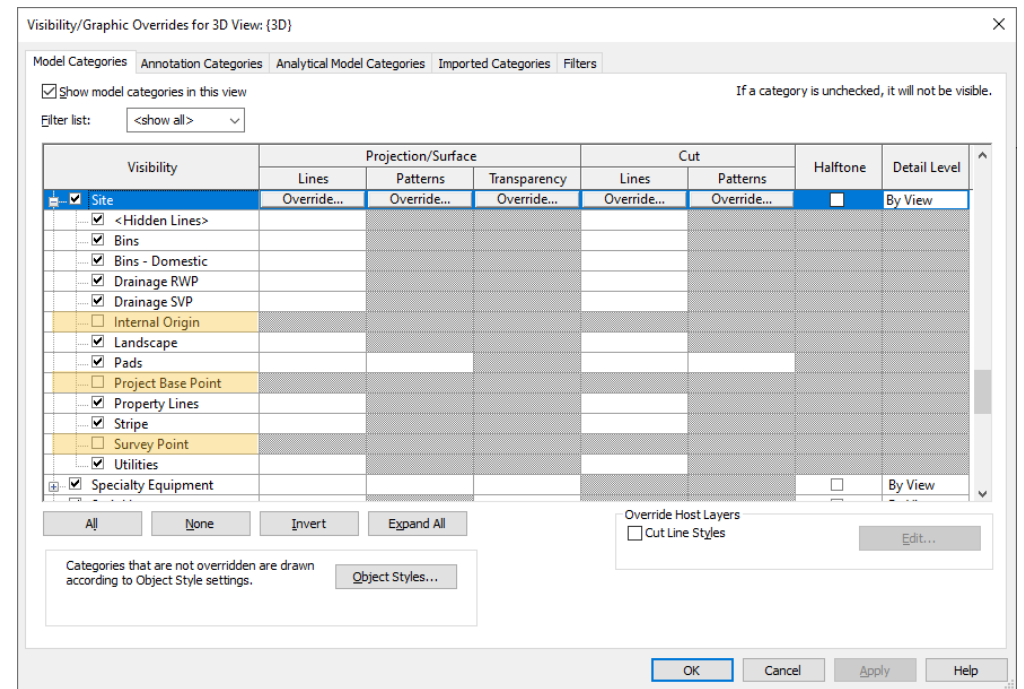


信息会附着到空间和房间边界对象（如墙），并可在大多数查看器中进行查看（例如 FZKViewer，左侧：无，右侧：第一级）：



**坐标基准：**用于选择“共享坐标”、“内部原点”、“项目基点”和“测量点”。

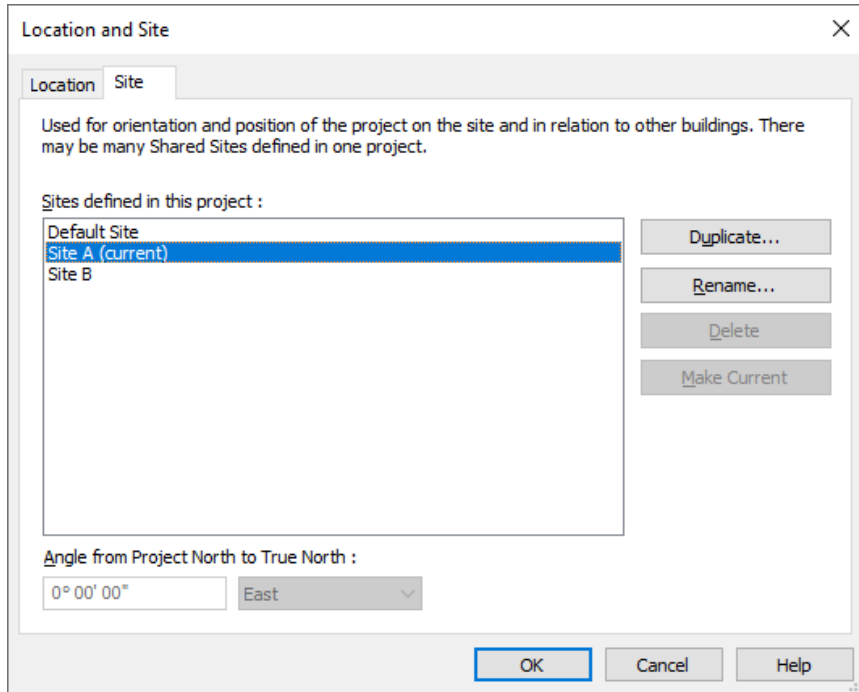
每个 Revit 项目最初都有三个原点，默认情况下通常处于隐藏状态，但可转至“视图”>“场地”的“可见性设置”将其更改为显示状态：



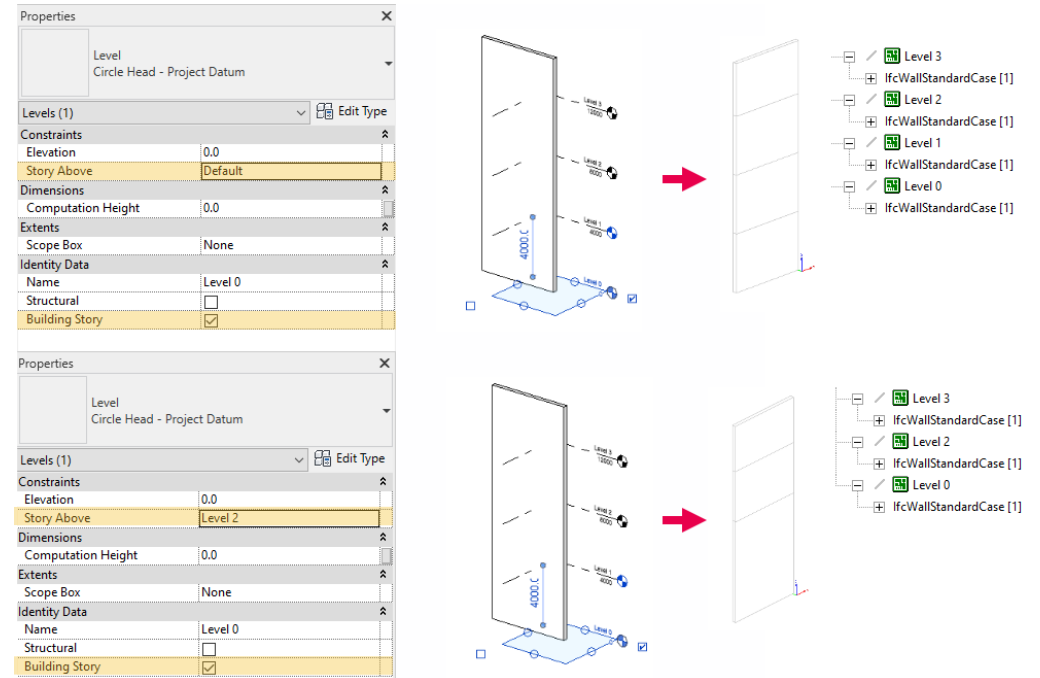
- **内部原点：**无法移动，也代表 Revit 允许几何图形存在的 20 英里区域的中心。所创建的任何类型的几何图形超出这一区域均会导致错误信息，应予以避免。
- **项目基点：**用于定义项目坐标，通常位于地面标高的轴网交点或建筑角点处。通常情况下，项目中所有点的坐标和高度都将参照此点。此点可以移动（手动或键入坐标）到所需位置，但不会移动项目（除非更改“项目北”，这在项目基点中也可以看到）。在 Revit 2020 之前，项目基点还有一个剪裁状态，但现已删除。自 Revit 2020 起，项目基点始终处于未剪裁状态。
- **测量点：**代表现实世界中的一个相关点，可以是剪裁或未剪裁状态。移动剪裁的测量点时，实际上会更改模型的共享坐标系；而未剪裁的测量点也可以移动（手动或键入坐标），但对共享坐标系没有任何影响，这与项目基点类似。

在默认样板中，所有点均应位于同一位置，并根据项目达成的共识进行调整。

共享场地是另一个概念，用于设置链接模型之间的关系。一个 Revit 项目可以包含多个共享场地，而此选项指的是当前选择的场地：



**按标高拆分墙、柱、风管：**导出时自动拆分跨越多个建筑楼层的所有图元。使用此选项时，必须检查定义为建筑楼层的标高，同时查看标高之上选项 - 默认值：将使用下一个更高的建筑楼层来剪切指定给当前标高的所有图元，除非明确选择其他标高。系统会将通过拆分创建的图元指定到剪切所依据的标高。

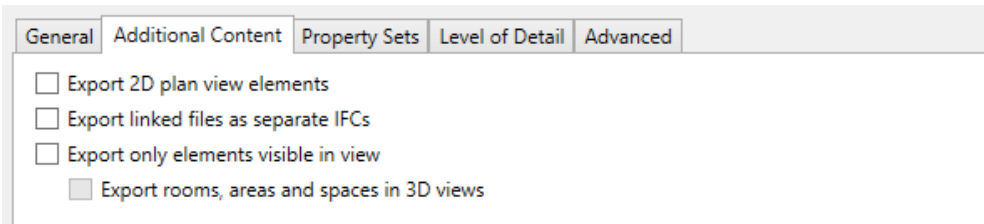


**包括钢图元：**导出结构钢，包括钢结构连接。

**文件标头信息：**用于在 IFC 文件标头中定义作者姓名、电子邮件、组织和授权。

**项目地址：**在导出时将覆盖项目信息中设置的建筑物和/或场地地址，如果选中更新项目信息，则会将该信息推送回 Revit。

### 其他内容



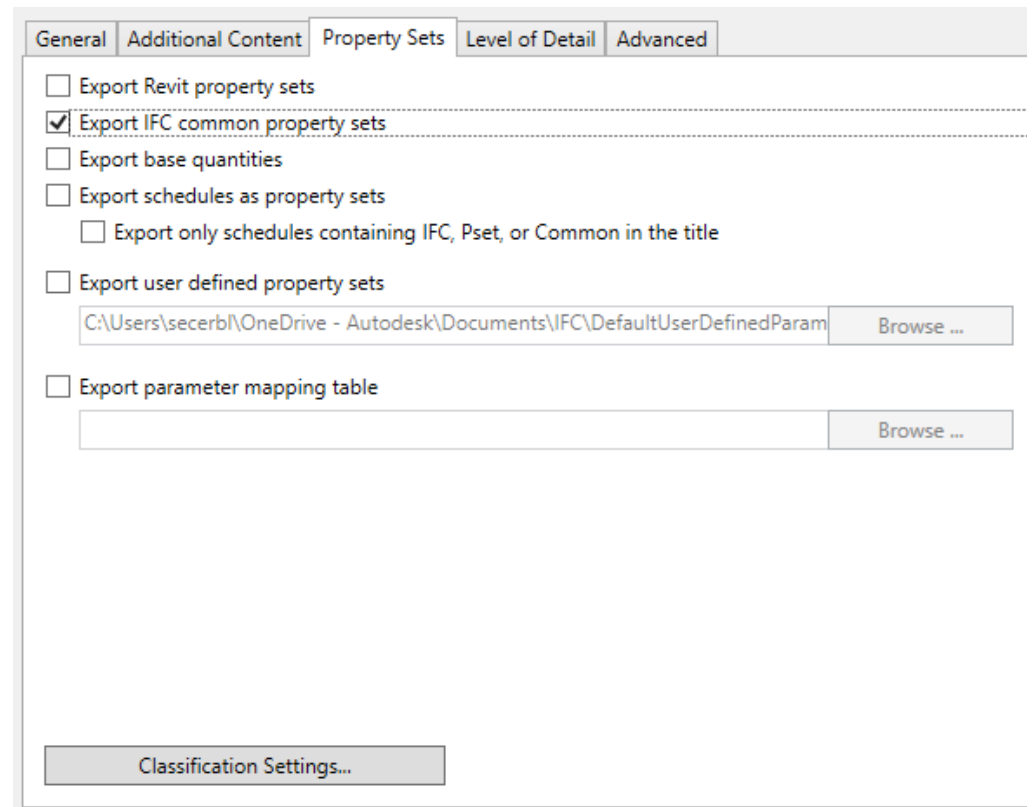
**导出二维平面视图图元：**用于导出 IFC 模式支持的二维图元，如注释和填充区域等。轴网被视为三维图元，可通过将轴网的 Revit 类别指定为 IfcGrid 类进行导出。需要注意的是，IFC 是一种面向三维对象的模式，一般只支持一定数量的二维图元，因此二维文档仍经常使用 PDF。

**将链接的文件导出为单独的 IFC：**使用相同的设置将所有链接的文件导出为单独的 IFC。从 Revit 导出时，无法将多个 Revit 项目合并为一个 IFC，但这些文件可一起显示在 Navisworks 或大多数 IFC 查看器中。

**仅导出视图中可见图元：**根据当前的活动视图确定要导出哪些图元。由于 Revit 中的三维视图不显示房间、区域和空间，因此可使用另一个选项来包括这些内容：**以三维视图导出房间、区域和空间。**

### 属性集

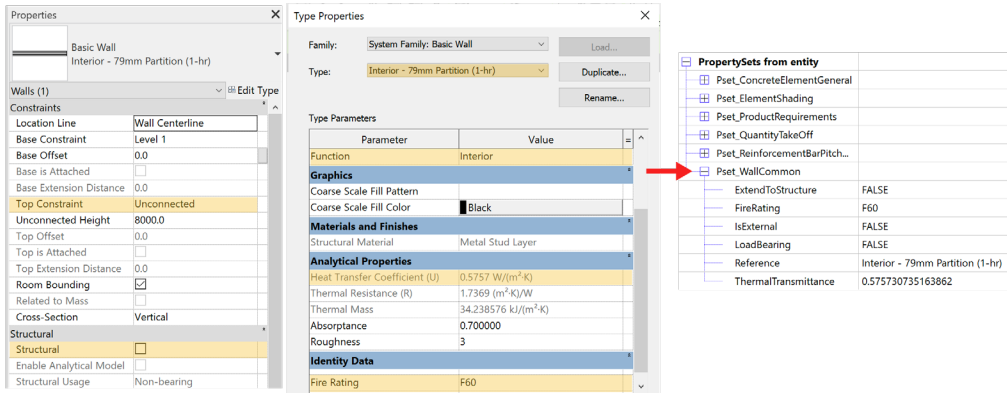
属性集包含模型中定义的信息，因此是重要性仅次于正确分类的导出设置。请注意，一般来说，不会导出空的属性。



**导出 Revit 属性集：**默认情况下处于未激活状态，因为此选项将根据内部分组导出所有 Revit 属性。这将导致 IFC 中包含很多过时信息，也会显著增加文件的大小。建议谨慎使用此选项，仅将其用于测试目的。

**导出 IFC 公用属性集：**导出 IFC 模式中定义的默认属性，默认情况下处于激活状态。现有的 Revit 属性将自动映射到 IFC 属性。

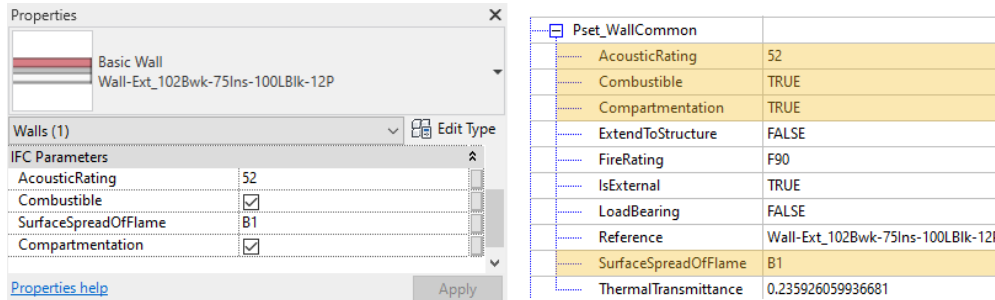
在导出后，常用属性集可通过 Pset\_ 前缀来识别：



IFC 模式中包含许多并非在所有项目中都常用的属性，这些属性在默认情况下并未包含在 Revit 中；使用此选项时，仅会导出 Pset 中定义的属性子集。完整的 Pset\_WallCommon 包含几个 Revit 在默认情况下未提供的属性：

属性	说明
Reference	构件类型（类型名称）
AcousticRating	隔音等级
FireRating	防火等级（类型参数）
Combustible	可燃材质
SurfaceSpreadOfFlame	火势
ThermalTransmittance	U 值（类型参数）
IsExternal	外部构件（类型参数，指定为“是/否”）
ExtendToStructure	顶部固定（表现）
LoadBearing	承重（实例参数）
Compartmentation	定义防火分区的构件

有多个选项可用于添加这些属性。第一个也是最简单的方法是将 IFC 模式中定义的具有相同名称和数据类型的属性添加到 Revit 中。要实现这一目标，最简单的方法是使用本手册之前介绍的 IFC 共享参数文件（请参见“使用 IFC 共享参数”）。这可以确保拼写和数据类型均正确无误。添加和填充这些属性后，它们将在导出时自动添加到 Pset 中：



另外，也可以将其他属性（只要数据类型相同）映射到相应的 IFC 属性。

**导出基准数量：**包括 IFC 模式中定义的另一种类型的属性集，这些属性集专用于估算和 QTO 应用。对于墙体，这些数量通常如右侧所示：

BaseQuantities	
GrossFootprintArea	0.40 [m <sup>2</sup> ]
GrossSideArea	40.00 [m <sup>2</sup> ]
GrossVolume	3.160 [m <sup>3</sup> ]
Height	8000 [mm]
Length	5000 [mm]
NetSideArea	40.00 [m <sup>2</sup> ]
NetVolume	3.160 [m <sup>3</sup> ]
Width	79 [mm]

**将明细表导出为属性集：**可通过 Revit 明细表创建用户定义的属性集。所有不属于 IFC 模式中定义的标准属性集的属性均可添加至自定义属性集。由于 Revit 项目可以有多个明细表，因此也可以将此选项限制为**标题中包含“IFC”、“Pset”或“常用”的明细表**。

所有属性都将收集在明细表中，并且在导出后可以在 IFC 中找到：

<My IFC wall properties>				
A	B	C	D	E
Family and Type	Base Constraint	Top Constraint	Unconnected Height	Length
Basic Wall: Interior - 79mm Partition (1-hr)	Level 1	Unconnected	8000	5000

My IFC wall properties	
Base Constraint	Level: Level 1
Family and Type	Basic Wall: Interior - 79mm Partition (1-...
Length	5000 [mm]
Top Constraint	?
Unconnected Height	8000 [mm]

注意：只有 IFC 模式中定义的正式属性集才能以“Pset\_”开头。

这一工作流的优点在于无需担心数据类型或配置文件，但明细表不易在项目之间传递，因此还有另一个选项，即创建用户定义的属性集。

**导出用户定义的属性集：**相当于将明细表导出为属性集，但使用文本文件作为配置文件。默认的样板文件位于：C:\ProgramData\Autodesk\ApplicationPlugins\IFC 20xx.bundle\Contents\20xx

其中包含详细说明和示例。

基本结构：

```
# Format:
#   PropertySet: <Pset Name> I[instance]/T[type] <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>
```

<> 内的所有内容均将被替换：

<Pset Name>：Pset 的名称，请勿使用 Pset\_ 作为前缀，只有标准的 IFC Pset 才能以此作为前缀

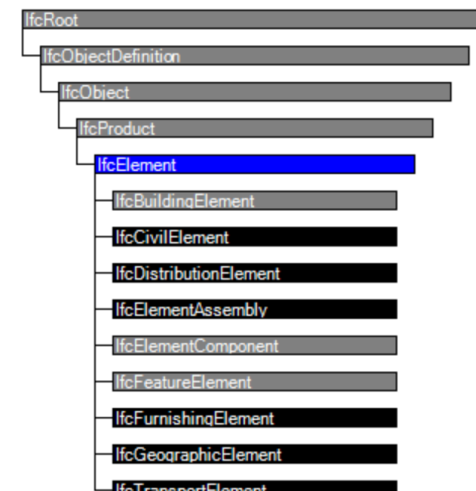
I[nstance]/T[ype]：用于指定实例或类型属性，这在当前版本中已经过时，因为系统会自动选择使用 I 还是 T

<element list separated by ‘,‘>：此处列出了将应用 Pset 的一个或多个 IFC 类，例如 “IfcWall, IfcSlab, IfcColumn”。如果 Pset 要应用于所有图元，请使用下一个更高级别的实体（使用 IfcBuildingElement 建筑图元，如墙、门等；或使用 IfcElement，以包括 Civil 和 Distribution 图元）。这可以通过在 IFC 文档中搜索实体继承来检查。

<Property Name>：Revit 中显示的属性名称

<Data type>：样板文件中列出了受支持的 IFC 数据类型，最常用的类型为文本、整数、实数、长度、体积和布尔。Revit IFC 导出目前支持 40 种 IFC 属性类型。并非 Revit 中的每个属性类型都可以直接映射到 IFC 类型，因为 IFC 使用不同的方式来指定某些单位。对于无法直接映射到 IFC 数据类型的 Revit 数据类型，可以将其映射到基本类型，如实数或整数等。这将使用 Revit 内部单位导出未转换的值。

#### 实体继承



<[opt] Revit parameter name, if different from IFC>：这是一个可选字段，如果 Revit 属性的名称也应该用于 IFC 属性，则可以跳过。如果 IFC 属性应该具有不同的名称，则可以在此处输入。

**注意：所有条目均以 <制表符> 分隔，而文件应以 UTF-8 格式保存。**

示例：

PropertySet: My.Pset.I.IfWall	
Phase.Created	Text Phase
Base.Constraint	Text
Room.Bounding	Boolean
Length	Length

My Pset	
Base Constraint	Level: Level 1
Length	5000 [mm]
Phase	New Construction
Room Bounding	TRUE

**导出参数映射表：**可将自定义 Revit 属性映射到标准映射属性，前提是数据类型相同。与用户定义的属性集类似，这也可以通过一个基于文本的映射文件实现。此文件中不包含默认的样板，但语法非常简单：

IFC Common PropertySet Name <TAB> IFC Property Name <TAB> Revit Property Name

使用这一方法，可根据项目或公司标准对 Revit 属性进行命名，并在导出时根据正确的 IFC 术语进行映射。

IFC Property Name	Revit Property Name
Pset_WallCommon Compartmentation	Brandabschnitt
Pset_WallCommon Combustible	Entflammbar

**分类设置：**这是本部分的最后一个选项，可输入模型中所用分类系统的主要信息。

要了解关于分类的更多信息，请参阅“在 Revit 中使用分类”这一章。

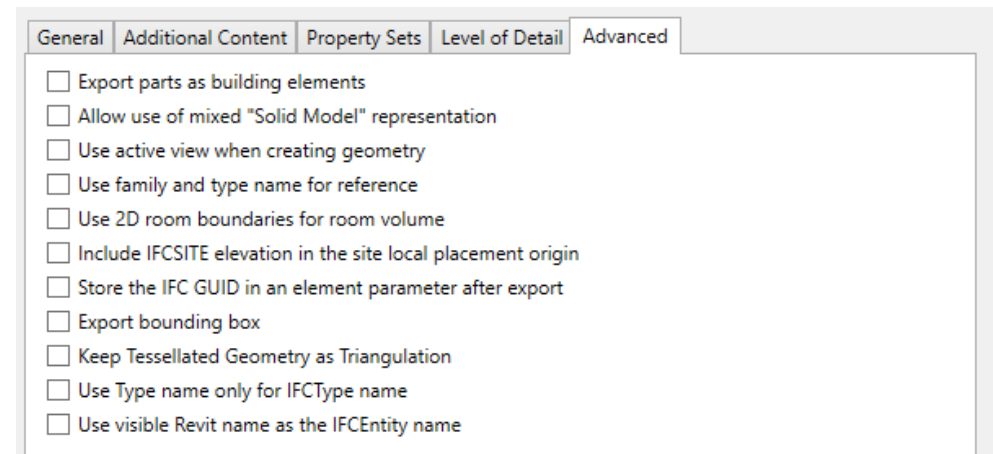
## 详细程度

此选项可供您选择细分几何图形的详细程度。默认情况下，详细程度设置为“低”。由于详细程度会影响文件大小和数据质量，因此建议在导出前评估这一选项。



## 高级

此选项卡提供了高级选项，您可以根据需要使用：



**将构件导出为建筑图元：**可在处理构件时使用。默认情况下只会导出原始图元，而激活此选项后，可将构件导出为单独的图元。

**允许使用混合的“实体模型”表示：**可导出扫描实体与 B-rep 模型的组合。IFC 数据模型中的几何对象通常由一个或多个扫描实体对象生成，或仅由 B-rep 对象生成。在 IFC 模式中，默认情况下不会启用这两种表示的组合。特别是对于更复杂的构件，这会导致文件变大或表示不正确，因为图元完全由 B-rep 对象表示。实体模型表示在一个类中组合了这两种表示，使复杂模型能够在较小的文件大小下获得最佳的几何结果。然而，需要注意的是，使用这一设置导出的 IFC 文件不再符合默认的 IFC 模式，因此必须得到所有项目参与人员的认可。对于某些用例，可能需要使用一个未经修改的默认模式进行导出。

**创建几何图形时使用活动视图：**使用当前视图的详细程度（粗略、中等和精细），并根据对象在 Revit 中的显示方式导出所有对象。

**使用族和类型名称作参照：**将影响 IFC 中的参照命名方式。默认情况下，Revit 类型名称将用于 IFC 参照。激活此选项后，族名称将与类型名称一起使用：

<input type="checkbox"/> Use family and type name for reference		<input checked="" type="checkbox"/> Use family and type name for reference	
Pset_WallCommon		Pset_WallCommon	
ExtendToStructure	FALSE	ExtendToStructure	FALSE
FireRating	F60	FireRating	F60
IsExternal	FALSE	IsExternal	FALSE
LoadBearing	FALSE	LoadBearing	FALSE
Reference	Interior - 79mm Partition (1-hr)	Reference	Basic Wall:Interior - 79mm Partition (1-hr)

**使用二维房间边界计算房间体积：**简化了基于二维空间边界的房间体积计算。在默认设置下，Revit 房间几何图形将用于确定 IFC 中的体积。

**包括场地本地放置原点的 IFCSITE 高程：**选择此选项可包括距 IFCSITE 本地放置 Z 轴偏移的高程。清除此选项可将其排除。

**导出后将 IFC GUID 存储在图元参数中：**选择此选项可在导出后将生成的 IFC GUID 存储在项目文件中。这会将“IFC GUID”参数添加到图元及其类型，以及项目、场地、建筑 GUID 的项目信息。

**导出边界框：**可导出边界框表示。

**保持细分几何体作为三角剖分：**选择此高级选项可使用与较旧的 IFC 4 参照视图查看器兼容的三角剖分方法。



仅为 IFCType 名称使用“类型”名称：为 IFCType 名称排除族名称：

<input type="checkbox"/> Use Type name only for IFCType name		<input checked="" type="checkbox"/> Use Type name only for IFCType name	
<b>Entity Type (IfcWallType)</b>		<b>Entity Type (IfcWallType)</b>	
IFC OID	403	IFC OID	391
GUID	3Zu5Bv0LOHrPC10026FoO\$	GUID	3Zu5Bv0LOHrPC10026FoO\$
GUID (readable)	e3e052f9-0156-11d5-9301-0000863f263f	GUID (readable)	e3e052f9-0156-11d5-9301-000086...
Name	Basic Wall:Interior - 79mm Partition (1-hr)	Name	Interior - 79mm Partition (1-hr)

使用与 IFCEntity 名称相同的可见 Revit 名称：将影响 IFC 中图元名称的生成：

<input type="checkbox"/> Use visible Revit name as the IFCEntity name		<input checked="" type="checkbox"/> Use visible Revit name as the IFCEntity name	
<b>Entity Information</b>		<b>Entity Information</b>	
Type	IfcWall	Type	IfcWall
Internal Type	IfcWall	Internal Type	IfcWall
IFC OID	211	IFC OID	211
GUID	2_orgaMHPBYfLiBEbZlqv6	GUID	2_orgaMHPBYfLiBEbZlqv6
GUID (readable)	becb5aa4-5916-4b8a-956c-2ce9634b4e46	GUID (readable)	becb5aa4-5916-4b8a-956c-2ce9634b4e46
Name	Basic Wall:Interior - 79mm Partition (1-hr):348711	Name	Walls : Basic Wall : Interior - 79mm Partition (1-hr)

## 在 Revit 中使用分类

### 分类基础知识

分类有助于以简单高效的方式对 BIM 数据进行分组和分类。除了基于构件类别的标准 IFC 分类外，还有各种国内和国际分类体系，例如：

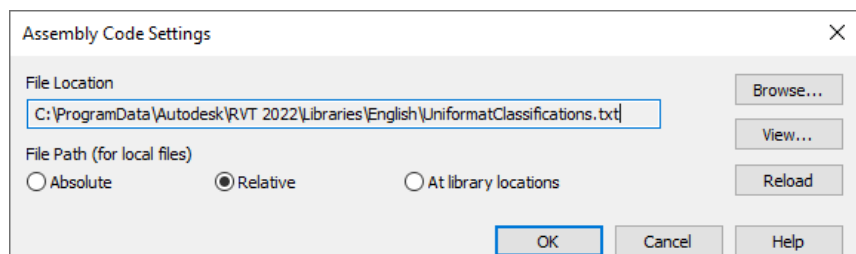
- Uniclass2015
- Omniclass/Uni Format/Master Format
- ASTM E1557
- FICM

Revit 会读写 IFC 数据，因此支持相应 IFC 模式的 IFC 分类。要导出正确的 IFC 分类，只需选择正确的“映射表”。

### Uniclass 2015

Uniclass 2015 是一个面向英国建筑业所有部门的统一分类体系。它包含统一的表格，可对所有尺度的项目进行分类。此体系于 1997 年首次发布，可按照公认的标准对项目信息进行结构化。

现行有效的 Uniclass 版本与 BIM 流程兼容。



部件代码：“管理”->“其他设置”->“部件代码”

Revit 中基于类型的默认分类体系便是 Uniclass 体系。它以文本文件的形式分发，随每个 Revit 许可一起提供。在默认安装中，此文件位于：

C:\ProgramData\Autodesk\RVT 20XX\Libraries\<your localization>\UnifomatClassifications.txt

Uniclass 分类是基于类型的分类体系，将指定给“部件代码”参数。

要导出部件代码，无需再执行其他操作，它将自动导出为 IFCClassification<sup>11</sup>

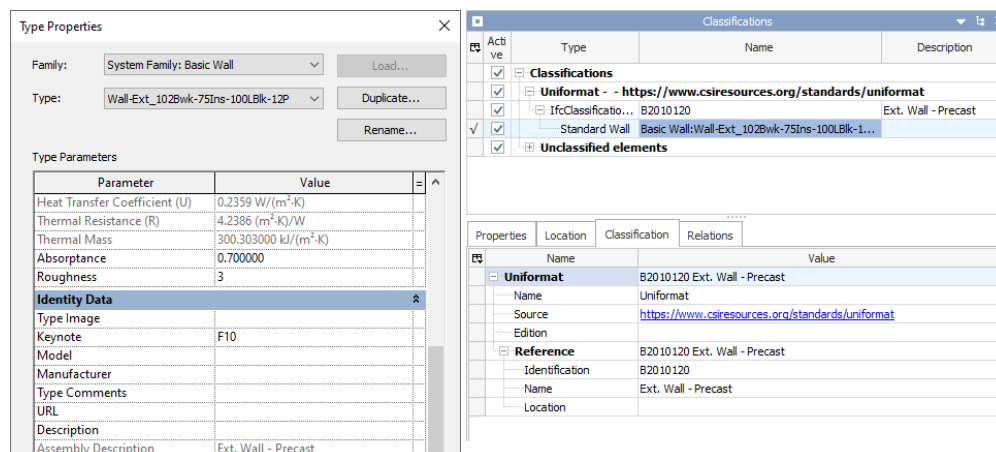


图 11：指定给墙类型系统族的部件代码

图 12：作为 IFC 实体的统一格式分类的部件代码

11. [https://standards.buildingsmart.org/IFC/RELEASE/IFC4\\_1/FINAL/HTML/schema/ifcexternalreferenceresource/lexical/ifcclassification.htm](https://standards.buildingsmart.org/IFC/RELEASE/IFC4_1/FINAL/HTML/schema/ifcexternalreferenceresource/lexical/ifcclassification.htm)

### OmniClass®

OmniClass® 分类体系是施工规范协会 (CSI) 面向施工行业推出的综合性分类体系，为电子数据库和软件提供了一个贯穿整个项目生命周期的分类结构。Revit 中分类的默认路径为：<sup>12</sup>

C:\Users\\AppData\Roaming\Autodesk\Revit\

要将 OmiClass® 分类手动导出到 Revit 对象，必须选中 IFC 导出选项：“修改设置” -> “属性集” -> “分类设置”。图 13 提供了所需的数据。由此产生的分类如图 14 所示。

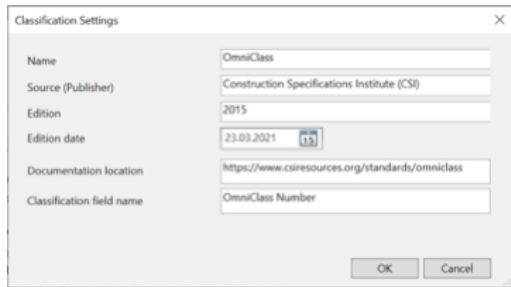


图 13: Revit 分类设置

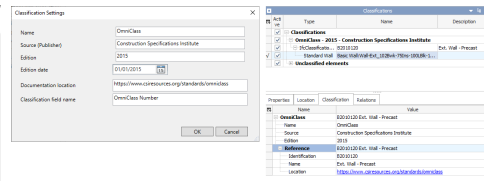


图 14: OmniClass 列已分类 - 结果为 IFC

### 通过 Autodesk Classification Manager for Revit 进行分类

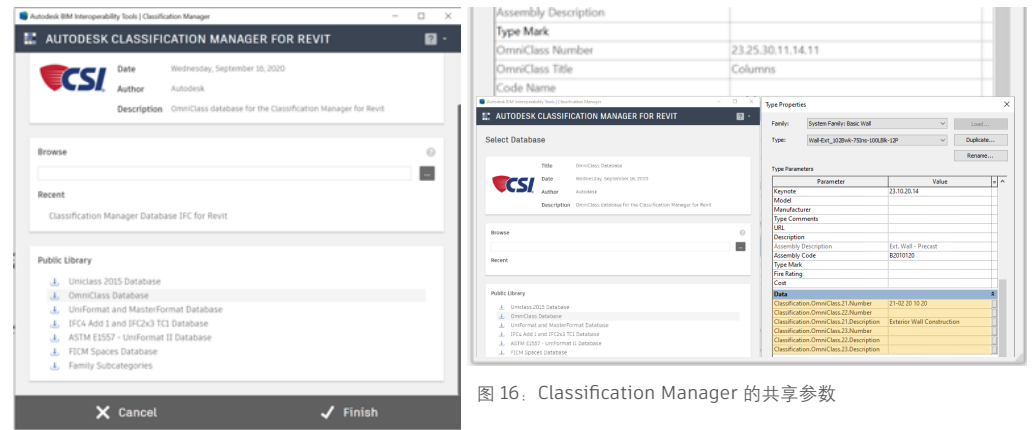


图 16: Classification Manager 的共享参数

图 15: Revit Classification Manager

另一种 Revit 图元分类方法是使用 Classification Manager for Revit。

此插件能够以交互方式对 Revit 图元进行分类。IFC 导出设置如图 13 所示，只不过必须采用共享参数名称。

要了解详细信息，请访问：<https://interoperability.autodesk.com/>

12. 注释记号表文件可以直接在 Revit 中找到：Annotate/Keynote/Keynoting Settings。注释记号是对模型图元进行注释的一种方式。Revit 准备采用这一方式；实际上，您可以直接创建一个按图纸过滤的注释记号图例；这样一来，当您在图纸中插入图例时，只会列出在该图纸中定义的注释记号，从而将其用作简单的注释工具。注释记号表指的是 **Masterformat**，这是 CSI 发布的另一个分类列表。最新版本为基于 Masterformat 2004 的版本。Masterformat 标准以及 OmniClass 均用于列出工作结果。它还结合了施工实践。

### 高级分类/多个分类

对于 Revit 中的分类，基本上每个文件只能采用一个分类体系。使用以下共享参数，可在一个模型中添加多个分类体系。<sup>13</sup>

多个分类共享参数的名称如下<sup>14</sup>：

- ClassificationCode
- ClassificationCode(2)
- ClassificationCode(3)
- ClassificationCode(4)
- ClassificationCode(5)
- ClassificationCode(6)
- ClassificationCode(7)
- ClassificationCode(8)
- ClassificationCode(9)
- ClassificationCode(10)

建立分类的语法为：

[ClassificationName]Code:Title

示例：

[Maturity]01:STATUS

Active	Type	Name	Description
<input checked="" type="checkbox"/>	Classifications		
<input checked="" type="checkbox"/>	ByHeight - -		
<input checked="" type="checkbox"/>	IfcClassificationReference	2.00	Height
<input checked="" type="checkbox"/>	ByLength - -		
<input checked="" type="checkbox"/>	IfcClassificationReference	3.00	Length
<input checked="" type="checkbox"/>	IfcClassificationReference	4.00	Length
<input checked="" type="checkbox"/>	IfcClassificationReference	5.00	Length
<input checked="" type="checkbox"/>	ByMaterial - -		
<input checked="" type="checkbox"/>	IfcClassificationReference	CONCRETE	WALL

图 17: IFC 中的多个分类

Data	
ClassificationCode	[Maturity]01:STATUS
ClassificationCode(2)	[ByMaterial]CONCRETE:WALL
ClassificationCode(3)	[ByHeight]2.00:Height
ClassificationCode(4)	[ByLength]5.00:Length
ClassificationCode(5)	[ByPrice]Low:Price
ClassificationCode(6)	
ClassificationCode(7)	
ClassificationCode(8)	
ClassificationCode(9)	
ClassificationCode(10)	

图 18: Revit 中的多个分类

13. 现在，多个分类的使用受限。不支持 IfcClassification 属性，包括来源、版本、版本日期、名称、说明、位置和参照标记。

14. ClassificationCode(1) 无效。

## 其他用例与提示

### 将楼板导出为 IFC

Revit 楼板大多使用两个独立的图元进行建模：用于楼层的承重底板和用于每个房间的楼板结构。

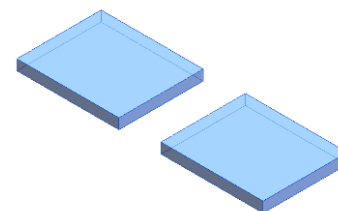
进行 IFC 导出时，所有底板在默认情况下均指定给 IFCSlab 类。就 IFC 而言，这可能是一个错误的分类，鉴于关联的属性集不同，底板应导出为 IFCSlab 类，而楼板应导出为 IFCCovering 类。

为此，Revit 中的楼板指定为 IFCEXportAs “IFCCovering” 和 IFCEXportType “FLOORING”。或者，可以使用语法 IFCCovering.FLOORING 将类和类型均指定给 IFCEXportAs。

### 对底板建模以导出为 IFC

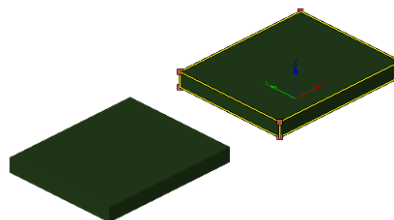
即使在 Revit 中可通过未连接的多边形创建楼板/天花板几何图形，但在模型中应予以避免，因为在导出为 IFC 时，这些连接的 Revit 对象在 IFC 中将被视为独立图元，所有属性值均会指定给所产生的各个 IFC 对象。

Dimensions	
Slope	
Perimeter	33600.0
Area	34.960 m <sup>2</sup>
Volume	16.431 m <sup>3</sup>
Elevation at Top	0.0
Elevation at Bottom	-470.0
Thickness	470.0
Identity Data	
Image	
Comments	
Mark	
Phasing	
Phase Created	New Construction
Phase Demolished	None



Default:

Entity Information	
Type	IfcSlab[Floor]
Internal Type	IfcSlab[Floor]
IFC OID	325
GUID	0sVQDJH5bAmuGSchlJzfHc
GUID (readable)	367da353-4459-4ac3-843f-9a...
Name	Floor:Floor-Grnd-Bearing_65...
Description	?
Object Type	Floor:Floor-Grnd-Bearing_65...
Predefined Type	FLOOR
Layer Name	A-FLOR-___-OTLN
Color	Color [R:165, G:42, B:42, A:255]



Element Properties	
Name	Value
PropertySets from entity	
Pset_ProductRequirements	
Pset_ReinforcementBarPit...	
Pset_QuantityTakeOff	
Pset_SlabCommon	
Pset_ElementShading	
BaseQuantities	
GrossArea	17.48 [m <sup>2</sup> ]
NetArea	34.96 [m <sup>2</sup> ]
NetVolume	16.431 [m <sup>3</sup> ]
Perimeter	33600 [mm]

Customized:

IFC Parameters	
IfcExportAs	IfcCovering.FLOORING

or

IFC Parameters	
IfcExportAs	IfcCovering
IfcExportType	FLOORING

Entity Information	
Type	IfcCovering
Internal Type	IfcCovering
IFC OID	209
GUID	0sVQDJH5bAmuGSchlJzfH2
GUID (readable)	367da353-4459-4ac3-843f-9a...
Name	Floor:Floor_Timber_22Cbd-2...
Description	?
Object Type	Floor:Floor_Timber_22Cbd-2...
Predefined Type	FLOORING
Layer Name	A-FLOR-___-OTLN
Color	Color [R:127, G:127, B:127, A:...

Properties	
Provision for void Cutout 500x1000	
Generic Models (1) Edit Type	
Comments	
Mark	
Phasing	
Phase Created	New Construction
Phase Demolished	None
IFC Parameters	
IfcExportAs	IfcBuildingElementProxy
IfcObjectType	PROVISIONFORVOID

Element Properties	
Name	Value
Entity Information	
Type	IfcBuildingElementProxy
Internal Type	IfcBuildingElementProxy
IFC OID	614
GUID	0nvWUihvEjuAaNVjygL4
GUID (readable)	31e607ac-b2be-4eb7-82a4-5d...
Name	Provision for void-test:Cutout...
Description	?
Object Type	PROVISIONFORVOID
Layer Name	A-GENM-___-OTLN

图 19：楼板的 IFC 实体和预定义类型

## 剪切洞口

在集成设计过程中，代理对象主要用于剪切洞口的初步设计和协调。在 IFC 中，这些对象称为“空心配置”的对象，可与字母数字信息和尺寸标注一起在域模型之间交换。

代理图元可能源自原生 Revit 孔图元，也可能是简单的空心族。

要导出空心配置的对象，可将原生 Revit 对象指定为 IFCExportAs “IFCElementProxy” 和 IFCObjectType “PROVISIONFORVOID”。

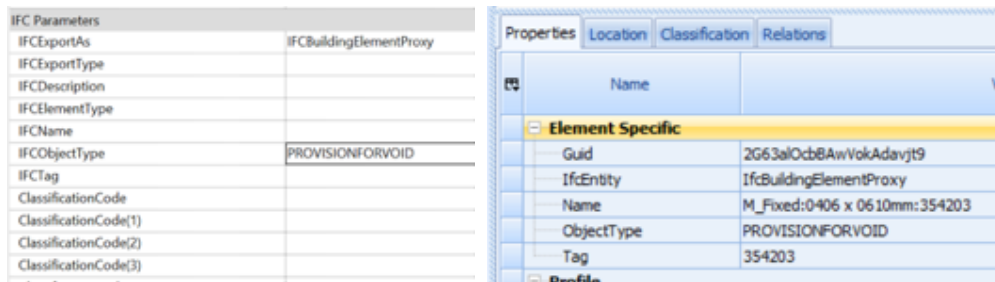
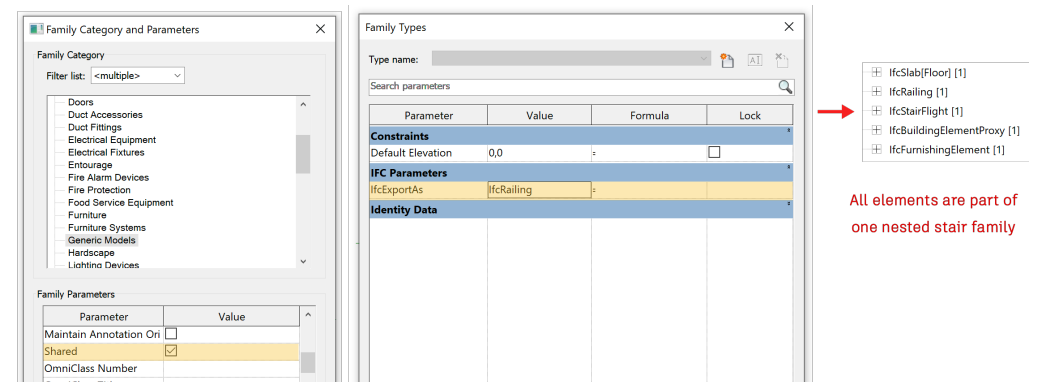


图 20: 空心配置

## 嵌套族

导出嵌套族时，所有图元在默认情况下均将指定给一个类/实体。然而，可以将嵌套族作为实体单独分类。为此，需要共享这些族，且它们必须具有自己的 IFCExportAs 参数：



## 指定部件

部件对于结构梁系统、梁轴网和钢筋笼等构件的更高级别分组至关重要。与 Revit 组不同，部件将作为 IFCElementAssembly 类导出为 IFC，并指定有更高级别的属性。

要导出图元部件，可将原生 Revit 对象指定为 IFCExportAs “IFCElementAssembly” 和 IFCObjectType “RIGID\_FRAME”。

## 分区

要从 Revit 导出 IFC 分区，可通过一组指定给房间对象的共享参数实现。

IFC 中的分区是指可分类的空间的集合。从 Revit 导出分区分类时，每个模型只能有一个分类。

Revit 中的分区分类参数为“ZoneClassificationCode”。语法与高级分类/多个分类的语法相同。

ZoneClassificationCode: [ZoneClassificationName]Code:Title

Room Name and Classification			Zone Classification			Zone Name, ZoneDescription, ZoneObjectType		
A	B	C	D	E	F			
Name	ClassificationCode(3)	ZoneClassificationCode	ZoneName	ZoneDescription	ZoneObjectType			
Room	{ROOMS}01.01.01.Single Apartment	{ZONE}01.ZoneClass	TOP1	TOP 01	Small			
Room	{ROOMS}01.01.02.Double Apartment	{ZONE}02.ZoneClass	TOP2	TOP 01	Medium			
Room	{ROOMS}01.01.02.Double Apartment	{ZONE}02.ZoneClass	TOP3	TOP 01	Big			

G	H	I	J	K	L	M	N	O
ZoneName 2	ZoneDescription 2	ZoneObjectType 2	ZoneName 3	ZoneDescription 3	ZoneObjectType 3	IFCDescription	IFCName	IFCObjectType
Apartment 01	Apartment 01 in Building 01	Single-Apartment	Site 01	Building 01 at site 01	Family Home	Room Description A	Room Number	Room-Object1
Apartment 02	Apartment 02 in Building 01	Double-Apartment	Site 02	Building 01 at site 02	Family Home	Room Description B	Room Number	Room-Object2
Apartment 02	Apartment 02 in Building 01	Studio	Site 02	Building 01 at site 02	Practise	Room Description C	Room Number	Room-Object3

Zone Name 2, ZoneDescription 2, ZoneObjectType 2			Zone Name 3, ZoneDescription 3, ZoneObjectType 3			Room Parameters		
--	--	--	--	--	--	-----------------	--	--

分区相关参数可提供更详细的分区信息。上图列出了可导出的 Revit 参数。

房间名称和分类指定给 IFC 中的房间。

ZoneClassificationCode 是分区的分类参数。

ZoneName、ZoneDescription 和 ZoneObjectType 用于定义分区对象。共有三个独立的分区定义（ZoneName、ZoneName 2 和 ZoneName 3）。

注意：IFCName 参数将映射到 Number，而 IFCDescription 将映射到 IFCSpace - Description。

## 附录

### Dynamo 和 IFC

本附录提供了一些用于准备或增强 IFC 数据的 “Dynamo” 示例。

#### 向 Revit 添加分类

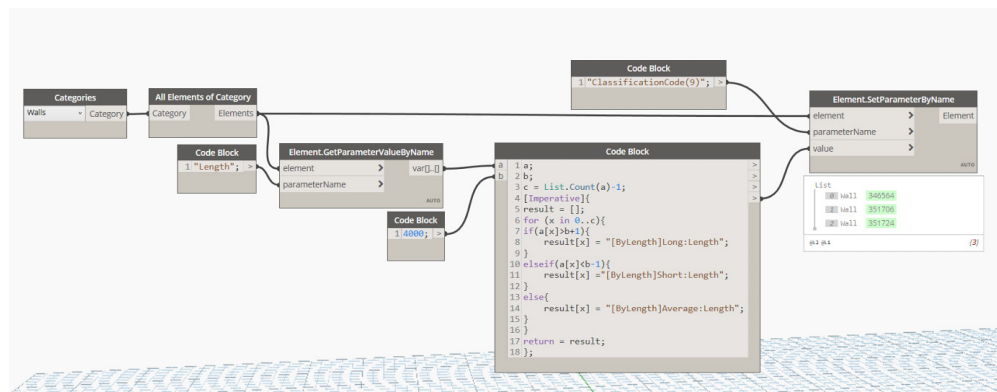
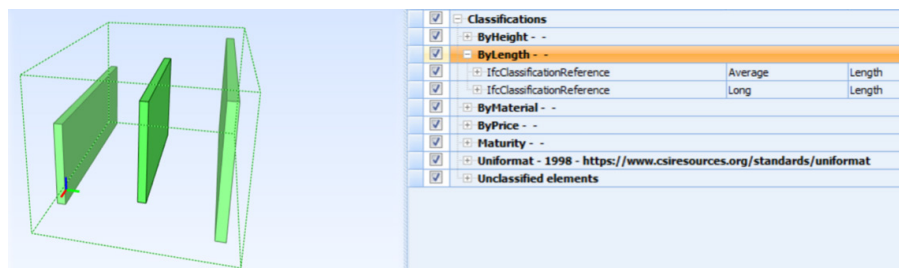


图 4：用于墙分类的 Dynamo 脚本

说明：

从 Revit 模型中选择图元；在命令式代码块中评估分类结果。请注意，**[ByLength]** 为分类的名称，而标题和**长/短/平均**为相应的代码。

结果将推入 “ClassificationCode (9) ” 参数。



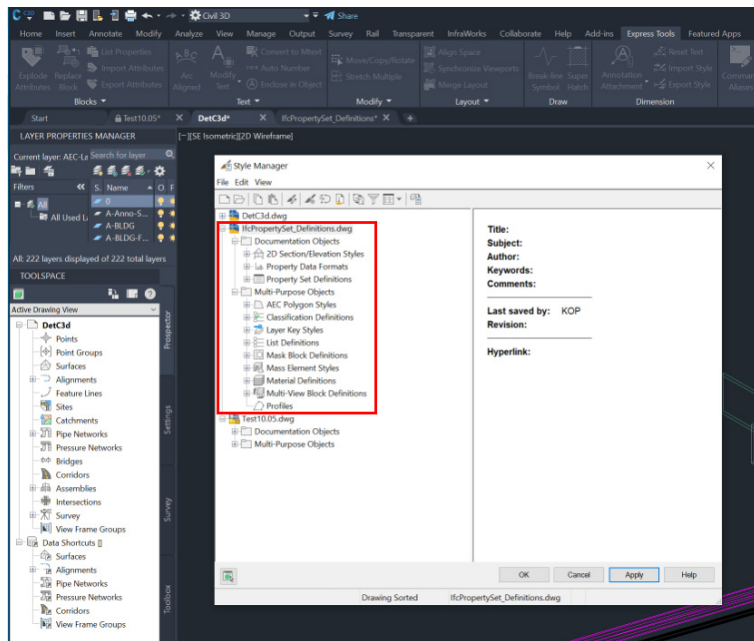


## 从基于 AutoCAD 的产品导出为 IFC

将数据从基于 AutoCAD 的产品（如 Civil 3D、AutoCAD MEP 等）导出为 IFC 时，有一些基本注意事项。

AutoCAD 数据必须为结构化数据，才能进行 IFC 导出。此操作在“样式管理器”中完成（AutoCAD 命令：“STYLEMANAGER”）。此命令将打开一个对话框，用于创建和编辑可定义图形中对象外观的样式，最重要的是用于 IFC 导出。

AutoCAD 中的样式用于定义对象（如墙、管道和窗）、文档对象（如二维剖面/立面、属性数据格式和属性集定义）以及多用途对象（如图层索引、分类设置和材质定义）。

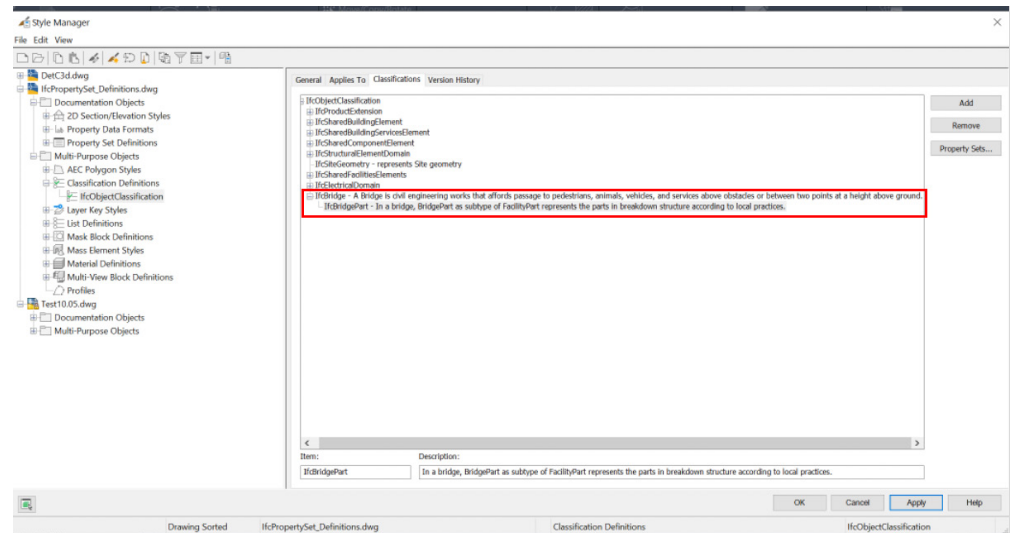


## 创建和指定 IFC 类

首先，选择分类所适用的对象。然后（或在此之前），创建分类。“分类”窗口右上角提供了一些按钮，用于添加和删除类或为类指定属性集。

IFC 类的结构与各自的 IFC 模式对应。子类可通过选择父类来创建。

现在，可以为每个类指定所选的属性集。



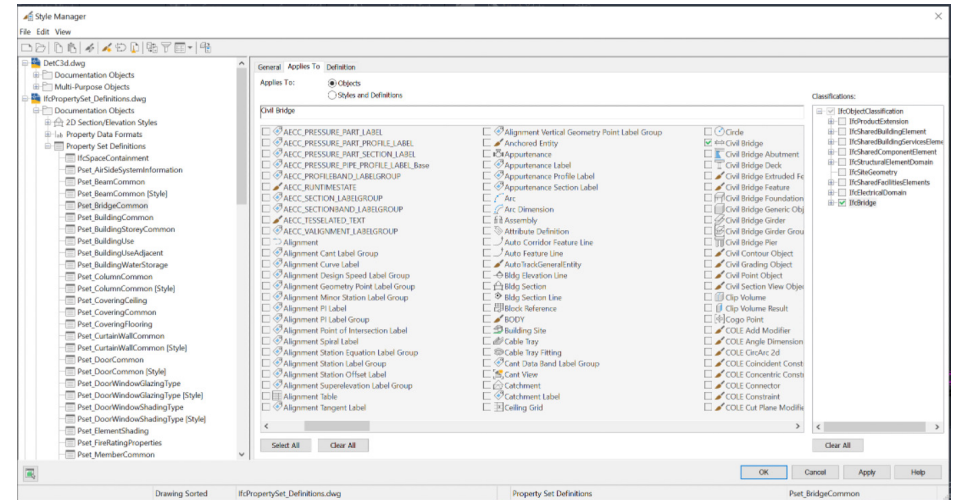
### 属性、属性数据格式和属性集

创建属性时需严格遵循一些规则。

创建新属性前，必须检查此属性所需的数据格式是否已经存在。如果没有，必须在“属性数据格式”（关联菜单 -> “新建”）中创建一个新的样式。  
16

Style	Description	Ignore D
lab Area	Area calculations	No
lab Case - Sentence	Sentence case text	No
lab Case - Upper	Upper case text	No
lab Fixed Note - Text	Fixed Note - Text	No
lab GradingObjects-Degree	GradingObjects-Degree	No
lab GradingObjects-Length	GradingObjects-Length	No
lab GradingObjects-Percentage	GradingObjects-Percentage	No
lab GradingObjects-RunOverRise	GradingObjects-RunOverRise	No
lab GradingObjects-Toggle	GradingObjects-Toggle	No
lab GradingObjects-Volume	GradingObjects-Volume	No
lab IfcAbsorbedDoseMeasure	IfcAbsorbedDoseMeasure A measure of the absorbed radioa...	No
lab IfcAccelerationMeasure	IfcAccelerationMeasure A measure of acceleration.	No
lab IfcAmountOfSubstanceMeasure	IfcAmountOfSubstanceMeasure An amount of substance measure ...	No
lab IfcAngularVelocityMeasure	IfcAngularVelocityMeasure A measure of the velocity of a bo...	No
lab IfcAreaMeasure	IfcAreaMeasure An area measure is the value of th...	No

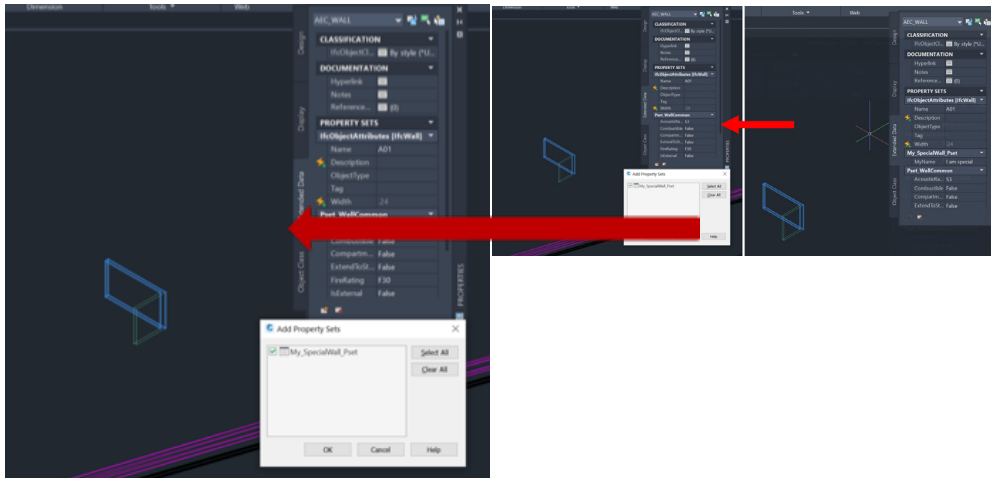
现在，新属性集创建完毕，将为其指定类和属性。（在本手册中为“Pset\_BridgeCommon”。）



- 常规：定义 Pset 的名称并添加说明
- 应用于：指定对象（民用桥梁）
- 分类：选择 IFC 类 (IFCBridge)
- 定义：添加属性。

Name	Description	Type	Source	Default	Units	Format
GrossAreaPlanned	GrossAreaPlanned	Real		0.0000...	Square meters	Area
Reference	Reference	Text				IfcIdentifier

现在，单击“添加属性”图标（图中红色箭头所示）  
并选择 Pset（此处为 My\_SpecialWall\_Pset），即可将这些属性指定给 AutoCAD 对象。



左侧：指定 Pset My\_Special\_Wall；右侧：Pset My\_Special\_Wall 已指定，值为 “I am special”

现在，数据可导出为 IFC。

在导出过程中

- 选择相应的 IFC 模式
- 选择要导出的对象类型
- 选择资源和指定方式

最后，导出所选的数据。



## IFC 项目的数字质量管理 - TÜV 南德意志集团 Tobias Schmidt

对于希望依靠 BIM 实现通用项目实施的委托方或建筑业主来说，IFC 的使用非常有趣。在很多项目情况下，需要实施通用的 IFC 应用策略：紧急的项目审批使委托方无法制定单独的 BIM 策略，或技术采购部门在实施不同的软件解决方案时已经确定了最佳项目可行性和最大项目吸引力，或项目信息经理已经确定了一个依赖于公认标准的信息模型。

对于 BIM 项目中的委托方和受托方而言，IFC 作为一种数据媒介，有望简化整个信息管理过程：通过软件应用创建的信息模型也可用于其他系统，而无需投入大量人力复制、修复或完善信息模型。当整个项目策略和整个信息管理过程均进行调整以完全支持将 IFC 作为交付成果并将开放 BIM 作为项目工作文化时，这一质量措施即可实现。

TÜV 南德意志集团认识到，委托方表示“我想要使用 IFC”，以及受托方（如技术顾问和承包商）在创作和协调时单击“导出为 IFC”，并不是足以实现最佳 IFC 的质量措施。研究 ISO 19650 中所述的信息管理可以发现，IFC 不仅仅是一种数据格式。IFC 意味着在所有利益相关方之间，在所有行业以及整个项目或资产生命周期中，建立一个结构完善、协调一致的工作文化。

### TÜV 南德意志集团认为，项目工作文化和 IFC 实施有三大要素

如果作为委托方的建筑业主能够在 BIM 项目中与所有受托方共同制定一个可靠的信息管理流程来支持高质量的 IFC，则可以确保 IFC 的成功应用：

- **在项目启动时明确 IFC 的整体要求：**由于 IFC 标准现已发展成为一个大型的“数据生态系统”，具有多种选择和不同特点，因此委托方应制定和整合“交换信息要求”，以确定应使用 IFC 实现项目和建筑文档的哪些用例；由 BuildingSmart 开发的模型视图定义（请参见“BuildingSmart MVD 数据库”）使用户能够深入了解 IFC 能够以最佳方式支持哪些项目主题；在每一个可靠的 IFC 项目中，模型视图定义都是“交换信息要求”的一部分，因为这些 MVD 帮助受托方记录了来自各个行业和专业模型的所需图元；MVD 创建了精简且结构清晰的信息要求，无需传递、管理和更新来自所有相关行业模型的所有（因此也是不必要的）信息；委托方和受托方均可通过定义的 MVD 从客观的 IFC 模型中获益，因为这减少了信息数量并提高了信息质量，能够为所有项目参与者提供更好的支持
- **以协调一致的方式构建 IFC 模型：**要通过 IFC 实现协调统一的信息模型创作、协调和交接，BIM 执行计划 (BEP) 和总信息交付计划 (MIDP) 至关重要；通过 BEP 和 MIDP，受托方能够践行“交换信息要求”的组织和程序性 BIM 主题，并在技术层面上记录所有行业和规划参与者如何创建协调一致的“可联合”IFC 模型并加以使用；BEP 和 MIDP 还促进了所有规划参与人员在创建模型之前就具体设置和流程进行协调（如 BIM 协调），可确保所有行业、所有相关方都能够推动高质量的 IFC 导出，从而获得整体效果最佳的 IFC 模型；需要特别注意的是：

- 应以各自的原生格式共同商定项目设置和建模方法，这将直接影响 IFC 行业或技术模型的质量，MVD 就是通过这种方法实现的
- 以相互协调的方式确定导出设置，以便每个 IFC 行业模型都能够以最佳方式整合到整体模型中，在节省时间的同时实现最佳的数据完整性（例如，用于碰撞检查、数量和成本核算、AsBuilt BIM 文档等）

**共同进行信息管理，而非“相互指责”**：在项目处理过程中，IFC 的主要应用模式为由用户共同创建、协调和使用基于 IFC 的信息模型；需要注意的是，所有行业都应就 IFC 的“共同要素”进行合作，无论是在技术层面还是在整个项目层面，以便各受托方和相关专家能够相互支持，打造最佳的 IFC 项目；对于受托方和委托方，在使用 BIM 时，重点在于可行性、附加值和实施质量，以及更高的生产力和数据完整性。ISO 19650 提及了以下方面：

- 受托方定期将所定义的信息模型提交给委托方，供委托方验收
- 定期检查参照信息和共享资源的可用性；生成信息；完成质量保证检查；审查信息（模型）并批准共享

借助这三大“IFC 最佳实践”，委托方和受托方可以为在项目中联合应用 IFC 奠定坚实基础。至关重要的一点是，在所有行业 and 项目阶段之间协调 IFC 版本（IFC 2.3、IFC 4.X）、模型视图定义和专用用例（包括相关的联合 BIM 模型导出设置）等基本参数，以便在技术层面以及整个项目 workflow 层面实现最佳的 IFC。

### IFC 在项目中的应用 - TÜV 南德意志集团用于 BIM 模型自检的“IFC 质量要点”

根据 TÜV 南德意志集团 BIM 团队在全球范围内对 IFC 项目进行审计和咨询的经验，要实现最佳的 IFC 质量，需要注意三大检查类别和“IFC 质量要点”。如果项目同时具备这些要素，则说明一个真正开放的 BIM 文化的重要方面（当然不是所有方面）都得到了妥善实施：

## 1. 模型结构和模型完整性

在所有行业间保持一个特定于项目的统一模型结构至关重要，因为这是所有行业模型相互协调的基础，在使用 IFC 时尤为如此，例如创建联合模型并以此为基础执行跨行业用例，如工程量估算、冲突检测等。仅当项目中涉及的所有技术模型均按照 ISO 16739 和 BuildingSmart 命名法采用统一的模型结构（包括参数 IFC PSet 的命名）时，在创建联合模型时才能尽可能减少数据损失。

如果不能满足这方面的要求，则会导致 IFC 模型无法用于自动设计审查和技术应用，例如消防、管道和污水网络计算、能源计算等。

以下是 TÜV 南德意志集团提供的一些 IFC 检查最佳实践，有助于确保您的 IFC 模型在所有行业间保持特定于项目的统一模型结构：

- 项目基点相同：每个领域模型应具有相同的全局定位。这体现在模型的经度、纬度、底部高程和正北方向旋转上；共同的项目基点是第一个质量检查项，对领域模型的协调和“可检查性”最为重要
- 每个项目只应有一个（而非多个）IFCsite 实例；如果一个项目由多个 IFCsite 实例定义，则无法保证行业模型根据一个物理测量点进行协调

- 确保在所有行业模型中只有唯一的 GUID，并且在某个 IFC 模型中没有重复的 GUID，即没有重复的图元，否则可能导致工程量估算错误和冲突处理责任不明确等
- 在几何完整性方面，检查 IFC 模型中是否存在整合（或遗留）的二维对象，因为二维图元无法准确表示各个图元的几何图形，而且二维图元在冲突检测期间也无法显示
- 检查轴网线：每个领域模型均应包含轴网线；未经单一轴网系统标准化的行业模型无法保证紧密结合
- 确保没有 ProxyElement，因为构件应指定为 IfcBuildingElementProxy；请考虑使用合适的 IfcEntity，以便妥善执行其他用例，如消防概念、管道/风管计算和成本核算等

## 2. 建模指南

在所有项目 IFC 模型间采用统一的建模指南至关重要，这是妥善进行工程审查的基础，需要将一个同类 IFC 设置传递到制造和工程环节。

当一个项目的各个行业模型具有不同的结构时，可能无法采用统一的建模指南，这会造成不一致、不连贯的 IFC，进而导致无法成功地连续使用 IFC 模型（例如在施工阶段和运营阶段）。

通过以下几项检查，可以轻松地在建模层面实现行业间一致的 IFC 质量：

- 相对于主体楼层的合理偏移：检查所有构件是否在相对于主体楼层的合理偏移范围内创建；在使用项目相关设置进行指定和代码检查时，可以轻松检查这一项
- 验证所有主体构件是否具有几何图形：被其他构件分解的构件必须具有几何表示
- 检查主体构件是否缺失几何图形：分解为其他构件的构件可能没有几何表示
- 限制范围内的楼层高度（每个项目的自定义资产）也是检查模型完整性的一项标准，建议检查中间底板之间的距离（= 楼层高度），以查看底板（通过使用 IFC 实体分类选择）是否表明该项目是按楼板建模的；这是一个至关重要的常规 VDC 项目
- 检查材质图层厚度之和（构件总厚度）；这项检查可确保材质图层厚度之和等于构件总厚度；如果构件的材质图层总厚度不等于其几何厚度，则表明在构件的原始建模或导出过程中可能存在问题。

- 避免过于详细的大型模型：检查几何表示是否过于详细，以确保项目不包括几何图形过于详细的构件，这可以通过过于详细的 LoD（开发级别）判断，否则会导致创作或协调速度非常缓慢，从而影响项目生产力；您可以根据项目情况设置最大多边形数量，然后让“模型检查”运行各个构件，以检测每个对象构件的多边形数量是否过多
- 检查分解构件材质是否（仅）在构件层面定义以表明分解构件（部件）；这对提取正确的工程量估算结果和正确的材质定义至关重要
- 分析 IFC 模型中的机电构件是否至少与另一个机电构件相连，以及任何机电构件是否为系统的一部分；本规则旨在检查所有机电构件是否至少与另一个机电构件相连，这可以表明不存在未确定或未连接的项目，否则将影响工程量估算，还可以表明 IFC 模型中有些图元尚未成为一个协调良好的运行系统的一部分
- 建筑模型应具有空间：检查建筑模型是否包含空间构件，并且每个“空间”是否具有唯一的标识符；这可以避免空间重复或重叠，反过来又可以避免后续出现错误的空间量和 Roombook
- 复杂墙体中的洞口应与墙体相关，而非与某个图元相关；IFC 模型中的洞口如果没有完全穿过多层墙体，则可能会产生不协调的洞口

### 3. 信息要求

统一且条理清晰的信息要求是在行业间以及后续生命周期阶段进行可靠信息传递的基础，例如，用于基于 BIM 的招标、维护优化或可维护性设计、进度管理等。

质量错误可能会造成信息不协调、遗漏或不一致，进而导致误解、重复和信息有误，对于涉及多个领域以及与多个生命周期阶段相关的 BIM 用例（如设计到施工用例或施工到运营用例）来说尤为如此。

为在信息要求方面获得 IFC 质量基础，请检查以下项目，并以特定于项目的其他验证来扩展检查清单：

- 正确的 PSet：检查 IFC 行业模型的各个图元是否定义了正确的 PSet，并且最初没有添加或覆盖任何单独的属性名称或属性内容；原始 BuildingSmart IFC 文档中定义的 PSet 可确保 BIM 项目顺利展开并协调良好，从而避免某些行业模型最初由 BuildingSmart PSet 开发，而其他模型可能已经包含独特的属性结构或内容，这将导致联合模型层级的一般信息交换和信息处理失败；您可以检查构件是否包含以“Pset\_”开头的默认 PropertySets，并仔细查看所有在开头缺少“Pset\_”的项目
- 由 IfcEntity 定义每个构件，这对日后根据 ISO 16739 正确使用 IFC 分类至关重要；在 IFC 中，图层和分类并不是属性，实际上是“实体”；所有实体均通过这些重要的关系与 IfcBoiler、IfcBuilding 或 IfcSpace 等其他实体相关联
- 检查每个构件是否由 IfcType 定义，错误或未定义的类型会导致大多数 BIM 用例失败
- 确保每个构件均有一个“IFCAsset”属性；未由 IFC 资产 ID 参数定义的图元无法被设施管理部门识别
- 验证每个构件是否按照 BuildingSmart 的 IFC 类型分类进行分类
- 在属性层级，确保每个构件均有名称、类型和材质信息，这可以通过清晰的人机可读信息提高 IFC 项目信息模型的可用性，因此对实现工作流自动化（如使用其他程序或 Model Checker）至关重要
- 将项目的“交换信息要求”和“BIM 执行计划”与所应用的通用 IFC 属性进行交互检查，以确保每个必需的 IFC 属性均存在并正确填写，例如
  - AcousticRating
  - FlammabilityRating
  - ThermalTransmittance
  - LoadBearing
  - FragilityRating
  - FireRating
  - 等等
- 为实现准确的工程量估算，请检查每个行业模型和每个相关图元中是否存在相关的 IFC QuantitySets，并且 QuantitySets 的内容是否由创作工具准确定义（而非人工定义！）；例如，为了从模型中提取正确的墙体工程量估算结果，应检查以下设置：Pset\_WallCommon.LoadBearing = TRUE 且 Pset\_WallCommon.IsExternal = TRUE；同时进行以下一致性检查：
  - 构件属性必须一致
  - 构件厚度必须一致
  - 构件轮廓必须一致
  - 门窗尺寸必须一致
  - 门窗顶部高程必须一致
  - 墙高必须一致
  - 柱长必须一致
  - 构件高程必须一致
  - 等等



- 检查所有项目相关 Pset\_BuildingStoreyCommon 属性是否均已就绪：作为一项基本的虚拟设计和施工 (VDC) 措施，应按楼层开发每个 IFC 模型，以推动设计分析与文档用例的发展；考虑到 Pset\_BuildingStoreyCommon 的多个建筑属性直接在 IfcBuildingStorey 实例中处理；重要的 Pset\_BuildingStoreyCommon 属性示例如下：
  - EntranceLevel
  - AboveGround
  - GrossAreaPlanned
  - NetAreaPlanned
  - SprinklerProtection
  - SprinklerProtectionAutomatic
  - Pset\_BuildingStorey BaseQuantities
  - NominalHeight
  - GrossFloorArea
  - NetFloorArea
  - GrossVolume
  - NetVolume
- 确保所有相关的 IFC 模型均包含 Compartmentation：检查构件是否具有 Compartmentation 属性；缺失属性表明...

---

#### 作者简介：

Tobias Schmidt 是 TÜV 南德意志集团的著名专家和 BIM 总监。TÜV 南德意志集团致力于通过全球专家网络提供 BIM 咨询和顾问服务，这些专家不仅具有建筑技术知识以及业务和流程咨询专业知识，而且积累了丰富的技术经验。TÜV 南德意志集团的 BIM 咨询和顾问可帮助您确定兼具可行性与盈利性的最佳 BIM 战略，以实施适当的“交换信息要求” (EIR) 和“BIM 执行计划” (BEP)，并优化建筑的资本支出和运营支出。

## EIR 与 BEP<sup>17</sup> - Peter Kompolschek

EIR 和 BEP 是在项目中成功投标和实施 BIM 的核心文档。

在介绍订购流程之前，必须澄清一些基本术语：<sup>18</sup>

- 委托方 - 信息的接收方。
- 受托方 - 信息的提供方。<sup>19</sup>
- 委托 - 提供信息的商定指示。

通常情况下，信息交付委托流程分为三步，如图 1 所示。



图 1: 投标流程的工作流

### 投标邀请

委托方为每个主受托方的委托确立交换信息要求 (EIR)，并酌情考虑组织信息要求 (OIR)、资产信息要求 (AIR) 和项目信息要求 (PIR)。

向受邀为相关委托业务投标的每个潜在主受托方发出一份 EIR。

### 投标响应

潜在的主受托方响应 EIR 并提供 BIM 执行计划 (委托前)。

### 委托

主受托方中选后，该方确认 BIM 执行计划，并提供一套关于在其责任范围内执行交付成果的确切信息。

### EIR (交换信息要求)

委托方制定交换信息要求，列出所有适用的信息要求。交换信息要求将提供给潜在的受托方。

信息要求可以说明需要信息的原因、具体内容、交付时间、获取方式以及使用人员 (组织信息要求 (OIR)、资产信息要求 (AIR) 或项目信息要求 (PIR))。告知受托方需要这些信息的原因，有助于他们根据委托方的业务需求，采用创新的信息制作和交付方法。进一步的信息要求应简要说明目的、预期结果和/或需要满足的委托方的业务需求和信息需求。

### BEP (BIM 执行计划)

BIM 执行计划文档由主受托方与委托方和其他受托方商定后更新，以确认具体项目应使用的具体信息。信息交付策略应阐明主受托方为满足 EIR 中规定的信息要求所采用的方法。此外，交付团队的结构 (受托方概览) 和/或交付团队内的任务小组细分是交付策略的一部分，也应在此说明。交付团队的信息交付策略还应包含协同制作信息的一系列目标/目的。

### 作者简介:

Peter Kompolschek 是一位奥地利的建筑师，也是一位著名的 BIM 专家。除了担任大型建筑和基础设施公司的 BIM 顾问和经理之外，他还是奥地利标准协会、CEN 和 CELEC 等多个标准机构的活跃成员。

17. 摘自“Guideline for the implementation of BIM Execution Plans (BEP) and Exchange Information Requirements (EIR) on European level based on EN ISO 19650-1 and -2” (基于 EN ISO 19650-1 和 -2 的欧洲 BIM 执行计划 (BEP) 和交换信息要求 (EIR) 实施指南)

18. 所有术语和概念均符合 EN ISO 19650-1 和 -2 标准的要求

19. 应为每个交付团队指定一个主受托方

