

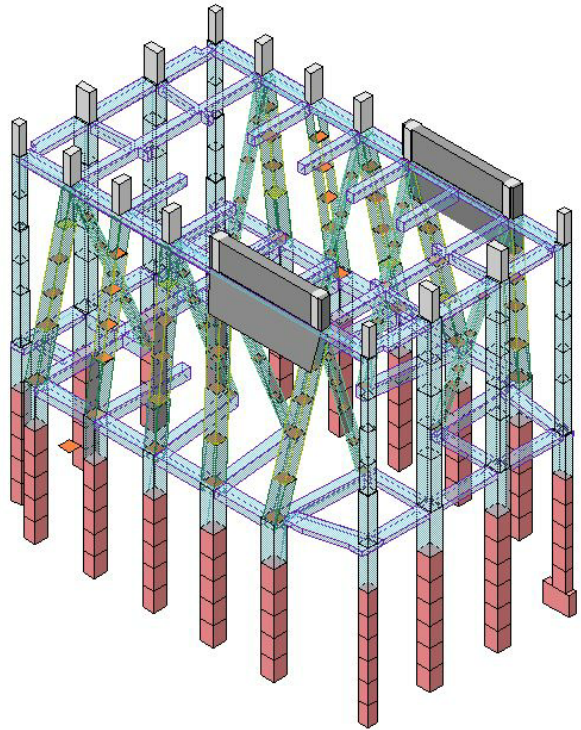
GRAPHISOFT  
**ArchiCAD®**

IFC  
Reference Guide

GRAPHISOFT®



# IFC Reference Guide



**Graphisoft**

## **ArchiCAD IFC Reference Guide, version 1.0**

© 2001 by Graphisoft. All rights reserved. Reproduction, paraphrasing or translation without express prior written permission of Graphisoft is strictly prohibited.

The World Square project used for illustrating the fly cover and Foreword is copyrighted by and used with the permission of Fender Katsilidis, Architects, Melbourne & Sydney, Australia.

Published by GRAPHISOFT R&D Rt., <http://www.graphisoft.com>

First edition.

Printed in Hungary.

### **Trademarks**

IFC and IAI are registered trademarks of the International Alliance for Interoperability. ArchiCAD is a registered trademark and PlotMaker, Virtual Building, StairMaker and GDL are trademarks of Graphisoft. Windows is a trademark of Microsoft Inc. Macintosh, Power Macintosh, QuickTime and TrueType are registered trademarks of Apple Computer Inc.

ISBN 963 00 7352 8

# Foreword

The images on the cover of this guide and below show an intriguing view of the complex ground structures for a new 81 storey residential tower in Sydney, Australia designed by architects **Fender Katsilidis**. Built over an existing site only partially constructed, the new building had to found itself on a new structural base.



A shared understanding, and in this case accuracy of existing site conditions, was vital for both architect and engineer to resolve the complex transition of structural members from lower levels to the new building main platform. This model demonstrates well the new business challenges facing AEC professionals and suggests opportunities & benefits for IFC collaboration.

New facilities are created in a business, regulatory, service systems and operational context that continues to become more complicated and demanding. Construction is increasingly complex, financial management and probity require more attention, yet there is less time to design and document. Construction contracts forge new methods for streamlining construction activities on site that applies further pressure. Clients need facility data to move in promptly and manage the new property with maximum return.

The object integrated model, ArchiCAD's innovative concept for over 18 years, puts in place a strategic foundation for a new paradigm of facility development based on the management of information. IFC based exchange now extends this concept to allow many more of the project development participants to create & share comprehensive project information. In contrast with the legacy 2D drawing approach based on a proprietary vendor format, the IFC technology eclipses both of these technology and market limitations by defining an innovative open standard for the description of AEC objects globally available to all in the construction industry

Companies that have products supporting the IFC protocol enable you to rethink the way you solve common business problems. The benefits can be any one of the following examples:

- Access object data for improved design and coordination in ArchiCAD – getting a building service engineer’s HVAC ductwork or water piping layout as 3D objects is one such example.
- Export your design for analysis by another specialist package – such as a thermal analysis enabling better understanding of your building performance, or advanced visualization, analysis of the logic of the design etc.
- Export a model for costing, estimation bidding or procurement – the integrated nature of the IFC information makes your database more valuable because your information is easy to access and analyze by such packages as costing applications.
- Your model is the definitive source of data for construction planning solutions such as form working applications which identify formed surfaces in the model, apply standard components for formwork assembly etc and then derive a list of parts and construction schedule.
- Extend further your service benefits to clients or users by easily exporting your facility data into their business systems.

Many more opportunities exist to leverage IFC object data for improved design quality, reduction of errors, better coordination and new services to owners and other partners in the development process. See the **IAI Implementer Support Group** website <http://www.iai.fhm.edv> for descriptions of currently available products.

ArchiCAD’s virtual building technology and IFC interface enables you to integrate and coordinate information from these sources; it extends the potential of the Virtual Building concept into real object collaboration with your project partners; and it will improve your level of service and competitive edge in the market place.

We welcome your feedback on several levels – how we have engineered the IFC interface, how you are able to apply it to your business and as this is a partner to partner application, how you make it work in your project team environments.

*Please contact your distributor with any problems or project feedback.*

---

# Introduction

## About this Guide

This IFC Reference Guide provides technical explanations of the operations of the IFC Add-Ons for ArchiCAD.

## IFC Add-On

IFC import/export Add-Ons are available for ArchiCAD releases 6.5 and 7.0. The IFC Add-Ons must be in the directory: [ArchiCAD folder]\Add-Ons\Import-Export.

**Note:** For Macintosh users the Add-On uses memory comparable to approximately 2-4 times the size of the IFC file.

## Version Support and Releases

ArchiCAD is certified as follows:



IFC™ R1.5.1  
IMPLEMENTATION  
Certified in accordance with official IAI  
facilitated approval procedures for  
IFC™ Release 1.5.1 - 31 May 2000



IFC™ R2.0  
IMPLEMENTATION  
Certified in accordance with official IAI  
facilitated approval procedures for  
IFC™ Release 2.0 - 23 May 2001

For latest update on IFC capabilities refer to the Graphisoft website <http://www.Graphisoft.com/support/IFC>.

Current availability is:

---

<b>IFC Version</b>	<b>Certifi- cation</b>	<b>Add-On</b>			
		<b>Windows</b>		<b>Macintosh</b>	
		ArchiCAD 6.5	ArchiCAD 7.0	ArchiCAD 6.5	ArchiCAD 7.0
<b>1.5.1</b>	21 May 2000	IFC_151.apx	IFC_151.apx	IFC_151 In/Out	IFC_151 In/Out
<b>2.0</b>	25 May 2001	IFC_200.apx	IFC_200.apx	IFC_200 In/Out	IFC_200 In/Out

---

Table 1: IFC Add-Ons

For system requirements see [http://www.graphisoft.com/products/architecture\\_and\\_design/graphisoft\\_archicad/technical\\_specifications](http://www.graphisoft.com/products/architecture_and_design/graphisoft_archicad/technical_specifications).

# The IFC Integrated Object Model

## What is an IFC?

IFC stands for “Industry Foundation Classes”, the set of internationally standardized object definitions for use in the Construction Industry developed by the International Alliance for Interoperability (IAI).

The business objective of the IAI, through its over 600 members in nine regional organisations around the world (refer <http://www.iai-international.org>) is:

To integrate the AEC/FM industry by specifying Industry Foundation Classes (IFC) as a universal language to improve the communication, productivity, delivery time, cost, and quality throughout the design, construction, operation and maintenance life cycle.

## Inter-operability Definition

The IFC concept is based on the idea of objects (or *elements* in ArchiCAD terms) brought together in an integrated model (ArchiCAD's *Virtual Building*). These objects are defined to support the whole lifecycle of facility development from inception through design, documentation and construction, then facility management and finally demolition and or disposal.

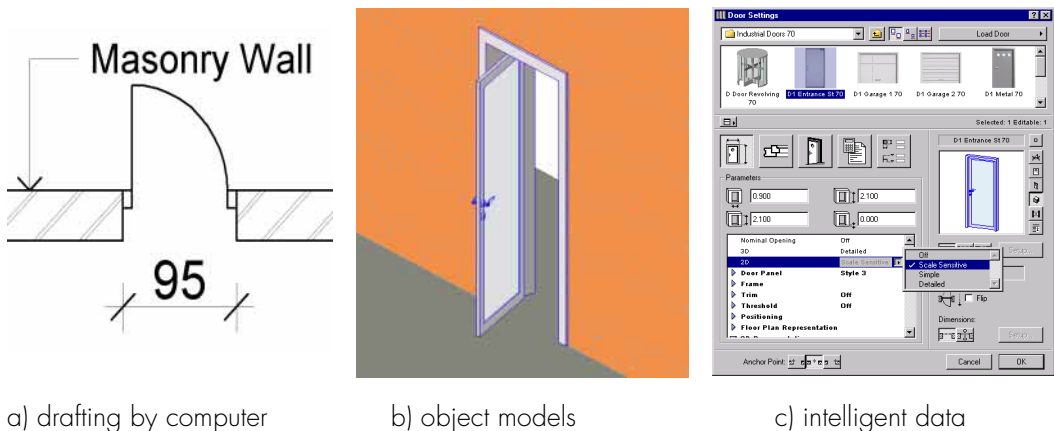


Figure 1: Evolution of 2D graphics to Object integrated models

IFC is to “Project Model” exchange (wall, door, window) what DXF is to graphic entity exchange (line, arc, circle).

IFC is available to all participants in the Construction Industry, for use globally, including use by all construction industry software vendors. IFC offers a higher-level “common language” for the sharing of intelligent objects between disciplines across the building lifecycle.



## Why Use IFC?

In each project, the CAD Manager, with the Project Leader and the party with whom the exchange is to be made, should review why information needs to be shared, what is to be exchanged and how.

Why would you choose IFCs instead of traditional techniques such as 2D files in DWG/DXF or DGN format?

The principal benefit of IFCs is their object description – not only does the IFC protocol preserve the full geometric description in 3D, but it also knows its location and relationships, as well as all the properties (or parameters) of each object such as finish, serial number, material description, thermal conductance, cost etc. For ArchiCAD users this offers great potential to access the accurate geometry of building service systems and structural elements, i.e. to improve the information to be coordinated in the model, rather than relying on the incomplete functions of current 2D CAD.

**Tip:** Note that Graphisoft's GDL object browser technology also allows information to be saved and transmitted in IFC format. See *Accessing GDL Product Data* below.

Always remember that a combination of techniques may be more useful than sticking to one solution only. Choose what suits the project resources, skills available and the specific task in hand.

## Planning an IFC Exchange

The first action is to determine that your partner and his software application has an IFC translator. Having validated this, the next important step is to select an appropriate version that both partners can support and understand the functionality offered by that version. ArchiCAD's support of the IFC protocol offers the following IFC compatibility:

IFC Version	ArchiCAD Release	Notes
1.51	6.5, 7.0	The first release of the IFC translator. This version supported the concept of building carcass.
2.0	6.5, 7.0	Substantial extension of functionality over 1.5.1. Increased scope for sharing: <ul style="list-style-type: none"><li>• Architectural design model for cost estimating,</li><li>• HVAC design for cost estimating,</li><li>• Architectural design for thermal load calculations and/or HVACsystem design,</li><li>• Client brief / space layout for architectural design.</li></ul> Additional concepts are provided for special transfers.
2x <sup>1</sup>		The IFC2x platform release will be stable until at least 2005 to allow the industry to deliver mature interface support. Additional capabilities on top of the IFC2x platform may be introduced, but the solutions remain downward compatible, so that previous IFC2x files can be still read by the enhanced IFC2x interfaces.

<sup>1</sup>No commercial release until 2002. Contact your distributor for details.

Table 2: IFC versions and ArchiCAD support

## Determining the Scope of a Transfer

Who are the parties to the transfer and what information should be transferred? Is it one way or two way? How are versions to be managed? While it would normally be preferable to use the latest IFC release, it may be that the one partner has a limited choice only. Once these preliminary decisions have been made, consideration needs to be given to specifying the detailed content of the exchange.

This task is really no different to planning a .DWG or .DGN transfer, except now the quality and richness of object information can radically improve the information to be shared and indeed lead to a rethink of the processes involved in design collaboration and estimation for example.

## Views

IFC cover a diverse range of information within building construction and the model does not distinguish who should be exchanging that information or at what point in a project the information is being exchanged.

Software applications are more usually concerned with specific requirements and should not have to implement or use every class that is contained within the IFC model. Therefore, subsets of the model are defined that, when isolated from the complete IFC model, still act as a coherent model. These subsets are called *views*.

Views are used to support the many different data needs of say architect, structural engineer, HVAC engineer or cost estimator for example.

IFC Version	View	ArchiCAD Functionality
1.5.1	CAD view	The following entities are supported: Beam, Building, Building Storey, Column, Door, Floor, Opening Element, Project, Proxy, Roof Slab, Site, Space, Space Boundary, Wall, Window.  This entity set is comprehensive enough to underpin a wide range of data sharing needs, based on the carcass of a building.
2.0	BLIS <sup>2</sup> view set	Four focused processes are supported: <ul style="list-style-type: none"> <li>• Architectural design &gt;&gt; Quantities take off / cost estimating</li> <li>• HVAC system design &gt;&gt; Quantities take off / cost estimating<sup>3</sup></li> <li>• Architectural design &gt;&gt; Thermal load calculations / HVAC system design</li> <li>• Client brief / space layout &gt;&gt; Architectural design</li> </ul> See the BLIS reference below for detailed description of the view definitions and objects included.
2x		Currently under development but will include all the views above <sup>2</sup> BLIS is a worldwide consortium of AEC application developers dedicated to developing IFC compliant products. See their website <a href="http://www.blis-project.org/">http://www.blis-project.org/</a> for more information. <sup>3</sup> This view is not supported by ArchiCAD

Table 3: IFC Views

In the case of the ArchiCAD IFC Add-Ons the default scope for export is to include everything.

**Note:** ArchiCAD does not support the IFC 2.0 BLIS view for HVAC system design to Quantities/cost estimating.

More selective functions are provided in the export settings of the Add-Ons to filter the export of data from ArchiCAD. See *Customizing the IFC data - Filter* below.

## Set up a Pilot or Test

Once the full scope has been agreed, test the process by doing a pilot. This activity ensures that both parties get what they expected, so that when crucial deadlines occur, the process can be confidently undertaken. The following guidelines may assist you and your team:

- Start with very simple examples.

- Validate at both ends.

- Progressively expand data set to the full level.

- Document procedures at both ends.

- Report to Project leaders in all disciplines so that they understand and agree to the process, resources and timetables.

**Note:** Remember that sharing information is both a technical task and a management process.

## Useful links for Using the IFC Exchange Protocol

To check current versions of **Graphisoft translators** refer to <http://www.graphisoft.com/ifc/>

To understand more about the **International Alliance for Interoperability (IAD)**, the worldwide chapters and the IFC development process see <http://iai-international.org>

To understand more about the **BLIS project** and definition of IFC 2.0 BLIS views see <http://www.blis-project.org/>

To join the Graphisoft **IFC User group** see <http://www.graphisoft.com/support/IFCuser>

To find out the current availability of **IFC compliant applications**, IFC development tools and forthcoming products see <http://www.aia.fhm.edu/ImplementationOverview.htm>

# Working with the IFC Translator

## IFC Exchange Processes

### IFC Objects versus 2D Drawings

In relation to the currently common DWG import/export there is a very important difference involved with IFC exchanges. In the DWG 2D case, exchange is predominantly by *layer* i.e. the information to be sent is represented by layer naming conventions which defines the class of usually 2D data being exchanged. This data is not integrated – in fact no building model exists; it is only by visual inspection that coordination of information becomes possible.

In the case of an IFC exchange *objects* are being transferred, and these objects are related to the building model both geometrically in 3D space and by relations such as the room on the 3<sup>rd</sup> Floor, part of a network etc. Objects actually have much more information defined within them – all the parameters defined in the IFC model specification and in ArchiCAD's case many other attributes that describe the object in increasing detail that makes them useful and easy to use.

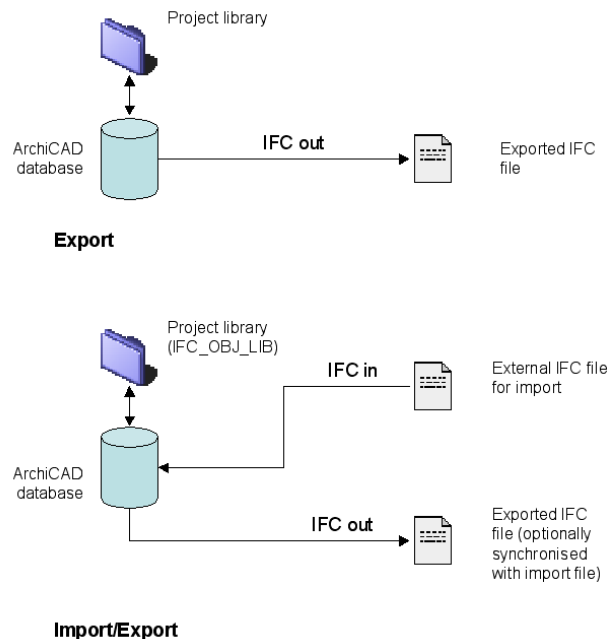


Figure 2: IFC Exchange Processes

Figure 2 shows an overview of the processes involved in importing and exporting an IFC file. In this new paradigm, layering is merely a way of ordering and presenting objects, not a proxy for defining information. When undertaking an IFC exchange we must consequently manage the transfer at the **Object level**. It is important that you understand the requirements for synchronization of objects between you & you partner's databases so that you can manage the versioning of data and audit this process.

## IFC Object Identifiers

The IFC model tracks objects involved in organization to organization exchange by an identifier, the Global Unique Identifier GUID. For every object created, a GUID is assigned that stays with the object for its life. If the object is deleted its GUID is "deleted" too.

## Initiating an IFC Export

In this first case, the user, say an Architect, has decided to issue an IFC file to one of his service engineering consultants. A first consideration is how the translator interprets door and window objects. ArchiCAD has a powerful parametric and intelligent element types for doors and windows; their behavior is connected to the wall elements they are placed in. The IFC model does not yet support the same level of functionality and as you might expect anyway, all CAD vendors have slightly different approaches to the implementation of these elements as well as other parts of the model. Accordingly, you must understand how the IFC definition relates to ArchiCAD, as these objects have unique considerations involved.

We recommend you use the special library objects for IFC doors and windows, for if you use another door or window object it will be transformed into a proxy object (a general purpose object that carries almost any information a sender wants to exchange) and importantly, will not be recognized when re-imported into ArchiCAD. In this case the Add-On assumes only rectangular openings.

The standard IFC door & window types are:

<b>ifcDoors</b>	<b>ifcWindows</b>
IFC_DOORDBLSWING,	IFC_WINDOWAWNING
IFC_DOORSGLSWING,	IFC_WINDOWCASEMENT
IFC_DOORREVOLVING,	IFC_WINDOWDOUBLEHUNG
IFC_DOORROLLUP,	IFC_WINDOWFIXED
IFC_DOORSLIDING	IFC_WINDOWPIVOTING
	IFC_WINDOWSLIDING

Table 4: ArchiCAD IFC Door & Window Library Objects (see *Appendix A & D* for more details)

These objects have the predefined parameters expected and supported in the IFC model and the GDL objects created by the Add-On have these (and other useful) parameters included. These objects are created normally when you import a new IFC file (but see below for extended options). The Add-On also has a facility to create these objects in your current library (see *Customizing the IFC data – Create IFC library* below). This latter method will ensure your normal project library for instance, includes these objects at project initiation, thus setting the database up for its later IFC export. See *Appendices B & C* for detailed descriptions of these objects. *Appendix D* sets out the Property Sets associated with each ifcObject.

So, having set up as above, the current ArchiCAD database is now saved in the appropriate IFC version format (see *Exporting & Importing IFC Files – Save As IFC file* below) and with any options defined (see *Customizing the IFC data – Options* below) an IFC file is created.

## Importing an IFC File

In this second case, you have been sent a file for the first time on your project. Open the file selecting the appropriate IFC version format, and select a location for the library of IFC objects that will be created (see above). You can either use the default name (IFC\_OBJ\_LIB) and location for the folder or, specify a unique name and location (i.e. Projects/P20642/Libraries/P20642\_IFC.lib).

**Tip:** It is recommended your IFC library be always located in your project specific directory, rather than allowing it to default to the current ArchiCAD library folder, where it may be merged with other project IFC library folders of the same name.

**Note:** Importing an IFC file into ArchiCAD *always* creates a new .pln file. See *Using IFCs on Projects* for techniques to manage IFC exchanges in a working environment.

## Exporting a Modified Imported IFC File

In this third case, you have edited the database created by the import and now want to send back a revised IFC file to your project partner.

A number of decisions must be made about the following:

- treatment of object identifiers (GUID)
- update of the IFC building elements

## Updating IFC Building Elements

If you make changes to the imported elements, for example you change the width of a wall, it will be reflected in the exported model – that is the same object has a different width. If you keep the original wall the GUID will remain the same; if however you delete the wall and replace it with an identical one, a new GUID is created with the new wall building element.

The synchronization of objects is achieved by “referring” to the IFC file that was responsible for the initiation of the database; i.e. in the case of an imported file, that file, in the case of an exported file the first or most recent file produced.

By choosing the relevant IFC data file, the Add-On is able to reference the object data in the two locations (ArchiCAD and IFC file), carries out a synchronization based on their GUIDs and creates a new export file as follows:

- if the GUID exists in both sources, update the data in the IFC file with the information from ArchiCAD
- if the GUID does not exist, create new objects for the IFC file
- if an object remains in the IFC reference file, but is not in the ArchiCAD data, show those objects as deleted

Several refinements of this process are possible. See *Exporting & Importing IFC Files – Save As IFC file* below for more information about managing IFC file synchronization.

## Archiving IFC Data

After exporting an IFC file it is recommended to save the .pln file to ensure that the future exports have the same GUIDs for the building elements. Using an archive (extension .pla) will gather all the IFC data and library objects consistently.

For IFC 2.00 translations there is also a special XML file created (see *Customizing the IFC Data – Edit selected item IFC properties* below) for storing property data. This should also be included in the archive.

# Setting Up for Import & Export

## Owner History

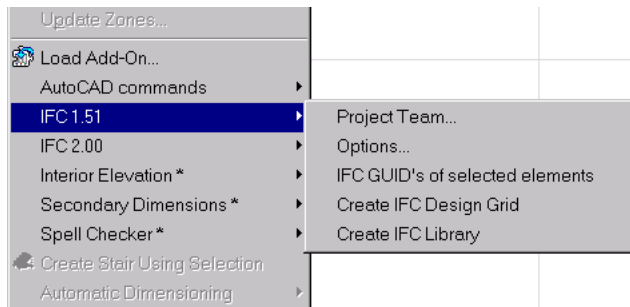
The IFC models support identifying the creator (or modifier) of objects in the database. This provision differs in each release.

## Defining the IFC 1.51 Project Team

Before exporting an IFC file you define your project team and their parameters such as address etc. This information is inserted in your exported file to identify you as the sender of the information, and optionally as the modifier of existing, or creator of new, objects in the file. The Project Team entity *only* exists in the IFC 1.51 model.

**Note:** The IFC 2.00 translator has the same dialog, but the data is not used in an exported file.

Choose the **Tools** menu, **IFC 1.51, Project Team...**, and fill in the project information dialog boxes.



## Project Team





The Project Team optionally describes the persons, organization or organization's staff involved in the IFC process.

The screenshot shows a 'Team member' dialog box with the following fields and values:

Type:	Person & Org.
Role:	Architect
	Graphisoft R&D Kft
Name:	Bill Brown
Organization:	Graphisoft R&D Kft
Description:	Virtual model builders Model maintenance Digital product data
Address 1:	Záhony utca 7
Address 2:	Budapest
PostCode:	1031
Country:	Hungary
Phone:	+36 (1) 437 3000
Email:	info@graphisoft.hu

Buttons: OK, Cancel

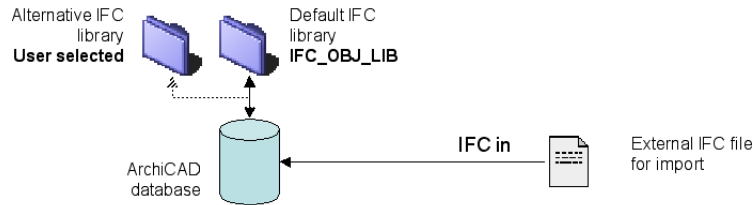
For each project, define all your project staff who are authorized to work on the project.

For each person participating in the project and IFC exchange fill out their details as appropriate.

# Exporting & Importing IFC Files

The key IFC operations are performed by the **File Open...** and **File Save as...** menu items.

## Open IFC File



### Import

Choose the **File** menu, **Open...**, **File format: IFC\_151** or **IFC\_200** and select the IFC file.

After selecting the IFC file the Add-On asks if you want to create a new library to store the IFC objects.



If you select **Yes**, you can set the new object library.



The Add-On checks for existing IFC libraries and will offer you options to merge or not.

The result is either an existing library is reused or a default or custom library created. If the used library doesn't contain IFC door and IFC window objects (see **Initiating an IFC Export**), the Add-On creates them.

## Save As IFC file

Choose the **File** menu, **Save As...**, **File format: IFC\_151** or **IFC\_200** and set the name of file.

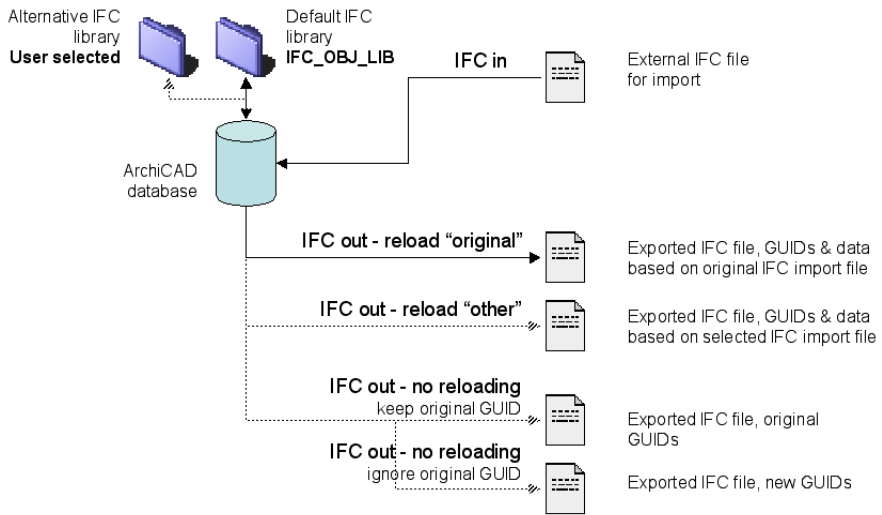
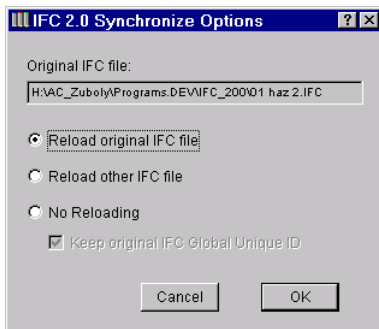


Figure 3: Synchronizing Options for IFC export files

If the ArchiCAD database (.pln file) originated from an IFC file the Add-On asks you to specify how you will treat the new file with respect to the originally imported IFC file.



If you select either of the Reload options the Add-On will update the changes with reference to the specific imported model file chosen and will export the combined result in the destination file. In this way the IFC file will also contain those items which are not understood by ArchiCAD such as extension property sets added by another application or some HVAC or FM related information.

### Original IFC File

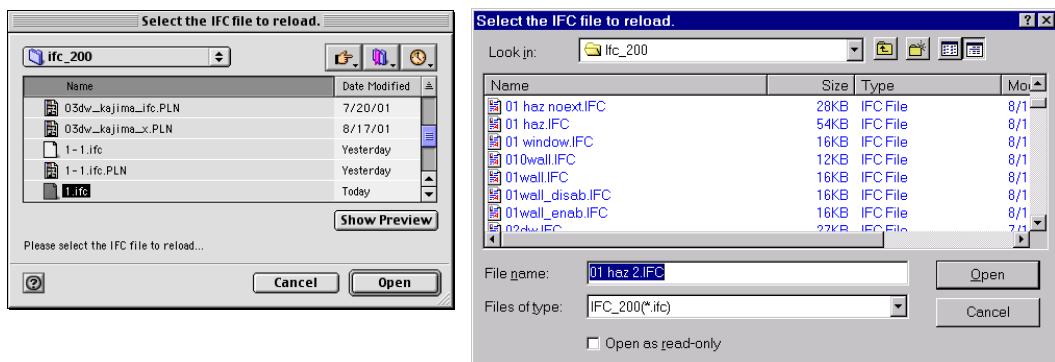
This text box displays the full path of the original IFC file. It cannot be edited in the box, but may be copied (if you have a long or complex path name) to specify a new file path if you change the synchronize option from other than using the original.

### Reload Original IFC File

Selecting this option selects the “original IFC file” as the reference IFC import file to be synchronized. It is the file that either created the database, was the initial source of the particular data set or the most recent IFC file exported. Unchanged objects keep all their original parameters (like Global Unique ID, Instance Number, ...). This file is the default name in the dialog.

### Reload other IFC File

If you select to reload another IFC file, the dialog prompts you select it, and uses its entities as the reference IFC export file for synchronization.



The result is similar as for “original”.

**Note:** The Add-On checks that this referenced file is a valid IFC file for this purpose.

### No Reloading

If you select not to synchronize your new export file, a further option is enabled that allows you to decide between one of two methods for the treatment of IFC GUIDs:

### **Keep the Original IFC Global Unique ID**

Enabling this option causes ArchiCAD elements to be exported with the original unique ID (which are stored in the ArchiCAD database). No reference is made for synchronization.

Disabling this option causes ArchiCAD to create a new set of GUIDs.

**Note:** This last option means that the export file has a different set of GUIDs, no longer connected to the original IFC model. This option supports a “template” function and allows new projects to be created based on an original.

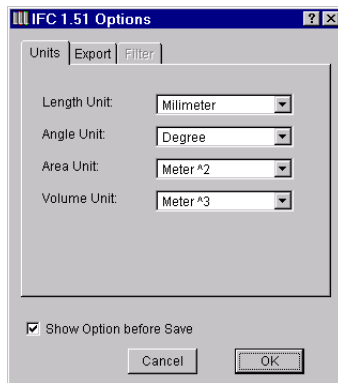
# Customizing the IFC Data

## Options

A number of options are provided to filter and customize the information included in the export file.

**Note:** The IFC 2.00 Add-On supports PSET data configuration. See *Using PSET Data* below for details for this option. In the examples below note the extra Tab **Property Set** for IFC 2.00.

## Units

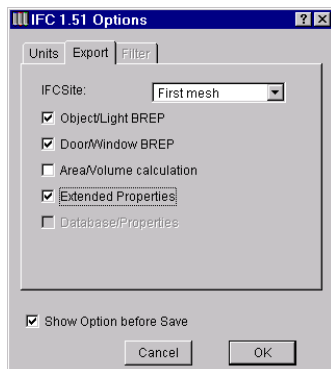


Specify the units for Length, Angle, Area and Volume.

Enabling the **Show Options before Save** ensures that the user confirms these parameters at the time of export.

**Note:** This option only applies if you are creating a new file.

## Export



Specify the way certain objects are processed. Enabling the **Show Options before Save** checkbox ensures that the user confirms these parameters at the time of export.

**Note:** This option only applies if you are creating a new file.

### **IFCSite**

Create site data by choosing one of three methods:

- **No geometry:** Exclude geometry data for the site entity.
- **First mesh:** Use the first mesh (if there are more than one) as the source of the site data.
- **Site object:** Use the object **IFC\_SITE.GSM** (must be in uppercase text) as the source of data for the site.

### **Object/Light BREP**

If this checkbox is enabled ArchiCAD's object geometry will be exported in BREP format (an IFC BREP is a simple form of boundary representation, in which all faces are planar and all edges are straight lines). In this case the exported file size will be significantly bigger than without this geometry.

Normally, it is better to disable this flag, because a smaller IFC file is created.

### **Door/Window BREP**

If this checkbox is enabled ArchiCAD's door and window geometry will be exported. In this case the exported file size will be significantly bigger than without the geometry.

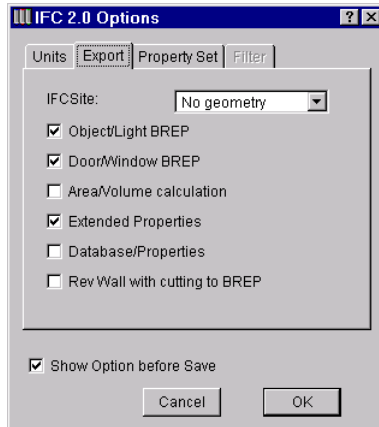
**Note:** If the IFC object is used (see *Working with the IFC Translator, Initiating an IFC Export*), geometry of doors and windows is exchanged using the IFC property sets (see *Appendix E*).

Normally, it is better to disable this flag, because in this case, the use of the default bounding box geometry results in a smaller IFC file.

### **Area/Volume Calculation**

If this checkbox is enabled the Add-On will calculate the elements' area and volume attributes. The area is usually the profile area of the extruded segment but in the case of a wall the area is the average area of the space boundary side of the wall (not the area of extruded profile).

Disable this flag if the calculation is not required.



## Extended Properties

It is recommended you export the extended properties because the Add-On saves the ArchiCAD attributes and parameters in the PSET data (for example the ArchiCAD object library name).

This setting is useful when you read the IFC file back into ArchiCAD as the Add-On will try to use the same library part as it is defined in the extended properties.

If the same library is loaded in export and import cases then the Add-On can use the original library element and doesn't read the (BREP) geometry. Otherwise, the Add-On creates a new library element from the BREP information (or 2DGeometry - IFC 2.00 or higher). The shape will be same but the color of the object will be different and monochrome.

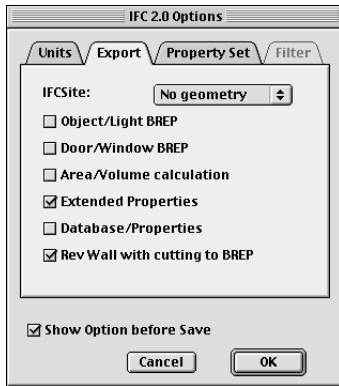
## Database Properties

The Add-On can export the ArchiCAD Database and Properties to the IFC file. However the Add-On cannot reconstruct (read back) the database and properties because the API interface doesn't allow it.

In most cases it should be disabled.



## Revolved Wall with Cutting to BREP

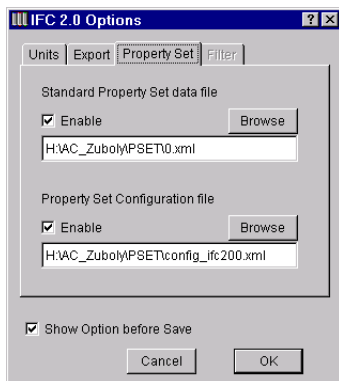


**Note:** This only applies to IFC 2.00 or higher.

If this checkbox is enabled the Revolved wall with cutting will be exported as a BREP instead of Advanced: Attribute Driven Clipped Revolved Solid geometry. Some CAD systems can interpret this geometry better.

## Property Set

The IFC 2.00 Add-On supports configuration of customizable PSET data. See *Using PSET Data* below for a description of the PSET configuration file format and mapping mechanism, and the PSET data operations when this option is enabled.



You can enable or disable the saving of the standard IFC property set in a user defined XML file. See *Appendix E: PSET Data* for details of 1.51 Door & Window Property Sets. Refer to IAI International web site <http://iai-international.org> for details of IFC 2.00 PSET definitions.

You can enable or disable the configured PSET processes and you can use a different configuration file for each plan file.

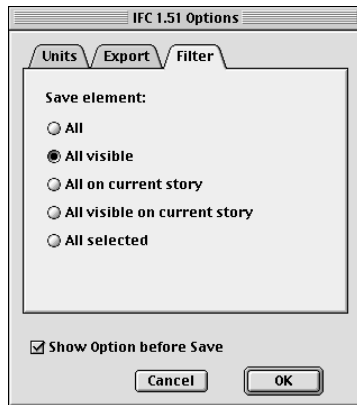
## Standard Property Set Data File

This file stores the standard common properties. The default location is a folder named XML\_IFC\_WP in your ArchiCAD application folder. If you have multiple projects this may be overwritten so we recommend locating the file in your working project directory, e.g. Projects/P20642/Libraries/XML\_IFC\_WP.

## Property Set Configuration File

This file stores the ArchiCAD and IFC property parameter mapping (see *Using PSET Data* below). This file should also be located in your working project directory, as described above.

## Filter



Filter elements in the current ArchiCAD database for export.

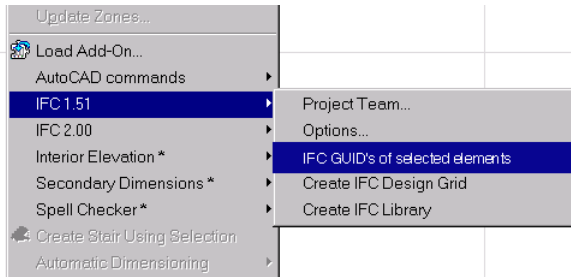
**Note:** This dialog is accessible only before the save and if the **Show Option before Save** is set from the **Tools, IFC 1.51 or 2.00, Options...** dialog box.

The filter option only applies to the creation of “new” IFC files in two cases:

- from a database created in ArchiCAD for its first IFC export, or
- from a working database - where there have been previous IFC exports or the database was created from an IFC import – and where the Synchronise option is **No Reloading**.

This filtering function ensures that the IFC data file is a consistent body of data. Refer to *Exporting & Importing IFC Files – Save As IFC File* for the synchronization options for the **No Reloading** option. If the **Keep original GUIDs** is enabled exported objects will have IDs based on the database and will be related; if disabled, new GUIDs will be created to establish a new database having now no relation to the original database.

## IFC GUID's of Selected Elements



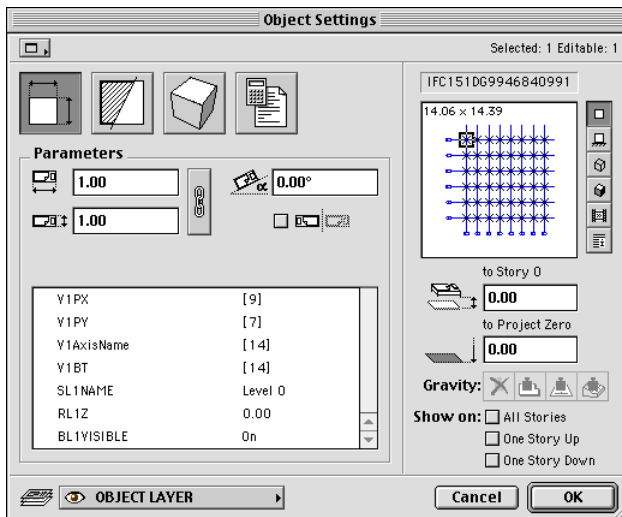
Choosing this menu will list the selected elements IFC Global Unique ID (GUID)s and ArchiCAD Unique ID (AC\_UID) in the ArchiCAD Report window.

## Create IFC Design Grid

To exchange an IFC Design Grid (the Japanese term for a design grid is Torishin) requires creating one with the Tools menu. Follow this sequence:

1. Draw lines and arcs of the design grid. Use a different color to draw the crossing lines. The tool can use the line color information to differentiate between the axes.
2. Select them (with marquee if you need a multilevel design grid).
3. Select the **Tools** menu, **IFC\_1.51** or **IFC\_2.00**, **Create IFC Design Grid** to convert the lines to a design grid object named as IFC151DG<ID\_NUMBER>.GSM.

You can change the object by adjusting its parameters:



## Variable name Description

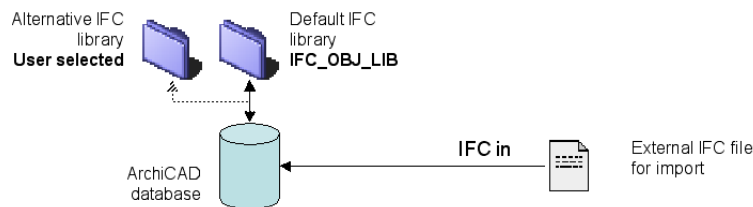
<b>VIPX</b>	The distance of cross points in X direction.
<b>VIPY</b>	The distance of cross points in Y direction.
<b>VIAxisName</b>	The name of the axis.
<b>VIBT</b>	The position of the axis name (toggle left, right or top, bottom).
<b>SL1Name</b>	The name of grid level (not visible in ArchiCAD, appears only in IFC file).
<b>RL1Z</b>	The height of the grid level.
<b>BL1Visible</b>	The visibility flag of the grid level.

After saving to an IFC file the object will be converted to IfcDesignGrid.

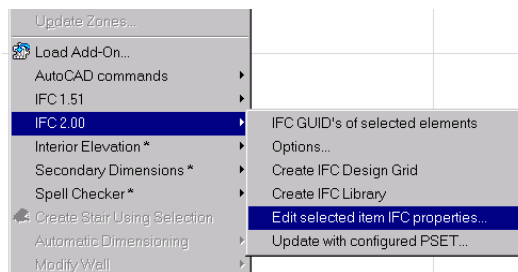
## Create IFC Library

This menu item allows a user to create the IFC compatible standard library objects for doors and windows. If chosen it places them in the current active library.

**Note:** See *IFC Exchange processes – Importing an IFC File* for discussion of IFC library selection and creation.



## Edit Selected Item IFC Properties



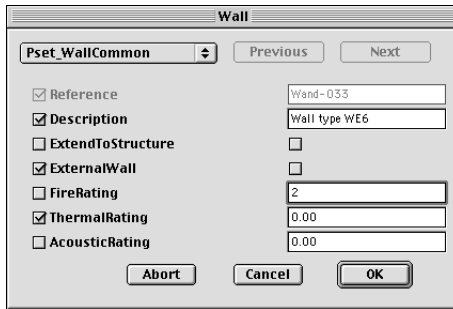
**Note:** This only applies to IFC version 2.00 or higher.

This menu item allows you to review and edit the element's IFC properties. The IFC properties are saved in an XML file which can be found by default in the XML\_IFC\_WP subdirectory of the ArchiCAD application folder.

**Note:** See *Customizing the IFC Data – Property Set* for the user defined options for locating XML property set data.

The name of the XML file is same as the ArchiCAD project except for the extension (.pln -> .xml).

If you select this menu item, the standard Property Set of the currently selected elements are presented in a dialog.



The first column shows the enable/disable status of the appropriate simple properties. If this option is selected the property data will be exported.

**Note:** See *Appendix B: Exported Data* and *Appendix C: Imported Data* for detailed specifications of the list of element Property Sets (PSET) and their property inclusions.

The second column is the name of the simple property and the third is the value. If the current element has more Property Sets then you can select other Property Sets from the list box placed in the upper left corner.

There are some protected Simple Properties which are set according to their ArchiCAD elements. For example: *Reference* is the ID of the element, *Size* is defined by the size of the element.

### OK

After changing the parameters selecting OK writes the changes to the XML file.

### Cancel

Performs no update for the current element and moves to the next selected element type in the dialog.

### Abort

The Add-On stops writing any changes in the XML file and returns to ArchiCAD.

# Configuring PSET Data

## IFC Property Sets (PSET)

The IFC Object Model comprises a set of well defined ways of breaking down information into classes and the structure of information that defines an object. The information structures provide a formal specification of attributes that belong to classes.

However, there are many types of information that users might want to exchange that are not currently included within the IFC Model.

For this purpose the IFC Model provides the Property Definition mechanism (part of which is within the *IfcKernel* schema with the remainder being within the *IfcPropertyResource* schema). Property Definition is a generic mechanism that allows model users and developers to define, connect and use data-driven, expandable properties with objects.

The Add-on allows users to define and exchange custom sets of properties for selected ArchiCAD elements.

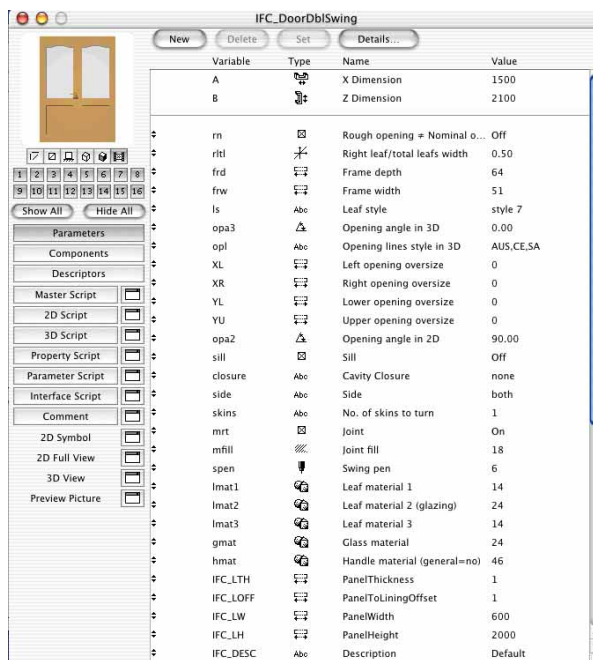


Figure 6: An ArchiCAD element, e.g. type "Door", with many custom properties can be exchanged in an IFC file using the PSET entity.

**Note:** Refer to the *IFC2x Model Implementation Guide* for a detailed description of IFC Properties and related concepts.

## IFC 2.0 PSET Configuration

The IFC 2.0 add-on can read a PSET configuration file, where the user can define the mapping between the ifc Properties and ArchiCAD door, window, object, lamp and zone parameters.

The configuration file is an XML format file, named: "config\_ifc200.xml". The Add-On locates this file in <ArchiCAD folder>\XML\_IFC\_WP folder. The PSET configuration file format is:

```

    <?xml version="1.0"?>
<Ifc_Pset_Config>
  <Groups>
    < Group name="Door_Jpn">
      <LibName type="Door" name="door 01"/>
      <LibName type="Door" name="door 02"/>
      ...
    </ Group>
    ...
  </ Groups >
  <Link>
    <IfcPropertySet Name="Pset_Occurence_Ctrl">
      <IfcSimpleProperty Name="unique_id" Type="IfcString"
        ACParamName="@_UID" ACParamType="Integer"/>
    </IfcPropertySet>
  </Link>
  <Mapping>
    <Command name="Load">
      < Group name="Door_Jpn">
        <IfcPropertySet Name="Pset_Door_GS">
          <IfcSimpleProperty Name="hand" Type="IfcString"
            ACParamName="hand" ACParamType="String" />
          ...
        </IfcPropertySet>
      </ Group>
    </Command>
  </Mapping>
</Ifc_Pset_Config>
  <?xml version="1.0"?>

```

**Note:** Refer to <http://www.wc3.org> for details of XML standards.

The configuration file contains one <Ifc\_Pset\_Config> structure which may contain a <Groups>, a <Link> and a <Mapping> structure. The configuration file may define different PSET mapping for three processes:

- **Save** (export) - the Add-On saves ArchiCAD library part parameter data in user defined property sets in the new IFC file.
- **Load (Open** or import) - the Add-On adds to ArchiCAD library part parameters, property set data in the selected IFC file.
- **Update** - the Add-On updates library part parameters in an existing ArchiCAD plan file using the selected IFC file.

## <Groups>

<Groups> contains a list of <Group> definitions. A <Group> is a named set of ArchiCAD library parts sharing a common mapping to a named set of element parameters. See <Mapping> below for details of the mappings.

```
<?xml version="1.0"?>
  <Ifc_Pset_Config>
    <Groups>
      <Group name="Door_Jpn">
        <LibName type="Door" name="door 01"/>
        <LibName type="Door" name="door 02"/>
        ...
        <LibName type="Door" name="door 16"/>
      </Group>
      <Group name="Shutter_Jpn">
        <LibName type="Door" name="shutter 01"/>
        ...
      </Group >
      <Group name="Window_Jpn">
        <LibName type="Window" name="combination_1"/>
        <LibName type="Window" name="combination_2"/>
        ...
        <LibName type="window" name="multiwindow 1x4"/>
      </Group >
      < Group name="multiwindow 3x4">
        <LibName type="window" name="multiwindow 3x4"/>
      </Group >
      ...
    </Groups>
    ...
  </Ifc_Pset_Config>
</?xml version="1.0"?>
```

For example (referring to the example above):

“Door\_Jpn” contains the doors (door 01, door 02, ... door 16)

“Shutter\_Jpn” contains the shutters (shutter 01, ...)



“Window\_Jpn” contains all the windows (combination\_1, ..., multiwindow 3x4...), however

“multiwindow 3x4” contains only the multiwindow 3x4 library part.

An ArchiCAD element may be a part of more than one group. For example, the multiwindow 3x4 library part is a member of both “Window\_Jpn” and “multiwindow 3x4”.

The <Group> definition is:

```
<Group name="Shutter_Jpn">
```

where *name* is the user given name of the group.

The library part definition is:

```
<LibName type="Door" name="shutter 01"/>
```

where *type* is the ArchiCAD element type (door, window, object, lamp, zone) and *name* is the library part name.

**Note:** *name* in both structures is case sensitive!

### <Link>

```
<Ifc_Pset_Config>
...
<Link>
  <IfcPropertySet Name="Pset_Occurrence_Ctrl">
    <IfcSimpleProperty Name="unique_id" Type="IfcString"
      ACPParamName="@_UID" ACPParamType="Integer"/>
  </IfcPropertySet>
</Link>
...
</Ifc_Pset_Config>
```

#### **Not currently used.**

In a future development this feature can define the link information between the IFC and ArchiCAD elements. The link is the ArchiCAD unique ID which is saved in the PSET “Pset\_Occurrence\_Ctrl”,

the *name* of the IFC SimpleProperty is “unique\_id”,

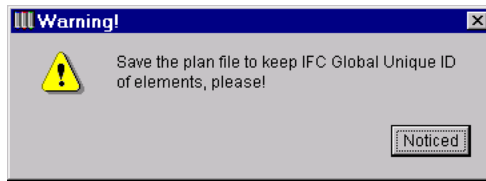
the IFC data *type* is “IfcString”,

the ArchiCAD parameter name *ACPParamName* is “@\_UID”, and

the ArchiCAD parameter type *ACPParamType* is “Integer”.

The “@\_UID” string references the ArchiCAD uid but this information is not used, because the Add-On saves the IFC Global unique ID in the appropriate ArchiCAD element’s own data section.

The link works automatically if the plan file is saved following the (exporting) saving of the IFC file.



A Warning reminds the user to save the ArchiCAD database.

## <Mapping>

The <Mapping> structure defines the mapping of the named library part groups to IFC property sets. Mappings are configured for any of the three functions of **Save**, **Load**, and **Update**.

### <Mapping>

```

<Command name="Save">
  <Group name="Door_Jpn">
    <IfcPropertySet Name="Pset_Door_GS">
      <IfcSimpleProperty Name="hand" Type="IfcString"
        ACPParamName="hand" ACPParamType="String" />
      <IfcSimpleProperty Name="opening_angle_3D" Type="IfcReal"
        ACPParamName="opening_angle_3D"
        ACPParamType="real" Conversion="Length"
        Optional="true"/>
      ...
    </IfcPropertySet>
  </Group>
  ...
</Command>

```

### </Mapping>

The <Command name=addon\_function> structure defines for a particular function the mapping conversions that will be processed. The value of *addon\_function* is "Save", "Load" or "Update".

The <Group name=library\_part\_group\_name> defines the list of property set definitions to be mapped for the named library part group. The *library\_part\_group\_name* must match the Group name defined in the <Groups> structure.

The structure <IfcPropertySet Name=Pset\_name> defines the name of the Ifc PropertySet and contains the list of mapped Ifc SimpleProperties.

The structure <IfcSimpleProperty Name=propertyName Type=ifcDataType ACPParamName=parameterName ACPParamType=parameterType /> defines the IFC property to ArchiCAD parameter mapping.

*Name* is the name of the SimpleProperty.

*Type* is the IFC data type: “IfcBoolean”, “IfcInteger”, “IfcReal” or “IfcString”)

*ACParamName* is the ArchiCAD parameter name.

**Note:** The *library\_part\_group\_name*, *Pset\_name*, *Name* and *ACParamName* are case sensitive!

There are reserved strings: “@\_A”, “@\_B” refers to parameter A, parameter B (see *IFC Reference Guide Customizing the IFC Data – Edit selected item IFC properties*, and *Appendix E: PSET Data* for the method to edit PSET data and the default IFC Property Sets.

If the parameter is an array the format is:

parameterName(index1) [(index2)]

e.g.: “turn\_width(2)(3)”

*ACParamType* is the data type of the ArchiCAD parameter:

“bool”, “integer”, “real” or “string”)

The above tags are mandatory.

## Optional Tags

The optional tags “Conversion” and “Optional” are used to carry out units translation and customize error reporting.

### Converting parameters’ units of measure

*Conversion*=unit\_type

where *Conversion* may be one of “Length”, “Angle”, “Area” or “Volume”. (See the *IFC Reference Guide: Customizing the IFC data- Options...*)

For example:

```
<IfcSimpleProperty Name="opening_width" Type="IfcReal" ACParamName="f_w"  
ACParamType="real" Conversion="Length" />
```

The Ifc parameters are converted according to the **Tools, IFC 2.00, Options, Units** settings.

### Checking for parameters during processing

The *Optional* tag allows two modes of error reporting when processing property set data.

*Optional*=trueOrFalse

where *Optional* may be either “true” or “false”.

```
<IfcSimpleProperty Name="horizontal_division" Type="IfcInteger"  
ACParamName="horizontal_division"  
ACParamType="Integer" Optional="True" />
```

When *Optional*="true", if the selected parameter is present in the ArchiCAD element or in the IFC Pset its information is processed, otherwise if the parameter is absent then no error message is provided.

When *Optional*="false", or the tag is not defined in the mapping – the default case, a missing parameter causes an error message to be reported (see *Reporting* below).

The <*Command name*=addon\_name> structure applies to all three processing options.

## Using PSET Data

The PSET configurations described above modify the actions of the two standard IFC functions of **Save as...** (exporting) and **Open...** (importing) an IFC file. An additional function **Update** provides for editing existing library part parameters in a working ArchiCAD database.

### IFC 2.00 Options – Property Sets

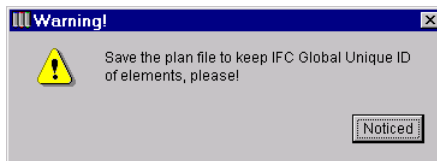
See *Customizing the IFC Data – Options* to configure the settings for XML PSET mapping and data files.

### Save as IFC File

See *IFC Reference Guide Exporting & Importing IFC Files – Save As IFC file* for the method to export the IFC file.

As the IFC file is created, the Add-On checks for PSET function definitions in the XML configuration file. If both are present, (the Config file and a mapping definition for the “Save” function), it adds the library part parameter data to the IFC file according to the mapping specification.

At the conclusion of the command, the user is reminded to save the ArchiCAD database to store the IFC GUID data created on export.

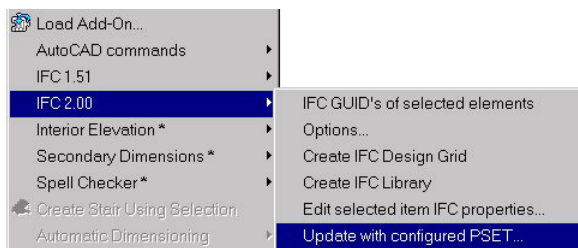


### Load PSET Data

See *IFC Reference Guide Exporting & Importing IFC Files – Open IFC file*: for the method to open (import) an IFC file.

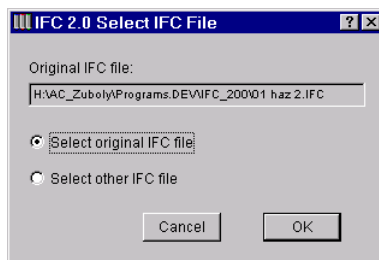
As the IFC file is read, the Add-On checks for PSET function definitions in the XML configuration file. If both are present, (the Config file and a mapping definition for the “Load” function), it adds the PSET data to the new ArchiCAD library parts according to the mapping specification.

## Update with Configured PSET

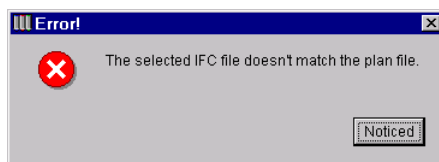


When the **Tools, IFC 2.00, Update with configured PSET...** is selected, the Add-On checks for PSET function definitions in the XML configuration file. If both are present, (the Config file and a mapping definition for the “Update” function), it adds the PSET data to the existing ArchiCAD library parts according to the mapping specification.

**Note:** This updates the configured parameters only.



If the selected IFC file does not match the GUID data in the ArchiCAD database, an error is advised and the user prompted to select the correct IFC file.



## Reporting

The IFC Add-On creates reports for the three PSET processes:

For example: **Update**

```
-Update ArchiCAD Element parameters-
typeID: 4 uid: 60 Ifc instNum: #8413 Ifc GUID: frG~XdzBH<0RviJs0$MU
Error: 'Property not found'! Libpart name: 'shutter 04' Parameter name:
'p_10' type: 'real' PropertySet name: 'Pset_ShutterCommon_Jpn' Property
name: 'slat_panel_width' type: 'IfcReal'.
Error: 'Undefined Parameter'! Libpart name: 'window 21' Parameter name:
'jw_left2' type: 'real' PropertySet name: 'Pset_WindowCommon_Jpn'
Property name: 'jamb_width_side' type: 'IfcReal'.
Error: 'Parameter type mismatch'! Libpart name: 'multi-corner-window
3x3' Parameter name: 'turn_width' type: 'real' PropertySet name:
'Pset_MultiWindowCommon_Jpn' Property name: 'turn_width' type:
'IfcReal'.
Error: 'Parameter dimension failure'! Libpart name: 'window 21'
Parameter name: 'jw_left2' type: 'real' PropertySet name:
'Pset_WindowCommon_Jpn' Property name: 'jamb_width_side' type:
'IfcReal'.
Error: 'String length failure'! Libpart name: 'window 21' Parameter
name: 'jw_left2' type: 'real' PropertySet name: 'Pset_WindowCommon_Jpn'
Property name: 'jamb_width_side' type: 'IfcReal'.
Error: 'Write property failure'! Libpart name: 'window 21' Parameter
name: 'jw_left2' type: 'real' PropertySet name: 'Pset_WindowCommon_Jpn'
Property name: 'jamb_width_side' type: 'IfcReal'.
```

### Error Messages

#### Property not found

The specified property was not found in the IFC PSET definition.

#### Undefined Parameter

The specified parameter was not found in the ArchiCAD library part.

#### Parameter type mismatch

String to real conversion was unsuccessful or the parameter was an array and no index was defined (e.g. "turn\_width(2)(3)") ??

#### Parameter dimension failure

The index was higher than the limit.

#### String length failure

String length was longer than 255 byte.

#### Write property failure

The Add-On cannot write the property information (program failure).

## Example

```

<?xml version="1.0"?>
<Ifc_Pset_Config>
  <Groups>
    <Group name="Door_Jpn">
      <LibName type="Door" name="door 01"/>
      <LibName type="Door" name="door 02"/>
    </Group>
    <Group name="Window_Jpn">
      <LibName type="window" name="multiwindow 3x4"/>
      <LibName type="window" name="window 01"/>
      <LibName type="window" name="window 02"/>
    </Group>
    <Group name="multiwindow 3x4">
      <LibName type="window" name="multiwindow 3x4"/>
    </Group>
  </Groups>
  <Link>
    <IfcPropertySet Name="Pset_Occurence_Ctrl">
      <IfcSimpleProperty Name="unique_id" Type="IfcString" ACPParamName="@_UID"
        ACPParamType="Integer"/>
    </IfcPropertySet>
  </Link>
  <Mapping>
    <Command name="Save">
      <Group name="Door_Jpn">
        <IfcPropertySet Name="Pset_Door_GS">
          <IfcSimpleProperty Name="hand" Type="IfcString"
            ACPParamName="hand" ACPParamType="String" />
          <IfcSimpleProperty Name="opening_angle_3D"
            Type="IfcReal" ACPParamName="opening_angle_3D"
            ACPParamType="real" Conversion="Angle" Optional="True" />
          <IfcSimpleProperty Name="glass_hatch" Type="IfcBoolean"
            ACPParamName="glass_hatch" ACPParamType="bool" />
          <IfcSimpleProperty Name="matframe" Type="IfcInteger"
            ACPParamName="matframe" ACPParamType="Integer" />
          <IfcSimpleProperty Name="keylen_a" Type="IfcReal"
            ACPParamName="keylen_a" ACPParamType="Real" />
          <IfcSimpleProperty Name="tfont" Type="IfcString"
            ACPParamName="tfont" ACPParamType="String"
            Optional="True" />
        </IfcPropertySet>
      </Group>
    </Command>
  </Mapping>
  <Link>
    <IfcPropertySet Name="Pset_DoorCommon_Jpn">
      <IfcSimpleProperty Name="type_name" Type="IfcString"
        ACPParamName="d_type_name" ACPParamType="String" />
    </IfcPropertySet>
  </Link>
</Ifc_Pset_Config>

```



```

        <IfcSimpleProperty Name="type_no" Type="IfcString"
          ACParamName="type_no" ACParamType="String" />
      </IfcPropertySet>
    </Group>
    <Group name="Window_Jpn">
      <IfcPropertySet Name="Pset_Window_GS">
        <IfcSimpleProperty Name="kline" Type="IfcBoolean"
          ACParamName="kline" ACParamType="Bool" />
        <IfcSimpleProperty Name="keylen_a" Type="IfcReal"
          ACParamName="keylen_a" ACParamType="Real" />
        <IfcSimpleProperty Name="tfont" Type="IfcString"
          ACParamName="tfont" ACParamType="String" />
      </IfcPropertySet>
    </Group>
    <Group name="Window_Jpn">
      <IfcPropertySet Name="Pset_WindowCommon_Jpn">
        <IfcSimpleProperty Name="type_name" Type="IfcString"
          ACParamName="w_type_name" ACParamType="String" />
        <IfcSimpleProperty Name="type_no" Type="IfcString"
          ACParamName="type_no" ACParamType="String" />
      </IfcPropertySet>
    </Group>
    <Group name="multiwindow 3x4">
      <IfcPropertySet Name="Pset_MultiWindowCommon_Jpn">
        <IfcSimpleProperty Name="thickness_mume_1"
          Type="IfcReal" ACParamName="th_mume(1)"
          ACParamType="Real" Conversion="Length" />
        <IfcSimpleProperty Name="thickness_mume_2"
          Type="IfcReal" ACParamName="th_mume(2)"
          ACParamType="Real" Conversion="Length" />
        <IfcSimpleProperty Name="turn_width_1" Type="IfcReal"
          ACParamName="turn_width(1)(1)" ACParamType="Real"
          Conversion="Length" />
        <IfcSimpleProperty Name="open_width_12" Type="IfcReal"
          ACParamName="open_width(3)(4)" ACParamType="Real"
          Conversion="Length" />
      </IfcPropertySet>
    </Group>
  </Command>
  <Command name="Update">
    <Group name="Door_Jpn">
      <IfcPropertySet Name="Pset_Door_GS">
        <IfcSimpleProperty Name="glass_hatch" Type="IfcBoolean"
          ACParamName="glass_hatch" ACParamType="bool" />
        <IfcSimpleProperty Name="matframe" Type="IfcInteger"
          ACParamName="matframe" ACParamType="Integer" />
        <IfcSimpleProperty Name="keylen_a" Type="IfcReal"
          ACParamName="keylen_a" ACParamType="Real" />
      </IfcPropertySet>
    </Group>
  </Command>

```

```

        <IfcSimpleProperty Name="tfont" Type="IfcString"
        ACPParamName="tfont" ACPParamType="String"
        Optional="True" />
    </IfcPropertySet>
</Group>
<Group name="Door_Jpn">
    <IfcPropertySet Name="Pset_DoorCommon_Jpn">
        <IfcSimpleProperty Name="type_name" Type="IfcString"
        ACPParamName="d_type_name" ACPParamType="String" />
        <IfcSimpleProperty Name="type_no" Type="IfcString"
        ACPParamName="type_no" ACPParamType="String" />
    </IfcPropertySet>
</Group>
<Group name="Window_Jpn">
    <IfcPropertySet Name="Pset_Window_GS">
        <IfcSimpleProperty Name="kline" Type="IfcBoolean"
        ACPParamName="kline" ACPParamType="Bool" />
    </IfcPropertySet>
</Group>
<Group name="Window_Jpn">
    <IfcPropertySet Name="Pset_WindowCommon_Jpn">
        <IfcSimpleProperty Name="type_name" Type="IfcString"
        ACPParamName="w_type_name" ACPParamType="String" />
        <IfcSimpleProperty Name="type_no" Type="IfcString"
        ACPParamName="type_no" ACPParamType="String" />
    </IfcPropertySet>
</Group>
<Group name="multiwindow 3x4">
    <IfcPropertySet Name="Pset_MultiWindowCommon_Jpn">
        <IfcSimpleProperty Name="thickness_mume_1"
        Type="IfcReal" ACPParamName="th_mume(1)"
        ACPParamType="Real" Conversion="Length" />
        <IfcSimpleProperty Name="thickness_mume_2"
        Type="IfcReal" ACPParamName="th_mume(2)"
        ACPParamType="Real" Conversion="Length" />
        <IfcSimpleProperty Name="turn_width_1" Type="IfcReal"
        ACPParamName="turn_width(1)(1)" ACPParamType="Real"
        Conversion="Length" />
        <IfcSimpleProperty Name="open_width_12" Type="IfcReal"
        ACPParamName="open_width(3)(4)" ACPParamType="Real"
        Conversion="Length" />
    </IfcPropertySet>
</Group>
</Command>
<Command name="Load">
    <Group name="Door_Jpn">
        <IfcPropertySet Name="Pset_Door_GS">

```

```
<IfcSimpleProperty Name="hand" Type="IfcString"
ACParamName="hand" ACParamType="String" />
<IfcSimpleProperty Name="opening_angle_3D"
Type="IfcReal" ACParamName="opening_angle_3D"
ACParamType="real" Conversion="Angle" Optional="True" />
<IfcSimpleProperty Name="glass_hatch" Type="IfcBoolean"
ACParamName="glass_hatch" ACParamType="bool" />
<IfcSimpleProperty Name="matframe" Type="IfcInteger"
ACParamName="matframe" ACParamType="Integer" />
<IfcSimpleProperty Name="keylen_a" Type="IfcReal"
ACParamName="keylen_a" ACParamType="Real" />
<IfcSimpleProperty Name="tfont" Type="IfcString"
ACParamName="tfont" ACParamType="String"
Optional="True" />
</IfcPropertySet>
</Group>
<Group name="Window_Jpn">
<IfcPropertySet Name="Pset_Window_GS">
<IfcSimpleProperty Name="kline" Type="IfcBoolean"
ACParamName="kline" ACParamType="Bool" />
<IfcSimpleProperty Name="keylen_a" Type="IfcReal"
ACParamName="keylen_a" ACParamType="Real" />
<IfcSimpleProperty Name="tfont" Type="IfcString"
ACParamName="tfont" ACParamType="String" />
</IfcPropertySet>
</Group>
</Command>
</Mapping>
</Ifc_Pset_Config>
```

# Accessing GDL Product Data

The model of the facility can be thought of in two ways, the first as the whole facility in a single database, and the other as a collection of individual objects, in ArchiCAD called library parts. These parts are created using GDL - Geometric Description Language (See *Graphisoft GDL Reference Manual*).

With emerging internet based construction product sources supporting GDL format, the GDL Plugin allows you to access these sites using your web browser and incorporate the product data in your projects.

**Note:** Your project partners using AutoCAD can exploit GDL data by using the **GDL Adapter**. Refer to the GDL Technology web site <http://www.gdlTechnology.com> for details.

Here we will describe how you can save GDL web objects in IFC format.

## Install the GDL Object Plugin

Refer to GDL technology website for details of how to download and install the plugin (<http://www.gdlTechnology.com>).

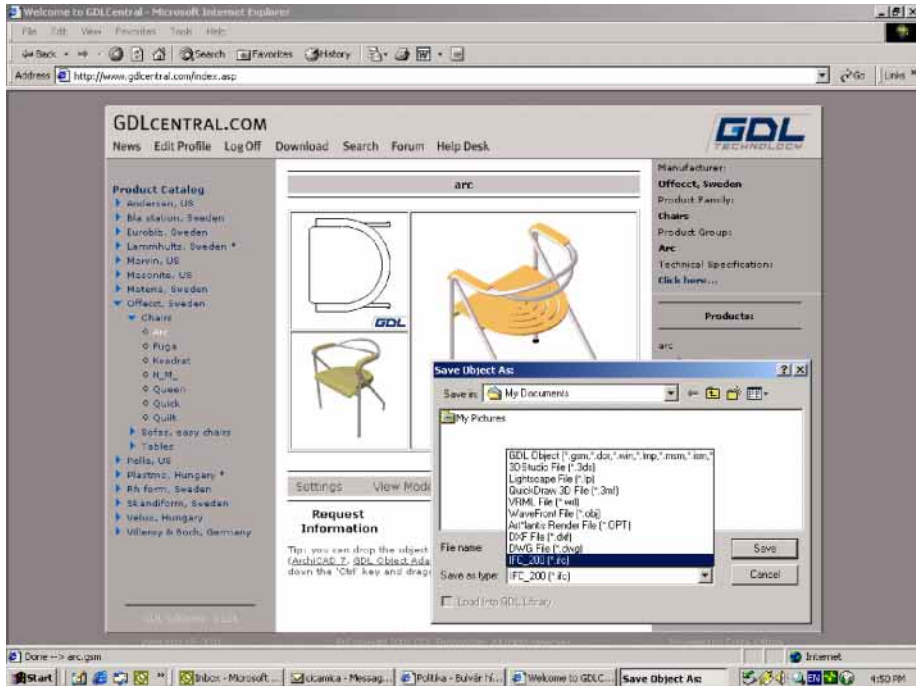
The screenshot shows a web browser window displaying the GDL Technology website. The browser's address bar shows <http://www.gdltechnology.com/>. The website has a blue header with the GDL Technology logo. The main content area contains a registration form with the following fields and options:

- Title:** Please fill out this form to download the latest version of the GDL Plug-in.
- Required fields:** E-mail, First Name, Last Name, Company, Country, Language Version, and CAD Connectivity.
- Form Fields:**
  - E-mail:
  - First Name:
  - Last Name:
  - Company:
  - Country:
  - Language Version:
  - CAD Connectivity:
- Tip:** click on the link above, to get more info on CAD Connectivity from Help Desk.
- Newsletter:**  I would like to receive GDL Technology's newsletter via e-mail.
- Buttons:** Download Plug-in, Reset Fields

**Note:** Set the CAD Connectivity option to “Yes” so that the version you download supports saving of Objects in various file formats (which includes the IFC format).

## Save as IFC

We assume you go to GDL Central (<http://GDLcentral.com>), register with their site and choose from a Product Catalog.



**Note:** Not all web sites will permit you to save objects. Check the site’s commercial and operational details.

Identify and display the product required. Click on the **Save** button or select (right click on WIN, control click on MAC) the object in the Plugin 3D window, choose the **Save Object...** option, select from **IFC 1.51** or **IFC 2.00** formats and specify a location. The IFC data file will be created at your specified location.

---

## Downloading GDL Objects

An alternative method is to insert GDL objects into your working ArchiCAD database using the GDL Plugin to download them. Two methods are possible:

### Save as Object

Click on the **Save** button or select (right click on WIN, control click on MAC) the object in the Plugin 3D window, to save the object to your work place. Choose **Object** format and specify a location.

**Note:** The GDL Plugin sends all components of the selected object, which may include macros and related files. The file GDList.txt records all the components to be downloaded. Check that you have received them all.

### Drag and Drop

Select the object in the Plugin 3D window (right click on WIN, control click on MAC) and drag the object into ArchiCAD. You can drag it to the floor plan, the **Object Settings** Browse dialog, or to the **Web Objects** tab of the **Load Library** menu.

**Note:** By saving the object with this method, ArchiCAD tracks the download source and date for later checking and update.

Using the direct insertion of the object method allows you to export the objects from the ArchiCAD database in IFC format.



---

# Appendices

**Appendix A: IFC Door & Window Library**

**Appendix B: Supported IFC Entities**

**Appendix C: Exported Data**

**Appendix D: Imported Data**

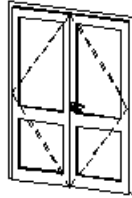
**Appendix E: PSET Data**



# Appendix A: IFC Door & Window Library

## ifcDoors

IFC\_DOORDBLSWING



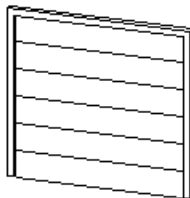
IFC\_DOORSGLSWING



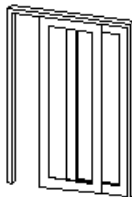
IFC\_DOORREVOLVING



IFC\_DOORROLLUP

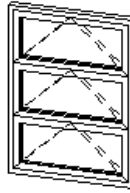


IFC\_DOORSLIDING

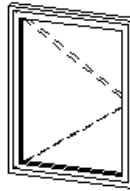


# ifcWindows

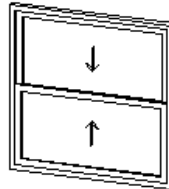
IFC\_WINDOWAWNING



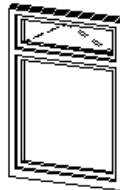
IFC\_WINDOWCASEMENT



IFC\_WINDOWDOUBLEHUNG



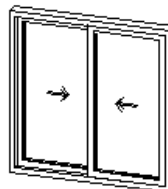
IFC\_WINDOWFIXED



IFC\_WINDOWPIVOTING



IFC\_WINDOWSLIDING



## Appendix B: Supported IFC entities:

### Supported Elements in IFC 1.51

<b>IFC</b>	<b>ArchiCAD</b>
IfcWall	Normal Wall
IfcColumn	Column or Object
IfcBeam	Beam or Object
IfcDoor	Door Object
IfcWindow	Window Object
IfcCovering	Object
IfcRoofSlab	Roof
IfcFloor	Slab
IfcBuiltIn	Object
IfcDiscreteElement	Object
IfcDistributionElement	Object
IfcElectricalAppliance	Object
IfcEquipment	Object
IfcFixture	Object
IfcFurniture	Object
IfcProxy	Object or Mesh
IfcDesignGrid	Object

### Supported Elements in IFC 2.0

All the elements supported in IFC 1.51 plus:

- IfcElectricalAppliance
- IfcPermeableCovering
- IfcRailing
- IfcRampFlight
- IfcRamp
- IfcStair
- IfcStairFlight
- IfcSystemFurnitureElement
- IfcVisualScreen

The additional elements are converted to ArchiCAD objects. The IFC attributes of the above elements are stored in the object's parameters.

# Appendix C: Exported Data

## Wall

**Brick shaped wall** with or without gable(s):

The geometric representation type is **Standard** for Attribute driven extruded segment if there is no cutting needed (without IfcHalfSpace), but it is **Advanced** for Attribute driven extruded segment with IfcHalfSpace(s).

**Revolved wall** with or without gable(s):

The geometric representation type is **Advanced** and described by Attribute driven revolved segment with or without IfcHalfSpace(s).

**Trapezoid or polygon wall** with or without gable(s):

The geometric representation type is **Arbitrary** and that means that a BRep describes the geometry of the object.

The next parameters are saved in the same way for the previous wall types:

The fill of the wall is saved as IfcMaterialLayerSet. Only the thickness and the name of material are saved.

The following parameters are saved as IfcExtendedPropertySet:

LAYERNAME,  
 INFO,  
 REFMATNAME,  
 SIDEMATNAME,  
 OPPMATNAME,  
 CONTPEN,  
 CONTLTYPE,  
 CONTPEN3D,  
 FILLPEN,  
 FILLBGPEN,  
 USECOMPENS,  
 USECOMPBGPEN

### Wall Connection:

The Add-On stores the wall connection information in the IfcRelConnectsPathElements. This item relates to two wall which are connected, and contains the information of the mode of connection i.e. START, ATEND, ATPATH, and relates to IfcLineConnectionGeometry where the geometry of the common limit is described.

## Column

*Rectangular or Circular column* with or without gable(s):

Geometric Representation Item is Standard or Advanced mode of Attribute driven extruded segment with or without IfcHalfSpace.

The fill of the column is saved as IfcMaterialLayerSet. Only the thickness and the name of material are saved.

The following parameters are saved as IfcExtendedPropertySet:

LAYERNAME,  
 INFO,  
 VENTHICK,  
 CORETYPE,  
 COREANCHOR,  
 COREPEN,  
 COREFILL,  
 COREFILLPEN,  
 COREBGPEN,  
 VENPEN,  
 VENFILL,  
 VENBGPEN,  
 CONT3DPEN,  
 CONTLTYPE,  
 MATER,  
 WRAPPING

---

## Beam

*Rectangular beam* with or without gable(s):

Geometric Representation Item is Standard or Advanced mode of Attribute driven extruded segment with or without IfcHalfSpace.

The fill of the beam is saved as IfcMaterialLayerSet. Only the thickness and the name of material are saved.

The next parameters are saved as IfcExtendedPropertySet:

LAYERNAME,  
INFO,  
BEAMSTRUCTTYP,  
HOLETYPE,  
HOLECONTUREON,  
PEN,  
LTYPEIND,  
REFPEN,  
REFLTYPE,  
SECTPEN,  
SECTFILLPEN,  
SECTBKG PEN,  
HOLEWIDTH,  
HOLEHEIGHT,  
HOLELEVEL,  
FILLNAME,  
REFMATNAME,  
SIDEMATNAME,  
OPPMATNAME,  
LEFTMATNAME,  
RIGHTMATNAME,  
SHOWALWAYS,  
SHOWABOVE,  
SHOWBELOW

## Door

If the door is from one of the IFC standard types:

IFC\_DOORDBLSWING,  
 IFC\_DOORSGLSWING,  
 IFC\_DOORREVOLVING,  
 IFC\_DOORROLLUP,  
 IFC\_DOORSLIDING

then only the opening shape is exported, and the ifc door parameters are exported in the IfcSharedPropertySet and in the IfcOccurrencePropertySet.

If the door is not an IFC standard type then the door shape is exported using the **arbitrary** geometric representation type and that means that a BRep describes the geometry of the object.

If the Opening of the door is rectangle profiled then its geometric representation will be

Standard mode of Attribute driven extruded segment. Otherwise its representation will be advanced mode of Attribute driven extruded with Arbitrary Profile Definition.

The following parameters are saved as IfcExtendedPropertySet:

INFO,  
 SILL,  
 HEADDEPTH,  
 SILLDEPTH,  
 JAMBDEPTH,  
 JAMBDEPTH2,  
 HINGEATSTART,  
 SWINGTOINTERIOR

---

## Window

If the window is IFC standard type:

- IFC\_WINDOWAWNING,
- IFC\_WINDOWCASEMENT,
- IFC\_WINDOWDOUBLEHUNG,
- IFC\_WINDOWFIXED,
- IFC\_WINDOWPIVOTING,
- IFC\_WINDOWSLIDING

then only the opening shape is exported, and the ifc window parameters are exported in the IfcSharedPropertySet and in the IfcOccurrencePropertySet.

If the window is not an IFC standard type then the window shape is exported using the **arbitrary** geometric representation type and that means that a BRep describes the geometry of the object.

If the Opening of the window is rectangle profiled then its geometric representation will be the Standard mode of Attribute driven extruded segment. Otherwise its representation will be the advanced mode of Attribute driven extruded with Arbitrary Profile Definition.

The following parameters are saved as IfcExtendedPropertySet:

- INFO,
- SILL,
- HEADDEPTH,
- SILLDEPTH,
- JAMBDEPTH,
- JAMBDEPTH2



## Roof

If the roof side surfaces are perpendicular to the base surface then a Standard or an Advanced mode of a Attribute Driven Extruded Segment is exported with Arbitrary or Trimmed Curved profile.

Otherwise an Arbitrary Geometric Representation item is exported which means a BRep.

The topMaterial, sideMaterial and bottomMaterial parameters of the roof are saved as an IfcMaterialLayerSet.

The following parameters are saved as IfcExtendedPropertySet:

- LAYERNAME,
- INFO,
- SIDETYPE,
- SECTPEN,
- SECTFILLPEN,
- SECTBKGPEN,
- SECTCONTPEN,
- USECOMPENS,
- USEBKGPEN,
- SECTFILL,
- REFMATNAME,
- SIDEMATNAME,
- OPPMATNAME,
- SHOWALWAYS,
- SHOWABOVE,
- SHOWBELOW

---

## Slab

The slab is exported as IfcFloor in a Standard or Advanced Geometric Representation mode using Attribute Driven Extruded Segment where the profile is Arbitrary or Trimmed Curve.

If the slab has any holes, they are exported as IfcOpeningElement.

The topMaterial, sideMaterial and bottomMaterial parameters of the slab are saved as an IfcMaterialLayerSet.

The following parameters are saved as IfcExtendedPropertySet:

LAYERNAME,  
INFO,  
SECTPEN,  
SECTFILLPEN,  
SECTBKG PEN,  
SECTCONTPEN,  
USECOMPENS,  
USEBKG PEN,  
SECTFILL,  
REFMATNAME,  
SIDEMATNAME,  
OPPMATNAME,  
SHOWALWAYS,  
SHOWABOVE,  
SHOWBELOW

## Object, Lamp

The objects are generally exported as IfcProxy. The shape representation is Arbitrary meaning a BRep geometric representation item.

The following parameters are saved as IfcExtendedPropertySet:

LAYERNAME,  
INFO,  
PEN,  
MAT,  
LTYPEIND,  
SECTFILL,  
SECTFILLPEN,  
SECTBGPEN,  
SECTCONTPEN,  
USEOBJMATERIALS,  
ANGLE,  
LEVEL,  
POSX,  
POSY,  
XRATIO,  
YRATIO,  
SHOWALWAYS,  
SHOWABOVE,  
SHOWBELOW

Those objects, which were imported from an IFC file, have an IFC section, where the Add-On registers the original type of the item. These objects are exported as the same kind of elements as they were originally in the IFC file.

## Zone

The Zone is exported as IfcSpace with IfcSpaceBoundary. The IfcSpace geometry is a Standard representation of an IfcAttributeDrivenExtrudedSegment defined by an arbitrary profile. The IfcSpaceBoundary has no geometry if it is a part of a wall. In this case the IfcRelSeparatesSpaces define relation of the wall and the IfcSpaceBoundary. If the IfcSpaceBoundary is logical only (not part of wall), then the IfcSpaceBoundary has an IfcPolyLoop geometry.

---

# Appendix D: Imported Data

## Wall

If the `IfcWall` is `IfcAttributeDrivenExtrudedSegment` with or without `IfcHalfSpace(s)` then it will be a straight wall with or without gable(s).

If the `IfcWall` is `IfcAttributeDrivenRevolvedSegment` with or without `IfcHalfSpace(s)` then it will be an arc wall with or without gable(s).

Otherwise the Add-On tries to convert the `IfcWall` to trapezoid wall. If this is not successful then to polygon wall and if it fails too it converts the wall into an ArchiCAD object registering in the IFC section that the original `Ifc` type was `IfcWall`.

The `IfcMaterialLayerSet` is converted to ArchiCAD fills and composite structure.

If ArchiCAD exported the wall, and the wall contains the original `IfcExtendedPropertySet` (named 'Graphisoft AC65'), then the Add-On restores these original parameters. The parameters are described above. If the wall doesn't have this kind of `IfcExtendedPropertySet` then the default parameters remain in the wall item.

The Add-On uses the Wall Connection information to adjust the wall reference line. This feature assures to create real connection between the walls.

ArchiCAD connects the wall automatically if the wall reference lines intersect.

## Column

If the `IfcColumn` is `IfcAttributeDrivenExtrudedSegment` with or without `IfcHalfSpace(s)` and the Profile is Rectangle or Circle then it will be a normal column with or without gable(s).

Otherwise the Add-On converts it to an ArchiCAD object item registering in the IFC section that the original `Ifc` type was `IfcColumn`.

The `IfcMaterialLayerSet` is converted to ArchiCAD fills and composite structure.

If ArchiCAD exported the column and the column contains the original `IfcExtendedPropertySet` (named 'Graphisoft AC65'), then the Add-On restores these original parameters. The parameters are described above. If the column doesn't have this kind of `IfcExtendedPropertySet` then the default parameters remain in the column item.

## Beam

If the `IfcBeam` is `IfcAttributeDrivenExtrudedSegment` with or without `IfcHalfSpace(s)` and the Profile is Rectangle then it will be a normal beam with or without gable(s).

Otherwise the Add-On converts it to an ArchiCAD object item registering in the IFC section that the original `Ifc` type was `IfcBeam`.

The IfcMaterialLayerSet is converted to ArchiCAD fills.

If ArchiCAD exported the beam and the beam contains the original IfcExtendedPropertySet (named 'Graphisoft AC65'), then the Add-On restores these original parameters. The parameters are described above. If the beam doesn't have this kind of IfcExtendedPropertySet then the default parameters remain in the beam item.

## Door

If the IfcPropertyTypeDef TypeDefName is a standard IfcDoor type name and the next properties are:

ParameterTakesPrecedence	true
ArbitraryShapeRepresentation	false

then the Add-On uses the standard representation of the door and sets the same parameters to the items as they are in the IfcSharedPropertySet and the IfcOccurrencePropertySet.

The standard IfcDoor types are:

IFC\_DOORDBLSWING,  
 IFC\_DOORSGLSWING,  
 IFC\_DOORREVOLVING,  
 IFC\_DOORROLLUP,  
 IFC\_DOORSLIDING

If the IfcDoor isn't a Standard IfcDoor but the door is exported by ArchiCAD and the IfcDoor contains the original IfcExtendedPropertySet where the Add-On stored the information of the ArchiCAD door item, then the Add-On tries to open the original door object and set the parameters. If the previous action failed but the IfcDoor has geometry, then the Add-On creates a door library part using the geometry and tries to generate a WallHole if the door has a non-rectangular profile.

If the door type is not recognized, and has no geometry (BREP), the IFC\_SimpleDoor is used.

## Window

If the IfcPropertyTypeDef TypeDefName is a standard IfcWindow type name and the next properties are:

ParameterTakesPrecedence	true
ArbitraryShapeRepresentation	false

then the Add-On uses the standard window item and sets the same parameters to the items as there are in the IfcSharedPropertySet and the IfcOccurrencePropertySet.

The standard IfcWindow types are:

IFC\_WINDOWAWNING,  
IFC\_WINDOWCASEMENT,  
IFC\_WINDOWDOUBLEHUNG,  
IFC\_WINDOWFIXED,  
IFC\_WINDOWPIVOTING,  
IFC\_WINDOWSLIDING

If the IfcWindow isn't a Standard IfcWindow but ArchiCAD exported the window and the IfcWindow contains the original IfcExtendedPropertySet where the Add-On stored the information of the ArchiCAD window item, then the Add-On tries to open the original window object and sets the parameters. If the previous action failed but the IfcWindow has geometry, then the Add-On creates a window library part using the geometry and tries to generate WallHole if the window has a non-rectangular profile.

If the window type is not recognized and there is no geometry (BREP) the IFC\_SimpleWindow is used.

## Roof

If the IfcRoof shape can be converted to an ArchiCAD roof then the result will be roof.

Otherwise the Add-On converts it to an ArchiCAD object item registering in the IFC section that the original Ifc type was IfcRoof.

If ArchiCAD exported the roof and the roof contains the original IfcExtendedPropertySet (named 'Graphisoft AC65'), then the Add-On restores these original parameters. The parameters are described above. If the roof doesn't have this kind of IfcExtendedPropertySet then the default parameters remain in the roof item.

## Slab

If the IfcFloor shape can be converted to an ArchiCAD slab then the result will be a slab.

Otherwise the Add-On converts it to an ArchiCAD object item registering in the IFC section that the original Ifc type was IfcFloor.

If ArchiCAD exported the slab and the slab contains the original IfcExtendedPropertySet (named 'Graphisoft AC65'), then the Add-On restores these original parameters. The parameters are described above. If the slab doesn't have this kind of IfcExtendedPropertySet then the default parameters remain in the slab item.

## **Object, Lamp**

The `IfcCovering`, `IfcBuiltIn`, `IfcDiscreteElement`, `IfcDistributionElement`, `IfcElectricalAppliance`, `IfcEquipment`, `IfcFixture`, `IfcProxy`, `IfcSite`, and `IfcDesignGrid` are imported as ArchiCAD object registering in the IFC section the original Ifc type.

If the object or lamp was exported by ArchiCAD and it contains the original `IfcExtendedPropertySet` (named 'Graphisoft AC65'), then the Add-On restores these original parameters. The parameters are described above. If the object or lamp doesn't have this kind of `IfcExtendedPropertySet` then the default parameters remain in the item.

## **Zone**

The `IfcSpace` is imported as an ArchiCAD zone item and the side boundaries will define the limit of the zone.

If ArchiCAD exported the zone and the zone contains the original `IfcExtendedPropertySet` (named 'Graphisoft AC65'), then the Add-On restores these original parameters. The parameters are described above. If the zone doesn't have this kind of `IfcExtendedPropertySet` then the default parameters remain in the zone item.

# Appendix E: PSET Data

## PreDefined PropertySets (Pset\_) for Doors and Windows in R1.5.1

### PropertySet (Pset) Name

Attribute or Relation name	Definition	Data Type or Related Object
----------------------------	------------	-----------------------------

### Shared PropertySets

<b>Pset_DoorCommon</b>	Defines properties common for all Door Types. Note that it is assumed that the door is viewed at from the exterior side (more precisely from the side in which the x-axis of the placement coordinate system points to, see diagram in geometry use case for IfcDoor). It is important e.g., for the distinction between the left and right hung).	
DoorReference	User defined reference for this door type in this project (e.g. type 'D-1')	IfcString
Description	Specific description for this type of door within this project.	IfcString
NominalHeight	Nominal Door Height as usually specified in the product information (rounded actual height)	IfcPositiveLengthMeasure
NominalWidth	Nominal Door Width as usually specified in the product information (rounded actual width)	IfcPositiveLengthMeasure
LiningWidth	Width of lining (or door frame), measured from inside of lining (at door panel) to outside of lining (at rough opening)	IfcPositiveLengthMeasure
LiningDepth	Depth of lining (or door frame), measured from front face to back face horizontally.	IfcPositiveLengthMeasure
ThresholdHeight	Height of the door threshold (if present), measured from the bottom to the top face of the threshold.	IfcPositiveLengthMeasure
ThresholdDepth	Depth of the door threshold (if present), measure from the outside to the inside face of the threshold.	IfcPositiveLengthMeasure
HardwareGroup	Reference to the hardware group used for this door type. It is implemented as a reference to the shared property set (Pset_HardwareGroup) which defines information about the door hardware.	IfcPropertySet (Pset_HardwareGroup)
ParameterTakesPrecedence	Indicates whether the parameter, given by the property type information of the door should take precedence (TRUE) over the standard shape representation using explicit geometry (see geometric use case at IfcDoor), or not (FALSE). Only valid, if the ArbitraryShapeRepresentation property is set to FALSE.	IfcBoolean



ArbitraryShapeRepresentation	Indicates whether the shape of the door is defined using the arbitrary shape representation type (see geometric use case at IfcDoor) - (TRUE), or not (FALSE). If TRUE than all parameters given by the property type information of the door, if present, only reflect abbreviations for the convenience of non CAD applications.	IfcBoolean
IsExterior	Indication whether the door type is designed for use in exterior walls (TRUE) or not (FALSE)	IfcBoolean
FireRating	Fire rating of door assembly. Given according to the national fire safety classification.	IfcString
ThermalRating	Rating for thermal transmissivity ('U' value). "0" indicates value not set.	IfcReal
AcousticRating	Rating for thermal transmissivity (Sound Transference Factor =STF) for door assembly. "0" indicates value not set.	IfcReal
LiningMaterial	Reference to the material object specifying the material of the lining.	IfcMaterial
Manufacturer	The organization that manufactured or assembled the item.	IfcOrganization
ModelLabel	The model number and/or unit designator assigned by the manufacturer of the manufactured item.	IfcString
ModelDescription	A physical description of the manufactured item as provided by the manufacturer of the manufactured item.	IfcString
<b>Pset_DoorDblSwing</b>	A double panel door where the panels are hinged at both left and right jambs and swing.	
CommonDoorProperties	Reference to the 'parent' SharedPropertySet (Pset_DoorType). Contains all other values for this type of properties which definition is equal for all doors.	IfcPropertySet (Pset_DoorType)
LeftPanel	Left door panel, as viewed from the exterior side (the side the x-axis of the placement coordinate system points to). Implemented as reference to the 'nested' property set Pset_DoorPanel. Both door panels are assumed to be of equal size. In any other case the ArbitraryShapeRepresentation (see Pset_DoorCommon) has to be used.	IfcPropertySet (Pset_DoorPanelType)
RightPanel	Right door panel, as viewed from the exterior side (the side the x-axis of the placement coordinate system points to). Implemented as reference to the 'nested' property set Pset_DoorPanel.	IfcPropertySet (Pset_DoorPanelType)
LeftPanelSwingStartAngle	As viewed in the 'XY' plane of the Door's LCS, where zero angle is aligned to the positive 'Y' axis.	IfcAngleMeasure
LeftPanelIncludedSwingAngle	Measure of arc the panel is designed to swing. Note that positive angle denotes counterclockwise arc, negative angle denotes clockwise arc.	IfcAngleMeasure
RightPanelSwingStartAngle	As viewed in the 'XY' plane of the Door's LCS, where zero angle is aligned to the positive 'Y' axis.	IfcAngleMeasure

RightPanelIncludedSwingAngle	Measure of arc the panel is designed to swing. Note that positive angle denotes counterclockwise arc, negative angle denotes clockwise arc.	IfcAngleMeasure
SwingToExteriorNotInterior	If this flag is TRUE the y-axis points to the side of the wall, the filler opens to, if FALSE the y-axis points to the opposite direction.	IfcBoolean
<b>Pset_DoorPanel</b> Properties for a Door Panel		
PanelThickness	Thickness of the door panel.	IfcPositiveLengthMeasure
PanelToLiningOffset	Offset of the inner panel face from the inner lining face (measured along the x-axis of the placement coordinate system).	IfcLengthMeasure
PanelHeight	Overall height of this panel. Should be included for convenience use by applications that cannot derive this from the geometric representation.	IfcPositiveLengthMeasure
PanelWidth	Overall width of this panel. Should be included for convenience use by applications that cannot derive this from the geometric representation.	IfcPositiveLengthMeasure
StandardPanelType	Description of the standard operating type of the panel, according to the national classification system.	IfcString
PanelMaterial	Reference to the material object specifying the material of the panel.	IfcMaterial
PanelHasOpenings	Indication whether the door panel has openings (TRUE) or not (FALSE). Only solid panels are supported by the advanced geometric representation using geometry parameters given by the property set (see property ParameterTakesPrecedence).	IfcBoolean
GlazingArea	Total area of glazing area within the door panel. Omitted if property PanelHasOpenings is set to FALSE.	IfcAreaMeasure
GlazingThickness	Thickness of the glass. Omitted if property	IfcPositiveLengthMeasure to FALSE.
GlazingMaterial	Reference to the material object specifying the material of the glazing within the frame. Omitted if property PanelHasOpenings is set to FALSE.	
Finish	Finish selection for this panel	IfcString
Color	Color selection for this panel	IfcString
<b>Pset_DoorRevolving</b> A door that revolves on a central hinge. Note: detailed description of revolving doors are not included in this release of IFC. Use standard explicit geometric representation for the shape representation of this door type.		
CommonDoorProperties	Reference to the 'parent' SharedPropertySet (Pset_DoorCommon). Contains all other values for this type of properties which definition is equal for all doors.	IfcPropertySet (Pset_DoorCommon)
<b>Pset_DoorRollup</b> A door which rolls up behind the door header. Note: detailed description of roll-up doors are not included in this release of IFC. Use standard explicit geometric representation for the shape representation of this door type.		

CommonDoorProperties	Reference to the 'parent' SharedPropertySet (Pset_DoorCommon). Contains all other values for this type of properties which definition is equal for all doors.	IfcPropertySet (Pset_DoorCommon)
<b>Pset_DoorSglSwing</b> A single panel door where the panels are hinged and swing.		
CommonDoorProperties	Reference to the 'parent' SharedPropertySet (Pset_DoorCommon). Contains all other values for this type of properties which definition is equal for all doors.	IfcPropertySet (Pset_DoorCommon)
DoorPanel	Reference to the swing door panel, given as reference to the 'nested' property set (Pset_DoorPanelType).	IfcPropertySet (Pset_DoorPanelType)
LeftNotRightSwing	Indication whether the door panel swings left hand (TRUE) or right hand (FALSE).	IfcBoolean
SwingStartAngle	As viewed in the 'XY' plane of the Door's LCS, where zero angle is aligned to the positive 'Y' axis.	IfcAngleMeasure
IncludedSwingAngle	Measure of arc the panel is designed to swing. Note that positive angle denotes counterclockwise arc, negative angle denotes clockwise arc.	IfcAngleMeasure
SwingToExteriorNotInterior	If this flag is TRUE the y-axis points to the side of the wall, the filler opens to, if FALSE the y-axis points to the opposite direction.	IfcBoolean
<b>Pset_DoorSliding</b> A door which slides in a track. Only sliding doors with one or two equally sized panels are described in detail within this release of IFC.		
CommonDoorProperties	Reference to the 'parent' SharedPropertySet (Pset_DoorCommon). Contains all other values for this type of properties which definition is equal for all doors.	IfcPropertySet (Pset_DoorCommon)
IsSplit	Indicates whether the sliding door is split into two sliding panels (TRUE) or not (FALSE). If TRUE both panels are of equal size. If FALSE only a single panel exists. In any other case the ArbitraryShapeRepresentation (see Pset_DoorCommon) has to be used.	IfcBoolean
LeftDoorPanel	Reference to the left (or single) door panel, given as reference to the 'nested' property set (Pset_DoorPanelType).	IfcPropertySet (Pset_DoorPanelType)
RightDoorPanel	Reference to the right door panel, given as reference to the 'nested' property set (Pset_DoorPanelType). Omitted if there is only a single panel (property IsSplit set to FALSE).	IfcPropertySet (Pset_DoorPanelType)
<b>Pset_WindowCommon</b> Defines properties common to all Window Types. Note that it is assumed that the windows is viewed at from the exterior side (more precisely from the side in which the x-axis of the placement coordinate system points to). It is important e.g., for the distinction between the left and right hung).		
WindowReference	User defined reference for this window type in this project (e.g. type "W-1")	IfcString
Description	Specific description for this type of window within this project.	IfcString

NominalHeight	Nominal window height as usually specified in the product information (often: rounded actual height)	IfcPositiveLengthMeasure
NominalWidth	Nominal window width as usually specified in the product information (often: rounded actual width)	IfcPositiveLengthMeasure
LiningWidth	Width of lining (outer window frame), measured from inside of lining (at window panel) to outside of lining (at rough opening)	IfcPositiveLengthMeasure
LiningDepth	Depth of lining (outer window frame), measured from front face to back face horizontally.	IfcPositiveLengthMeasure
ParameterTakesPrecedence	Indicates whether the parameter, given by the property type information of the window should take precedence (TRUE) over the standard shape representation using explicit geometry (see geometric use case at IfcWindow), or not (FALSE). Only valid, if the ArbitraryShapeRepresentation property is set to FALSE.	IfcBoolean
ArbitraryShapeRepresentation	Indicates whether the shape of the window is defined using the arbitrary shape representation type (see geometric use case at IfcWindow) - (TRUE), or not (FALSE). If TRUE than all parameters given by the property type information of the window, if present, only reflect abbreviations for the convenience of non CAD applications.	IfcBoolean
Manufacturer	The organization that manufactured or assembled the item.	IfcOrganization
ModelLabel	The model number and/or unit designator assigned by the manufacturer of the manufactured item.	IfcString
ModelDescription	A physical description of the manufactured item as provided by the manufacturer of the manufactured item.	IfcString
HardwareGroup	Reference to the hardware group used for this window type. It is implemented as a reference to the shared property set (Pset_HardwareGroup) which defines information about the window hardware.	IfcPropertySet (Pset_HardwareGroup)
LiningMaterial	Reference to the material object specifying the material of the lining.	IfcMaterial
FireRating	Fire rating of complete window assembly. Given according to the national fire safety classification.	IfcString
ThermalRating	Rating for thermal transmissivity ('U' value) of the complete window assembly. Instantiation of IfcPropertyWithUnit.	IfcReal & IfcDerivedUnit (IfcThermalTransmittance Measure)
AcousticRating	Rating for acoustic transmissivity (Sound Transference Factor =STF) for the complete window assembly.	
<b>Pset_WindowAwning</b>	A window that is hinged a the top or bottom and swings inward or outward.	IfcReal
CommonWindowProperties	Reference to the 'parent' property set (Pset_WindowCommon). Contains all other values for this type of properties which definition is equal for all windows.	IfcPropertySet (Pset_WindowCommon)

IsSplit	Indicates whether the window is split into two window panels (TRUE), or not (FALSE). If TRUE both window panels are of equal size. If FALSE only a single window panel exists. In any other case the ArbitraryShapeRepresentation (see Pset_WindowCommon) has to be used.	IfcBoolean
VerticallyNotHorizontallySplit	Indicates whether the window is split vertically (middle of width) into two window panels (TRUE), or is split horizontally (middle of height) into two window panels (FALSE). This property requires the property IsSplit to be set to TRUE.	IfcBoolean
FirstWindowPanel	Reference to the upper or left window panel (depending on property value of VerticallyNotHorizontallySplit), given as reference to the 'nested' property set (Pset_WindowPanelType).	IfcPropertySet (Pset_WindowPanelType)
SecondWindowPanel	Reference to the lower or right window panel (depending on property value of VerticallyNotHorizontallySplit), given as reference to the 'nested' property set (Pset_WindowPanelType). Omitted if there is only a single panel (property IsSplit set to FALSE).	IfcPropertySet (Pset_WindowPanelType)
FirstTopNotBottomHung	Indicates whether the first (left or upper) window panel is hung at the top (TRUE), or at the bottom (FALSE).	IfcBoolean
SecondTopNotBottomHung	Indicates whether the second (left or upper) window panel is hung at the top (TRUE), or at the bottom (FALSE).	IfcBoolean
FirstPanelSwingAngle	Measure of arc the first (left or upper) panel is designed to swing. Note: "0" is taken to be when the window panel is closed.	IfcAngleMeasure
SecondPanelSwingAngle	Measure of arc the second (right or lower) panel is designed to swing. Note: "0" is taken to be when the window panel is closed.	IfcAngleMeasure
SwingToExteriorNotInterior	If this flag is TRUE the y-axis points to the side of the wall, the filler opens to, if FALSE the y-axis points to the opposite direction.	IfcBoolean

**Pset\_WindowCasement** A window that is hinged on one side and swings inward or outward.

CommonWindowProperties	Reference to the 'parent' property set (Pset_WindowCommon). Contains all other values for this type of properties which definition is equal for all windows.	IfcPropertySet (Pset_WindowCommon)
IsSplit	Indicates whether the window is split into two window panels (TRUE), or not (FALSE). If TRUE both window panels are of equal size. If FALSE only a single window panel exists. In any other case the ArbitraryShapeRepresentation (see Pset_WindowCommon) has to be used.	IfcBoolean

VerticallyNotHorizontallySplit	Indicates whether the window is split vertically (middle of width) into two window panels (TRUE), or is split horizontally (middle of height) into two window panels (FALSE). This property requires the property IsSplit to be set to TRUE.	IfcBoolean
FirstWindowPanel	Reference to the upper or left window panel (depending on property value of VerticallyNotHorizontallySplit), given as reference to the 'nested' property set (Pset_WindowPanelType).	IfcPropertySet (Pset_WindowPanelType)
SecondWindowPanel	Reference to the lower or right window panel (depending on property value of VerticallyNotHorizontallySplit), given as reference to the 'nested' property set (Pset_WindowPanelType). Omitted if there is only a single panel (property IsSplit set to FALSE).	IfcPropertySet (Pset_WindowPanelType)
FirstPanelLeftNotRightHinged	Boolean indicating if the first panel has its hing at the left side (TRUE) or at the right side (FALSE).	IfcBoolean
SecondPanelLeftNotRightHinged	Boolean indicating if the second panel has its hing at the left side (TRUE) or at the right side (FALSE).	IfcBoolean
FirstPanelSwingAngle	Measure of arc the left panel is designed to swing. Note: "0" is taken to be when the window panel is closed.	IfcAngleMeasure
SecondPanelSwingAngle	Measure of arc the right panel is designed to swing. Note: "0" is taken to be when the window panel is closed.	IfcAngleMeasure
SwingToExteriorNotInterior	If this flag is TRUE the y-axis points to the side of the wall, the filler opens to, if FALSE the y-axis points to the opposite direction.	IfcBoolean
<b>Pset_WindowDouble Hung</b>	A double panel window in which one or both panels slide vertically in tracks.	
CommonWindowProperties	Reference to the 'parent' property set (Pset_WindowCommon). Contains all other values for this type of properties which definition is equal for all windows.	IfcPropertySet (Pset_WindowCommon)
UpperWindowPanel	Reference to the upper window panel that (eventually) slides vertically, given as reference to the 'nested' property set (Pset_WindowPanelType). Both window panels are assumed to be of equal size. In any other case the ArbitraryShapeRepresentation (see Pset_WindowCommon) has to be used.	IfcPropertySet (Pset_WindowPanelType)
LowerWindowPanel	Reference to the lower window panel that slides vertically, given as reference to the 'nested' property set (Pset_WindowPanelType). Both window panels are assumed to be of equal size. In any other case the ArbitraryShapeRepresentation (see Pset_WindowCommon) has to be used.	IfcPropertySet (Pset_WindowPanelType)

UpperPanelFixed	Boolean indicating if the upper window panel is fixed (TRUE) or operable (FALSE).	IfcBoolean
CounterBalanced	Boolean value indicating if the window hardware includes counter balancing weights for lower panel.	IfcBoolean
<b>Pset_WindowFixed</b>	A window in which the panel or panels are not operable.	
CommonWindowProperties	Reference to the 'parent' property set (Pset_WindowCommon). Contains all other values for this type of properties which definition is equal for all windows.	IfcPropertySet (Pset_WindowCommon)
IsSplit	Indicates whether the window is split into two window panels (TRUE), or not (FALSE). If TRUE both window panels are of equal size. If FALSE only a single window panel exists. In any other case the ArbitraryShapeRepresentation (see Pset_WindowCommon) has to be used.	IfcBoolean
VerticallyNotHorizontallySplit	Indicates whether the window is split vertically (middle of width) into two window panels (TRUE), or is split horizontally (middle of height) into two window panels (FALSE). This property requires the property IsSplit to be set to TRUE.	IfcBoolean
FirstWindowPanel	Reference to the upper or left window panel (depending on property value of VerticallyNotHorizontallySplit), given as reference to the 'nested' property set (Pset_WindowPanelType).	IfcPropertySet (Pset_WindowPanelType)
SecondWindowPanel	Reference to the lower or right window panel (depending on property value of VerticallyNotHorizontallySplit), given as reference to the 'nested' property set (Pset_WindowPanelType). Omitted if there is only a single panel (property IsSplit set to FALSE).	IfcPropertySet (Pset_WindowPanelType)
<b>Pset_WindowPivoting</b>	A window that is hinged at the top and bottom midpoints and which may be rotated to open	
CommonWindowProperties	Reference to the 'parent' property set (Pset_WindowCommon). Contains all other values for this type of properties which definition is equal for all windows.	IfcPropertySet (Pset_WindowCommon)
WindowPanel	Reference to the pivoting window panel, given as reference to the 'nested' property set (Pset_WindowPanelType).	IfcPropertySet (Pset_WindowPanelType)
PivotsVertically	Boolean indicating if the window panel pivots vertically (rotation axis in the middle of width)	IfcBoolean
PivotsHorizontally	Boolean indicating if the window panel pivots horizontally (rotation axis in the middle of height). If both, PivotsVertically and PivotsHorizontally, is set to TRUE, then the window pivots in both directions.	IfcBoolean
<b>Pset_WindowSliding</b>	A two panel window in which one or two panels slide in tracks horizontally.	

CommonWindowProperties	Reference to the 'parent' property set (Pset_WindowCommon). Contains all other values for this type of properties which definition is equal for all windows.	IfcPropertySet (Pset_WindowCommon)
LeftWindowPanel	Reference to the left window panel that (eventually) slides horizontally, given as reference to the 'nested' property set (Pset_WindowPanelType). Both window panels are assumed to be of equal size. In any other case the ArbitraryShapeRepresentation (see Pset_WindowCommon) has to be used.	IfcPropertySet (Pset_WindowPanelType)
RightWindowPanel	Reference to the right window panel that slides horizontally, given as reference to the 'nested' property set (Pset_WindowPanelType). Both window panels are assumed to be of equal size. In any other case the ArbitraryShapeRepresentation (see Pset_WindowCommon) has to be used.	IfcPropertySet (Pset_WindowPanelType)
LeftPanelFixed	Boolean indicating if the left panel is fixed (TRUE), or slides (FALSE).	IfcBoolean
RightPanelFixed	Boolean indicating if the right panel is fixed (TRUE), or slides (FALSE).	IfcBoolean
<b>Pset_WindowPanel Type</b>	Properties for a Window Panel. Note: all dimensions described are from the point of view of the finished face (exterior) of the window.	
FrameWidth	Width of panel frame, measured from inside of panel (at glazing) to outside of panel (at lining)	IfcPositiveLengthMeasure
FrameDepth	Depth of panel frame, measured from front face to back face horizontally.	IfcPositiveLengthMeasure
FrameToLiningOffset	Offset measured horizontally (perpendicular to the panel and glazing plane) between the inner surface of the frame and the inner surface of the lining.	IfcLengthMeasure
PanelHeight	Overall height of this panel. Should be included for convenience use by applications that cannot derive this from the geometric representation.	IfcPositiveLengthMeasure
PanelWidth	Overall width of this panel. Should be included for convenience use by applications that cannot derive this from the geometric representation.	IfcPositiveLengthMeasure
StandardPanelType	Description of the standard operating type of the panel, according to the national classification system.	IfcString
FrameMaterial	Reference to the material object specifying the material of the frame.	IfcMaterial
GlazingThickness	Thickness of the glass.	IfcPositiveLengthMeasure
GlazingMaterial	Reference to the material object specifying the material of the glazing within the frame.	IfcMaterial
<b>Pset_HardwareGroup</b>	A typed set of hardware for doors or windows	
HardwareGroupReference	User defined reference for this standard collection of hardware elements within this project.	IfcString



## Appendices

---

HardwareDescription	Specific description for this type of hardware within is project.	IfcStringth
Manufacturer	The organization that manufactured or assembled the item.	IfcOrganization
ModelLabel	The model number and/or unit designator assigned by the manufacturer of the manufactured item.	IfcString
ModelDescription	A physical description of the manufactured item as provided by the manufacturer of the manufactured item.	IfcString
Finish	Finish applied to hardware	IfcString

---

# Contents

<b>Introduction</b>	<b>1</b>
About this Guide	1
IFC Add-On	1
Version Support and Releases	1
<b>The IFC Integrated Object Model</b>	<b>3</b>
What is an IFC?	3
Inter-operability Definition	3
Why Use IFC?	4
Planning an IFC Exchange	4
Determining the Scope of a Transfer	5
Views	5
Set up a Pilot or Test	6
Useful links for Using the IFC Exchange Protocol	6
<b>Working with the IFC Translator</b>	<b>7</b>
IFC Exchange Processes	7
<i>IFC Objects versus 2D Drawings</i>	7
<i>IFC Object Identifiers</i>	8
<i>Initiating an IFC Export</i>	8
<i>Importing an IFC File</i>	9
<i>Exporting a Modified Imported IFC File</i>	9
Archiving IFC Data	10
<b>Setting Up for Import &amp; Export</b>	<b>11</b>
Owner History	11
<i>Defining the IFC 1.51 Project Team</i>	11
<i>Project Team</i>	11
<b>Exporting &amp; Importing IFC Files</b>	<b>13</b>
Open IFC File	13
Save As IFC file	14
<i>Reload Original IFC File</i>	15
<i>Reload other IFC File</i>	15
<i>No Reloading</i>	15

<b>Customizing the IFC Data</b>	<b>17</b>
Options	17
<i>Units</i>	17
<i>Export</i>	17
<i>Revolved Wall with Cutting to BREP</i>	20
<i>Property Set</i>	20
Filter	21
IFC GUID's of Selected Elements	22
Create IFC Design Grid	22
Create IFC Library	23
Edit Selected Item IFC Properties	23
<b>Configuring PSET Data</b>	<b>25</b>
IFC Property Sets (PSET)	25
IFC 2.0 PSET Configuration	26
<Groups>	27
<Link>	28
<Mapping>	29
<i>Optional Tags</i>	30
<b>Using PSET Data</b>	<b>32</b>
IFC 2.00 Options – Property Sets	32
Save as IFC File	32
Load PSET Data	32
Update with Configured PSET	33
Reporting	34
<i>Error Messages</i>	34
Example	35
<b>Accessing GDL Product Data</b>	<b>39</b>
Install the GDL Object Plugin	39
Save as IFC	40
Downloading GDL Objects	41
<b>Appendices</b>	<b>43</b>
Appendix A: IFC Door & Window Library	44
Appendix B: Supported IFC entities	46
Appendix C: Exported Data	47
Appendix D: Imported Data	55
Appendix E: PSET Data	59