

IFC Alignment Import/Export Plug-In for Autodesk AutoCAD Civil 3D - Including an IFC C# Early Binding together with the respective Generator

Felix Rampf

October 5, 2017

**IFC Alignment Import/Export Plug-In für Autodesk AutoCAD Civil 3D -
Inklusive eines IFC C# Early Bindings zusammen mit dem entsprechenden
Generator**

1 Introduction

1.1 Building Information Modelling

Introduction the wide range of building softwares using different file formats, which are quite frequently not compatible with one another, creates a challenge for civil engineers. Unlike other industries there are no legal regulations controlling the use of specific software in order to achieve optimum interoperability between the different sections such as planning offices, structural analysts, building services engineers and maintenance companies working on a big construction project. This results in every section using their own programs that are highly functional for the sector's own purpose and specifically developed for the particular industry. (Borrmann *et al.*, 2015, p. 79) Therefore, a lot of the data has to be manually entered many times for each development state constituting a serious source of errors.

To avoid these problems and to guarantee a better workflow the BIM principle has been established, which is more and more used for construction programs.

The scope of Building Information Modelling (BIM) is a continuous usage of a digital building model that serves as the foundation for all development states and sections involved in the construction process (Borrmann *et al.*, 2015, pp. 83). With the spreading of BIM and its growing importance and application, it is now more important than ever to develop, improve and establish an international standard for a working file format that is highly regulated and accessible for everyone.

On the initiative of Autodesk, a private alliance of twelve companies has been formed in 1995

to prove the benefits of interoperability between different software programs being used in the building industry. They came to the three critical conclusions that

- *“interoperability [is] viable and [has] great commercial potential(...)[,]*
- *any standards must be open and international, not private or proprietary(...)[and]*
- *that the alliance must open its membership to interested parties around the globe.”*
(buildingSMART International, 2017)

In 2008 the alliance was renamed to buildingSMART International (bSI) in order “to better reflect the nature and goals of the organization” (buildingSMART International, 2017). Furthermore, they complied with the industry’s need for a standardized file format and developed the Industry Foundation Classes (IFC), which is a comprehensive format for sharing and exchanging BIM data independently from the software manufacturer. The first version of IFC was published in 1997 as “*IFC 1.0*” reaching to the current “*IFC 4.0*” standard (published in 2013) after many development steps (figure 1).

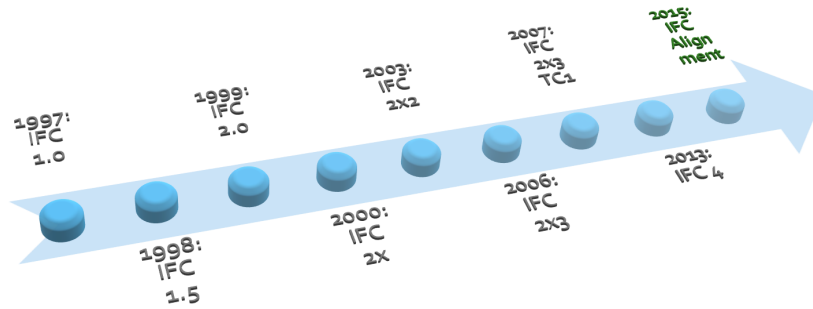


Figure 1: Releases of different IFC versions until 2011 (Borrmann *et al.*, 2015, p. 85)

1.2 Industry Foundation Classes

Before the existence of bSI, groups of engineers and scientists attempted to develop file formats similar to IFC but their approaches were usually limited to mapping geometric data. The result was the neglect of the second important part of the application of BIM, the semantics – these are essential for the data exchange between different sections – making the result unattractive for exchange scenarios. IFC files first combine both, the geometric and the semantic, aspects of BIM data. This makes it a highly complex data schema, which requires a detailed implementation approach.

However, due to an “additional standardization as an International Organization for Standardization (ISO) standard” (Borrmann *et al.*, 2015, p. 85) IFC is more attractive than ever

before. Many countries have already made the IFC format compulsory for public construction projects with Singapore, Finland, the US and Great Britain being pioneers. Although IFC has so far been focusing on mapping data for buildings, the IFC Alignment 1.0, as well as IFC Alignment 1.1 extension and future implementation of further infrastructure data such as cross sections, bridges and tunnels, indicate the considerable potential of IFC.

1.3 Software Objective and Application

The included software basically aims to simplify and expand the exchange of files containing construction data throughout the building process of a civil engineering project by providing an IFC import/export function for one of the mayor CAD programs on the market, which is used by many companies. This will hopefully reduce the amount of errors resulting from entering the same data multiple times at different stages of the construction.

Although only tested for the import and export of the IFC Alignment part, the IFC C# Early Binding already contains all the entities to facilitate the import and export of any objects listed in the IFC data schema. Furthermore, the respective Generator gives the possibility to read any EXPRESS schema (even for different file formats) and create the IFC entities accordingly, which makes the implementation of future updates to the IFC data schema really easy.

2 Software Documentation

2.1 Overview

The software consists of three different stand-alone parts:

- *IFC C# Early Binding Generator*
- *IFC C# Early Binding*
- *C3D IFC Alignment Plug-In*

Each product will be explained in more detail in the following. The basic work flow of the software is shown in figure 2.

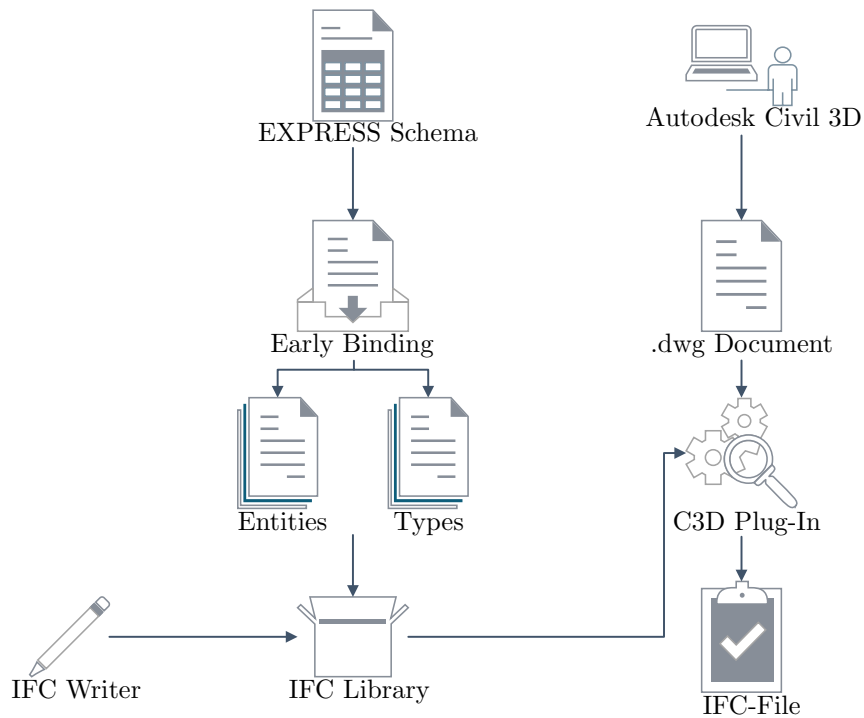


Figure 2: Work flow of the included software

2.2 Development State

Currently the software only fully supports the export (*.dwg to *.ifc). However, an import function is in the process of development and already works for importing horizontal alignments and Triangulated Irregular Network (TIN) surfaces.

Although the software was tested throughout the developing process to ensure a bug free

work flow, bugs are still possible due to the early release of the software.

Export:

- Surface
- Horizontal Alignment
- Vertical Alignment

Import:

- Surface
- Horizontal Alignment

2.3 IFC C# Early Binding Generator

IFC C# Early Binding Generator is a string reader that is able to read EXPRESS schemas. Such data schemas are used to describe all the entities, their attributes, and the respective inheritance hierarchy for a specific file format. The mapping of these entities or rather objects and their assigned attribute values are mapped following the Standard for the Exchange of Product Model Data (STEP).

In the console the user is asked to specify the EXPRESS schema and the location where he wants the entities being created. The Generator will then create all the entities with their attributes and the inheritances given in the EXPRESS schema and also adds methods so that every IFC object can be mapped following the STEP.

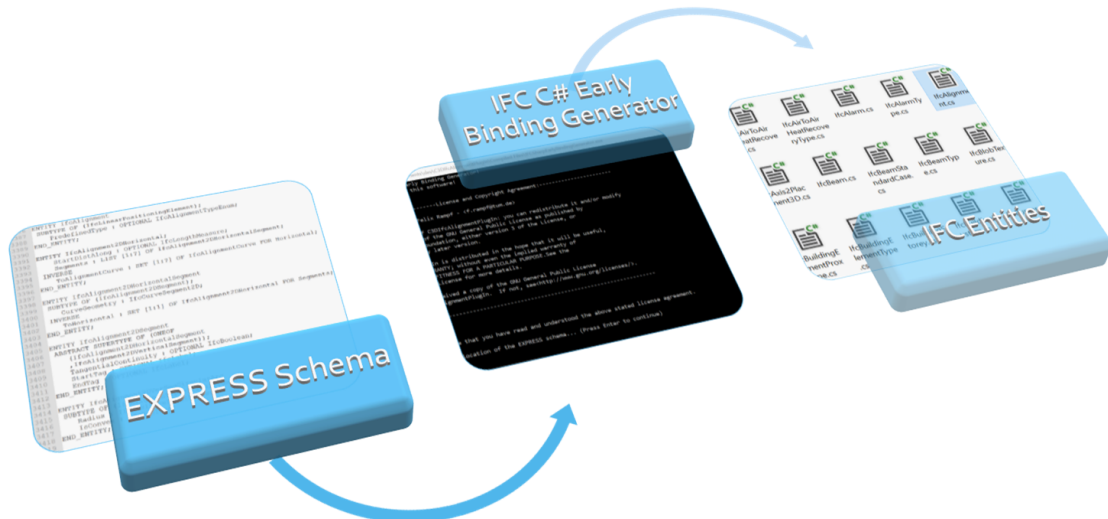


Figure 3: Principle of operation of the IFC C# Early Binding Generator

2.4 IFC C# Early Binding

The IFC C# Early Binding is a class library extends the entities created by the Generator with three additional entities:

- *IfcBase.cs*
- *IfcWriter.cs*
- *IfcReader.cs*

The *IfcBase.cs* class is only used internally and not listed in the IFC EXPRESS schema. *IfcWriter.cs* facilitates the mapping of IFC objects according to the STEP. It also adds the line numbers and header section to generate a complete *.ifc file. The last additional class – *IfcReader.cs* – is able to read a STEP string and creates all the mapped IFC objects including their attribute values. This class is only working for the import of the IFC Alignment 1.1 extension with certain limitations but can be extended for remaining entities.

The additional entities are compiled as a class library (*.dll) together with all the entities created by IFC C# Early Binding Generator. This *.dll file can then be used as a reference in **any** other C# code.

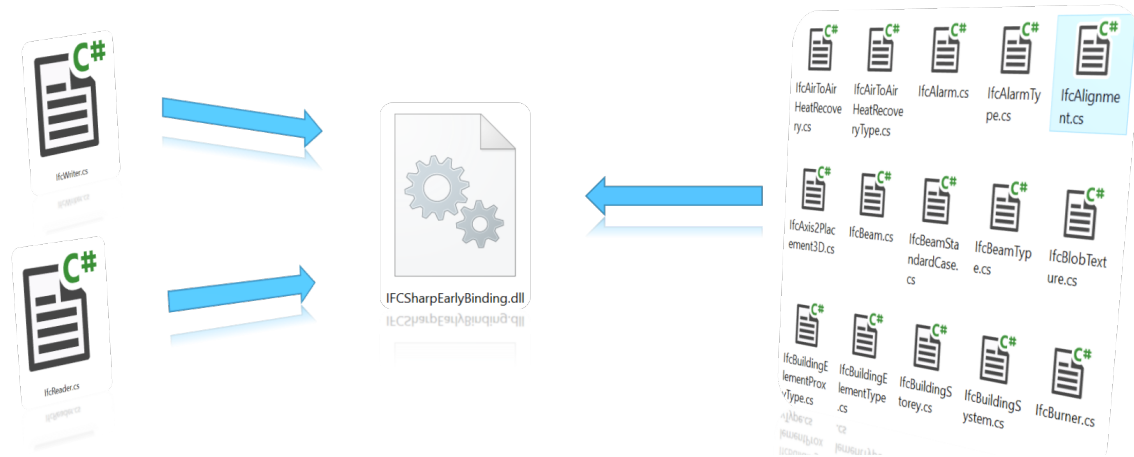


Figure 4: IFC C# Early Binding containing the additional entities

2.5 C3D IFC Alignment Plug-In

The C3D IFC Alignment Plug-In reads all the alignments including all alignment entities, the respective profiles, and surfaces contained in the current Autodesk AutoCAD Civil 3D drawing and creates the IFC objects accordingly. At this point metres have to be used as units. The actual import and export is facilitated by a new ribbon button showing in the “Add-Ins” tab looking like figure 5. A coordinate reference system, which will then be mapped onto the created *.ifc file, can also be further specified by the user. Any other entity that should

be mapped from the drawing onto the *.ifc file can easily be implemented to the liking of the customer.

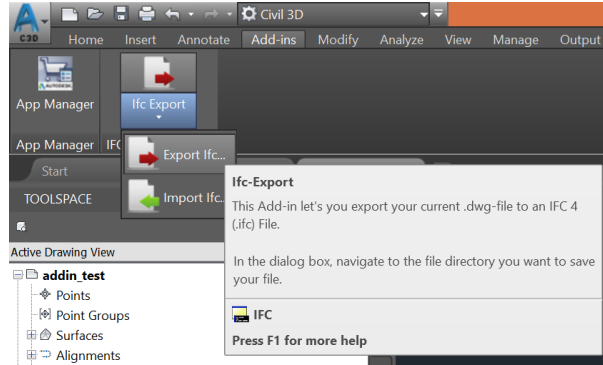


Figure 5: Split Button created by the plug-in in Civil3D

3 Scope

The software described in this paper will be available at:

<https://bitbucket.org/FlixFix/c3difcplugin>

More detailed documentation on the plug-in can be found in my bachelor's thesis contained in the BitBucket project and published on the university's website:

<https://www.cms.bgu.tum.de/de/lehre/abgeschlossene-arbeiten/abgeschlossenebachelorarbeiten>

Due to my lack of experience and the early release, I am aware of the fact that it will not be bug free at this stage. However, it does work correctly for the Civil 3D plug-in and thus should work for implementation within other programs as well. It is published in the hope that the code might be improved in cooperation with engineering companies and more experienced software developers to make it more usable and 'bullet proof'.

This could mark a further step towards making IFC a lot more accessible for a wider range of people. I consider that this should be the main goal for everyone using and spreading IFC – as a file format that focuses on interoperability rather than individual development and profit for a single company.

We believe significant improvements in cost, value and environmental performance can be achieved through the use of open sharable asset information in the creation and operation of civil infrastructure and buildings worldwide. (buildingSMART International, 2017)

References

- Borrmann, A., König, M., Koch, C. & Beetz, J. (2015). *Building Information Modeling: Technologische Grundlagen und industrielle Praxis*. VDI-Buch. Wiesbaden: Springer Vieweg.
- buildingSMART International (2017). buildingSMART International Homepage. <http://buildingsmart.org/about/about-buildingsmart/>. Accessed: 2016-05-09.