Int. J. on Recent Trends in Engineering and Technology, Vol. 11, No. 1, July 2014

Extracting Building Data from BIM with IFC

Raninder Kaur Dhillon[1], Mayur Jethwa[2], Hardeep Singh Rai[3]
[1] Department of Information Technology, Guru Nanak Dev Engineering College, Punjab, India
Email: raninderdhillon@gmail.com
[2] Department of Computer Science & Engineering,Guru Nanak Dev Engineering College, Punjab, India
Email:veenajethwa@gmail.com
[3] Department of Civil Engineering,Guru Nanak Dev Engineering College, Punjab, India
Email:hardeeprai@gmail.com

***Abstract*— The building activity i.e. BIM generates a great number of data and information of various kinds. Several programs that use this data for various applications have been developed in different regions of the world. After briefly introducing the Industry Foundation Classes,a standard for exchanging building data, this paper presents an integrated framework which applies Information Technology (IT) and IFC which extract geometric and material layers' data from BIM in IFC data model. This IFCModelParser parses IFC files and strores data in text as well as excel files which can be used for further applications like code compliance.**

***Index Terms*— Building Information Modelling, BIM, CAD, IFC,Parser**

I. INTRODUCTION

In the past 10 years, design tools in the Architecture, Engineering and Construction (AEC) industry has been improved from 2D modeling to 3D modeling. Building Information Modelling (BIM) as a powerful set of design management's tool has recently become a topical research area. Technical developments in Building Information Modeling (BIM) offer the potential for a new generation of software tools that can automate the checking of compliance with building codes. BIM provides the parametric information which makes possible accurate cost estimates, simulations, scheduling, and energy analysis. BIM also facilitates coordination with engineering, fabricating and construction partners.

There has been a growing trend in the Architectural, Engineering, Construction / Facilities Management (AEC/FM) industry to use building information modelling (BIM) tools. The BIM data produced by these tools comes in various formats. Most of these data formats are proprietary due to their commercial nature [1]. There are several non-proprietary building data models currently available. One of these is the Industry Foundation Classes (IFC), an open standard data model developed by the International Alliance for Interoperability (IAI), which was formed in 1995 by an international consortium of organisations in the AEC/FM industry. The IFC data model allows the building geometry and materials property information to be exported from a BIM authoring tool to a standard format such as the IFC compliant STEP (Standard for Exchange of Product Model Data) physical data file [2].

II. INDUSTRY FOUNDATION CLASSES (IFC)

A. IFC Objectives

The IFC schema has as its scope to: "to define a specification for sharing data throughout the project life-

DOI: 01.IJRTET.11.1.1500

cycle, globally, across disciplines and across technical applications" [4].The Industry Foundation Classes (IFC) are designed and maintained by the "International Alliance for Interoperability" (IAI). Members of the IAI are architects, engineers, facility managers, academic institutions, government agencies, technical associations and software vendors.

The IAI is organized in Chapters. A Chapter represents a country or a group of countries acting together. There are 11 chapters with 19 countries and more than 500 member companies. The Industry Foundation Classes provide a specification of a data model that covers the domain of building information. It can be used as a shared data model or integrated data base by many occupation groups. In contrast to exchange plans via drawing files like dxf or dwg, the IFC exchange is strictly model based. A wall is not a set of lines but an object with specified attributes and relations [3].

B. Technologies Used In IFC Development

The IFC schema is specified using the EXPRESS data definition language, as defined in ISO10303-11:1994. IFC data files are clear text files following the STEP physical file format, as defined in ISO10303-21:1994. IFC files following the STEP physical file format are governed by the IFC schema in EXPRESS.

C. Structure of the STEP physical file

The ".ifc" file is a STEP physical file, SPF, and thereby a structured ASCII text file. The basic SPF structure divides each file into a header and a data section. The header section has information about:

- the IFC version used
- the application that exported the file
- the date and time when the export was done
- (often optionally) the name, company and authorizing person of the file

Example

```
HEADER;
FILE_DESCRIPTION(('IFC 2x platform'),'2;1');
FILE_NAME(
'Example.dwg',
'2005-09-02T14:48:42',
('The User'),
('The Company'),
'The name and version of the IFC preprocessor',
'The name of the originating software system',
'The name of the authorizing person');
FILE_SCHEMA(('IFC2X2_FINAL'));
ENDSEC;
```

The data section contains all instances for the entities of the IFC specification. These instances have a unique (within the scope of a file) STEP Id, the entity type name and a list of explicit attribute.

Example

```
DATA;
#7=IFCCARTESIANPOINT((0.,0.,0.));
#8=IFCDIRECTION((0.,0.,1.));
#9=IFCDIRECTION((1.,0.,0.));
#10=IFCAXIS2PLACEMENT3D(#7,#8,#9);
ENDSEC;
```

D. Architecture of IFC

There are four layers in the IFC Model. The layers follow the "gravitation" concept which means that elements of a certain layer can only refer to entities of the same or a lower layer.

The IFC Model Architecture for IFC 2x consists of the following layers.

- Resource Layer
- Core Layer
    - -Kernel
    - -Extensions
- Interoperability Layer
- Domain Layer

1. Resource Layer: This layer contains the fundamental concepts expressed as entity types such as geometry (point, line and curve) topology (vertex, edge, face and shell), geometric model (CSG, B-Rep, Geometric Set). The elements of this layer can be referenced by elements of all other layers.
2. Core Layer: The Core layer provides the basic structure of the IFC object model and defines most general concepts that will be specialized by higher layers of the IFC object model.

The Core includes two levels of generalization:

1. The Kernel
2. Core Extensions

The Kernel provides all the basic concepts required for IFC models within the scope of the current IFC Release. It also determines the model structure and decomposition. The Kernel can be seen as a template model that defines the form in which all other schema within the model are developed. Its constructs are very general and are not AEC/FM specific, Core Extensions, provide extension or specialization of concepts defined in the Kernel. More specifically, they extend those constructs for use within the AEC/FM industry.

Each Core Extension is a specialization of classes defined in the Kernel and develops further specialization of classes rooted in the IfcKernel. Additionally, primary relationships and roles are also defined within the Core Extensions.

3. Interoperability Layer: This layer defines basic concepts for interoperability between different domain extensions. Shared building elements like beam, door, roof, window or ramp are defined in this layer.
4. Domain Layer: Domain Models provide further model detail within the scope requirements for an AEC/FM domain process or a type of application. Examples of Domain Models are Architecture, HVAC, FM, Structural Engineering etc.

The primary IFC element hierarchy is based on the accessing structure,

Project > Sites > Buildings > Stories > Spaces> Elements

That is, a project is the top-level container made up of one or more sites. A site is a container of one or more buildings (and all of their parts). A building contains one or more stories (and its parts) and a story is made up of one or more spaces (and its parts) and spaces are defined of one or more elements [5].

III. BUILDING INFORMATION MODELLING (BIM)

A Building Information Model is a digital representation of the building process that facilitates exchange of information in digital format. BIM provides the parametric information which makes possible accurate cost estimates, simulations, scheduling, and energy analysis. BIM also facilitates coordination with engineering, fabricating and construction partners.

A. BIM Software and Tools

BIM models (e.g. Architectural, Structural, MEP etc) can be created through a number of software products. Some of these products are from Autodesk (Revit Architecture, Revit MEP, AutoCAD MEP) Graphisoft (ArchiCAD, ArchiCAD MEP) and Bentley (BIM) [6]. The associated functions (e.g. 4D scheduling or 5D costing) can be interoperable software. The following table (Table 1) describes list of the most common BIM software available from different companies world-wide [15]:

IV. TECHNOLOGICAL DEVELOPMENT

Automated data exchange between commonly used software tools for building design, construction, and operation has been a goal of the buildings industry for decades.

The BuildingSMART has developed an open standard for data exchange between building industry software tools called the Industry Foundation Classes (IFCs). The IFCs provide software developers and users of Building Information Models (BIMs) a standard for sharing consistent, accurate building information amongst software tools used throughout a facility's life cycle [7]. Eastman et al. [8] in their paper reviewed five IFC based efforts to automate rule checking in buildings. These are Singapore CORENET project, the HITOS project by Norwegian Statsbygg, the effort by the Australian Building Codes Board, the International Council in the US and the General Services Administration effort. Schutz et al [9] in their paper developed a context sensitive BIM that integrates structural data for a real-time analysis in an emergency command system such as fires, floods etc. This system is based on a CAFM-database and an IFC-based architectural model.

Fazio et al. [10] in their paper presents an integrated framework which applies Information Technology (IT) and the international standard Industry Foundation Classes (IFC) to ensure that the building envelope satisfies energy requirements as well as other requirements such as moisture and thermal performance, concurrently. The framework is designed to extract geometric and material layers' data of a house from CAD drawings in IFC data model, link to performance evaluation applications, such as HOT2000 and MOIST3.0, and compare evaluation results with a set of criteria. Dimyadi et al [11]the paper describes work undertaken to share building geometry and other information with the Fire Dynamics Simulator (FDS) fire simulation model using IFC data model. The selected building model was created in Revit Building 9 and exported as an IFC2x2 SPF. This file was then processed using a specifically developed parser tool to extract the geometrical information and this information then used to create an FDS input file. Yang and Xu [12] in their paper provides a pilot implementation of online code checking system and considered Object Oriented Code knowledge model and the IFC-compliant information model as more effective representations for design data. In their paper [13], Greenwood et al proposed a development of code compliance checking, by defining within the IFC model, a domain extension for England and Wales Building Regulations using the established methodologies of the BuildingSMART Alliance. Raninder and H.S. Rai in their paper [14] in their paper presents BIM challenges and issues in developing countries, with India as a case study.

TABLE I. BIM SOFTWARE

Software	Company	Website
ArchiCAD	Graphisoft	www.graphisoft.com
Architectural Desktop	Autodesk	http://usa.autodesk.com
Bentley Architecture Bentley Systems	Bentley Architecture Bentley Systems	www.bentley.com
Data Design System	Data Design System	www.dds-bsp.co.uk
DProfiler	Beck Technology	http://dpearth.com
JetStream	NavisWorks	www.navisworks.com
REVIT	Autodesk	http://usa.autodesk.com
SDS/2	Design Data System	www.dds-bsp.co.uk
Tekla Structures	Tekla Corporation	www.tekla.com
VectorWorks Architect	module of Vectorworks From Nemetschek	www.nemetschek.net
ONUMA Planning System	ONUMA Inc. Add ons for ArchiCAD	www.onuma.com

The leading architectural CAD packages, such as Architectural Desktop (ADT) (Autodesk, 2001), ArchiCAD (Graphisoft, 2002), Microstation TriForma (Bentley System, 2002), and AllPlan (Nemetschek, 2002), are currently providing/developing facilities to support the IFC format in architectural and structural design. Another active implementation area is in the development of the IFC toolboxes as underlying IFC information supporting platforms for storage, management, exchange and sharing of IFC product model data. The EDM package (EPM Technology, 2002) provides full set of APIs for processing IFC and XML objects with Web support. The IFC Toolboxes from Eurostep (Eurostepsys, 2002a) and PDTec (PDTec, 2002) also facilitate the IFC object reading, writing, and accessing from other software applications.

V. OPEN SOURCE IFC LIBRARIES FOR WORKING WITH BUILDING DATA

There are various commercial as well as open source tools available for working with IFC data. Some of the open source libraries available are:

1. IFC-SDK: This is an Open Source C++ library for reading and writing IFC files. It does not depend on any external libraries and can be compiled on most modern compilers. It was tested on Windows (g++ and VC++ 2003+2005) and on Linux (g++ 3 and 4).
2. Ifc-OpenShell: IfcOpenShell is an open source (LGPL) software library that helps users and software developers to work with the IFC file format. IfcOpenShell uses Open CASCADE (the Open CASCADE Community Edition) internally to convert the implicit geometry in IFC files into explicit geometry that any software CAD or modelling package can understand.
3. IfcPlusPlus: IfcPlusPlus is an open source C++ class model, as well as a reader and writer for IFC files in STEP format.

Features:

- Easy and efficient memory management using smart pointers.
- Parallel reader for very fast parsing on multi-core cpu's
- Additionally, there's a simple IFC viewer application, using Qt and OpenSceneGraph.

4. Open IFC Java Toolbox: Features:
    - open source (non-commercial use only)

- latest Java technology
- runs directly from the internet
- tested with real project models
- complete suite for processing, visualisation of IFC STEP files
- drives the industry towards Building Information Modeling (BIM)

5. Ifc-dotnet: It is a library which provides .Net classes and serializers/deserializers for working with Industry Foundation Classes. Ifc-dotnet is very much in Alpha stage of development Currently Ifc-dotnet supports the serialization (export) and deserialization (import) of STEP and ifcXml files to .Net classes representing the IFC2x3 standard.
6. xBIM toolkit: The eXtensible Building Information Modelling (xBIM) Toolkit The xBIM Tookit (eXtensible Building Information Modelling) is an open-source, software development BIM tool that supports the BuildingSmart Data Model. xBIM is implemented on the .NET platform. xBIM allows developers to read, create and view Building Information (BIM) Models in the IFC format. There is full support for geometric, topological operations and visualisation.

VI. IFC MODEL PARSER: A FRAMEWORK TO EXTRACT DATA FROM IFC FILES

There are various commercial and open source libraries for working with IFC data but most of them have weaknesses like:

- Most of them work with IFC 2X3 which is not the latest version of IFC.latest is IFC 2X4.
- Some projects are outdated, not currently maintained like IFCsvr.
- Do not provide all the information in detail.

Considering these weaknesses in current toolboxes, a framework is created named IFCModelParser which works on IFC files created by BIM software like Revit, Archicad etc. IFCModelParser is GUI based application to parse IFC Models and to export them to Excel Spreadsheets and Text File. The Project is created in Java and is capable of determining the minute details of the model. The test has showed that the project can handle creation of hundreds of worksheets per model with no problem. It can parse IFC 2X3 as well as IFC 2X4 files. IFC 2x4 file parser is thoroughly tested and now currently is in beta version. IFC 2x3 is not thoroughly tested and is currently in alpha version. Fig. 1 shows the framework of IFCModelParser in which a building model is created using a BIM software (can be ArchiCAD or REVIT or any other). This model is exported in IFC format (which can be IFC 2X 3 or IFC 2X 4). IFCModelParser works on .ifc file to create a text file or excel file as output which contains extracted data from IFC entities.

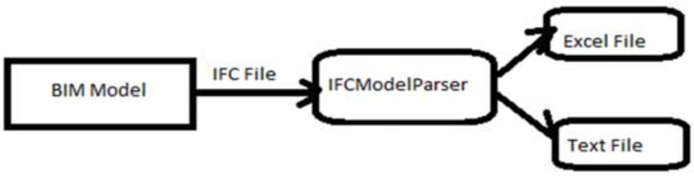

Fig. 1: Framework of IFCModel parser

Features of IFCModelParser

- Dynamic EXPRESS parser: On -the-fly parsing of any Industry Foundation Classes(IFC) schema (IFC2x3, IFC4 are supported by default).
- Saves to Text file and Excel : The output of parsed model is stored in Excel SpreadSheet and Text file. And can write upto hundreds of page of worksheet in few minutes..
- Console Output : Have the capability to output the result in the console.
- Hyperlinks: For navigation with the excel files.

VII. RESULTS AND DISCUSSION

This parser is written in Eclipse-standard-kepler-32 as it has great auto-complete and code generation functions. It's also completely free and open source. This parser aims to smooth the process of sharing and getting information from the building models. It supports all IFC versions and has an open system structure to be extended by its community users. It contains various packages like com.gne.ifcapplication, com.gne.ifcapplication.controller, com.gne.ifcapplication.model.project, com.gne.ifcapplication.view, com.gne.ifcapplication.viewexcel

It contains about 18 classes. Fig. 2 shows various classes used, their properties and methods used to get properties of IFC entities.

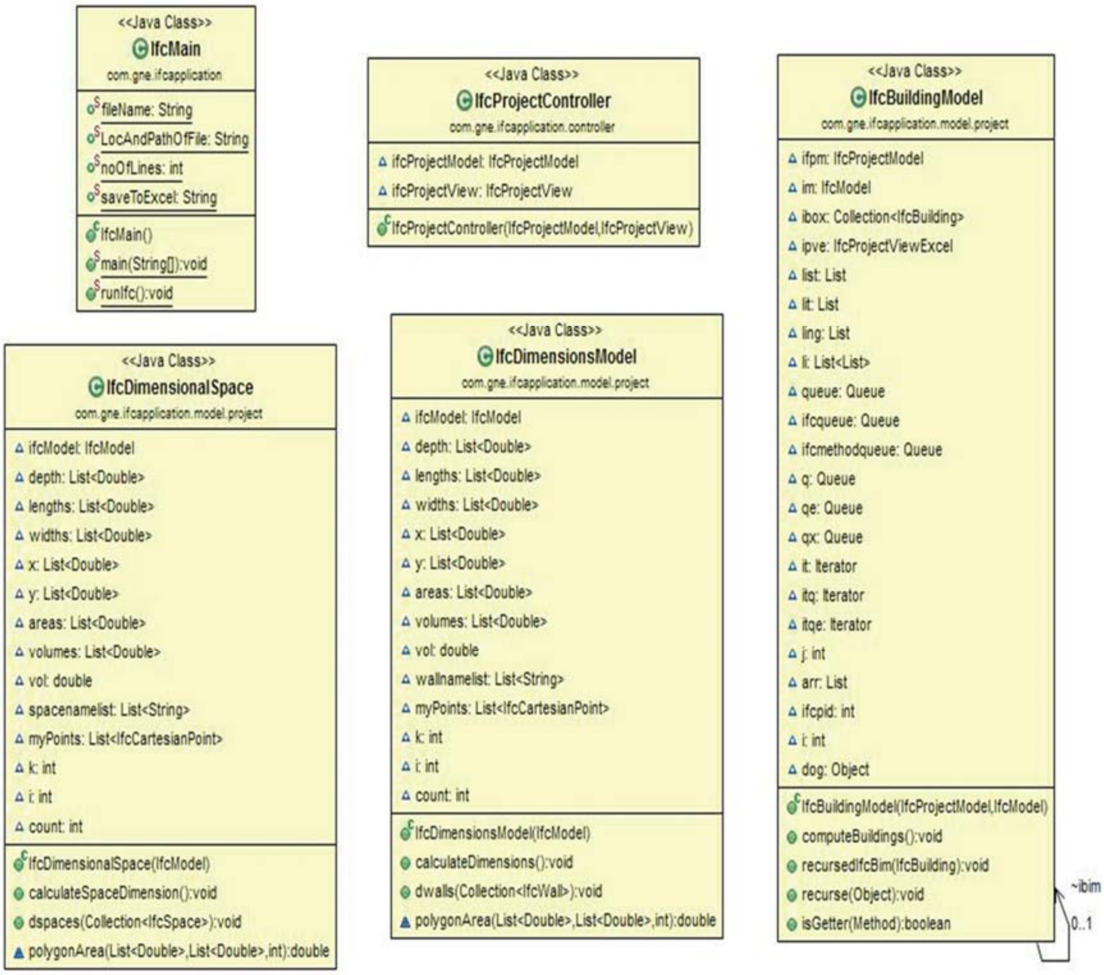

Fig. 2: UML Class diagrams for IFCModel parser

IfcMain is the main class that imports various classes and generates excel output as shown in the code below.

```
  public static String fileName = "C:\\ifc 4 examples\\ifc 4 real examples\\HelloWall_IFC4\\Release\\example.ifc"; // "C:\\example.ifc";
  public static String LocAndPathOfFile = "ifclogging.txt";
  public static int noOfLines = 7 + 5 + 1;
  public static String saveToExcel = "c:\\IfcProject.xlsx";
      public static void main(String[] args)
  {
      saveFiles svf = new saveFiles();
      saveToExcel = svf.savefi();
      fileName = new FileChooser().setFileName();
      IfcFileProgress progress = new IfcFileProgress(true);
      PrintToFile.printToFile(LocAndPathOfFile, true);
      System.out.println(fileName);
      runIfc();
      PrintToFile.delelines(noOfLines);
```

```
        progress.End();
    }\\
    /**
     * Run ifc.
     */
    public static void runIfc()
    {
        System.out.println(saveToExcel);
        IfcProjectModel ifcprojectmodel = null;
        IfcProjectView ifcprojectview = null;
        try
        {
              ifcprojectmodel = new IfcProjectModel(fileName);
              ifcprojectview = new IfcProjectView(ifcprojectmodel);
        } catch (Exception e)
        {
              System.out.println("Some problem in Project Model Ifc4");
        }
        try
        {
      IfcProjectController ifcprojectcontroller = new IfcProjectController
(ifcprojectmodel, ifcprojectview);
        } catch (Exception e)
```

…………………….

A. Output in Text file

As described in Fig. 1, IFCModelParser generates two types of output: text file and excel file. Following figure (Fig. 3) shows the text file output containing different IFC entities.

```
class ifc2x3javatoolbox.ifc2x3tc1.SET  =
ifc2x3javatoolbox.ifc2x3tc1.SET<ifc2x3javatoolbox.ifc2x3tc1.IfcRe
presentationContext>  =   getRepresentationContexts   =  class
ifc2x3javatoolbox.ifc2x3tc1.IfcProject   =   #268437230
IfcGeometricRepresentationContext
ifc2x3javatoolbox.ifc2x3tc1.IfcGeometricRepresentationContext
[getStepLine, getPrecision, getTrueNorth,
getCoordinateSpaceDimension, getWorldCoordinateSystem,
getStepLineNumber, getContextIdentifier, getContextType]

class ifc2x3javatoolbox.ifc2x3tc1.SET  =
ifc2x3javatoolbox.ifc2x3tc1.SET<ifc2x3javatoolbox.ifc2x3tc1.IfcRe
presentationContext>  =getRepresentationContexts= class
ifc2x3javatoolbox.ifc2x3tc1.IfcProject   =   #268439815
IfcGeometricRepresentationContext
```

Fig. 3: Text File

Various functions are defined to compute information regarding various parts of a building like

- computeSpaces() - Method in class com.gne.ifcapplication.model.project.IfcSpaceModel compute spaces.
- computeStoreys() - Method in class com.gne.ifcapplication.model.project.IfcStoreyModel compute storeys.
- computeWalls() - Method in class com.gne.ifcapplication.model.project.IfcWallMaterial compute walls.

B. Output in Excel

This execl file contains hyperlinks to other entities in excel sheet as shown in Fig. 4. Further it contains different sheets, representing information about different IFC entities like the one shown below gives information about IFCProject. Others can be IFCSite, IFCBuildingStorey, IFCWall1, IFCwall2 etc.

IfcProject	Not used in the file	StepLine	Phase
StepLine	IFCPROJECT('0eAxUR9d		
Phase	Project Status	EncodedValue	DecodedValue
LongName		EncodedValue	DecodedValue
StepLineNumber	41		
RepresentationContext	IfcGeometricRepresen	StepLine	Precision
UnitsInContext	IfcUnitAssignment	StepLine	Units
Name	Project Number	EncodedValue	DecodedValue
OwnerHistory	IfcOwnerHistory	StepLine	OwningUser
GlobalId	0eAxUR9d92zBqyhldLT	EncodedValue	DecodedValue
EncodedValue	'Project Status'		
DecodedValue	Project Status		
EncodedValue	''		
DecodedValue			
StepLine	IFCGEOMETRICREPRESE		
Precision	1.0E-9		
StepLineNumber	36		
CoordinateSpaceDimen	3		
WorldCoordinateSysten	IfcAxis2Placement3D	StepLine	StepLineNumber
ContextType	Model	EncodedValue	DecodedValue
StepLine	IFCUNITASSIGNMENT((		
Units	IfcConversionBasedUn	Name	StepLine
Units	IfcConversionBasedUn	Name	StepLine
Units	IfcConversionBasedUn	Name	StepLine
Units	IfcConversionBasedUn	Name	StepLine
Units	IfcSIUnit	Name	StepLine

Fig. 4: Excel file output containing hyperlinks to other IFC instances

Excel file generated from IFCModelParser contains all the information contained in building elements. e.g. If an IFC file doesn't contain information about wall length, height, area etc. then this parser computes it from co-ordinates and display it.

Part of IFC file :

```
#45 = IFCWALL('3Ep4r0uuX5ywPYOUG2H2A4', #2, 'Wall xyz', 'Description
of Wall', $, #46, #51, $);
#46 = IFCLOCALPLACEMENT(#36, #47);
#47 = IFCAXIS2PLACEMENT3D(#48, #49, #50);
#48 = IFCCARTESIANPOINT((0., 0., 0.));
#49 = IFCDIRECTION((0., 0., 1.));
#50 = IFCDIRECTION((1., 0., 0.));
#51 = IFCPRODUCTDEFINITIONSHAPE($, $, (#52, #81));
#52 = IFCSHAPEREPRESENTATION(#20, 'Body', 'Brep', (#80));
#53 = IFCCLOSEDSHELL((#60, #67, #70, #73, #76, #79));
#54 = IFCPOLYLOOP((#55, #56, #57, #58));
#55 = IFCCARTESIANPOINT((0., 0., 0.));
#56 = IFCCARTESIANPOINT((0., 200., 0.));
#57 = IFCCARTESIANPOINT((5400., 200., 0.));
#58 = IFCCARTESIANPOINT((5400., 0., 0.));
```

Following figureFig. 5 shows the values of parameters which are not provided explicitly. They are deduced from above IFC file with IFCModelParser.

Wall Calculated N	Wall Calculate	Wall Calcul	Wall Calculated	Wall Calculated Area
Outer Wall Right	2800	4800	300	1440000
Outer Wall Left	2800	4800	300	1440000
Outer Wall Back	2800	3800	300	1140000
Outer Wall Front	2800	3800	300	1140000

Fig. 5: Excel file output containing parameters of a wall

VIII. Conclusion

IFCModelParser is a powerful tool to generate data from IFC Model in a human friendly format. Its extensive navigational capability makes it very perfect FDS fire simulation, automated building code checking, energy performance evaluation applications. It aims to smooth the process of sharing and getting information from the building models. Future work is using this parser for various applications like rule checking on buildings.

References

[1] Vanlande R., Nicolle C. and Cruz C., "IFC and building lifecycle management", *Automation in Construction*, Vol. 18(1), pp. 70-78, 2008.

[2] ISO 10303-21, "Industrial Automation System – Exchange of Product Model Data – Part 21:Implementation Methods; Clear Text Encoding of the Exchange Structure," *International Organization for Standardization*, 2002.

[3] Clemen C. and Gründig L., "The Industry Foundation Classes – Ready for Indoor Cadastre?", *Shaping the ChangeXXIII FIG Congress*, Munich, Germany, October 8-13, 2006.

[4] Liebich T., *IFC 2x Edition 3 Model Implementation Guide*, Version 2.0 May 18, 2009.

[5] CHAPTER NINE, Industry Foundation Classes, *BuildingSMART chapters* - http://buildingsmart.com/organization/chapter

[6] Brewer G., Gajendran T., Le Goff R., "Building Information Modelling (BIM): an Introduction and International Perspectives", *Research Report*, July 2012.

[7] Robert J. Hitchcock , Justin Wong ,"Transforming IFC Architectural View BIMS for Energy Simulation: 2011", *Proceedings Building Simulation Conference*, *Sydney, Australia*, November 14-16, November 2011.

[8] Eastman, C., Lee, J. min, Jeong, Y. suk, & Lee, J. kook., "Automatic rule-based checking of building designs", *Automation in Construction*, 18(8), 1011-1033,2009. Elsevier B.V. doi:10.1016/j.autcon.2009.07.002.

[9] Schütz, R., Bernoulli, T., Wießflecker, T., and Walder, U., "A Context Adaptive Building Information Model for Real Time Structural Analysis in a Disaster Management System", *International Conference on Information Technology in Construction, Santiago, Chile,* 2008.

[10] Fazio P., He H. S., Hammad, A., Horvat, M., "IFC-Based Framework for Evaluating Total Performance of Building Envelopes", *Journal of Architectural Engineering*, Vol. 13, No. 1, March 2007, pp. 44-53

[11] Dimyadi J., Spearpoint M., Amor R., "Sharing Building Information using the IFC Data Model for FDS", *Proceedings of 9th IAFSS Symposium on Fire Safety Science, Karlsruhe, Germany,* 21-26 September,2008.

[12] Yang, Q. Z., & Xu, X. Design knowledge Modelling and software implementation for Building Code Compliance checking. *Building and Environment*, 39(6), 689-698. (2004). doi:10.1016/j.buildenv.2003.12.004

[13] Greenwood, D., Lockley, S., Malsane, S., & Matthews, J. Automated compliance checking using building information models. *COBRA* 2010 (pp. 266-274).

[14] Dhillon Raninder K., Rai Hardeep, "Code Compliance with Building Information Modeling", *Fourth International Conference on Recent Trends in Information, Telecommunication and Computing - ITC 2013*.

[15] RAIC Practice Builder, "Building Information Modelling", *Article* 2007. http://www.raic.org/practice/bim/bim-practice-builder_e.pdf