

## PalArch's Journal of Archaeology of Egypt / Egyptology

### EXTRACTING CODE COMPLIANCE DATA FROM IFC WITH PYTHON LANGUAGE

*Er. Narinder Singh<sup>1</sup>, Prof. Harvinder Singh<sup>2</sup>, Prof. Hardeep Singh Rai<sup>3</sup>.*

<sup>1</sup>Research scholar, IKGPTU, Kapurthala Road Jalandhar

<sup>2,3</sup>Guru Nanak Dev Engineering College Ludhiana

Email: <sup>1</sup>nsmanjhpur@gmail.com, <sup>2</sup>hs.gndec@gmail.com, <sup>3</sup>hardeep.raai@gmail.com

**Er. Narinder Singh, Prof. Harvinder Singh, Prof. Hardeep Singh Rai: Extracting Code Compliance Data from IFC With Python Language -- Palarch's Journal Of Archaeology Of Egypt/Egyptology 17(9). ISSN 1567-214x**

**Keywords: BIM , IFC , PYTHON, SCHEMA**

#### ABSTRACT

The study aimed to identify the barriers to the implementation of the e-learning system (Blackboard) among female students at the Community College at Imam Abdul Rahman bin Faisal University. To achieve the objectives of the study, the researcher adopted the descriptive approach. The study sample which was chosen randomly consisted of (300) female students from the community college at Imam Abdul Rahman bin Faisal University. The researcher built the study tool "questionnaire" which covers four areas: Administrative barriers, technological barriers, financial (physical) barriers, and social barriers. The data was processed by the statistical method. The findings indicated that the areas of the barriers which faces the implementation of the blackboard system among female students at the community college at Imam Abdul Rahman bin Faisal University were as follows: Social barriers which ranked first and with a (high degree) followed by financial barriers (moderate degree), administrative barriers (moderate degree) and the technical barriers (low degree) and with the mean 2.60, 2.20, 2.13 and 1.37 respectively. The study recommended promoting the technical aspect in the university through using the blackboard system and providing specialists and technicians with suitable training regarding the use of this system and holding educational sessions for students and community members to enhance the concept of e-learning.

## 1. Introduction

BIM has fully changed the way of planning, analyzing, designing, building and management of an Infrastructure. Before the invention of BIM the building designs were dependent upon the 2D drawings, which causes loss on information and coordination [1]. With the advancement in BIM technology, the problem gets solved mostly. It provides a common platform to all engineers involved in AEC market. so that they can work together on the information of the infrastructure [2, 3, 4].

Earlier Architecture worked on different software & Engineer developed the model in different software, which become very difficult to consult their information with each other [2, 3]. But with introduction of BIM, a lot of BIM software came into market which stored the information of a structure in the standard format of IFC(industry Foundation Classes) . Hence, a software tool is created in python language, which can fetch the properties of structural elements (BEAM, COLUMN, WALL, SLAB etc.) of Building for application of checks according to the Indian Standard Codes (IS) [2, 5, 3, 6, 4, 7, 8, 9, 10, 11]..

### 1.1. Building Information Modeling (BIM)

Conventional Building plan was fully dependent upon the 2D Drawings which results in poor grouping and loss of processed data [1, 12]. But with adoption of BIM, this problem has been reduced to maximum extent. Because BIM uses 3D Models in spent of 2D Drawings . BIM has changed the way of Planning, Building, executing and Management of any Infrastructure[1, 12].It create a platform on which all engineers can work together with the processed data of the structure [1, 12].

It was very difficult to job synchronously by engineers before the introduction of the BIM when each engineer develops a model in different softwares. They were unable to exchange data with each other which lead to compatibility issues [1]. Then with the advent of BIM software these compatibility issues were resolved. All the BIM softwares are capable of collecting the information of each part of structure in the standard format (.IFC) Industry Foundation Classes [2, 5, 3, 6, 4, 7, 13, 14, 15, 16].

### 1.2 Industry Foundation Classes (IFC) & Development

The Industry Foundation Classes (IFC) are most important tool and important resource for interoperability in the(AEC) construction industry and its development is as below:

- 1985-1990 -> CAD era
- Drawing Interchange Format (DXF) –
- 1990-1995 -> STEP-Exchange(-G)
- 1995-Today -> IFC
- Releases - IFC 2x4, IFC4
- Data File Formats: STEP, XML
- 2017-beyond -> ifcOWL/RDF, ?

IFC format is useful to describe the model of buildings [17, 12]. It describes

the sharing and exchange of information among IFC compatible software. It stores each entity of the infrastructure that are known as objects. It describes how to show the individual properties of the objects[12].IFC holds data like geometry, quantities, calculations, etc. And the data is further used by professionals such as Architectural Engineer, Structural Engineer, Plumbing Engineer, Electrical Engineer[2, 3].

### 1.3 IFC File Structure

The IfcObjectDefinition is one of the pillars of data structure. The IFC file format is a true interoperability standard with a precise data structure and IFC structure can be considered as a storage system for organizing and transferring digital data to facilitate interoperability between the various BIM professionals within a following workflow.

IFC file structure of an IFC model defines a model integrated with hundreds of entities that are organized in a hierarchical manner. Some examples of these

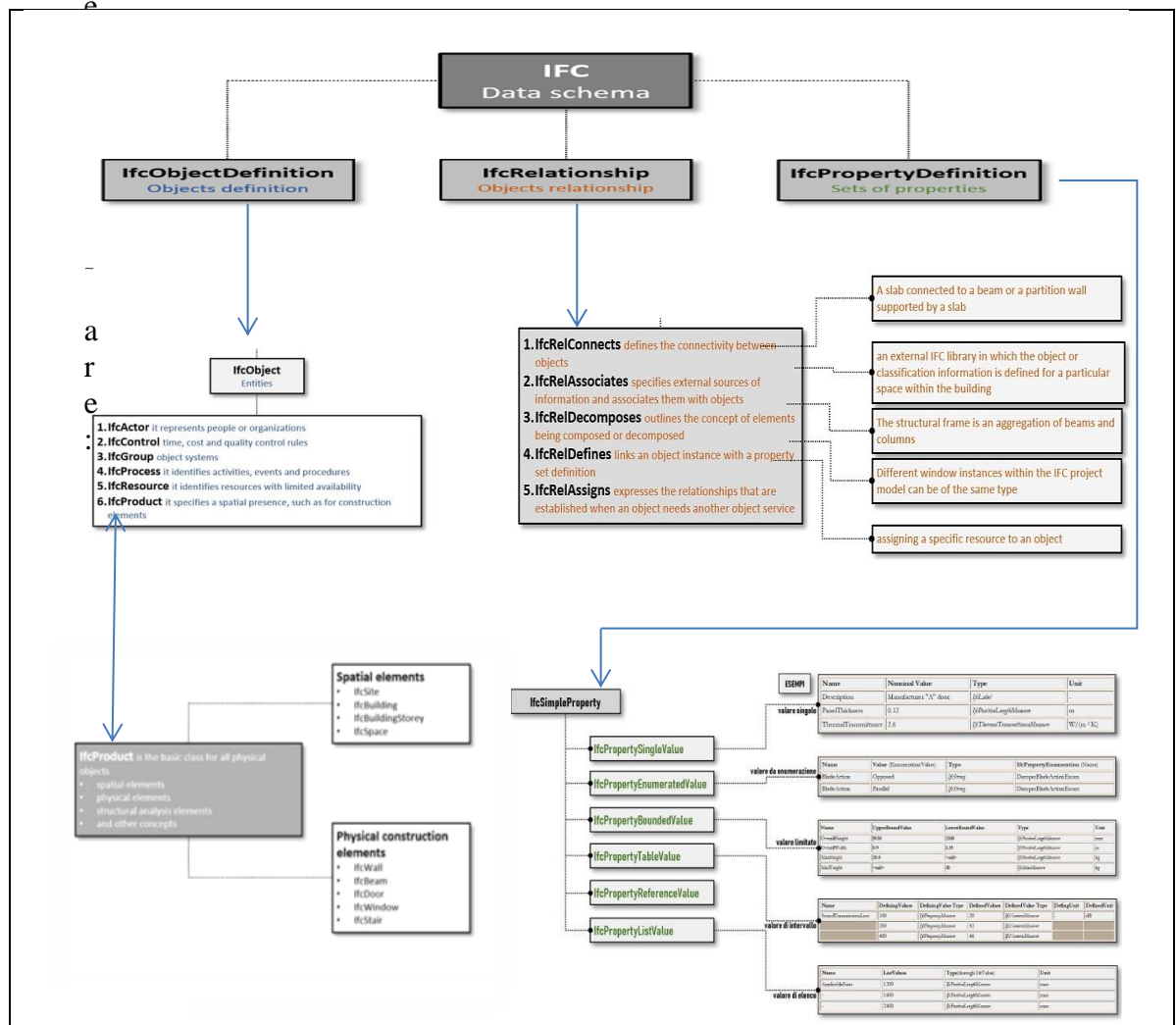


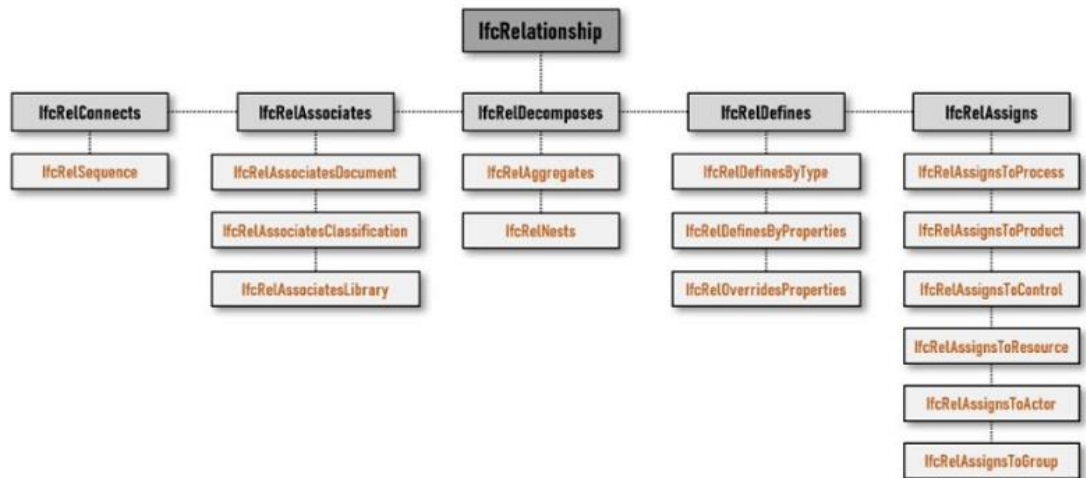
Figure 1: IFC file data structure

- Construction elements, such as IfcWall , IfcWindow , IfcDoor
- Geometric elements, such as IfcExtrudedAreaSolid (solid extrusion from an area)
- Basic constructs, such as IfcCartesianPoint .
- The IFC logic is based on 3 fundamental concepts, which represent the structure itself:
  - The IfcObjectDefinition defining the entities themselves (objects)
  - The IfcRelationship defining the relationship between entities
  - The IfcPropertyDefinition defining the properties associated with the entities.

**1.4 Ifcrelationship :**

Describes different types of relationships. Specifically, there are 5 basic types of relationships in the IFC model, which are subtypes of the IfcRelationship class:

1. IfcRelConnects is a connectivity relationship that connects objects based on certain criteria. For example, a slab connected to a beam or a partition wall supported on a floor slab
2. IfcRelAssociates serves to relate external sources of information and associates them with objects or property definitions. The association relationship is one-way. An external IFC library in which the object or the information on the classification for a particular space inside a building is defined, could represent a valid example



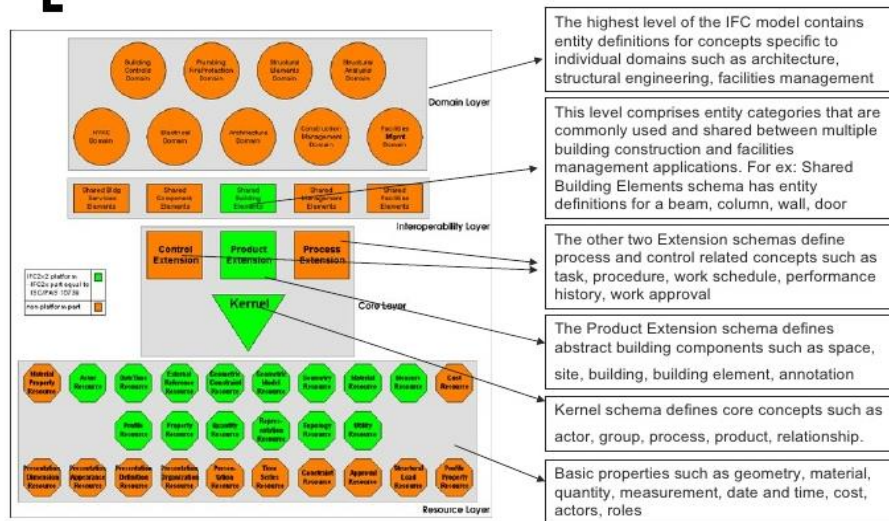
*Figure 2. IFC file structure / Scheme showing the IfcRelationship subdivision*

3. IfcRelDecomposes defines the general concept of elements that are composed or decomposed. Decompositions imply a hierarchy, where, the definition of the whole depends on the definition of the single parts and vice versa. Therefore, a cost item can be included in other items or a structural frame can be considered an aggregation of beams and columns
4. IfcRelDefines is a relationship that allows an object instance to inherit a property set. For example, different instances of windows within the IFC model

may be of the same type, (belonging to the same catalogue or manufacturer). Thereby they share the same information

5. IfcRelAssigns makes explicit assignment relationships that arise when an object needs the services of another object. For example, we can assign a certain resource to an object. Each of these 5 types allows the development of further relationships between objects, types, properties, etc. (subtypes trees) as shown above

### 1.5 Industry Foundation Classes (IFC) Express-G



*Figure 3: Schemetic diagram of IFC Model*

IFC schema is created by using graphical notation that is known as EXPRESS-G i.e. international data definition language [4, 17, 12]. It easily helps to make the development and reviewing of the model. IFC exists in another XML form, which is known as ifcxml [8].

## 2. Research Methodology

The .Ifc file is read simply as a simple text file character by character

For file handling in Python following expression. “r”, for reading. “w”, for writing. “a”, for appending. “r+”, for both reading and writing is used e.g.

```
f = open('test.txt', 'w')
#Python code to demonstrate
# write character by character
f = open("test.txt",'w')
f.write("my first file\n")
f.write("This file\n\n")
f.write("contains three lines\n")
f.close()
#Python create test.txt file with output
"my first file
This file
```

contains three lines”  
 After getting character by character all the character are joined to form a string until a semicolon occurs.  
 Those lines are not processed for which # exists at the starting of the string.  
 The commonly used function of strings i.e. split() is used to split the string in various subparts with multiple delimiter.

```
#program for split()
import re
text = 'The quick brown\fox jumps*over the lazy dog.'
print(re.split('; |, \\*|\\n',text))
# with output
['The quick brown', 'fox jumps', 'over the lazy dog.']
```

Figure 4: split function

The commonly used function of strings i.e.join() is used to join the multiple sub-parts into single.

```
I = ['The quick brown', 'fox jumps', 'over the lazy dog']
print(''.join(I))
# with output
['The quick brown fox jumps over the lazy dog']
```

Figure 5: join function

### 2.1 hierarchical Relationship Between Ifc Entities

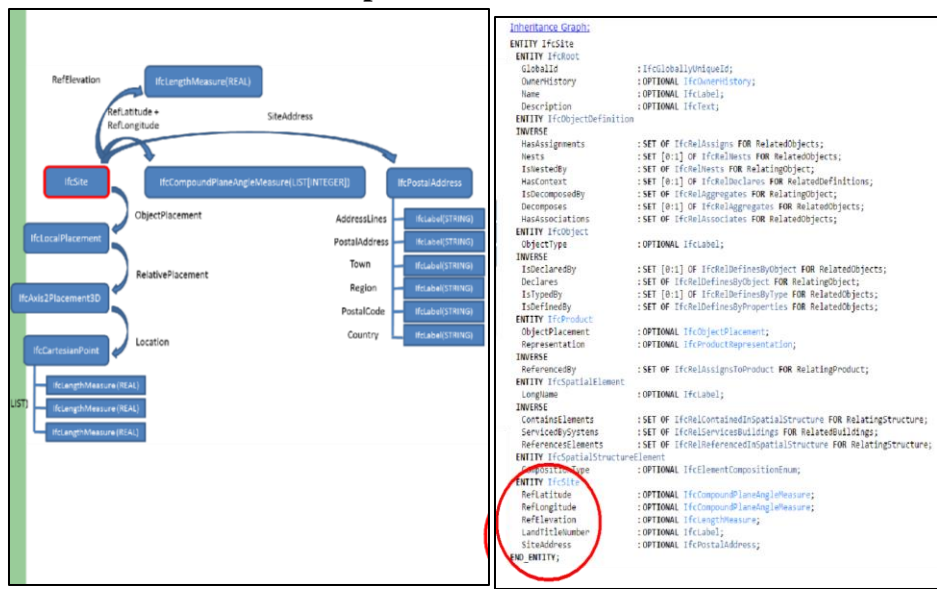


Figure6: Hierarchy

Methodology of working of Ifc format is shown graphically as how tagging processes to search the required output

**IFC Model**

Figure 6: Inheritance Graph of entities

**2.2 Sample Example Of Ifc File In Text Form**

STEP Physical File produced by: The EXPRESS Data Manager Version 5.02.0100.07 : 28 Aug 2013

```

* Module:          EDMstepFileFactory/EDMstandAlone
* Creation date:   Sat Jul 13 15:49:59 2019
* Host:           DESKTOP-RAALN2P
* Database:       C:\Users\AppData\Local\Temp\{3162C3F6-5262-44F7-B887-11EA1D78F2AA}\ifc
  * Database version: 5507
* Database creation date: Sat Jul 13 15:49:54 2019
* Schema:         IFC2X3
* Model:          DataRepository.ifc
* Model creation date: Sat Jul 13 15:49:54 2019
* Header model:   DataRepository.ifc_HeaderModel
* Header model creation date: Sat Jul 13 15:49:54 2019
* EDMuser:        sdai-user
* EDMgroup:       sdai-group
* License ID and type: 5605 : Permanent license. Expiry date:
* EDMstepFileFactory options: 020000

```

```

.....
FILE_DESCRIPTION(('ViewDefinition [CoordinationView_V2.0]'),2;1);
FILE_NAME('Project Number','2019-07-13T15:49:59',(''),(''),'The EXPRESS
Data Manager Version 5.02.0100.07 : 28 Aug 2013','20180328_1600(x64) -
Exporter 19.0.1.1 - Alternate UI 19.0.1.1','');
FILE_SCHEMA(('IFC2X3'));
ENDSEC;
DATA;
#1= IFCORGANIZATION($,'Autodesk Revit 2019 (ENU)',$,,$);
#5= IFCAPPLICATION(#1,'2019','Autodesk Revit 2019 (ENU)','Revit');
#6= IFCCARTESIANPOINT((0.,0.,0.));
#11= IFCDIRECTION((1.,0.,0.));
#270= IFCAXIS2PLACEMENT3D(#6,$,$);
#2864=IFCOPENINGELEMENT('0_qacqdsTCuupFa9EEvqFn',#41,'Basic
Wall:Generic - 200mm:383810',$,'Opening',#2863,#2858,$);
#273= IFCCARTESIANPOINT((0.,1.08002495835535E-12));
#275= IFCAXIS2PLACEMENT2D(#273,#23);
#276=
IFCRECTANGLEPROFILEDEF(.AREA.,'1000X2100mm',#275,2100.,1000.);
#277= IFCCARTESIANPOINT((1300.,-1424.,1050.));
#279= IFCAXIS2PLACEMENT3D(#277,#15,#21);
#280= IFCEXTRUDEDAREASOLID(#276,#279,#19,3048.);
#281= IFCSTYLEDITEM(#280,(#258),$);
#284= IFCSHAPEREPRESENTATION(#102,'Body','SweptSolid',(#280));
#286= IFCPRODUCTDEFINITIONSHAPE($,$,(#284));

```

```
#288= IFCAXIS2PLACEMENT3D(#6,$,$);
#289= IFCLOCALPLACEMENT(#219,#288);
#290= IFCOPENINGELEMENT('2X8JR_puD129HUzckoR6iG',#41,'Basic
Wall:Generic - 200mm:381109',$,'Opening',#289,#286,$);
```

### 2.3 Sample Example For Parcing Of .Ifc File

python Programming Language is used for wall attributes with output pasted below:

Python 2.7.16 (v2.7.16:413a49145e, Mar 4 2019, 01:37:19) [MSC v.1500 64 bit (AMD64)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

```
===== RESTART: C:\Python27\wall new.py
=====
```

```
# -*- coding: cp1252 -*-
import ifcopenshell
ff = ifcopenshell.open('C:\Python27\masonry.ifc')
wall = ff.by_type("IfcWallstandardcase")[0]
print wall
#267=IfcWallStandardCase('3L0b4BA1b9wRqiy6SvBDii',#41,'Basic
Wall:Generic - 200mm:381109',$,'Basic Wall:Generic -
200mm:1044',#219,#265,'381109')
print(wall.Representation)
#265=IfcProductDefinitionShape($,$,(#224,#263))
print(wall.ObjectPlacement)
#219=IfcLocalPlacement(#129,#218)
print(wall.HasOpenings)
#295=IfcRelVoidsElement('1CuejoVD9FueFp2MOmBFLk',#41,$,$,#267,#290
),)
print(wall.ConnectedTo)
#2510=IfcRelConnectsPathElements('1LmmAoCRPD6wTg67$ubp1I',#41,'3L
0b4BA1b9wRqiy6SvBDii|3L0b4BA1b9wRqiy6SvBDjO','Structural',$,#267,#58
0,(,),(,).ATEND,..ATSTART.),)
#2515=IfcRelConnectsPathElements('0Fz8qmuLD7pwxKf3gsgxyk',#41,'3L0b
4BA1b9wRqiy6SvBDii|3L0b4BA1b9wRqiy6SvBDjQ','Structural',$,#267,#630,
(,),(,).ATSTART,..ATSTART.)

#2519=IfcRelConnectsPathElements('22B1kEbQfA_896Mh6mCoCT',#41,'3L
0b4BA1b9wRqiy6SvBDii|3L0b4BA1b9wRqiy6SvBDjU','Structural',$,#267,#68
0,(,),(,).ATSTART,..ATEND.)
print(wall.ContainedInStructure)
#2310=IfcRelContainedInSpatialStructure('1w2kgyXAD2C9c7QsP4npsv',#41,
,$,$,(#173,#267,#366,#426,#476,#530,#580,#630,#680,#730,#830,#884,#984,#
1034,#1134,#1184,#1260,#1310,#1360,#1410),#130),)
print(wall.Decomposes)
()
print(wall.FillsVoids)
```



```

()
print(wall.HasAssociations)
#2393=IfcRelAssociatesMaterial('20vhLKyr9m81aPIR20782',#41,$,$,(#267),
#312),)
print(wall.HasStructuralMember)
()
=====
=====

```

- At the end, we matches the tag to IfcCartesianPoint and id with ref. we de-code to get the coordinates of two points that are Starting Point and Ending Point of the structural entities
- By collecting all the data in the digitized form related to structural entities of building we can perform the required analysis , design or other information of the building entities.

### 3. Result And Discussion

The conventional technique was tedious, difficult and prone of errors. The Python library IfcOpenShell is used to interact with IFC data stored and to apply the structural checks on the entities, which are being formulated after fetching data from the .ifc file produced by any 3D Structural software such as Revit, Free CAD, ArchiCAD etc.The calculations are being performed to measure the Dimensions,Area, quantities and various properties of material used in construction. The calculations are performed as the .ifc file is having the data related to the above entities. So the parameters as , Span effective length effective height of various wall, coulumn etc. are being used to calculate Selenderness Ratio of these entities. The Python Language is being used to create the dependencies as it is fully object oriented programming language and there are various In-Built Libraries to perform calculations. The Ifcopenshell ,Build, Panda Libraries is fruitful application for design and analysis and makes further changes possible before actual commencement and make it more efficient

### 4. Conclusion And Future Aspects

It is Concluded that how the IFC file of different domains is useful to retrieve the various structural data of a structural entity. The Research is emphasis on the automated code compliance and python Language is being used for the extraction process. Automated Code Compliance saves time,manual efforts and provide accurate results.

But, still it is not fully developed and much effort is required for completion. The values extracted from the IFC of structural domain can be used to check performance, analysis compatibility on the BIM model. Hence improve the efficiency of checks applied according to the Indian Standard Codes. Reporting could be improved and improvement in graphical representation of the results.

## References

- M.Fischer, Linking cad and expert systems for constructability reasoning, Proceedings of the Fifth International Conference on Computing in Civil and Building Engineering 322 (10) (1993) 156370.
- P. Nawari. O. Nawari, Ph.D., A framework for automating codes conformance in structural domain, Journal of Computer and In-formation Technology 1 (1).
- Q.Z. Yang ,Xingjian Xu, Design knowledge modeling and soft-ware impleme David Greenwood, Stephen Lockley,Sagar Malsane,Jane methews, Automated compliance checking using building information models, The Construction, Building and Real Estate Research Conference of the Royal Institution of Chartered Surveyors.
- Susmita Sinha,Anil Sawhney, Extracting information from building information models for energy code compliance of building envelope. URL [Http://blog.sanauulla.info/2013/05/23\\_parsing-xml-using-dom-sax-and-stax-parser-in-java](http://blog.sanauulla.info/2013/05/23_parsing-xml-using-dom-sax-and-stax-parser-in-java).
- Sharpe R, Oakes S., Advanced IT processing of Australian standards and regulations, The International Journal of Construction Information Technology 3 (1) (1995) 7389.
- Zarli A, Debras P., Integration of corba and web technologies in the vega dis., Proceedings of the European Conference on Integration in Manufacturing, Gothenburg, Sweden 3 (1) (1998) 184 97.
- S. M. Inc., J2ee overview the java 2 platform enterprise edition. URL <http://java.sun.com/j2ee/>
- Building, C. A. of Singapore (BCA), e-plancheck system in CORENET. URL <http://www.bca.gov.sg/>
- C. D., Development and implementation of automated code compliance checking in the U.S., International Code Council, 2007.
- Delis E.A.,Delis A., Automatic firecode checking using expert-system technology,journal of computing in civil engineering, ASCE 2 (1995) 141–156.
- Ding, L., Drogemuller, R., Rosenman, M., Marchant,Gero, D. J., Automating code checking for building designs: in: K. brown, k. Hampson, P.Brandon (eds.), clients driving construction innovation, Cooperative Research Centre (CRC) for Construction Innovation (2006) 113–126.
- Eastman, C. M., Jaemin Lee, Yeonsuk Jeong, Jinkook Lee, Review automatic rule-based checking of building designs, Automation in Construction 18 (2009) 10111033.E. Technology, Express data manager. URL <http://www.EPMtech.jotne.com>
- notation for building code compliance checking, Building and Environment 39 (6) (June,2004) 689698.

C. Wu, Ifc model viewer to support nd model application, Automation in Construction 15 (2) (2006) 178– 185.