AUTOMATED DESIGN OF STRUCTURES IN BIM ENVIRONMENT

Abstract

BIM, a cutting-edge information technology-based technique, is gaining widespread adoption in the AEC industry. The construction sector is increasingly recognizing the value of incorporating innovative technologies. BIM is being employed by various project stakeholders, including planners, designers, engineers, and subcontractors, to enhance project performance. However, the successful implementation of BIM hinges on the seamless integration and exchange of the extensive data within the BIM model across different project phases and among various parties. The primary obstacle to this success is interoperability, primarily driven by the industry's reliance on proprietary software for data exchange. For the project team to successfully achieve interoperability, vendor-neutral tools and software must be used for communication. In this study, data extraction from an architectural BIM model of a building is done using open-source software to create a structural model. To emphasize that openBIM comprises more than just a standardized file format, an integrated process is designed. This work's main goal is to encourage the use of open-source technologies.

Keywords: BIM, IFC, ifcopenshell, interoperability, openBIM, python, structural model.

Author

Satinder Kaur Khattra
Assistant Professor,
Department of Civil Engineering,
COAE&T Punjab Agricultural University,
Ludhiana, Punjab, India.

I. INTRODUCTION

In the field of information technology, automation is also quickly taking shape in the construction sector. Nearly every stage of the life cycle of a construction project is currently being altered by technological improvements, from the initial conceptualization to the construction phase to project completion and even deconstruction. Engineers are continuously seeking innovative approaches to enhance their performance in today's fast-paced economy, aiming for increased efficiency, faster execution, and the judicious selection of optimal construction methods to achieve unprecedented milestones. Given the distinctive and multifaceted nature of construction projects, the reliance on paper-based drawings and collaboration among professionals from various fields, along with the growing complexity of such projects, necessitates the adoption of specialized techniques within the Architecture, Engineering, and Construction (AEC) industry [1]. Effective coordination and collaboration across the many specialized disciplines engaged in a construction project are essential to its successful completion, since they ultimately improve the process's overall efficiency and quality.

The application of building information modeling (BIM) has gained popularity as a result of recent advancements in new breakthroughs. In light of the data innovation that is increasingly being adopted in the AEC industry, BIM is an advanced technique. Building information models, or BIM models for short, are the end product of the BIM process. They are an integrated entity made up of data on the complete structure from several disciplines, such as structural, MEP, environmental, and construction management data. The American National BIM Standard (NBIMS) indicates that
"BIM is characterized as a computerized portrayal of physical and practical qualities of an office. As such it fills in as a common data resource for information about an office outlining a trustworthy justification for decisions during its lifecycle from origin ahead" [2].

BIM implies different things to different experts. For some expert BIM is a product, to others it is an interaction for planning and recording data of structures. Some say it is a comprehensive way to deal with plan, development and support of a structure. There can be numerous abbreviations for this idea. The sole hypothesis behind it is to gather the structure foundation and development data at a typical spot, so that shared information might be used by the individuals in various phases of a project.

Today BIM is an area of most extreme interest which can give a game-changing answer for address the difficulties experienced in the AEC business. It is a recently arisen digital technology for the generation of virtual 3D model having project related information and giving experts the new experiences and instruments to all the more proficiently plan, develop and oversee structures and framework. BIM is characterized by [3] as quite possibly of the most encouraging improvement in the AEC enterprises by giving carefully built exact virtual models. Individuals working in many spots, with various devices and in various times can join their work to make a typical outcome. It also thinks about better examination of conditions through reproduction.

BIM Models are the records holding semantic data connecting with building parts which can be traded for dynamic about a development project. Exactly when used fittingly will save time, cash and simplify the development interaction. The development business has begun to see the advantages of BIM. Draftsmen, creators, project administrators, and workers for hire are utilizing BIM to work on their cycles. A few contextual investigations have demonstrated the way that BIM can diminish revise. Development costs are turning out to be more unsurprising. The progressions made in the model are consequently conveyed all through the undertaking in this manner dispensing with the coordination blunders.

1. **Interoperability and openBIM:** The adoption of Building Information Modeling (BIM) delivers substantial advantages to the AEC industry. BIM consolidates comprehensive data about every aspect of a structure into a unified repository, enabling anyone involved in the project to access this information for various purposes. Building simulations, energy analysis, 4D scheduling, quantity estimation, structural modeling, and quantity estimate are just a few of the creative uses for BIM data. However, it has been noted that a number of businesses that use BIM do not completely capitalize on its advantages because of issues such as poor professional communication during the design and construction stages. The integration and interchange of the wealth of information included within BIM models among various project stakeholders is a clear indicator of the effective application of BIM and across various project stages, a concept technically referred to as interoperability. The process of importing and exporting data between different software applications is known as interoperability. In this procedure, one software program imports the model for later use after exporting it in a common format for data sharing and interchange [4]. To do structural design and analysis, for example, a structural engineer imports the architectural model; for collaborative design, the engineer then exports an updated model to the architect. There has always been a need for communication between people and tools in the same sector, therefore this idea is not new. Nonetheless, this idea has become more well-known in the context of BIM since information interchange and integration are essential to the accomplishment of BIM projects. The primary cause for the challenges faced by some firms in fully leveraging BIM is the... But since information interchange and integration are essential to the success of BIM, this idea has become crucial in the context of BIM. The main cause of the lack of interoperability is that BIM software is proprietary. Many software programs are used in the construction sector with the intention of supporting building information modeling (BIM); yet, in reality, very little information is actually exchanged. Most of these software applications are designed to function as standalone systems and are not inherently built for seamless data sharing with other programs. Proprietary software solutions often create their unique data exchange formats specific to their domain, which confines users to sharing data within their chosen BIM softwareUsers might save a lot of time and improve stakeholder communication if they were free to share and exchange data via a shared platform with good interoperability. The yearly interoperability expenses for the U.S. capital facilities industry were projected to be $15.8 billion, according to a paper on the costs of poor interoperability published by the National Institute of Standards and Technology (NIST) [5].

   OpenBIM presents a potential solution to address these challenges. Essentially, OpenBIM involves working with BIM using open standards. BuildingSMART International [6] launched this project with the intention of advancing BIM interoperability on a worldwide scale. It is based on a process and workflow that, independent of the particular BIM tools and apps they use, allows all project stakeholders to work together and exchange project information using neutral, non-proprietary file formats. The goal of OpenBIM is to eradicate erroneous interpretations and promote error-free teamwork. The main goal is to collaborate with one another without bias. While openBIM uses standard file formats to transport data, communication between project members in a closed BIM environment is limited to formats that work with specific software (Fig. 1) [8]. This means that each participant's tool can be smoothly integrated into the workflow.

Industry Foundation Classes (IFC), an object-oriented standardized language and a shared data format, are the foundation of OpenBIM. The goal of IFC is to reach a high degree of interoperability to make data exchanges across BIM systems in the building sector easier. Building component semantics, including related geometry, characteristics, and relationships, are stored in IFC, an openBIM data repository. This encourages data reuse for analysis and other downstream processes by facilitating data sharing and exchange amongst IFC-compatible apps. Later on in this article, IFC will be covered in more detail.

To solve the interoperability issues in building projects, a lot of work has been put into developing information sharing standards during the last few years. [9] created an IFC-based integration tool to take an architectural model and extract a PKPM structural model from it. Also, the creation of an IFC and Web-based application allowed for a bi-directional conversion between different structural analysis models [10].

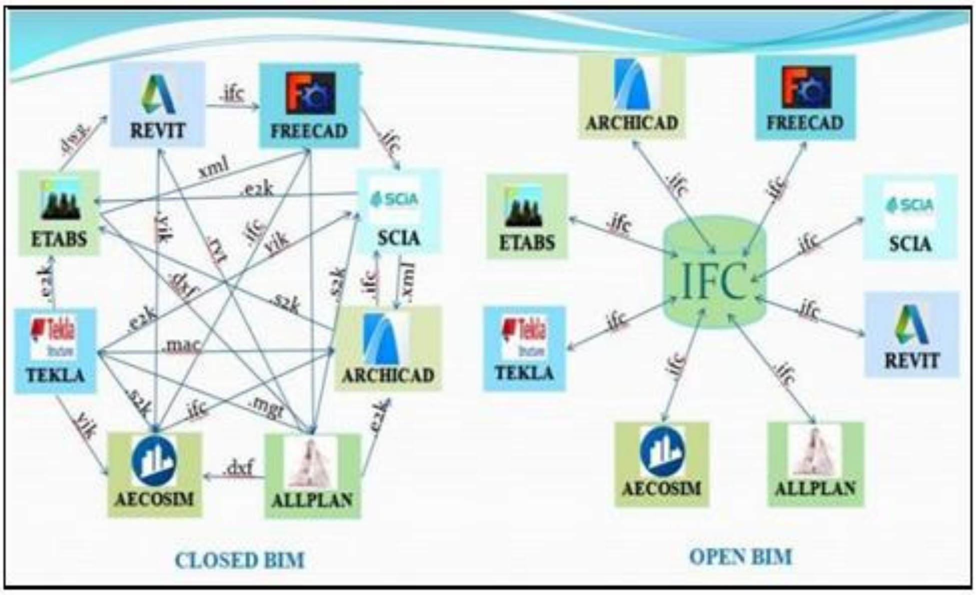


Figure 1: Closed BIM vs open BIM

2. **BIM in Structural Engineering:** One area where structural design and analysis is essential to BIM deployment is this one. Structural analysis and design are difficult jobs that take a lot of time and are prone to errors when done by hand. BIM provides structural engineers with a number of benefits. For structural analysis purposes, a significant quantity of input data can be immediately taken from the BIM physical model, saving a significant amount of time compared to building this model from drawings. In a BIM-based design process, this method facilitates automation and improves accuracy [11].

High interoperability between BIM authoring tools and engineering design/analysis software is necessary to properly utilize BIM for structural design objectives. The option to export structural analytical models in non-proprietary file formats is currently absent from BIM authoring tools. For example, Tekla can interface with software such as SAP2000 and STAAD, and Autodesk Revit can export structural analysis models to the file format of popular structural tool Robot Structural Analysis. When producing BIM data for use in downstream applications, the derivation of a structural model is an essential step [13].

In this study, data needed to create a building's structural model was taken from the architectural BIM model using open-source tools. We show that openBIM is much more than just standardized file formats by implementing an integrated process. The widely used Python programming language and the open-source IfcOpenShell package are the main instruments utilized in this project. This work's primary goal is to encourage the usage of open-source technology.

II. MATERIALS AND METHODS

The research technique includes three steps: (1) preparing the BIM model; (2) converting the model into IFC file format; and (3) determining exchange requirements in order to transfer BIM information for structural purposes while preserving the significance of various pieces of information in the extraction. (4) creation of the ifc file parsing framework (5) use of the parsing tool.

First, a building model is created using a BIM application, and the model is exported file in the.ifc format. To extract the target information from the architectural model that is pertinent to the structural model based on the exported IFC model, a Python technique that integrates with the ifcopenshell package was developed. The code takes an IFC file as input. By adding the ifcopenshell [14] to the IFC Parser software, it was possible to read IFC files. The indirect method's process for data transformation between the structural analysis model and the BIM model is depicted in Fig. 2.

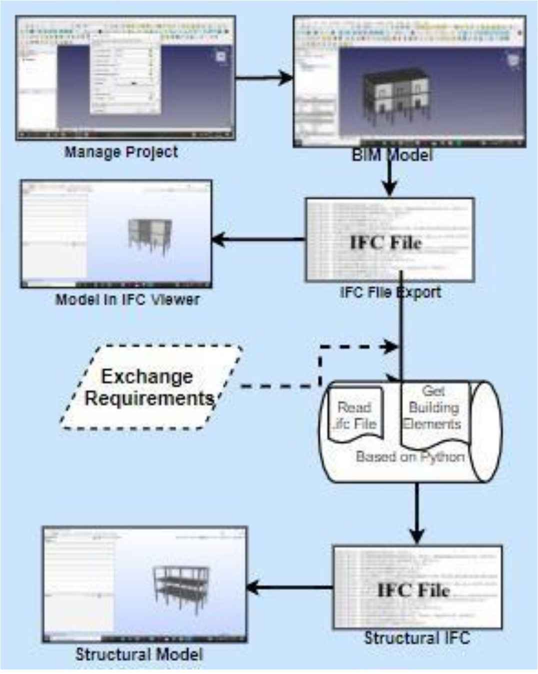


Figure 2: BIM information extraction framework

1. **Building Model Preparation:** The main requirement for the system described here is a legitimate, well-organized BIM model that includes geometry and material information. All of the necessary data must be included in the architectural model in order for the structural model to be automatically generated.

   FreeCAD, a flexible parametric 3D computer-aided design (CAD) modeling application that satisfies all requirements to function as an open-source platform, was used to create a 3D BIM-based model. FreeCAD is very expandable, scriptable, and customisable. It makes it possible to add new features and open-source libraries without changing the foundational architecture of the system. The program has an integrated Python interpreter, which enables the use of Python code within the program [15]. It is the only modeling program available for free that is comparable to the commercial industry's current offerings, like Revit or ArchiCAD.

   It offers far superior IFC functionality, and in FreeCAD, IFC files are given first-class treatment. [18] shown how semantically driven high-level spatial restrictions can directly impact exact parametric models that combine mobility, visibility, and topological constraints using FreeCAD. [12] developed BIM plug-ins to be able to import and edit digital representations of cultural heritage models created by using FreeCAD, an open source computer-aided design (CAD) system photosynthetic techniques.

   The prototype for this study was created using the BIM workbench of FreeCAD, as demonstrated in Figure 3 of the modeling process. Through FreeCAD's add-on manager, this workbench—an external tool—can be integrated. Setting important parameters, like grid spacing, the working plane, and structural level management, was the first phase in the modeling process.

The BIM architectural model started to take shape as the basic modeling parameters were set. The placement of beams and columns on the designated grids marked the beginning of this phase. The modeling procedure was organized along a predetermined path. First, the project specifics, grid measurements, and structural levels were managed. Next, columns, beams, walls, doors, windows, and slabs were created.

To utilize the architectural model as a reference for generating the BIM structural model, each structural element was individually modeled, and the structural properties and behavior of these elements were incorporated. All building elements were modeled as distinct objects, each of which was then assigned an IFC type corresponding to the appropriate categories and subcategories. For instance, beams were assigned the type "ifcbeam."

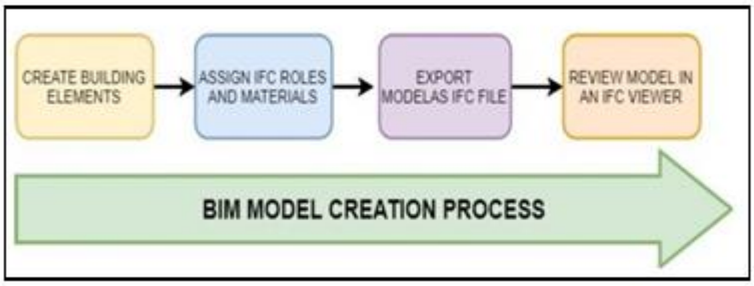

CREATE BUILDING ELEMENTS
ASSIGN IFC ROLES AND MATERIALS
EXPORT MODELAS IFC FILE
REVIEW MODEL IN AN IFC VIEWER
BIM MODEL CREATION PROCESS

Figure 3: BIM model preparation

The beam does not have all the properties that are necessary for a good description if it is not given to the entity Ifcbeam. This implies that subsequent coordinating programs won't understand it appropriately. After then, materials were allocated to these components. For example, concrete was used for beams, columns, and slabs, and timber materials were used for doors and windows. The data required to automate the transformation process and guarantee that remodeling is not necessary while studying the structural behavior in structural analysis software applications was added to the architectural model. Figure 4 shows the generated BIM model.

2. **Conversion of BIM Model into IFC File Format:** The architectural model is converted into the IFC file format using the 'Export' function within FreeCAD. This transformation involves translating the model information into a universally understood language that can be interpreted by the structural software applications it is intended for. To ensure that all relevant information is included in the exported IFC model, the model can be subjected to a thorough review using an IFC model viewer.

For the purpose of this study, the IFC model was reviewed using the open-source Solibri Model Viewer (SMV), which is designed to be used in BIM projects. It is possible to open and view standard IFC files using this software, as shown in Figure 5. The main source of data will be the IFC file that is produced by this procedure. It's crucial to remember that IFC models provide a semantic, object-oriented data framework for the interchange and storing of building data. IFC files are based on a schema that lists the entities and their relationships inside the file. The ISO standard framework for storing BIM model data is called IFC. IFC is an open, vendor-neutral BIM platform that buildingSMART developed and maintains. It focuses on semantic information for building objects, including geometry, related characteristics, and relationships. Encouraging interoperability between various software systems is IFC's main objective. The EXPRESS language, a model description language outlined in ISO 10303-11, is used to describe IFC.

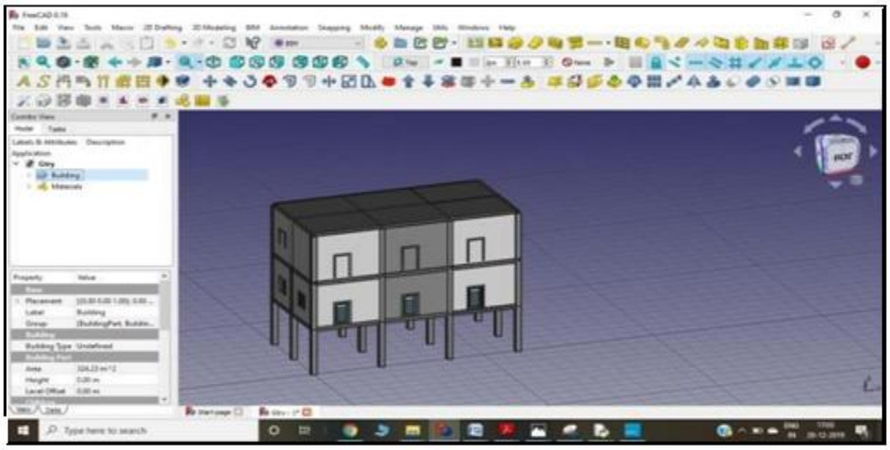


Figure 4: BIM Model in FreeCAD

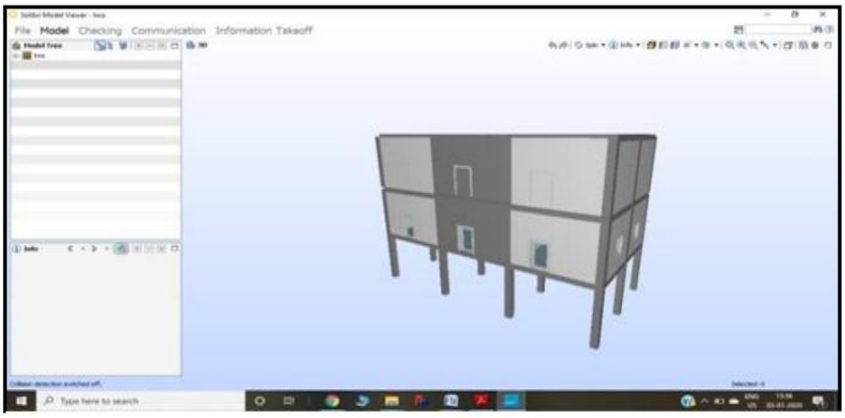


Figure 5: Model review in Solibri model viewer

As a result, it inherits EXPRESS's object-oriented programming concept [17].

Additionally, the IFC data schema offers a framework for information sharing between different sectors of the construction industry as well as a fundamental modular structure for the information model. It specifies an interchange file format for BIM model data as well as an abstract data schema.

Exchange Requirements for Structural Model

Though precise information about several disciplines can be stored in architecture models, structural engineers are only interested in a portion of the information produced by architects. The structural model necessitates only data related to structural analysis, such as the global position and geometric representation, the structural function of building elements, and the types and qualities of materials. During the conversion process between these two types of models, information from the architectural model that is connected to the structural model is identified, categorized, processed, and changed to the structural model.

To ensure the precision of data exchange specifically for structural purposes within collaborative design, it became essential to identify the Exchange Requirements (ERs) to enhance the dependability of structural design information. This phase is focused on recognizing the ERs relevant to the structural model. An ER constitutes a collection of data that must be shared to fulfill domain-specific necessities for a particular project phase. To facilitate a successful conversion, it's crucial that these ERs encompass the semantically vital information for each object within the exchange. The subset of constituent IFC entities needed for a given exchange and their corresponding definitions should be explicitly outlined.

Therefore, in an IFC-based BIM model intended to generate a structural model, the important data consists of the global identifier, the number of IfcObject instances, and the characteristics associated with these IfcObject instances. This project has found objects like IfcBeam and IfcColumn that are directly relevant to the creation of a structural model. Additionally, sub-elements like IfcProject, IfcRoot, and IfcElement—which offer thorough object definitions and are essential for defining interactions between elements and systems—have also been defined. As a self-defined data structure, this target information has been described in an information dictionary.

3. **Framework for Parsing IFC File:** The Express language, which is used to define the exported IFC model, is a modeling language rather than a programming language. Thus, in order to facilitate the creation of the structural model, data regarding building items and their attributes in the IFC model must be transferred to a computer language. The Python language is used in this investigation for this reason. An IFC Parser program, mostly written in Python, has been developed to extract BIM objects straight from an exported IFC model, hence facilitating the extraction of information important to the structural model from the architectural model. Python is unable to read IFC files. By adding the Ifcopenshell to the IFC Parser software, it was possible to read IFC files [14]. In theory, the FreeCAD model might be visually represented by opening it with IfcOpenshell within the application.

Python is a fantastic programming language that is object-oriented, versatile, open source, and simple to learn. Since Python is a progressively built language, its variables are automatically defined. Maintaining a clear code base and an application that is easy to use is further facilitated by the programming language's basic syntax principles. Python also works well when combined with other programming languages. Python has a vast and robust standard library that sets it apart from other programming languages. You can select from a variety of modules in the standard library based on your needs. Data may be visualized and presented more effectively and aesthetically thanks to the tools and APIs for data visualization that the language offers. Python is the scripting language used by FreeCAD, which gives it great flexibility when modeling parts. Python makes it possible for users without any programming experience to begin creating programs. Code written in a language like Python, which provides methods to extract entities from IFC instance models using the entity and attribute relations described in the IFC schema [18], is an effective tool for collecting target information.

An open-source software library called IfcOpenShell was created to help users and programmers deal with the IFC file format, which is frequently used in building information modeling (BIM). A Python module that makes it easier to parse IFC files is part of this project. Using this Python library, IfcOpenShell is able to precisely capture the location of IFC files as specified by the Ifc Schema through querying and parsing. The support of this library is extended to all IFC2x3 and IFC4 entities.

To use IfcOpenShell in Python, it is typically installed in Python's library under Site packages and can be imported into the Python interpreter using the "Import Ifcopenshell" command. Notably, IfcOpenShell contains numerous libraries and modules, and importing all of them can lead to slower performance. To address this issue, the "Import Ifcopenshell" command allows the utilization of libraries related to IFC exclusively, reducing memory usage.

It's important to note that IFC files cannot be natively read by Python; they require the utilization of ifcopenshell for access. An IFC file containing an architectural model can be imported into Python in two distinct ways.

- Import File = ifcopenshell.open(r”path\of\file”)
- Import File = ifc.open (r”path\of\file”)

The tool developed for this project operates based on the IFC schema and the inherent data structure of the IFC-based BIM model to systematically extract IFC data instances through their internal relationships with other instances. Therefore, it necessitates a comprehensive understanding of the IFC schema's structure.

In essence, an IFC model is made up of IFC objects arranged in a tree structure. Every item in the IFC schema, called an IFC entity, is a part of a hierarchy that organizes the items in the building model. The three main parts of the schema are properties, attributes, and entities at its most basic level.

In the IFC data model, an IFC entity is a uniquely specified item that has the ability to exist on its own. Consider an entity as something similar to what is meant by a "class" in most computer programming languages. These entities contain data about many facets of a structure's life cycle, such as geometric features like IfcDepth, building elements like IfcBeam, and basic constructs like IfcDirection.

The attribute is a pivotal concept within the IFC schema. It essentially refers to the properties associated with an entity, which can be either required or optional as stipulated by the IFC Model. For instance, Fig. 6 illustrates numerous attributes linked to the entity IfcBeam. It is essential to have a basic understanding of these characteristics for this investigation.

In the IFC schema, all entities are related with properties like name, materials, shape, size, connections, and more. In essence, entities are object-oriented components.

One component of IfcBeam, for instance, is "Object Placement," which describes where the beam is located inside the model.

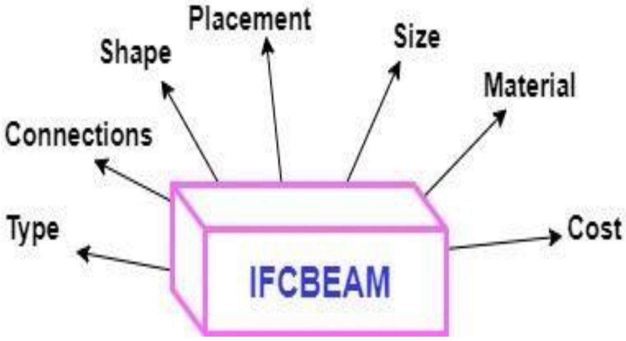


Figure 6: Definition of attributes of a Beam

An entity is made up of both direct and indirect characteristics. An attribute that an entity inherits from its supertype is known as an indirect attribute. IfcBeam, for example, receives "Name," an indirect attribute, from IfcRoot, its supertype.

These attributes form the basis for specifying a building component entity's geometric properties in an IFC exchange file. The Project entity is at the top of this structure, and as such, "IfcProject" serves as the general repository for all the data in an IFC exchange file.

It's important to note that "IfcProject" itself is not a visible entity.

"IfcProject" is linked to characteristics like GlobalID, OwnerHistory, Name, and Description even though it is not an element that can be seen in the schema. Usually, a building's hierarchical structure contains just one "IfcProject." This "IfcProject" might include zero or more sites, and each site might include one or more structures, each of which could have one or more stories.

See Figure 7 to gain a better understanding of the IFC concepts of entities and attributes. "IfcRoot," the supertype of all entities in the IFC model, represents the highest level of abstraction. Entities classified as abstract can only exist as a subtype of another entity. The main abstract entities are shown in Figure 7 and are indicated in green.

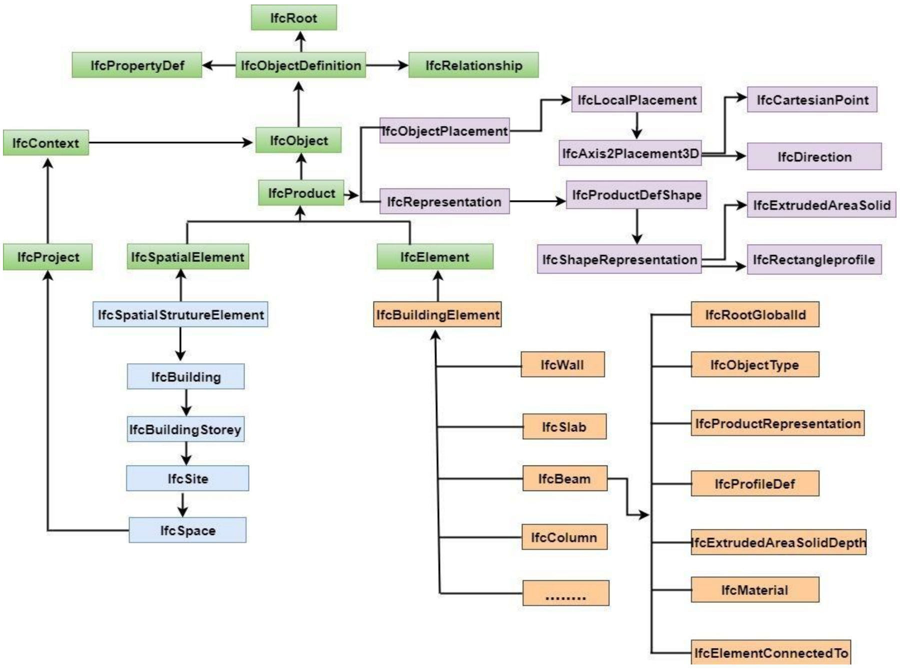


Figure 7: Project hierarchy in IFC Schema

IfcRoot is the parent of three important subtypes that are fundamental to the IFC Model: IfcPropertyDef, IfcObjectDef, and IfcRelationship. IfcRelationship describes the relationships between different objects, IfcObjectDef is a generalization that includes all physical objects or processes, and IfcPropertyDef includes any attributes that can be connected with entities.

Derived from IfcObjectDef, IfcProduct is the entity that defines common type information across all building products. Within the IfcProduct category, you have IfcRepresentation and IfcObjectPlacement. These usually contain geometric representations and the spatial arrangement of building elements. Subtypes of IfcObjectPlacement, such as IfcAxis2Placement2D, IfcAxis2Placement3D, and IfcLocalPlacement, define the precise spatial positioning. In Figure 7, these entities and their subclasses are shown in purple.

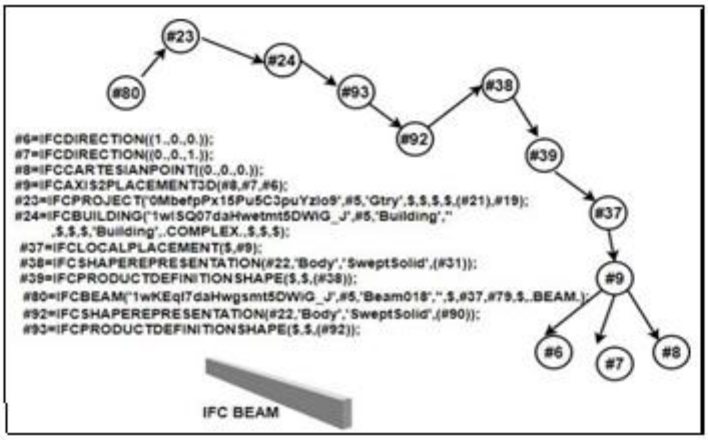
The IfcLocalPlacement object provides the ability to ascertain the precise location of any IfcProduct instance. IfcElement and IfcSpatialElement are both members of IfcProduct in the IFC instance model. As a result, gathering product knowledge is crucial before digging into the specifics of each part. All properties are inherited from the entities at the supertype level, thus IfcBuildingElement is a subtype of IfcElement and shares the same attributes as IfcElement.

The entity IfcBuildingElement is the primary functional component of a building that's designed to model physical information within the IFC standard. This category encompasses various building elements, such as beams, walls, columns, slabs, doors, windows, roofs, and more. In Figure 7, you can see this entity and its subclasses represented in orange.

To provide the IFC spatial structure, the IfcSpatialStructureElement entity is employed. IfcSpatialStructureElement serves as the parent entity for four important sub-entities: IfcBuilding, IfcBuildingStory, IfcSpace, and IfcSite, all depicted in blue in Figure 7. These entities play a crucial role in organizing building elements within the IFC Model.

When it comes to the exchange of project data, IfcProject, IfcBuilding, and IfcBuildingStorey are mandatory levels, whereas IfcSite and IfcSpace represent optional levels that can be transferred (provided they contain the necessary data). It's important to note that Figure 7 serves as an abstraction of the IFC schema, illustrating the inheritance of key entities and some of their attributes. The diagram's purpose is to simplify the understanding of the complex IFC schema as much as possible.

4. **Execution of Parsing Tool:** The extraction process initiates by querying the IfcProject instance within the input IFC file. Subsequently, it proceeds to locate IfcSite, IfcBuilding, IfcBuildingStorey, and IfcBeam entities that are encompassed within the building stories and beyond. As depicted in Figure 8, the parsing tool follows a hierarchical approach, starting with the identification of the highest-level IfcProject entity. It then traverses down the structure, extracting IfcSite, IfcBuilding, IfcBuildingStorey, IfcSpace entities, and so forth. The outcome of this work entails the extraction of structural building elements, including reinforced concrete columns, beams, slabs, along with non-physical IFC entities linked to the selected building elements.



```
#6=IFCDIRECTION((1.,0.,0.));
#7=IFCDIRECTION((0.,0.,1.));
#8=IFCCARTESIANPOINT((0.,0.,0.));
#9=IFCAXIS2PLACEMENT3D(#8,#7,#6);
#23=IFCPROJECT('0MbefpPx15Pu5C3puYzlo9',#5,'Gtry',$,$,$,$,(#21),#19);
#24=IFCBUILDING('1wISQ07daHwetmt5DWIG_J',#5,'Building','',
      ,$,$,$,'Building',.COMPLEX.,$,$,$);
#37=IFCLOCALPLACEMENT($,#9);
#38=IFCSHAPEREPRESENTATION(#22,'Body','SweptSolid',(#31));
#39=IFCPRODUCTDEFINITIONSHAPE($,$,(#38));
#80=IFCBEAM('1wKEqI7daHwgsmt5DWIG_J',#5,'Beam018','',$,#37,#79,$,.BEAM.);
#92=IFCSHAPEREPRESENTATION(#22,'Body','SweptSolid',(#90));
#93=IFCPRODUCTDEFINITIONSHAPE($,$,(#92));
```

Figure 7: Entity relationships for IFCBEAM

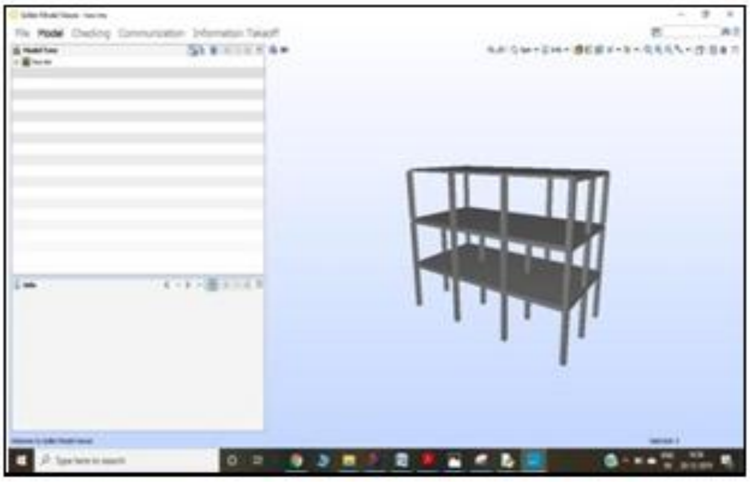


Figure 8: Extracted Structural Model

When the IFC Parser tool is run, an.ifc file is produced, enabling a variety of BIM applications to use the output file. The structural model was imported into Solibri Model Viewer from the IFC file following its creation. In Fig. 8, this structural model is displayed. The information model needed for structural analysis and design will be given to the structural engineers as a result.

III. CONCLUSION

Undoubtedly, BIM represents a revolutionary concept, yet realizing its full potential and keeping abreast of advancing technology necessitates addressing the ongoing research challenge of BIM interoperability and data exchange. Currently, many building models are confined to proprietary formats, limiting their use to the specific BIM authoring tool they were created in. The study presents a framework to produce a building's structural model from its architectural building information model as a solution to this problem. This framework uses Python and the IfcOpenShell open-source library to create an IFC parser tool, utilizing free and open-source applications. The suggested method automates the process of obtaining the geometry and material information needed to build a structural model from an IFC-based model. This article outlines the essential elements of the integrated workflow and explains why openBIM is more than just a file format standard. The field of building information modeling applications' overall interoperability is expected to be greatly impacted by this research.

REFERENCES

[1] Dubois, A., & Gadde, L. E. (2002). The construction industry as a loosely coupled system: implications for productivity and innovation. Construction management & economics, vol. 20,no.7, 621-631, DOI: 10.1080/01446190210163543.

[2] National Institute of Building Sciences. National BIM Standard (NBIMS), from http://nnbimsdoc.opengeospatial.org, accessed on 2019-11-12.

[3] Eastman, C., Teicholz, P., Sacks, R., & Liston, K. (2011). BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors. John Wiley& Sons, Inc., Hoboken, New Jersey.

[4] Lee, G., Sacks, R., & Eastman, C. M. (2006). Specifying parametric building object behavior (BOB) for a building information modeling system. Automation in construction, vol. 15, no.6, 758-776,

DOI:10.1016/j.autcon.2005.09.009.

[5] Gallaher, M. P., O‟Connor, A. C., Dettbarn, J., & Gilday, L. (2004). Cost analysis of inadequate interoperability in the US capital facilities industry (NIST GCR 04-867). *RetrievedNovember*, *8*, 2012.

[6] Building SMART. Summary of IFC Releases, from http:// www.buildingsmart-tech.org/specifications/ifc-releases/ interoperability-openbim-ifc, accessed on 2019- 02-10.

[7] Rahimian, F. P., Chavdarova, V., Oliver, S., & Chamo, F. (2019). OpenBIM-Tango integratedvirtual showroom for offsite manufactured production of self-build housing. Automation in construction, vol. 102, 1-16, 10.1016/j.autcon.2019.02.009.

[8] Laakso, M., & Kiviniemi, A. O. (2012). The IFC standard: A review of history, development, and standardization, information technology. ITcon, vol. 17, 134-161, itcon.org/papers/2012_9.content.01913.pdf.

[9] Liu, Z. Q., Li, Y. G., & Zhang, H. Y. (2010). IFC-based integration tool for supporting information exchange from architectural model to structural model. Journal of Central South University of Technology, vol. 17, no. 6, 1344-1350, DOI: 10.1007/s11771- 010-0640-z.

[10] Zhang, X. Y., Hu, Z. Z., Wang, H. W., & Kassem, M. (2014). An Industry Foundation Classes (IFC) web-based approach and Platform for Bi-directional conversion of structural analysis models. Computing in Civil and Building Engineering, 390-397, 10.1061/9780784413616.049.

[11] Ramaji, I. J., & Memari, A. M. (2018). Interpretation of structural analytical models from the coordination view in building information models. Automation in Construction, vol. 90, 117- 133, DOI: 10.1016/j.autcon.2018.02.025.

[12] Logothetis, S., Karachaliou, E., & Stylianidis, E. (2017). From OSS CAD to BIM for cultural heritage digital representation. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XLII-2/W3, 439–445, DOI: 10.5194/isprs-archives-XLII-2-W3-439-2017.

[13] Sacks, R., Koskela, L., Dave, B. A., & Owen, R. (2010). Interaction of lean and building information modelling in construction. Journal of construction engineering and management, vol. 136, no. 9, 968-980, DOI: 10.1061/ASCECO.1943- 7862.0000203.

[14] IfcOpenShell. IfcOpenShell – the open source ifc toolkit and geometry engine. From http://ifcopenshell.org/bimserver.html, accessed on 2019-01-06.

[15] Riegel, J, Mayer, W, & van Havre, Y. (2016). FreeCAD, from http://www.Freecadweb.org.Version:0.19, accessed on 2018-12-06.

[16] Schultz, C., Bhatt, M., & Borrmann, A. (2017). Bridging qualitative spatial constraints and feature-based parametric modelling: Expressing visibility and movement constraints. Advanced Engineering Informatics, vol. 31, 2-17, DOI: 10.1016/j.aei.2015.10.004.

[17] Borrmann, A., Beetz, J., Koch, C., Liebich T. (2018). Industry foundation classes: A standardized data model for the vendor-neutral exchange of digital building models. Building Information Modeling, Springer International Publishing AG, 81–126.

[18] Mazairac, W., & Beetz, J. (2013). BIMQL–An open query language for building information models. Advanced Engineering Informatics, vol. 27 no. 4, 444-456, DOI: 10.1016/j.aei.2013.06.001.

[19] Zhang, C., Beetz, J., & Weise, M. (2015). Interoperable validation for IFC building models using open standards. Journal of Information Technology in Construction (ITcon), 20(2), 24- 39, from http://www.itcon.org/2015/2.

[20] Atazadeh, B., Kalantari, M., & Rajabifard, A. (2016). Comparing three types of BIM-Based Models for managing 3D ownership interests in multi-level buildings. 5th International FIG 3D Cadastre Workshop, 18-20 October 2016, Athens, Greece, 183- 198.