American Journal of Engineering Research (AJER)
e-ISSN: 2320-0847 p-ISSN : 2320-0936
Volume-5, Issue-4, pp-91-98
www.ajer.org
Research Paper Open Access

Using Bounding Volumes for BIM based electronic code checking for Buildings in Egypt

Mohamed Magdy Abdelaziz Nour[1]
[1](Architecture Department/ Faculty of Fine Arts, Helwan University, Egypt)

ABSTRACT *: The building and construction industry in Egypt suffers a lot from the process of issuing building permits. It is still paper based. Designs are checked manually against frequently changing and complex sets of building legislation acts. In Egypt, this is a particularly cumbersome, tedious and iterative process that often leads to misunderstandings, legal escalations, inconsistency in assessments and time delays.*

This paper focuses on the technical part of testing the conformance of buildings against rules and legislations. In order to conduct electronic checks for buildings, both buildings and buildings' legislations have to be represented digitally. This type of check is only possible, when the bilateral digital representations of both, buildings and codes / legislations can be coupled together. For buildings, the availability of Building Information Models (BIMs) and its open standard -Industry Foundation Classes (IFC)- opens a new window of opportunity for representing buildings digitally. However, the mapping of "legislation-acts" into a digital format that can be used for checking buildings' conformance is an awkward process that has not been established yet. It is envisaged that "legislation-acts" can be mapped to BIM based semantic filters to recognize building objects on one hand and to construct geometrical bounding volumes – representing geometrical and spatial constraints on the other hand.

Keywords – BIM*, Code Checking, Electronic Building Permits, IFC.*

I. INTRODUCTION

The building and construction industry is the largest industry in Egypt. It suffers a lot from the process of issuing building and demolition permits. Building permits in Egypt is still paper based. Designs are checked manually against frequently changing and complex sets of building regulations. In Egypt, this is a particularly cumbersome, tedious and iterative process that often leads to misunderstandings, legal escalations, inconsistency in assessments and time delays. Furthermore, citizens seeking building permits have to deal with several Governmental isolated islands of bureaucracy.

This research aims at using the latest cutting edge BIM technology to automate the permitting process. The emerging BIM technologies open new windows of opportunities to automating the checking of compliance with building codes and legislations. It is envisaged that if a web service ("e-permit") based on a process model that unifies all contact points with citizens into a single entity; instead of multiple Governmental authorities that have to give their consent individually, this would lead to an automated online e-permitting system. This system is capable of determining the degree of conformance to building regulations and codes, reporting non-conformance issues and tracking the status of the permit request in a transparent manner that isolates the employee in charge of issuing the permit from the person requiring the permit. Furthermore, it eliminates the inconsistencies associated with same legislative rules being interpreted differently by different individuals (employees).

This paper reports on on-going research work and focuses on the technical part of testing the conformance of buildings' designs to rules and legislations. In order to conduct an electronic check for buildings' designs, both buildings' designs and buildings' legislations have to be represented digitally. This type of check is only possible, when the bilateral digital representations of buildings and codes / legislations can be coupled together. For buildings, the availability of Building Information Models (BIM) enables the digital representation of buildings. However, the mapping of the different and often changing "legislation acts" into a parametric digital format that can be used for checking buildings' conformance is an awkward process that has not been established yet. It is envisaged that legislation acts can be mapped to BIM based semantic filters to recognize objects' types on one hand and to generate geometrical bounding volumes – representing geometrical and spatial constraints – for geometrical conformance testing on the other hand.

In the meantime, all over the world, there are two main categories of building codes. These are the performance codes and the prescriptive codes. The performance codes aim to assure the availability of a minimum level of performance levels of the building towards certain criteria and measures, regardless how these performance measures are achieved. Whereas, the descriptive codes enforce certain aspects (mostly minimum and maximum dimensions, number of elements, capacities, types and specifications of materials, etc…) that have to be met in order to be able to achieve conformance with rules, codes or legislations. The performance codes are out of the scope of this paper and the main focus is on the prescriptive type of codes.

Building Information Modeling (BIM) is a modern technology that has risen in conjunction with the Object Oriented Modeling (OOM) paradigm and particularly in conjunction with the developments achieved within the Object Oriented Programming (OOP) domain. In general, models are created for testing and examination. Furthermore, they enable a better understanding of existing or future systems. In the meantime, a model never fully corresponds to reality. There is always a modeling aim behind any model, where the modeling process works on emphasizing certain relevant features and omitting irrelevant ones in a process called "Abstraction". In the BIM technology, real life objects like walls, beams, columns and so forth are modeled by abstraction, where features relevant to the Architectural, Engineering, Construction and Facilities Management (AEC-FM) domains are captured. A digital example of such models is the IFC model ISO standard [1]. It is modeled using the EXPRESS (ISO10303 -11) [2] object oriented modeling language, where models can be represented in both textual and graphical formats. The EXPRESS schema consists of several entities. Each entity corresponds to a class of objects. The instantiated objects are exchanged between software applications in the form of an IFC STEP (ISO10303-21) [3] file. The major advantage of the IFC/BIM model over its predecessor CAD (Computer Aided Drafting) technologies is that objects are semantically rich and include attributes and property sets that are defined by the buildingSMART [4] initiative and its chapters all over the world. The entities of the IFC model include not only the semantic features of the modeled object, but also include the geometrical representations and topological relationships of the objects. The geometrical modeling takes place according to the ISO10303-42 standard [5].

It is not enough to have an information model of the building. It is also pertinent to encode the building legislations' rules and regulations into a digital format. This format should be user-updatable (parametric); that it can cope with any updates. These updates can be necessary to account for the application of such rules in different geographical regions (cities, quarters, streets, etc.) with variance of parameters. For example, in a certain city, the maximum height of buildings can be 3 floors, whereas in another city the maximum height might be 6 floors. In the meantime, it can be dependent on the width of the street where the plot of land is located, … and so forth. Hence, there is a need to define a generic rule algorithm that takes input parameters from legislative rules. It is also envisaged that the appropriate building legislations for each plot of land should be anchored to it on a national GIS system.

There are several commercial software packages that are capable of conducting checks and examining the BIM model against certain pre-defined rules and codes. Among the examples of such software tools are EDM Model Checker [6], Solibri Model Checker (SMC)[7], Navisworks [8] , Tekla Structures [9], and VICO Software [10]…etc . However, the main applications used in this domain are the EDM and SMC.

The EXPRESS Data Manager (EDM) [6] is a powerful object oriented EXPRESS based tool box that is capable of dealing with any model that is based on an EXPRESS schema. It has a full range of functionalities starting from checking any EXPRESS schema for correctness, creating new schemata, reading data that belongs to any EXPRESS schema in the form of STEP-21 files (provided that the schema itself is available), querying data models, and finally checking against rules. For this purpose, rules are declared in the same way as they are declared in the EXPRESS language. Furthermore, the EDM has developed the EDMExpressX schema definition that is a superset of EXPRESS ISO 10303-11. It includes extra statements needed to define an EDMruleSchema. This schema is then compiler using the EDMExpressX compiler. Moreover, the EDM possess APIs (Application Programming Interfaces) for function calls by major programming languages such as Java, C++ and Visual Basic. It is quite complicated, needs a lot of programming, and modeling knowledge by the end user to be able to formulate the rules. Moreover, it is not possible to conduct any rule check without the ultimate good knowledge of the data's underlying EXPRESS schema. All rules are checked using the semantics of the underlying schemata as well as the defined rules (also in EXPRESS). A major drawback is that it does not support pure geometrical checking of objects.

On the contrary, Solibri Model Checker has BIM model viewing capabilities, and hence can deal with the geometrical representation and the topology of objects for rule checking as shown in Fig. 1 and Fig.2. Its main strength lies in its ability to check models for completeness and consistency. In the meantime, it possesses a set of predefined rules that can be replicated by the end user. However, the extent of user customization is limited to changing parameters' values. [11].

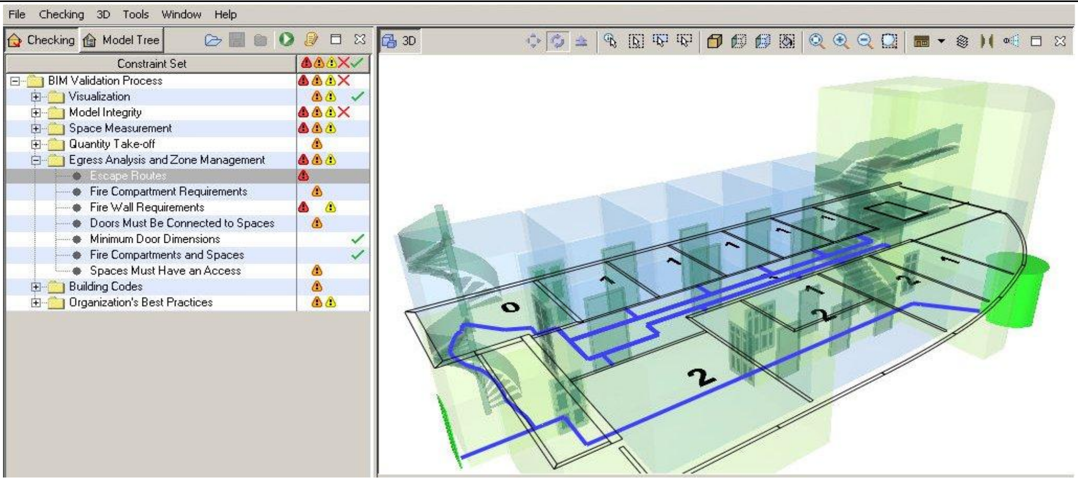

Figure 1: Solibri Model Checker, checking escape roots in a building. **Source: Solibri**

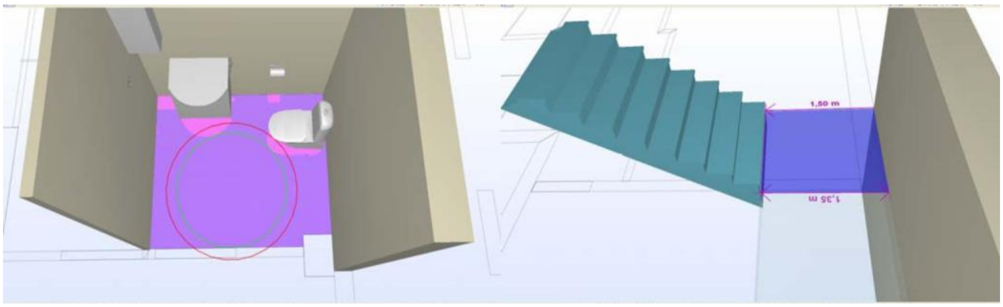

Figure 2: Solibri Model Checker, checking for accessibility and staircase dimensions **Source: Solibri**

The majority of such tools - with the exception of EDM - have hard coded rules that can be used for checking models with very little opportunity for the end user to define his own rules. Furthermore, they are proprietary black box technologies that do not show their checking algorithms or internal ways of working. Even, with the EDM, writing a rule schema proves to be a difficult and awkward process.

In the meantime, there exists research work and national governmental initiatives all over the world that address this problem. The main research work is directed towards the mapping of legislative rules into a computerized digital format that can be used for code checking. Example of such research is the SMARTCodes [12] initiative declared by the International Code Council (ICC) in the USA in 2006. This research effort has been extended by a mark-up approach later in the UK [13]. Both research efforts have been targeted towards providing the end user with an interface that can update the legislations' rules. Finally, in 2012 ICC, Solibri and Fiatech cooperated with other software vendors to develop AUTOCodes [14] in a trial to produce digital building model codes for the US. Furthermore, modern informatics' scope of application has been extended to include new fields relative to the digitalization of legislative rules such as: "legal informatics", "legislation modeling" and "digital representation of regulations" [15]. On the other hand, among the successful national governmental applied initiatives is the CORENET project in Singapore [16].

The CORNET (Construction and Real Estate Network) initiative -by Singapore's ministry of national development- proves to be one of the most successful efforts in this domain. It started in 1995 with the aim of achieving interoperability between all stakeholders involved in the construction and real estate sector in Singapore. The latter is based on an IT infrastructure that serves the industry stakeholders. The project is implemented by Singapore's Building and Construction Authority together with other organizations. It supports the main stages of the buildings' life cycle including design, procurement, building and maintenance. It started with the development of three main modules for the design phase: 1) e-Submission, 2) e-PlanCheck and 3) e-Info. The e-Submission module encompasses all relevant stakeholders of the permitting process. This includes the Urban, Building and Construction, Electricity and Power, Transport, Housing and Development authorities … etc. for digital document submission instead of paper workflows. Whereas, the e-PlanCheck started first in the nineties by making use of AI (Artificial Intelligence) and FB (Featured Based) CAD technologies by checking the compliance of 2D plans. In 2002, the system shifted to the BIM IFC model checking and abandoned the old 2D plans. Now, it uses the FORNAX [17] platform together with EDM, where the legislation rules are mapped to EXPRESS. The FORNAX is an independent platform developed by NovaCITYNETS Pte. Ltd, which is an e-Government solution provider in Singapore. It provides high level semantics to the objects of the IFC model. It wraps IFC objects by semantically rich FORNAX objects that enable the checking process. FORNAX has also been used as a pilot implementation in Norway and in New York city. Furthermore, it has been used for testing Japanese and Australian models.

From the above, it can be seen that there is no one technology that has established itself in this domain and that it is subject for every nation to find its way in defining its legal acts and rules in a digital format that can be coupled with the IFC/ BIM model to enable building designs' permit conformance checking. Thus, the coming sections include a trial to capture basic Egyptian building permits legislative rules for residential buildings in new settlements into a digital format. On the other hand, the IFC model standard is used as the digital model for the buildings designs. It provides the semantic base for object type recognition and querying the model for needed information.

II. METHODOLOGY

To be able to conduct rule checking on IFC models, buildings design models have to undergo several operations such as:

1) Parsing and interpretation of the IFC model.
2) Mapping the model into a database.
3) Viewing the model in 2D and 3D geometrical forms.

On the other hand the legislative rules have to be mapped into a digital format that can be applied on the IFC model for conformance testing.

In order to achieve the above mentioned tasks, the author developed an IFC Java toolbox that is capable of carrying out the above mentioned functionalities with no need to the EDM database. IFC Java runtime objects representing the IFC model are obtained. This is done through an early binding operation with the IFC EXPRESS model schema to the Java language. The result is an IFC Java library of classes that corresponds to the IFC EXPRESS model schema. Table 1 shows the mapping of the main data types from the EXPRESS modeling language to the STEP-21 file format to the Java Early binding model.

Table 1: Mapping EXPRESS data types to STEP-21 and finally to Java

EXPRESS element	mapped in to STEP-P21:	Mapped into Java
ARRAY	List	List
BAG	List	List
BOOLEAN	Boolean	Boolean
CONSTANT	NO INSTANTIATION	NO INSTANTIATION
DERIVED ATTRIBUTE	NO INSTANTIATION	NO INSTANTIATION
ENTITY	Entity Instance	Class
ENTITY attribute	Entity Instance Name (ID)	Reference to Object
INVERSE Attribute	NO INSTANTIATION	NO INSTANTIATION
LIST	List	List
LOGICAL	enumeration	Class
NUMBER	real	double
PROCEDURE	NO INSTANTIATION	NO INSTANTIATION
REAL	REAL	Double
REMARKS	NO Inst.	NO Inst.
RULE	NO INSTANTIATION	NO INSTANTIATION
SCHEMA	NO INSTANTIATION	Package (early binding)
SELECT	As an entity	Class (early binding)
SET	list	Set
STRING	String	String
TYPE	As an entity	Class (early binding)
UNIQUE rules	NO INSTANTIATION	NO INSTANTIATION
WHERE RULES	NO INSTANTIATION	NO INSTANTIATION

The IFC tool box then parses STEP-21 files generated by BIM compatible software like ArchiCAD, Revit, Micro Station … etc. As a result, IFC Java runtime objects with the same names and attributes are generated. The generated model replicates 100% the directed graph structure of the IFC EXPRESS model. It also includes the inverse attributes of the IFC EXPRESS entities as well as metadata about the optional attributes as shown in Fig. 3 to the left. Furthermore, the IFC Java classes are extended through inheritance to include extra implementation functionalities that are needed for applications using the model, such as

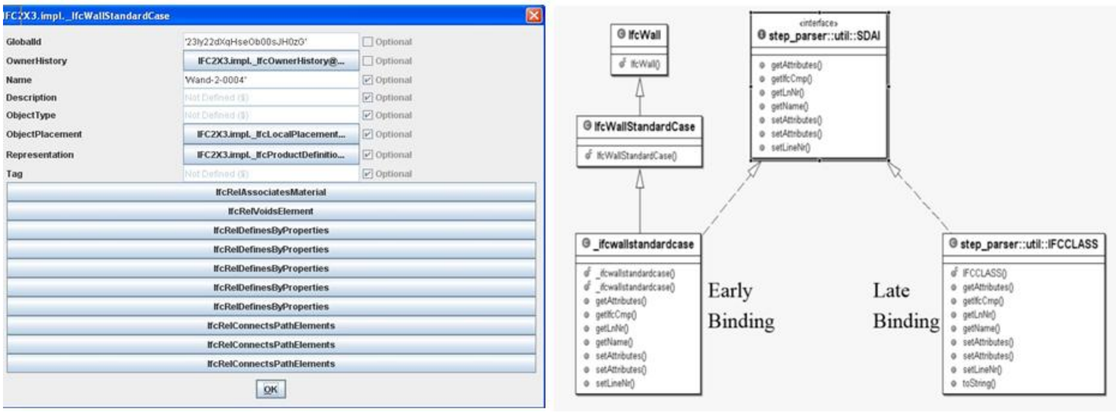

Figure 3 Right: Attribute values and Metadata Left: Right: UML class diagram showing the IFC java implementation classes

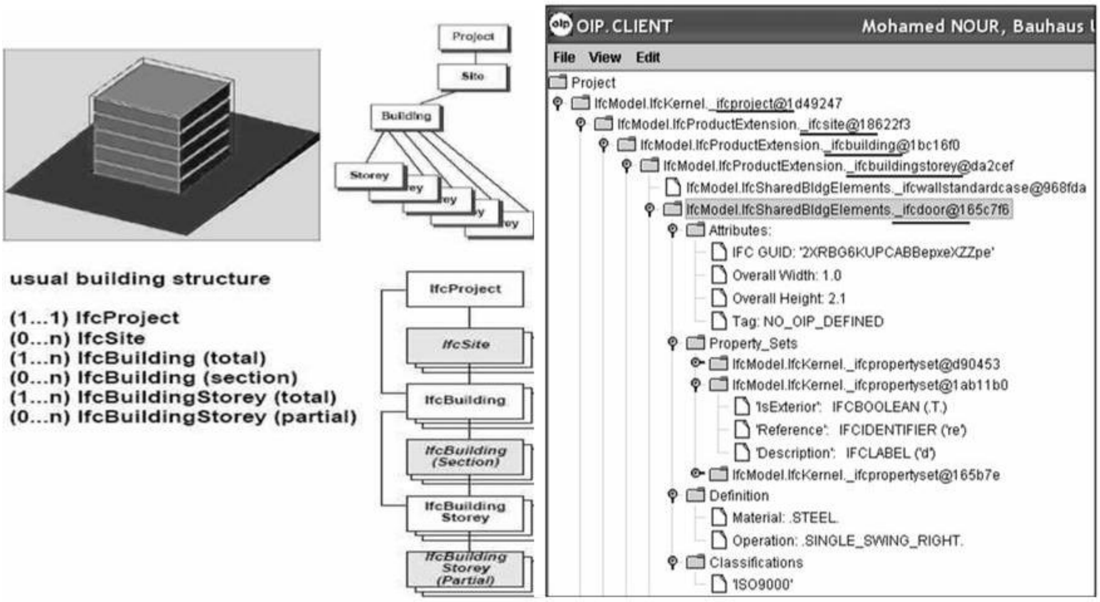

Figure 4 Right: The IFC model hierarchial tree structure source: IFC2X3 model implementation guide. Left: The java Tree view of the IFC model created by the Author as a part of the IFC toolbox

visualization, queries, etc. as shown in Fig. 3 to the right.

A further step is the visualization of the IFC Building Model objects. Such data visualization takes place in several forms. The first form is the IFC hierarchical tree structure of the building, where it is divided into sites, buildings, stories, building sections … etc. as shown in Fig. 4. Other 2D and 3D geometrical forms can be seen in Fig. 5 and Fig. 6.

As a next step the IFC model is mapped to a relational database (Microsoft Access) as shown in Fig. 5 to the right. SQL (Structured query language) is used for checking the semantic part of the model. This can include explicit data such as the height of a certain floor or a space, or the value of a property set item for an IFC object, etc… . Mapping the IFC object oriented model to a relational database is done in a manner that mainly depends on the GUID (Global Unique ID) of the IFC object to be the main key for data finding. Moreover, the complexity of the objectified relationship classes in the kernel of the IFC model is bypassed by directly linking leaf data to the main object. This can result in some redundancy by repeating same data for different objects, but it proves to be very efficient in information retrieval and queering the model.

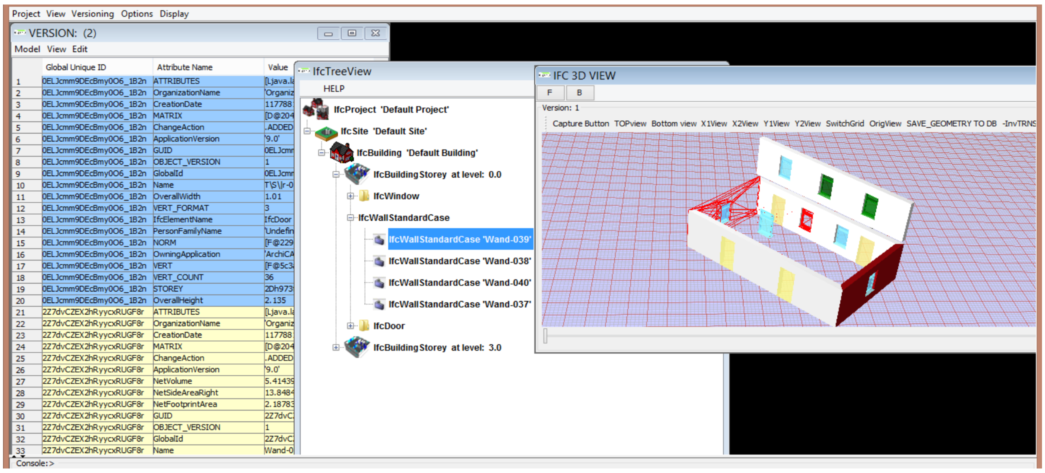

Figure 5 Left: IFC mapped relational model. Middle: IFC Project Tree. Right IFC Java3D viewer. Source: Author

A Java 2D and 3D graphics viewer that are developed by the author are also used for visualization as shown in Fig. 5 and Fig.6. Visualizing the IFC model is carried out mainly by making use of the graphics engine of Java2D and Java3D. For more information, the reader can refer to [18]. This is quite useful for creating bounding volumes for the IFC objects and at each node of the IFC model's hierarchical tree. This helps very much in imposing geometrical rules and measure on the underlying building design. The IFC Java2D model proves also to be necessary for checking measures on projection planes.

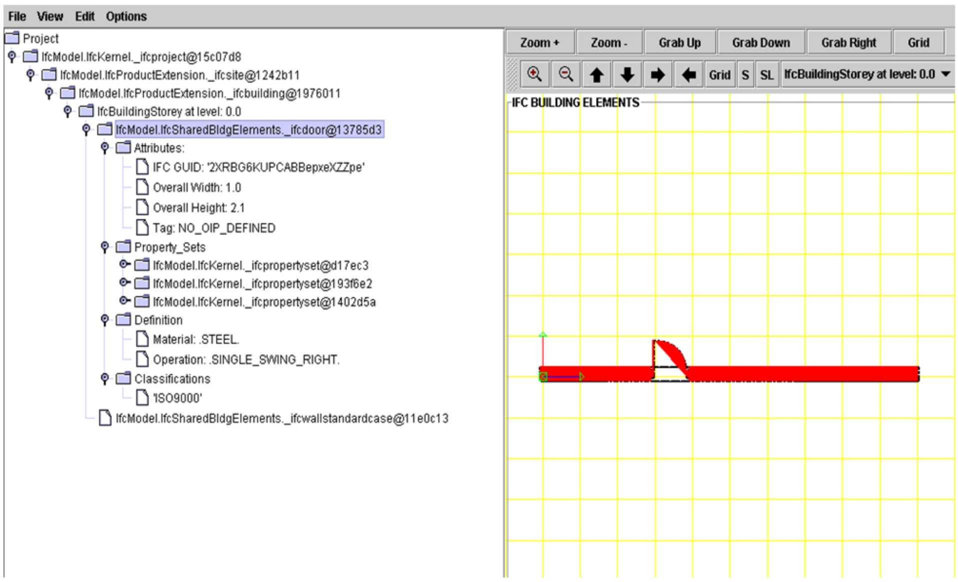

Figure 6 Left: IFC Project hierarchical tree. Right: 2D Java View of the IFC Model

Finally, the legislative rules can be easily encoded in Java for checking buildings' designs. Java is much easier to use for this purpose rather than EXPRESS due to the fact that EXPRESS is not a programming language, whereas Java is one of the most popular programming languages in the world with libraries in all fields of programming. In addition, a variety of popular Java text editors and compilers are available. For example the ECLIPSE platform and java editor.

III. ALGORITHMS FOR CHECKING BUILDINGS' DESIGNS

For the purpose of checking building designs, the semantics of the IFC model is used for filtering and finding explicit data. Moreover, the IFC model data is mapped to a relational database as shown in the previous section, where the execution of SQL queries is possible. In the meantime, the IFC model is represented graphically in both 2D and 3D using Java. All objects' geometries are triangulated (tessellated) using the Java3D platform. This process enables dealing with geometrical polygons representing the objects.

The geometry of each object is a subset of 3-dimensional space. The shape of this object (subset) can be represented as a collection of polygons, primitive solid shapes or curved surfaces. For legislative rules' checking, the rules themselves have to be interpreted and mapped to geometrical objects that are used to detect violations. The relation between the rules' objects and the BIM objects determine the conformance state of the design. For this purpose, different types of graphical queries are executed to determine the relation between the rules' objects and the BIM objects. These types of queries can only be executed on the basis of the semantic queries of the IFC model. The IFC instance type of the object has to be identified first. As a second step, the aggregation relationship in the model has to be fetched. For example, we have to filter the model for external walls of the ground floor. The result is a set of objects that are contained in the ground floor.

The Java based graphical queries enable the execution of proximity queries that returns information about the relative placements of the rule objects and the BIM objects.

1. Intersection query: It determines whether both objects intersect (interpenetrate) or not.
    a. Penetration Distance: It is the smallest distance needed to separate between the two objects. (The minimum distance needed to translate one object to make it disjoint from the other object).
2. Distance query: It computes the distance between two objects.
    a. Spanning Distance: The distance between the most separated points between the two objects.
    b. Hausdorf Distance: The greatest distance of any of the points of one object from all the points of the other object. (The maximum deviation of one set of points from the other).
    c. Separation Distance: Is the length of the shortest line joining any of the points of the two objects.

The above testing functionalities are made possible by using the collision detection functionalities of the Java3D library that is originally used in computer gaming and simulation industries.

IV. MAPPING LEGISLATIVE RULES AND CHECKING BUILDING DESIGNS

In this paper, a subset of the Egyptian legislative rules for residential buildings in new settlements is considered. These rules are:

1- Maximum foot print of built up area on the ground floor should not exceed 50% of the total land plot area.
2- Basement foot print should be below zero level and its area should not exceed 75% of the total land plot area.
3- Maximum foot print for typical repetitive floors should not exceed 110% of the ground floor foot print.
4- All structural columns should lie inside the foot print area of the ground floor.
5- Foot print of built up area on the roof should not exceed 25% of the area of the ground floor.
6- Width of treads on stairs should not be less than 0.8m
7- Service shafts should have a minimum area of 10m² and a minimum dimension of 2.5m².
8- Maximum height of the building.
9- Maximum number of apartments.
10- Minimum areas for spaces (toilets, shafts, kitchens, bedrooms … etc.).
11- Should the building height be more than 16m, then it must have an elevator.

As a first step, both the geometry of the land plot building site and the BIM design have to be adjusted in the same orientation with relevance to the North direction and absolute global coordinates.

1) The bounding volume (Bounding Box) of the BIM model is generated. This is done automatically in Java3D, because it is based on a scene graph hierarchy. In the meantime, another bounding volume (a bounding box) that expresses the legislative constraints is created. This box is based on the maximum allowable built up foot print, maximum cantilevers on typical floors and maximum height of the building. Both bounding boxes are checked against their intersection. If the legislative bounding box includes the BIM model's bounding box, i.e. every point in the building geometry lies inside the bounds of the legislative box or polytope, then the building design has passed this preliminary test. Moreover, this test assures that the ground floor built up foot print is ok, the overall height of the building is ok and that cantilevers on typical floors are within allowable range.
2) The basement can be checked by querying directly the IFC model for the area of the basement slab and the height of the ground floor slab. Both are explicit attributes of the IFC model, where their values can be easily read.
3) This is checked in step 1.
4) Bounding Box representations of both the ground floor "B1" and the legislative constraints relevant to the ground floor "C1" are compared. If "B1" lies entirely inside "C1", then the check is ok.

5) Bounding Box representations of both the roof floor “BR” and the legislative constraints relevant to the roof floor “R1” are compared. If “BR” lies entirely inside “R1”, then the check is ok.
6) -: 11) This can be checked by querying directly the IFC model for the BIM objects’ values. This is made available through the mapping of the IFC model into a relational database.

V. CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER RESEARCH

The problem of obtaining building permits electronically has proven to be an important international problem. Several nations have been trying to automate the permitting process. The appearance of modern technologies in the construction industry such as the BIM technology has opened a wide window of opportunities for performing buildings’ designs legislative checks electronically. To achieve this aim, both the buildings’ designs as well as the legislative acts have to be represented digitally in a homogeneous manner that enables their contribution to the conformance checking process.

This paper has introduced a novel approach that depends on simple boundary representations of BIM objects in addition to representing legislative rules and constraints digitally for conformance testing. It abandons the complexity of the EXPRESS rule schema definition and presents an open approach that is not dependent on any proprietary software tool such EDM or Solibri for performing rules checking. Furthermore, it depends on open standards like the IFC and a popular programming language like Java.

As a next step, it is envisaged that more algorithms can be developed for mapping more complicated legislative rules into parametric digital formats that can be used for conformance checking. A user friendly graphical interface that includes an interactive Java runtime compilation functionality would present a breakthrough in such a domain.

Shifting from prescribed rules conformance checking to performance rules would also present a major breakthrough, where the building can be checked against its performance rather than its specifications.

REFERENCES

[1] ISO 16739 -2013. Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries.
[2] ISO 10303-11 1994. EXPRESS. Industrial automation systems and integration – Product data representation and exchange – Part 11: Description methods: The EXPRESS language reference manual.
[3] ISO 10303-21 STEP 1994. Industrial Automation Systems and Integration — Product Data Representation and Exchange — Part 21: Implementation Methods: Clear Text En-coding of the Exchange Structure, ISO 10303-21:1994 (E), ISO, Geneva, 1994.
[4] BuildingSMART 2016. International Home for Open BIM. http://www.buildingsmart.org/ last accessed on 13th of March 2016.
[5] ISO 10303-42:2003. Industrial automation systems and integration -- Product data representation and exchange -- Part 42: Integrated generic resource: Geometric and topological representation.
[6] EDM 2016. Express Data Manager from Jotne IT. http://www.epmtech.jotne.com/products/express-data-manager-edm last accessed on the 20th of March 2016.
[7] Solibri 2016. “Solibri Model Checker”, http://www.solibri.com/products/solibri-model-checker/ last accessed 13th of March 2016.
[8] Navisworks 2016. Clash Detection and Coordinatin Tools. http://www.autodesk.com/products/navisworks/features/coordination/list-view last accessed on 13th of March 2016.
[9] TEKLA 2016. Tekla Structures, Clash Check Manager, http://teklastructures.support.tekla.com/190/en/mod_clash_check_manager last accessed on 13th of March 2016.
[10] VICO 2016. VICO Offfice Constructability Manager 3D BIM clash detection, http://www.vicosoftware.com/products/3d-bim-software-clash-detection/tabid/229117/Default.aspx last accessed 13th of March 2016.
[11] Eastman, C., Lee, J., Woodbury, R., and Crawford, J. 2009. Automatic Rule-based checking of building designs, Automation in Construction, 18(8), pp. 1011-1033
[12] SmartCodes, ICC, International Code Council, USA, 2006, http://web.stanford.edu/group/narratives/classes/08-09/CEE215/ReferenceLibrary/International%20Code%20Council%20(ICC)/SMARTcodes%20fact%20sheet%2011-01-07.pdf last accessed on the 5th of April, 2016.
[13] Hjelseth, E. 2012. Converting Performance based regulations into computable rules in BIM based model checking software. In eWork and eBusiness in Architecture, Engineering and Construction: Proceedings of the 9th European Conference on Product and Process Modelling (ECPPM) conference, pp. 1-10. Sophia Antipolice, France.
[14] Fiatech. 2012. AutoCodes Project: Phase 1, Proof of Concept Final Report. Construction Industry Institute. The University of Texas, Austin, USA.
[15] Vitali, F., and Zeni, F. 2007. Towards a country independent data format: the Akoma Ntoso experience. In Proceedings of the V legislative XML workshop, pp 67-86. Florence, Italy.
[16] CORENET e-Plan Check System. 2006. The Importance of IFC for Implementation in Singapore https://www.jacic.or.jp/acit/round2-teo2.pdf last accessed on the 13th of Feb. 2016
[17] FORNAX Plan Check Expert 2016. http://www.novacitynets.com/fornax/about.htm last accessed on the 14th of Feb. 2016
[18] Nour M. & Beucke K. “Manipulating IFC model data in conjunction with CAD”. In: Scherer R. J., Katranushkov P. and Schapke S. (eds): Proceedings W78 CIB – 22nd Conference on Information Technology in Construction, 19-21 July, Dresden, Germany. 2005 Institute for Construction Informatics, Technische Universität Dresden, Germany, ISBN 3-86005-478-3.