ACTIVe3D : Semantic and 3D Databases for Civil Engineering Projects

Christophe Cruz
Active3D-Lab
21 Avenue Albert Camus
BP 66606, 21066 Dijon, France
+(00) 380 539 595
c.cruz@active3d.fr

Renaud Vanlande
Active3D-Lab
21 Avenue Albert Camus
BP 66606, 21066 Dijon, France
+(00) 380 539 595
r.vanlande@active3d.fr

Christophe Nicolle
Le2i – UMR CNRS 5158
Université de Bourgogne
BP 47870 – 21078 Dijon, France
+(00) 380 396 867
cnicolle@u-bourgogne.fr

Abstract

In this paper, we present a web-based platform called ACTIVe3D BUILD SERVER. This platform lets geographically dispersed project participants—from architects to electricians—directly use and exchange project documents in a centralized virtual environment during the life cycle of a civil engineering project. This platform is based on IFC which associate in the same model the semantic and the 3D representation of business objects which describes a building. IFC files are textual files whose size can reach 100 megabytes. The elements defined in these files are organized in cyclic and hierarchical graphs generated from various civil engineering contexts. Due to their structure and their size, their handling and sharing is a complex task. We propose a method to analyze and restructure IFC instances as a set of XML trees. This reorganization is related to the analysis of the various contexts present in IFC files. This representation in the shape of tree allows the actors of a project to exchange partial IFC files by extracting IFC elements from a specific context, to merge various partial IFC files into only one and to store the knowledge of the building into IFC tables managed by a relational DBMS. Some trees are used to dynamically construct 3D scenes which lets participants move around in the building being designed and obtain information about the objects that compose it.

1. Introduction

A large variety of CAD softwares is proposed in civil engineering area. CAD software is specialized for a civil engineering profession (metal structure, concrete structure, heat engineer, plumber...) and has its own modeling concepts. The data exchange between CAD softwares is not possible, due to the heterogeneity of modeling field. Buildings therefore take longer to design and build and cost more to construct and operate. Current CAD softwares model the building with 2D geometry. Each element of the building is represented by a set of vectors. In this formalism, no semantic information specific to the building objects is modeled. To bridge this gap, the International Alliance of Interoperability proposed a standard called IFC that specifies object representations for construction projects [4]. IFC format is a business model which associates business semantics with 2D/3D geometry. The addition of business semantics makes it possible to limit the redundancies, to identify instantaneously each element composing the building and to qualify more quickly the building. Industry foundation classes (IFC) include object specifications, or classes, and provide a useful structure for data sharing among applications. A door IFC, for example, isn't just a simple collection of lines and geometric primitives recognized as a door; it is "aware" that it's a door and has a door's attributes.

The adoption of this format by all the leaders of CAD softwares allowed a better interoperability in the exchanges of information between the various civil engineering professions. Each civil engineering profession intervenes in the project in a particular context, enriching the plan with its own vocabulary, its own concepts and its own business's objects. At the end of the project, the IFC file corresponding to the building contains the set of the building elements, in a multi-context definition. IFC files are textual files whose size can reach 100 megabytes. Several IFC files can coexist on the same civil engineering project.

A civil engineering project consists of phases codified by the profession (Primary studies, pre-project summary, pre-project detailed, project, and so on). Every phase generates a set of specific-scale plans, or text documents that the architect validates with the client, administrative authorities, or financial partners before moving to the next phase. Only the architect can manipulate all the phases of a project. For example, the architect generally validates the "pre-project summary" phase by obtaining planning permission from administrative authorities. At this level, administrative authorities can visualize and print plans contained in the previous phase ("Primary studies") at the 1/200 scale and, upon completing their work, indicate the planning permission electronically. All authorized participants then receive an automatically generated email indicating the revised document's availability. Next, in the "pre-project summary" phase, the architect must define the space structures' real measurements at the 1/50 scale. At this level, other team members can consult the plans and use specific building software to specify the electrical network, plumbing, and so on. Once the plans are completed, the architect can correct them and reevaluate project costs as needed. At each step of the project, plans are created, exchanged and modified by all members of a project. All the exchanges are based on the IFC files which are the numerical translation of the architect plan. The IFC model is used as intermediate model for the exchange of information between CAD software.

Our goal is not to consider the IFC as a simple intermediate format between heterogeneous CAD softwares, but to consider the IFC as the base of the development of a Building Information System (BIS). This Building Information System provides a set of processes, methods and databases placing the IFC to the heart of the management of a civil engineering project.

To address this problem, a research team in computer science from the LE2I laboratory at the University of Bourgogne, in collaboration with the "Groupe Archimen", carried out research on the handling of languages derived from XML, in particular IFC and X3D languages. The "Groupe Archimen" is a civil engineering and facility management firm. The results of this collaboration are the creation of a private company called ACTIVe3D-LAB (http://www.active3d.net) which employed ten people (three administrative staff members, eight developers, and five University of Bourgogne researchers including two PhD students) and the development of the ACTIVe3D BUILD SERVER to improve civil engineering project management.

This Web-based platform lets geographically dispersed project participants—from architects to electricians to plumbers—directly use and exchange project documents in a centralized virtual during the life cycle of a civil engineering project. A 3D visualization lets participants move around in the building being designed and obtain information about the objects that compose it. This platform derives from research on how to represent computer graphics modeling semantics within a relational database.

Currently, 126 people use the application daily and manage sixteen projects. The ACTIVe3D BUILD SERVER was rewarded for the technological innovation gold medal to the international show BATIMAT in Paris, November 2003.

The rest of this paper is articulate in three sections. The first section describes the complexity of the IFC structure at the model level and the instances level. The second section presents our method to analyze and restructure IFC instances as a set of trees. This reorganization is related to the analysis of the various contexts present in IFC files. This representation in the shape of tree allows the actors of a project to exchange partial IFC files by extracting IFC elements from a specific context, to merge various partial IFC files into only one and to store the knowledge of the building into IFC tables managed by a relational DBMS. Some trees are used to dynamically construct the graphical interface of the building manager tool. The last section presents the development features of our method.

2. IFC Description

The "Industrial Foundation Classes" (IFC) is an ISO norm to define all components of a building in a civil engineering project. An example of IFC file structure is given in script 1. This file described a building with more than 111000 business objects (one lines per object). To understand the complexity of the IFC, this section presents the IFC model level and the IFC instances level.

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION (('ArchiCAD generated IFC file.'), '2;1');
FILE_NAME ('Karlstr.IFC', '2002-06-19T15:48:48', ('Architect'), ('Building Designer Office'), 'PreProc - IFC Toolbox Version 2.x (00/11/07)', 'Windows System', 'The authorising person.');
FILE_SCHEMA (('IFC2X_FINAL'));
ENDSEC;
DATA;
#1 = IFCORGANIZATION ('GS', 'Graphisoft', 'Graphisoft', $, $);
#3 = IFCPERSON ($, 'Undefined', $, $, $, $, $, $);
#4 = IFCORGANIZATION ($, 'OrganizationName', $, $, $);
#5 = IFCPERSONANDORGANIZATION (#3, #4, $);
#7 = IFCSIUNIT (*, .LENGTHUNIT., $, .METRE.);
…..

#111029 = IFCRELCONTAINEDINSPATIALSTRUCTURE
 ('25wKeWex98fQp5Pukf_Ilc', #6, 'BuildingStoryContainer',
'BuildingStoryContainer for Building Elelements', (#111007), #110989);
#111030 = IFCRELAGGREGATES ('216Bv$yJj3tQjFeDohe6fQ', #6, 'BuildingContainer', 'BuildingContainer for BuildigStories', #30, (#34, #16236, #29699, #56800, #62077, #67336, #72633, #91702, #110989));
#111031 = IFCRELAGGREGATES ('17XMUtNDr8FeFMtR6rOcy5', #6, 'SiteContainer', 'SiteContainer For Buildings', #28, (#30));
```

#111032 = IFCRELAGGREGATES ('0pMN8yq8vDRfwN_tnJREKC', #6, 'ProjectContainer', 'ProjectContainer for Sites', #26, (#28));

ENDSEC;

END-ISO-10303-21;

Script 1. Example of IFC file

1.1 IFC Model

IFC files are made of objects and connections between these objects. The attributes in the objects, describe the "business semantic" of the objects. The connections between objects are represented by "relationship elements". The IFC model is an object model modelled with the EXPRESS language. This model describes approximately 600 classes. There are three types of IFC classes: object classes, relationship classes and resource classes.

1. The object classes consist in a triplet (GID, OS FU), where GID defines the identifier of the IFC object, OS defines the ownership features of this object and FU are the functional units. These functional units define the context of use of the classes (i.e. the geometrical representation, its localization, its composition, etc). In the script 1, the #5 element of the type IfcPersonAndOrganisation reference the #3 and # 4 elements.
2. The resources classes constitute the set of attributes used in the description of the functional units. These resources are organized in a hierarchical graph.

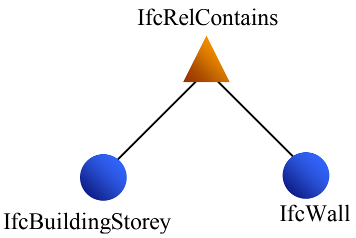


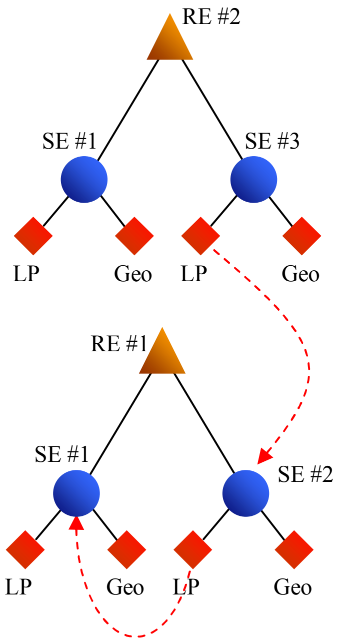
Figure 1. A building storey and a wall are connected by an element IfcRelContains.

The relationship classes represents the various relations (relation of capacity, relation of aggregate, etc.) between the object classes and has functional units. The corresponding elements are prefixed by IfcRel. The IfcRelAggregates element from Script 1 having the identifier #111030 constitutes a relation of aggregate between the element #30 and the following element list (# 34, # 16236, # 29699, # 56800, # 62077, # 67336, # 72633, # 91702, # 110989). The element #110989 is also referred by the element #111029 which is a link called IfcRelContainedInSpatialStructure. This means that if an element can be referred by several elements then two elements can mutually refer them by the intermediary of one or more relations. This mutual reference forms a cyclic graph.

1.2 IFC Instances

The study of the IFC instances reveals the complexity of the overlap between instances of relationship classes and instances of object classes. At this level, there exist two types of link between objects. We called them the indirect link and the direct links. The indirect links are defined by the instances of the relationship classes.

The direct links are defined by the instance of resource classes. The indirect links are characterized in the figure 1 by ▲. The object instances of the architecture field become semantic elements. In the figure 1, these elements are graphically represented by ●. The resource instances are represented by ◆.



RE : Relationship Element

SE : Semantic Element

LP : Local Placement

Geo : Geometry

Figure 2. Example of direct link between semantic objects

Figure 1 shows the indirect links between the semantic elements using a relationship element. Figure 2 shows the direct bonds between semantic elements, they are noted in red. There are two types of direct links. The first type defines the resources of the element. These resources are structured using a tree structure. The second type defines a direct link between two semantic elements. The IFC model defines only one type of links between two semantic elements. This is the placement link between the semantic elements for design of a building in a 2D/3D scene. This relation is carried out by the IfcLocalPlacement attribute of the semantic element. It defines the reference mark of the current object compared to the reference mark of the father object of the direct relation. The set direct link formed by the IfcLocalPlacement attribute forms a tree structure of the 2D/3D scene. The main difficulty is to handle at the same time the cyclic semantic elements and the hierarchical structure of the 3D elements.

3. Active3D Methodology

IFC files are textual files whose size can reach 100 megabytes. Several IFC files can coexist on the same civil engineering project. IFC files are cyclic graphs combined with hierarchical graphs. Due to their size and their structure, their handling and sharing is a complex task. To resolve this problem we have developed a method to analyze and to decompose the IFC structure. The main IFC advantage is to associate semantics and 3D into IFC objects. The handling and management mechanisms for IFC files (such as fusion of files into only one, extraction of partial files dedicated to only one context, the visualization and the storage) will have to deal with a multitude of semantic values for the same object, according to the context of use [10]. To achieve these goals, we defined a hierarchical structure of context called the contextual view. It is a partial view of the IFC file which allows handling of IFC elements belonging to a specific context. This solution reduces the graph complexity translating a multi-contexts cyclic graph into a set of mono-context trees. This process is done using business rules. An example of a business rule is “a door is in an opening element in a wall”. Resulting from this process, many contextual trees are built starting from the IFC files, such as the contextual tree of capacity defining the object composition (a building contains two floors, a floors contains beams, walls, and so on.). The figure 3 presents the tree of capacity.

The main tree is the geometrical contextual tree which contains the topological relations between the various objects. From these trees, the resulting 3D scene corresponds to a particular business view [1, 2, 3, 5]. The figure 4 shows the 3D representation of plumbing view of an IFC file. From this 3D view, the users can reach directly the various functional units composing the IFC file.

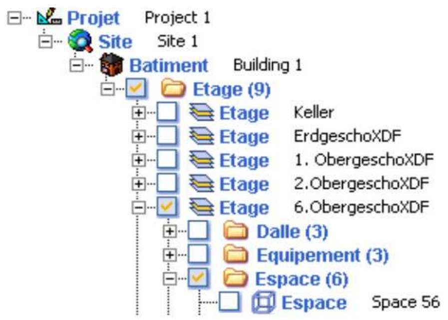


Figure 3. IFC Tree of capacity

Through this interface, trade information can be manipulated. These business views are textual information, from which specific documents can be generated or associated (technical reports, management information, etc.). In the 3D scene generation process, all the geometry defined in IFC trees is converted into triangular surface model [7]. During this conversion, the 3D objects are associates with the GID. The GID is the general identifier used to identify each business object of an IFC file. In the script 1, the GID of the IFCRELAGGREGATES object is '0pMN8yq8vDRfwN_tnJREKC'. This GID is used to link the 3D visualization with the information stored in the databases. All insertion of new data in any base is reference by a GID correspondent to an IFC object. All trees generated in the platform are XML trees [9]. We have developed a specific database schema dealing with the semantic and the 3D aspects of the IFC. The trees and the component elements are stored in a relational database and manipulated using the SQL. From this database and the GID, all types of information can be attached to the 3D visualization of business Object.

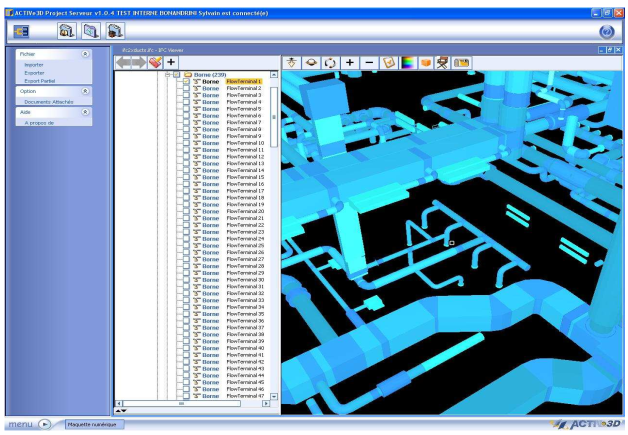


Figure 4. A 3D scene corresponding to the plumbing context

From the definition of the contextual trees, we have defined the corresponding XML grammars. Thus, each functional unit and each context are manipulated as XML documents. These grammars, specific to a civil engineering profession, are used to format IFC data exchanged during the life cycle of a civil engineering project. The contextual trees are transformed using XSL style sheets. These transformation processes are web services. IFC Services provides on the ACTIVEe3D server are XSL processes associated with a context. The use of XSL is extended to generate other documents such as technical reports and so on. In the same way, the graphic contextual trees are transformed into X3D documents. Thus, the 3D scene is personalized according to the service called. Moreover, the graphic elements preserve connections with the information system containing management information. These connections allow the 3D scene to carry out queries on the information system concerning the graph elements.

A complete description of this method and the presentation of the resulting IFC schema from our relational database are beyond the scope of this paper.

4. Development

We're running the ACTIVe3D BUILD SERVER on a Dell dual-processor server (Pentium 4, 2.4 GHz, 512 Mbytes DDR RAM, 180 Gbytes disk space, Red Hat Linux 7.2 OS). An Oracle 9i relational database management system (RDBMS) manages the data layer. We developed the broadcast and behavioral layers in Java using Sun Microsystems' JDK 1.4.1) to facilitate development and ACTIVe3D BUILD SERVER porting to other operating systems [5]. We use Microstar's Sax and Sherry's Dom parsers for all XML streams. We're currently developing the 3D visualization module called the "IFC Viewer" in GL4JAVA using the NetBeans 3.4.1 development environment. The IFC Viewer represents both the IFC Tree and the 3D visualization of a Building. All objects composing the building are managed through the tree or dynamically using the 3D scenes, activating object by clicking on it.

The IFC Viewer was certified ISO/PAS 16739 in May 2003 by the IAI. It allows the user to see the complete description of the building with an IFC tree view. Moreover, it lets participants move around in the building being designed and obtain information about the objects that compose it. This 3D is "semantically" dynamic. The structuring of the scene is calculated in real time from the IFC database according to geometrical features (distance from the point of view, object hides...) and semantic features (rights users, type of user, stage of visualization of the project...).

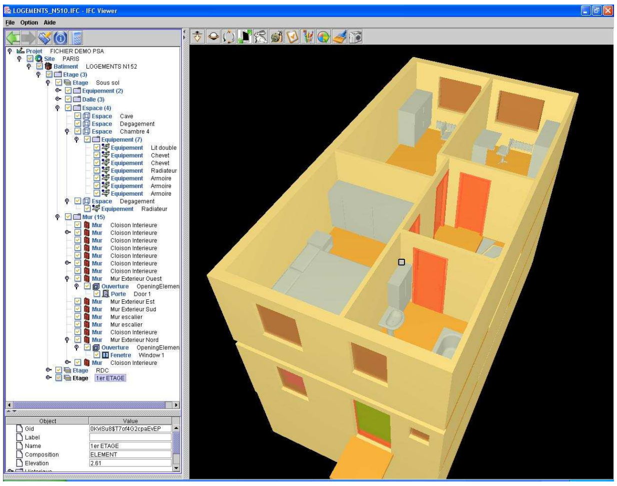


Figure 5. A 3D scene corresponding to the architectural context

The figure 4 presents a snapshot of the IFC Viewer. The entire platform is developed using web-services that are associated dynamically in a collaborative application at the user connection according to the user rights [6].

We evaluated the IFC loading, conversion and visualization performance. Table 1 shows the results of the tests.

File size		4.4 Mb	7 Mb	22 Mb
Active3D	Load file	12''	18''	1'9''
	Compute geometries	11''	1'23''	7'

Table 1. Measure of loading, converting and 3D visualization

5. Conclusion and Future Works

The "Industrial Foundation Classes" (IFC) are an ISO norm to define all components of a building in a civil engineering project. IFC files are textual files whose size can reach 100 megabytes. Several IFC files can coexist on the same civil engineering project. Due to their size, their handling and sharing is a complex task. In this paper, we presented an approach to automatically identify business objects in the IFC files and simplify their visualization and manipulation on the Internet using XML contextual trees. We construct business information systems based on the IFC. This information system is made of IFC database and web-services.

Now, we are study the update of the IFC database from multiple IFC sources. This multiple update of IFC files generates structural and semantic conflicts (for example, a heating pipe crossing a door). The resolution of these conflicts passes by a semantic and geometric analysis of the associated contextual trees. Another way is to extend our system to facilities management. In this area, the mining of IFC data is crucial to define strategic choice in the management of the building.

Acknowledgments

We thank the ANVAR and the ANRT for their support. We also thank many people at the Active3D-LAB, especially Olivier Gaudard, Florent Barth, Sylvain Bonandrini, Samuel Desgouilles, Sebastien Delune and Estelle Jacquemin for their time and their cooperation.

References

[1] Boon-Hee Kim, Jun Hwang, Young-Chan Kim, *The design of high-level database access method in a Web-based 3D object authoring tool*, The Fourth International Conference on Distributed Communities on the Web, 3-5 April 2002, Sydney, Australia

[2] Bowman, D., Davis, E., Badre, A., and Hodges, L. *Maintaining Spatial Orientation during Travel in an Immersive Virtual Environment.* Presence: Teleoperators and Virtual Environments, vol. 8, no.6, 1999, pp. 618-631

[3] Howard Abrams, Kent Watsen, Michael Zyda: *Three-tiered interest management for large-scale virtual environments*, VRST 1998, pp. 125-129

[4] International Alliance for Interoperability. Ifc, industry foundation class. http://www.iai-international.org/

[5] Kornel Szabo, Peter Stucki, Patrick Aschwanden, Thomas Ohler, Renato Pajarola, Peter Widmayer*, A Virtual Reality Based System Environment for Intuitive Walk-Throughs and Exploration of Large-Scale Tourist Information*. Proceedings of the Enter95 Conference, 1995, pp. 10-15

[6] Christophe Nicolle, Kokou Yétongnon*, XML Based Toolkits for the Interoperability of Web Information Systems*, Chapter book, Web-Enabled Systems Integration: Challenges & Practice, 2002

[7] Rémi Ronfard, Jarek Rossignac: *Full-range Approximation of Triangulated Polyhedra*. Computer Graphics Forum 15(3): 67-76 (1996)

[8] Alex Vakaloudis, Babis Theodoulidis: *The Storage and Querying of 3D Objects for the Dynamic Composition of VRML Worlds*. IV 1998: 267

[9] Extensible Markup Language (XML): http://www.w3.org/XML/

[10] F. Recanati. *Direct Reference: From Language to Thought*. Blackwell Publishers, Oxford, UK, 1993