

# **CIMEDE: An architectural modeler for a modular industrial building system**

## *A Case Study*

Anne Van de Vreken, Vincent Delfosse, Dimitri Schmitz, Cyril Lorquet, Pierre Leclercq

*Université de Liège, LUCID-ULg (Lab for User Cognition and Innovative Design)*

*1, Chemin des Chevreuils, Bât B52*

*4000 Liège*

*Belgique*

[A.VandeVreken@ulg.ac.be](mailto:A.VandeVreken@ulg.ac.be), [vincent.delfosse@ulg.ac.be](mailto:vincent.delfosse@ulg.ac.be), [dimitri.schmitz@ulg.ac.be](mailto:dimitri.schmitz@ulg.ac.be),  
[cyril.lorquet@ulg.ac.be](mailto:cyril.lorquet@ulg.ac.be), [pierre.leclercq@ulg.ac.be](mailto:pierre.leclercq@ulg.ac.be)

**Key words:** Architectural modeler, Computer Aided Architectural Design (CAAD), Design support Tools.

**Abstract:** CIMEDE is a wooden industrialized building system focused on modularity, sustainability and cost effectiveness. This paper presents an applied research project focused on the implementation of software able to efficiently assist the various stakeholders involved in the design and the construction of CIMEDE housings (architects, clients, process managers, builders, factory workers...). The main objective is to develop an easy to use and bespoke 3D CAD modeler addressing the multiple stakeholder's needs, and able to integrate the whole complex process of industrialization.

## **1. PROJECT OVERVIEW**

This paper presents the work conducted within the CIMEDE project. This project is funded by the Walloon "Plan Marshall" and has lasted 3 years for a total of 2 millions euros budget. It brings together ten partners from both the industrial and scientific domains. This applied research aims to develop a widely industrialized building system to produce low-cost, single family homes that meet high standards of low-energy consumption as well as sustainability objectives.

CIMEDE develops adaptable and sustainable building concepts that take advantages of advanced manufacturing systems and methods. It promotes the use of prefabricated parts and assemblies, and develops the tools for design, configuration, production and assembly. Off and on-site production and assembly processes are designed, that will be heavily dependent on the use of manufacturing infrastructures.

As part of this research, a CAD software has been implemented. It allows the modeling of a house, by using a library of prefabricated CIMEDE components. The design of the software has strongly focused on the user interface. Taking benefits from the many constraints given in the CIMEDE technology, the software presents minimal user encoding and interactions. For instance, with its integrated automatic constraint recognition system, as the user builds his model, the application automatically detects whether the manipulated component can be integrated or not. The application's data model holds the types of building components, the information useful for the manufacturing process (such as quantity survey). To ensure interoperability with other CAD tools, the complete building model can be exported in a standard exchange format (IFC).

This paper focuses on how a modeling tool can support the construction process of buildings using prefabricated parts and assemblies. The paper is structured as follows: Section 2 discusses the research strategy chosen. Section 3 describes the system architecture required for the development of a virtual environment supporting the construction of buildings. Section 4 describes the development challenges encountered during the project, in terms of data model and BIM interoperability. Finally, section 6 lists the conclusions and future work to be undertaken to further enhance the CIM3DE application.

In this paper, CIMEDE identifies the company that initiated the research project, whereas CIM3DE describes the modeling software dedicated to CIMEDE's Building Construction System. We define a Building Construction System [BCS] as *"a set of parts and rules where the details are solved before actual buildings are planned, the same parts and their details are reused for a large number of buildings which are different as products but generated by a similar process"* (Richard, 2005).

## **2. RESEARCH STRATEGY**

### **2.1 Users Requirements**

The CIM3DE software is based on all the technical rules that compose the [BCS] to support the design process of the buildings. It is an integral part of the [BCS].

The software requirements were defined during the research process. Four software roles were found:

1. The architect designs his project using CIM3DE;
2. CIM3DE is used to support communication between customers and architect;
3. The software supports the industrial work (cost establishment, etc.);
4. The final customer designs his house alone.

CIMEDE experts have chosen the first two roles as main requirements. Therefore the software user is an architect. However the model, produced by CIM3DE, will be useful for other stakeholders. For the CIMEDE customer, the software is a 3D visualization tool, which can be used during and after the design process. The knowledge about the building system is embedded and hidden into the software. So the customer is able to use the software to visualize his project despite his lack of construction skills. For the future architect who will use the [BCS] through his projects, the software should be able to draw floor plans, cross sections and elevation. And finally for Industrials, CIM3DE lists the components to produce. All the components are linked to technical details and raw material by the ERP software (Enterprise Resource Planning) of the firm.

### **2.2 Modeling Strategy**

The choice of a main user with architectural skills has driven the integration of traditional architectural practice into the modeling process. The software has to allow a gradual design of the building with a lot of uncertainties at the beginning of the project. Therefore the strategy has been based on a drawing analogy (**Erreur ! Source du renvoi introuvable.**) rather than a component based modeling (**Erreur ! Source du renvoi introuvable.**). That approach enables to model a semi-defined project at the first stage of the design.

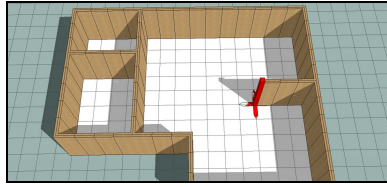


Figure 1. Drawing Strategy

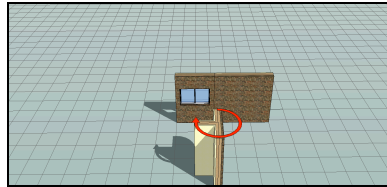


Figure 2. Assembly Strategy

The choice of the approach was driven by the estimation of the quantity of items that should be added in the library if an assembly strategy is chosen. This strategy was confirmed by the decision of producing long walls with variable length based on 10cm steps. This last decision could cause a big increasing of the quantity of component with an assembly approach. The advantage of the drawing strategy is then reinforced.

### 2.3 Lack of information about the industrial system during software design

Time constraints imposed by the research project led to use a concurrent engineering approach where system designer, simulation laboratories, industrial experts and software designer worked together as soon as possible. (Bogus, Molenaar, et al., 2005) describes the possibility to start two processes at the same time in function of four relationship types. These four relationships are considered in the present paper to describe the possibility of linking the design process of the Industrial Building system to the development process of the software CIM3DE (figure 3):

1. Two Independent Processes have no relationships. That situation has no sense in this project where the CIM3DE software has to be linked to the building system.
2. Dependent Processes are organized with sequences. A process begins when the second process is ended. The time constraints of the research project forces to dismiss this option.
3. The research project was described first for using Interdependent Processes. In this case, all the activities should be achieved

together. However some unpredictable technical constraints about the [BCS] has disallowed that kind of project management. Moreover the development process of the architecture of the software needs large working periods with weak links to [BCS] Design. Interdependent Processes are therefore impossible in this project.

4. Finally Semi-Independent processes, which require only partial information from other processes before they can begin, have been chosen.

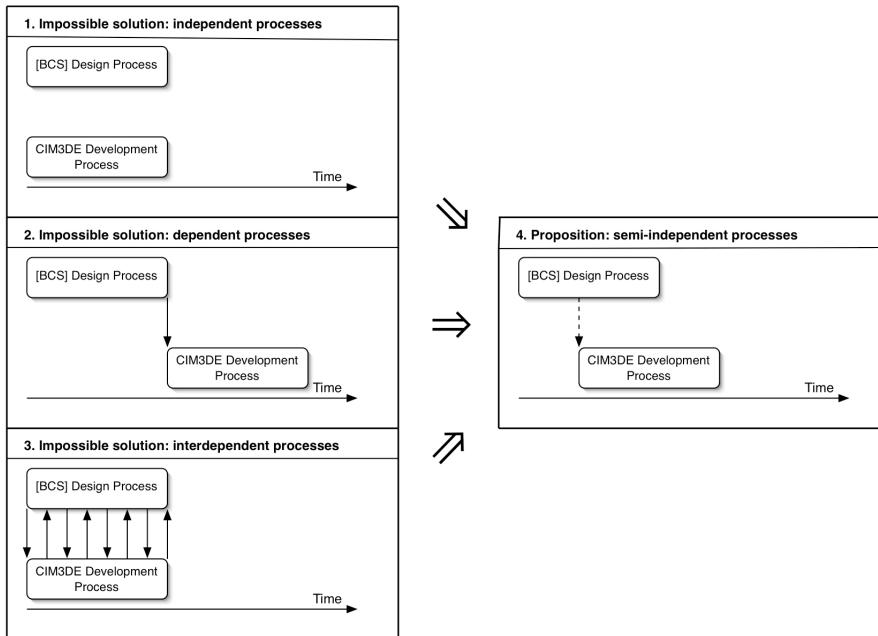


Figure 3. Relationships between Design process of the industrial [BCS] and the development process of the CIM3DE software: Rejected solutions and proposed solution

Exchanged information between Design process of [BCS] and Development of CIM3DE have been divided into 3 parts (figure 4):

1. Topological information about position and adjacency between components is used to represent the components as black boxes. The components are modeled in a parametric and symbolic 3D model.
2. Behavior information consist of rules, which has to be validated to ensure adequacy between topological model and what the [BCS] allows.

3. Technical information explains how to build a component of the [BCS]. The components are seen as a white box. The ERP software manages the production using technical information.

The CIM3DE Development process contains 2 activities “Parametric and symbolic 3D model” and “Rules”. The first one has started as soon as possible when the external definition of components was finished in the [BCS] Design process. The second activity began when the description of relationships between components was done. The technical information, that needs more resources during the [BCS] Design process, is used directly by the ERP Software of the firm without links to CIM3DE.

That approach has allowed building a semi-independent relationship between the design of [BCS] and the development of CIM3DE. The duration of the project has been reduced using parallelization, despite the obligatory exchange of information between the two processes.

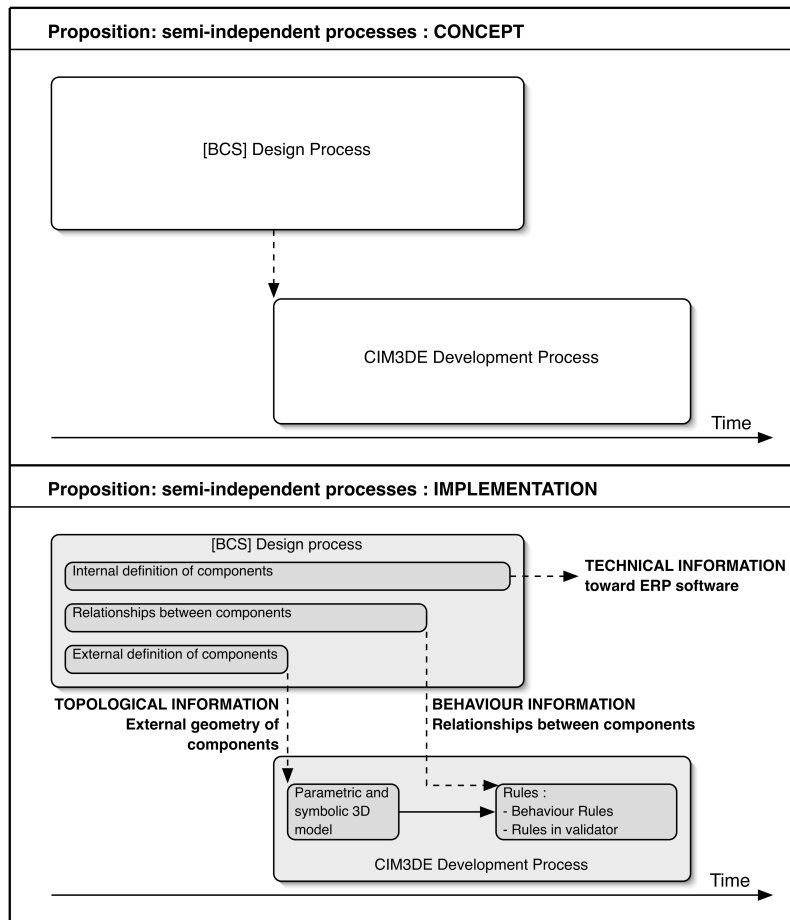


Figure 4. Information exchanged between the [BCS] Design and the CIM3DE development.

The rules are implemented either as modeling constraints during design, either in a validator tool, which check rules after the design. This last tool has been implemented for several rules, which slowed the design process if they are integrated in the software as constraints during modeling tasks.

### 3. CIM3DE CASE STUDY

This section describes the CIM3DE<sup>1</sup> software. It focuses on the specificities of the tool, compared to other traditional CAD software.

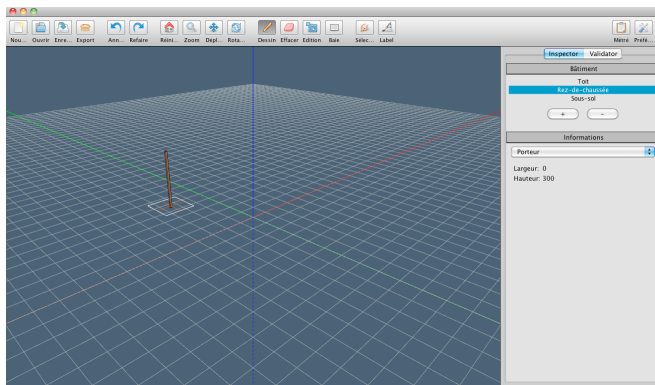


Figure 5. CIM3DE environment

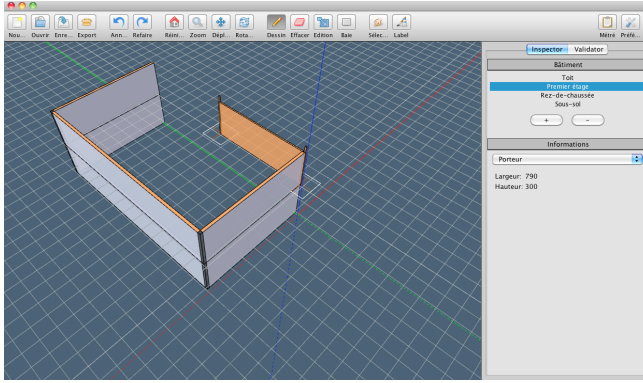
Within the building design environment the user can navigate freely through the 3D environment to view the components and the building, select components, use tools to create a building by linking components together, and specify its properties using the information panel on the right side. Figure 1 gives an overview of the CIM3DE CAD environment.

Currently, the user can either begin constructing a building from the empty environment, or load a partially constructed building previously saved. Then, he can export his model as an IFC file, which can be used in several commercial CAD applications for further processing optimization (the export processing is defined in Section 4, BIM Interoperability).

Traditional navigation tools are available in CIM3DE: multi-level undo and redo operations, camera tools (Reset, Zoom, Move, Rotate), design tools (Drawing, Delete, Edit, Bay), selection and properties tools...

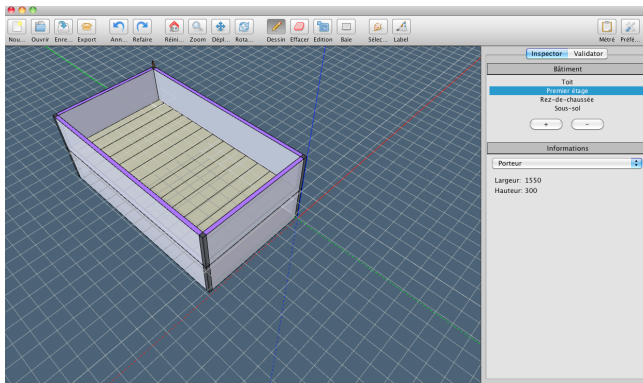
As the user drags a wall or a bay component through the model, the system detects the component and highlights it, as shows Figure 2. The user does not have to align the parts exactly - the system automatically detects which components can be mated together and aligns each of the components exactly for the user.





*Figure 6. Highlighting of components design*

When the user defines a four-sided volume floors and roofs are automatically generated by the system as shown on figure 3.



*Figure 7. Automatic generation of a floor from a four-sided volume*

Once a component or a volume is released, the user can specify its properties by acting on the sidebar. Figure 4 shows the label tool, which allows the user to specify properties of a volume previously defined in CIM3DE environment.

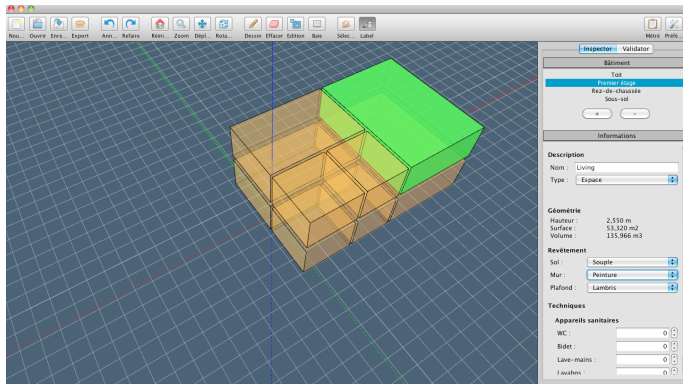


Figure 8. Volume selection to label

This is performed using the validation manager included in CIM3DE that aids the user in aligning parts exactly within the environment and maintains the relationships between each of the modules and their construction order (the validation manager is defined in Section 4.1.3).

As each component is added to the building, its properties and position are stored within the data model. With this information on the complete building, the software can provide the quantity of components to manufacture. This information can be exported as a spreadsheet for further quantity survey processing. Figure 5 proposes a view of an example building constructed with CIM3DE, and its associated quantity survey.

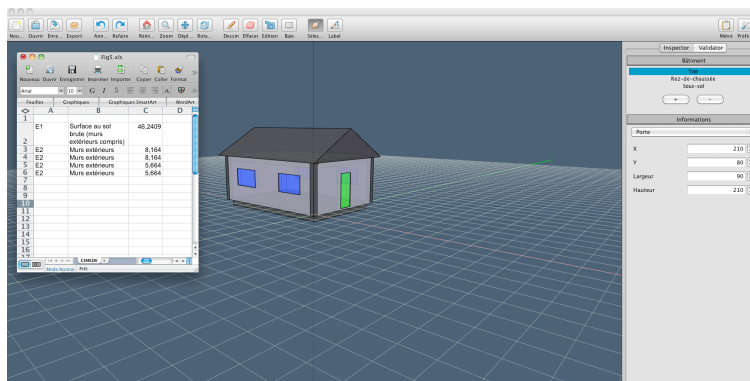


Figure 9. A building constructed in CIM3DE and its associated quantity survey.

## 4. DEVELOPMENT CHALLENGES

### 4.1 Data model

In a traditional approach, the development of the CIMEDE modeler would have started after the design choices for the constructive technology had been made. Unfortunately, due to planning constraints, both tasks have been conducted in parallel, which had a significant impact on them. On one hand, the constructive technology could benefit from the many questions raised by the software team, in need for a non-ambiguous and fully defined system, leading to significant, and sometimes fundamental, changes in its own technology. On the other hand, the development process had to be adapted to cope with the underspecified technology design.

To deal with this challenge, the design of the software data model was split in two parts: a “fundamental” layer, responsible for storing an abstract representation of the building, base on the most stable technology choices; and a “domain” layer, responsible for mapping the abstract concepts of the fundamental layer into architectural concepts. Moreover, this layer also encapsulates the details of the CIMEDE technology that are more likely to be modified.

#### 4.1.1 The fundamental layer

The fundamental layer is responsible for managing an abstract description for the walls and spaces of a given floor. It is organized in two parts. Firstly, a mathematical graph is used to store the walls as graph edges and the wall extremities or connections as graph nodes. (see *Figure 10*). From this graph, the bounded areas are automatically extracted and stored as a space representation. Secondly, a module has been developed for attaching any kind of properties to the elements of the graph.

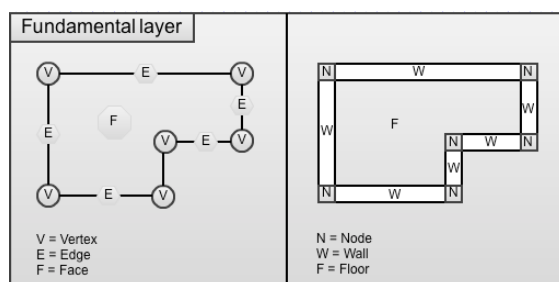


Figure 10. Fundamental layer

This model brings a lot of flexibility on how it can support the modification in the CIMEDE technology. On top of the fundamental layer, we have implemented different versions of the modeler, each time adapted to a different set of constructive technology choices. Here are some examples of the expressivity offered by this layer:

1. The wall width can be expressed in different ways around the abstract wall ‘fiber’ represented in the graph. Our application supports different wall widths calculation of external walls, bearing walls, etc.;
2. The wall connections are very important in CIMEDE, as they must resolve all the junctions of the technologies carried in the walls. They are considered as simple ‘black-box’ nodes in the fundamental layer, and are resolved at the upper levels through the use of dedicated properties;
3. Custom space types are implemented with attributes, like names (kitchen, dinning room, etc.), or types (stairs, technical areas, etc.);
4. With just a few attributes (orientation and initial position), the system can automatically compute the full layout of the floor elements;
5. Openings in the walls are also stored as wall attributes (doors, windows or empty bays).

#### 4.1.2 The domain layer

The domain layer of the final version of the modeler uses a common unique graph for all the bearing walls on all floors. It also maintains one graph per floor for all the walls of the floor. So, the software handles two different graphs per level. In order to automatically detect each space of a given level, the only required operation is merging them.

Regarding the roof level, a graph was chosen to represent the roof elements, because the bearing walls, handled in the common graph, define the areas described by the roof.

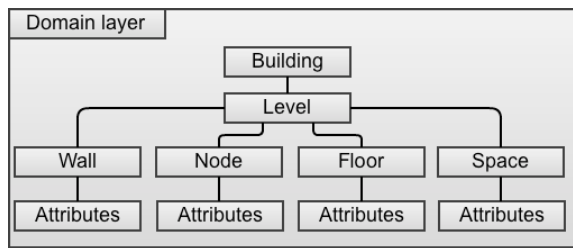


Figure 11. Domain layer

### **4.1.3 Validation**

One of the specificity of the CIMEDE modeler is to be able to conduct a lot of automated validations and processing based on the constructive technology requirements. There are two levels of validation: “by-design” validation and “post-design” validation. In the first case, the graphical user interface of the modeler itself constraints the user to some specific shapes. For instance, at drawing, only the two X/Y wall directions are allowed, only one set of bearing walls is maintained for all the floors, the wall sizes are matching predefined constrained dimensions, etc.

For the post-design validation, a specific tool has been developed. Its goal is to offer freedom to the user, allowing him to reach some non-coherent state during the modeling, avoiding him to fight with the interface when encoding a complex solution requiring multiple steps. However, at the end of the design, on demand, he can run a rule-based engine checking the conformity of his solution with the numerous CIMEDE design rules. For instance, assuming the maximum length of a floor element is fixed to 10 meters, the validation tool highlights every floor that doesn’t respect this rule.

## **4.2 BIM interoperability**

The development resources have been used on what really mattered in this research project: providing an internal building model and a modeller interface fully customized to the specific construction constraints of the CIMEDE [BCS]. At the same time, CAD tools users have a lot of expectations when it comes to features that should be included in a modelling tool. Amongst those expectations, we have the computations of the relevant plans of the project, like the generation of the floor and elevation plans, some printing capabilities, the production of photo-realistic artworks, the export in different formats, or the many possible evaluations one can run on a building. To provide answers to these legitimate functional requests, we have added one last single feature in our modeller: an IFC file export, IFC being an interoperable file format supporting the BIM paradigm which aims providing a complete data model for all the geometrical and semantic aspects of the building.

Following the definition of (Ashrae, 2009), the ambition of BIM is "to capture both physical and functional aspects of the building in order to generate an accurate model that is useful throughout the entire life of the

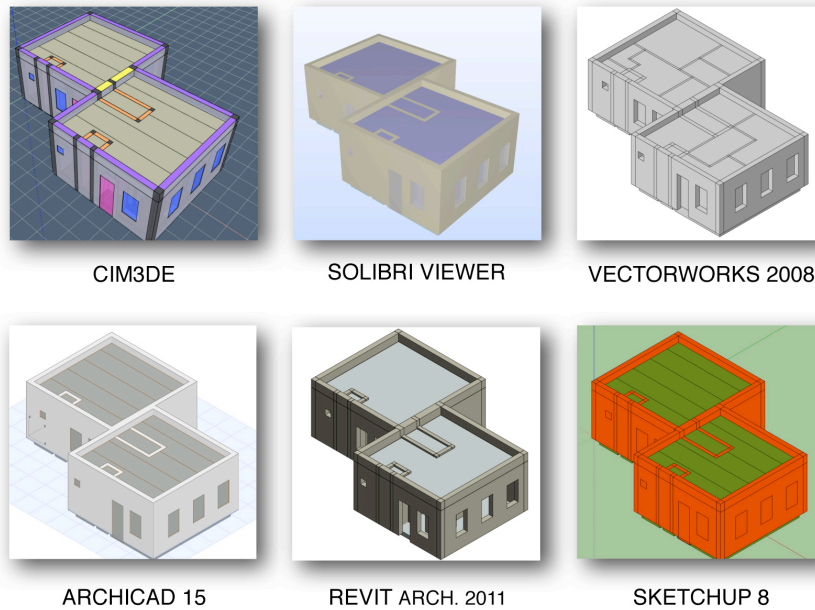
building, from initial design through occupancy and operation". Archicad from Graphisoft or Revit from Autodesk are examples of BIM software. Next to the commercial solutions, the IFC (Industry Foundation Classes) has emerged as the best open standard candidate for BIM interoperability (Liebich et al. 2011).

We had previous experience with IFC (Delfosse, Schrayen et al., 2012). In the CIMEDE project, we have evaluated how easy and useful an IFC exported was to develop. Our development strategy for the export module has been the following:

1. Build a rich CIMEDE model, having all the possible constructs of the software;
2. Build an equivalent model in a robust IFC-compliant tool. In this case, we have chosen ArchiCAD 15 for its good IFC support and because it was a good target candidate defined by the CIMEDE consortium. The goal was to specify all the constructs of the CIMEDE model in the ArchiCAD model;
3. Export the model from ArchiCAD into IFC;
4. Incrementally, we have reverse-engineered the IFC file and added the IFC export code into our modeler;
5. Incrementally, we have tested the IFC generated by our code, first in ArchiCAD only, then in many other IFC compliant tools.

Having previous experience in IFC, the only sensitive point we were identifying before this CIMEDE development was the representation of the exported geometries in IFC. Fortunately, the geometries of the CIMEDE elements were simple: only planar elements, mostly vertical and horizontal, with some possible sloped roofs. Also, we have seen from the ArchiCAD reverse-engineering that the observed IFC geometries were never more complicated than necessary.

This export module gave very good results when tested in 6 IFC compliant tools: ArchiCAD, but also Revit Architecture, Solibri, Ecotec, Sketchup or VectorWorks. The figure below shows a view of the original Cimedede model and the result of the IFC export in different commercial modeling tools.



*Figure 12. An IFC generated by CIM3DE and imported in other CAD tools*

## 5. CONCLUSION AND FUTURE WORKS

The work discussed in this paper presents the following main characteristics:

1. The development of a 3D modeler based on the standard components and constraints of the CIMEDE BCS;
2. An intuitive 3D modeling tool, based on the 2D architect drawings used at the preliminary stage of a conceptual process;
3. A generic and flexible data model, able to export a spreadsheet file dedicated to ERP applications useful for manufacturers;
4. An IFC export module, ensuring interoperability with the traditional CAD tools.

Using this approach, the system automatically takes care of a number of the modeling aspects. This leads to a tool that can much faster create projects compliant with the CIMEDE technology. For instance, when the modeling

of a given building in CIM3DE takes half a day, the same task requires at least a full day of work in traditional CAD software.

Future modules should be added to the system. An automatic link to the PEB (the belgian certifying energy assessment software) is envisioned, as it would spare the required re-encoding of the whole building project in the PEB software. Also, CIM3DE could provide more suitable outputs to serve the manufacturing of the building elements. The ultimate goal of CIM3DE is to automatically drive the manufacturing machines, while, at the same time, producing assembly plans of the building.

## 6. REFERENCES

- Ashrae, 2009, "An Introduction to Building Information Modeling", American Society of Heating, Refrigerating and Air-conditioning.
- Azhar, S., Hein, M. and Sketo, B, 2008, "Building Information Modeling: Benefits, Risks and Challenges", *Proceedings of the 44th ASC National Conference*, Auburn, Alabama, USA.
- Bogus, S.M., K.R.Molenaar, and J. E. Diekmann, 2005, "Concurrent Engineering Approach to Reducing Design Delivery Time", *Journal of construction engineering and management*, november, p. 1179-1185.
- Delfosse V., Schrayen J., Juchmes R., Leclercq P., 2012, "Some advice for migrating to IFC", *CAADRIA 2012*.
- Evergreen, 2009, *PrefabHouses, Maisons préfabriquées, FertigHäuser*, Köln : Mul
- Leclercq P., Van de Vreken A., Lorquet C., Dawans A. 2011, "CIMEDE:Architectural modeler for modular industrial building system", *Proceedings of CAAD Futures 2011*, Liège, Belgium.
- Liebich T., Adachi Y., Forester J., Hyvarinen J., Karstila K., Reed K., Richter S., Wix J., 2011, Industry Foundation Classes IFC2x Edition 3 Technical Corrigendum 1,. Available from <http://www.iai-tech.org/ifc/IFC2x3/TC1/html/index.>
- Nardi, B., 1996, *Context and Consciousness: Activity Theory and Human-Computer Interaction*, MA: MIT Press, Cambridge.
- Rabardel, P., 1995, *Les hommes et les Technologies. Approche Cognitive des Instruments Contemporains*, Armand Colon, Paris.
- Richard, R.B, 2005, "Industrialised building systems: reproduction before automation and robotics", *Automation in Construction*, 14, p. 442-451.