INTERNATIONAL DESIGN CONFERENCE - DESIGN 2016
Dubrovnik - Croatia, May 16 - 19, 2016.

# FUNCTIONAL SPECIFICATION METHODOLOGY FOR AN ARCHITECTURAL MODELER SUPPORTING A MODULAR CONSTRUCTIVE SYSTEM

C. Boulanger, A. Rahhal, V. Delfosse, C. Lorquet and P. Leclercq

*Keywords: activity theory, architectural modeler, software design methodology*

## 1. Introduction

This article presents a methodological approach which helped the software development of an architectural modeler that aims to support the life cycle of buildings with a modular construction system. The modeler assists the building composition processes, the production of useful documents like drawing plans, quantities and specifications, and the evaluation of the thermal performance of the building at any phase of its design. Conducted within a multidisciplinary team of architects engineers, computer engineers and ergonomists, this methodology is based on a constant interaction between the various skills, so that the different perspectives are systematically integrated into the decision process. The methodology is based on the analysis of three fields of functions (composition - production and evaluation) which are considered simultaneously through a theoretical framework stemming from ergonomics: Activity Theory [Engeström 1987].

At first, the software development context is presented. Then, the choice of the Activity Theory as a framework for the analysis will be motivated. The next section relates the multiple concepts of the methodology to each others, in order to identify the principles as well as the inputs and the outputs.

Each step in its implementation is then described, followed by the results in terms of the software interface. Finally, the discussion that will follow will consider the contributions and limitations of this methodological approach in the understanding of a system, in order to offer an efficient IT support.

## 2. Context of software development

The architectural modeler is part of a research project, funded by the General Administration of Economy and Employment of the Walloon Region. The project objective is to define precisely a modular construction system incorporating industrial technologies and innovative components for sustainable development.The first benefit of this system is its modularity that will allow not only an economy of scale associated to a certain level of standardization of the modules, both from a manufacturing and technical studies standpoints.The second benefit is an economy of time by being able to make the building in the factory without being dependent on the weather, and then mount it on-site, all by assembling architectonic innovative and efficient technical components.

To achieve these objectives, the project coordinator, has enlisted partners in the fields of engineering, acoustics, thermal and the LUCID-ULg for developping the software. The contribution of LUCID-ULg to the project includes the development of an computer assisted design software, oriented to composing buildings using various components and dedicated to the composition of an integrated building model, directly related to business concepts implemented in developing the concept.

## 3. Activity Theory as a guideline of the analysis

### 3.1 The decomposition of the activity

The first elements of the Activity Theory are issued from the work of Leontev that describes three levels characterizing an activity [Leontev 1974]. On the lowest level are found the operations, which are actions that have become automatic. Actions constitue the intermediate level. The actions are guided by one goal and are specifically linked to the mental representation of the result to be achieved, which requires the implementation of knowledge and expertise. Finally, the activity corresponds to the upper level and it is supported by a broader or more strategic target.
Since 1974, this model has evolved a lot to reflect the socio-cultural and contextual factors of the activity [Engeström 1987]. The last relevant model is the one of Engeström that describes an activity structured by its seven basic components influencing each other, as shown in Figure 1.

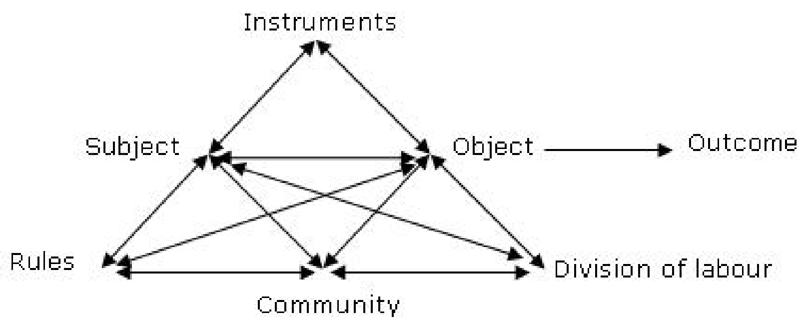


**Figure 1. Schematic of the activity according to Engeström [1987]**

The *subjec*t carries out actions to reach a goal, the *object*. To achieve this, it uses artefacts, *Instruments*, that are means, which may be material or not. In this activity, the subject is related to other people, the *community*. Relationships are influenced by *rules* and work organization, *division of labor*. The *outcome* represents the result of the activity.

### 3.2 The usefulness of this decomposition in our methodology

This structural model is one of the activity analysis tools that take into consideration the context and enables us to understand the role of artefacts integrated into social practice [Kuutti 1996], [Nardi 1996]. Relying on The Activity Theory can improve design approaches for establishing a link between the description of the activity and the design of artefacts [Bjørkli et al. 2007]. These authors used this theory to design navigation support tools and concluded that using Activity Theory can contribute significantly to the design process, but by providing solutions that include several alternatives and not a single one. Rabardel [1995] redefines the term artefact into the term instrument to translate the fact that artefact becomes an instrument when it is involved in a use. Artefact is not individually consider yet, but is defined by the relation between the subject and the object of activity by means of the instrument. This processes is named "Instrumental Genesis" [Rabardel 1995].
The peculiarity of the activities we have described in this project is that they are not observable. Indeed, the construction system is under study and how to design and build the building lies in the vision of how it could happen. It is therefore not an analysis of an already existing activity. It is considered as an activity in which the tool will take an integral part. We have to anticipate the future activity. We used the model of Engeström to describe the activity underlying the software, based on the description of the elements of the model. In addition, the base model of Leontev with the three levels of activity - the activity itself, the actions and the operations - is also helpful. Indeed, in a software development activity, the actions to be implemented can be considered as the software functions, while operations are all the manipulations to perform using the various buttons and interactions to realize the action. The action is guided by a goal, while the operation is the means to achieve this goal.

How these two models of the Activity Theory are used is specified in the description of the methodology of the next section.

## 4. Description of the methodology

We crossed the model of Engeström with the structural model of Leontev. Engeström's model serves as a framework for the analysis of needs, and that of Leontev, organizes the stages of data collection. Thus, our methodology is structured in three steps: analysis of the contexts through the activities; analysis of the functions through the actions and analysis of the operations. By describing each action in each activity, using the concepts of the model of activity, our goal is to define how the operations can be organized to carry out the actions and so to define how the software will be used in each phase.

### 4.1 Step 1: analysis of activities: the context

Knowing that the software should be useful during the building life-cycle, we consider each phase of the life cycle of a building as a different activity, like in the upper level of Leontev model, each activity is oriented to a specific result. Each activity is described according to the model, from general elements, describing the context of the activity, i.e. who are the actors of the activity - *subject*; who are the stakeholders and the partners - *community*; what result is expected - *outcome*; and how the work is organized to achieve the result - *division of labor*. Model elements considered in this step are shown in black in Figure 2.

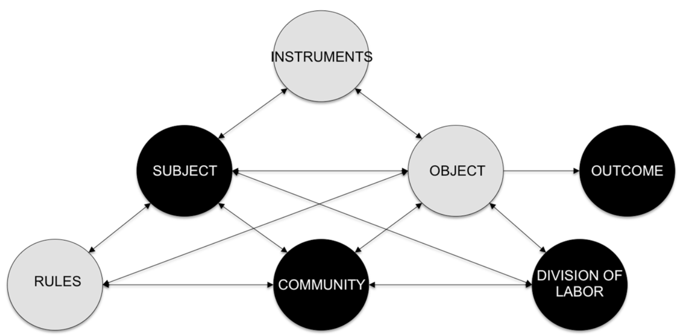


**Figure 2. Schematic description of each activity**

### 4.2 Step 2: analysis of actions: functions

Step 1 allows us to describe the context of each activity. Step 2 consists on describing more accurately the actions that are being implemented in the context of each activity. The actions correspond to the intermediate level of the Leontev model and can be assimilated to the general functions of the software. Every action aims to achieve an object. The description of the actions to be implemented in the activity is defined by the object of the action, *object*, and the rules and constraints that guide, *rules*, as shown in black in Figure 3. This step analyzes more precisely each object of the activity and the rules to be considered by the subject to achieve this object.

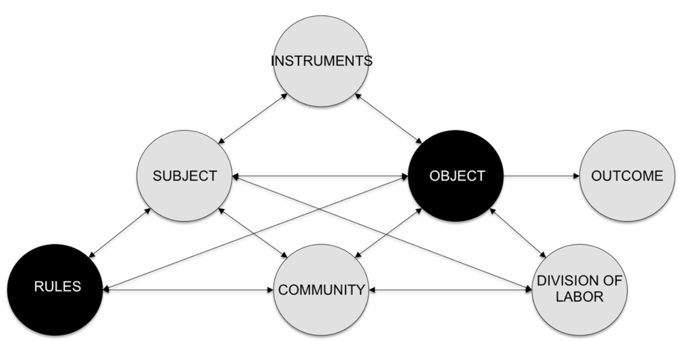


**Figure 3. Schematic description of each action in the context of the activity**

### 4.3 Step 3: operations specification

Finally, each action is implemented by operations, corresponding to the lower level of the Leontev model. The operations are the concrete implementation of the actions using the instrument software. Therefore for each action in each activity, this third step consists to specify the executable operations to realize the action. This corresponds to the third level of analysis, which complements the global description of the activity. Figure 4 shows this third level of analysis which focuses on the instrument and the specific operations carried out through it.

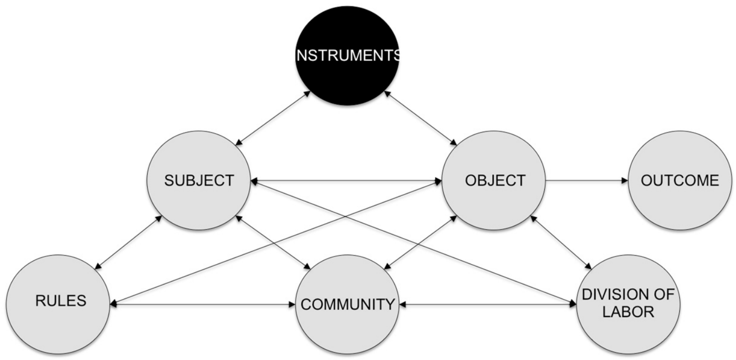


**Figure 4. Schematic description of each operation to achieve the object of the action**

## 5. Implementation methodology

### 5.1 Step 1: analysis of activities: the context

The place of the software in the construction project was enacted at the first meeting of the project, setting the specific objectives of the LUCID-ULg team. The software operates in most building design and construction phases, as shown in Figure 5. A building project shows a cycle of four phases.

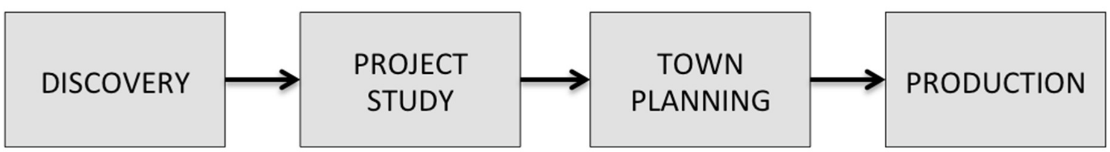


**Figure 5. Building design and construction phases**

*5.1.1 Discovery*

This activity is the first step in which the software is used. This activity consists of modeling quickly the main elements of the project using the basic components. A schematic view of the building is provided to the client, as well as outline plans and a pre-estimate calculation of the construction budget. In this context, the activity aims to interest the client in order that he designs its building according to the construction principle (*outcome*). The *subject* takes the role of a sales representative. He is related to a potential customer, *community*. The organization of the activity, *division of labor*, consists of various exchanges between the sales representative and the customer. It's about the customer who discovers the constructive concept and the sales representative who discovers the client's initial requests.

*5.1.2 Project study*

This activity is the study of the construction project by an architect, and the result is a detailed architectural design of the building (*outcome*). In this activity, the *subject* is an architect who communicates with design engineers (*community*) in order to validate its structural, functional or any other assumptions of the project. He also exchanges ideas with the client (*community*) The *division of labor* can be described as iterative cycles of exchanges between the architect, the client and the design engineers, each cycle leading to a composition, a modification or an adjustment of the project.

*5.1.3 Town planning*

This administrative phase seeks to obtain an authorization to construct the building. It requires the integration of all building context variables and the implementation of urban plans respecting some specific conventions. The *subject* is the architect who has to realize the plans for the building permits (*outcome*), that he can communicate to the city planning department of the municipality (*community*).

*5.1.4 Production*

Finally, this phase is the real construction of the building, according to the concept proposed. It involves the production of components, their assembly in the factory and finally their on-site assembly.
The component-based concept implies a change in the types of data required for completing a task. The manufacturer (*community*) will have a list of all components with their technical data, provided by the contractor (*subject*). The specifications of each component are standardized and have been studied in advance in the research project. As a consequence such study won't be required anymore when working on a specific building project. The contractor shall not dispose of traditional execution plans, but the assembly drawings and the mounting plans for these components, as well as the position plans of the various points of service (outcome).

### 5.2 Step 2: analysis of actions: functions

After having defined the contexts by mean of activities, we focuse on actions that have to be realized in the context of each activity. In this project, we know that the main actions to be supported by the software are of three types: building composition and representation actions, documents production actions and building evaluation actions.

*5.2.1 Composition, production and evaluation actions*

Composition actions in the context of the constructive concept, consist in placing or moving one or more components or to replace or swap components. A component is then the object of composition actions. The components are defined as elements for which the underlying composition can not be modified. If multiple variants of a same component had to be proposed, they would be represented as different components in the concept. For example, the interior wall element is available in three different components: an interior wall having medium acoustic performances, an interior wall having high acoustic performances and an interior glazed wall. The precise composition of the first interior wall component (with medium acoustic performance) was studied in the research project and is already fixed. It can not be changed depending on the project. The architect doesn't specify the materials to be used to realize the interior wall in order to reach the performance he seeks, he will just directly choose the quality of performance of the interior wall. The chosen element features components that meet these requirements. The study of the project is therefore an architectural study that combines optimization of space, aesthetics and a good evaluation of the required performances, depending on the customer's specific needs. Technical studies are part of the concept and they are no longer deployed in each individual case.
We have identified various types of components. They can be divided in two groups : the components that can be usefully analysed and shared between multiple different projects, and the components that are specifically added for a given single project. The first group is the heart of the constructive system and therefore is of particularly interest for us. We identified six families of such components: modules (represented by the metal structure), facades, doors, roofs, terraces and balconies. These types of static components come in several models that are selected from a catalog. "Special equipment", like electricity, heating, or air conditioning, are part of the variable components. While the research defines the principles and technologies that will be systematically used in the design of buildings, their implementation is very specific to the architecture and the purpose of the building. The components for 'interior wall' share characteristics of these two groups. Like the first group, they are available in various

versions which differ in terms of composition and thickness, depending on the desired acoustic performance. But they also vary in length and shape depending on its position in a specific project.

Production actions consists of delivering documents. A document is then the object of production actions.

Evaluation actions are those that allow a feedback on the quality of the building concept. Three types of avaluation are proposed by the research project consorsium : energetic, daylight and quantities evaluations. A specific result of an evaluation is then the object of evaluation actions.

In order to accurately define the possible modeling actions, we have analyzed each type of actions (composition, production and evaluation) and their objects, into each activity. We focused then on the rules governing the action related to the component, i.e. the rules which allow to choose and position the component in the project.

### *5.2.2 Composition, production and evaluation actions in the contexts of activities*

In the context of discovery activity, composition actions are necessary to allow the client to discover the concept using his own building project. No contract has been realized yet. So, the demonstration can be limited to basic components, as modules and facades. Production actions aim to deliver an idea of budget and a first 3D representation of the building project. Evaluation actions aren't significant at this step.

In the project study activity, the building project has to be very detailed. Then, composition actions imply all types of components (modules, facades, interior walls, doors, wall-coatings and point of service), but integrate environnement too. Production actions aim to deliver detailed plans and budget. Evaluation actions are necessary to evaluate the quality of the building concept in terms defined by the project consortium : energetic, daylignt and quantities evaluations.

In the "town planning" and "production" activities, composition and evaluation actions can occur but aren't the more important in these contexts. Main actions are production actions. Standardized plans are the object of the first activity. According to the promoters of the research, the layout of data according to these conventions is easily made in traditional CAD tools. So the software must be able to export its models into these external CAD tools. The software supports the export of its projects into a DXF file format. DXF is a major interoperable format supported by most of the CAD applications. List and technical data of elements, assembly drawings and mountain plans are the objects of "production" activity.

### *5.2.3 Types of rules*

Three types of rules have been defined: architectural rules, concept rules and the underlying rules of the concept. Their influence on the activity is shown in Figure 6.

1. Architectural rules are those that come from the rules of the art in architecture: They are the skills of the software user while designing the project, the architect. These rules therefore influence the relationship between the subject and the object.
2. The concept rules are those related to the construction system itself and they allow or not to perform certain compositions. Therefore, they interfere with the actions of the user by preventing him from making compositions that are not in adequacy with the constructive concept. They include all the geometrical data of the components, the principles of connections between components of the same or different families, and the parameters that can be assigned to the various components, such as a coating type or a type of acoustic performance, for example. Hence, these rules influence how the subject uses the instrument, the software.
3. The underlying rules are also related to the construction system but they are automatically inferred in the project without affecting the user's actions. They correspond, for example, to choosing the type of the building structure, while taking in account the number of floors that compose it. These rules act directly on the object in the software without interfering on the subject ations.

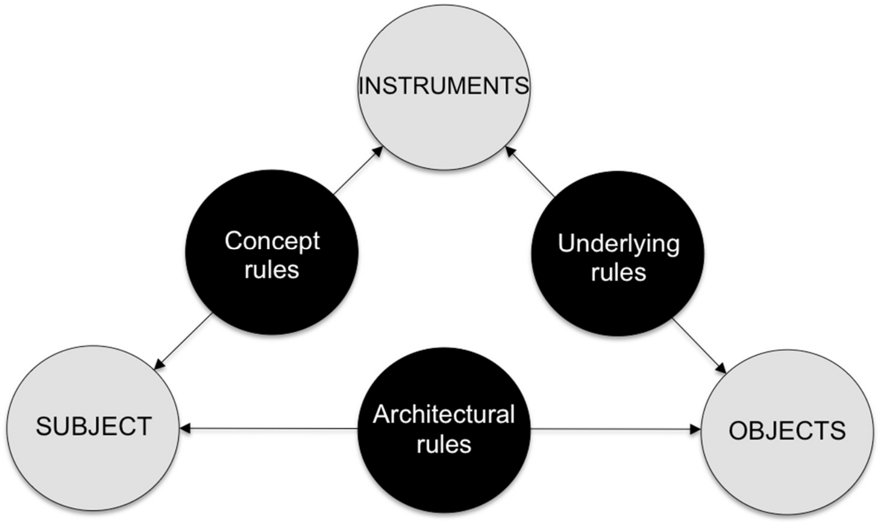


**Figure 6. Influence of three types of rules on the activity**

The software organization for the modeling actions, is based on the decomposition into types and then into families of components, on one hand, and on the integration of concept rules and the underlying ones, on the other hand. As for the architectural rules, they are not integrated in the internal model of the software since it is designed to support a constructive concept and not to replace the role of the architect, who retains control of the architectural concepts. However, they are at the heart of the software interface because the architect must be able to be implement them. Thus, if the software does not aim at providing architectural solutions, it must assist the user to evaluate the different possible solutions by providing sufficient feedback.

The means used by architects to assess their project are manifold: check the dimensions and the surfaces, check the degree of sunshine, view the 3D shape volume, visualize the building in its global context. These means therefore seem essential for optimum use of the software.

Actions of production of specific plans and budget were also analyzed by defining precisely the expected object of the action, its format, the desired details and the rules for determining the precise contents (calculation specifications, standards export), according to the concept.

We were able to list the main functions of the software:

1. Functions for choosing and placing components, governed by specific rules for each component;
2. Functions for component parameters specifications: assigning a coating to the walls, facades, floors and ceilings;
3. Project evaluation functions: evaluation of daylighting, 3D visualization, measurement tools, import of environmental context;
4. Documents production functions: plans, sections, elevations, assembly drawings, mounting drawings, simplified and accurate project budget, governed by the concept.

### 5.3 Step 3: operations specification: the interface

Step 2 allowed us to list the functions of the software. Among these functions, the first two groups correspond to the design of the project, requiring the expertise of the user. The other functions require a proper integration of constructive rules in the heart of the internal system, but no action from the user is needed, except executing the command for the desired function. This consideration involves essentially an interface dedicated to the composition functions (Figure 7), while the other functions result from commands, rather than from user actions.

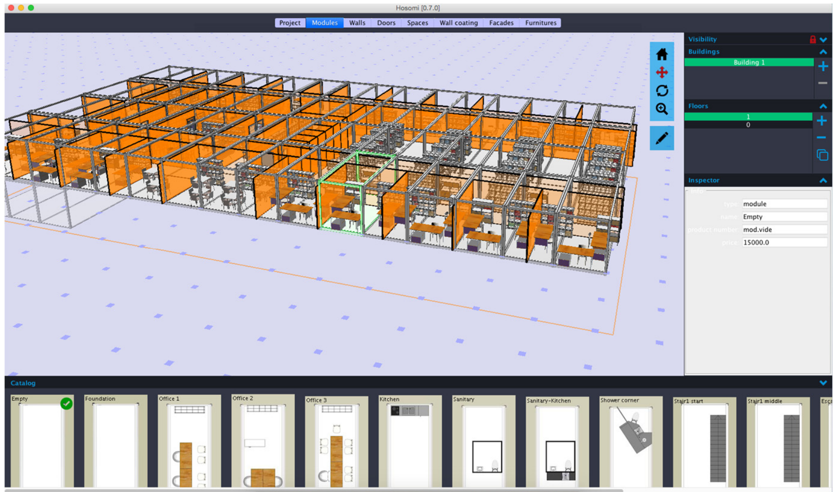

Figure 7. Main interface dedicated to composition functions

Stage 3 is the specification of the operations, i.e. the means to achieve the actions via the software. It consists in the definition of the shortest way to perform each action. For example, for choosing a module, it is necessary to access the different displayed modules and their characteristics; for placing a module, it must be taken, oriented and finally placed. A similar decomposition is performed for each action of each component in order to identify the most appropriate operating modes for each one, while maintaining consistent identical operating modes.
For the component selection functions, a model catalog has been developed for each static or semi-static (interior wall) component. Each catalog item is represented by a thumbnail illustrating the differences between the items (Figure 8).

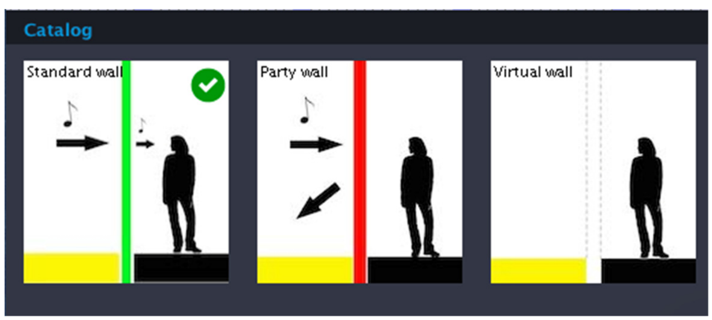


Figure 8. Thumbnails of interior wall illustrating the characteristics of each wall

Positioning tools have been implemented for the component placement functions, such as translation, rotation or zoom. Depending on the component family or activity, the relevant positioning tools are displayed, using coherent icons. Figure 9 illustrates positioning tools of doors (on the left) and positioning tools of interior walls (on the right).

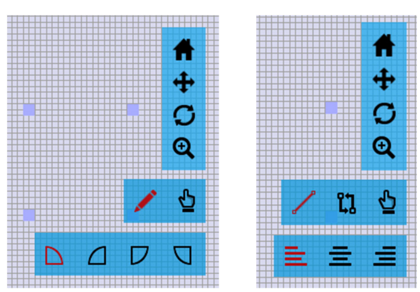

Figure 9. Examples of positioning tools adapted to activities

The functions of component parameters specifications are grouped in an interface widget called 'inspector'. The properties displayed are the ones available for the selected component, so it is always possible to select the component and update its values and characteristics. Figure 10 illustrates the inspector for facade.

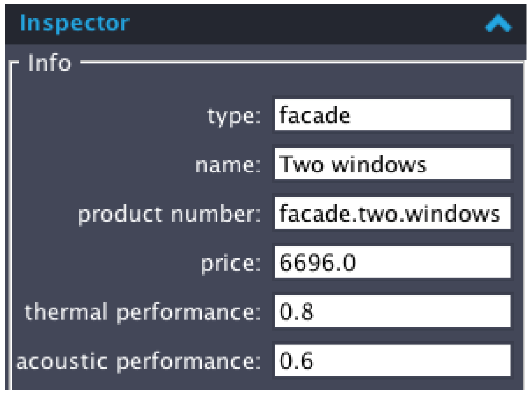


**Figure 10. Inspector for façade**

The evaluation functions of a projet are very specific, such as energy or sunlight evaluation. They are available via the main menu. On the contrary, systematic evaluation functions, such as 3D visualization or measures verification, are present immediately on the main interface. The documents production and data export functions, are also accessible via the main menu. Figure 11 represents these two level of accessibility depending on type of evaluation.

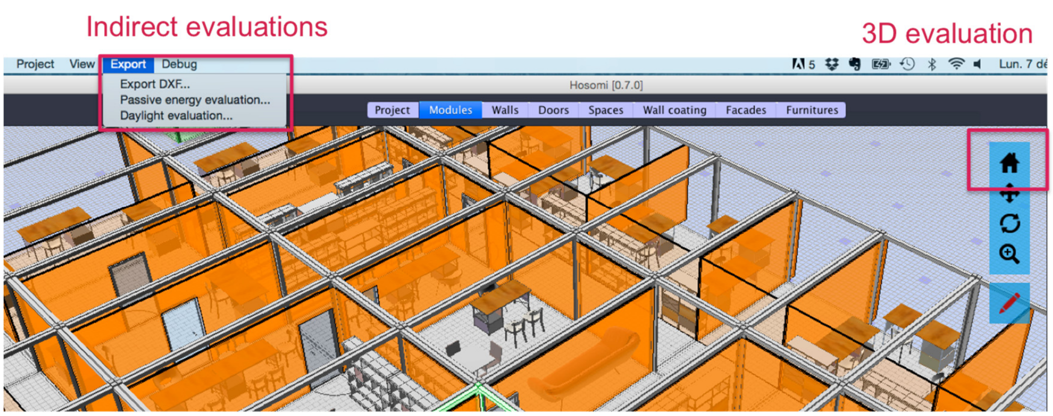


**Figure 11. Accessibility to different types of evaluations**

## 6. Contributions and limitations of this methodological approach

The decomposition into activities led to the identification of the different contexts useful when using the software. It also led to the realization that similar actions are performed by actors with different profiles and skills. Therefore, these actions must be performed regardless the skill of the user. Other actions require specific skills related to the field of knowledge. These are not guided by the software, but remain free so that the specific skills of each user can be expressed. The analysis of activities helped us to establish that the conception functions, requiring user actions, are the most important ones. As such they guided the overall design of the interface. For every action, the identification of rules for its implementation has helped to clarify which design decisions were under the responsability of the concept and which ones were the responsablility of the user. The listing of these rules has allowed to collect the necessary information to be encoded in the software data model. Finally, the decomposition into operations allowed the precise definition of the operation modes and the software interface.

If this methodological approach leads to conventional functional specifications, it has the merit of providing a framework for data collection, allowing to stay continuously connected with the execution context of functions. For example, the fact that composition actions are executed in different contexts by different actors, requires a suitable interface that works with users having different skills. Similarly, the analysis of rules for the implementation of actions shows that the architectural skill remains the one of the user when the software should focus on integrating the specific constructive system rules.

Within this software development project, this method has given a guidance and a structure in understanding the issues and challenges of the underlying activities. Our aim is to develop a flexible design space that enables future users to work with their own structure. The software reflects the constraints of the modular concept, but remains flexible in the order of realizing activities. Software evaluations have been realized with architects and non architects. These evaluations have confirmed that the software enables differents types of working and organizing activities but have shown that navigation tools have to be improved. Further work consists of evaluate the software in use within the first prototype of a modular building based on the developed concept. This evaluation in a real situation will permit to test and attest that the conception processes of the software continues through his use with the concept of Instrumental Genesis [Rabardel 1995], by wich users will adjust artifacts (or the software) to their specific conditions and needs.

**Acknowledgement**

This article has been written in the context of a project funded by the Walloon Region.

**References**

*Bjørkli, C. A., Røed, B. K., Bjelland, H. V., Gould, K. S., Hoff, T., "La théorie de l'activité comme aide à la conception d'équipement de navigation", Available at: <http://www.activites.org/v4n1/v4n1.pdf>, Activités, Vol.4, No.4-1, 2007, pp. 170-178.*

*Engeström, Y., "Learning by Expanding: An Activity-theoretical Approach to Developmental Research", Orienta-Konsultit, Helsinki, 1987.*

*Kuutti, K., "Activity Theory as a Potential Framework for Human-Computer interaction research", In: Nardi, B. (Ed.), Contexte and Consciousness: Activity Theory and Human-Computer Interaction, Cambridge, MA, MIT Press, 1996, pp. 17-44.*

*Leontev, A. N., "The problem of activity in psychology", Soviet Psychology, Vol.13, No.2, 1974, pp. 4-33.*

*Nardi, B., "Contexte and Consciousness : Activity Theory and Human-Computer Interaction", Cambridge, MA, MIT Press, 1996.*

*Rabardel, P., "Les Hommes et les Technologies. Approche Cognitive des Instruments Contemporains", Paris, Armand Colin, 1995.*

Vincent Delfosse, Research Engineer
University of Liege, LUCID-ULg
Quartier Polytech 1, 4000 Liege, Belgium
Email: vincent.delfosse@ulg.ac.be