

# A Model-Driven Approach to design two Assistance Tools for Building Construction Coordination

S. Kubicki, G. Halin & J.C. Bignon

*CRAI, Research Centre in Architecture and Engineering, Nancy, France*

A. Guerriero & D. Hanser

*CRP Henri Tudor, Public Research Centre, Luxembourg*

**ABSTRACT:** This communication describes an area of research into coordination of the collective activity in the building construction stage. The theoretical approach tries to identify coordination mechanisms existing during building construction. We suggest hypotheses on assistance methods to develop taking into account the cooperative specificities of AEC projects. The methodology of tool development is centered on a model-driven approach (MDE). In this perspective we describe the theoretical framework of MDE and similar works of collective activity modeling. Then we present two coordination assistance tools under development. This description follows the scheme: 1) position regarding coordination-specific activities, 2) design and model-use for software development, 3) functionalities of the tool and 4) validation perspectives.

## 1 INTRODUCTION

The AEC sector is an industrial field which is distinguished from others by some particularities. The composition of teams is ephemeral and heterogeneous. The building as a product has to face many constraints such as functional, technical, economic, esthetic constraints varying from one project to another. Time development of a project is sequential, i.e. there are working periods contrasting with waiting periods.

Cooperation during building construction consists essentially of the coordination of the independent actors' teams, which don't have a global "vision" of the project context.

We identify firstly organization forms and coordination mechanisms in building construction projects based on a theoretical findings of XX<sup>th</sup> century scientists. This analysis allows us to suggest hypotheses about coordination assistance tool requirements.

We then present the methodological framework guided by models that we use to analyze and to develop new tools.

Finally two assistance tools are suggested and described. The first one is a meeting report tool, assisting writing and diffusion of this document, and beyond distribution of coordination information. The second one is a multi-view interface representing the project's context in multiple interlinked views.

## 2 BUILDING CONSTRUCTION COORDINATION

### 2.1 *Organizations and coordination, a theoretical background*

XX<sup>th</sup> century theories on organizations focus essentially on their formal structure. Studies by Henry Mintzberg appear especially interesting when it comes to distinguishing between organization forms (Mintzberg 1978). We retain here three major forms: "hierarchical" organizations, "transversal" organizations and "adhocratic" organizations.

"*Hierarchical*" organization covers traditional enterprise forms identified in theories of scientific management (Taylor 1911). It's characterized by a bureaucratic organization (Weber 1921) and managed by organization charts.

When organization becomes more complex and dynamic, there is more standardization of methods and process. In some cases these organizations cover numerous project contexts. We then talk about "*transversal*" organizations.

"*Adhocracy*" concept introduced by Toffler covers a more "democratic" vision of collective work (Toffler 1970). Thus decisions should be distributed between actors and personal strategies should be preserved.

The characteristics of coordination are related to these three main forms of organizations. Mintzberg distinguishes essentially between three coordination mechanisms (Mintzberg 1978):

- In direct supervision one person is responsible for the work of others. This person has to plan the process and to communicate it explicitly to the actors,
- Standardization appears when coordination of the different workers is incorporated in the program in early design stage. Communication need is then reduced,
- Mutual adjustment ensures work coordination by the way of informal communication.

Schmidt's approach of coordination mechanisms is a little different and aims at computational coordination mechanisms in CSCW (Computer Supported Cooperative Work) system design. According to him a coordination mechanism is composed of (Schmidt & Simone 1996):

- A protocol: "an integrated set of procedures and conventions stipulating the articulation of interdependent distributed activities",
- An artifact: "a permanent symbolic construct" which can be static or dynamic and thus varying with the state of the protocol. It represents a generally quite high level of granularity of these protocols.

According to Andersen who studies "dimensions of coordination" [An00], we can find two principal communication modes: oral and artifact-based. Oral communication is based on a knowledge background shared by the actors involved in coordination activity. Then, interaction is based on the "focus" or "object of coordination". This type of communication is found in relatively "simple" domains. Artifact-based communication appears in more complex cases (numerous actors involved, large dimension of the project, high variability). Its advantage is that it is more simple and comprehensive for the actors through the use of representation specific to their domain of expertise.

This analysis is closer to the one proposed by Godart (Godart et al. 2001), which distinguishes between two types of coordination: implicit (e.g. mutual adjustment) and explicit (e.g. application of procedures described in rules and contracts).

Moreover we can make a link between these coordination modes and specific organizations (Figure 1). In the following parts of this paper we focus on building construction activities.

## 2.2 Coordination modes during building construction

In "hierarchical" organizations protocols and artifacts are the main methods and tools used to coordinate the teams of actors. This is due to the fine and explicit definition of tasks and processes. Direct supervision is the principal form of coordination.

Schmidt distinguishes between artifacts of which the state is static and those of which the state changes according to the state of the protocol (Schmidt 1997). In the building work phase we can find two principal coordination artifacts:

- A "static" artifact: the planning of the building work which describes the sequence of activities linked to the construction of the building,
- A "dynamic" artifact: the building meeting report which reflects the state of the building work at a given moment and determines a shared context necessary for various actors to perform their activities.

In "adhocratic" organizations oral communication is very often used. Heterogeneous actors grouped for the realization of a task or activity use "mutual adjustment" to coordinate implicitly. Artifacts furnished by hierarchy are used to better understand problems and to solve them in coherence with direct supervision. During building construction this coordination form appears for example when two contractors coordinate themselves on a specific task (e.g. informal meeting or phone call).

In "transversal" organizations, standardization of skills or results is the main mechanism of coordination. This is an organization form largely used in cases of inter-project interactions (actors working on many projects at once). Standardization implies fewer needs of communication (oral or artifact-based) and a medium level of explicit/implicit interactions. In the AEC sector this organization form describes the work of actors implied in numerous projects.

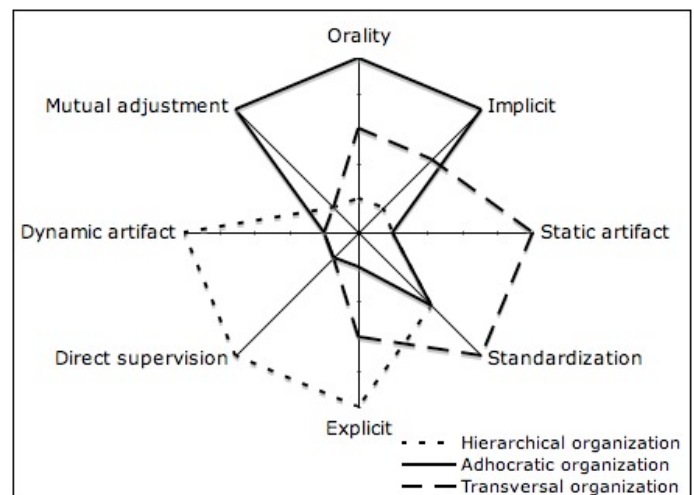


Figure 1: Organizations and characteristics of coordination

### 2.3 Hypothesis of IT assistance strategies

The characterization of these forms of coordination in the AEC sector allows a better understanding of the functioning of these organizations in this domain. It also allows us to better appreciate particular characteristics of each coordination mode to propose new adapted and functional tools.

In direct supervision, tools can support the production of coordination artifacts (writing assistance). They can also encourage diffusion to actors and information consultation (e.g. personalization of information according to the user).

As far as mutual adjustment is concerned, the analysis of professional practices leads us to the finding that information traceability is difficult. Indeed the coordination is essentially based on implicit interactions. From this point of view we recommend tools improving comprehension of the cooperation context and with which the user could navigate in the context. For that it is necessary to synthesize information and create essential relations between concepts to make them easier to understand.

## 3 MODEL DRIVEN APPROACH FOR TOOL AND HUMAN COMPUTER INTERACTION DESIGN

In this part we will describe the methodological framework that we have set up in order to guide the development of specific coordination assistance projects.

### 3.1 Model Driven Engineering: a unifying approach

Our approach is based on model development, steering both domain analysis and tool engineering (see part 3.2). This method is largely inspired by existing methods in the software engineering domain.

Since 2000 the Object Management Group has developed an approach called Model Driven Architecture (MDA) for software systems development (Soley & OMG 2000). Their objective is to define a framework of certified industrial standards (MOF, UML).

In parallel, the Model Driven Engineering (MDE) research area is an evolution aiming to unify different technical spaces (XML, ontology etc.). MDE does not focus only on one technology, it is an integrative approach (Bézivin 2005).

Concretely MDE recommends use of *meta-models* to define *domain languages*. *Models represent real systems*. Each model has to be *conformed* to its meta-model (Favre 2004).

The *Transformation* concept is a central one. It allows the models to be productive. A transformation is itself described with a model.

### 3.2 MDE approach: our motivations

Model engineering methods retain our interest for different reasons:

- First MDE allows us to express AEC domain knowledge and to confront it with software development specialists,
- In tool development, the well-defined meta-model architecture favors future interoperability between tools,
- Tools themselves will be easily adapted to the user in the particular AEC domain. In fact the user could modify the model of a tool and then specialize it relatively to his particular everyday needs,
- Finally the development of tool models and their visualization mode models allow us to define correspondences between AEC domain concepts and visualized concepts of a specific tool.

### 3.3 Modeling cooperation context

We use a three-level (M2, M1, M0) architecture inspired by the MOF (OMG 2000) to model cooperative activity. It enables us to design specific cooperation models for developing AEC-specific tools.

#### 3.3.1 Cooperation models: a state of the art

Some other projects have suggested activity models. DARE project (Bourguin 1999) focuses on the design of a CSCW tool for software development. The suggested activity model aims to allow the user to customize his environment to his specific task. This model is largely inspired by the Activity Theory (Engeström 1987) and articulates the concepts of *task/sub-task*, *object*, *subject/community*, *tool* and *role*. The objective is to develop a tool integrative environment. From this point of view activity modeling allows the definition of tools and views necessary to the developer to accomplish his tasks.

Another reference in activity modeling is the Rational Unified Process (RUP) developed by Rational Software Corporation. It suggests an iterative software development process. It aims to furnish to development teams an adaptable environment guiding the collective process. The concepts of the RUP model (Kruchten 1999) are closer to those expressed above. This model is centred on *processes*, decomposed in *stages* and *iterations*. The process is executed by a *role*, concerned by *artifact(s)* and uses *tool(s)*. We noted finally the *guidance* concept used to define guides for the user (e.g. associating a set of tools and artifacts to a process executed by a role).

#### 3.3.2 Our cooperation meta-model (M2)

Our proposition consists of defining a relational cooperation meta-model (Figure 2) that takes into account the existing relations between the elements of a project. This meta-model will then be instanti-

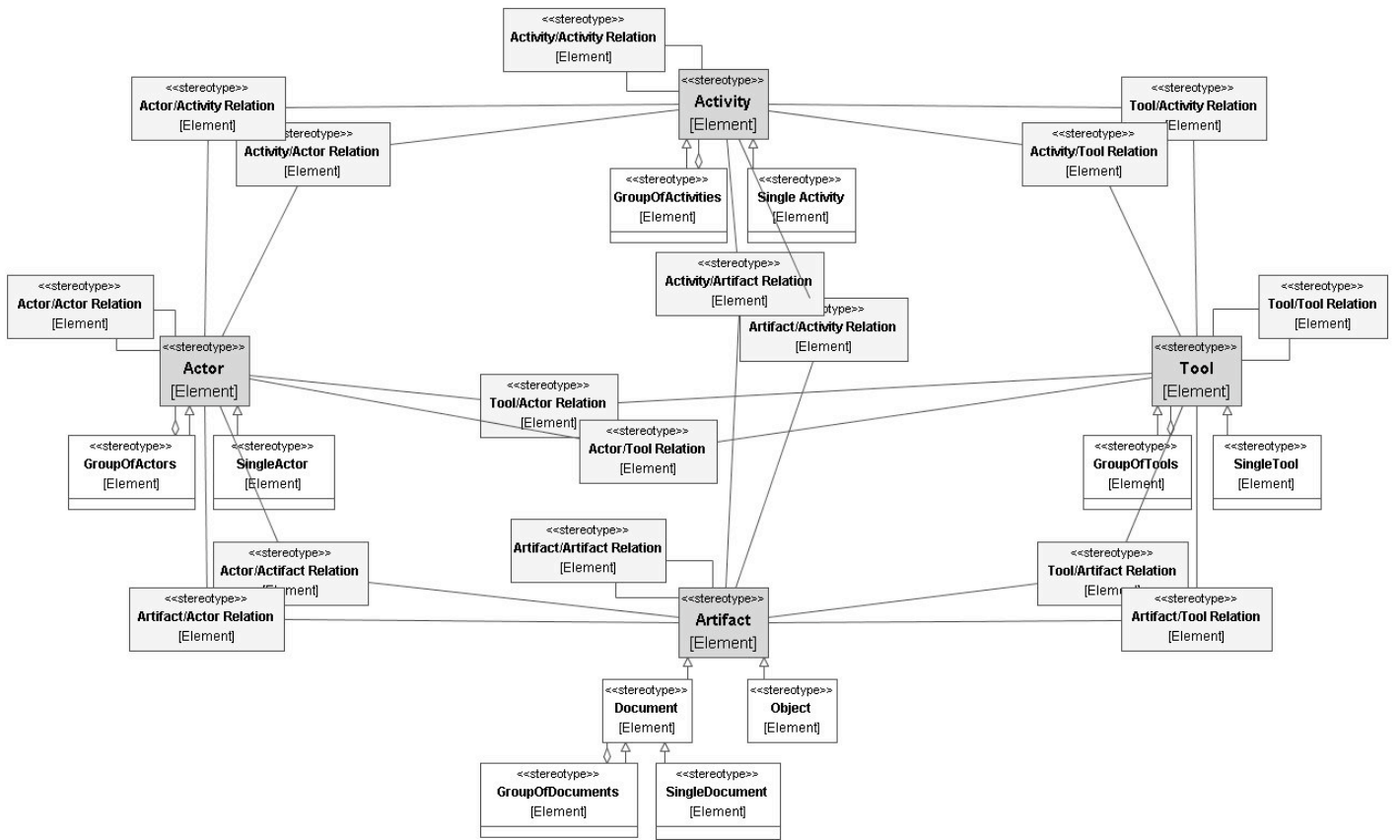


Figure 2: Cooperation Context Meta-Model extract (M2)

ated in domain-specific models (see part 4.2). The context of cooperative activities has to represent relationships and interactions between the actors, their activities, the artifacts and the tools they use:

- *Activity* (M2): the activities inside a project have several “scale” levels: project, phase, and task. They should be explicit or implicit,
- *Actor* (M2): in a project, each actor has a limited capacity of action and restricted decision-making autonomy. The actor acts inside the activities that constitute the project and keeps up a relationship with the environment,
- *Artifact* (M2): The generic concept of artifact describes any piece of information or other “thing” manipulated, used or produced by actors in an activity (Kruchten 1999). It could be a document which represents a professional “deliverable” part of a contract. It could be also a “model” of the object to design,
- *Tool* (M2): Tools are a kind of resource needed to run a process. Their availability for a user could be defined in his operational role in an activity. Tools use one or several visualization modes. These visualization modes are defined by models which describe the elements displayed,
- *Relationships* (M2): a relationship identifies a type of link existing between two elements described above.

We will now describe two propositions of tools based on this double theoretical background of building construction coordination (section 2) and activity modeling (section 3).

## 4 “DIRECT SUPERVISION” COORDINATION ASSISTANCE

The tool developed in this section is intended to centralize exchanges around the meeting report. It essentially supports direct supervision in hierarchical organizations. It is the typical situation of building work, where the architect writes a meeting report with particular points to be adjusted.

### 4.1 Objectives

The meeting report plays a fundamental role in the activity of construction. It is written after each meeting, it describes all particular points on the building site and then it is diffused to all concerned actors. This document consequently constitutes an important piece which allows, if necessary, the clarification of certain conflicts.

The tool we propose has been developed as part of the project Build-IT<sup>1</sup>. It is based on two principal objectives: firstly, to assist building report writing and, secondly, to accelerate document consultation. This prototype has been developed on the basis of exchanges between about twenty people of the AEC sector (architects, engineers, owners, contractors)

<sup>1</sup> “Build –IT” is a project which has been developed at the Henri Tudor Public Research Center in partnership with the CRTI-B (Information Technologies Resource Center for the Building). This project aims to reinforce electronic cooperation in Luxemburgish AEC sector.

and the analysis of about one hundred documents resulting from real building work.

## 4.2 Model instantiation

After having analyzed documents transmitted by professionals we have established a model (M1), instantiation of the cooperation meta-model developed above in section 3.3.2. This instantiation focuses on the meeting report. This concept is introduced as a *document* composed of specific sections.

Seven following sections compose the report:

- Reference: information allowing the identification of the meeting report, the meeting following which the report has been written, the writer, and the building work to which it corresponds,
- Presence and distribution list,
- List of generalities, information relative to weather and workforce on the building site,
- Notes, information for all the actors,
- List of remarks regroups particular points identified during inspection of the building site and the person in charge. These remarks can be associated to a picture,
- Progress describes the state of the building work at the time of the meeting, and eventual delays,
- Diary: information relative to the next meeting.

## 4.3 Build-IT Tool functionalities

Three types of functionalities have been developed in the meeting report prototype: functionalities about writing, about consultation of the meeting report and functionalities about reaction to a remark.

### 4.3.1 Writing functionalities

Seven forms corresponding to the 7 identified sections above guide the writing of building report. The writer can complete these forms and then distribute the document. The latter can be consulted in two forms: on the one hand, a static form, by generating a pdf document and on the other hand, a dynamic form, by consulting an application or a dedicated web interface.

### 4.3.2 Dynamic consultation functionalities

The tool offers the possibility to filter information according to the use and the actor (filter on person in charge, building element, grouping of building elements...). The search in three levels allows us to gradually restrict the field of search: a first search level within various current construction sites, a second search level inside a construction site and finally, the last one, inside the meeting report. Thus the consultation of documents, often a heavy task in current practice, is particularly simplified. This search functionality will allow the user, with activation of filters, to identify more simply e.g. all the remarks linked to an organization or all the urgent particular points... This type of information is really difficult to find in paper documents.

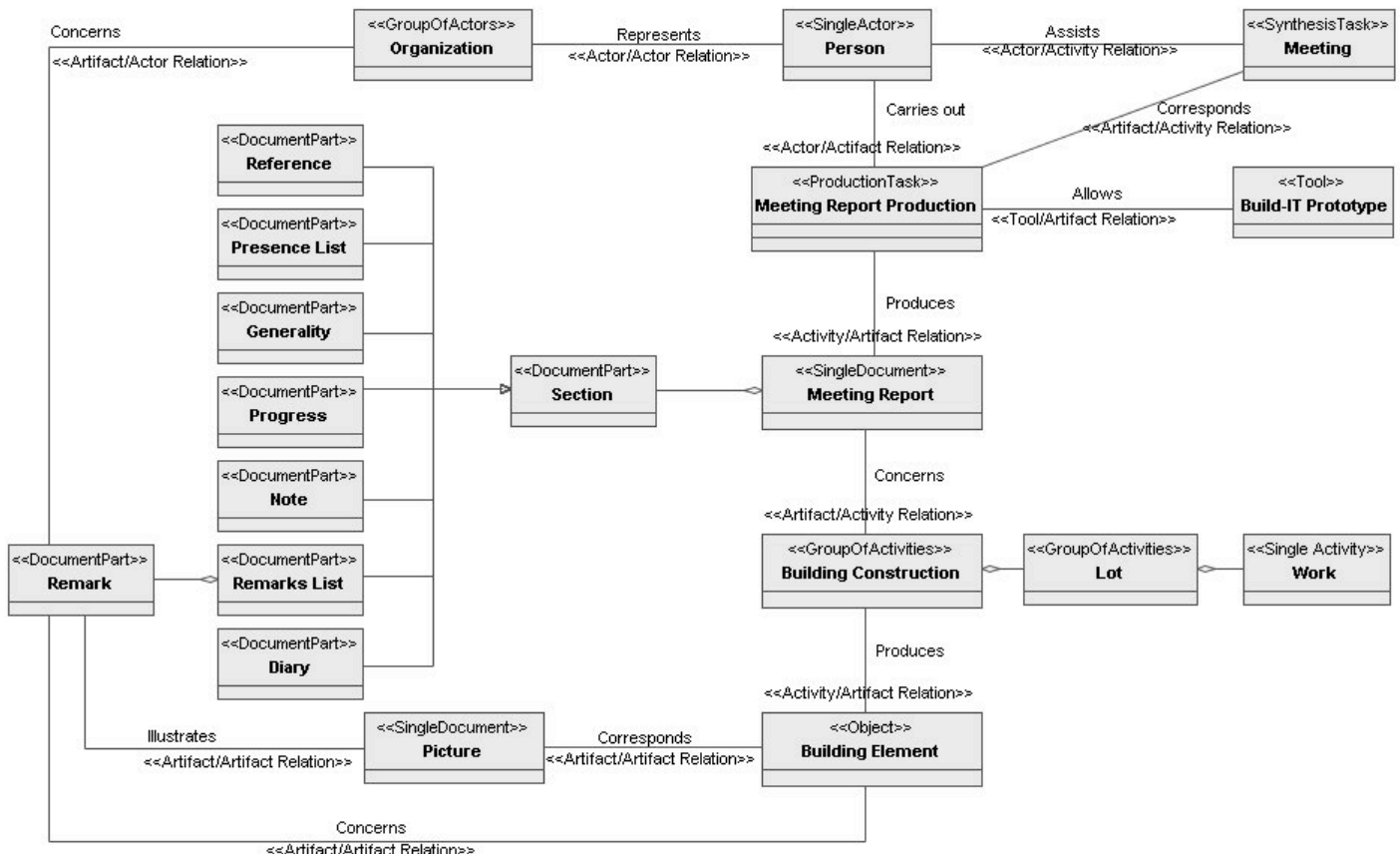


Figure 3: Meeting report model (M1)



Figure 4: Build-IT Prototype

#### 4.3.3 Reaction functionalities

By means of the tool, the reader can react to a remark if he feels that its content is erroneous or requires further information. In the current practice this approach is generally made by sending an email or by telephone conversation.

However the centralization of information and the traceability of exchanges linked to the meeting report inside a tool is a way, on the one hand, to enhance coordination between various contributors and on the other hand, to protect oneself in case of conflict because it is easier to find the source of the problem.

#### 4.4 Validation

The tool has been presented in the context of a workgroup to AEC sector professionals who have contributed with their pertinent remarks to develop some functionalities. At present a first experiment has begun on a building site.

This experiment will allow us to verify the relevance of the tool in the real world, the consistency of visualized data, the ease of use and the appropriation of the tool.

## 5 “MUTUAL ADJUSTMENT” COORDINATION ASSISTANCE

These first experiments carried out on meeting report instrumentation reveal the importance of information diffusion control. We have previously noticed that the meeting report could be enhanced with other information sources e.g. planning or building description. From the point of view of information visualization it lets us define relationships existing between the concepts manipulated in the different views of the project.

### 5.1 Coordination objectives

This development merges with the building construction coordination theory developed in part 2. It appears useful to suggest new tools to assist mutual adjustment coordination activities in complement with the previous proposition of a direct supervision tool. Such a tool could also favor a democratic vision of building construction (i.e. decision sharing etc.).

In order to succeed in this enterprise our propositions have to take into account the informal character of interactions and communications in this particular coordination activity. That's why we privilege an approach based on the relational visualization of the cooperation context.

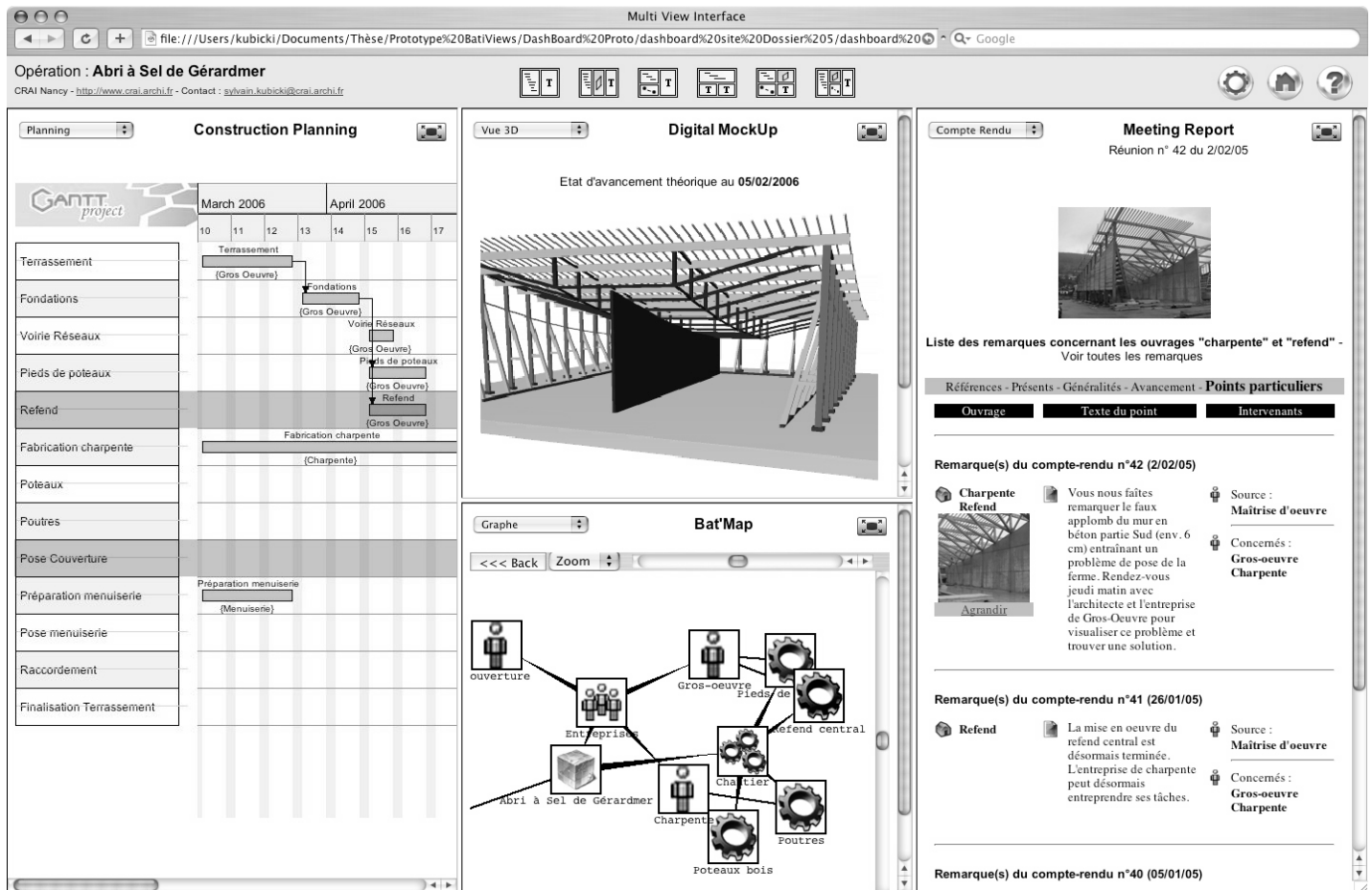


Figure 5: Bat'iViews prototype

Our hypothesis is that putting into relationship the pieces of information coming from different documents and tools could favor context comprehension by actors (i.e. process in progress, other actors, available documents etc.).

## 5.2 “Bat’iViews”: a context navigation interface

Bat’iViews<sup>2</sup> interface focuses on:

- Allowing coordination information distribution, using existing visualization modes (meeting report, planning) and emerging visualization modes (3D mock-up, graph),
- Navigating in cooperation context by the exploration of the different views and by the relationships between concepts of each views (interactions),

The tool suggests a view arrangement allowing the user to navigate in the context. This arrangement is predefined related to the role of the user. Meanwhile he could modify it in order to answer his particular needs of visualization.

At this stage different views are suggested in the tool such as meeting report, planning, 3D-mockup, relational graph and document list.

Each one of these views allows the user to interact on displayed information and then to generate interactions between views. We call « view interac-

tion » the capacity of one view to refresh the multi-view interface.

## 5.3 Domain model and view models

In this project, the model-driven approach described in part 3 allows us:

- To describe the concepts of the AEC domain and their relations (M1 model),
- To describe visualization modes existing or emerging and especially the concepts that they represent in order to build the views,
- To define and implement the interactions that each view should realize.

We then need to model the views used and especially their information representation mode.

This will allow us to define correspondences existing between models:

- The domain model and a view model: building the interface of a view,
- Two view models: defining relationships between entities in different views, e.g. a planning task corresponds to an object of the 3D mock-up and is represented by a node in Bat’Map graph (Halin et al. 2003).

## 5.4 Generating views by transformations

After having defined these models, it is necessary to describe correspondences between them in order

<sup>2</sup> <http://www.crai.archi.fr/bativiews>

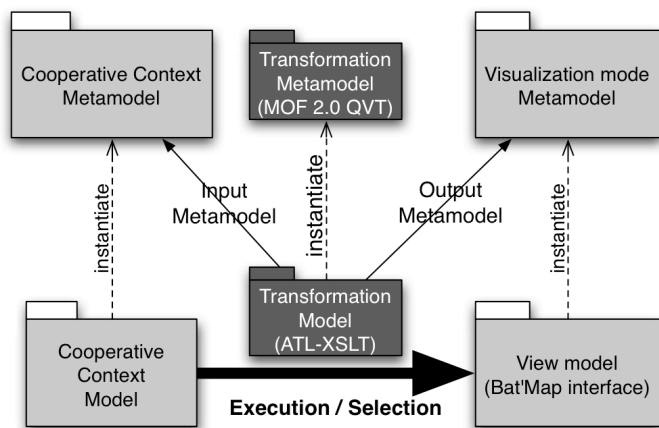


Figure 6: Model transformations

to build the views. We based this work in a research field existing in MDE and suggesting the definition of meta-models necessary for Human-Machine Interface generation (Bull & Favre 2005; Sottet et al. 2006).

The model transformations that we have described in part 3.1 provide a method for building final views. Figure 6 describes the transformation technologies used. The view model describes the concepts which could be represented. These concepts represent entities of the cooperation context. Filters could be applied during this selection in order to select information relevant for the user.

### 5.5 Perspectives and validation

This tool is still under development. A mock-up has been realized and will be experimented on and validated with professionals. Then we will realize a prototype tool which will integrate the model approach to automate the generation of the views. This second stage will allow us to validate the designed models.

In parallel we are trying to integrate some other useful visualization modes (4D, textual documents) and to define interactions that they could generate.

Finally we will define the other models suggested to design Human-Machine Interface within the meaning of MDE research: view arrangement model, user task model etc.

## 6 CONCLUSION

Reflecting upon efficiency in the building sector begins with a study on the quality of the exchanges between the actors. The AEC sector is characterized by specific coordination modes. Our theoretical approach allowed us to associate organization forms with coordination mechanisms. To support coordination we propose two tools resulting from the cooperation meta-model based on a MDE approach: on the one hand, the Build-IT prototype intended to support exchanges around the meeting report and on

the other hand, Bat'iViews allowing users to navigate in the cooperation context. Currently these two tools are in the process of being validated. Already we are envisaging the BIM (Building Information Model) as a coordination artifact and we are thinking about its integration in the coordination assistance tools developed.

## REFERENCES

- Bézivin J. 2005. On the Unification Power of Models. *Software and Systems Modelling (SoSym)*, 4(2): 171-188.
- Bourguin G. 1999. D.A.R.E. (Distributed Activities in a Reflexive Environment). 6th European Conference on Computer Supported Cooperative Work, Evolving Use of Groupware. September 12-16, 1999. Copenhagen, Denmark.
- Bull I. R. & Favre J. M. 2005. Visualization in the Context of Model Driven Engineering. *International Workshop on Model Driven Development of Advanced User Interfaces, MDDAUI 2005 @ MODELS*. October 2, 2005. Montego Bay, Jamaica.
- Engeström Y. 1987. *Learning by expanding: an activity-theoretical approach to developmental research*, Orienta-Konsultit Oy, Helsinki.
- Favre J. M. 2004. Towards a Basic Theory to Model Driven Engineering. *Workshop on Software Model Engineering, WISME 2004*, joint event with UML2004. October 11, 2004. Lisboa, Portugal.
- Godart C., Halin G., et al. 2001. Implicit or explicit coordination of virtual teams in building design. *CAADRIA 2001*. April 19-21, 2001. University of Sydney, Key Centre of Design Computing and Cognition, Australia.
- Halin G., Hanser D., et al. 2003. User Adaptive Visualization of Cooperative Architectural Design. *IJAC - International Journal of Architectural Computing*, 01(02): 89-107.
- Kruchten P. 1999. Unified Process Model (UPM) - A model of the Rational Unified Process. *IPTW'99*. October 05, 1999. Villard-de-Lans, France.
- Mintzberg H. 1978. *The structuring of organizations: A synthesis of the research*, Prentice-Hall, Englewood Cliffs, NJ.
- OMG 2000. Meta Object Facility (MOF) Specification. Object Management Group, <http://www.omg.org/technology/documents/formal/mof.htm>.
- Schmidt K. 1997. Of maps and scripts, The status of formal constructs in cooperative work. *GROUP'97, ACM International Conference on Supporting Group Work*. November 16-19, 1997. Phoenix, Arizona.
- Schmidt K. & Simone C. 1996. Coordination mechanisms: Towards a conceptual foundation of CSCW systems design. *Computer Supported Cooperative Work: The journal of collaborative computing*. Kluwer Academic Publishers, 5: 155-200.
- Soley R. & OMG 2000. *Model Driven Architecture*.
- Sottet J.-S., Calvary G., et al. 2006. Towards Mapping and Model Transformation for Consistency of Plastic User Interfaces. *Workshop on The Many Faces of Consistency in Cross-platform Design. ACM conf. on Computer Human Interaction, CHI 2006*. April 22-27, 2006. Montréal, Canada.
- Taylor F. W. 1911. *The Principles of Scientific Management*, Harper & Row.
- Toffler A. 1970. *Future Shock*, Random House, New York.
- Weber M. 1921. *Economie et société*, Plon, Recherche en Sciences Humaines, Paris.