A Scenario Approach to Validate and Demonstrate the Tool Usefulness in Cooperative Design

Halin Gilles[*], Hanser Damien[**], Otjacques Benoît[***], Bignon Jean-Claude[*]
[*]CRAI, UMR-MAP CNRS-Culture n°694. Ecole d'Architecture de Nancy.
2 rue Bastien-Lepage, 54000 Nancy, France.
halin,bignon@map.crai.archi.fr - http://www.crai.archi.fr
[**]Centre de Recherche Public Henri Tudor. 29 avenue J.F. Kennedy, L-1855, Luxembourg.
Damien.Hanser@tudor.lu - http://www.tudor.lu
[***]Centre de Recherche Public-Gabriel Lippmann. 162a, Av. de la Faïencerie. L-1511 Luxembourg. otjacque@crpgl.lu - http://www.crpgl.lu

The social and economic context of architectural co-operative projects does not support the experimentations with dedicated tools. Therefore, we chose to achieve functional validations and operations of communication about tools usefulness with the use of scenarios. The scenario techniques are usually used in computer science to capture user's needs and to validate software specifications. After having analyzed the different contexts of scenario usage in computer science, we characterize the specific context of the use of tools in cooperative activities. Then, we report some experiences of functional validation, of tool demonstration and of usefulness validation based on the use of scenarios. We explain in each experience the objectives and their implication in the scenario definition.

Keywords: *Scenario, experimentation, cooperation, collaborative tools.*

Introduction

The production of a building is not exactly like the production of a manufactured product. The design or realization collective activities are developed in a particular context (Halin et al., 2004). The work of design is there weakly repetitive. The particularity of the place (situated production), the unit of the operation (non serial production), but also the recomposition of the teams for every project generates slightly repetitive methods and requires a strong adaptation. Carrying out experimentations in a real context is therefore always difficult.

The traditional method to validate a scientific assumption consists in setting up an experiment under real, controlled, and replicable conditions of use. However, as mentioned before, the complexity and the business constraints in the construction domain made such experiments expensive and heavy to set up. Thus, we choose to adopt another ap-

proach that respectively reduces such constraints and increases the knowledge we have about the collaboration during the early design phases. The use of scenarios appears like a useful alternative. If we define a scenario as a codification of a real situation, it can be considered as a model from which a simulation of the real activity can be achieved.

The use of scenario stays efficient because it permits to be liberated from too uncertain states and to work in a mastered time.

However, it presents some limits as any model. It is always difficult to model "the uncertain" that is, however, a component of the real processes. A scenario is only a partial representation, which limits its ability to validate or refute a conjecture.

The first part of the paper describes the typical uses of scenarios in computer science and the particular context of its use in a cooperative activity. The specific context of cooperation in architecture is exposed in the second section. The last part presents the analysis of three experiences using a scenario in an architectural cooperative context.

I. Scenario usages in computer science

A scenario allows the description of interactions between several actors in a specific context. It can describe a real or imaginary, past or future situation. It can materialize as a "story board" in the making of a movie or an animation or as the "instructions for use" for describing how to use a domestic or a professional tool. It can also act as a "proof" in a police investigation.

I.1 The system place

When a computing system belongs to the production of a scenario, it can take a more or less important place. This place is central when the subject is to determine the needs that will be covered by the future system, to study the mechanisms needed for the implementation of the functions, to validate the relevance of the functions included in the system or to communicate to the future users on the contribution of the new system in their organization. On the other hand, when the study concerns the process engineering of an enterprise, the place of the system is identical to those of the other actors. The system plays the role of a potential actor with which any other actor can interact.

Then, the use of scenario is characterized by a context, that conditions its shape, its content, its cycle of life (Rolland et al., 1998).

I.2 The scenario context

By considering the context of a scenario, (Pohl et al., 1997) define three types of scenario (cf. Fig 1) :

A. System internal scenarios describe interactions between the implementation objects.
B. Interaction scenarios represent knowledge about the interaction of the system with its context.
C. Contextual scenarios represent, in addition to the direct interactions between the system and its context, also information about the context of the system (objectives, organizational policies, process, available resources).

In a cooperative work, this context description may evolves toward an assembly of C type contexts (cf. Figure 2), that we call D. This new context is then the union of each C type context of each actor. Indeed,

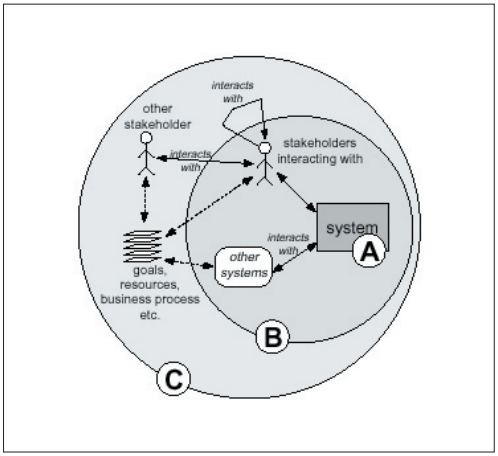


Figure 1
Three scenario types according to (Pohl et al., 1997)

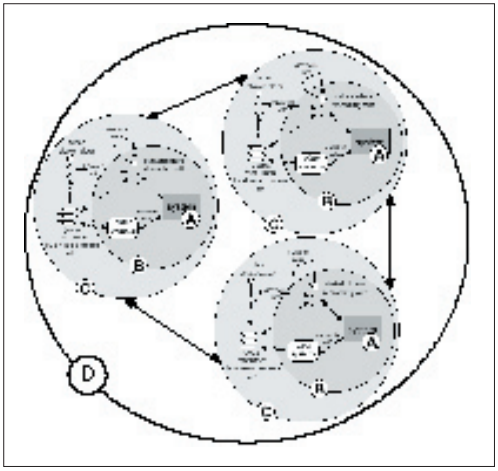


Figure 2
The context of a scenario describing a cooperative activity.

every actor of a cooperative project is going to interact with other actors by the use of its system. This cooperation activity participates to the processes of each actor's enterprise.

I.3 The form and content of scenario

A scenario can take various forms: it can be a narrative text or a more formal expression as a state diagram. In requirements engineering, scenarios has the form of sequence diagrams describing the interactions between the users and the system, which is considered as a black box. Other medias can be used to amplify the impact of a scenario: pictures, sketches, screens shots or videos.

The content of a scenario can be described according to several aspects (Rolland et al., 1998): its behaviour, the sequence of the described interactions, the involved organizations (structure, role, ..), the directed actors or agents (feature, roles in organization, aspirations, objectives), the type of argument (pas compris ?) (position stand, solution arguing, conflict description, …), the abstraction level (actor or system instances or instance types).

In order to understand the utility of the approach by scenario to validate or to demonstrate tool dedicated to cooperative activity, we have to study more precisely in which context this activity takes place.

II. Cooperative design context and IT

II.1 Understanding the context of collaboration-cooperation

The early phases of a building project reflect a particular collaborative context (Kvan, 2000) that largely uses informal and non-linear coordination processes (Visser, 2002). The group activities that appear in these phases (e.g. preliminary studies) are difficult to describe in a process and mainly depend on the project characteristics. A common approach to describe design activities is to explain that each project is a completely new 'business context' with new relations (contracts, hierarchy) between actors that are independent from others.

This intuitive description is supported by the social theory corpus (Halin et al., 2004) that isolates regulation and human relationship categories applicable to this context. For instance, in a cooperative activity involving actors from different companies in a mid-scale project, we can observe the following elements:

- each business actor has his own working processes and habits;
- unifying the working processes among organizations is currently impossible;
- software and levels of IT maturity are frequently heterogeneous among actors;
- competition between actors is sometimes rough.

In this context, people who want to reinforce cooperation needs to find a way to:

- promote flexible organization processes;
- increase group identity to reduce resistance form each actor against the group;
- create appropriate awareness among actors to increase self-regulation capabilities.

II.2 Real-business experiment, a thorny question

First of all we must remind the situation that we faced in our research work. When we started working on this topic in 1998-2000, the computer mediated collaboration technologies were in full rise but nobody wanted to take the risk to apply them to „real-business“. The situation has not changed very much and the real applications of such tools in everyday life (in un-sponsored conditions) are still rare. In a previous publication we made the assumption that the core problem is not the development of numerous functionalities to support collaborative work but rather the adaptation of such functionalities to the user needs (Hanser et al., 2001). The developments mentioned below led us to check our assumption on a theoretical level that should be supplemented with a terrain validation.

The traditional method to validate a scientific assumption consists in setting up an experiment under real conditions of use. However, as mentioned before, the complexity and the business constraints in the construction domain made such experiments expensive and heavy to set up. Thus, we choose to adopt another approach that respectively reduces such constraints and increases the knowledge we have about the collaboration during the early design phases.

III. Some scenario usage experiences

III.1 Usage of scenario to assess usefulness and ease-of-use of software

The goal of this experience was to assess the potential of adoption of an experimental application dedicated to manage small construction projects (Otjacques et al., 2003). The Davis'TAM model (Technology Acceptance Model) (Davis, 1989) aims to predict the degree of acceptation of new technologies and relies on two key concepts: perceived usefulness and perceived ease-of-use. Perceived usefulness refers to the degree to which a person believes that using a particular system would enhance his or her job performance. Perceived ease of use, in contrast, refers to the degree to which a person believe that using a particular system would be free of effort. According to the TAM model, the use of a new technology is defined by behavioral intentions to use the technical system, which are determined by people attitudes toward using and by perceived usefulness. Perceived usefulness depends on perceived ease-of-use and on external variables that may refer, for instance, to individual characteristics, the features of the task to be done, or the organization.

The evaluation methodology that has been retained was to set up an experiment with students in penultimate year (4th) of a Master in Architecture class. They had already spent significant periods of time as trainee in real offices of architects, which made them qualified to give some valuable feedback in the context of the experiment.

The experimental software offers several original features, of which the evaluation would requires a significant period of time. Therefore, in order to collect worthwhile data, the experimenters have decided to focus only on four specific functionalities and to study them intensively. Considering the importance of direct manipulation for getting a correct perception of the strengths and weaknesses of an application, the researchers decided to use dedicated scenarios to evaluate each of the features under examination.

Before the real tests, the adopted protocol has been pre-tested by a pilot group of people, in order to validate its capacity to be operated within the context of the real test. Among others, were especially validated the understanding of the questionnaire by the subjects, the satisfaction of time constraints and the presentation of the scenarios.

In the context of this paper, the focus is set on the evaluation of the usage of scenarios rather than on the prototype assessment. At this level, several lessons have been learned from the experiment:

- First, it appears that the multi-level scenario approach (one global scenario followed by several dedicated ones) seems to be efficient to progressively familiarize the subjects with the software to be tested.
- Second, it seems important that the subjects play an active role within the scenario. One may imagine that, to some extent, the usefulness might be evaluated without direct manipulation but it is obvious that the ease-of-use couldn't. Moreover, the TAM model states that the perceived usefulness is influenced by the perceived ease-of-use, which strengthen the need to confront the user with the software.
- Third, asking the subjects to fill out a limited and dedicated questionnaire after each sub-scenario allows reducing the cognitive effort required by the subjects to memorize the evaluated features.

Some limitations have also been observed with the usage of scenario in such a context:

- First, the experimenters have noted that a scenario cannot simulate the whole complexity of real life. For instance, for operational reasons, the test had to relate to only one project while most of the architects handle several ones simultaneously.
- Second, the scenario can't simulate the real life psychological conditions. For instance, the stress that an architect may face due to a lack of time or an unexpected problem on the building site cannot be reproduced.
- Third, by definition, a scenario has a limited duration, which make it difficult to take the learning process into account.
- Fourth, students can serve as experimental subjects to a given extent but they cannot be fully assimilated to professionals.

III.2 Scenario as a testing framework

In this experience we used scenario technique to define a process that reflects the coordination problems occurring during design tasks. In this scenario we chose to represent the system as an ordinary actor of the process. We model thus the interactions between actors and the system as generic as possible (e.g. upload, download, notifications, etc.). The resulting scenario is quite general and can be contextualized in a cooperative activity that used a specific tool.

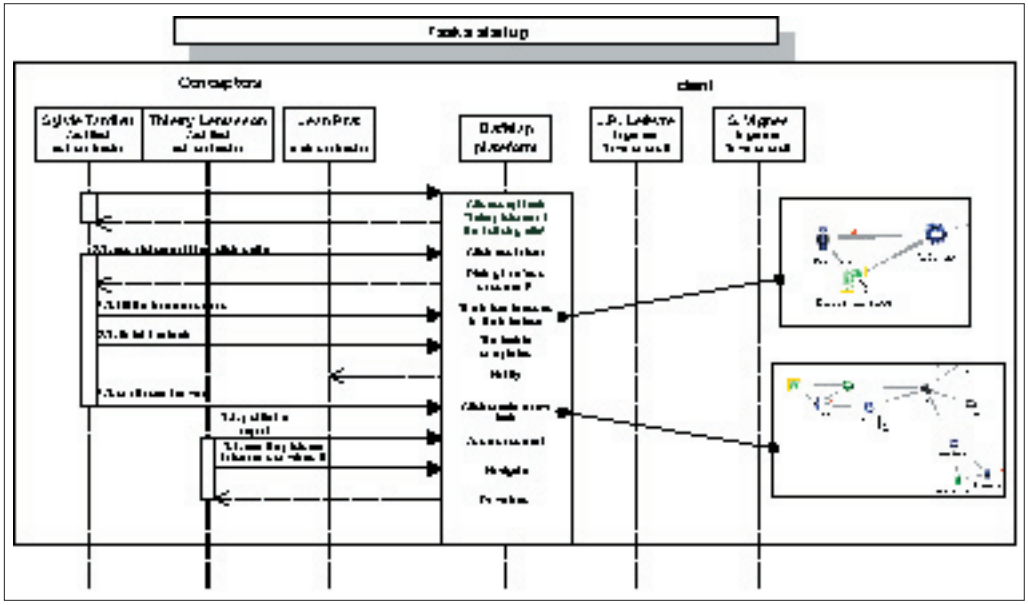
This scenario was used to experiment and validate the contextual visualization of cooperation proposed in (Halin et al., 2004). In this experiment we tested two visualization approaches with the same scenario. The first approach used a traditional graphical interface and the second used the hypermedia visualization tool. Two students groups did the first experiment, a first group was called 'classical', which uses the traditional user interface, and the second group was called 'graphical' that uses the hypermedia visualization to perform the scenario. This experiment allowed identifying the pertinence of a contextual visualization within a simulated co-operative framework. In order to increase the reliability of the observations we made during this experiment, the tools that we developed were using a unified structure of concepts, a similar graphic identity and a unique database. The classical and graphic tools that we used share the same database and manipulate the same concepts, thus we can switch the views from classical to graphic to allow a trustful comparison.

The actors of the 'graphic group' carried out the scenario more quickly than the 'traditional group' which faced the difficulty of finding in the traditional interface either the function to be used or the exact place where to put in the information. The context visualization and the dynamic creation of nodes and links, available in the graphic interface, revealed here their efficiency compared to a classical (hierarchical) visualization of information. Figure 3 shows a part of the scenario contextualized in this experimental context.

III.3 Scenario for communication

The objective of this experience is to communicate

Figure 3
Example of a contextualized scenario in a validation context



with a video support to a professional community on the interest or the relevance to use a contextual visualization tool (Halin et al., 2004) in a cooperative setting. The scenario writing is based on the analysis of a first experience of communication to which we participated. This first experience of video making had for object to demonstrate that the functions of a research prototype of CSCW were useful in an architectural design project.

The presentation of this video during public demonstrations put in evidence its maladjustment to communicate a clear message to a non-aware public in groupware techniques, this for several reasons: too many complex functions have been presented, interactions with the system have been not understood, distributed context of cooperation has been not always discerned, too many characters, roles of the characters have not been understood, presentation of the user's interface was not adapted to a non aware public.

In order to tackle these communication problems, we built a new animation by using a software development process with iterative and incremental qualities. A first version of the animation was created with an animation programming tool. This version evolved according to the reactions of the targeted public (i.e. architects). The objective of the first development phase has been reached. A final scenario has been obtained according to some rules extracted from the analysis of the past experience:

- directing only a few characters (2 or 3) with a clearly defined role,
- choosing a subset of the functions to illustrate (the most important or the easiest to understand),
- dentifying some situations of real cooperation (3 or 4) and their context,
- making the interactions with the system (animate screen capture) more explicit,
- limiting the length of the animation to five minutes.

The scenario has been written in a tabular shape (cf. Table 1) in order to specify any information necessary to the characterization of the context to be simulated in the animation.
This first version of the animation must now be presented to the architects. We wish to capture their reactions in order to make evolve the animation if the remarks are about the realization or about the scenario or to improve the user interface and the functions of the tool if the remarks are about the usefulness or the ease-of-use of the presented functions.

III.3 Scenario, a promotion support

In this experience we identified two opportunities. First, it consists in using scenario to present research results to professionals: in this case we use a scenario to transfer, initiate and manage a technological change and acceptance process (Davis, 1989; Orlikowski et al., 1997). The scenario is more familiar and contextual than other presentation methods. An instance of such an application is currently used in Luxemburg in a R&D project[1] to promote innovative collaborative manners using IT. This project is intended to specify a software infrastructure dedicated to the building context. In this project we use scenario approach as a communication media to transfer our conclusions to the professionals. Such a method allows us to reduce the resistance to change expressed by the professionals against the use of IT.
Last we identified an opportunity to use the scenario approach to demonstrate the usability and reliability of our concepts in a realistic project context. Our goal is thus to overcome the reticence from the professionals in order to carry out 'real business' experiments and make some innovative techniques more popular.

Conclusion

These experiences were some opportunities to explore the potentialities of scenario-based representations. They showed that the usage of scenarios offers:

- a good representation of context;
- an easy way to familiarize professionals with IT concepts;
- a highly expressive way to present complex technologies;
- a low abstraction level that put users in confidence;
- a generic level that allows researchers to test and compare their prototypes.

The scenario construction process is a real opportunity to stay close to end users. It also offers op-

Phase of the project	Actor	Role	Place	Realized activity	Tool function
Initialization	Urbanist engineer	Project manager	Town hall office	Description of the organization of the actors	Creation of actors, of links between actors.
Scheduling	Urbanist engineer	Project manager	Town hall office	Description of the tasks to achieve. Deposit of the existing documents.	Creation of tasks, of documents and ok link. Document upload.
Project progress	Urbanist engineer	Project manager	Rehabilitate square	Verification of the plans	
Project progress	Architect	Sketch producer	Agency	Deposit of his sketch.	Change of state of a task
Project progress	Urbanist engineer	Project manager	Town hall office	Make his remarks on the present stage of the square	Navigation, Change of state of a task

Table 1
scenario used in the animation

portunities to initiate reflections about 'change management' or innovation in application of technology. The critical point is certainly the analysis of the practices, which is to build scenarios. The relevance of the applications directly depends on the quality of this analysis. In conclusion, using scenarios appears to be an efficient tool to confront users with new software, especially when real life tests can't be carried out. Nevertheless, the researchers must also be aware that users confronted to a scenario will probably never behave exactly in the same way than they would in real life.

References

Halin, G., Hanser, D. and Bignon, J.-C.: 2004, User adaptive visualization of cooperative architectural design, International Journal of Architectural computing, 02(01).

Rolland, C., Ben Achour, C., Cauvet, C., Ralyté, J., Sutcliffe, A., Maiden, N.A.M., Jarke, M., Haumer, P., Pohl, K., Dubois, E. and Heymans, P.: 1998, A Proposal for a Scenario Classification Framework,, Requirements Engineering Journal, Springer Verlag, 3, pp. 23-47.

Pohl, K. and Haumer, P.: 1997, Modelling Contextual Information about Scenarios, in Third International Workshop on Requirements Engineering, Barcelona, Spain.

Kvan, T.: 2000, Collaborative design : what is it ?, Automation in Construction, 9, pp. 409-415.

Hanser, D., Halin, G. and Bignon, J.C.: 2001, Relation Based groupware for heterogeneous design teams, in Education in Computer Aided Architectural Design in Europe, Helsinki, pp. 86-91.

Otjacques, B., Post, P. and Feltz, F.: 2003, Management of information flows during construction projects, in 20th International Conference on IT for Construction CIB W78, Auckland, New Zealand, pp.278-285.

Davis, F.: 1989, User Acceptance of Information Technology: Systems Caracteristics, User Perceptions and Behavioral Impacts, International Journal of Man-Machine Studies, 38(3), pp. 475-487.

Orlikowski, W.J. and Hofman, J.D.: 1997, An Improvisational Model for Change Management: The Case of Groupware Technologies., MIT Sloan Management Review, 38(2), pp. 11-21.

Footnotes

[1] The Build-IT project is piloted by the Centre de Recherche Public Henri Tudor (www.tudor.lu) and the Centre de Ressources des Technologies de l'Information pour le Bâtiment (CRTI-B; www.crtib.lu) which group architects, civil engineers, constructors and public administration to define national standards and IT opportunities for the building domain.