Designing Communication-Intensive Web Applications: Experience and Lessons from a Real Case

Vito Perrone
HOC laboratories - Politecnico di Milano (Italy)
perrone@elet.polimi.it
and
Davide Bolchini
TEC-Lab - University of Lugano (Switzerland)
davide.bolchini@lu.unisi.ch

ABSTRACT

Who uses requirements engineering and design methodologies besides the people who invented them? Are researchers - at least - actually trying to use them in real-world complex projects and not in "paper project"? In this paper, we dare to recount the experience and the lessons we gained in trying to use seriously and in-depth a requirements engineering method (called AWARE) combined with a conceptual user-centered design method (called W2000) for the development of a real-world web application. The project is recounted through the process followed and the artefacts produced, as well as by crystallizing our experience in using and transferring the method to industry in practical and methodological recommendations.

Keywords: web applications, goal-oriented requirements analysis, user-centered conceptual design, technology transfer.

1. INTRODUCTION

Despite requirements and design models methodologies are increasingly available in the research arena, current and prospective developers need to be further exposed to best practices and real-world project experiences, where existing conceptual tools are tentatively but actually exploited, and where useful reccomendations can emerge.

This paper recounts the experience and the lessons-learned emerged from applying goal-oriented requirements engineering (in the form of the AWARE model [6]) and user-centered conceptual design (in the form of W2000 method [7] for the development of a complex web application for the Italian Ministry of Public Works.

On the one hand, the project involved a variety of different stakeholders and goals (from the ministry committees, to the editorial staff, to collaborating institutions and organisations, to a diversified set of user profiles) and seemed therefore particularly suitable for the adoption of a goal-oriented perspective in requirements engineering. On the other hand, the communication potential of the application was rich, as rich and structured was content to be conveyed through the interactive means. In this view, the project seemed also particularly suitable for the adoption of a structured approach to the conceptual design of information, operation and navigation architecture.

During the project, we targeted three meta-objectives:

1) As to the project success, we tried to keep a stakeholder-centered perspective throughout the project from requirements elicitation to prototyping, so to possibly satisfy all the stakeholders.
2) As a traceability concern, we tried to diminish the gap between requirements and design, trying to iterate during the process and following up as realistically as we could with the documentation.
3) As a methodological concern, we tried to continuously monitor the capability of the existing methodologies (in terms of concept expressiveness and documentation potential) of coping with the issues we met during the project phases.

The lessons learned are interesting for different aspects: they concern the specific methodologies per sé (AWARE and W2000), the paradigm underlying the methodologies (goal-oriented and user-centered conceptual design), and their exploitation in a real environment (timing, resources, organization and documentation). Particular attention is also devoted to the technology transfer experience we had as part of the project.

The remainder of the paper is the following. Section 2 recalls the distinctive features of the requirements and design methodologies we used in the project. Section 3 explains and comments on examples of the analysis

and design artefacts we produce along the project with the methodologies at hand. Section 4 points out key lessons learned we would like to share and discuss for practical and methodological improvement. Summary and concluding remarks and presented in Section 5.

2. RELATED WORKS

After a first stage where pioneer approaches to web development were predominant and time-to market was the unique driving factor for the deployment of websites, nowadays practitioners are devoting more and more interest to the quality and effectiveness of final products. On one hand, these attributes are strongly related to fitness to users' and stakeholders' requirements and to the usability of the final website. On the other hand, complexity of modern websites is significantly growing up. In this light, structured and systematic approaches to requirements and design, have the potential of playing an important role for shaping effective interactive applications. However, several obstacles have been acknowledged as general barriers for technology transfer, as documented in recent industrial experience and surveys [2][5]. Very few reflections on practice have been reported experience in exploiting requirements and design approaches in the web and hypermedia industry, except for an experience on specifying web services at the implementation-level [17].

If we look for modeling methods and tools that have gained significant acceptance in the ICT development industry, we see that UML appears extensively used for modeling software modules and system architectures. However, it is still highly controversial whether UML can be used effectively to model properly stakeholders' requirements [16]. Furthermore, the lack of UML [9] in addressing the most important issues in the design of hypermedia applications is widely documented by several extensions (a list of which can be found in [12]) that have been proposed over the last years.

To cope with the specific needs of modern Web applications, our groups have developed two methodologies – AWARE and W2000 – which root their foundations in more than a decade of experience gained working on and with HDM [10], one of the first well-recognized design models specially tailored for hypermedia applications.

Let us briefly recall the essential features of AWARE and W2000.

AWARE *(Analysis of Web Application Requirements)* [6] offers simple primitives enabling to document and specify goal-oriented hypermedia requirements. It provides a set of conceptual tools that web analysts might find useful for describing and reasoning with website requirements. Like traditional goal-based approaches (such as *i** or KAOS), it recognizes the central role of the stakeholders and their goals for requirements elicitation and analysis. However, whereas *i** [13] provides constructs to model actors and their dependencies with respect to their goals and tasks, AWARE extends the analysis of *i** to the elaboration and definition of hypermedia-specific requirements [6]. To this end, AWARE provides a requirements taxonomy to bind requirements to hypermedia conceptual design and help organize the design activity accordingly. The requirements taxonomy is used to define the so called requirement dimensions. AWARE proposes a basic set of dimensions (Content, Structure of content, Access Paths to Content, Navigation, Presentation, User Operation, System Operation, Interaction) that can be easily extended accordingly to the specific design needs.

W2000 [7][10] is a structured method for the conceptual design of the user experience. It organizes the design of a (complex) hypermedia application around four main activities:

- *Information design:* defining the basic conceptual information units (entities) as perceived by the user and the different access structures (collections) enabling users to reach them on the basis of their needs.
- *Navigation design:* defining the navigation structures basically in terms of nodes (i.e. units of interaction) and links allowing users move among them.
- *Presentation design:* defining the page structure in terms of lay-out aspects, graphical elements and page organization relying upon the navigation design.
- *Business Process Design:* defining the service architecture of a Web application in terms of processes and operations.

Following the W2000 design strategy, the conceptual design is achieved at two levels of detail: *in-the-large* design, where a coarse-grain solution is quickly sketched to meet initial requirements, and *in-the-small* design, where solutions are detailed to be used as input for the implementation activities. The path between the former and the latter levels is not straightforward but is traversed in several iterations.

3. ANALYSIS AND DESIGN ARTEFACTS

The project (whose phases, activities and workflows are fully described in [15]) consists in developing an institutional Web site whose main purpose is to provide information about the CSLP's structure and activities and access to the archive of "Opinions", "Laws" and other documents produced by the CSLP. Furthermore, the project also includes the editorial

application which should be used to feed the Web site archive.

3.1 Stakeholders

Although the relative small dimension of the site, a number of stakeholders have to be taken into account to devise an effective communication and a usable Web site. These stakeholders cover some typical roles of projects concerning institutional and organizational communication even of greater complexity. Figure 1 shows a classification of the Web site's stakeholders.

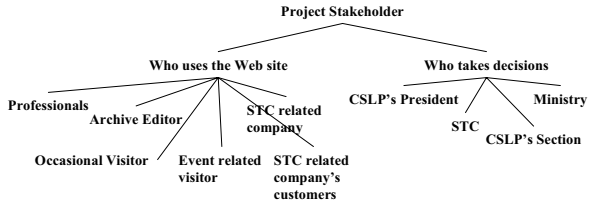

Figure 1: CSLP project's main stakeholders

Among the site's visitors: *Professionals* are technical people (engineers, architects, works managers, public body's agents, etc.) who work in technical fields in charge of the CSLP; *STC related companies* are those companies whose work strongly depends on normative documents produced by the STC (e.g. a geologic laboratory has to renew its authorization every years and the STC defines and publishes on the Web site the procedure and the constraints); *STC's related company's customers* are privates or other companies who are customers of companies whose activity is subordinated to some public authorization; *Occasional visitors* are those who have heard of the CSLP in some way and probably can found what they are looking for on the Web site; *Event related visitors* are those who are waiting for a specific activity to be accomplished (e.g. an assembly and the relative "opinion" to be published); *Archive editor* is some CSLP's employ who is in charge of adding new documents to the Web site's archive.

3.2 Requirements

Except the last category, almost all the users have two main generic and abstract goals for the Web site, that is, understanding what the CSLP (and the STC) can offer and finding out the needed information. However, these goals assume a different meaning as well as the user category varies, bringing to different sub-goals and requirements. These differences definitively impact all the design dimensions of the Web site. In Figure 2 the goals and requirements of two user categories are depicted. Requirements for Professional Visitors are only partially reported for the sake of conciseness. From the diagrams some considerations can be done about requirements.

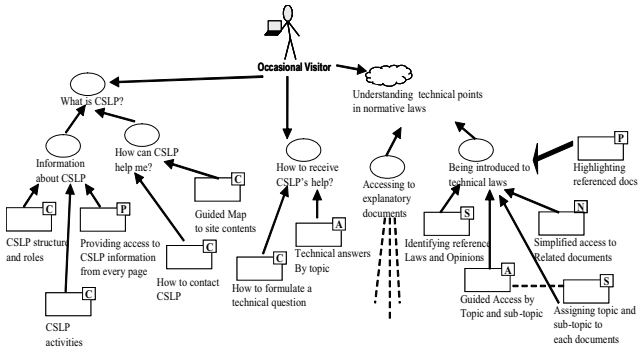

Figure 2. Excerpts of AWARE analysis for Occasional and Professional Users: from goals to requirements. For a detailed legenda see [6].

These user categories have a common soft-goal [8] but they also have specific goals. As shown, only occasional visitors need some information about CSLP while it is supposed professional users know CSLP, its role and activities by their daily work. By this goal, two content and a presentation requirements are devised. They state that information about CSLP and its activities should be provided to first-time users enabling them to take out a clear picture of how CSLP may help them and in which way (content requirements). Moreover, since these users are not familiar with the web site and they could land in whatever page, a link towards this information should be always available and evident (presentation requirement). Passing to the shared soft-goal – *Understanding technical points in normative laws* – the refinement tree makes evident the different needs behind the common initial goal. Both users aim at *Accessing explanatory documents* (requiring several kinds of documents, as "Opinions", "Laws", "Specification templates", "Guidelines", "Technical answers", etc., to be accessible through the Web site) but the main difference is that occasional visitors do not have a clear idea about the content of the Web site and needs to be guided for discovering needed documents, while professional users know very well what is offered by CSLP, are accustomed to the Web site but may need sophisticated tools for catching all needed information. Among other, this goal bears the need for a classification of documents by topic. This need is reified in two kinds of requirements, a structural one that entails topics and sub-topic being included to the documents structure, and an access one that asks for some access paths that exploit the topics classification. These requirements are shown as related each other by the dashed line which connects the requirement boxes.

Summarizing on all the stakeholders the whole requirements set consist of 28 content, 16 access, 5

structure, 5 navigation and 4 presentation requirements.

3.3 Conceptual Design

Adopting the Jackson terminology [1] requirements belong to the problem world and design to the solution world. Although this distinction seems to be straightforward, when passing from theory to practice, a broad agreement about what is requirement and what is design does not really exist, at least from our experience. It often depends on the reference community where examples are discussed. To someone, the requirements classification, in terms of design dimensions, has too much to do with the design world while to others our conceptual design is too far away from the machine to be considered a solution. In our approach goals have doubtless to belong to the problem while requirements are used as a means for describing what the analysed problem requires as a possible solution, which in turn will be described by a conceptual model. The conceptual model focuses on the user experience neglecting the description of details related to the machine. The combined use of AWARE and W2000 should provide a seamless path from the problem world to the solution world, at least from the user point of view. They are specially suited for communication-intensive Web site where a proper communication strategy is a key of success.

Looking at the requirements classification, in terms of design dimensions, a clear picture of the application design can be figured out. Observing that the most numerous sets are content and access requirements, we can derive that most of the effort has to be put in designing *entity types* and *access structures,* while the relative minority of presentation, navigation and structure requirements assumes a design where the designer has an high level of freedom in defining the navigation and presentation aspects and content structures are not particularly constrained. In the following some examples taken from the overall design are presented with the main purpose of showing W2000 in action and its relation with the requirements set. One of the entity types that has been added to the new version of the Web site is "Technical law. In Figure 3, its in-the-small description is reported as extracted (and translated) from the design document. In the first iteration of the analysis process only *multiplicity, semantic, source and comment* are usually compiled with several references to the recorded interviews and to goals and requirements. By the *multiplicity* attribute the designer aims at evaluating how many instances of this entity should be included in the Web site. In W2000 multiplicity is specified in terms of *minimum*, *maximum* and *expected* instances. Usually the most important is the last one since it provides useful information for designing *navigation paths* (by means of W2000 *collections*) and *pages*. Concerning the relation with requirements, in the *semantic* or more often in the *comment* attributes, explicit references to goals and requirements are provided. We do not impose a specific way to take into account requirements in the design artefacts but rather we suggest to use one of these two attributes (possibly always the same) and to use a syntax like that used in this example, that is, *Stakeholder:Goal/Requirement (kind)*. An example of attribute introduced in advanced iterations is the *preview segment.* It is used to specify how to briefly represent an entity within a list. An entity may have more than one preview segments (this feature is often used to design different previews for devices with different visualization capabilities). In the in-the-large version the preview is usually specified by means of a textual description, while the in-the-small version specifies the specific attribute that are used as preview among the target entity's attributes set.

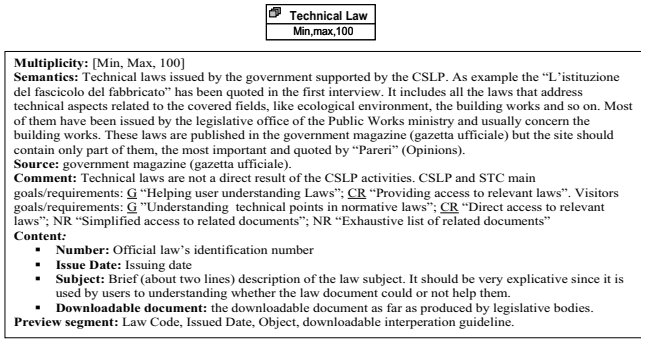

Figure 3. In-the-small description of entity type: "Technical law".

The detailed description of the *content* is usually added in later iterations when passing from the *in-the-large* to the *in-the-small* design, even if it may vary from project to project. As shown in the previous paragraph, only few requirements concern navigation needs of users. In particular only a kind of *semantic association* has been devised – "Useful References" (Figure 4) – with a different semantic for each navigation direction.

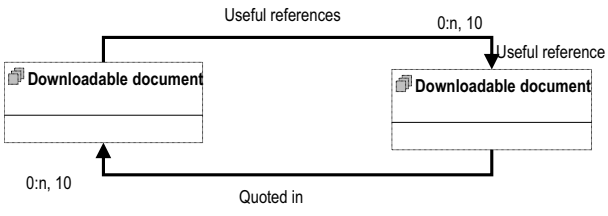

Figure 4. Example of semantic navigation design.

This kind of association can involve several entity types which are represented by a *role entity* "Downloadable document", also specified in the design. Lack of space prevent us to report the full description of this association. In W2000 an

association is specified by: *Source entity, target entity, multiplicity, semantic,* two *direct names* (that will be shown in the Web pages, one for each direction), *topology* (one for each direction and describing how the list is organized, e.g. "grouped by entity type and ordered by date"), *preview segments* (specifying which preview segment are used among those available for the involved entities). This association has been designed for supporting navigational requirements that asked some navigable connections between documents with the purpose of helping the user understanding of every document (e.g. G "Understanding technical points in normative laws"; NR "Simplified access to related documents"; NR "Exhaustive list of related documents"). From the content management point of view, we decided (designed) that when the archive editor inserts a new document in the Web site, it should also specify some reference documents. In defining this association, an advantage induced by the systematic reasoning enabled by W2000 has come out. The cited access requirements required to specify some references for a given document. Being in W2000 all associations bidirectional (except explicit different decisions) an immediate question rises up: which is the semantic of the opposite direction? We assigned to this direction the meaning of "quote". Thus, if a document X has a document Y as reference, then Y has X as quote. By means of this observation, when now a user finds out interesting documents, he can also find a list of reference documents and a set of documents which quote it as reference, expanding the correlation possibilities. As a clear advantage, from the content management point of view, no more effort is required to the archive operator since quoting documents are automatically derived. Moreover, looking to the dimension of the quotes list, a user can also figure out the importance of the current document.

Once content has been designed by means of entity types, the next step in the W2000 method is to define how user can reach such content, that is, the *access structures*. Access structures are modelled by *collections* of entities or other collections. A collection containing another collection defines an *access path*, while a collection containing only entities is called *base collection* and its member are instance of entity type selected on the basis of some *population criteria.* In Figure 5 the access paths dealing with entity types "Opinion" are depicted.

Looking to these collections we can notice as access requirements shown in the previous paragraph have been used in the design stage. Collections properties are modelled by several attributes: *semantic, comment, population criteria* (an informal, semi-formal or formal description of how instances are selected), *member multiplicity, topology and filter* (if any, it specifies some filter attributes of member entities by which it is possible to further filter the instance set). References to requirements should be included in the collection description as already described for entity types. Finally, last step in the design process is the definition of pages structure and content, shaped on the basis of the information and navigational units defined in the earlier design activities. For space reasons, we cannot here report sketches on the page prototype designed and discuss in detail the design decisions taken for publishing design. For a more in-depth explanation, please refer to [15].

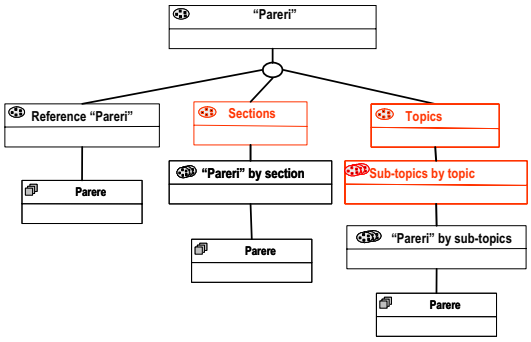

Figure 5. Access structures to "Opinions" (it. Pareri).

4. LESSONS LEARNED

Drawing upon the experience gained working on this project and other projects conducted with other companies, a number of considerations can be extracted. In this section we put these considerations in form of lesson learned which, we hope, could help in similar situations and provide some inputs to the research communities related to requirements and conceptual modelling. To this purpose, we have tried to abstract from the specific characteristics of the project, even is in some cases lessons rise out from the specific case. Lessons concern different aspects of the overall experience, ranging from considerations about the project set up and technology transfer, passing from requirements up to design.

Lesson 1: *industry needs for proper "mediators" towards the new technologies*.

As mentioned in [5], industry often does not employ systematic approaches, like those usually promoted by the academy, and does not retain an appropriate knowledge. In most of cases, requirements and conceptual design are only sketched in natural language. The increasing request of quality and effectiveness is, however, driving industry towards the research of such approaches. In this light and due to the inherent complexity of such approaches, a proper mediation of experts is required in order to avoid novice practitioners feeling lost and isolated (overcoming the "negative perception" as mentioned in

Errore. L'origine riferimento non è stata trovata.). A possible approach that turned out to be effective, at least in this and others our experiences, is to provide the basics of the methodologies in a short and intensive course, and then carrying out together (trainers and trainees) case studies (possibly real-world ones) in their domain. After this experience, CM people felt familiar with our methods and able to adopt them in other projects (referring only to the support documentation).

Lesson 2: *organizational constraints could prevent a full exploitation of new methods*

As mentioned in the project workflow description (paragraph 3.2), organizational constraints didn't allow us to organize users interviews. Internal organization and current practice of target companies could require new technologies being only partially used, reducing their potential power. On the other hand, we cannot ask a drastic change in the proven practice of a company without seriously compromising the chances of convincing them to try new technologies. Thus, in our experience the introduction of new methods should be carefully matched with the current organizational constraints trying to merge both their needs as much as possible, but a perfect matching is quite impossible to be reached at least in the first attempts.

Lesson 3: *Potential users interviews can rarely be performed in practice*

Although users play a crucial role for the final success of Web applications, organizing interviews or other forms of requirements elicitation enabling a direct contact with final users is hardly feasible. User requirements are thus simulated by some stakeholder belonging to the client company. In this light, it is important to carefully examine this source of information in order to prevent forms of bias. In our case, the interviewed stakeholders were specialized in different technical areas and with different technical profiles. This sample was quite good for simulating "professional visitors" but rather different from "occasional visitors". Further elicitation activities were needed to cover this kind of users.

Lesson 4: *Communication requirements may be controversial across different branches within an organization.*

Communication requirements can be conflicting. As mentioned at the end of paragraph 3.2, a coincidence in the composition of the interviewed teams brought to conflicting requirements concerning the CSLP image to be communicated. Driven by what elicited by the STC head, our first specification contained the presentation requirement "Providing users with direct access to STC's services". As a consequence our first solution designed the Web site home page so that STC services gained preferential access paths, apart from the rest of CSLP's activities. This didn't seem controversial to other interviewees until the CSLP President (in the third interview), disapproved this part in the first prototype clearly stating the goal "Communicating a unique image for all the CSLP's activities".

Lesson 5: *Stakeholders selection is crucial for the quality of requirements.*

Stakeholders selection for elicitation should not firstly involve stakeholder which may impose communication requirements that are not coherent with the overall institutional communication strategy. The elicitation process should instead involve, in a balanced way, the overall organization, using the weight technique to face up with potential contrasts. In the cited case, the higher weight assigned to the President imposed a change in the communication requirements which in turn were operazionalized in a change in the home page design.

Lesson 6: *Keeping track of common soft-goals in the requirements tree of different user categories can help designers to define a balanced communication strategy*

As shown in the case of "Occasional Visitor" and "Professional Users", some user categories can share high level goals but then different requirements can be discovered along the analysis process. Keeping track of the common origin can be useful in the design process in order to clearly distinguish between common access structures and specialized ones. Such a distinction enables designers to define the right communication strategy in both cases. In the former, the design should take into account the multiple targets considering the right trade-offs. In the latter, a specific communication strategy can be defined.

Lesson 7: *In requirements validation activities, in order to gather more and more effective feedback, a "tangible" artefact is preferred.*

Due to the abstract nature of goals and requirements, a discussion around requirements models is turned out to be ineffective or also misleading. In our view there are two reasons against using these artefacts for requirements validation. The first reason has to do with the abstraction level. Discussing about needs is sometime too abstract to obtain a concrete feedback. In the CM project from the first interviews we gathered the need of offering by the Web site a kind of document called "technical answers" (produced by the STC). Also in the second interview (based upon the first requirements model) this new kind of information (absent in the previous version of the web site) seemed as important as other kinds of documents ("opinions", "technical laws", "guidelines", etc.). From these requirements, our first design solution was to design similar access structures for all kinds of documents, including "technical answers" gaining the validation of our idea. Only after the first prototype where this kind of solution was implemented, our idea was rejected and new communication requirements rise out. Thus, we

can formulate lesson 7.1 saying that discussing about possible solutions (conceptual design) is better than discussing about requirements also in the early stage of the requirements validation process. Than, lesson 7.2 adds that using prototypes may considerably improve the feedback from requirements validation, especially in case of communication-intensive Web applications. In general, the adoption of friendly notations can help but, drawing upon our experience where a simplified version of goal-oriented approaches and a communicative notation is adopted, models prove to be good means for analysis but fairly inappropriate for communicating with stakeholders.

Lesson 8: *The iterative interplay between goal/oriented elicitation, high/level conceptual design and prototyping is crucial for refining the requirements set.*

If we consider the requirements set as all the needed inputs towards the software design and implementation, we should use in this phase all the techniques that in this paper have been called requirements models, conceptual design models and prototypes. As stated in the previous lessons, first two techniques are fundamental for the analysis, while the latter serves as perfect communication means for stakeholders validation. In particular, requirements, as intended in this paper, provide an abstraction from specific solution so that the hidden requirements and goals can be elicited, while conceptual design can be used to materialize abstract needs into possible solutions that in turn can be easily transformed into a high level prototype [14]. Fast prototyping systems like [14] (which uses W2000 models as input) aim at reducing time and effort needed to pass from a conceptual model to a visual prototype.

5. CONCLUSIONS AND FUTURE WORKS

We have presented a case study of developing a real-life web application of medium-high complexity for a public administration ministry. We employed a goal-oriented requirements analysis method (AWARE) coupled with a structured approach to conceptual design (W2000). Besides trying to transfer this knowledge to the industrial partners collaborating to the project, we had the opportunity to verify on the fields the advantages and limits of the approach (which will be develop in a further work) and to gather methodological and practical recommendations for future projects. Future work will also consolidate the methodologies in the effort of making them more lightweight (thus easier to teach), more suitable for brainstorming (rather than for describing fully developed solutions) and more usable by developers.

6. REFERENCES

[1] M. Jackson, “The World and the Machine”, a Keynote Address at ICSE-17; in Proceedings of ICSE-17, ACM Press, 1995.

[2] C. Barry, M. Lang, “A Survey of Multimedia and Web Development Techniques and Methodology Usage”, IEEE Multimedia, April-June, 2001.

[3] C. Britton et al., “A Survey of Current Practice in the Development of Multimedia Systems”, Information and Software Technology, vol. 39, no. 10, 1997, pp. 695-705.

[4] F. Garzotto, V. Perrone, “On the Acceptability of Conceptual Design Models for Web Applications”, in Proceedings of ER’03 Workshops, International Workshop on Conceptual Modeling Quality (IWCMQ’03), October 2003 , Skokie, Illinois, USA.

[5] H. Kaindl et al., “Requirements Engineering and Technology Transfer: Obstacles, Incentives and Improvement Agenda”. Requirements Eng. 7(3): 113-123 (2002).

[6] D. Bolchini, P. Paolini: “Goal-Driven Requirements Analysis for Hypermedia-intensive Web Applications”, Requirements Engineering Journal, Special Issue RE03, Springer 2003.

[7] L. Baresi, F. Garzotto, P. Paolini, and V. Perrone, “Hypermedia and Operation Design”. Deliverable D7 of the European IST project UWA (Ubiquitous Web Applications), www.uwa-project.org.

[8] E. Yu, “Modeling Organizations for Information Systems Requirements Engineering”, in Proc. of the 1st International Symposium on Requirements Engineering, RE'93, San Jose, USA, 1993.

[9] G. Booch, I. Jacobson, and J. Rumbaugh, “The Unified Modeling Language User Guide”. The Addison-Wesley Object Technology Series, 1998.

[10] F. Garzotto, P. Paolini, D. Schwabe, “HDM - A Model-Based Approach to Hypertext Application Design”. ACM Transactions on Information Systems, Vol. 11, No. 1, January 1993.

[11]P. Fowler, M. Patrick, A. Carleton B. Merrin, “Transition package: an experiment in expediting the introduction of requirements management”, in proceedings of ICRE 98.

[12] V. Perrone, “Conceptual Modeling of Multi/Cross channel Web Applications”, PhD dissertation thesis, Politecnico di Milano, 2004.

[13] Yu E., Modeling Organizations for Information Systems Requirements Engineering, in Proc. 1st Int. Symposium on Requirements Engineering, RE'93, San Jose, USA, 1993.

[14] Roberto Paiano, Andrea Pandurino, “WAPS: Web Application Prototyping System”, in Proc. of 4th International Conference, ICWE 2004, Munich, Germany, July 26-30, 2004.

[15] Perrone, V., Bolchini, D., Progettazione e realizzazione del sito Internet e Intranet del Consiglio Superiore dei Lavori Pubblici, TR05.1, TEC-Lab, University of Lugano, 2005.

[16] Stutz, C., Siedersleben J., Kretschmer, D., Krug, W., “Analysis beyond UML”, in Proc. IEEE Joint International Conference on Requirements Engineering, RE’02, Essen, Germany, 2002.

[17] Gibson, J.C., “Developing a Requirements Specification for a Web Service Application”, IEEE Joint International Conference on Requirements Engineering, RE’04, Kyoto, Japan, 2004.