TRAMA: A Traceability Analysis Method for (Interactive) Applications

A dissertation presented by
Giovanni Randazzo

Supervised by
Prof. Paolo Paolini

Submitted to the
Faculty of Communication Sciences
University of Lugano

For the degree of
Ph.D. in Communication Sciences

November 2005

Board

Thesis supervisor:	Prof. Paolo Paolini
PhD Coordinator:	Prof. Marco Colombetti
First reviewer (external):	Prof. Giuseppe Visaggio
Second reviewer (internal):	Prof. Carlo Ghezzi

Final version – March 2006

Abstract: if you have no time, read just this

This dissertation presents TRAMA, a TRaceability Analysis Method for (interactive) Applications. The method was originally conceived for being used on interactive application projects, but some experiences showed that its tools and its concepts could be applied in wider domains, including information systems, knowledge management systems, educational applications, etc. This is the reason why the word *interactive* has been put between brackets in the title.
TRAMA is a method, i.e. it is a systematic procedure to perform analysis for documenting tracing, and not a methodology, i.e. a body of rules and postulates which analyse the principles of inquiry in the traceability field. TRAMA is a Requirements Traceability method, i.e. a method to explicitly trace and document relationships between requirements and the different phases of a project's life-cycle, helping ascertain how and why system development products satisfy stakeholder requirements. In particular, TRAMA focus on Design Tracing, i.e. on analysing and documenting the impact of requirements on design elements and the reasons for design choices.

The first attempts in this research went in the direction of finding a way to record the process that brings from requirements to design. In fact, a common opinion in the Requirements Traceability (RT) field is that solution design, i.e. the design of the application solutions, may be derived directly from a requirements refinement activity; some works [Pohl et al., 1994; Egyed et al., 2000] consider design as the result of a refinement process and try to document this process establishing a requirements-to-design traceability. Despite this trend, experiences and case studies[1] conducted for TRAMA highlight a different situation: the design process does not seem to be a fully rational and explicit sequence of actions; according to Arciszewsky [Arciszewsky et al., 1995], "design is an intuition and induction process more than a derivation one". In fact, *at least in the TRAMA case studies*, designers keep requirements in mind as a background knowledge, and they build up the application architecture almost from scratch, as a result of an inductive and a partly intuitive activity. Some requirements remain implicit at the beginning of the project but they are considered in design; often, at the end of the project, designers do not remember the actual reason for that choices. Since requirements are understood as "base information" about how the application should be and why, designers are able to draw a design that satisfies those requirements at least partially, thanks to their skills and to their professional experience. In common industrial cases these relationships are anyway still not explicitly specified; this problem makes it very hard to verify, to evaluate, to revise, and to reuse efficiently design solutions in relation with high-level requirements.

Taking into account these elements, TRAMA has intentionally not been developed to explicitly record the mental processes that brings us from general requirements to concrete design solutions; on the contrary, this method's tracing activity tries to move intuition and induction to a more rational cause-and-effect motivation. This kind of analysis must not repress or stiffen the design process, but it helps us to better understand the reasons for design choices and it forces us to make explicit requirements that are both implicit or unexpressed. In particular, TRAMA is intended to allow a trace documenting activity from five different points of view:

[1] Se also section 3.

- *client validation*: the method helps in gathering a structured argumentation to show to the client that all the needs have been taken into consideration;
- *design versioning*: the method allows analysts to highlight different design areas, identifying the application elements which satisfy the goals of a specific stakeholder;
- *non-traceable design*: the method provides conceptual tools to document the motivations of design elements that do not derive from requirements;
- *"negative" design*: the method allows to keep track of old design choices that have been eliminated or modified during the project;
- *reverse requirements specification*: the method provides tools to check the consistency between design and requirements, to "tune" requirements specification according to the real stakeholders' goals and to extract consistent requirements specification from design;
- *evaluating usability based on design documents*: the method helps in selecting the elements in the design involved for a specific task, in evaluating the quality of the product with respect to the high-level goals and in identifying test procedures which should be rerun to validate an implemented design change.

TRAMA is therefore an effective method to discover, elicit, analyse and document "ex-post" traces, i.e. the method does not record the design process but it helps designers in understanding both the impact of requirements in their projects and the motivations or "sources" of specific design decisions after the design has been drawn. The method is based on traceability matrices which cross requirements with design in a forward direction and design with its sources (requirements, visions, constraints, etc.) in a backward direction.

Requirements-to-Design matrix called RIM (Requirements Impact Model/Matrix) and illustrated in Table 1, can be filled and read both horizontally, highlighting how single requirements are taken into account into the design, and vertically, showing how a single design element satisfies the project requirements.

		DESIGN ELEMENTS			
		Content 1	Content 2	Access path 1	…
REUQIREMENTS-RELATED INFORMATION	**VISIONS**				
	Vision 1				
	Vision 2				
	…				
	GOALS				
	Goal 1				
	Goal 2				
	…				
	REQUIREMENTS				
	Requirement 1				
	Requirement 2				

Table 1. A template for the RIM matrix

Design-to-Sources matrix called DMM (Design Motivations Model/Matrix) and illustrated in Table 2, traces back single design elements to the motivation why a certain decision is relevant for the project. These motivations can be:

- *the designer expertise*, i.e. particular "good design" principles that are part of the designer's skills and that she/he applies in any case;
- *a specific understanding of the domain*, i.e. recurring good solutions in a domain that the designer applies because she/he learnt it in other cases in the same domain;
- *a particular constraint*, e.g. budget limitations, time, technology limitations, etc.;
- *a law obligation*, e.g. copyright issues, personal data treatment, etc.
- *requirements-related information*, i.e. a vision, a goal, a requirements, etc.
- *an arbitrary choice*, i.e. a choice without particular reasons, usually a single detail that could be set any of a number of way, e.g. the structure of a game was in three steps (instead of four or two).

		DESIGN MOTIVATIONS						
		Visions	Goals / Requirements	Designer expertise	Understanding of the domain	Constraints	Law obligations	Arbitrary choices
DESING ELEMENTS	Content 1							
	Content 2							
	Access path 1							
	…							
	Negative design element 1							
	Negative design element 2							

Table 2. A template for the DMM matrix

As a kind of self-standing process, the TRAMA activity workflow is structured as follows:

- *Preliminary plan*: understanding the stakeholders of the traceability analysis, the traceability goals, the constraints (time and budget, related to ROI) and the expected results.
- *Information re-organisation*: understanding requirements and design from documents or from interviews with designers and organising it in terms of structured specifications.
- *Information "normalisation"*: structuring requirements and design information in "normal" terms, bases on a strong methodology (e.g. AWARE for requirements and IDM for design).
- *Elicitation*: surfacing relationships between requirements and design in terms of the impact of requirements on the design ("How these requirements have been considered in the design?") and of motivations for design choices ("Why this solution has been adopted?").
- *Analysis*: tracing relationship and developing the Requirements Impact and the Design Motivations Matrices (RIM and DMM).
- *Specification*: documenting stakeholders, goals and analysis results.
- *Validation*: checking the results with requirements analysts, designers, project managers, and clients.

Benefits in the use of TRAMA are mainly the following:

- TRAMA is a powerful communication means to show to the clients that all their requirements have been considered and how, and that there are no unmotivated elements in the design;
- TRAMA is a structured practice for checking requirements and design consistency for revision, for surfacing missing design elements and missing requirements; the method supports reverse requirements engineering;
- TRAMA is an advanced tool to tune up and re-align a design in the maintenance phase and to assign priorities to design elements;
- TRAMA specifications provide a complete project knowledge summary of requirements-related information, of design elements and of relationships between them, as vital information allowing effective system reengineering, workflow organisation, and more focused verification procedures to be performed.

Acknowledgements

The author is grateful to all the people who contributed directly or indirectly to this dissertation, and in particular:

- the TEC-Lab team, Luca Triacca, Marco Speroni, Nicola Piccinotti, Daniele Gobbetti and Davide Bolchini for sharing contents, ideas, and visions;
- the L@E team for their proactive support collaboration to the traceability analysis process; in particular, I thank Nicoletta Di Blas, Paolo Paolini, Camilla Fontana and Caterina Poggi.
- my family, Rosario, Rita, Gloria, Giusi, Mauro and Carlo for their invaluable support and love
- all the other people in Milano, Como, Lecce and Lugano whom I did not mention but who gave me precious help in this long and winding road.

Lugano, 24 October 2005

Contents

0.Introduction: just a beginning

<<There's no point in being exact about something if you don't even know what you're talking about.>>

John von Neumann

0 Introduction: just a beginning

0.1. The traceability problem

In the software requirements engineering community, traceability has been for a long time studied as a crucial quality factor for software development projects. In particular, a big push in this field was given by the works of Gotel & Finkelstein [Gotel & Finkelstein, 1994] and Pohl [Pohl et al., 1994] in the early 90's. In these last 15 years, traceability for interactive applications has been studied as a part of the requirements analysis process (see also [Jarke, 1998]) and perceived as the activity to trace relationships from and to the requirements specification. A number of models and methodologies have been developed in order to manage and record these information during a project's life-cycle [Potts & Bruns, 1988; McLean et al., 1989; Lee, 1991; Pohl et al., 1994; Gotel & Finkelstein, 1995, Egyed et al., 2000; Grünbacher et al., 2001; Arkley et al., 2002; Dick, 2002].
Thanks to thus research, Requirements Traceability (RT) is viewed as a measure of system quality and it is mandated by many standards governing the development of systems (e.g., IEEE Std 830-1993 and IEEE/EIA 12207). The importance of RT is highlighted by the fact that, for instance, the US Department of Defence has spent about 4% of its IT costs on traceability [Ramesh & Jarke, 2001] – often without getting an adequate value for this money, as traceability in many organisations is haphazard, the standards provide little guidance, and the models and mechanisms vary to a large degree and are often poorly understood. In fact, unfortunately, the penetration degree of these approaches in industrial cases and in companies' workflows is very low. In current industrial practices, RT is still perceived as extra-work, without clear advantages in terms of its *return on investment* (ROI).

> "It is a very valuable but seldom used technique in today's development processes. Traceability analysis is rarer still in the Internet development industry, where it is even more essential" [Leon, 2000].

> "It is rare to find a software project team that can honestly claim full requirements traceability throughout a project, especially if the team uses object-orientated technology" [Ambler, 1999].

The reasons for this situation can be identify in the fact that current traceability methods requires a large amount of time to be spent in order to keep track of requirements along a full project's lifecycle. The effort to maintain traceability is not perceived by developers and managers to be cost-effective. It is considered by management to be extra, optional work, for which insufficient resources are allocated. This position agrees with the opinion of engineers who believe that maintaining traceability information is costing them too much work. Capturing information on the design history of a project may take over 50% of an engineer's time [Wieringa, 1995]. Despite all this work, the benefits of maintaining traceability are not clear: in projects where tools and techniques to maintain traceability are used, problems with traceability are still reported. According to Pinheiro [2002], the less intrusive the tracing activity is the more efficient and accurate the tracing process will be. In large part this occurs because any intrusive process will be rejected or neglected by developers trying to deal with tracing.

Another problem related to the introduction of RT methods in industrial practices could be linked to the fact that some of these methods have a tool-based approach. I refer here to a kind of "myth" that has

never been wrote down but that circulates in several forms in the RT community. According to the Merriam-Webster dictionary, a myth is "a usually traditional story of ostensibly historical events that serves to unfold part of the world view of a people or explain a practice, belief, or natural phenomenon". This myth is a story about a world were the problem of checking the quality of a software application has been solved by the mean of a tracing practice; a world where software developers write down in detail every step of their work, the reasons of every choice, their assumptions, their goals and their beliefs related to the piece application they are working on; a world were these people can spend half of the project time in documenting and recording all these information using complex tools or formal languages to link it each other in a (more or less) meaningful way; a world where, at the end of the day, someone could draw useful conclusions for the quality of the application from this huge network of relationships.
In pragmatic terms, most of the currently available RT approaches give to the traceability problem these answer: "while it is difficult maintaining the huge mass of dependences among the many objects produced by a large software system development effort, some current approaches require the use of a software tool to become usable and manageable; so, bring all your documents, specifications and artefacts produced during the project, record it into a support tool and trace all the relationships that you consider meaningful; other relationships will be automatically created by the tool itself" [typical traceability solution]. Unfortunately, this tool-based solutions do not consider that in the actual practice, some specifications are not taken, some documents are not written or are written after the application is implemented and that some "knowledge" (about reasons, beliefs, etc.) is never recorded or explicitly considered. Furthermore, current tools have problems in maintaining relationships concerning artefacts expressed in natural language, often ambiguous, or artefact created independently by non-interoperable tools and that evolves autonomously. Besides, some tool-based practices have access problems for the user (communication problems) and methodologies are often not clear, not complete or too formal for their adopters.

0.2. Requirements to Design

The research presented in this dissertation proposes a method to keep trace of "requirements-to-design" traces. With "*design*" I mean here the conceptual, high-level description of the functionalities of an application and of the system's solutions to strategic needs. In the last years, some RT models proposed software-based approaches directed towards the automation of requirements to design tracing processes (see also [Dick, 2002]). These models are based on the assumption that a system design activity moves in a explicit and structured way, considering one by one possible solutions to requirements and refining these requirements in a continuum to design elements. On the contrary, the case studies conducted for this research[2] and "real world" experiences performed with professional designers in the last 3 years, highlight that requirements do not fade naturally into system solutions; requirements and design stand into separate and different conceptual area instead: the firsts are in the space of problems, the latter in the space of solutions. During a design process there are not explicit relationships between these two "spaces"; in fact, designers take in account requirements as background information but then they design the system following their own skills and competences. In other words, in order to consider stakeholders

[2] See also section 3.

needs, requirements are understood and "absorbed" by designers as a whole. Therefore, the design process is not a fully rational and explicit sequence of actions; designers build up the application architecture almost from scratch, as a result of a an inductive[3] and in part intuitive practice. In fact, in such a decision making activity not only the general requirements knowledge described before is involved, but also a wider knowledge about the specific application domain, some project constraints, "good design" and usability principles, designer skills, etc. – in most of the cases in a implicit or almost unconscious way. On the other hand, design structure and requirements structure are inhomogeneous, while a requirement does not impact on a single design element but on a number of design elements and on their interactions. Since requirements are understood as base information about how the application should be and why, skilled and experienced designers are able to draw a design that satisfy in a certain measure those requirements. In common industrial cases these relationships are anyway still not explicitly specified; this problem make very hard to verify, to evaluate, to revision and to reuse efficiently design solutions in relation with high-level requirements. This fact includes that in some cases it is not clear why a certain piece of the system has been developed, and how the product answer to needs stressed by strategic requirements; in other words it is not clear how to evaluate, to validate or to motivate the *quality*[4] of the final product. Another problem raised out from this situation is that revision activities are very hard to bring on, while it is not clear the impact of requested changes and their effect on the application requirements compliance.

A proposal to find a reasonable and usable solution to these problems is presented in this dissertation as TRAMA, a TRaceability Analysis Methodology for (interactive) Applications. The method is a first attempt to reduce the complexity of current methodologies considering requirements-to-design relationships between objects of adequate granularity; TRAMA can be applied even in case of lack of documentation: it is also useful to write an *ex-post* specification of the work done; TRAMA can be used without any specific software tool: objects are related each other using simple matrices; TRAMA analysis discover or highlight the main reasons for conceptual design choices and which is the impact of a goal or of a requirement on the application. This traceability approach does not focus on modelling the process of evolving requirements into design, but it pretend to provide to designer an effective tool conceived to discuss and analyse the design choices after they have been taken, in order to refine it according to the main requirements and in order to eliminate unmotivated elements. TRAMA consists in a structured analysis process, in a general conceptual model of entities and relationships to trace and in a set of conceptual tools supporting traces inquiry, analysis and documentation. The case studies where TRAMA has been applied have shown that the methodology is easy to use and to learn, and that the tracing activity is reduced to an average of the 5% of the time spent for the entire project.

[3] See also [Arciszewsky et al. 1995].
[4] For an explanation of the concept of quality, see also Cantoni et al. [2003].

0.3. Dissertation's structure

The TRAMA research presentation is organised in 6 main sections:

1 TRAMA research foundations

This section illustrates the main concepts related to the traceability field, with a particular focus on the Requirements Traceability (RT) domain; the section describes the scope, the focus, the hypothesis and the goals of the research treated in this dissertation as well.

2 Review of related works

This section proposes a review of the state-of-the-art in the RT field, with a particular emphasis on those works that influenced in some way the TRAMA research. A short review of the main traceability conceptual tools and software tools is also provided. Finally, some open problems in current traceability practices are highlighted.

3 Case Studies

In this section all the case studies on which the different versions of TRAMA was applied are described. Since this is an empirical research, each experimentation bring key elements to improve the method, to modify it, to refine it, to test it and to provide at the end a general approach. The sequence of case studies traces therefore an history of how TRAMA has been developed, as well as examples of use of the method.

4 The TRAMA method

In this section the current version of the TRAMA method is described, both as a process, i.e. as a sequence of actions divided in phases, and as a tracing approach, i.e. as a model including conceptual structures, tools, purposes, etc. In the section is presented first a tracing activity workflow allowing TRAMA to be properly applied; then, a TRAMA approach is described in terms of purposes, processes, conceptual trace model and tools. Finally, the benefits of the method and some of its limits are discussed.

5 Teaching TRAMA

This section presents the modules, the activities and the courses conceived to teach the TRAMA method to different targets and in different situations. In particular, three modules and four courses are provided.

6 Conclusions & Future works

This section wraps up the proposal in its key elements, highlighting the problems and the solutions described in the dissertation, its sources, its hypothesis, its main characteristics and benefits and limits of the approach. Finally, an input for future research is provided.

ANNEXES

Annex 1 - Educational Material

In this annex, all the slides packs and the slides for the courses related to the TRAMA educational plan are collected.

Annex 2 - TRAMA in a nutshell

In this annex, a short document describing TRAMA and 10 slides summarising the approach are provided.

Annex 3 - Case studies reports

In this annex, all the traceability reports related to the case studies presented in section 3 are collected.

1.TRAMA research foundations

"Walking on water and developing software from a specification are easy if both are frozen."
Edward V. Berard

Abstract

This section illustrates the main concept related to the traceability field, with a particular focus on the Requirements Traceability (RT) domain; the section describes also the scope, the focus, the hypothesis and the goals of the research described in this dissertation.

Traceability is the ability to explicitly trace and document relationship between the requirements phase and other phases of a project life-cycle. In the literature, a major distinction is highlighted between pre- and post-Requirements Specification traceability and between backward and forward traceability. A number of purposes for a tracing activity may be highlighted, according to the point of view of different actors involved in a project life-cycle: the client, the project manager, the designers, etc. A general meta-model describing a tracing approach can also be described, highlighting its purposes, the processes supported, the tools used and the conceptual trace model at the heart of the approach.

The TRAMA approach proposed in this dissertation is a method for tracing designs of interactive applications, supporting post-Requirements Specification Traceability in both a forward and backward direction. The research method bases on a empirical approach: an iterative sequence of experimentations and case studies will modify, improve, refine and test the method to provide at the end a general model.

1.1. Contextualisation

1.1.1. Traceab… what?

Traceability is the degree to which a relationship can be established between two or more products of the development process [IEEE, 1990]

Traceability can be simply defined as the ability to explicitly trace and document relationships between the different phases of a project's life-cycle. A specification can be considered as "traceable" if the origin of each of the artefacts or objects described in such a specification is clear and if it facilitates the referencing of each object in future development or enhancement documentation [Gotel & Finkelstein 1994]. A great contribution in research for traceability comes from the Requirement Engineering field (RE); in this context the definition of Requirements Traceability (RT) can be adopted. According to Palmer [1997], RT helps ascertain how and why system development products satisfy stakeholder requirements. In short, RT is the ability to determine which documentation entities of a software system are related to which other documentation entities according to specific relationships; from this point of view traceability can be seen as the mean whereby an analyst is able to discover from one hand the impact of such entities on the application and from the other hand the reasons or the "sources" of specific entities [Spence & Probasco, 1998]. Formally speaking, a traceability system can be defined as a semantic network in which nodes represent objects (also stakeholders and sources), among which traceability is established through links of different types and strengths [Ramesh et al., 2001]. From a more "dynamic" point of view, RT is defined as the ability to follow a specific item at input of a phase of the software lifecycle to a specific item at the output of that phase; RT enables each requirement to be traced to its origin in other documents and to the software components satisfying the requirements. Traceability gives essential assistance in understanding the relationships that exist within and across various artefacts produced during the acquisition process. These relationships help establish traces of the process through which critical acquisition decisions are made and help ascertain how and why outputs of an acquisition process satisfy stakeholder requirements. According to Hamilton & Beeby [1991], traceability can be viewed as the ability to discover the history of every feature of the outputs of an acquisition activity so that the impacts of changes in acquisition requirements can be identified.

In literature a major distinction is highlighted between pre- and post-Requirements Specification traceability and between backward and forward traceability [Gotel & Finkelstein, 1994].

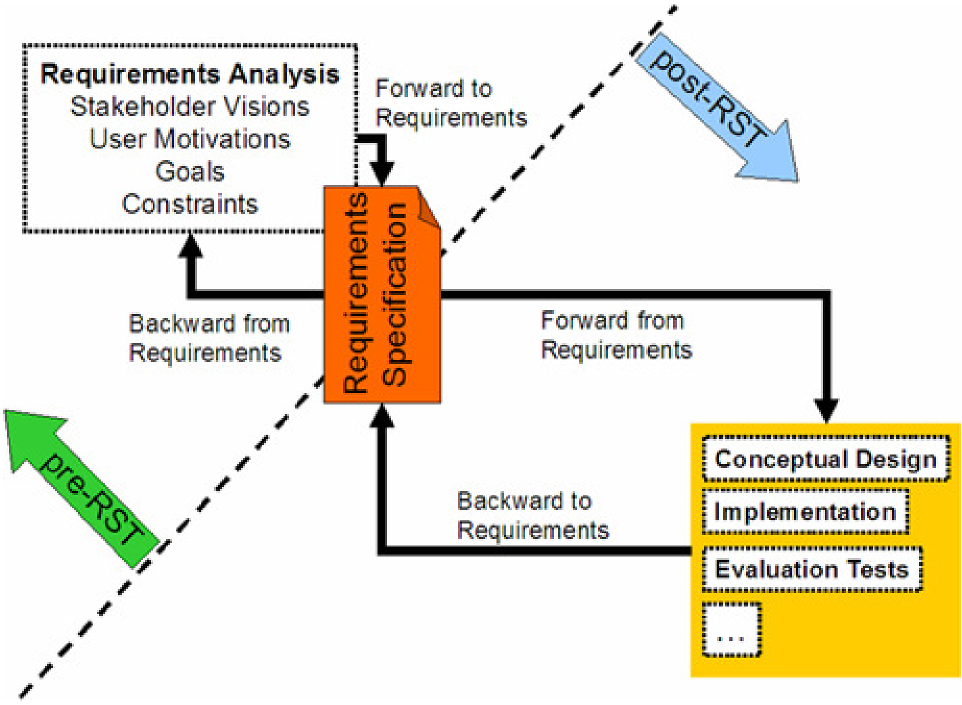


Figure 1. A simplified picture of traceability types

Pre-Requirements Specification traceability

According to Gotel & Finkelstein [1994], pre-Requirements Specification traceability (pre-RST) is concerned with those aspects of a requirement's life prior to its inclusion in the requirements specification (i.e. requirements production and refinement). It is a technique that attempts to document the rationale and socio-political context from which requirements emerge, thus linking the business world with that of information technology [Jarke, 1998]. Pre-RST also serves to answer questions that arise during the project's life-cycle, including: "Who is responsible for including this requirement?", "To whom should I refer to for more information?", "Who was responsible for copying this information into this document?", and "Was this requirement a result of a meeting of stakeholders or just one individual?" [Gotel & Finkelstein, 1995]. Pre-RST facilitates the reopening of previously closed specifications, tracing back to the sources of requirements, and then the (possible) reworking of a specification in the forward direction; sources of requirements may be the following:

- *Stakeholder Visions*: stakeholders are those who have a direct interest in the success of the website (e.g. clients, sponsors, representatives, opinion makers, etc.); stakeholder visions are the assumptions of a stakeholder which dictate his/her "weltanshaung" on the project [Bolchini et al., 2005b].
- *User Motivations*: they shape the emotional, psychological, social or individual elements which can trigger a person (a final user) to use an interactive application [Bolchini et al., 2005b].
- *Goals*: they are defined as high-level targets of achievement for a user or a stakeholder; goals may represent a wished state of affairs (for main stakeholders) or a wished experience (for users) and may arise from visions or motivations.
- *Constraints*: they are defined as those elements that implicate a restriction on the degree of freedom the requirement analyst have in providing a solution; constraints can be economic, political,

technical, or environmental and pertain to project resources, schedule, target environment, or to the system itself.

In literature, major contribution to pre-RST comes from Contribution Structures [Gotel & Finkelstein, 1995] and PRO-ART [Pohl et al., 1994] methodologies; a deeper explanation of these two works will be carried on in section 2.

Post-Requirements Specification traceability

Post-Requirements Specification traceability (post-RST) is concerned with those aspects of a requirement's life which result from its inclusion in the RS (i.e. requirement deployment and use). This kind of traceability provides a way to elicit and discover the impact of requirements and how requirements have been taken into account on the following project elements:

- *conceptual design*: high-level definition of the information structure, of the features and of the services/capabilities that the application will own;
- *technical design*: in-detail definition of the software (and/or hardware) components the application will be made of;
- *experience design*: definition of all the elements contributing in building the user experience, including organisational concerns, technical set-up and use scenarios;
- *implementation*: it's the "tangible" part of the application, i.e. classes, routines, lines of code, interfaces, etc.
- *tests*: including technical test verifying if the application works properly, usability tests and accessibility test.

In literature, major contribution to post-RST comes from CBPS methodology [Egyed et al., 2000] and from the idea of rich traceability applied in the DOORS tool [Dick, 2002]; a deeper explanation of these two works will be carried on in section 2. The TRAMA method is an example of post-RST.

Backward traceability

Backward traceability records information and data on the past history of the product, providing knowledge about the sources of a specific element (e.g. a requirement, a design element or a piece of code) and about the reasons of a specific decision in the previous project items (e.g. in goals, requirements, etc.). In other words, backward traceability to previous development stages depends upon each requirement explicitly referencing its source in previous documents.

- Backward from requirements - This trace type lets the analyst verify that the system meets the user community's needs, an important consideration attempting to justify the budget; therefore it is important to understand the source of requirements (e.g.: a requirement from a key customer likely has a different priority than one from a junior programmer).
- Backward to requirements - This trace type verifies compliance of design, software or tests built to requirements; this approach do not take into account software features that cannot be traced back to requirements because their source is another element such as a constraint or an organisational aspect.

Forward traceability

Forward traceability explains what will happen to a certain product and all the processes and output that the product in question went into. Forward traceability to all documents spawned from the software requirement specification depends upon each requirement in the software requirement specification having a unique name or reference number.

- Forward to requirements - This trace type maps stakeholder needs, visions and goals to the requirements, so that the analyst can determine the impact to requirements as needs change; changing needs, either from a change in strategy, an increased understanding of the problem domain, or an environmental change, is a reality of the software industry and an important issue that must be managed effectively.
- Forward from requirements - With this trace type, the analyst assign responsibility for fulfilling a requirement to the design or to the various system components that will implement it, letting the responsible ensure that each requirement is fulfilled.

Nowadays, it is widely agreed that tracing requirements is essential in developing large systems and RT is intended to become an important feature of software systems. Many standards governing the development of such systems (for example, IEEE Std 830-1993 and IEEE/EIA 12207) require the development of RT documents; the US Department of Defence has produced standards with the same goal (e.g., MIL-STD-2167-A and MIL-STD-498) and spends about 4% of its Information Technology (from now on: IT) costs on traceability [Ramesh, 2001] – often without getting an adequate value for this money, as traceability in many organisations is haphazard, the standards provide little guidance, and the models and mechanisms vary to a large degree and are often poorly understood.

1.1.2. Quality of Service

The concept of RT is deeply related to the Quality of Service (QoS) and to the Software Quality (SQ) concerns [Kenny, 1996]. Quality *per se* can be seen from two points of view which are strongly intertwined and which affect each other:

- *ad intra*: quality is considered by mean of the intrinsic characteristics of the application (e.g. performance, accuracy, up-to-date);
- *ad extra*: quality is the correspondence between services offered and stakeholders'[5] goals; it can see as the combination of the quality of the user experience, the user satisfaction and the main stakeholder's[6] satisfaction

[5] The word *stakeholder* is here used as the sum of main stakeholder and of final users (see next footnote)
[6] A *main stakeholder* is anyone who has an interest on the success of the application, e.g. clients, sponsors, decision makers, etc. [Bolchini, 2003]

EXAMPLE 1.1. Let's try to consider the quality of a chair[7]. I will take for instance my office chair: it is a standard, plain office chair, with wheels and a back. Is it comfortable? Yes, it is: it let me work without pains for an entire day. Is it stable and solid? Yes, it is: it can support more than 100 Kg. Therefore, it can be said that the intrinsic quality of my chair is good. But what if I should use the same chair to relax at evening watching TV? Maybe I would not feel it very comfortable. And what if it should be used in a western movie in a scene where the good crashes a chair on the head of the bad? Please, don't do it. The extrinsic quality of this artefact is therefore deeply related to the use of the object and to the goals and motivations the users of that object have.

EXAMPLE 1.2. There is another case where the comfort ability of a chair doesn't make its quality. In a US company, the time spent for meetings was too much; in fact, people usually drank coffee and chatted for part of the time, sitting around the meeting table. One day, the CEO of that company decided to change all the chairs in the meeting rooms with very uncomfortable ones. The result was that meetings time reduced of 50%. In this case, the quality of the chairs was good in relation to the goals of the main stakeholder.

In the EX1.1 it can be introduced the concept of usability, i.e. "the effectiveness, efficiency and satisfaction with which specified users can achieve specified goals in particular environments" (ISO 9241-11)[8]. My chair is usable if I use it as office chair, but it is extremely unusable in the western movie. In the EX1.2 the new chairs have a very low level of usability in a strict sense, but their quality can be considered anyway good: they match with the needs of the main stakeholder. Therefore it can be introduced a more extensive definition of quality; according to Kenny [1996], Software Quality is:

- the totality of features and characteristics of a software product that bears on its ability to satisfy given needs, for example to conform to specifications;
- the degree to which software possesses a desired combination of attributes;
- the degree to which a customer or user perceives that software meets his or her composite expectations;
- the composite characteristic of software that determine the degree to which the software in use will meet the expectations of the customer.

In other words, it can be said that quality is a multifaceted characteristic of an application; the quality degree of a project may depend on services and features provided, user satisfaction and context of use, customer and main stakeholders satisfaction, compliance with strategic goals and impact on the organisation. Therefore, it becomes crucial to keep in a global picture the relationships between these elements: traceability can improve the quality of the systems development process, providing a global picture under control.

---

[7] Adapted from Cantoni et al. [2003]

[8] For a further explanation of the concept of usability and of how it is concerned with quality, I refer to Cantoni et al. [2003].

1.1.3. Traceability purposes

Traceability can be seen as a powerful communication mean, whereby software producers can prove to their client that the requirements have been understood, that the product will fully comply with the requirements and that the product does not exhibit any unnecessary feature or functionality. Traceability can also facilitate communication among various stakeholders involved: project manager and project planner, customer, requirement analyst, designer, verifier and maintainer. Each one of these actors has his/her own view on traceability and can use traceability information for different purposes, as described in the following lines [Dömges, 1998; Gotel, 1994] and in Figure 2.

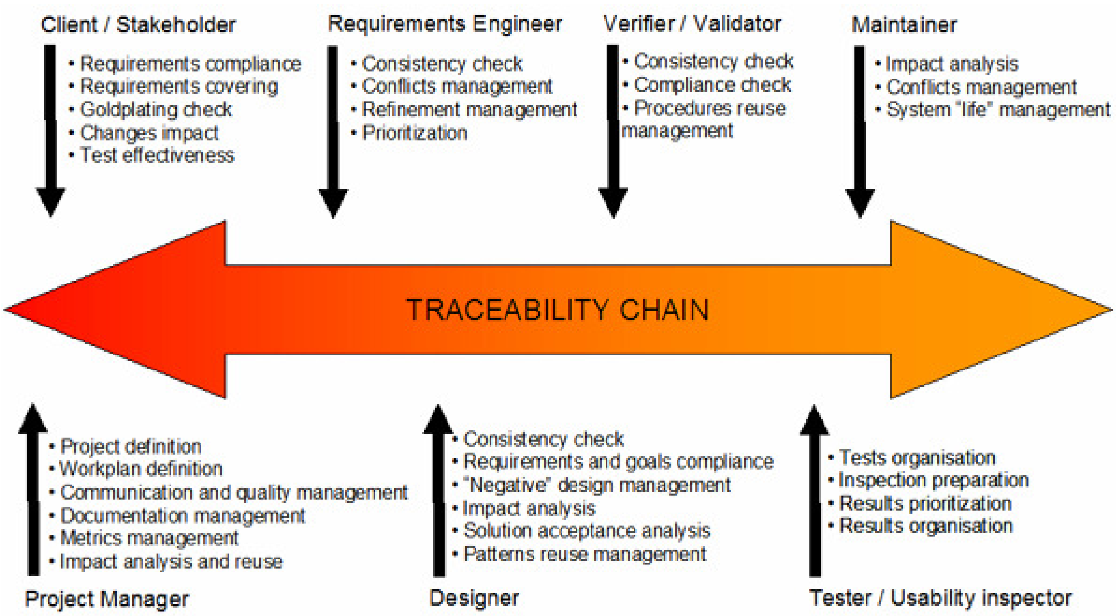


Figure 2. Traceability as communication mean and its use among the different actors involved in a project life-cycle

(i) Client/Customer/Stakeholder

Clients have in most of the cases a certain number of problems in evaluating the quality and the effectiveness of a software application *a priori*, i.e. before its effects have been produced. Usually there is a knowledge and understanding gap between stakeholders and the development team; clients can hardly see how and where the applications provided may fit to their needs and goals. Traceability analysts can guide these people in evaluating such applications; traceability is a communication "bridge" between a client (usually with marketing or economics background) and a software house, a web agency or anyway the internal development team (with engineering or informatics background) that allows to check the following elements:

- *Requirements compliance* – Traceability can shows the relationships between strategic goals, requirements and solutions in the application, allowing clients evaluating the compliance degree of the product with their needs. Therefore, the overall quality of the application can be understood without any need to consider single technical or software details.
- *Requirements covering* – Relationships between requirements and elements or pieces of the application may highlight the progress state of the project. Clients can understand which percentage

of the stated requirements are met and which part of the job is completed. A thorough traceability analysis may also provide stakeholders that all the strategic goals have been satisfied and how the application will address to their needs.

- *Goldplating check* – According to Dömges [1998], goldplating is adding superfluous features that aren't motivated by actual requirements; since requirements are not the only source for design choices[9], this definition can be limited excluding those elements that becomes from constraints, stakeholder visions, etc. Goldplating is therefore defined as the presence of features that are not motivated by any explicit reason. Traceability analysis highlights goldplating by linking all the application features with their motivations; if no explicit reasons are specified, two options can be considered: there is a reason but it has not been make explicit yet or it is a case of actual unmotivated feature. This kind of analysis lets clients ascertained that (costly) goldplating have been avoided because all components of the implementation can be traced to at least one reason, but it avoid also the risk of eliminate useful but only implicitly motivated features.
- *Changes impact* – It is not unusual to observe that after the end of a project, clients may ask to developers further changes to the applications. Reasons can be identify in lack of proper needs analysis or in lack of proper communication to the client. Anyway, this is something that happens in most of the cases for interactive application, and in particular for web applications. In this context, traceability analysts can help clients in evaluating the consequences of their requests, i.e. the impact of a requested change on the entire application and on the way the system meets stakeholders goals.
- *Tests effectiveness* - If the tracking information system records which requirements are satisfied by which parts of the implementation, and which tests must be performed to ascertain the "presence" of a requirement, then clients can better understand the value, the results and the implications of technical tests and usability evaluations. In addition, acceptance testing can refer directly to the user requirements being tested for, making it relevant from a stakeholder point of view.

(ii) Project manager and project planner

Project managers can use the traceability information to control project progress. In a project life-cycle, project managers are supported by a correct traceability approach in accomplishing their different tasks:

- *Project definition* - An early traceability analysis during the work definition allows project managers to control that the work team and the client have the same perception of the project; this includes the delivered and the not delivered artefacts, how much does it costs, who will perform the work, how the work will be done and which benefits will be achieved.
- *Workplan definition*, development and managing - Matching goals with design elements is crucial to organize efficiently the time plan, giving priorities to the development of the core elements of the application and avoiding useless or superfluous features. Project managers can prevent conflicts and check the progresses of the different tasks related each other, with test procedures and with the main strategic goals. Conflicts between requirements can be discovered earlier and unexpected product delays avoided.

[9] This concept will be extensively treated in section 4.

- *Communication and quality management* – Traceability is a powerful communication mean with clients, providing to project managers arguments and evidences of the project quality in terms of satisfaction of goals, needs and expectations [Palmer, 1997].
- *Documentation management* – Traceability analysis allows complete and refined documentation and specifications; the traceability chain[10] provides a preferential way to order, link and organize each document or deliverable.
- *Metrics management* – All the relationships traced between parts of the application, features and services on one hand, and test procedures on the other hand, becomes crucial to give to project managers quantitative data to identify trends, support decisions and as indication of the good health of the project.
- *Impact analysis and reuse* – Project planners use a tracing approach to perform impact analysis; requirements can be tracked to determine the impact of a required change on the entire project, on the workplan, on other feature of the application, on goals, etc. Requirements not yet satisfied by the implementation can be collected, and the work to be done to satisfy these remaining requirements can be estimated. Future systems will have reduced development time and effort because past implementation decisions can be reused.

(iii) Requirements engineer

Requirements engineers keep and elicit visions, strategic goals, constraints, user profiles, etc. from stakeholders and motivations, user goals, etc. from users[11]. A pre-requirements specification traceability analysis is needed to keep these relationships between stakeholders and goals, between users and goals, between goals and sub-goals in the refinement process and between sub-goals and requirements[12]. These people use the traceability information for the following purposes:

- *Consistency check* – Traceability analysis is used by requirements engineers to keep the consistency between the different information they consider, and in particular between requirements as indications for the design on one hand and goals and constraints as source and motivations for requirements on the other hand.
- *Conflicts management* – Conflicts between goals are usual, in particular between stakeholders goals and user goals. Traceability helps the analyst in finding a good compromise between conflicting goals, considering the relevance of stakeholders that own such goals and evaluating the impact that changes may have on other goals, sub-goals or requirements.
- *Refinement management* – During goals refinement activities it is crucial to keep all the relationships between high-level goals and derived or refined sub-goals. Traceability may also help in keeping an history of all the refinement changes performed in different moments of the project life-cycle and for different reasons (technology changes or constraints, budget constraints, timing, etc.).
- *Prioritization* – The traceability chain links as in a flow, stakeholders with goals and requirements; if all the relations are kept and updated, the requirements analyst can give a relative priority to each

[10] According to Triacca [2001] a traceability chain shows the relationships traced between the components of an application framework, i.e. goals, requirements, tasks and design, and their influence on subsequent components.
[11] For the definitions of vision, motivation, etc. see also Bolchini et al. [2005b].
[12] Some structured methods (such as i*, KAOS or AWARE) provides conceptual tools to document the relationships between a stakeholder and the goals it express and between a requirement and the goal(s) it fulfil.

requirement or to groups of requirements that meet the needs of a certain stakeholder; requirements related to more relevant stakeholder should be considered with higher priority respect to others.

(iv) Designer

Designers of software products are responsible to shape the information architecture of the application, considering the content structure, transitions between pieces of contents, interactive features, access to contents and features and navigation architecture. To keep the consistency of the entire project, designers take in consideration goals and requirements highlighted during the requirements analysis; nevertheless, a major part of the final design has other motivations than requirements: for instance, some elements could have pure technical reasons or being just based on "good design" principles. Usually, part of these reasons are not recorded and part are not explicitly perceived or understood. A traceability analysis allows eliciting hidden or unconscious knowledge and helps designers to show that the elements indicated in the conceptual design are not unusual, unnecessary or unmotivated. In particular, designers use the traceability information for accomplish the following tasks:

- *Consistency check* – A tracking information system should record the results of design, the justification of the results, the alternatives considered, and the assumptions made in a decision; therefore a traceability analysis prevents from consistency problems between different parts of the project and may help in solve inconsistencies with technical implementation or with strategic goals.
- *Requirements and goals compliance* – Designers use traceability to understand dependencies between the requirements and to check whether all requirements are considered by the design; therefore, they can more easily verify that a design satisfies the requirements or not. If a design element is not directly liked to a specific requirement, they can find arguments in traceability documents to justify their decisions in a more general relation with strategic goals or with non-functional requirements. A traceability approach force designer to ask themselves the "why" question (before the client do it…).
- *"Negative" design management* – With "negative" design [Randazzo, 2004] I refer to the design elements that for any reason have been rejected or eliminated from the application. In most of the cases, the knowledge of which are these elements and why they have been deleted is crucial to measure their impact on the project. Traceability analysis support designers in keeping these kind of "design history", avoiding time-consuming features that for the same reasons would be rejected and considering alternate solutions for other similar cases.
- *Impact analysis* – Traces between the different elements of a project allow designers to evaluate possible consequences for changing a design feature in terms of compliance with requirements and goals or in terms of needed changes in implemented prototypes and applications. From another point of view, designers can understand the impact on the design of a change in requirements and take consequent decisions. Designers can use traceability information also to estimate the impact of a change in available implementation technology on the design assumptions and hence on the design alternatives.
- *Solutions acceptance analysis* – Starting from traceability documents, designers can understand the reasons why a certain design was accepted and another rejected, even when the design was

produced long time ago by a no more present designer. These reasons may relate design decisions to non-functional requirements, to unexpressed constraints or to more general stakeholders' visions.

- *Patterns reuse management* - A traceability chain relates a specific need with a certain design solution; if the design is accepted, such a solution can be considered as a good one at least from a stakeholder point of view. Therefore, designers may reuse design components for similar needs in other projects because the assumptions under which the component will work are recorded in the traceability report. Besides, the tracking information system may become a kind of "corporate memory", i.e. a library of solutions patterns and a way to refers to specific solutions in a fast and direct way; this "corporate memory" can be used in the work team to speed up decision-making in future development projects.
- *Design revision* – Traceability documents keep the knowledge about the relationships between requirements and design in a structured way; if there is a need to tune up or to revise a former project, designer can understand and/or remember previous decisions taken and properly "adjust" the application.

(v) Verifier / validator

Verifiers in large projects provides a further consistency check of the final application; they base their job on traceability information to verify that all the strategic goals have been properly satisfied, that all the requirements have been taken into account, that design doesn't have goldplating, that software meets with design specifications and that the application has been properly tested.

Validators use traceability relationships between requirements and test plans to prove that the system "completely" meets the needs of the customer. In addition, test procedures can be identified that should be rerun to validate an implemented change. This saves test resources and allows the schedule to be streamlined.

(vi) Tester / usability inspector

Testers perform a detail evaluation of the system's technical performances: the application should not "crack" or generate errors in any condition of use. Usability inspectors are concerned with the application "easy of use": they check that the declared goals can be reached by users by the mean of the application in a efficient end effective way.

Testers can use traceability information from two points of view. First, they can perform their tests in a more systematic way; e.g. they can test features in relevance order or organize tests grouping features by stakeholder or by goal they meet. Second, in case of problems surfaced during the tests, they can indicate which exactly are the pieces of software or the design elements to review; they can also suggest a priority order for these problems based on the impact they have on the satisfaction of strategic goals.

Usability inspectors have to taken into account high-level goals of the product, evaluating it according to its real scope. Keeping traces between these two activities can help usability inspectors performing a more effective and efficient evaluation and showing that the main goals have been consistently tested. Usability experts can also use entire parts of the traceability analysis to plan and prepare their evaluation: in fact, inspectors need to know dependencies between user profiles, goals and features in the application to properly test the usability of that solutions. As for the testers case, to usability problems can be assign a priority and the inspectors can indicate on which element of the project they have an impact.

(vii) Maintainer

Maintainers "keep alive" the application; this is particularly true for interactive and web-based applications, where key success factors are to be up-to-date and always to adapt the communication and the business channels to new user or stakeholders' needs.

Maintainers use the traceability information to decide how a required and accepted change will affect a system, i.e., which modules are directly affected and which other modules will experience residual effects. Documenting an engineer's design rationale helps the maintainer to understand the system. If a required change is implemented, understanding the existing solution structure helps to prevent the system from degrading. A maintainer can this way estimate the impact of a change in requirements on other requirements, discover conflicts dependencies, estimate the impact of a change in requirements on the implementation and estimate the permissibility of a change in implementation with respect to (unchanged) requirements.

1.2. Tracing approach

According to the Merriam-Webster dictionary, a method is "a way, technique, or process of or for doing something" and "a systematic procedure, technique, or mode of inquiry employed by or proper to a particular discipline or art". In order to conceive, to apply or to understand a traceability method, the only knowledge about the meaning of the word "traceability" itself and about how traceability can be applied by different actors is not enough. A wider understanding about the elements composing a general traceability method and about processes and tools characterizing it is needed. Such a knowledge is indicated in literature with the name of *tracing approach*.

The concept of tracing approach (TA) refers to a generic term for methods, techniques and models enabling tracing activities [von Knethen & Paech, 2002]. A general TA, e.g. a traceability method, is characterised by (a) the purposes the activity may have, (b) the processes involved in the tracing activity, (c) the conceptual trace model on which the activity is based and (d) one or more tools enabling, facilitating or documenting such an activity [von Knethen & Paech, 2002]; Figure 3 summarises these four aspects shaping a kind of meta-TA.

(a) Purposes

As it has been shown in the previous paragraph, different stakeholder may have different goals and benefits in adopting a tracing technique. Therefore, different methods may differ because of stakeholder and goals supported. In other words, a tracing approach can be characterise by:

- *Who wants to trace, who is the user of a traceability chain* - Many different stakeholders (project sponsors, project managers, analysts, designers, maintainers, end users, etc.) are involved in the system development life cycle. The traceability needs of these stakeholders differ due to differences in their goals and priorities, and many problems of traceability stem from these differences in interest and understanding [Ramesh et al., 1993].

- *Why and when traceability is provided* - Traceability activity can be performed in different moments of a project's lifecycle: during the system evolution (to refine the application design) or after the system evolution, i.e. at the beginning of a new project for a design tuning activity or for a reengineering process.

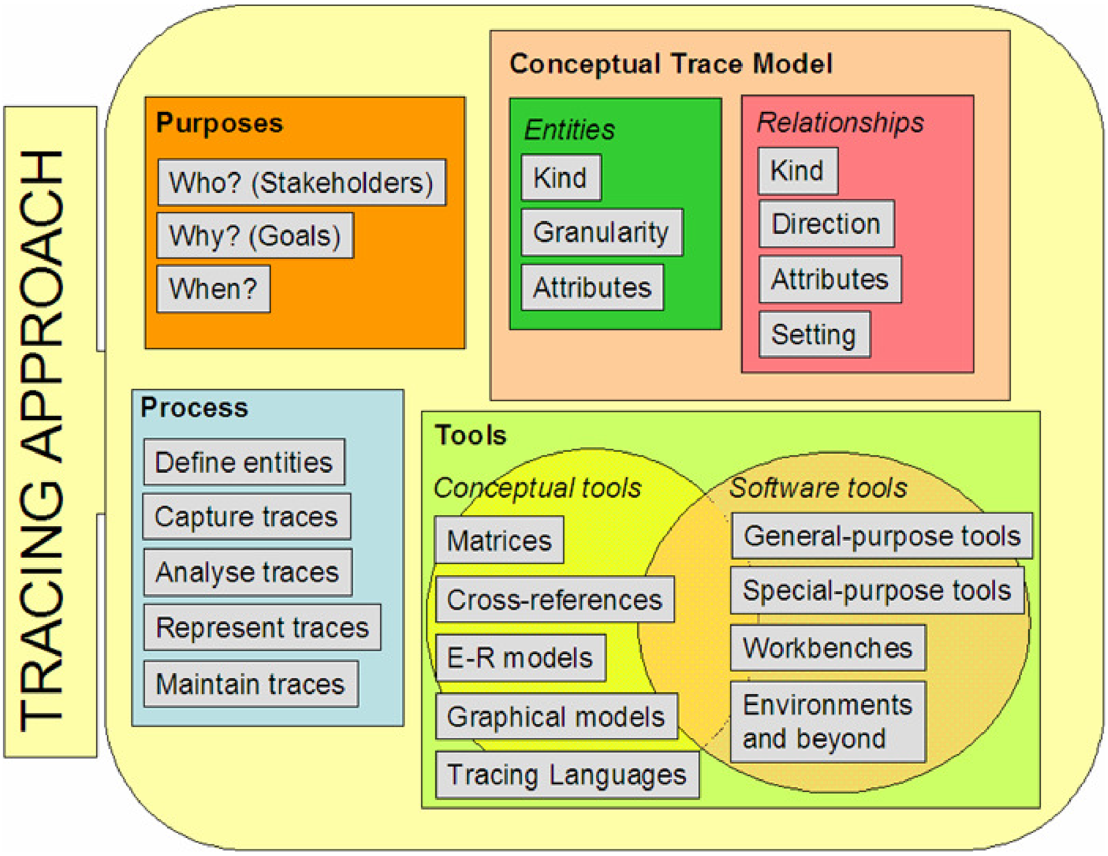


Figure 3. Core concepts of a tracing approach, adapted from [von Knethen & Paech, 2002]

(b) Process

Different tracing techniques may be characterised by the kind of activity or task supported; the process can involve one or a combination of the following:

- *Define "entities"*, i.e. elicit and define with stakeholders the objects to keep related each other, e.g. requirements, design elements, test procedures, etc.
- *Capture traces*, i.e. trace the relationships between the different elements of the trace model.
- *Analyse traces*, i.e. interpret the relationships and highlight problems or weaknesses raised out from traceability, e.g. poor requirements covering, useless or unjustified design elements, etc.
- *Represent traces*, e.g. provide tools, procedures, checklists, etc. helping stakeholders and analysts in document, illustrate and display the traceability knowledge; summarise the results in a traceability report.
- *Maintain traces*, e.g. keep tracing information up-to-date as far as new decisions are taken or any change is made to the system status.

(c) Conceptual trace model

A conceptual trace model, also called reference model [Ramesh & Jarke 2001], defines what "trace entities" and "traces" are and which traces should be captured. Therefore, a conceptual trace model determines what is relevant and it identifies and formalises which aspects of the system are to be recorded and worked with. Such a model should provide also guidelines to identify a common way of dealing with the traces. Two sub-concepts, "entity" and "relationship", refine the concept.

(c.i) Entity

An entity is an object or an item of the traceability activity that represents the input or output of the system development process. According to Spence & Probasco [1998], a traceability item (i.e. an entity) is defined as "any textual, or model item, which needs to be explicitly traced from another textual, or model item, in order to keep track of the dependencies between them". Examples of various types of entity include goals, requirements, assumptions, designs, system components, decisions, rationale, alternatives, critical success factors, etc. A trace may also capture the human co-operation in the design process, that is, how stakeholders contribute to the development. The entities that should be traced are determined by the purposes supported by the TA; according to von Knethen & Paech [2002] the concept considers three aspects of an entity:

- *the kind of the entity*: it describes which software documents (e.g., requirements, test cases, design elements, etc.) should be involved in the conceptual trace model; examples of kinds of entity are: temporary work products and permanent work products [Lindvall, 1994]; requirements, specifications, and implementation [Ramesh & Edwards 1992];
- *the granularity of the entity*: also called "different levels of traceability" [Lindvall 1994], it determines the detail level of the entities involved, e.g. classes or attributes/methods of an object-oriented analysis, paragraphs or sentences of a textual requirement document;
- *the attributes of the entity that should be added*: they are traceability information because they allow, for example, tracing a requirement back to its source; examples of attributes are: effort [Carlsharmre & Regnell, 2000], priority (determined by the customer) [Tvete & Sundnes, 1999], source [Kirkman, 1998], status proposed/approved/designed/incorporated/ validated [Tvete & Sundnes, 1999], status captured/specified/planned/realised [Carlsharmre & Regnell, 2000, status new/assigned/classified/selected/applied/rejected [Carlsharmre & Regnell, 2000], status optional/mandatory/deleted/desirable [Kirkman, 1998].

(c.ii) Relationship

The concept of "relationship" investigates different tracing approaches concerning the relationships that are suggested to be captured/maintained. The concept considers four aspects:

- *the kinds of relationships described*: from the literature, three general kinds of relationships can be identified: relationships between documentation entities on the same abstraction, relationships between documentation entities at different abstractions and relationships between documentation entities of different versions of a software product [Wieringa, 1997];
- *the relationship direction*: backward and forward, in pre- and post-RST, as shown in the first paragraph of this section;

- *the relationship attributes*: e.g., status, completion date, authorisation, responsible, priority, etc.
- *the setting of relationships*: this concept distinguishes between two kinds of approaches for setting relationships:
    - *implicit relationships* - links that do not require manual setting, e.g. name tracing, where if names and abbreviations are used in the same way and are meant to denote the same things in two documents, then a degree of traceability between them may be established;
    - *explicit relationships* - they are manually implemented references between documentation entities and came from external considerations supplied by the developers; so, for example, the linkage, or relationship, between a textual requirement and a use case that describes the requirement is determined solely by the decision of the developers that such a relationship has meaning.

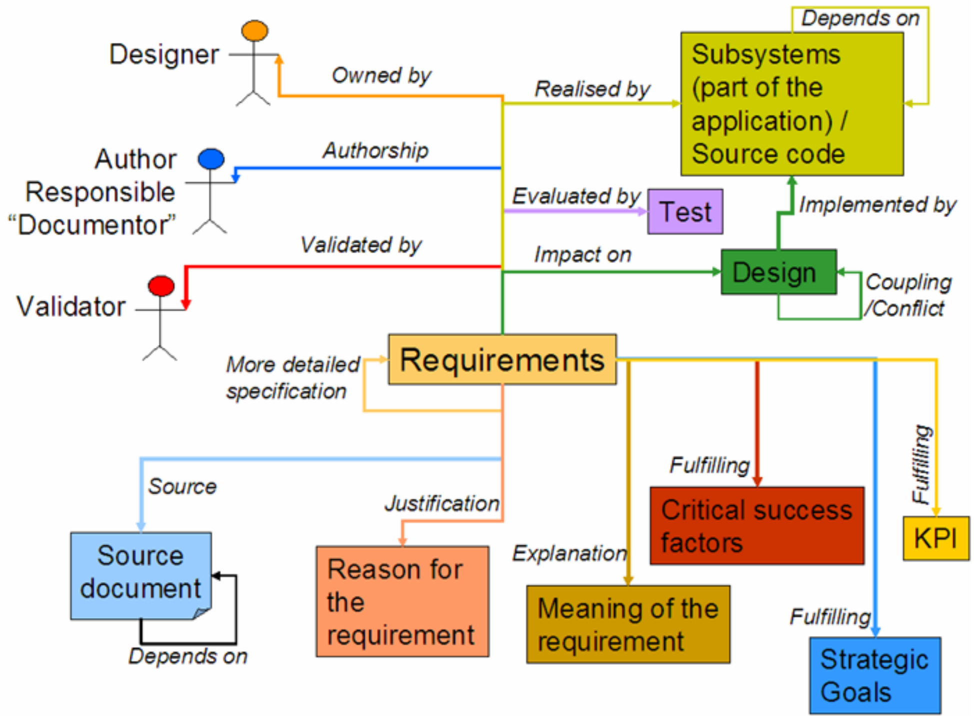


Figure 4. Some kinds of link that may be represented by a traceability approach, adapted from [Wieringa, 1997]

(d) Tools

Traceability tools answer to the following problem: in what way providing access to and presenting the traced information? These tools may be conceptual tools, software tools or a combination of the two.

Conceptual tools are general techniques to represents entities and relationships of a conceptual trace model; according to von Knethen & Paech [2002] and to Wieringa [1995] the techniques used to keep trace of requirements include the following:

- *Traceability matrices* - A simple way to represent links between items is a matrix in which the horizontal and vertical dimensions list the items that can be linked, and the entries in the matrix represent links between these items; the items in both dimensions may or may not be the same.

Although some links may have a higher "arity", with matrices only binary links between items can be represented. An advantage of the matrix representation is that it is easy to understand and it provides a format that can be discussed by stakeholders with different backgrounds.

- *Cross-references* - A requirement specification is a document with many cross-references among parts of the document, as well as with references across different documents. Links between documentation entities are embedded as pointers (e.g. hyperlinks) in a text, which may be an informal natural language text or a formal specification. Even links between diagrams can be viewed as cross-references. This involves embedding phrases like "see section x" throughout the project documentation (e.g., tagging, numbering, or indexing of requirements, and specialised tables or matrices that track the cross-references). Cross-references allow the related documents to be navigated through. The use of cross-references is simple to understand, and software that maintains cross-references and can produce reports about them, can be implemented easily. Cross-referencing is useful for written specifications but not for a concise representation of links such as can be done with matrices. Cross-references are always binary links, so that links of higher "arity" cannot be easily represented.
- *ER models* - Links between items can also be represented by Entity-Relationship models. The linked items are entities, the links are relationship instances. The ER representation has the advantage that links with "arity" higher than two can be represented. Moreover, an ER model of links can be implemented using any database technology. This view of RT has been taken by many repository designers. The use of database technology has the advantage that ad hoc query and reporting facilities are easily available.
- *Graphical models* - In these models documentation entities are represented by entities and relationships between them by relationships; they are based on a formal graphical notation (e.g. UML).
- *Tracing languages* - In these models entity and traces are represented by the mean of a formal language; the types of languages used for traceability include DB query languages (as SQL) and regular expressions.

Software tools may help, support and guide the analyst in the tracing activities; some approaches imply the use of a particular software tool to be applied. According to Gotel & Finkelstein [1994], the following families of software tools can be defined:

- *General-purpose tools* - These tools include hypertext editors, word processors, spreadsheets, database management systems and prototyping tools. They can be hand-configured to allow previously manual and paper-based requirements traceability tasks to be carried out on-line. This generally involves establishing cross references and placing conditions upon their automatic update.
- *Special-purpose tools* - A number of tools support single and well-defined activities related to requirements engineering. Of these, some achieve restricted types of requirements traceability. For example: the KJ-editor assists the organisation of idea formulation, providing traceability between ideas and requirements [Takeda et al. 1993]; PORC assists interview transcript analysis, providing traceability between interview transcripts and derived requirements [Langford 1991]; and the T tool assists test case generation, providing traceability between requirements and test cases [Sodhi 1991]. Although there may be a limited degree of explicit control and guidance, support is generally

implicit in the use of the tool, which automates any mundane and repetitive tasks needed to provide this requirements traceability.

- *Workbenches* - When a collection of the above types of tool are organised to support a coherent set of activities, less restricted types of requirements traceability can be supported. The degree of support depends on the focal activity of the composite tool. Typically centred around a database management system of some form, these software types comprise dedicated tools for documenting, parsing, editing, interlinking, organising, and managing requirements. They often provide facilities to help assess and carry out any changes made to these requirements.
- *Environments and beyond* - Requirements traceability can potentially be provided throughout a project's life if tools supporting all aspects of development are integrated. The basis used for internal integration tends to define how requirements traceability is established: through the use of a common language (e.g., the Input/Output Requirements Language in Technology for the Automated Generation of Systems [Sodhi 1991]); through the use of common structures (e.g., the relations of an Entity-Relation-Attribute Model in Genesis [Ramamoorthy et al. 1988]); through the use of a common method (e.g., the Information Engineering Method in the Information Engineering Facility [Texas Instruments 1988]); or through the use of specialised requirements traceability tools or sophisticated repository structures where a number of interlocking tools are combined to support many languages, methods or structures (e.g., Teamwork/RqT [CADRE 1992]). Those with the flexibility to incorporate third-party environments tend to provide requirements traceability support through the use of powerful repositories and underlying database management systems. These are used to relate the products of the individual components (e.g., the Digital CASE Environment [Sodhi 1991]).

1.3. Scope and focus

1.3.1. Focusing the problem

In the last years, some RT models proposed software-based approaches directed towards the automation of tracing processes[13]. These models are based on the assumption that a system design activity moves in a explicit and structured way, considering requirements one by one, towards possible solutions and refining requirements in a continuum to design elements. Current industrial practices show however that requirements do not fade naturally into design choices and that system solutions do not derive directly from requirements refinement: design is an intuition and induction process more than a derivation one[14]. For instance, some requirements remain implicit at the beginning of the project but they are considered in design and often, at the end of the project, designers do not remember the actual reason for these choices. It can be therefore discussed if the design process is understandable or not as a fully rational and explicit sequence of actions. An hypothesis that raise out from the professional experience of several designers may be the following.

---

[13] See also [Dick, 2002 ; von Knethen, 2002 ; Alexander, 2003 ; Maletic et al., 2003 ; Sherba et al., 2003 ; Spanoudakis, 2003]
[14] See also [Arciszewsky et al., 1995]

Requirements and design stand into separate and different conceptual area: the firsts in the space of problems, the latter in the space of solutions. During a design process there are not explicit relationships between these two "spaces": designers keep requirements in mind as a background knowledge; they "absorb" requirements as a whole and they build up the application architecture inductively and in part intuitively, following their own skills and competences.

The PhD research presented in this dissertation does not want anyway to validate the hypothesis below; the fact is that the process that brings requirements knowledge through the designer mind towards elements of the system is still something not very clear, and its investigation involves complex psychological issues that cannot be treated in these pages. This research does not aim to a general theory of traceability activities and, on the other hand, there is not a theory on requirements elicitation nor a theory on design. This method does not focus therefore on modelling the process of evolving requirements into design, but it pretends to provide to designers an effective tool conceived to discuss and analyse the design choices after they have been taken, in order to refine these choices according to the main requirements and in order to eliminate unmotivated elements. In other words, the traceability approach here proposed is intended to face pragmatically the problem that, in common industrial cases, traceability relationships are sill not explicitly specified; this make very hard to verify, to evaluate, to revision and to reuse efficiently design solutions in relation with high-level requirements. The problem can be summarised in three elements:

i. Project specifications do not document the decision-making process, just the results of the process [Potts & Bruns, 1988]
ii. Project specifications do not document the reasons of choices, just the solutions [MacLean et al., 1989]
iii. Project specifications do not document possible or proposed solutions, just the accepted ones [MacLean et al., 1989].

This fact includes that in some cases it is not clear why a certain piece of the system has been developed, and how the product answer to needs stressed by strategic requirements; in other words it is not clear how to evaluate, to validate or to motivate the quality[15] of the final product. Another problem raised out from this situation is that revision activities are very hard to bring on, while it is not clear the impact of requested changes and their effect on the application requirements compliance. The analysis method proposed in these pages tries to answer to the problems below considering three aspects of a traceability approach:

i. *Following the process*. RT is the ability to follow a specific item at input of a phase of the software lifecycle to a specific item at the output of that phase [Hamilton & Beeby 1991]. The proposed method do not take into account the mental process that brings from general requirements to concrete design solutions; on the contrary, the tracing activity tries to move intuition and induction to more rational cause-effect motivations.
ii. *Arguing the reasons*. RT gives essential assistance in understanding the relationships that exist between the project artefacts and their motivations. In fact, in such a decision making activity not

[15] See also the concept of quality detailed in the previous paragraph

only goals and requirements are involved, but also a wider knowledge about the specific application domain, some project constraints, "good design" and usability principles, designer skills, etc.

iii. *Documenting rejected solutions*. RT is the ability to discover the history of every feature of the outputs of an acquisition activity so that the impacts of changes in acquisition process can be identified [Hamilton & Beeby 1991].

Since requirements are understood as "base information" about how the application should be and why, skilled designers are able to draw a design specification that satisfy at least in a certain measure those requirements. The kind of analysis here proposed must not repress or stiffen the design process, but it should help in better understand the reasons for design choices and in force to better make explicit requirements that are both implicit or unexpressed.

1.3.2. Focusing the domain

The approach proposed in this dissertation is a design traceability method supporting post-RST in both a forward and backward direction. The focus on post-RST is justified by the fact that much research is done in pre-RST to capture rationale [Watkins & Neal, 1994; Gotel & Finkelstein, 1994], but there is a lack of traceability through the entire system development process [Strasunskas, 2003]; the focus on design is justified by the fact that many critical risks in software engineering are architectural [Boehm, 1991]. The approach is called TRAMA: TRaceability Analysis Method for (interactive) Applications. Two domain-related definitions need to be discussed here: the definition of *Interactive Application* (i.e. why is such an application different than any other software application) and the definition of *Design* of an interactive application (i.e. why TRAMA will consider design in terms of application features and contextual information).

Interactive Applications are all those software-based systems which include an active interaction between a human user and a machine as a central element of their working.

According to the meaning considered in the method, Interactive Applications are for instance web sites, CD-ROM, iTV applications, information systems applications, knowledge managements applications, e-learning and educational applications, etc. The TRAMA approach focus on this kind of software application and not widely to all kind of computer applications in consideration of the distinctive features of Interactive Applications (IA):

- IA requirements are high-level descriptions of the application features: in these cases requirements do not focus on how software components have to be coded but on what they should do or provide and why;
- IA design is a conceptual rather than a logical/technical specification: this kind of designs shape the application structure in terms of contents, access paths to contents, navigation possibilities, interface structure, etc. without considering the technical implementation (usually a further specification is produced to detail and document objects, classes, pieces of code, etc.);
- IA base on a informational, hypermedia structure rather than on a sequence of operations: in most of the cases, this kind of applications focuses on information and navigation capabilities; if

operations are provided (e.g. in e-commerce applications) they usually are treated in terms of user experience as a separate piece of the system, with a stand-alone, step-by-step and next-only navigation style.

These characteristics show that the huge family of IA needs to be discussed as something different than all the other software applications [Lowe, 2003], and a specific traceability method needs to be developed. In fact, both in the field of requirements analysis[16] and of design specifications[17] specific methodologies have been proposed in the last years.

As mentioned before, a traceability method for IA have to include a wide definition of design: in cases like information systems or, more again, like educational applications, it is impossible to understand requirements impact considering only the software application; contextual information need to be taken into account as well [Armani et al., 2004]. For instance, in the e-learning community the concept of Instructional Design is commonly shared: "The systematic and reflective process of translating principles of learning and instruction into plans for instructional materials, activities, information resources and evaluation." [Reigeluth, 1999]; this definition includes students' activities, teachers' activities, tools, workflows, etc. and specific design languages[18] have been developed to shape what they call an "educational environment". The same elements may be taken into account in other IA domains. According to this point of view, this dissertation will adopted the following definition of *design*:

> The design of an IA may include application-related elements as well as contextual-related elements. Application-related elements define contents, structure of content, access paths, navigation, presentation and user operations[19]. Contextual-related elements define actions, activity flows, resources, tools and locations[20].

1.3.3. Research hypothesis and goals

Some experiences I carried on in the last years with designers and project managers showed to me that (a) one-to-one relationships between a requirements and an element in the application are very uncommon and that (b) there are other motivations for design than requirements. These two remarks appear as obvious for all the professional designers I ever met, but they never have been scientifically validated. Therefore, I have put them as hypothesis for this dissertation, trying to find evidences of their exactness.

> *Hypothesis 1*
> Design structure and requirements structure not always homogeneous: in some cases, a requirement does not impact on a single design element but on a number of design elements and on their interactions.

[16] For an example of requirements analysis methodology for interactive applications, see also Bolchini et al. [2003]
[17] For a list of design methodologies for interactive applications, see also Bolchini et al. [2005a].
[18] See also Botturi & Belfer [2003].
[19] Garzotto et al. [1993]
[20] Botturi [2004]

Hypothesis 2
In some cases the motivations for design choices can be found not only in requirements but also in other elements such as a wider knowledge about the specific application domain, some project constraints, "good design" and usability principles, designer skills, etc.

The objective here is not to extensively prove these two hypothesis for all the cases[21], but that they can be applied at least in the experiences and in the case studies that will be presented in this dissertation, i.e. that they are valid at least in some cases.

Goal 1
The research aims at the validation and verification of hypothesis 1 and 2 in all the case studies taken into account.

If this will be true, the TRAMA method will consider the two hypothesis between the possible cases that may occur in a real project. On the other hand, this first goal is a starting point that will facilitate the investigation of (a) the possible relationships existing between requirements and design, i.e. the impact of requirements on design and (b) the possible relationships existing between design and requirements (or other elements), i.e. the motivations of design choices. In other words, *Goal 1* is a "research guide" that will pave the ground to face with the main research goal, i.e. the specification of the TRAMA method[22].

In order to understand the motivations for the main research goal (*Goal 2*), some problems detected in industrial and academic projects[23] need to be considered:

- In many cases reasons of design decisions are difficult to understand or to remember. Often a tuning or a reengineering activity is needed after a certain application life-time; if design motivations are not documented, impact of changes cannot be properly evaluated or some design elements may be considered as mandatory without a specific reason.
- "Negative" design is not always recorded. Which solutions have been formerly considered? Why some solutions have been discarded or not accepted? In this case too, impact analysis becomes hard to be performed; moreover, unaccepted solutions include often important elements of the project knowledge.
- There is not a good practice in traceability for IA design. As mentioned in the previous paragraph, IA have particular requirements and design characteristics that differ this kind of applications from the other software applications. The current state of the art does not include any specific traceability approach for IA: available methodologies are tailored for general software applications.
- As a consequence of the previous problem, there is not a "usable" methodology for post-RST in IA. The intrinsic usability of the approach should be assured according to some principles [Triacca, 2004]: (i) the tracing process have to be engineered and standardized, (ii) the method have to be systematic, (iii) the reusability of the method have to be enhanced in different fields (making it cost-

[21] The hypothesis have in fact a "negative" formulation, stating that their opposite is not always valid.
[22] The research method will be detailed in the next paragraph (1.3.4).
[23] See also section 2.

effective) and (iv) the notation of the method have to be as simple as possible, easily learnable, flexible, modular and scalable.

Taking into account these problems, the main goal of the TRAMA research can be expressed as follows:

> *Goal 2*
> The research wants to provide a methodological support to analytically organise, perform and document a post-RST activity in both a forwards and backwards direction.

In particular, the approach is intended to provide: (a) a method to analyse and record the requirements impact, (b) a method to analyse and record the design motivations, (c) a suggested workflow for these activities, (d) a set of heuristics for these activities, (e) an annotation method supporting these heuristics, (f) a notation documenting the traceability relationships and (g) the main requirements for a specification tool supporting the method. Moreover, the method proposed in this dissertation aims at the investigation of some specific aspects of a traceability activity.

> *Goal 3*
> The research wants to provide a support for the different aspects concerning a design traceability approach: client validation, design versioning, "negative" design, non-traceable design, reverse requirements specification and usability on design documents.

- *Client validation* – This activity is supported by a proper traceability approach (i) in a forward direction showing which requirements have been taken into account in the design and how, following evolving requirements in design, checking consistency and feasibility of requirements and estimating the impact of a change in requirements on the design and (ii) in a backward direction finding arguments to justify design decisions, checking whether all requirements are considered by the design and estimating the effect of a required design change.
- *Design versioning* – In some common cases, designers or project managers need to highlight different design aspects for different stakeholders. A proper backward traceability approach allows to understand which parts of the design are relevant for which stakeholder. The design-requirements-goals-stakeholders chain helps creating different versions of the design documentation, addressed to specific targets.
- *"Negative" design specification* – With "negative" design I mean those design objects that have been eliminated or modified during the project life-cycle. Proposed elements in the application may become part of the negative design (i) because of a direct rejection, (ii) because of a change in related objects or (iii) because of business, technology or law constraints. Keeping trace of old design versions and understand and remember former design decision is useful to remember why a decision and not another has been taken, validate negative decisions with stakeholders, understand why a design decision has been rejected or show the "negative" impact of a specific constraint or requirement on design.
- *Design sources elicitation* – According to the hypothesis 2, in design part of decisions are taken because of specific requirements, while other design elements may come from designer expertise, technology constraints, "graphic" constraints, budget constraints, laws obligations, etc. In any case,

documenting the sources of the design elements helps the developer interpreting the specification and allows identifying the parts of the applications that can be modified with no impact in the overall requirements covering.

- *Reverse requirements specification* – Sometimes requirements specifications are written after design or after implementation phase, just for documentation. In these cases, a proper traceability approach may help in producing an effective requirements specification according to the real stakeholders' goals and requirements. "Ex-post" traces are anyway useful to check the consistency between design and requirements, to tune up existing requirements specification according to the actual application and to extract consistent requirements specification from design. Such a reverse requirements specification is a beautiful tool to keep trace of strategic decisions, to provide design decisions with argumentations and to collect information and material for a consistent usability test.
- *Usability evaluation on design documents* – According to Triacca [2004], the usability evaluation should be done as soon as possible in an application development life-cycle: it is better to anticipate the main errors and problems before implementation, because the error correction is more expensive in advanced development phases. Since scenarios for usability evaluations are goal-based, keeping trace of the relationships between requirements and design artefacts helps selecting the elements in the design involved for a specific task, evaluating the quality of the product with respect to the high-level goals and identifying test procedures that should be rerun to validate an implemented design change.

In order to summarise the goals of the research described in this dissertation, the main questions TRAMA will try to answer can be listed as follows:

- What are the criteria to take properly in account traceability information in the design phase of an IA life-cycle?
- What is the traceability approach to adopt? What is the related conceptual model to take into account? What are the objects, actions, considerations, etc. to trace?
- Who are the 'users' for traceability? Who will make use of the knowledge surfaced by traceability and why?
- How relationships between requirements and design and between design and its sources can be represented and documented?
- Is traceability a self-standing activity (or discipline)?

This last question is worth some words more. In the case studies carried out until today, the traceability activity performed has been perceived by designers and by project managers as a separate and independent activity, other than design or requirements analysis. A parallel can be done with RE, that has not been for a long time considered a different activity (and discipline) than software design/engineer. In the same way, today traceability is considered a part of the requirements engineer process. This dissertation will try to focus on a different hypothesis:

> *Hypothesis 3*
> If one considers requirements as the strategy to satisfy stakeholders' goals and design as how the application must behave, a traceability activity forward to and backward from design elements can be defined as the argumentation activity about why design solutions satisfy requirements. Due to

organisational problems (e.g. resources allocation) and psychological issues (e.g. problems in self-observation), a traceability expert is a different role than a designer or a requirement analyst; a traceability expert is a facilitator in project meetings, with specific competences in eliciting and understanding why a certain decision is being or has been taken.

The research described in this dissertation does not aim at the automation of the traceability process nor at stating a general theory on traceability activity; in the same way, there are not general theories on requirements elicitation or on design processes. The TRAMA method will be conceived to be usable for human users and it will not necessarily rely on a software tool to be applied. Anyway, the discovery of a general theory on tracing activities could become a "collateral" result of the research process.

If one considers the originality characteristics of the research, the following elements have to be mainly mentioned:

- TRAMA focus on post-RST and it is specifically tailored for IA;
- TRAMA investigates forward traceability relationships from requirements to design elements, as a powerful elicitation tool to analyse the impact of requirements on the application and its requirements covering degree;
- TRAMA analyse backward traceability relationships from design elements to their sources, highlighting the parts that are deeply related to requirements or to constraints and that cannot be modified;
- TRAMA supports directly important project activities such as the changes impact analysis, the negative design documentation and the "reverse" requirements specification.

1.3.4.Research method

TRAMA can be considered as an empirical research. According to the Wikipedia Encyclopaedia, an empirical research is "any activity that uses direct or indirect observation as its test of reality. If a-theoretical, it is a form of inductive reasoning (...) and it may also be conducted according to hypothetico-deductive procedures"[24]. In this case, experimentations and case studies will modify, refine and test the method to provide at the end a general approach. These results, in an iterative process, will be used to perform new experiences and to improve the model. An integrated view of the entire process is summarised in Figure 5.

This iterative process will bring TRAMA from a first coarse-grain version towards its final state, an "Alfa 1.0" version of the method ready to be extensively tested on industrial cases. The coarse-grain version of TRAMA is a first attempt raised out from direct professional experiences in the field of IA design. As it is shown in section 3, this first version of the method will be applied on a real non-academic situation and its basic features will be tested. The experience will be carried on in strict collaboration with the project manager and with some designers and developers that have collaborated to the project itself; after the project on which TRAMA is applied will be finished, a meeting with the work-team will be organised to

[24] http://www.wikipedia.org/

discuss the results reached by the use of TRAMA, to surface the problems in its use and to highlight its benefits.

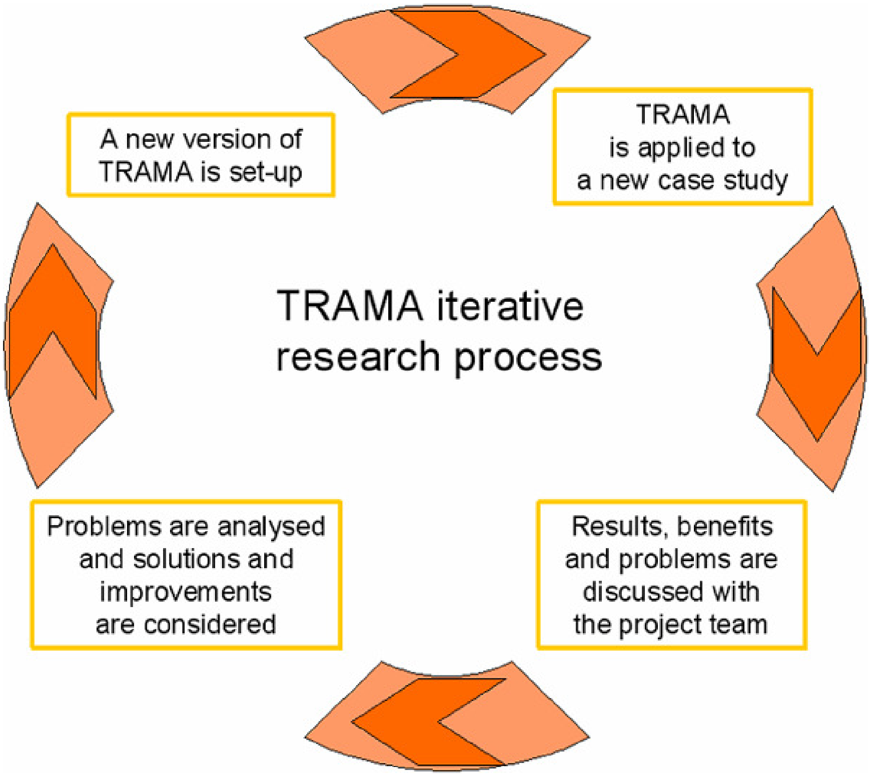


Figure 5. A schema of how the TRAMA research will be carried on

Then, a summary of positive and negative elements of the current version of the method will be summarised and analysed; solutions to the problems encountered and other improvements are considered to produce a new TRAMA version. When a new refined version of the method is set-up, it is ready to be tested on a new case study. The activities here described will be performed for each project on which TRAMA were applied; the testing cycle will be repeated for 6 subsequent iterations. Experimentations will be performed both on academic projects and on industrial cases, after the design phase and during the design phase, considering in a separate way the different aspects of the problem. Academic projects were chosen between the European and national projects of TEC-Lab[25], the Technology Enhanced Communication Laboratory of the University of Lugano (Switzerland), and of HOC-Lab[26], the Hypermedia Open Centre of the Politecnico di Milano (Italy). Single features of the method were also tested on students works at University of Lugano and at Politecnico di Milano. Industrial projects will be selected between the different collaborations that both TEC-Lab and HOC have with industrial partners and companies active in IA development.

---

[25] http://www.tec-lab.ch
[26] http://hoc.elet.polimi.it

1.3.5.Expected results

The research described in this dissertation is expected to validate the first two hypothesis: case studies should shown, at least in some cases, that a single requirement does not impact on a single design element but on a number of design elements and that there are other motivations for design choices than requirements. In these cases, the research should show that the automatic derivation of design from requirements and the reverse derivation of requirements from design elements do not find a meaning and cannot be applied in the actual industrial practices. The hypothesis 3, i.e. the fact that traceability is a self-standing activity in projects, will be validated at least for the case studies presented.
The research is also expected to reach its main goal, providing a usable method for post-RST of requirements impact and of design motivations. TRAMA is intended to allow the structured analysis of how and why each requirement in a project has been taken into account in the application design, of which design element is related to which requirement or goal and of dependences, interactions and relative relevance involving these components. In the same way, TRAMA should allow to surface the reasons why a certain decision has been taken during the project, meaningfully linking design elements with its sources, i.e. with requirements or with other possible motivations.
A further important result that this research is expected to reach, is to provide with the traceability method a set of heuristics allowing to analytically organise the traceability activities. These structured heuristics should facilitate the analysis of traceability relationships also for non-expert analysts, providing a standard workflow to apply the method, a suggested set-up for the meetings with stakeholders and a checklist of aspects and of elements to take into account during these activities. The research is also intended to provide a set of structured heuristics allowing to keep trace of negative design elements, understanding the motivations of those choices that have been changed, rejected or discarded during the project.
TRAMA wants also to attach to the method simple conceptual tools helping to perform and document the analysis results; these tools could be a notation or a knowledge representation method specifically tailored to produce communicative documents and materials to carry on the traceability activities and to efficiently present the information raised out from the analysis.
Two results that the research is expected to provide as side effects are a set of requirements in order to develop a tool supporting TRAMA traceability activities and an educational plan for an advanced course on traceability and on the TRAMA method.

2.Review of related works

<<Computers are getting smarter all the time: scientists tell us that soon they will be able to talk to us. (By "they" I mean "computers": I doubt scientists will ever be able to talk to us.)>>
Dave Barry

Abstract
In the last 10 years traceability for interactive applications has been studied as a part of the requirements analysis process. In this section the main works on the field will be highlighted and in particular the "contribution structures" approach by Gotel & Finkelstein [1995], the KAOS framework by van Lamsweerde et al. [1998], the i* model by Yu [1993], the AWARE model by Bolchini et al. [2003], the PRO-ART environment by Pohl et al. [1994], the CBPS approach by Egyed et al. [2000], the Potts and Bruns model [1988] and the "rich traceability" approach by dick [2002].
Open problems in current traceability practices can be summarised in problems of: adoption, context, communication, specification, tool-dependency and guidelines.

2.1. State-of-the-art

In the last 15 years, Requirements Traceability (RT) has been identified in the literature as a quality factor, i.e. a characteristic a system should possess and include as a non-functional requirement. Tracing requirements is viewed as an essential activity in developing large systems, identifying errors, surfacing inconsistencies and managing changes [Pinheiro & Goguen, 1996]. This topic has been studied as a part of the requirements analysis process [Jarke, 1998]; in general, researches in the field have been conducted between individual requirements and goals [Bolchini, 2003] or between two different phases of software development, e.g. between requirements and rationale behind them [Dick, 2003], or between requirements and architecture [Pohl et al., 1994], or source code, etc. A big part of RT approaches proposed are pre-RST models, focusing on the formalization of tracing processes and on implementing tools supporting the developed methods; most of these approaches at requirements level deal with semi-formal specification, e.g. scenarios or use cases, and they hardly consider plain language documents [Strasunskas, 2003].

In this section I will briefly summarize some major contributions in the Requirements Traceability field, taking sides about some ideas proposed and highlighting the aspects and the concepts that have been relevant for the TRAMA research work, and in particular:

- common concept related to requirements management from goal-oriented methodologies; KAOS [van Lamsweerde et al., 1998] and i* [Yu, 1993] for the concepts of stakeholder, goal, requirement and goal refinement; AWARE [Bolchini et al., 2003] for the concepts of visions and user motivations;
- the idea of Design Versioning, inspired by the Contributions Structures approach [Gotel & Finkelstein, 1995];
- the requirements to design tracing activity, formerly studied for the PRO-ART [Pohl et al., 1994] and CBPS [Egyed et al., 2000] methodologies;
- the idea of record the reasons for design decisions as in the Potts and Bruns model [Potts & Bruns, 1988];
- the idea of explicitly representing different design choices and the reasons for choosing one of them as in what I call the "Xerox approach" [McLean et al., 1989];
- the concept of design rationale as in [Arkley et al., 2002];
- the concept of rich traceability as in [Dick, 2002];
- the use of Requirements Traceability Matrices as conceptual tool to represent relationships from requirements to design elements, e.g. in [Hayhurst, 1997], [Finholt, 2003] and [WIC Program, 2004].

Then, a review of the main software tools supporting the tracing activity and the problems that the current research and tools still let open will be discussed.

2.2. At the requirements side: stakeholders, goals, etc.

2.2.1. Contribution structures

Orlena Gotel and Anthony Finkelstein proposed an approach called "Contribution Structures" [Gotel & Finkelstein, 1995] that provides a way to define links between authors/contributors and application artefacts (e.g. "contributed_to" and "contributed_by"). According to socio-linguistic theories, the contributors can have different roles: *principal*, the agent(s) who motivates the production of the artefact and whose position and/or belief is established by the information therein (i.e., committed to what it expresses and responsible for its effect or consequences); *author*, the agents(s) who chooses, formulates, and organizes the content and structure of the information in the artefact (i.e., responsible for its syntax and semantics); *documentor*, the agent(s) who captures, records, or transcribes the information in the artefact (i.e., responsible for its physical manifestation). The invisibility of the individuals and groups that gave rise to requirements artefacts has been identified as a primary reason for the persistence of requirements traceability problems. The authors introduce the concept of "social infrastructure", which refers to the overall system of agents in the process, along with the various relationships they are involved in. Social relations reveal information about the social network and answer the 5 questions of: (i) involvement, (ii) responsibility, (iii) working arrangement, (iv) change notification and (v) ramification. This approach enables then consistent change integration identifying appropriate agents.

TRAMA has been inspired from the contribution structures approach in tracing relationships between stakeholders and application artefacts; in fact TRAMA considers that different stakeholders may need different documentation versions in relation to their specific goals[27]; the traceability chain helps in identifying relevance relations between stakeholders, through goals and requirements, to design elements.

2.2.2. KAOS

Alex Van Lamsweerde suggests a goal-oriented requirements engineering framework called KAOS [van Lamsweerde et al., 1998] and enabling a pre-RST between the elements of the model. In this approach goal hierarchies express system goals and the requirements that support the achievement of system goals. The impact of changes to goals or requirements can be examined by traversing up and down the goal hierarchy. Traceability can be a way to keep all the changes in the track of the original goals; the author stresses the need to keep trace between the parts of all docs and specifications. The KAOS methodology provides a specification language for capturing why, who and when aspects in addition to the usual that requirements, a goal-driven elaboration method, and meta-level knowledge used for local guidance during method enactment. The language provides a rich ontology for capturing requirements in terms of goals, constraints, objects, actions, agents, etc. Links between requirements are represented as well to capture refinements, conflicts, operationalisations, responsibility, assignments, etc. The KAOS analysis method roughly consists of (i) identifying and refining goals progressively until constraints that

[27] See also the concept of "Design Versioning" in section 4

are assignable to individual agents are obtained, (ii) identifying objects and actions progressively from goals, (iii) deriving requirements on the objects and actions to meet the constraints and (iv) assigning the constraints, objects and actions to the agents composing the system.

TRAMA do not deal with pre-RST, therefore one can represent requirements-related information in different ways, form natural language to a formal method. Anyway, traceability between design elements and their motivations, as well as requirements impact analysis are facilitated by the use of a goal-oriented requirements management method. KAOS can be profitably used in particular for its fine-grain way to deal with goals and with refinement into sub-goals.

2.2.3. Distributed Intentionality (i*)

Eric Yu proposes an organizational modelling framework called i*, which stands for "distributed intentionality" [Yu, 1993]; this approach captures the intentional structure of a software process and its embedding organization, in terms of dependency relationships among stakeholders. Stakeholders are represented as (social) actors who depend on each other for goals to be achieved, tasks to be performed, and resources to be furnished. I* uses the notions of actor, goal and (actor) dependency, as a foundation to analyse high-level goals together with non-functional requirements and to model architectural and detailed design. The i* framework includes the strategic dependency model for describing the network of relationships among actors, as well as the strategic rationale model for describing and supporting the reasoning that each actor goes through concerning its relationships with other actors. A strategic dependency model is a graph involving actors who have strategic dependencies among each other. A dependency describes an "agreement" between two actors; the type of the dependency describes the nature of the agreement. A strategic rationale graph captures the relationship between the goals of each actor and the dependencies through which the actor expects these dependencies to be fulfilled. These models have been formalized using intentional concepts from Artificial Intelligence, such as goal, belief, ability, and commitment.

As discussed before for KAOS, i* can be used in TRAMA to represent requirements-related information; the stakeholder-goals-requirements hierarchy is a useful tool to complete the traceability chain in the pre-RS area. The concept of strategic rationale has influenced the TRAMA concept of relationships rationale from a stakeholders' goals perspective[28].

2.2.4. AWARE

Davide Bolchini proposes a model called AWARE [Bolchini et al., 2003] that aims at capturing high-level communication goals, considering several user profiles, defining hypermedia-specific requirements, and reusing requirements for an effective usability evaluation. Technique is usable, informal, requires little training effort, and shows relative advantage to project managers. Starting from the i* framework,

[28] See also section 4

AWARE provide a method and a notation defining hypermedia requirements (concerning aspects such as content, navigation and presentation) for web applications. The model adopts a goal-driven approach coupled with scenario-based techniques, introduces a hypermedia requirement taxonomy to facilitate web conceptual design, and paves the way for systematic usability evaluation. Latest developments of the approach [Bolchini et al., 2005] introduces the concepts of vision, i.e. an assumption of a stakeholder which dictate his/her "weltanschauung" on the project, and of user motivations, i.e. the emotional, psychological, social or individual elements which can trigger a person to use an interactive application.

TRAMA considers this approach as the best way to "normalise"[29] requirements-related information because of its explicit taxonomy facilitating the matching between requirements and deign elements. In particular, TRAMA has been inspired by AWARE for the concept that a requirement may have impact on groups of design elements and not only on a single element. Furthermore, TRAMA uses the concepts of stakeholders, goals, requirements, visions and user motivations in the same way they are used by AWARE.

2.3. Requirements to Design: building the bridge

2.3.1. PRO-ART

Klaus Pohl introduces a tool-based requirements engineering environment, called PRO-ART (Process and RepOsitory based Approach for Requirements Traceability) [Pohl et al., 1994]; it is presented as a "focused traceability" approach that supports change integration and integrates requirements with architecture information. In other words, the model tries to identify relationships between requirements and application architecture on the base of scenarios. The model is conceived to define first generic traces and to specialise the most relevant ones in a second time. The use of scenarios should facilitate the representation of user requirements, reduce the complexity, support communication with customer and interrelate requirements with architecture. Another element of this approach stands in the use of meta-models describing artefacts, structuring requirements information and interrelating structured information. The PRO-ART tool is based on three main contributions: (i) a three-dimensional framework for requirements engineering which defines the kind of information to be recorded; (ii) a trace-repository for structuring the trace information and enabling selective trace retrieval;(iii) a novel tool interoperability approach which enables (almost) automated trace capture.

PRO-ART can be considered a first significant attempt to bridge the gap between requirements and realisation. TRAMA deal with the same problem but the approach is completely different, considering the weak points of PRO-ART: (i) the use of scenarios is too formal and risks to record only partial information and (ii) the method prescribe the use of a specific tool to be applied[30].

[29] For the concept of Requirements Normalisation see also section 4
[30] More details about open problems are listed in paragraph 2.7

2.3.2. CBPS

Alexander Egyed, Paul Grünbacher and Nenad Medvidovic proposes the CBSP (Component, Bus, System, Property) approach [Egyed et al., 2000; Grünbacher et al., 2001], which deals with refinement of requirements to initial architecture, as requirements may explicitly or implicitly contain information relevant to the system's architecture. The problem the authors underline is the existing natural gap between requirements and architecture. Taking into account the PRO-ART solution, the transition problem is so still unsolved. The approach helps refining requirements to an initial architecture, supports development with evolving requirements and architecture and facilitates the elicitation of architectural information out of requirements. CBSP works through a two-level process: a requirements negotiation process and a refinement process. The requirements negotiation process (WinWin) captures and structures the information, resolving differing concerns and providing a rationale view. The refinement process identifies artefacts relevant for architecture letting stakeholders (e.g. architects) classify artefacts (C,B,P,S), specifies interdependencies among artefacts, breaks up complex artefacts classifying it into various CBSP categories and minimises CBSP removing replaced artefacts and merging related ones.

The CBPS approach faced the TRAMA research with a delicate problem: the borders between requirements and application architecture (i.e. design). Is it possible to follow and document requirements fading into design? Is design the product of requirements refinement? Where requirements stop and where design starts? The case studies presented in section 3 seem to show that the hypothesis of a natural evolution between requirements and design is not necessarily true. Anyway, TRAMA does not exclude this possibility even if it have to consider also the opposite.

2.4. Design to Requirements: justifications, motivations, rationales, etc.

2.4.1. The Potts and Bruns model

Colin Potts and Glenn Bruns outline a generic model for representing design deliberations and the relation between deliberations and the generation of method-specific artefacts [Potts & Bruns, 1988]. The model is an attempt to delineate the generic elements of software design rationale, such as artefacts, issues, positions, justifications, and the relations among them. This model provides a simple representation that can be tailored to different design specific methods and used for representing the process of design deliberation as well as the artefacts that result from such deliberations. The authors also describe a rule-based, semi-structured, hypertext system that helps the user to examine and record design rationales easily. This way, a design history is kept as a network structure linking the nodes representing the different elements of their model. Such a history provides a basis for realizing many of the benefits mentioned above, such as better understanding of the issues and arguments underlying a design, and learning from past decisions. A design history is regarded as a network consisting of artefacts and deliberation nodes. Artefacts represent specifications or design documents. Deliberation nodes represent issues, alternatives or justifications. Existing artefacts, including requirements documents, give rise to issues about the evolving design. For example, if the artefact is an informal specification of a text

formatter, the issue may arise 'how is the input text going to be read?'. An alternative is one of several positions that respond to the issue. For example, the alternative 'we need a procedure to read lines' is one possible response to the above issue. Not all alternatives directly suggest the need to create new artefacts; many reflect the need to modify or refine existing artefacts, or state that no design changes need to be made. A justification is a statement giving the reasons for and against selecting the related alternative; for example, 'we should read the input line-by-line because there are two kinds of lines (text lines or command lines), which must be treated differently'.
Jintae Lee proposes an extension to the Potts and Bruns model [Lee, 1991], consisting of enriching the internal structure of justification in the original model by making explicit the goals presupposed by arguments, the relations among arguments, and the first-class nature of these relations. A language and a system supporting this extension of the model is also proposed.

The Potts and Bruns approach, starting from the fact that a design artefact typically documents the results of a phase of designing and not the process followed, is based on process modelling methods to represent a design deliberation model. TRAMA takes from this approach the idea to record the reasons for design decisions and the concept of deliberations as issues, alternatives or justifications for design choices. TRAMA is also inspired by the Lee extension in considering explicitly the goals related to motivations. The main difference is that TRAMA does not trace the process but it records ex-post motivations and justifications.

2.4.2. The "Xerox approach"

Allan MacLean, Richard Young and Thomas Moran from Rank Xerox Ltd focus their work on the design of interfaces for software applications and start from the observation that the product of user interface design should be not only the interface itself but also a rationale for why the interface is the way it is. The authors describe therefore a representation for design [McLean et al., 1989] based around a semi-formal notation which allows explicitly to represent alternative design options and reasons for choosing among them. This representation allows to describe a design space rather than a specific artefact. The design space consists of a decision space (alternative options which might be appropriate}, and an evaluation space (explicit reasons such as consistency and criteria for choosing from among the
possible options). The set of options which are selected for the final design describe the artefact, and the alternatives and reasons for the choices provide an argument (or rational) which supports and helps understanding of the choices made. Such a representation is expected to play a role in improving the coherence of designs and in communicating reasons for choices to others, whether designers, maintainers, collaborators or end users.

TRAMA uses two ideas adapted from this approach: (i) explicitly representing the rationale for application choices, as in the Potts and Bruns model; TRAMA extends this concept recording the rationale of each traceability relationship in the project, both requirements to design and design to motivations; (ii) explicitly representing different design choices and the reasons for choosing one of them; TRAMA extends this idea recording all the design solutions proposed during the project and not only the reasons why some of them have been chosen but also the reasons why the others have been rejected[31].

2.4.3. The University of Newcastle upon Tyne approach

Paul Arkley, Paul Mason and Steve Riddle propose a framework supporting all aspects of the lifecycle, as a vehicle for recording, analysing and tracing development and assessment artefacts [Arkley et al., 2002]. This framework is focussed on the recording of design rationale, over and above the "standard" inter-relationships between product artefacts, and has been developed in an aerospace systems engineering context. This framework consists of a number of traceability structures which classify the relationships between development and assessment artefacts according to a number of views including system architecture, argumentation and verification. Since the research is in the context of dependable, often safety-critical, systems the authors have also concentrated on application-specific views such as safety argumentation. Provision of tool support has been studied in terms of database schemas and modifications to commonly-used requirements management tools. The overriding focus of the framework is the recording of justifications for design decisions; the authors identify three problems which follow from this focus: (i) increased burden: overworked engineers do not take kindly to being told to record the reason for every decision they make; (ii) accessibility: it's not sufficient to document the design justification, this justification must be accessible when one really need to know why a decision was made; (iii) freshness: out-of-date justification is potentially more dangerous than no justification at all, since changes to the system will require changes to the traceability information.

The approach is based on the observations raised out from a survey of traceability practices carried on by the University of Newcastle upon Tyne with a number of engineers. TRAMA founds some of its hypothesis on the results of this survey and uses the concept of recording justifications for design decisions that were partially introduced by the approaches formerly described.

2.4.4. Rich Traceability

Jeremy Dick proposes an extension to the idea of recording the rationale of a traceability relationship; the approach, called "rich traceability" [Dick, 2002], encourages the use of a deeper semantics in the traceability relationship. "Satisfiability", for instance, is a richer relationship, requiring the ability to explain that one user requirement may be satisfied by the conjunction of several system requirements, or by any one of a set of system requirements. The author suggests to define a possible approach to richer traceability relationships, making use of textual rationale and propositional logic in the construction of traceability arguments. The underlying logic allows other, deeper kinds of analysis to be performed. The same structures can be applied to the management of requirements for product families as well. The use of "exclusive or" in rich traceability provides a way of

[31] See also the concept of Negative Design Management in section 4

representing alternative ways of meeting sets of requirements. It can therefore be used to represent the variance in system requirements addressed by different configurations of a product range.

TRAMA uses a kind of reach traceability approach to comment traces, because notes with a rich semantic are explicitly attached to relationships between elements of the model, in order to explain reasons, motivations, impact, etc.

2.5. Matrices as conceptual tools

TRAMA make use of particular matrices as conceptual tools to represent, analyse and discuss traces between the project elements. In literature and in industrial practices, this kind of tool is widely adopted due to its easiness of use and its understandability; a Requirements Traceability Matrix (RTM) is generally implemented in software development processes as a table that correlates the high-level requirements and detailed requirements of the software product to the matching parts of high-level design, detailed design, test plan, and test cases. Some major examples of adoption of RTMs may be identified in the following cases: (i) the NASA GCS Software Project, (ii) the NEESgrid system and (iii) the WIC Functional Requirements Document for a Model of Information System

(i) The Guidance and Control Software (GCS) Project at NASA Langley Research Centre [Hayhurst, 1997] aims at the development of an application which purposes are provide guidance and engine control of a planetary landing vehicle during terminal descent to the planet's surface and communicate sensory information about the vehicle and its descent to a receiving device. Requirements for this software are based on a simulation program used to study the probability of success of the 1976 Viking Lander mission to Mars. In this case, a RTM has been used to trace Functional Requirements to Design Code and Test Cases.

(ii) NEESgrid is a project developed by NCSA and the Department of Civil Engineering at the University of Illinois at Urbana-Champaign. The NEESgrid system [Finholt, 2003] links earthquake researchers across the U.S. with leading-edge computing resources and research equipment, allowing collaborative teams (including remote participants) to plan, perform, and publish their experiments. The RTM for this project was a representation of user requirements aligned against system functionality and contained the following information: User Requirements (category and description), System Components that should fulfil the User Requirements, Budget Status (budgeted/ not budgeted), Assessment of system integrators as to whether the system component indicated addresses the user requirement and brief descriptions of deliverables that result from the work performed by the system integrator to fulfil the user requirement. This RTM has been used to ensure that all requirements were met by the system deliverables. The creation of the NEESgrid RTM involved the following steps: identification of user requirements, identification of system components, estimation of effort spent on each system component and mapping of system components to user requirements. The RTM indicates that 61.3% of the user requirements were addressed by the system integration team and that 18.7% were not addressed. The remaining 20% have been discussed further in order to determine whether the system integration effort is adequately fulfilling those user requirements. The generation of the Requirements Traceability Matrix was based on project documentation and on conversations with the System Integration team. It required approximately 140 man-hours to be completed.

(iii) The Special Supplemental Nutrition Program for Women, Infants and Children (WIC) is a Federally funded nutrition program administered by the US government and local agencies. The WIC Program provides nutritious supplemental foods, nutrition education, and referrals to health care, at no cost, to low-income pregnant, postpartum and breastfeeding women, infants, and children up to five years of age. Information Systems (IS) in the WIC Program support a number of program operations and management functions, such as certifying applicants, monitoring food vendors, tracking participation and expenditures, and managing appointments. A Functional Requirements Document (FRED) for a Model WIC Information System [WIC Program, 2004] has been provided to describe in a comprehensive way the functions that can be automated to support the WIC Program, to help State agencies in the preparation of a Request for Proposals for automated services and to serve as guidance to in-house Information Technology staff in the development of a WIC IS. In this document, a RTM provided a detailed overview of all of the possible functions and activities related to each functional description and offered a "baseline" from which State agencies could customize their design to meet their system objectives. In particular, this RTM organizes and tracks the requirements discussed in the FRED, comparing how various vendors proposed to implement the requirements, tracking whether and how all requirements were met by the system design, identifying the similarities and differences in the implementation of the requirements in different states, assisting in the development test scripts for the functional demonstration phase of the system testing and supporting the documentation that all system requirements were met in the acceptance testing phase of the project.

2.6. Requirements Traceability Tools

A complete review of the available commercial traceability tools is not between the objectives of this dissertation. Therefore, only some common characteristics and a brief description of the most used tools will be here presented. More details can be find in [Randazzo, 2002] or at http://www.volere.co.uk/tools.htm. Regardless of the technology used (database or cross-references), the current generation of commercially available traceability tools typically provides the following functionality:

- storage of links between items; the items may be requirements, design items, explanations, etc. and they may be represented as fixed format database records or free format text; links may be annotated, e.g. with degree of strength;
- storage of links between texts; the texts may be requirements, documents, design documents, etc.;
- storage of requirements in free text format with a hierarchical numbering scheme;
- reporting facilities; examples are keyword searches, the traversal of links, producing cross-reference lists, producing traceability matrices, etc.

2.6.1. Analyst Pro by Goda Software, Inc.

(http://www.analysttool.com)

Analyst Pro uses a requirements management methodology that covers the entire life-cycle including, from the initial requirements-gathering phase through the separation phase where requirements and non-requirements are set apart. Analyst Pro utilizes a Configuration Management methodology that enables the development staff to analyze the impact of change on requirements and component assets. Analyst Pro incorporates the following features:

- Importing Requirements - Analyst Pro allows users to import requirements from existing documents from various formats (doc, html and text).
- Requirements Sharing - Analyst Pro allows users to share and trace requirements across projects.
- Requirements Change Management - Analyst Pro automatically records and lists any changes to your project, when the changes were made and who made the changes.
- Requirements Assignment Users can assign requirements to team members and track its status.
- Requirements Graphs - Users can create pie and bar graphs with a number of requirements versus attributes. The attributes include priority, version, status and source.

2.6.2. CaliberRM by Borland

(http://www.borland.com/caliber/index.html)

CaliberRM is a collaborative, Web-based requirements management system that facilitates communication among project teams by providing centralized requirement data to distributed team members and allowing documented discussions about requirements as well as allowing project teams to fully define, manage and communicate changing application or system requirements. Changes made to requirement data such as traceability, document references, status, user responsibility and more are recorded in CaliberRM's central repository. CaliberRM keeps team members up to date on changes made to requirements by automatically notifying responsible individuals of the changes. CaliberRM also enables team members to quickly identify potential requirement problems by highlighting ambiguous and commonly used terms defined in a shared glossary. The latest version of CaliberRM provides LiveLink integration with CaliberRBT so that requirements in CaliberRM can be associated with corresponding cause-effect graph files in CaliberRBT. CaliberRM allows project teams to provide input on requirements via standard browsers and remote clients can access the system through an Internet connection.

2.6.3. DOORS/ERS by Telelogic

(http://www.telelogic.com/products/doorsers)

DOORS (Dynamic Object Oriented Requirements System) is an Information Management and Traceability (IMT) tool. Requirements are handled within DOORS as discrete objects. Each requirement can be tagged with an unlimited number of attributes allowing easy selection of subsets of requirements for specialist tasks. DOORS includes an on-line change proposal and review system that lets users submit proposed changes to requirements, including a justification. DOORS offers unlimited links between all objects in a project for full multi-level traceability. Impact and traceability reports as well as reports identifying missing links are all available across all levels or phases of a project life cycle. Verification matrices can be produced directly or output in any of the supported formats including RTF for MS-Word, Interleaf and FrameMaker. The DOORS Extension Language (DXL) is a high level C-like language that provides access to virtually all DOORS functions for user extensions and customization. DOORS includes the following functionality:

- Control of data model for process management allows user to manage the relationship between data fully including its direction, type and even whether a relationship is allowed.
- Improved security control through the use of passwords, and timeouts which "lock up" DOORS after a specified period of inactivity.
- New templates to make document generation easier have been added to the DOORS template library. New templates include ISO 12207, ISO 6592 and IEEE software standards.

2.6.4. IRqA (Integral Requisite Analyzer) by TCP Sistemas e Ingeniería

(http://www.irqaonline.com)

IRqA is a state-of-the-art Requirements Engineering (RE) tool specifically designed to provide an integral support to the complete Requirements Engineering process. In IRqA the complete specification cycle is supported via standard models:

- Requirements Capture
- Requirements Management
- Requirements Analysis
- System Specification building
- Specification validation (specification vs. requirements)
- Acceptance Tests management
- Requirements Organization & Classification

2.6.5. Rational RequisitePro by Rational Software

(http://www.rational.com/products/reqpro/index.jsp)

RequisitePro is a requirements management tool designed for multi-user environments. It features integration of Microsoft Word and a requirements database. Software project teams can gather, enter and manage requirements "in situ" (within your documents) or in a database. Automated traceability tracks requirements and changes through implementation and testing. Related requirements can be linked together, so that as changes occur to one requirement users can easily see its impact on other related requirements. RequisitePro includes templates to simplify production of requirements documents. Rational RequisitePro supports a choice of databases (Oracle, Microsoft SQL Server, and Microsoft Access) which allow users to organize, prioritize, and trace relationships between requirements. Version 2001A includes the ability to treat linked files as a requirement and trace other requirements to your linked files.

RequisitePro also provides various views to enhance traceability. One of those views is the Traceability Matrix. This matrix displays requirements in a matrix format for easier coverage viewing. The matrix will provide visual feedback about what system requirements were derived from which customer requirements. Using the matrix, it is also easy to check coverage and make sure that all of the customer requirements were broken down into system requirements. Another useful view provided by RequisitePro is the Traceability Tree view. This view shows the requirements in a hierarchical fashion. The benefit of this view is in graphically showing relationships between requirements. If a requirement is modified, added or deleted, the user can visually see all of the other affected requirements. The affected requirements can then be properly scrutinized and modified to accommodate the original requirement change. This helps maintain a cohesive set of requirements by eliminating orphaned requirements and also by preventing outdated requirements from being left in the set.
RequisitePro also offers cross project traceability. Often times, especially with legacy systems, a number of projects will spawn off of a central project. These new projects will share a significant number of requirements with its parent and sibling projects. RequisitePro allows traceability of requirements to span cross-project. This greatly increases requirement reuse which can in turn foster design, code, and test reuse.

2.6.6. RDT (Requirements Design & Traceability) by Igatech

(http://www.igatech.com)

RDT supports several mechanisms to aid the user in requirements analysis and identification. These include a parser that imports text documents then identifies requirements by key words and structure. The tool provides functionality for deriving, allocating and assigning requirements and acceptance test procedures. Requirements can be traced from top level requirements down to the lowest level requirements. The tool is able to classify/categorize requirements during identification using requirements attributes. In addition the tool provides capabilities to capture architecture, functional decomposition and WBS in graphical format and display data as a tree view of requirements. RDT is able to generate documentation directly into MS Word, including requirements and test specifications, requirement allocation matrices, parent-child relationships and design documents. New features incorporated in version 3 include:

- The ability to share data between different sites, and the facility to collate this data back to the master database.
- Revision control, which allows users to look at all changes made to data, and when and by whom these changes were made.
- An RDT AxiomSys Bridge exists that allows the bi-directional transfer of requirements and tests between any part of the project database in RDT, and the software or system model(s) in AxiomSys 6.0.

2.6.7. RTM (Requirements Traceability Management) by Integrated Chipware Inc.
(http://www.chipware.com)

RTM supports multiple users working on the same requirements at the same time by implementing locking control on a requirement-by-requirement basis. RTM's toolset supports the ability to capture graphical information as traceable requirements objects. The tool utilizes the native tool, which created the graphics object. A class definition tool is included that allows the user to model any type of hierarchical project data (requirement document, hierarchies, system element structure and WBS). Once the hierarchy is defined generic relationships can also be established to allow cross-reference link information to be established between any active data item. Version 5.3 of RTM includes the following capabilities:

- An information modelling capability allows users to design change records or problem reports and associate them with specific requirements data.
- A complete test management solution including information concerning schedules, resources, test verification and results versus requirements.
- User defined forms to allow users to view information in familiar layouts.
- Change request capability allows users to propose and review changes to the current baseline requirements from within RTM.

2.7. Open problems

As it has been described in this section, the research efforts in the field of RT share the use of formal graphs or formal languages to represent the relevant entities and the traces between them, as well as the idea that the huge mass of traceability information produced need a software tool to be managed. In some cases, the use of this kind of tools suggests the possibility of an automation of the tracing process. Last trends in RT focus on adding explicit semantics to relationships, in particular for traces that involve conceptual design elements or pieces of the application (codes, classes, use cases, etc.). Furthermore, some works consider design as the result of a requirements refinement process and try to record this process establishing a requirements-to-design traceability.
The analysis of the state-of-the-art literature, as well as my personal experience in academic and industrial projects, highlights a number of open problems that need to be faced in order to improve both the quality and the acceptance degree of RT approaches. The negative elements related to current traceability methods and practices may be classified in problems of: (i) adoption, (ii) context, (iii) communication, (iv) specification, (v) tool-dependency and (vi) guidelines.

(i) Adoption problems
In industrial practices and projects, traceability of requirements is still a neglected activity, that managers do not know or do not understand or do not want to adopt. In fact, traceability is perceived not as a cost-effective phase of a project but as extra-work, since its benefits are often not clear in short terms. Moreover, time spent using current traceability approaches may becomes 50% of the entire project; an industrial case study [Ramesh, 2002] found that the cost of adoption of a traceability technique was more

than twice the normal documentation cost associated with the development of a system of similar size and complexity. This problem can be related to the fact that most of RT methods need often a quite long training time to be properly understood and applied and that, if adopted, they require a too long time to be accomplished; the NEESgrid example [Finholt, 2003] is symptomatic: in that case, traceability has required an effort of 140 man/hours.

(ii) Context problems

In current methodologies, the impact of strategic requirements is considered towards technical design features or application elements. Industrial experiences [Randazzo, 2005a] show that a wider "experience design" should be considered, including organisational elements, activities, roles, workflows etc. together with the technical-applicative aspect. From this point of view, traceability may be considered as an "under control complex picture". Current practices focus on the traceability of singles objects instead; these approaches are based on the use of conceptual tools that emphasise a "punctiform" view, i.e. a representation model highlighting one-to-one relationships but that does not consider the global picture as a whole.

(iii) Communication problems

Communication between the different actors of a project is not only a possible benefit of a traceability approach, but it is also a need when the results of an analysis have to be known and understood by the overall project work-team. Unfortunately, this aspect is not particularly stressed in current RT practices and some methodologies have strong access problems for their users. In fact, often RT methods are not fully understandable and clear, since they base their expressivity power on formalisms, structured graphs or formal languages; this can become a problem when the target of the traceability documentation is not a software engineer: in projects also managers, marketing people, graphic designers, etc. need to understand the results of the RT analysis and their consequences on the entire application. This kind of communication problems causes that benefits of the adoption of such an approach are not perceived by the main decision-makers – in relation with the adoption problems previously discussed. Another communication problem can be identified in the fact that current methodologies do not include explicitly a preliminary plan for RT: often benefits are not perceived because the analyst did not discussed with the project manager the goals this activity should reach, the expectations generated in relation to the application quality and the time, budget and resources constraints for this phase.

(iv) Specification problems

Current RT methodologies seem to assume that in each project, formal and precise specifications and documentations are naturally produced and provided. In real cases, this is not necessarily true; nowadays even big projects with huge budgets can have lack of documentations. Requirements may be not completely recorded because some particular goals or visions or some axioms or some motivations stay implicit between the work-team members; the requirements specification may also be out-of-date in particular moments of the project or at its end. In the same way, design specifications may be not updated during the project and may become unaligned with the actual application implemented; furthermore, design usually does not record all the relevant elements: in fact, in some kind of application (e.g. Information Systems or Educational Applications), extra-application elements such as workflows, roles, actor activities, etc. play an important roles in the decision-making process. Finally, both requirements and design specifications may be produced with a variety of methodologies or practices and mixing important information in formal and informal ways. In other cases, some kind of documentation may even be missing.

(v) Tool-Dependency problems

While it is difficult maintaining the huge mass of dependences among the many objects produced by a large software system development effort, some current approaches require the use of a software tool to become usable and manageable. "All or nothing" methodologies consider traceability the art of finding all the possible relationships between the greater number of elements and not only relevant traces between relevant elements; therefore, a big project may produce a huge mass of requirements, pages of design elements and a poorly understandable net of traceability relationships between them. For this reason, some methods are completely tool-based and cannot be applied without specific software applications. This can become a problem mainly during meetings and discussions, where the old "paper and pencil" method or a wall board are still the best way to facilitate reasoning, changing and debating between the different members of a work-team. Furthermore, commercial tools are often not too efficient in managing tracing's complexity: they have problems in maintaining relationships concerning artefacts expressed in natural language, often ambiguous, or artefact created independently by non-interoperable tools and that evolves autonomously.

(vi) Guidelines problems

In either the standards or the current literature there is a remarkable absence of clear guidelines on what traceability information must be captured and on how they should be used. In other words, there is not a clear activity support to guide non-expert analysts in setting-up a RT activity, in managing the information, in trace relationships between them, in record these traces, in understanding the consequences for the system under development and in communicating and presenting these results to project managers and designers.

Some of the problems here discussed will be faced by the TRAMA approach in order to find a consistent solution towards its industrial acceptance degree and its quality in terms of benefits perceived. The intrinsic usability of the method will be assured by following some principles, presented in [Triacca, 2004]:

- the tracing process have to be engineered and standardized;
- the method have to be systematic;
- the reusability of the method have to be enhanced in different fields (making TRAMA cost-effective);
- the notation of the method have to be as simple as possible, easily learnable, flexible, modular and scalable.

3.Case studies

<<Real programmers don't comment their code. If it was hard to write, it should be hard to understand .>>

Anonymous

Abstract

In this section all the case studies on which the different versions of TRAMA was applied will be described. Since this thesis describes empirical research, each case study brings key elements to improve the method, to modify it, to refine it, to test it and to provide at the end a general approach. The sequence of case studies will therefore trace an history of how TRAMA has been developed, as well as example of use of the method. Both academic and industrial case studies will be described; academic cases, (3.3, 3.4 and 3.5) are linked to research projects or to courses taken at the University of Lugano, Switzerland, or at the Politecnico di Milano, Italy, while industrial cases (3.1, 3.2 and 3.6) refer to particular studies or works performed for companies, institutions, museums, etc. The studies were conducted from November 2002 to May 2005. Each case here described has the same presentation structure: (i) a global description of the project, (ii) the goals of the traceability analysis, (iii) a summary of how the method was applied, (iv) the benefits achieved for the project under study and (v) the lessons learnt from the experience. The complete reports of these case studies can be found in the Annexes section.

3.1. SEE - Shrine Educational experience (November 2002)

(i) Project description

Since July 2002 the Israel Museum, in specific the "Shrine of the Book" section, keeper of the precious "Dead Sea scrolls", started a cooperation with the Hypermedia Open Centre (Italy) in order to build an innovative experience related to the fascinating topic of the scrolls found in eleven caves, near the archaeological site of Qumran by the Dead Sea. The experience includes a cooperative 3D environment, in which every visitor have the possibility not only to move, to manipulate objects and to chat with the other visitors, but also to perform "unusual" actions such as flying, looking through other visitors' eyes, whispering with someone in particular, etc. The application's primary target are schools of all the countries, with students aged between 13 and 19 years. The main goal is "edutainment", that is, to entertain and to be educational at the same time. The educational benefits can be synthesized as follows:

A. Increased knowledge about "Dead Sea Scrolls" and related issues, that can be of various nature (religious, historical, technical, social, etc.).
B. Possibility of intercultural "meetings" in a virtual space, with students (possibly) of different countries or of different cultures.
C. Possibility of practicing an innovative and engaging form of interaction, using virtual environments and set ups. The games students will be invited to perform will also have the role of consolidating "team-ship", creating relationships and ties among different schools.
D. Possibilities of getting acquainted with state of the art Information and Communication Technologies, modern multimedia, graphic, web and internet technologies.

A detailed storyboard rules each 45 minutes session in which students meet and interact. Short lectures alternate with engaging cultural games. Educational and cultural materials are downloadable from a 2D site. Students, represented by "avatars", meet in the 3D world and run through three "sessions". There are no more than 9 avatars (8 students and the museum guide). Between sessions, students are asked to make a research based on the comparison between Qumran culture and their own culture.

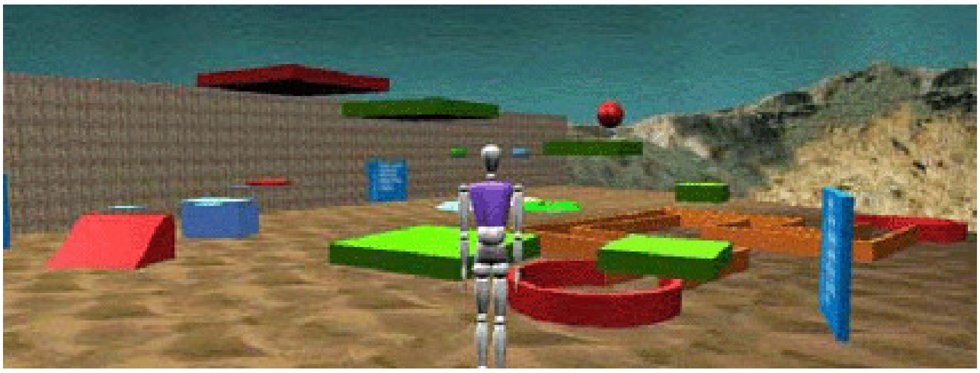


Figure 6. An image of the SEE 3D game space

(ii) Traceability goals

SEE project managers asked to TEC-Lab (the Technology Enhances Communication Laboratory at the University of Lugano) for a methodological help in re-organizing the huge number of documents and

material produced to describe the project. A traceability analysis has been therefore conducted to keep the rationale of the overall project and I have been charged to perform it.
Firstly, I identified the actors that could have a specific goal or interest in such an analysis: the project manager, the Israel Museum, the experience designers, the software developers and the educational institutions involved. As mentioned before, the project manager's goal was to keep the consistency between the different pieces of documentation produced and to re-organize these materials so that they could be a tool to communicate the project status to the various team members. The Israel Museum started this project to link its name to a technologically advanced project and to spread the knowledge about the "Dead Sea Scrolls"; the Museum was therefore interested in this analysis in order to verify if these goals were reached by the experience. Designers of the experience was also interested in the analysis in order to verify (and to demonstrate) the consistency and completeness of their choices in relation with the project's strategic goals. Software developers wanted to verify the consistency of their work in relation with design and they need a support to evaluate the impact of suggested design modifications. Finally, the educational institutions involved needed to verify the consistency of the experience with their study plan.

(iii) Analysis

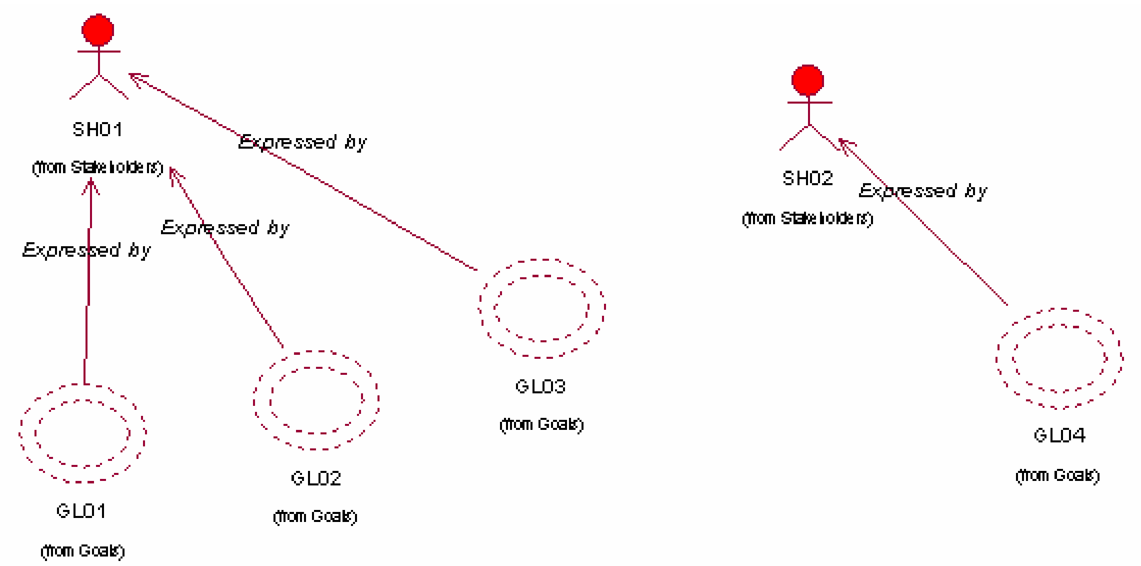
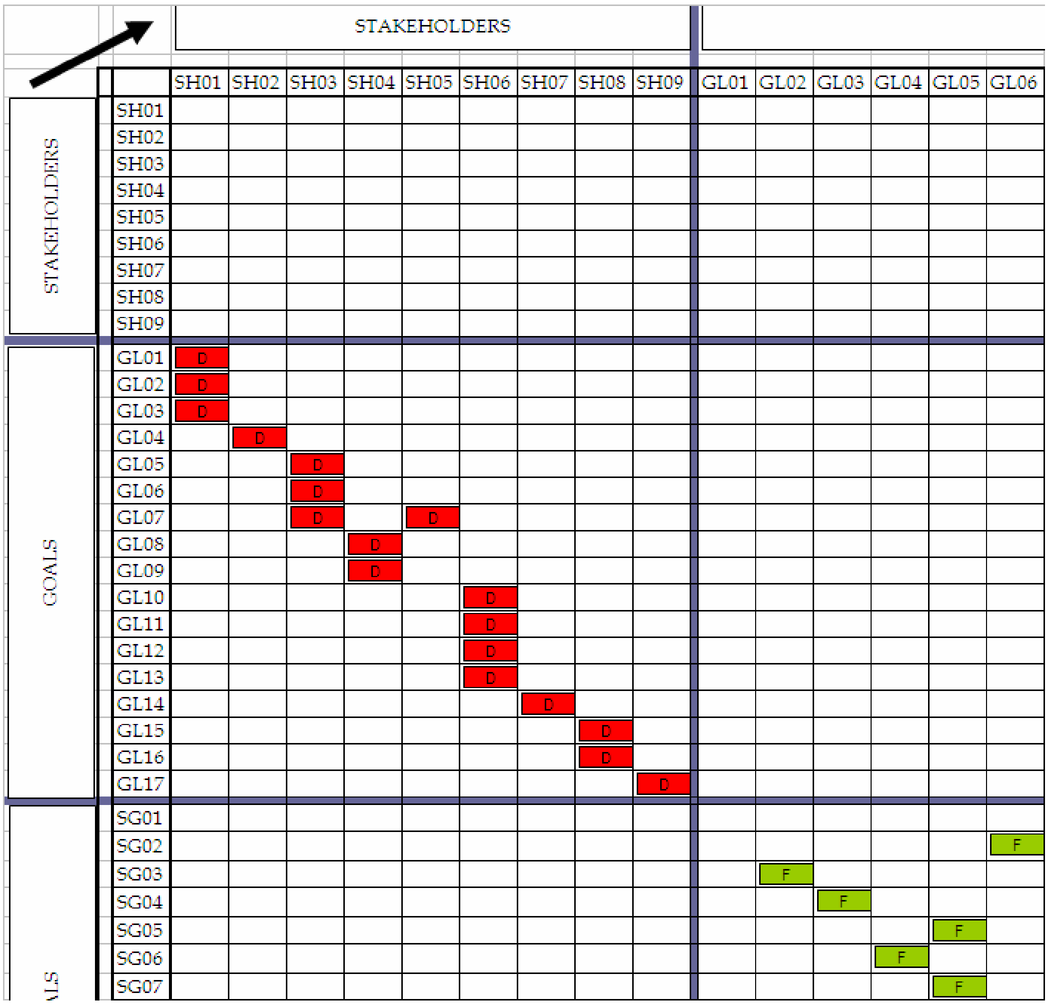


Figure 7. An expert from the UML-like schema of the SEE stakeholders and goals

Since a full requirements documentation were missing, the first step of my traceability work was to perform a reverse requirements specification activity, rebuilding the specification from design, from some documents explaining the project and from an interview with the responsible of the designers team. I based this activity on the AWARE methodology [Bolchini et al., 2003] and on its UML-like notation, as exemplified in Figure 7. During the analysis, 9 main stakeholders, 17 goals, 16 sub-goals, 46 requirements (functional and non-functional) and 3 constraints have been identified. In the same way, design elements have been re-organized in terms of sessions (3), activities (16), rendering (4), games

(2) and quizzes (1). To all these elements a unique ID has been assigned: e.g. *GL03* stands for "goal 3", *DO11* stands for "design object 11", and so on.
After this preliminary activity, relationships between these project elements have been traced. The traceability information was inferred from some documents describing the educational benefits of the project and from a short interview with the project manager.



		STAKEHOLDERS														
		SH01	SH02	SH03	SH04	SH05	SH06	SH07	SH08	SH09	GL01	GL02	GL03	GL04	GL05	GL06
STAKEHOLDERS	SH01															
	SH02															
	SH03															
	SH04															
	SH05															
	SH06															
	SH07															
	SH08															
	SH09															
GOALS	GL01	D														
	GL02	D														
	GL03	D														
	GL04		D													
	GL05			D												
	GL06			D												
	GL07			D		D										
	GL08				D											
	GL09				D											
	GL10						D									
	GL11						D									
	GL12						D									
	GL13						D									
	GL14							D								
	GL15								D							
	GL16								D							
	GL17									D						
ALS	SG01															
	SG02															F
	SG03											F				
	SG04												F			
	SG05														F	
	SG06													F		
	SG07														F	

Figure 8. A detail of the SEE traceability matrix

In a first attempt, exemplified in Figure 8, a simple matrix was used to represent the elements and the relationships between them. Both on the vertical and horizontal dimensions, stakeholders, goals, sub-goals, functional requirements, non-functional requirements, constraints and design objects were reported by the mean of their unique ID. The filled crosses represent traces between these elements. Different colours and letters inside the crosses represent different semantics assigned to the relationships: R in yellow for refinement, D in red for dependency, O in blue for "operationalisation" and F in green for fulfilment.
The second attempt was to transform the matrix in a graph, using a UML-like notation to represents the traceability entities and the different kind of relationships between them. The different elements was

represented with different shapes and the different relations were labelled with their semantic; each element can to be linked with any other element in the graph and multiple cardinalities were allowed. The result is what can be seen in Figure 9.

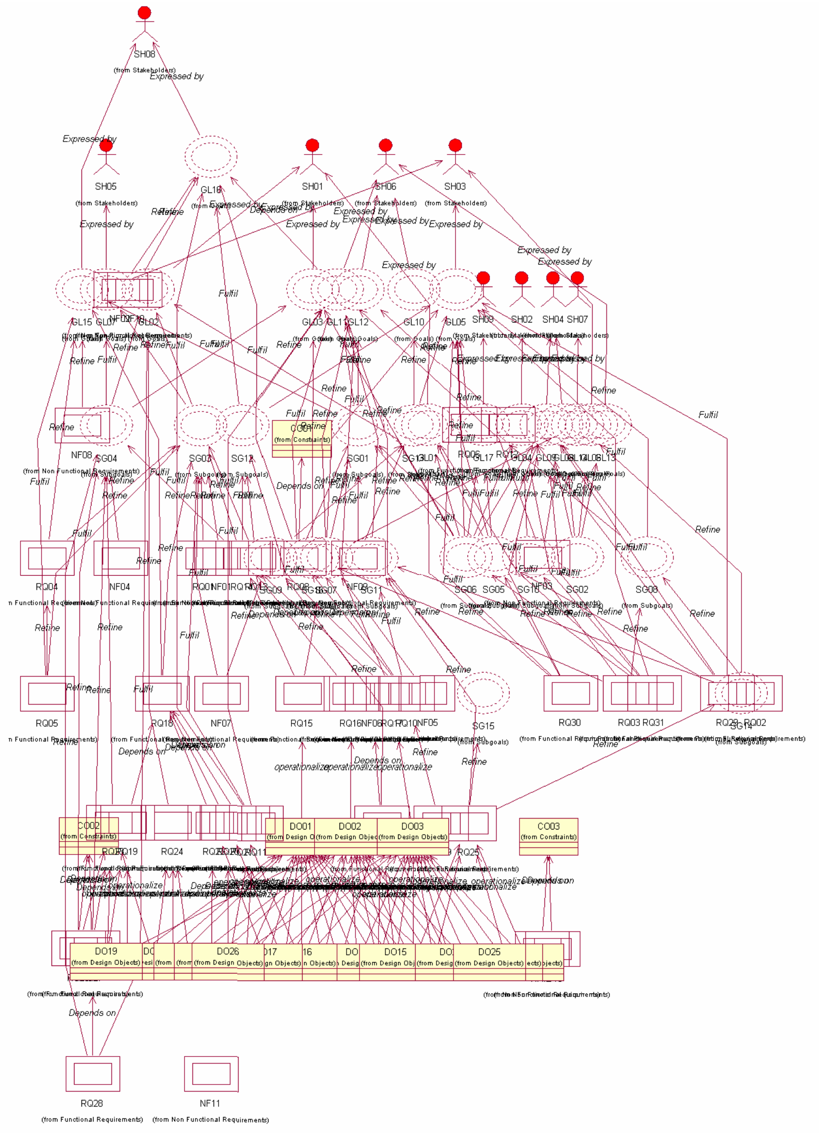


Figure 9. The complete SEE traceability graph

(iv) Benefits

A first huge benefit was the re-organisation of the project documentation: requirements and design specification have been re-produced in a consistent form enabling reasoning and confrontation between them. Discussions about requirements and design with the project responsible allows also to update these documents consistently with the actual state of the experience. Finally, the traceability specification has been used in the project as a compact summary of the project status, enabling its efficient communication to the overall work-team and towards the Israel Museum.

(v) Lessons learnt

Elicitation. As it is clear from the benefits declared in the previous paragraph, traceability relationships founded for the SEE project have not been used *per se* but as a tool to understand consistently the overall project. Most of the information have been inductively taken from existing documents describing the project and only two short interviews added some background details about the motivations of some choices. This case showed that traceability information do not surface naturally but they need to be elicited from stakeholders: these are really "new" information that cannot be inferred form existing elements or from existing stakeholder knowledge. In fact, some traceability information are understood, for instance by designers, during the analysis and were never been consciously thought before.

Redundancy. In this first attempt I tried to combine a pre-RST and a post-RST approach, establishing relationships not only between design elements and requirements-related information, but also between requirements and goals, goals and stakeholders, etc. This piece of pre-RST is of course useful in the analysis phase, but it is maybe a problem of a requirements management method more than of a traceability method. In fact, a number of requirements methodologies[32] provide support to keep the relations between stakeholders, goals, sub-goals and requirements. Therefore, this kind of analysis has been redundant, it was a simple repetition of what it was already done before; the fact can be seen as a problem not only form an efficiency point of view but also for the representation techniques, that "exploded" because of an excess of information reported.

Representation. Both the matrix and the graph have been in this case almost useless because of their dimensions and the fact that they are difficult to read and to understand. None of the two representation tool experimented in this case have been efficiently used in the analysis phase because it was not possible to understand a global picture of the results and of the current analysis status.

3.2. Munch und Berlin Exhibition – version 1 (April 2003)

(i) Project description

The analysis is concerned with the development of a web site for the "Munch und Berlin exhibition" at the *Staatliche Museen zu Berlin* (Germany); the exhibition has taken place from April the 12th to July the 13th 2003 and was produced by Dr. Sigrid Achenbach.

The design of the website www.munchundberlin.org represents the first practical result of the WED project developing a linguistic approach that considers the interaction of a user with a web site as a dialogue. This web site is optimized for visually impaired people, where the interaction is more natural, like in an oral dialogue. An example is the page schema, a short summary (orally read but invisible in the page) of the basic sections of the page that the screen reader reads before reading any other content. The page schema enhances accessibility under two aspects: it gives the user the possibility to decide which section s/he's interested in and it helps memorizing the page structure, being based on consistent templates which facilitate the user navigation and orientation.

[32] E.g. KAOS [van Lamsweerde et al., 1998] i* [Yu, 1993] and the method I used in this case, AWARE [Bolchini et al., 2003].

Figure 10. The homepage of www.munchundberlin.org

(ii) Traceability goals

Before to put the application on-line, a consistency check have been requested to "adjust" the last elements and to fix an up-to-date documentation of the overall project. TEC-Lab performed a traceability analysis focusing on the conciseness and on the understandability of the documentation to provide.

(iii) Analysis

In this case too, as for the SEE project, an explicit and precise requirements specification was not available. The first activity was therefore to describe stakeholders, goals and requirements for this web site using the AWARE methodology [Bolchini et al., 2003]. The result of this preliminary activity can be seen in Figure 11. On the other hand, design was already documented by the use of a tool raised out from the WED approach: IDM [Bolchini et al., 2005a]. The subsequent traceability analysis was divided in two aspects. First, the impact of requirements into design was investigated; how requirements were taken into account in the design and which elements in the design answer to a specific need? Then, the reasons of design choices were analysed and documented; why a certain solution have been adopted in the application and how design elements can be justified according to the project's strategic goals?

For the first part, i.e. the analysis of requirements impact, a simple notation have been chosen to represent and discuss how requirements have been considered and "solved" by the mean of the application. Figure 12 shows how pieces of notation from AWARE (at the left) and from IDM (at the right) have been used to represent a relationship, expressed by the dashed arrow, between a requirement and a design element.

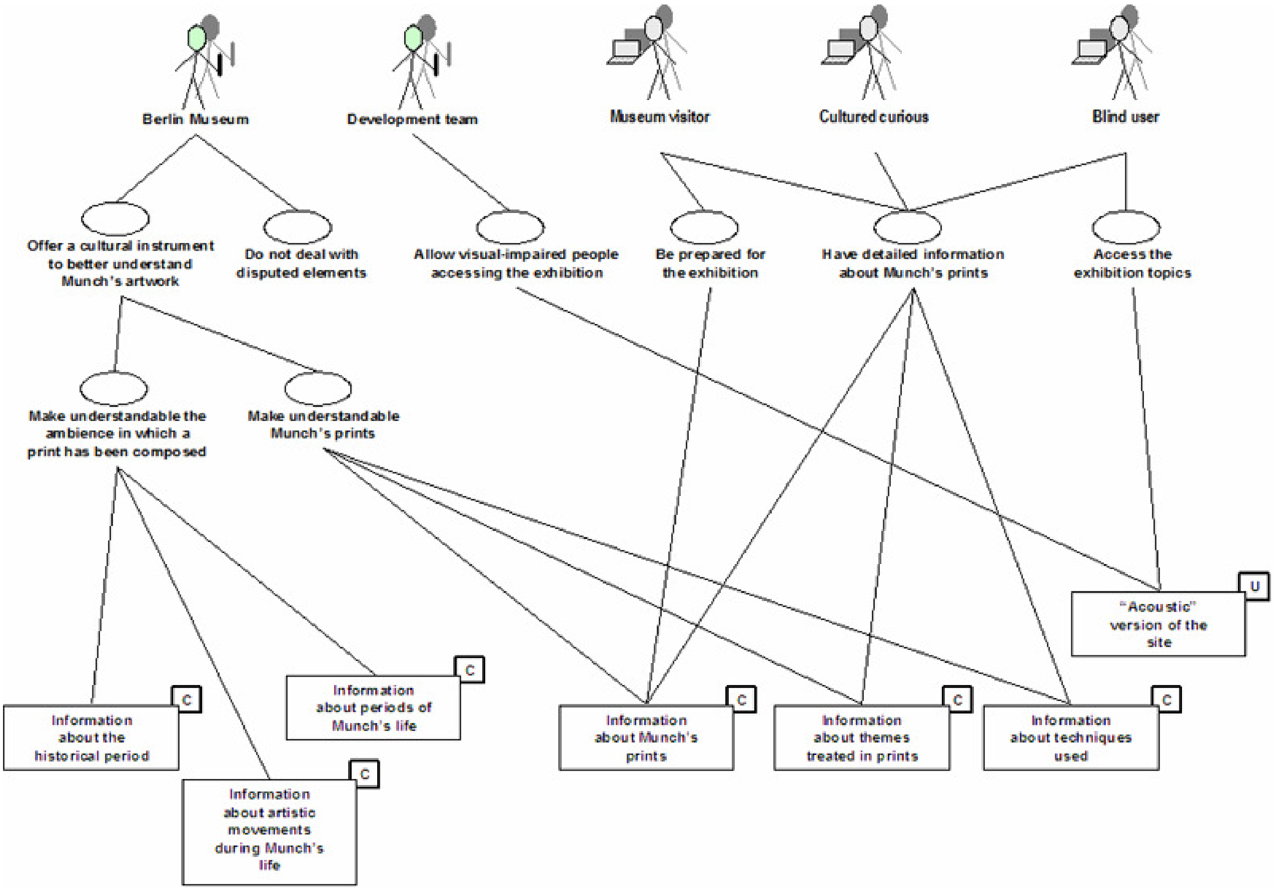


Figure 11. AWARE schema for the Munch und Berlin Exhibition site.

The main goal of the Museum is to offer a cultural instrument to better understand Munch's artwork; this means that the site have to give information about relevant topics of Munch's work, such as techniques used.

Figure 12. An excerpt from the specification of requirements impact for Munch und Berlin

As can be seen in the example, each trace is coupled with some notes expressed in natural language. Each note reports comments about the relative relationship and better explain why and how the requirement impacts on the design element. A requirement may impact on a single element, on more than one element or on no elements; the three cases are considered and may indicate an excessive answer to the requirements or the absence of a specific application element that fits with that requirement.

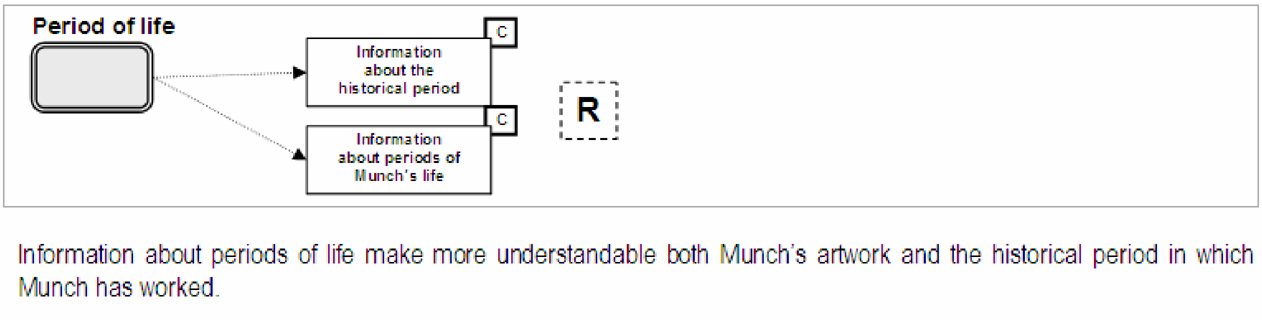


Figure 13. An excerpt from the specification of design reasons for Munch und Berlin

For the second part of the analysis, i.e. the design reasons, a similar notation have been adopted. In this case, exemplified in Figure 13, the IDM elements are at the left, the AWARE elements are at the right and the arrow express a relationship between a design element and the requirement(s) it fulfils. In this case too, some notes are couples with each relationship, commenting the reasons why a certain design choice have been taken. Reasons are divided in categories, following a specific taxonomy: [R] to indicate that the design artefact fits with a specific requirement, [P] to indicate that the design artefacts comes from a project designer's choose and [D] to indicate that the design artefact comes from a particular understand of the application domain. In most of the [P] and [D] cases, the AWARE elements in the notation are missing because no specific requirements represent the source for that design solution.

(iv) Benefits
The traceability analysis helped in better identifying a proper justification for all the elements in the design; this preventive activity allowed the final application to be accepted for publication. Moreover, the understanding of which elements were designed to answer to a specific need or according to a specific vision and of which ones were designed for other reasons helped in identifying those elements that could be slightly modified with no effect on the overall application quality in terms of requirements coverage. The traceability specification as a consistent and up-to-date document reporting requirements, design and interdependencies between the two has also been used to communicate the project status to the overall work-team. As a secondary benefit, the TRAMA analysis highlighted some weak points which helped in formulating some suggestions for further improvements.

(v) Lessons learnt
Step by step. This new approach to the analysis forced a step-by-step activity: an element at time has been considered to understand its traceability implications. In the "requirements impact" part, requirements have been considered one at time, trying to understand the impact of each one on the design; in the same way, in the "design reasons" part each design object has been analysed separately trying to understand the reasons of its presence in the application. As a consequence, a more structured and analytical activity has been possible to be performed.
Focus. The analysis relegates in a preliminary phase the relationships between stakeholders and goals, between goals and sub-goals and between sub-goals and requirements, describing these aspects on a requirements re-organization activity. The traceability analysis in itself focused on the contrary just on the relationships between requirements and design elements, distinguishing between a first phase treating

requirements impact and a second phase treating reasons for design choices. The selection of the aspects to consider in the traceability analysis and the articulation of this analysis in two steps seemed to be particularly useful in terms of usability and clearness of the method.

3.3. Pompei Archaeological Site (December 2003)

(i) Project description

This case is concerned with the development of a web application about the Pompei archaeological site; the prototype application has been developed by Politecnico di Milano (Italy) for the ministerial authorities in charge to manage the Pompei heritage. An encyclopaedic and more institutional web application is currently online and should not be replaced or replicated. The application that was the subject of the analysis aimed not at describing analytically the archaeological site but it should be more "applicative", enhancing the visits quality and number in Pompei. The main objectives of this new application was therefore twofold: from one hand, it should allow the user to visit Pompei "consciously", i.e. understanding better and in a more detailed way what she/he is going to see or what she/he has just yet visited. Contexts of use was therefore the house of the users, before or after a visit. Some computers and kiosks would be placed in the park as well, just for demonstration: this solution is poorly functional but strongly promotional, in a web marketing perspective. On the other hand, the application should present to (potential) visitors a different key to understand the archaeological park throughout thematic paths and it should provide in a clear and simple way information about what Pompei was before the Vesuvius eruption, in order to attract the user in visiting it. As a subordinate goal, the application should attract in visiting also the wider vesuvian area around Pompei. The original characteristic of this application was its attention to the accessibility problem: the web site were developed with a novel technology that go behind the current approach enabling a more involving access experience for visual-impaired users.

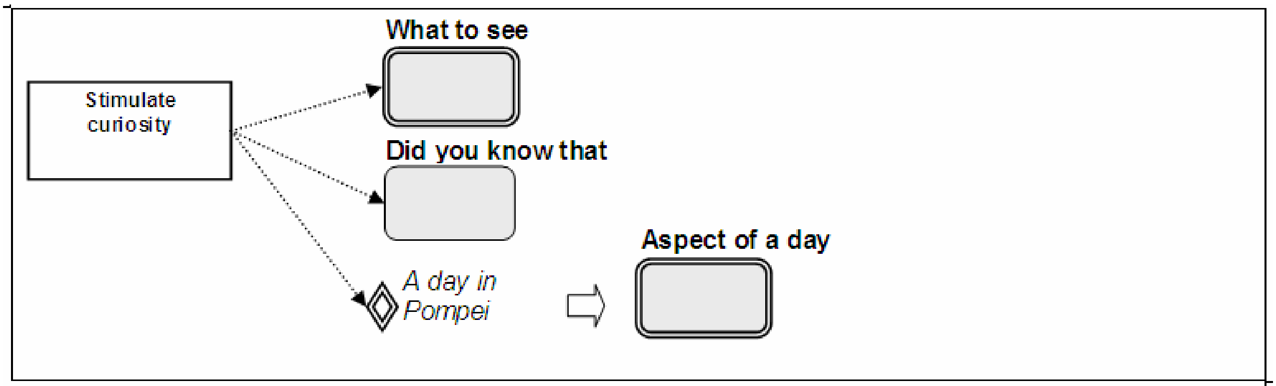
(ii) Traceability goals

TEC-Lab were charged to assist the Politecnico di Milano team in analysing the current application status, organising the traceability information in order to fit with two main goals: refine and align the requirements and the design documentation and pave the grounds for refining and correcting the design in a stakeholders and goals-oriented perspective.

(iii) Analysis

In this case, both requirements and design specification were formally described in specific documents. Requirements were represented using the AWARE notation [Bolchini et al., 2003], while design were described through the IDM methodology [Bolchini et al,, 2005a]; both methods have been developed at the University of Lugano and were shared by all the team members.

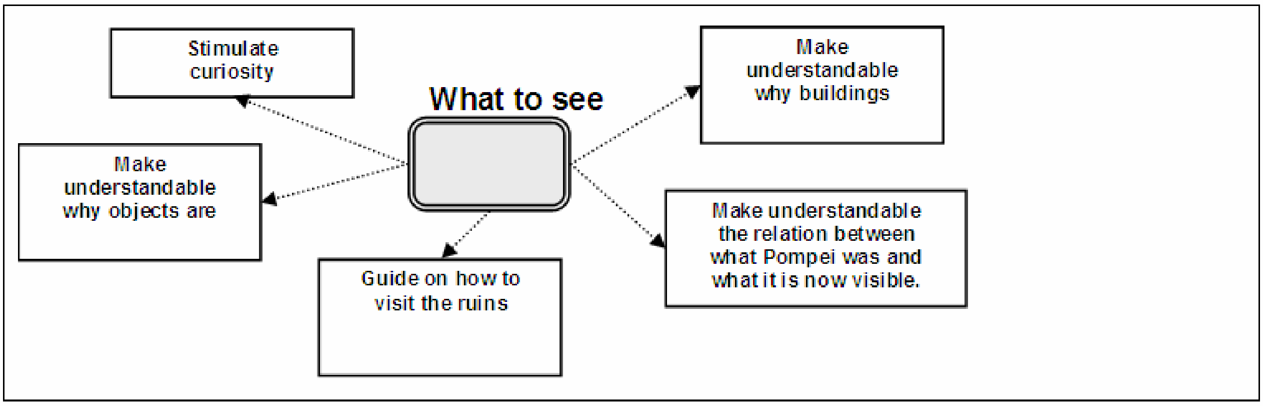
Firstly, requirements to design traceability were taken into account, identifying and commenting the impact that goals and requirements had on the actual design. The same notation as in paragraph 3.2 was used, as it is shown in Figure 14. Requirements and design elements were related by simple arrows; each relation was discussed autonomously and its impact, taxonomy and relevance was described in natural language. The impact explain what it means that a certain requirement has an impact on certain design elements, i.e. how and why requirements are taken into account in the design. The taxonomy tries to classify traces according to their scope; in this case I founded relations linked to a communication strategy, relations linked to an understanding of the domain or cause-effect relations. The relevance establish an importance rate for the relationship described, trying to find priorities between them; the priority (low, medium or high) is calculated in terms of impact level of the requirements on the design according to the designers opinion.



- IMPACT: Three strategies to stimulate curiosity: a series of curious and attractive anecdotes (did you know that), a structured review of interesting thins to watch (what to see) and guided tours that reproduce the aspects of a day in Pompei.
- TAXONOMY: Relation linked to a communication strategy
- RELEVANCE: High

Figure 14. An excerpt from the Pompei's Requirements satisfaction model

Then, design-to-requirements traceability were considered in terms of reasons of design choices. In other words, relationships between design decisions and the reasons for which these decisions have been taken, were expressed using the notation exemplified in Figure 15. Again, to indicate the source of the decision, a specific taxonomy has been used: [R] to indicate that the design artefact fitted with a specific requirement, [P] to indicate that the design artefacts came from a project designer's choose and [D] to indicate that the design artefact came from a particular understand of the application domain.



R >> It helps in attracting the visitor about some details to see and in clarify why something is visible or not. Furthermore, the "what" to see may be the source for good advices about elements to focus on during the visit, highlighting their relationship with the past.

Figure 15. An excerpt from the Pompei's Design justification model

(iv) Benefits

In this case traceability information have been used in a very active way to improve the first version of the design. Some weak points were highlighted and some points, where the requirements were poorly solved in the design, were surfaced from the analysis. In the same way, the strength points were highlighted and, thanks to the analysis, all the decisions were appropriately justified for the project sponsors.

(v) Lessons learnt

Taxonomies. If the classification of design reasons appeared more or less useful as a starting point for a further more detailed explication, taxonomies attached to each requirements impact relationship were not used in any way and seems for the moment useless.

Relevance. A similar observation can be done for the relevance indication in the requirements impact model, where priorities appeared as too subjective and their semantic was not perceived as very clear.

Notation. The AWARE and IDM graphical elements used were very clear and understandable for the work-team; in fact, each members previously knew these methodologies. But when this documentation were proposed to the responsible of the Pompei archaeological site, it needed further explanations.

Panoramic. This kind of representation did not make possible to have a global picture of the analysis, helping stakeholders to understand what happens or to participate in the analysis itself.

3.4. Museum of Non-European Cultures (November 2004)

(i) Project description

The Museum of non-European cultures ("Museo delle Culture Extraeuropee") in Lugano assembles the collection of objects from Oceania, Africa and India. Although the collection is culturally significant, due to poor management and lack of promotional activities on the part of museum and city officials, it was virtually unknown in the local community. As a result, the museum received very few visitors, which led the city of Lugano to propose closing the building in 2003. Objects in the collection were to be sold or loaned to other ethnographic museums in Europe. A local citizen group successfully challenged this proposal and, in 2004, the city agreed to reappraise the museum's situation. Following this reappraisal, the city is now planning to invest money and resources to re-launch the museum. A permanent curator will be appointed in the coming months. In addition, they are considering developing a website and other interactive applications to support the re-launch.

(ii) Traceability goals

TEC-Lab and the Master in Technology-Enhanced Communication for Cultural Heritage (TEC-CH) received the task to design a general purpose website for the museum. As present no website exists and the only information available online is a QuickTime VR tour of the gallery which is located on the city of Lugano site. The traceability study performed in this case[33] had as goal the refinement of the first design produced by two participants of the TEC-CH Master in a feasibility study.

(iii) Analysis

As a first step for the traceability study, I reorganised and "normalised" in a structured way the huge amount of information raised out from the documentation provided by TEC-CH feasibility analysis. These information have been segmented in stakeholders, visions, users, motivations, goals and requirements. In this case, the design was already expressed in a structured way, in terms of topics, relevant relations and group of topics; the model used was IDM (Interactive Dialogue Model).

After this preliminary step, the analysis of requirements impact and of design reasons have been performed. The relationships raised out from this analysis have been represented into simple matrices that I called RIM (Requirements Impact Model) and DMM (Design Motivations Model).

The RIM matrix produced for this project and exemplified in Figure 16 lists vertically the requirements-related information, i.e. stakeholders, visions, users, motivations, goals and requirements, and horizontally the design elements in term of topics, relevant relations and groups of topics; the crosses filled by a "X" represent a relationship between a requirement and a design element. I considered this matrix line by line (goal by goal), answering the questions: "what is the impact that this specific goal has on the design?" and "which design elements fit with this goal or answer to this need?". I filled in this way the matrix, finding all the strategies used in the design as solution to problems and needs highlighted in the requirements phase.

[33] This case study is also described in [Randazzo, 2005b]

		TOPICS															
		Object	Themed tour	Artist	Interactive feature	Visual quiz	Visual comparison	Temporary exhibition	Activity / Event	About the museum	Visit the museum	The collectors	Contact	Site map	Visual quiz INCLUDES Object	Visual comparison ILLUSTRATED BY Object	Object INCLUDES Visual
VISIONS	Stakeholders are essentially united in the desire to see the museum stay open																
	We have to support the re-launch of the museum								X	X	X						
	Users will have very little motivation to visit our website or the museum				X												
	Our target audience has little or no interest in Oceanic or Extra-European art				X	X	X								X	X	
AKEHOLDERS	Museum director		X		X	X			X	X	X		X				
	Curator	X	X	X		X	X	X		X	X	X			X	X	
	Citizen group (who petitioned the city on behalf of the museum)							X	X	X	X						

Figure 16. An excerpt from the RIM for the Museum of non-European cultures

The DMM matrix used in this project and partially reported in Figure 17 lists vertically all the design elements in terms of topics, relevant relations and groups of topics, and horizontally the possible sources of the specific choice. "Sources" are justifications or motivations of the presence of a specific design solution in the application; these motivations can be retraced to a requirements-related information (a vision, a goal, etc.) or to other reasons such as a specific understanding of the domain , the designer expertise and skills, technology, "graphic" or budget constraints and laws obligations.

(iv) Benefits

This analysis allowed to identify some problems and inconsistencies included in the first design. Some goals where in fact not properly supported by the application and some motivations where not completely clear. The analysis highlighted also some aspects that were overemphasized respect to the high-level goals, and some other aspects that were relegated in a secondary plan and that should be considered differently. Finally, the traceability activity helped in make clearer the relationships between the different application parts and their strategic role, providing a useful communication tool to express the main benefits of the project.

(v) Lessons learnt

Separation. Requirements analysis and traceability analysis have been perceived and organised as two different and separate activities and not as part of the same project phase. Tracing involves an analysis of reasons implied by solutions chosen and an understanding of consequences of these choices.

Motivations. The analysis highlighted that reasons why a certain solution has been adopted may be different than "simply" answer to a specific requirement but other sources such as usability principles or compliance with a general framework can be identified.

Sources. In particular, one may observe that design topics had in general a motivation in one or more requirements, since relevant relations and groups of topics have been produced answering to designers specific ideas and principles of "good design".

Communication. This analysis is an excellent way to reason about how to present the project to stakeholders and about how to highlight its benefits in a stakeholders' needs-oriented way.

		Visions	Stakeholders	Goals	Users	Motivations	Requirements	Specific understanding of the domain	Designer expertise	Technology constraints	"Graphic" constraints	Budget constraints	Laws obligations
TOPICS	Object						X						
	Themed tour			X			X						
	Artist						X						
	Interactive feature			X					X				
	Visual quiz								X				
	Visual comparison			X				X					
	Temporary exhibition		X	X			X						
	Activity / Event		X	X			X						
	About the museum		X						X				
	Visit the museum				X	X							
	The collectors		X										
	Contact				X	X			X				
	Site map								X				
	Visual quiz INCLUDES Object							X					
	Visual comparison ILLUSTRATED BY Object							X					
	Object INCLUDES Visual comparison							X					
	Visual comparison INCLUDES WORK BY Artist							X					

Figure 17. An excerpt from the DMM for the Museum of non-European cultures

3.5. Munch und Berlin Exhibition – version 2 (February 2005)

(i) Project description

The analysis is concerned with the "Munch und Berlin exhibition" web site. The requirements analysis activity has been performed partially during the project and partially after the publication of the website. During the design process, the analysis has taken into account the curator of the exhibition as main stakeholder, eliciting its visions about the application and the strategic goals of the site. At the end of the project, in April 2004, a traceability analysis[34] has been performed to link the requirements material with the design solutions and to point out indications for improving the application. Even if the exhibition is now finished, the project team is keeping alive this web site for educational purposes.

(ii) Traceability goals

A further traceability phase has been conducted in February 2005[35] to cope with new and refined project goals. The new goals were the following: (1) to design a website which might work also as a fixed information kiosk in the museum; (2) to make the website more usable by visually-impaired users (refining the WED approach); (3) to promote knowledge and awareness about a temporary exhibition being hosted at the Museum (Munch's prints and drawings). Traceability was here performed to evaluate the impact of changing requirements and of proposed new solutions on the application.

(iii) Analysis

In this case study, the traceability analysis was performed after the official end of the project. The chief design architect has been interviewed to elicit the main knowledge about design motivations. During and after the project I tried to keep traces between visions, motivations and goals on one hand, and between requirements and design choices on the other. As described in paragraph 3.4 for the previous case study, to support the traceability activity and to represent its outcome, I used two simple RIM and DMM matrices: one considering the impact of visions, motivations and goals on the applications design (Figure 18), and the other one highlighting the types of motivations behind the design choices (Figure 19).

The Requirements Impact Model (RIM) allowed tracing the impact of the main goals (all owned by users and stakeholders) on the design. In this case only visions, motivations and goals were traced, because of the specific objectives of the activity that focused on identifying and evaluating the impact of changing project goals. This information assisted designers to check if motivations, goals and visions were understood and effectively interpreted during design. Moreover, it was shown if the application was fully compliant with the requirements and the product did not exhibit any unnecessary feature or functionality.

The Design Motivations Model (DMM) documented the sources of the design decisions. Also in this case, design choices could derive from different sources: from specific requirements or goals, from visions or motivations, from an understanding of the specific domain, from the expertise of the designer, or from constraints. I tried to provide the project team with a powerful tool for defending their choices with the client and proving that the solutions adopted fit with the strategic goals of the project.

(iv) Benefits

The rich knowledge gained during the interviews and the analysis enabled the project team members to understand the relationships between strategic goals and design solutions, i.e. to understand the main sources for the design decisions. Furthermore, traceability highlighted deficiencies of the website in terms of correspondence with (often never stated) requirements, and helped defining new and more effective solutions for the second release. These and other elements made the reasons behind the design better surface during ex-post traceability analysis and formed the basis for discussing possible improvements to be done for the second version of the site.

[34] See also paragraph 3.2

[35] The new tracing activity has been described in [Bolchini et al., 2005b]

DESIGN ELEMENTS	CONTENTS	Prints	Techniques	Periods of life	Artistic movements	Artists	Munch	The museum	The exhibition	Contacts	Credits	Listen to this Website	RELATIONSHIPS BETWEEN CONTENTS	From a print to the technique it was made with	From a technique to the prints made with that technique	From a print to the period of life during which the print was made	From a period of life to the artistics movements actives in Germany during that period	From an artistic movement to the artists that represent the movement	From an artist to the artistic movement he belonged to	ACCESS PATHS TO CONTENTS	A list of all the prints	A group of masterpieces	Some thematic tour of prints grouped by theme	A list of Munch's periods of life	A list of all the techniques
VISIONS																									
Frame works of art within historical background				X	X	X										X	X	X						X	
MOTIVATIONS																									
Be prepared for visiting the exhibition		X	X				X	X	X	X				X	X						X	X	X		X
Study Munch and his art		X	X	X	X	X	X							X	X	X	X	X	X		X	X	X	X	X
Appreciate the artworks in the exhibition		X	X									X													
GOALS																									
Design site & kiosk																									
Make site accessible		X										X													
Raise awareness on Munch's prints		X						X	X													X	X		
Provide leading themes				X	X	X										X	X	X						X	
Raise awareness on art. movement					X												X	X							
Raise awareness on polit. background				X	X	X										X	X	X						X	
Understanding what's interesting in the exhibition																						X	X		
Finding information about the paintings in the exhibition		X	X											X	X										
Finding information about Munch's life				X			X																	X	
Finding historical information about Munch				X	X	X	X									X	X	X	X					X	
Finding detailed information about Munch's work and art		X	X											X	X						X	X	X		X
Understanding the relevance of Munch in the history of art					X	X											X	X	X						
Finding information about the techniques used in the paintings			X											X											X
Efficiently accessing the exhibition's topics		X																				X	X		
Understanding the site structure and the browsing capabilities												X													

Figure 18. The RIM from the Munch un Berlin Project

CHOICES SOURCE	Visions	Motivations	Goals	Specific understanding of the domain	Designer expertise	Technology constraints	"Graphic" constraints	Budget constraints	Laws obligations
CONTENTS									
Prints	X	X	X		X				
Techniques		X	X						
Periods of life		X	X						
Artistic movements	X		X						
Artists	X		X						
Munch	X		X						
The museum		X							
The exhibition		X	X						
Contacts					X				
Credits					X				
Listen to this Website			X						
RELATIONSHIPS BETWEEN CONTENTS									
From a print to the technique it was made with					X				
From a technique to the prints made with that technique					X				
From a print to the period of life during which the print was made					X				
From a period of life to the artistic movements actives in Germany during that period	X								
From an artistic movement to the artists that represent the movement					X				
From an artist to the artistic movement he belonged to					X				
ACCESS PATHS TO CONTENTS									
A list of all the prints					X				
A group of masterpieces		X	X						
Some thematic tour of prints grouped by theme			X	X					
A list of Munch's periods of life					X				
A list of all the techniques					X				

Figure 19. The DMM from the Munch un Berlin Project

(v) Lessons learnt

Checklist. Each matrix can be used as a checklist supporting the traceability analyst in considering the relevance and the meaning of each possible pair of objects. Here crosses were considered one by one, allowing a detailed and structured analysis and avoiding loss of information.

Rationale. Putting just an "X" in crosses between objects do not help in communicating correctly the relationship semantic and rationale: side-comments after each matrix were always needed for a complete and useful understanding of the traces. Maybe some short notes and comments inside each cross could be more efficient.

Direction. It is not clear in which direction to fill the matrices: the RIM could be filled horizontally, requirements to design, or vertically, design to requirements; the DIM could be filled horizontally, design to motivations, or vertically, motivations to design. A further investigation about this problem is needed.

Structure. The case shows very clearly that a single requirement or goal has very often an impact on a number of different elements or on groups of elements in the design; in this case was rarely possible to identify a one-to-one correspondence between requirements and design. This fact suggests the hypothesis that requirements and design have an inhomogeneous structure and that the impact of the firsts on the second should be investigated in terms of groups of elements which interplay contributes to answer to the need.

3.6. Learning at Europe (May 2005)

(i) Project description

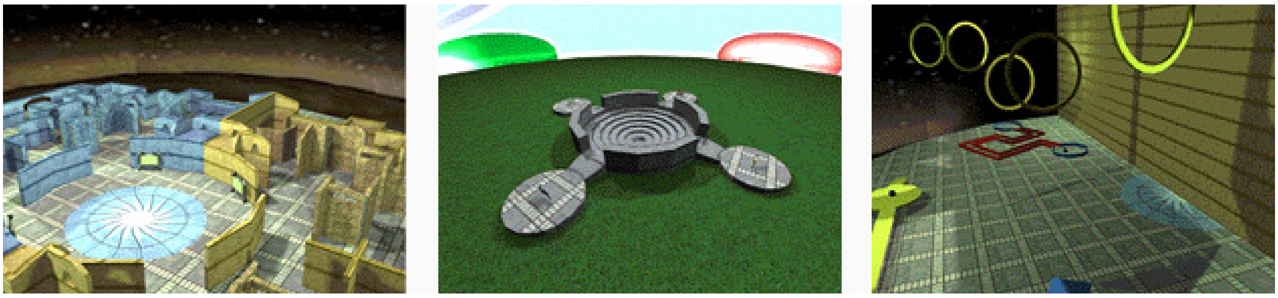


Figure 20. Some views of the L@E game space

LearningAtEurope (L@E) is an educational project aiming at fostering the development of a "European Identity" for the new generations of European students. L@E proposes an educational approach novel in several respects: advanced content, technology-enhanced e-Learning, a multicultural experience, coupled with engaging "games" and a cultural competition among different European classes. The project bases on the experience made with the SEE project, an educational e-learning experience about the "Dead Sea Scrolls"[36]. In a first full experimentation year, between 2004 and 2005, 48 classes from 6 European countries (Belgium, France, Italy, Norway, Poland and Spain), nearly 60 teachers and 1,000 students were involved. A new advanced experimentation year, between 2005 and 2006, will bring the project at an industrial stage. Before this new experimentation, a complete revision of the whole setting of the experience will be performed. A traceability analysis has been requested to facilitate this revision activity.

[36] See also paragraph 3.1

(ii) Traceability goals

This case study reports a traceability analysis for the L@E project, whose main goal is to reorganize the complex and various material describing and designing the experience, to pave the grounds for a reengineering activity. In particular, L@E team used this analysis for the following reasons: (1) internal communication, to communicate the project status to all the team members, (2) reverse requirements engineering, re-organizing and refining requirements and surfacing missing information, fundamentals to understand the project but never explicitly documented, (3) design tuning, surfacing missing design components and re-aligning the design with the project state-of-the-art and (4) design revision, to facilitate the project revision before a new experimentation period.

(iii) Analysis

The tracing analysis process for the L@E project, has been structured in: a preliminary plan, a basic information re-organization, two elicitation and analysis meetings and a specification activity.

The first preliminary activity helps in understanding why traceability was performed and which benefits would it bring to the project. These aspects were discussed during a first short meeting and a preliminary plan was produced, clearly summarizing these goals and setting up the subsequent activity. Two meetings of four hours each were established: a first one with the aim of bring together the various elements of the project and to start tracing the first relationships between the set of goals/requirements and the design elements; a second one, the day after the first, with the aim to refine the analysis considering one by one the motivations of the design elements.

The second activity was to re-organize requirements and design in a structured way; requirements and design documents were been produced, but not in a organized way: business or research goals were not distinguished from educational goals, technical elements of the application were mixed with the experience organization elements, etc. I called this activity with the name of requirements and design "normalization". The requirements "normalization" activity consisted in structuring the previous knowledge in terms of general goals, educational goals, visions and requirements. Since it was impossible to understand the project solutions without considering contextual information such as the format, the procedures, the workflow, the activities of users, etc., in the design "normalisation" activity five design categories were taken into account: static components, i.e. the "bricks" the experience is composed by; dynamic components, i.e. how static components are assembled in a workflow; transversal components being both static and dynamic or no one of the two; educational materials, i.e. contents of the educational experience; testing materials, i.e. all the elements used to measure the educational impact of the experience.

As in the two previous case studies, a RIM and a DMM matrix have been produced to trace the information related to the L@E project. Two particular aspects have to be highlighted in this case. First: the matrices have been discussed and filled with the project manager and the designers, during the two main meetings; a big paper was hanging on the wall (as in Figure 21) and the RIM and DMM was completed, discussed and refined directly by the work-team. Second: to annotate in real-time the observations and the reasons of designers, the crosses between the elements in the matrices were filled with a short comment about the why and how of such a relationship.

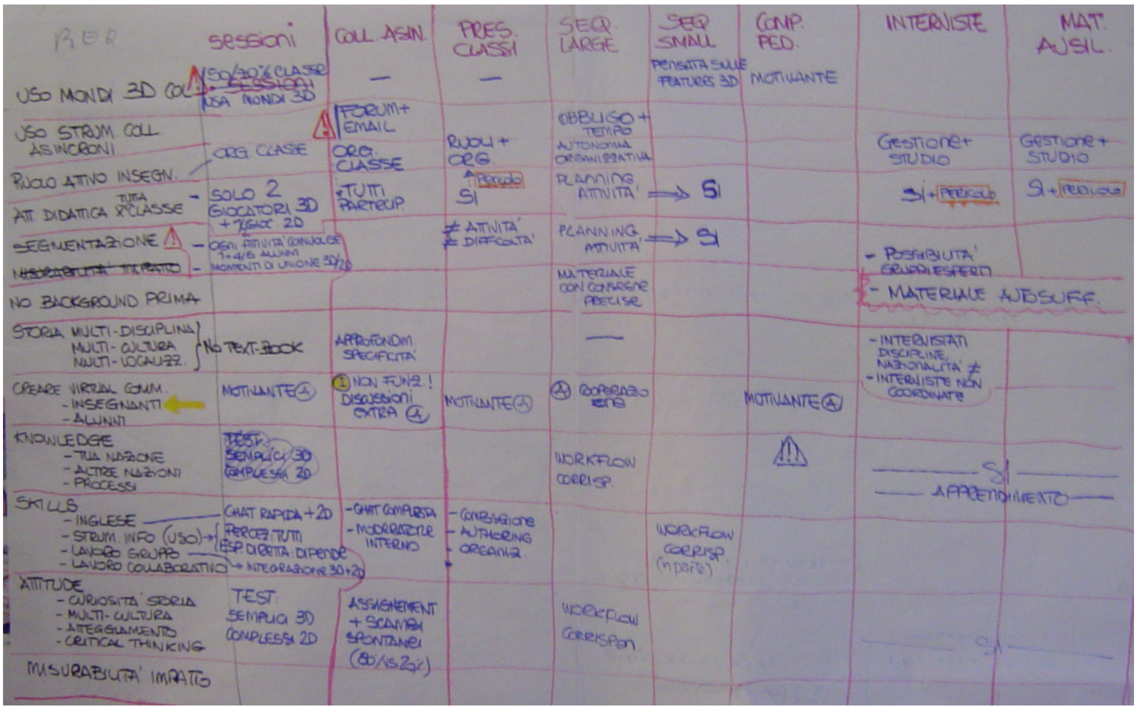


Figure 21. The RIM produced after the first meeting with designers and the project manager

	STATIC COMPONENTS				DYNAMIC COMPONENTS	
	D1 3D synchronous collaborative sessions	D2 Asynchronous collaboration	D3 Class presentations	D4 Games	D5 In-the-large sequence	D6 In-the-small sequence
EDUCATIONAL GOALS						
B1 Knowledge					Yes: workflow for knowledge	
B1.1 About local (national) history						
B1.2 About other countries' history						
B1.3 About general historical concepts						
B2 Skills						Workflow correspondance
B2.1 Use of "professional" English	Quick chat + 2D	Complex chat	Composition			
B2.2 Use of technological tools	Perception: all; direct experience: it depends	Internal moderator	Authoring	Yes: use of 3D features		
B2.3 Group work	Integration of 3D and 2D		Organization	Fast decisions in group		
B2.4 Collaborative work						
B3 Attitudes	Test: 3D simple, 2D complex	Assignment + spontaneous exchanges (80% vs. 20%)			YES: adequate workflow	
B3.1 Sense of curiosity for history						
B3.2 Multiple cultures / multiple identities						
B3.3 Improved attitude towards history						
B3.4 Critical thinking towards knowledge						
B3.5 Different attitude towards knowledge						
VISIONS						
V1 Integration in schools' curricula	Technology and easy connectivity	Forum and email	Variable technologies		Acceptable effort demand	
V2 Characteristics of and educational competition				The most enjoyable part of competition	Scores ensure that nobody gets frustrated	
REQUIREMENTS						
R01 The experience have to include the use of collaborative 3D worlds	50% / 70% of the class uses 3D	NO	NO	Motivating - Designed to use 3D features		Designed to use 3D features

Figure 22. An excerpt from the RIM matrix for L@E

The RIM matrix reported in Figure 22 lists vertically all the requirements-related information and horizontally all the design components in the terms described before. The crosses represents relationships between these elements and each cell can hold a comment about the "rationale", the reason and the meaning of the relationship. In L@E, cross cells have been filled according to two directions. The first one considers the matrix vertically, design element by design element; the question that designers with the help of the traceability expert have tried to answer was: "Taking into account a single design element, how does it fit with requirements?". The second one considers the matrix horizontally, helping the project manager to re-consider each cell, requirement by requirement, focusing on the real impact that each requirement have on the application.

During the second meeting, a more detailed analysis of the motivations for design decisions has been performed, using the DMM matrix in Figure 23. This tool has helped the team of L@E in understanding explicitly why certain choices have been taken for the experience and where they were allowed to perform changes. In fact, the DMM matrix highlights the design "sources" i.e. the arguments that justify the design.

	Visions	Requirements	Designer expertise	Specific understanding of the domain	Constraints	Law obbligations
STATIC COMPONENTS	****************	****************	****************	****************	****************	****************
D1 3D synchronous collaborative sessions		Offering to schools a collaborative learning experience based on new technologies		In past projects this technology has been a good motivation for e-learning	Use of the in-house 3D technology	
D2 Asynchronous collaboration		Create a community	Forums and emails are common tools to build up an asynchronous collaboration			
D3 Class presentations			Just to add an activity in the first (introductory) session			
D4 Games				Another good idea from past projects as a motivation for traditional learning		
DYNAMIC COMPONENTS	****************	****************	****************	****************	****************	****************
D5 In-the-large sequence	Acceptable effort demand	Workflow adequate to educational goals				
D6 In-the-small sequence			Workflow correspondance. Each session has not to be too "boring" or long or simple/difficult, etc.			
TRANSVERSAL COMPONENTS	****************	****************	****************	****************	****************	****************
D7 Educational competition in itself	It has to be a motivation for students in learning: open and serious			In past projects it has been a good motivation for e-learning		
EDUCATIONAL MATERIALS	****************	****************	****************	****************	****************	****************
D8 Interviews			This way to present information has worked very well in past projects			
D9 Auxiliary materials			This material go with interviews as background explaination (maps, biographies, etc.)			
TESTING MATERIALS	****************	****************	****************	****************	****************	****************
D10 Quick questions on knowledge	Values students' preparation, is not frustrating					
D11 Open-ended comprehension questions						
D12 Assignments & home-works		Improve knowledge, skills and attitudes				
D13 Monitoring Tools & Procedures		The educational impact has to be measurable				

Figure 23. The complete DMM matrix for L@E

The matrix lists vertically the design elements of the application and horizontally their motivations (sources). As in the previous case study reported in this dissertation, "sources" indicate the reasons because a specific design solution has been adopted and they have been of course related to visions and requirements for a part, but for another part they have been related to the following: the designer expertise, i.e. particular "good design" principles that are part of the designer's skills and that she/he applies in any case; a specific understanding of the domain, i.e. recurrent good solutions in a domain that the designer applied because she/he learnt it by other cases in the same domain; a particular constraint, e.g. budget limitations, time, technology limitations, etc.; a law obligation, e.g. copyright issues, personal data treatment, etc. In this case approximately 50% of design elements do not comes from requirements but from designer expertise or a specific understanding of the domain. The matrix has been filled horizontally, trying to answer the "why" question, design element by design element.

(iv) Benefits

L@E team uses this analysis profiting by the following benefits: *internal communication*, i.e. to communicate the project status to all the team members, to designers and engineers who implemented the application and who had just a partial understanding of the project, limited to what they did and developed; *reverse requirements engineering*, i.e. to force to a more structured vision of this knowledge, re-organizing and refining requirements and surfacing missing information, fundamentals to understand the project but never explicitly documented; *design tuning*, i.e. to force designers in distinguishing between details and base elements of the application, in order to surface missing design components and re-align the design with the project state-of-the-art; *design revision*, i.e. to provide all the information useful for the envisioned revision activity, highlighting the relationships between the project components and their priority related to requirements compliance, identifying mandatory design elements related to main goals or requirements, understanding which parts could be changed instead and surfacing some weak elements.

(v) Lessons learnt

The untouchables. In L@E project lot of documentation about educational benefits and elements of the project was produced but it was never clearly stated which elements was possible to modify without consequences for the overall requirements compliance of the application. This case seems to show that one can modify solutions related to requirements only if changes in requirements occur. If not, these solutions should be considered "untouchables". At the contrary, design element that are not strictly linked to a requirement, can be considered as solutions that may be changed or interpreted in a different way.

Meetings. Discussions with stakeholders and project team members are essential for an efficient traceability analysis. In this case, the elicitation and the analysis was conducted with the direct intervention of designers and of the project manager. If matrices have been profitably used as a tool to support the reasoning and to fix the opinions, a more precise help to set-up a similar meeting is needed.

Analyst. The role of the responsible of the tracing activity has been in this case similar to a facilitator in meetings; the analyst was a traceability expert helping stakeholders in expressing their thinking in a structured and analytical way.

Direction. RIM matrix have been filled both horizontally and vertically, according to two different points of view. The first one was the designers' point of view, who considered the matrix vertically, design elements by design elements, because each designer developed just a single part of the entire application. The second point of view is a more "client-centred" one and helped in reviewing the matrix according to the real impact of requirements on the design.

Phase. In this case traceability has been analysed after the first version of the experience was completely developed and tested. Some observations expressed by the project manager seems to suggest that sometimes a quietly detailed design is needed to profitably trace relationships towards high-level requirements. This experience shows that it is not completely useful to perform a tracing activity in the firsts steps of the project but that their results become interesting if performed after a first version of the design have been produced.

4.The TRAMA method

<<About half my designs are controlled fantasy, 15 percent are total madness and the rest are bread-and-butter designs.>>
Manolo Blahnik

Abstract
TRAMA can be described both as a process, i.e. as a sequence of actions divided into phases, and as a tracing approach, i.e. as a model including conceptual structures, tools, purposes, etc., as described in paragraph 1.2. This section presents first a tracing activity workflow allowing TRAMA to be properly applied; the workflow consists of the following activities: preliminary plan, information re-organisation, information "normalisation", elicitation, analysis, specification and validation. Then the section presents the TRAMA approach in terms of: purposes, processes, conceptual trace model, and tools.
Finally, the section discusses the benefits of the method and some of its limits.

4.1. Definition of TRAMA

This dissertation presents the results of a research about a method that I called TRAMA, a TRaceability Analysis Method for (interactive) Applications. At the beginning of the research, the method was conceived for being used on interactive application projects, but some experiences showed that its tools and its concepts could be applied in wider domains, including information systems, knowledge management systems, educational applications, etc. This is the reason why the word *interactive* has been put between brackets in the title.

TRAMA is a design traceability method supporting post-Requirements Specification Traceability in both a forward and backward direction. The approach bases on the use of structured matrices to facilitate the meetings with stakeholders and to analyse the surfaced information. The method helps in finding both impact relationships between requirements and design elements (forward traceability) and motivation relationships between design solutions and its sources (backward traceability). TRAMA supports a tracing activity even if requirements have never been documented or if the design development has not been explicitly and formally followed and recorded. The case studies presented in section 3 show that design is not the result of a requirements refinement activity *at least in those cases*. Therefore, the TRAMA method allows to support also those cases where designing is an intuition and induction process more than a derivation one. The approach simply does not take into account the mental process that brings from general requirements to concrete design solutions; in fact some requirements remain implicit at the beginning of the project but they are considered in design and often, at the end of the project, designers do not remember the actual reason for these choices. This causes problems that make very hard to verify, to evaluate, to revision and to reuse efficiently design solutions in relation with high-level requirements. From this point of view, the TRAMA tracing activity tries to move intuition and induction to more rational cause-effect motivations; the method does not repress or stiffen the design process, but it helps in better understanding the reasons for design choices and it forces to better make explicit requirements that are both implicit or unexpressed. TRAMA does not focus therefore on modelling the process of evolving requirements into design, but it pretend to provide to designer an effective tool conceived to discuss and analyse the design choices after they have been taken, in order to refine it according to the main requirements and in order to eliminate unmotivated elements.

The approach consists in a structured analysis process, in a general conceptual model of entities and relationships to trace, and in a set of conceptual tools supporting traces inquiry, analysis and documentation. TRAMA is based on traceability matrices which cross requirements with design in a forward direction and design with its sources (requirements, motivations, constraints, etc.) in a backward direction. Requirements-to-Design matrix called RIM (Requirements Impact Model/Matrix) can be filled and read both horizontally, highlighting how single requirements are taken into account into the design, and vertically, showing how a single design element satisfies the project requirements. Design-to-Sources matrix called DMM (Design Motivations Model/Matrix) traces back single design elements to the motivation why a certain decision is relevant for the project, e.g. satisfying a requirements, fulfilling a constraint, allowing more usability in the system, etc.

Before the detailed description of the elements included in the approach, a first question have to be answered: in which moment of a project life-cycle should one perform traceability? And in particular, in which moment should one perform the kind of traceability provided by TRAMA? As I showed before, this method helps in discovering, eliciting, analysing and documenting "ex-post" traces: TRAMA does not record the design process but helps designers in understanding both the impact of requirements in their projects and the motivations and the sources of specific design decisions *after* the design has been drawn. The experimentation results described in section 3 showed that a detailed design is possibly needed to profitably trace relationships towards high-level requirements and that relevant information have been surfaced after a first version of the design have been produced. The experiences suggest therefore to perform a tracing activity after the first design phase. A continuous activity during the rest of the project is then needed to keep the traceability specification up-to-date.

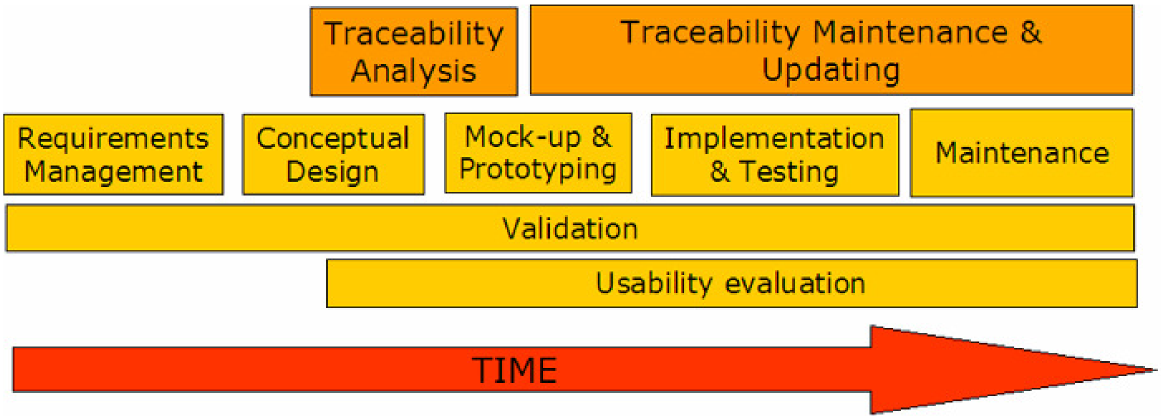


Figure 24. Traceability analysis in the project's overall life-cycle

4.2. A process guide

The first element of the TRAMA approach may be identified in the activity sequence that the method plan to be performed for an efficient and effective traceability analysis. A step-by-step process guide became therefore part of the proposed method. According to the Merriam-Webster dictionary, a process is "a series of actions or operations conducing to an end"; more in detail, the Wikipedia tell us that a process is "a naturally occurring or designed sequence of operations or events, possibly taking up time, space, expertise or other resource, which produces some outcome". TRAMA may be seen as a process, since it is composed by a sequence of activities designed to apply properly the method during a project. The activities workflow here presented should not be understand as a mandatory way to use TRAMA but as an help, a process support in the traceability practice. Furthermore, the sequence of action is not to be intended as a linear process but in a iterative way: each phase described identify the main activity to be performed but modifications or re-discussions of elements treated in previous phases are always possible.

As a kind of self-standing process, the TRAMA activity workflow is structured as follows:

1. Preliminary plan: understanding which the stakeholders of the traceability analysis, the traceability goals, the constraints (time and budget, related to ROI[37]) and the expected results are.

[37] Return on investment (ROI) is a straightforward financial tool that measures the economic return of a project or investment. ROI measures the effectiveness of the investment by calculating how many times the net benefits

2. Information re-organisation: understanding requirements and design from documents or from interviews with designers and organise it in terms of structured specifications.
3. Information "normalisation": structuring requirements and design information in "normal" terms, base on a strong methodology (e.g. AWARE for requirements and IDM for design).
4. Elicitation: surfacing relationships between requirements and design in terms of impact of requirements on the design ("How did you considered this requirements in the design?") and of motivations for design choices ("Why did you adopted this solution?").
5. Analysis: tracing relationship and developing the Requirements Impact and the Design Motivations Matrices (RIM and DMM).
6. Specification: documenting stakeholders, goals and analysis results.
7. Validation: checking the results with requirements analysts, designers, project managers and clients.

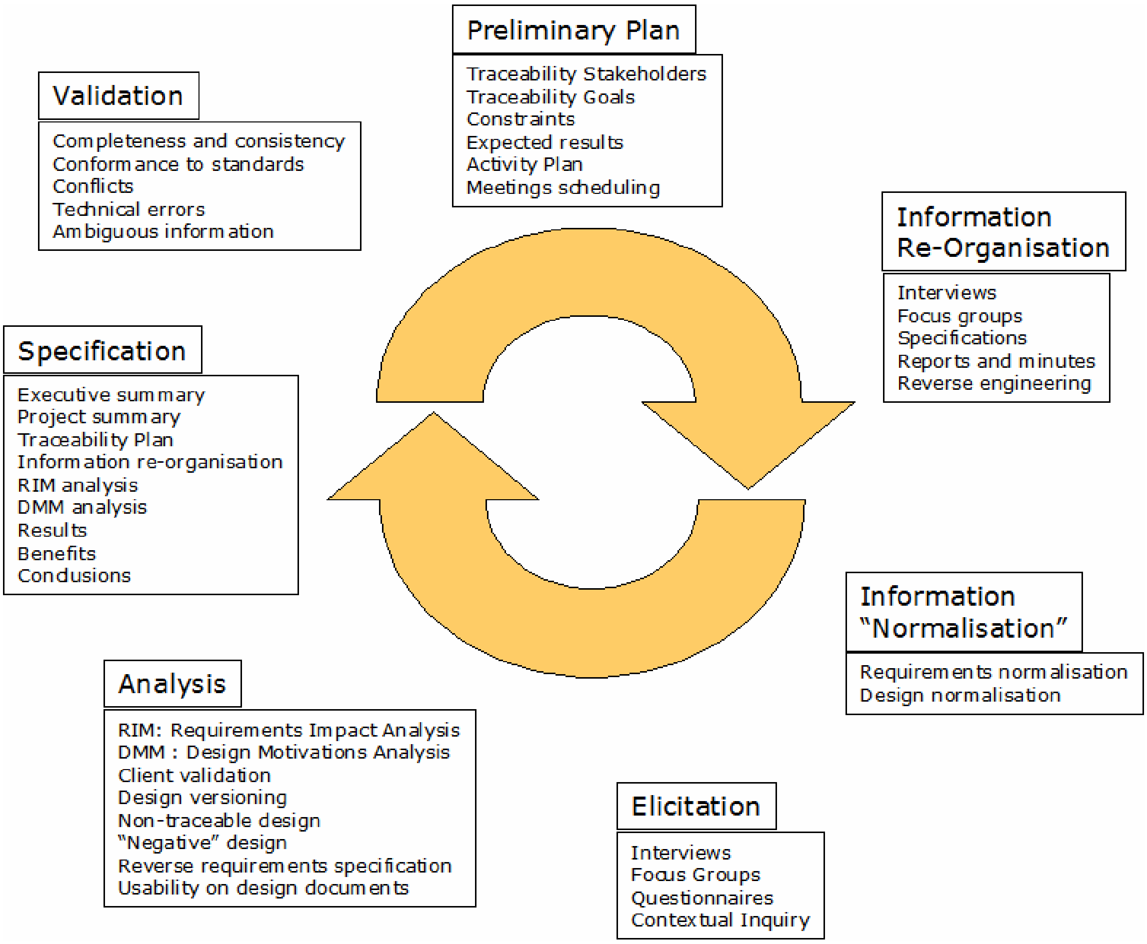


Figure 25. The TRAMA activities' iterative workflow

(benefits from investment minus initial and ongoing costs) recover the original investment [from: http://www.odellion.com].

4.2.1. Preliminary plan

Tracing requirements and design is a complex set of activities that can have very different purposes for a number of actors in a project's life-cycle[38]. The kind of information recorded and the analysis results differ due to which actors and which objectives are considered. From this point of view, a kind of "requirements analysis" for the traceability phase is required. In particular, it is essential to discuss with the "client" of this activity (usually, the project manager) for a preliminary traceability plan that includes: (a) the specification of who the stakeholders of this phase are, (b) a precise definition of all the goals that this activity is intended to reach, (c) all the time and budget constraint related to this phase and (d) which are the expected results of the traceability analysis. Furthermore, a setup of the tracing activity is also needed.

(a) Stakeholders

In the context of the traceability preliminary plan, a stakeholder can be defined as any actor of the application development-related activities which has a specific interest or goal in the results of the traceability analysis. The analyst discuss with a decision maker (the client or the project manager) about who are the people to involve in the tracing activity, both as sources of goals and opinion about the activity or as sources of useful information to surface traces between requirements and design elements. A stakeholders may therefore be the client itself, the project manager, the project planner, the requirement analyst, the designer, the verifier and/or the maintainer of the application. TRAMA has been tested with success for three particular kind of people:

- project managers, who use traceability information to control project progress and as communication tool with the client;
- requirements analysts, who use traceability information to check, refine and update the requirements specification;
- designers, who use traceability information to keep the consistency between design and requirements and to check the design compliance with strategic goals.

(b) Goals

As for every phase of a project's life-cycle, a precise definition of which the goals and the needs to fit are, is an essential element for the success of the phase itself. In this case, this is even more so true because of the variety of the possible purposes of a tracing activity. The analyst has here the responsibility to highlight what it can be done and what it cannot be done with such an analysis; during a first meeting with stakeholders, the analyst have to elicit the objectives of this activity, and have to help in selecting the aspects that could be more relevant for the stakeholders' needs. An "all purposes" analysis is not realistic in any case: first, time and budget could be serious constraints that limit the possible actions to perform during this phase; second, the experience shows that the more the traceability analysis' goals are focused, the more that analysis may be effective in terms of ROI. Pragmatically, the "magic number" of goals for this activity should be included between 2 and 4. More than four different goals risks to cause an activity overload and a negative costs/benefits balance.

[38] See also paragraph 1.1.3

(c) Constraints
Needs and desires often have to face with the actual resources provided for a certain activity. Tracing is not an exception: there will be always limitations of time and of budget in order to perform traceability in projects where the money spent is under strict control and where time-to-market is a quality measure of the production process. The preliminary plan have to define precise terms the effort needed for this phase, detailing the expected number of man/months, the number of days planned and the estimated cost for a traceability action in the project. Other possible constraints that the analyst have to preliminarily consider, may be particular law obligations (e.g. privacy issues) or other organisational elements.

(d) Expected results
A central element of a traceability preliminary plan is to define the expectation of the stakeholders about which benefits would the analysis bring to the project. The analyst should manage carefully these expectations, discussing it precisely in order to reach a common vision about what the tracing activity would give to the project. Different expectations about the results are usually the reason of a different perception of the success of this activity.

A last (but not least) argument to discuss in the preliminary plan is *the definition and the setup of the subsequent tracing activity*. Once defined the people to talk with to get and give information, their goals and expectations and the constraints included, the "actual" traceability phase can be performed in a structured way. According to the TRAMA method, this activity has to be carried on in strict collaboration with the different stakeholders. In particular, some meetings with project managers and designers have to be planned to elicit traceability information; these meetings can be planned only at this point because their number and duration depends on the activity's goals and expectations. Therefore, a complete activity plan have to be defined and described, including a meetings calendar, the main analysis phases, milestones, time, effort and costs for each phase.

4.2.2. Information re-organisation

TRAMA aims at discovering relationships between requirements and design and between design and its motivations. Therefore, to clearly discuss about this relationships with stakeholders and to avoid misunderstandings, it is needed to have structured and ordered elements both form the requirements and from the design side. In a perfect world, requirements information are explicitly organised and recorded during the project analysis phase and this specification is continuously updated during the project development; in the same perfect world, design is step-by-step documented in formal schemes and always kept aligned with the actual application implemented. Unfortunately, we do not live in a perfect world. In real-world cases, we should assume to have one or a combination of the following cases:

- *The requirements specification is unstructured or incomplete* - In the academic field, a number of beautiful requirements management approaches have been developed and tested. Unfortunately, one may observe that the penetration degree of these approaches in industrial practices is very low. In most of the cases, unstructured and informal approaches are used to record the information raised out form the firsts operative meetings. Sometimes there is not a clear and univocal perception of what a "requirement" or what a "goal" is: technical details of the applications and high-level visions related to a topic or a domain are mixed together with business-related expectations and application-related desires.

- *The requirements specification is absent* - In the worst cases, the requirements specification is not only confused or unstructured, but completely absent. In some projects the first recorded sign of what goals and requirements were, is the description of how the application is made. In frequent cases, the requirements specification is not used as a base to design the application, but it is an ex-post documentation used to describe the backgrounds of an existing product.
- *The design documentation is absent or incomplete* - For the design one can make the same observations that for requirements: often it is not clear what a design for an interactive application is, if it should describe all the technical implementation details or if it should be a conceptual picture of the applications contents, functionalities, navigation, etc. Sometimes this kind of specification is completely absent and just a technical documentation of how the application has been programmed is provided. In other cases, an unstructured specification of the elements of the application design is produced, but it includes a mix of indistinct contents, operations, navigation capabilities, organisation elements, roles, etc.

Sometimes, of course, requirements and design specification are recorded with scientific and formal approaches. Anyway, the TRAMA method cannot take this eventuality for granted but it should consider all the possibilities that can be encountered in the real world; TRAMA can therefore be applied anyway, no matter if there is previous documentation or not. For this reason, a main needed activity in this approach is what I call *information re-organisation*. This activity consists in understanding requirements and design information before to start the tracing process. The traceability analysts has somehow to understand what the goals, the requirements, etc. of the project and what the designed contents, functionalities, etc. of the application were; she/he has to "pick up" and to organise these elements in a requirements specification and in a design document. The information sources that the analyst may use to deduce the missing requirements or design knowledge may be the following:

- specific interviews or focus groups with requirements, analysts, designers, project responsible or other members of the work-team;
- existing documents, specifications, reports, minutes or annotations of some project meeting or activity;
- a reverse engineering activity, extracting the design form the actual application or (more difficult) inferring requirements from the design.

4.2.3. Information "normalisation"

This step is strictly related with the previous one. The knowledge gathered during the information re-organisation activity can be documented pragmatically using no matter what approach; only, the approach adopted have to answer to the needs of clarity, simplicity and correctness in terms of information structure. This concept will be clarify in the rest of this paragraph. TRAMA distinguish between the sub-activities of requirements normalisation and design normalisation.

Requirements normalisation is the activity of structuring the requirements-related information in a "normal" form. According to the Wikipedia[39] a normal form is "a representative element within an equivalence class, which is a simples or most manageable or otherwise tidiest and most desirable form, in terms of structure or syntax". In this case, requirements information should be transformed in a more manageable form in order to be traced towards the design elements which they have impact on. TRAMA do not impose the use of a specific approach to represent requirements but, for a better identification of traces, a goal-oriented methodology is suggested. I tested the method with KAOS [van Lamsweerde et al., 1998] and i* [Yu, 1993], but the best approach from a traceability point of view has been AWARE [Bolchini et al., 2003] in the experiences I made[40].

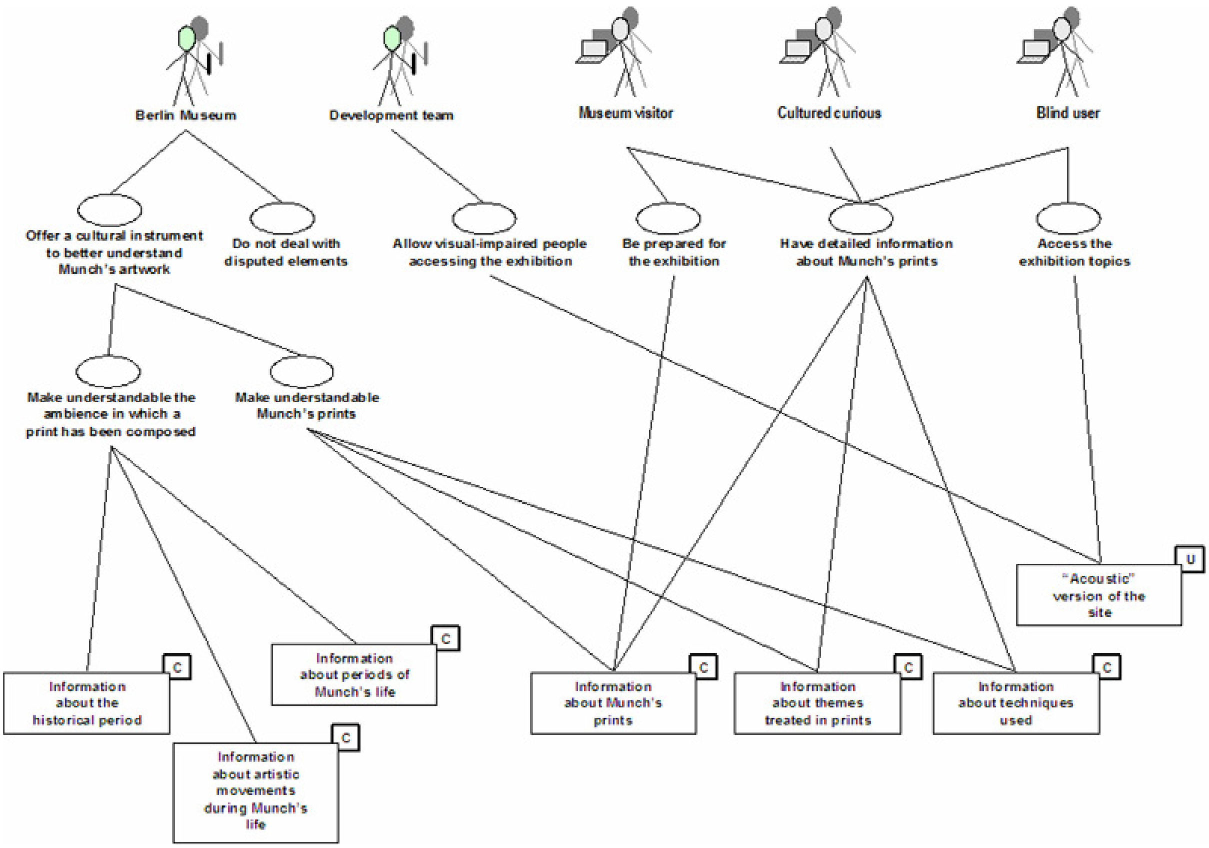


Figure 26. An example of AWARE schema from the "Munch in Berlin" project.

Figure 26 shows an example of requirements normalised with AWARE. In general terms, any methodology used to represent requirements-related information in TRAMA have to allow to structure the knowledge in

[39] http://www.wikipedia.org
[40] Some of these experience are reported in section 2.

terms of goals, goals refinement and requirements. In some cases, a support to document stakeholders, visions, users and motivations[41] could be also needed.

Design normalisation is a similar activity than "requirements normalisation" for the design. It consists in transforming the design knowledge gathered during the information re-organisation activity in terms of structured design [Woukeu et al., 2003]. Over the last decade, a number of structured design models and methodologies have been proposed for designing the features of an interactive application at a proper level of abstraction[42]. All these models have in common a short number of concepts, with different name but with similar meaning. In the case studies performed for TRAMA, a very powerful and agile model has been used: IDM [Bolchini et al., 2005a] the Interactive Dialogue Model based on dialogue primitives and characterized by a limited set of dialogic concepts used to shape the interaction between a user and the application. Figure 27 shows an example of IDM normalised design. Any approach one decides to adopt, TRAMA needs that the design methodology allows to describe how the interactive application will be and that it supports at least these kind of elements: contents, structure of contents, relationships between contents, access path to contents, navigation capabilities and presentation elements (pages, sections, etc.).

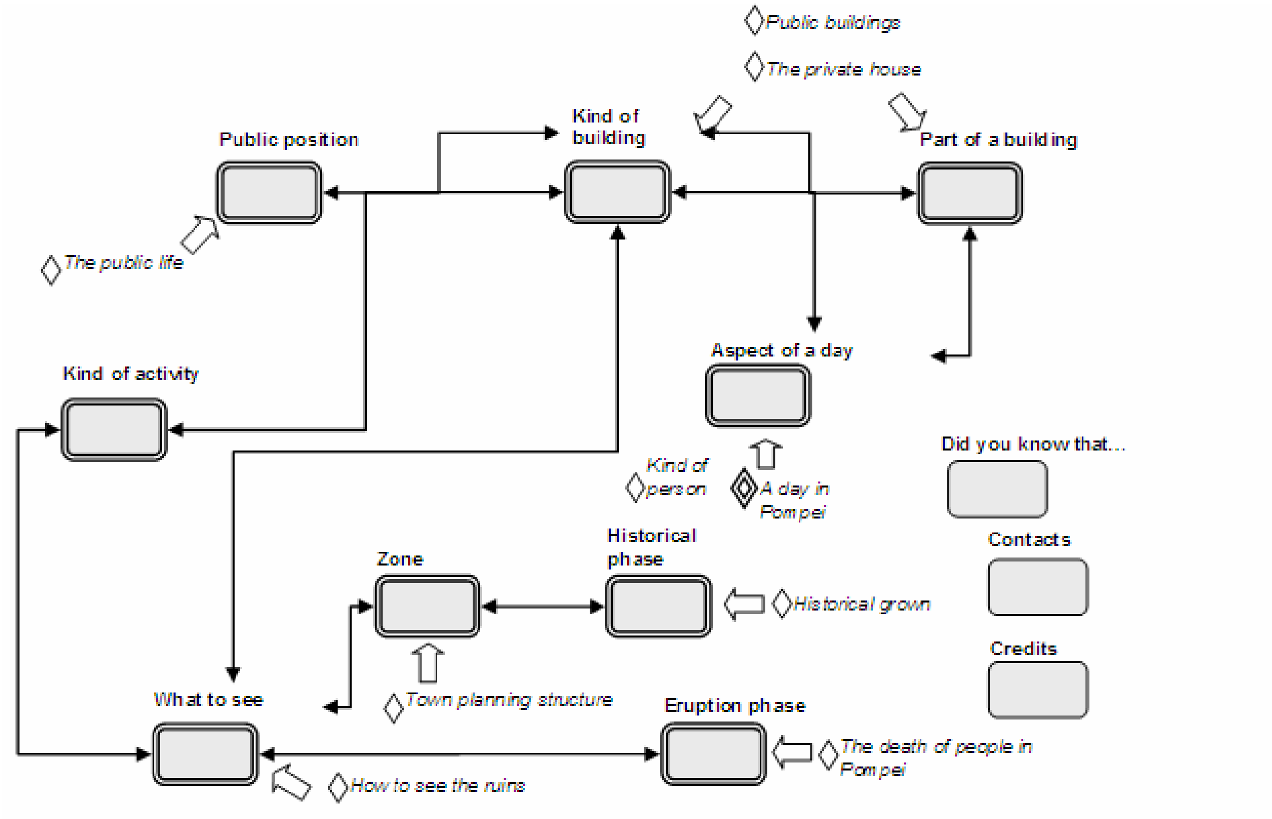


Figure 27. An example of IDM conceptual schema from the "Pompei" project

[41] All these terms will be better defined in paragraph 4.3.
[42] For a list of such methodologies, see also [Bolchini et al., 2005a]

But this is not enough. The TRAMA method is intended to produce a complex, global picture of the relationships existing between the different parts of an interactive application and its motivations, from one hand, and between requirements and their impact on design, on the other hand. Usually, the impact of strategic goals and requirements considers as target a structured design in traditional terms. But some experiences show that it is not possible to understand this global picture considering only contents, navigation, etc., i.e. considering only the "technical" design features or the application conceptual elements. In complex cases like information systems or, more again, like educational applications, it is impossible to globally understand the project solutions without considering "contextual" information. In other words a wider "experience design" should be considered, including as design not only the technical-applicative aspect but also organisational elements, the format, the procedures, the workflow, the activities of users, the roles, etc. These elements should be part of a good design properly normalised for a TRAMA tracing activity.

4.2.4. Elicitation

According to the Merriam-Webster dictionary, elicitation is "to call forth or draw out as information or a response (…) something latent or potential". In the Requirements Engineering field, elicitation is defined as "the process of identifying needs and bridging the disparities among the involved communities for the purpose of defining and distilling requirements to meet the constraints of these communities" [Christel & Kang, 1992]. In RT, and in particular in the TRAMA method, elicitation is intended as described below.

> Elicitation is the activity of surfacing relationships between requirements and design in terms of impact of requirements on the design ("How did you considered this requirements in the design?") and of motivations for design choices ("Why did you adopted this solution?").

In this phase the traceability expert adopt a number of different techniques to surface and understand all the relevant relationships existing between the different information previously re-organised and normalised. Traces are not "natural" information that raised out clearly from the development activity. Often the real motivations for an application choice remains implicit or unconscious or simply there is not a rational motivation: some decision could be taken just on the base of the designer expertise or for aesthetic reasons, etc. From this point of view, to record traceability relationships is not a simple observation of the reality but a fully elicitation activity, where new knowledge is created and surfaced. TRAMA supports this phase with a specific conceptual tool that is used both for elicitation and for analysis: the traceability matrix. According to the Wikipedia (http://www.wikipedia.org), "in a software development process, the traceability matrix is a table that correlates the high-level requirements (sometimes known as Marketing Requirements) and detailed requirements of the software product to the matching parts of high-level design, detailed Design, test plan (a.k.a. Test Outline), and test cases". In TRAMA, two kind of matrices are included in the method:

- A Requirements-to-Design matrix called RIM (Requirements Impact Matrix): the matrix lists vertically all the requirements-related information (e.g. visions, goals and requirements), and horizontally all the design components (e.g. contents, access paths, content structure and navigation capabilities between contents); the crosses represents relationships between these elements and each cell can hold a comment about the "rationale", i.e. the reason and the meaning of the relationship.

- A Design-to-Motivations matrix, called DMM (Design Motivations Matrix): the matrix lists vertically all the design component (as in the RIM's horizontal dimension), and horizontally their different motivations, answering to the question: "Why this element has been placed into the application?".

Different techniques and tools may be adopted to perform an efficient traceability elicitation phase, and different information sources may be used to surface this knowledge. There are not TRAMA-specific techniques, but all the general elicitation and requirements elicitation techniques can be used similarly in this case:

- *Interviews* are very common for this kind of activity because they allow a "live" contact with a person that could be a source of information. Here everything depends on the interviewer's skills and on the right selection of people to talk with. In large projects where many people are involved, this activity could take a lot of time.
- *Focus groups* are discussion meetings between the traceability expert and the project's work-team. Here again, it is possible a "live" contact with people working on the project, but it is not so focused as in an interview and only "public" opinions can be gathered; on the other hand, new knowledge may raised out from group discussion and a single meeting or a couple of meetings do not take so much time.
- *Questionnaires* can be used as a preliminary step in focus groups or interviews, just to set up the discussion agenda, or in the cases where too much people are involved in the project: there a combination of interviews (for the two or three project main responsible) and of questionnaires (for all the other project workers) may be used.
- *Direct observation* by following the entire project form the beginning can be an option in case of high budgets and large projects. Here a traceability expert follow the different project's phases as an internal observer and debrief step-by-step the motivations why the application is designed in a certain way. This technique presupposes many time and resources to be performed.

A combination of all these techniques is also possible. TRAMA has been applied with interviews and focus groups but never with questionnaires or contextual inquiries. In that cases[43], meetings have been setup in a standardised way:

- *Place*. A large meetings room with a table and some chairs. The room should have some free walls in order to hang up the papers with the matrices.
- *Tools*. A set of coloured pencils. A blackboard may be used but it cannot be taken away and flipcharts are too small for a traceability matrix. In TRAMA meetings some large-size papers have been used and hanged up on the walls in order to draw the matrices, as shown in Figure 28. Self-stick wall pads may also be a good solution.
- *Roles*. To carry on an efficient traceability meeting five different roles need to be covered:
  - the discussants, e.g. in focus groups the project work-team that animate the meeting;
  - a facilitator, i.e. a traceability expert in charge to address the discussion in a right direction, provoking answers, asking critical questions, etc.;

[43] See also [Randazzo, 2005a]

- a "wall writer", drawing the matrices on the wall papers and filling the crosses with the traceability information raised out from discussion;
- a secretary, recording and writing notes (on a PC) about the meeting;
- a chair officer, e.g. the project manager coordinating the overall meeting.

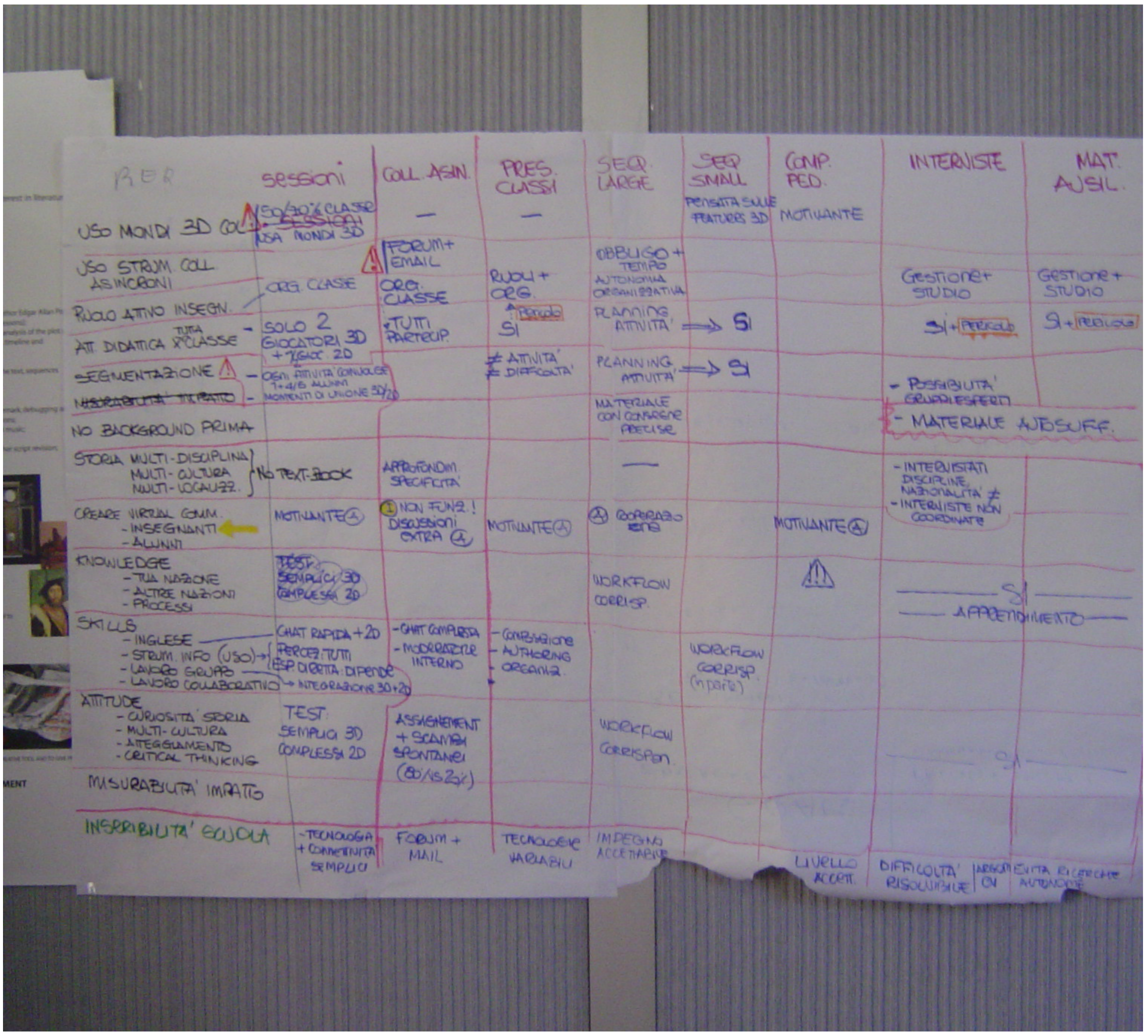


Figure 28. The RIM matrix drew on a wall-paper for the L@E project

4.2.5. Analysis

This phase consists in taking all the information surfaced by the different elicitation practices performed (interviews, focus groups, etc.) and in gathering all this knowledge in a structured and analytical picture. Pragmatically, the traceability analyst re-organise and re-order the RIM and DMM matrices developed in the previous phase. The analyst re-considers these matrices cross-by-cross, integrating notes and observations from the different elicitation sources. In particular, different points of view have to be integrated:

- *The designer's point of view*. Each designer develop different parts and different functionalities of a same application. His/her perception of the project is often limited to a "vertical" view on how these parts and functionalities answers to the strategic needs. The traceability analysis have to gather all these partial views, showing how the *entire application* fits with requirements through the interaction of its different parts.
- *The client/customer's point of view*. Often this point of view is mediated by the project manager. The focus here is how a single requirement has been taken into account in the application development. The analyst have therefore to consider all the information gatherer from an "horizontal" point of view, documenting the impact that all the strategic needs (expressed by goals and requirements) have on the application design.

	STATIC COMPONENTS				DYNAMIC COMPONENTS		TRANSVERSAL COMPONENTS	EDUCATIONAL MATERIALS		TESTING MATERIALS			
	D1 3D synchronous collaborative sessions	D2 Asynchronous collaboration	D3 Class presentations	D4 Games	D5 In-the-large sequence	D6 In-the-small sequence	D7 Educational competition in itself	D8 Interviews	D9 Auxiliary materials	D10 Quick questions on knowledge	D11 Open-ended comprehension questions	D12 Assignments & home-works	D13 Monitoring Tools & Procedures
GENERAL GOALS													
G1 Offering to schools a collaborative learning experience based on new technologies	3D world	Collaborative activities		Games involves collaboration	Workflow for knowledge	Workflow correspondance	Motivating						
G2 Basing the experience on historical contents								YES	YES	YES	YES		
G3 Basing the experience on a multicultural approach								Multiple points of view	About other cultures and histories				
G4 Allowing the educational impact to be measurable													Collect adequate information to measure impact
G5 Allowing to participate classes and pupils of every level and kind, not only the best classes in the								Self-sufficient material	Self-sufficient material	All based on given or simple materials			
G6 Minimising the internal management costs of the experience		Managed by teachers	Managed by teachers					Learning process managed by teachers	Learning process managed by teachers			Managed by teachers	
EDUCATIONAL GOALS													
B1 Knowledge					Yes: workflow for knowledge		Motivating	YES	YES	YES - motivating (factual notions)	YES - motivating (in-depth knowledge)	YES (attitudes)	
B1.1 About local (national) history													
B1.2 About other countries' history													
B1.3 About general historical concepts													
B2 Skills						Workflow correspondance							
B2.1 Use of "professional" English	Quick chat + 2D	Complex chat	Composition							Use of quick English		Yes	
B2.2 Use of technological tools	Perception: all; direct experience: it depends	Internal moderator	Authoring	Yes: use of 3D features								Yes	
B2.3 Group work	Integration of 3D and 2D		Organization	Fast decisions in group								Yes	
B2.4 Collaborative work											Quick group decision	Yes	
B3 Attitudes	Test: 3D simple, 2D complex	Assignment + spontaneous exchanges (80% vs. 20%)			YES: adequate workflow			YES	YES			Applying concepts and changing attitudes and learning	
B3.1 Sense of curiosity for history													
B3.2 Multiple cultures / multiple identities													
B3.3 Improved attitude towards history													
B3.4 Critical thinking towards knowledge													
B3.5 Different attitude towards knowledge													
VISIONS													
V1 Integration in schools' curricula	Technology and easy connectivity	Forum and email	Variable technologies		Acceptable effort demand		Acceptable level (not frustrating)	Acceptable difficulty level, and CV-related topic	Avoids the need to research for comprehension				Acceptable demand
V2 Characteristics of and educational competition				The most enjoyable part of competition	Scores ensure that nobody gets frustrated					Values students' preparation, is not frustrating	Values students' preparation, is not frustrating	Values students' preparation, is not frustrating	
REQUIREMENTS													
R01 The experience have to include the use of collaborative 3D worlds	50% / 70% of the class uses 3D	NO	NO	Motivating - Designed to use 3D features		Designed to use 3D features	Motivating						
R02 The experience have to include the use of tools for asynchronous collaboration		Forum and email			Sequence forces use (and time to use) Async. Tools								
R03 The experience have to include the teachers' active role	Roles + class organization	Roles + class organization	Roles + class organization		More autonomy to teachers			Roles + class organization	Roles + class organization				
R04 The educational activities have to involve the whole class	Only 2 players 3D + 1/2 2D	All participate	Yes - Danger	Yes - Danger	In-the-large sequence must maximize segmentation and involvement: planning of activities	YES: planning of activities maximizes involvement		Yes - danger	Yes - danger				
R05 The activities have to be modularised in order to facilitate class segmentation	Each activity involves 1 to 4/5 pupils; contact moments between 2D and 3D		Different activities, different difficulty	Yes - Danger		YES: planning of activities in view of segmentation		Possibility of expert groups;		Different activities, different difficulty	Different activities, different difficulty		
R06 The activities must require to students a minimum background knowledge					Precise deadlines for materials			Self-sufficient material		All based on given or simple materials			
R07 The activities must not presuppose that teachers know how to use technologies		Plain web technology (forum) and email	Any format can be used to present the class (power point, word, etc.)										
R08 The applications must allow to participate with a low technology level and include a degraded mode of use for low connections	Implementation technology and only 2 players per class												
R09 The historical contents have to highlight multiple opinions, disciplines, localisations and cultures			Specificity in deep		NO			Interviews to experts of diff.				Comparison of cultures and places;	
R10 The experience have to support the creation of a virtual communities of students, also after the end of the project	Motivating	Extra discussions	Motivating		YES: workflow for increasing cooperation		Motivating					Collaborative assignments	
R11 The experience have to support the creation of a virtual communities of teachers, also after the end of the project		It doesn't work!											

Figure 29. An example of RIM matrix from the L@E project

	Visions	Requirements	Designer expertise	Specific understanding of the domain	Constraints	Law obbligations
STATIC COMPONENTS	****************	****************	****************	****************	****************	****************
D1 3D synchronous collaborative sessions		Offering to schools a collaborative learning experience based on new technologies		In past projects this technology has been a good motivation for e-learning	Use of the in-house 3D technology	
D2 Asynchronous collaboration		Create a community	Forums and emails are common tools to build up an asynchronous collaboration			
D3 Class presentations			Just to add an activity in the first (introductory) session			
D4 Games				Another good idea from past projects as a motivation for traditional learning		
DYNAMIC COMPONENTS	****************	****************	****************	****************	****************	****************
D5 In-the-large sequence	Acceptable effort demand	Workflow adequate to educational goals				
D6 In-the-small sequence			Workflow correspondance. Each session has not to be too "boring" or long or simple/difficult, etc.			
TRANSVERSAL COMPONENTS	****************	****************	****************	****************	****************	****************
D7 Educational competition in itself	It has to be a motivation for students in learning: open and serious			In past projects it has been a good motivation for e-learning		
EDUCATIONAL MATERIALS	****************	****************	****************	****************	****************	****************
D8 Interviews			This way to present information has worked very well in past projects			
D9 Auxiliary materials			This material go with interviews as background explaination (maps, biographies, etc.)			
TESTING MATERIALS	****************	****************	****************	****************	****************	****************
D10 Quick questions on knowledge	Values students' preparation, is not frustrating					
D11 Open-ended comprehension questions						
D12 Assignments & home-works		Improve knowledge, skills and attitudes				
D13 Monitoring Tools & Procedures		The educational impact has to be measurable				

Figure 30. An example of DMM matrix from the L@E project

Both the points of view are essential to properly describe a project's traceability chain. These two points of views have a big impact on how a traceability expert may analyse the information gathered during the elicitation phase. In particular, they are mirrored in the two aspects taken into account by the TRAMA analysis:

- the justification or motivation of the design (designer's point of view), that can comes from requirements or from other kind of sources (an understand of the specific domain, the expertise of the designer, a constraint, etc.); these traces are called Design Motivations Model (DMM)
- the impact on design (client's point of view) of: visions, stakeholders-goals, users-motivations, domain issues, scenarios, constraints and requirements; these traces form the Requirements Impact Model (RIM)

The analyst considers these two matrices line by line and fills the crosses with the information gathered in the elicitation phase. The information written in each cross represent the trace "rationale" according to Dick [2002]: in other words, each cross explains and comments the relationship existing between the pair of the matrix elements that it links, highlighting particular and relevant aspects of the trace. The aspects that the traceability expert can highlight during this phase depend on the specific goals of the analysis that have been defined in the preliminary plan. These aspects may be seen as particular questions that the analyst have to answer to fit with the traceability goals and that guide him/her in shaping the RIM and DMM matrices. TRAMA is particularly conceived to help the traceability experts in facing with the following aspects[44]:

- *Client validation*: to set up a structured argumentation to show to the client that all the needs have been taken into consideration. In TRAMA the RIM matrix allows project managers and designers to map each goal and each requirements into design solutions, providing a powerful communication tool to show that everything (every strategic goal, etc.) has been considered in the application.
- *Design versioning*: to highlight different design areas for different stakeholders. If requirements are normalised with a proper goal-oriented methodology (e.g. with AWARE), each goal is linked to the stakeholder(s) who owns it; therefore, goals-to-design relationships in the RIM matrix allows to identify the application elements that satisfy the goals of a specific stakeholder.
- *Non-traceable design*: to document the motivations of design elements that do not derive from requirements. In the case studies conducted for this research, a big part of the design elements were not motivated by a requirement-related information: most of the choices came from usability or "good design" principles or were just due to the designer's expertise. The DMM matrix allows to distinguish the different motivations for design elements, relating design with its sources types and answering to the "why" question ("Why this design element has been placed into the application? Why in this way?").
- *"Negative" design*: to keep trace of old choices that have been eliminated or modified during the project. Rejected design choices can be (separately) listed in the DMM matrix; the crosses with the different sources types answer to the "why not" question.
- *Reverse requirements specification*: to check the consistency between design and requirements, "tune" requirements specification according to the real stakeholders' goals and extracting consistent requirements specification from design. This activity is supported by the RIM matrix that force analysts in surfacing consistency or inconsistency traces.
- *Usability on design documents*: to select the elements in the design involved for a specific task, evaluating the quality of the product with respect to the high-level goals and identifying test procedures that should be rerun to validate an implemented design change. The RIM matrix allows usability experts to perform inspections on specific design areas, properly considering the strategic goals that should be fit by those inspected elements.

TRAMA provides a set of pragmatic questions as well; these questions can be used by analysts as a guide or as a checklist to properly consider all the aspects involved in a relationship between project elements. For each cross in a matrix the traceability expert should ask himself:

- "Which design element fits with the needs of this stakeholder?"; "If I had to present the project to this stakeholder, which part of the design should I highlight?";
- "Which design element fits with this goal?"; "Which is the impact of this goal into the design?";
- "Which design element better fits with the needs of this user?"; "How can I arguing design choices to show that this user is considered in it?"
- "Which strategy is set-up in the design to fit with this user motivation?";
- "Which is the (positive or negative) impact of this constraint into the design?";

[44] These aspects have been already defined in paragraph 1.3.3 as an answer to the Goal 3.

- "Which are the design elements that fit with this requirement?"; "How can I show that this requirement has been properly taken into account in the design?";
- "Why the designer chose to put this element into the design?"; "How can I show that this element is not an extra-feature in the design?";
- "Why this element has been rejected or modified in the current design?"; "What is the impact of this choice into the project consistency with strategic goals?"

The analyst fills and completes this way all the matrices, finding all the strategies used in the design as solution to problems and needs highlighted in the requirements phase and surfacing other explicit or implicit, conscious or unconscious motivations of the designer for the design choices taken.

4.2.6. Specification

After the analysis has been properly conducted, the traceability expert have to present all the results in a structured document. This document is a real traceability specification, reporting which the stakeholders and the goals of the tracing activity were, which kind of activities have been actually performed, which information have been surfaced and which the consequences of these results may be. In other words, the role of a traceability documentation is resuming the elements surfaced during the analysis and organising it in a structured way; this kind of specification is able to summarise all the project components and the relationships between them, allowing a compact but complete understanding of the project status.
A typical traceability document is a compact (twenty pages) report, structured in 9 parts:

- An executive summary.
- A project summary, highlighting its goals, people involved, current status, etc.
- A traceability preliminary plan, summarising the goals, the stakeholders and the expected results of this activity.
- An information re-organisation and normalisation section, presenting how the project knowledge have been structure to allow a proper tracing activity; this section provides a general view of the current project status, a compact requirements specification and a compact design schema.
- A RIM matrix, with comments, highlighting the relationships between requirements and design.
- A DMM matrix, with comments, highlighting the relationships between design and its motivations.
- A summary of the results achieved by that analysis.
- A section highlighting the benefits that traceability brings or will bring to the project.
- A final conclusion, reporting the reasons why the tracing activity has been performed and the main consequences for the project, in terms of design areas to review, features to better implements, requirements to re-consider, etc.

4.2.7. Validation

Traceability validation is the activity to check the analysis results with requirements analysts, designers, project managers and clients. The specification is written for these people, so it must be written in a language which they can understand. Furthermore, the results should be written so that they may be verified. Validation works with a final draft of the traceability document, i.e. with negotiated and agreed information, after each meeting with project managers and designers. The validation phase is therefore a "transversal" activity that should be run and re-run continuously during elicitation and analysis, as well as after the specification has been written. Validation certifies that the traceability document is an acceptable description of the overall project, in terms of:

- Completeness and consistency of all the information reported.
- Conformance to standards adopted in the project and in the company (reports structure, responsibilities, etc.).
- Conflicts between traceability stakeholders' goals, e.g. between designers ("all our choices were strongly motivated") and clients ("some elements could be improved").
- Technical errors in the description of how the design is of what the requirements are, from a designers and requirements analysts point of view.
- Ambiguous information, expressed not clearly or using terms, schemas or other elements that in that particular project or in the company have other meaning.

As shown in Figure 31, the validation process may be seen as an elaboration that brings inputs and gives outputs. Inputs of this process are:

- the traceability document: it should be a complete version of the document, not an unfinished draft, formatted and organised according to organisational standards;
- organisational knowledge: knowledge, often implicit, of the organisation which may be used to judge the realism of the results;
- organisational standards: local standards e.g. for the organisation of the specification documents;

Outputs of this process are:

- list of problems: a list of discovered problems in the traceability document
- agreed actions: a list of agreed actions (that can be several or none) in response to problems discovered

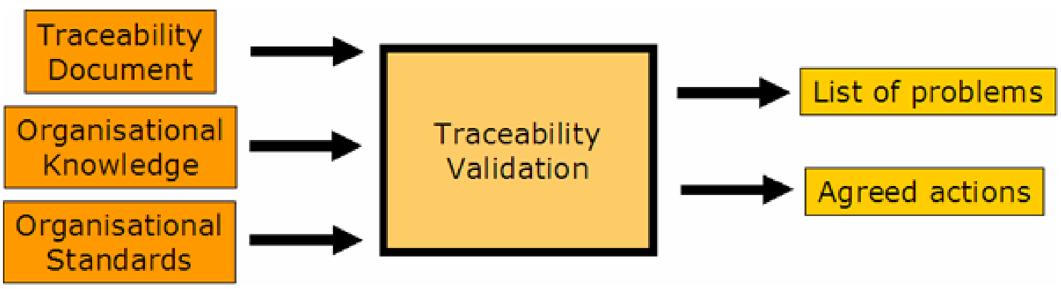


Figure 31. The validation process

4.3. The TRAMA tracing approach

A method can be described as a sequence of actions to be performed and of phases to be traversed in order to properly apply and accomplish the analysis method itself. A second way to describe TRAMA is to describe more in detail the global approach and the model taken into consideration by the method. In this section the method will be outlined in a more structured way, organising its elements in a full traceability model.

The concept of *tracing approach* refers to a generic term for methods, techniques and models enabling tracing activities [von Knethen & Paech, 2002]. A general *tracing approach*, i.e. a traceability method such as TRAMA, is characterised by (a) the purposes the activity may have, (b) the processes involved in the tracing activity, (c) the conceptual trace model which the activity is based on and (d) one or more tools enabling or facilitating or documenting such an activity [von Knethen & Paech, 2002]. A global picture of the approach is provided in Figure 32, that summarises the TRAMA approach showing purposes and processes supported, its conceptual trace model and tools used.

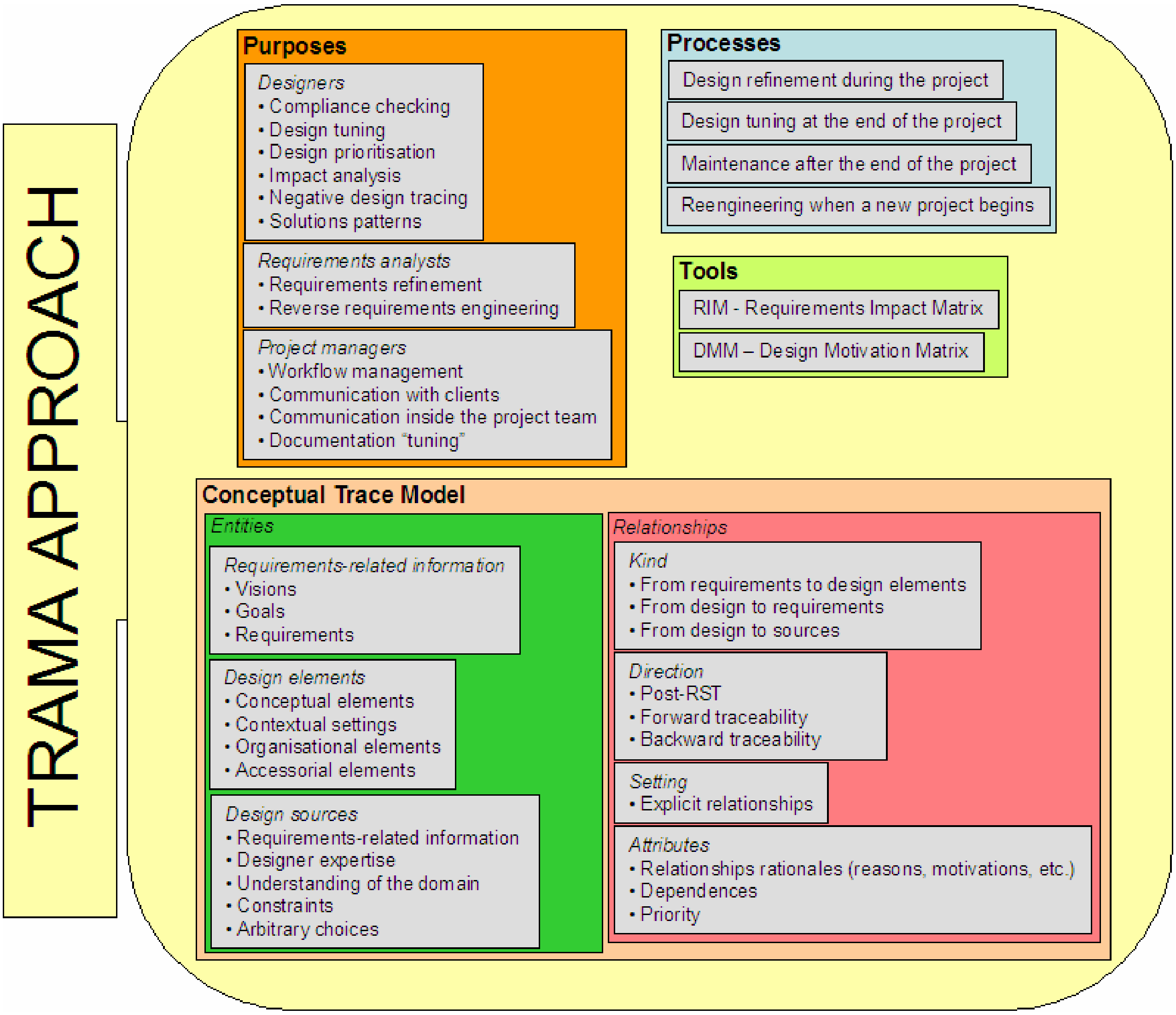


Figure 32. The TRAMA approach

In the next paragraphs, I will shape the main characteristic of TRAMA according to this tracing approach *meta-model*.

4.3.1. Purposes

Purposes of a tracing approach can be defined considering stakeholders, i.e. who will use the approach, and goals, i.e. traceability needs of these stakeholders. The TRAMA approach is mainly tailored for designers, but it may be profitably used also by requirements analysts and by project managers. The main possible purposes can be defined as follows:

- *Designers*
  - *Compliance checking*: designers may take advantage from TRAMA in order to check the compliance of design elements with requirements; with this method they can understand if a particular element of their design answers to one or more stakeholders' needs.
  - *Design "tuning"*: designers can base on compliance checking and on design motivations analysis provided by TRAMA to correct and refine or to reengineer the design according to strategic goals.
  - *Design prioritisation*: designers can evaluate the relative weight and the effort request for a design element according to requirements compliance, simplifying or enriching that element.
  - *Impact analysis*: designers can evaluate the impact of a requested change, analysing dependences between design elements and requirements, constraints, visions, etc. in the DMM matrix.
  - *"Negative" design tracing*: designers can keep trace of choices that for any reason have been rejected or eliminated from the application, avoiding to discuss again these solutions in future development of the project.
  - *Solutions patterns*: designers can keep a library of effective need-solution pairs as a structured sum of all the RMM matrices produced in several projects.
- *Requirements analysts*
  - *Requirements refinement*: requirements analysts may use TRAMA traceability information in order to refine the requirements specification. All the experiences conducted as case studies for TRAMA show that some requirements are sometimes let implicit or they are not recorded in a document, even if they are not obvious or trivial ones. The same experiences are good examples of how key requirements that have not been explicitly discussed in the analysis phase may surface in a TRAMA analysis considering design motivations.
  - *Reverse requirements engineering*: if requirements have not been documented in previous analysis phases, the requirements analyst can use the TRAMA information for a reverse engineering activity, understanding requirements from design and motivations.
- *Project managers*
  - *Workflow management*: project managers may take advantage from TRAMA for a better control of the overall project; the method provides a global picture of the entire project, highlighting the relationships between its different pieces and the reasons why those decision were formerly

taken; these elements are crucial to organize efficiently the time plan, giving priorities to the development of the core elements of the application and avoiding useless or superfluous features.

- *Communication with clients*: TRAMA is also a huge communication mean with clients, providing to project managers arguments and evidences of the project quality in terms of satisfaction of goals, needs and expectations.
- *Communication inside the project team*: TRAMA is a powerful communication tool for project managers and for designers that work on different elements of the application; while each designer develops a single application feature, a wider understanding of how requirements and educational goals are considered in the design is needed to refine and improve these solutions and to keep the *"fil rouge"* of the decisions taken during the project, understanding which elements cannot be modified and which ones may be altered in the revision process.
- *Documentation "tuning"*: the TRAMA analysis allows complete and refined documentation and specifications: the traceability chain provides a preferential way to order, link and organize each document or deliverable.

4.3.2. Processes

Processes define the kind of traceability activity (or activities) supported by the tracing approach and the context, i.e. the phase of the project's lifecycle in which traceability will be performed. TRAMA supports four different kind of activity in four different contexts: i) during the project for design refinement, ii) at the end of the project for design tuning, iii) after the end of the project for maintenance and iv) when a new project begins for reengineering.

i) Refinement

As discussed in the preface of this dissertation, a first design attempt is usually shaped by designer taking in account requirements as background information; in this phase, solutions are not conceived in a full explicit way and requirements are not considered one by one, at least in the cases analysed in section 3, even if the common sense (and years of experience) may tell us that this is how most of the industrial designers (and a part of the academic ones) work. This practice is not necessarily negative, since the results are often quite good in relation to stakeholders needs. TRAMA does not try to change this common way to work nor to record what happens during the design process *in se*: the method may be applied during the project after a first design has been produced, i.e. TRAMA tries to trace *ex-post* relationships surfacing motivations for design choices. From this point of view, the method may be applied in order to understand the explicit or implicit, conscious or unconscious reasons for solutions proposed in the design. These traces help designers in the refinement activity, i.e. in adjusting the first design attempt according to requirements and priorities in a explicit and structured way.

ii) Tuning

Even if the main design effort is done in the design phase, some adjustment must be always be performed during the entire project's life-cycle, because of technology limitations, business or resources constraints, new requirements or changes in requirements. TRAMA helps designer in this tuning activity, keeping traces of old design solutions and of reasons for changes; design to requirements relationships allow designers to understand the impact of a changed requirement into the application.

iii) Maintenance
After an application production project is ended, a continuous maintenance activity is needed to "keep alive" the application through the years. In particular, a real life use of the product by the final users and its effects on the organisation and on the business of the company, make clear if all the solution proposed were actually good and effective solutions. If this is not the case or if some changes occurs in the company (e.g. new constraints, new requirements, etc.), the application needs to be revised, updated or changed. TRAMA helps project managers and designers in adjusting weak solutions or in conceiving new solutions for old or new needs, understanding the impact of these changes on the overall application and on its compliance with requirements.

iv) Reengineering
Sometimes new needs, new requirements or, more in general, new relevant elements for the company bring to the decision to start up a new project in order to insert major modifications in the application or in order to develop a new application. Therefore in these cases, the old application may be simply tuned-up or completely reengineered. In both cases, TRAMA helps to understand or to remember why certain solutions have been formerly adopted even if some years have passed and if there is a new project team. TRAMA allows also to organise the redesign activity according to old dependencies with requirements, identifying the elements that can be improved and the "untouchable" elements linked to still valid goals and that should not be changed.

4.3.3. Conceptual trace model

A conceptual trace model, also called *reference model* [Ramesh & Jarke 2001], defines what "trace entities" and "traces" are and which traces should be captured. Therefore, a conceptual trace model determines what is relevant and it identifies and formalises which aspects of the system are to be recorded and worked with. Such a model should provide also guidelines to identify a common way of dealing with the traces. Two sub-concepts, (a) "entity" and (b) "relationship", refine the concept.

(a) Entities
Entities are the items or elements that the approach allows to trace [Spence & Probasco, 1998]. In the TRAMA method, traceable entities are defined as: (i) requirements-related information, (ii) design elements and (iii) design sources.

(i) Requirements-related information
As described in the case studies presented in section 3, TRAMA allows tracing not only requirements but also different kind of goals and visions. Although it is not strictly needed, the method suggest to "normalise" these information using a goal-oriented requirements engineering framework such as i*

[Yu, 1993] or KAOS [van Lamsweerde et al., 1998]: these techniques consider requirements as the result of a goal refinement process and facilitate the tracing activity. A general definition of how TRAMA considers visions, goals and requirements can be anyway done according to AWARE [Bolchini et al., 2005], an extension to the i* framework.

- *Visions* corresponds to a generic and project-independent opinion of stakeholders towards how to do or how to understand something; a vision can be defined as a strategic insight of a stakeholder in the domain, and it provides a way for modelling the assumptions of a stakeholder which dictate his/her "weltanschauung" on the project. In some cases visions are sources to refine goals and to define requirements, in other cases they have a direct impact on the design.
- *Goals* are considered as a wished state of affairs for the main stakeholders, but also a wished experience or an expectation for a class of users. They can be of different type (e.g. general goals, research goals, business goals, educational goals, etc.) and of different granularity (goals and sub-goals in a refinement process).
- *Requirements* are sufficiently high-level descriptions of properties or functionalities of the application as input for the design activity; they are consequences of visions or the result of goals refinement and they can be seen as indications for the subsequent design.

(ii) *Design elements*

In current methodologies and practices, the impact of strategic goals and requirements is analysed considering as target technical design features or application elements, in terms of pieces of contents and functionalities, UML-classes, pieces of code or objects. As introduced in the L@E case[45], it is sometimes impossible to understand the impact of requirements-related elements into the application considering only the software elements of the system. This condition may be always true, but it surely occurs in cases like information systems or, more again, like educational applications. If it is impossible to understand the project solutions without considering contextual information, a wider definition of "design" is needed that considers all the essential aspects to trace the consequences of high-level goals on the application. This definition includes in the "design" organisational elements, activities, roles, workflows etc. together with the technical-applicative aspect. This concept, that may appear rather unusual in the RE community, is at the contrary widely accepted in the e-learning community, where hypermedia design of e-learning applications and instructional design, e.g. the design of how the educational activity should be organised, are often integrated in so-called "educational environments"[46]. From this point of view, the difference between a low-level requirement and a design element can be considered rather subjective; TRAMA allows to use these definitions in an "utilitarian" way, defining the difference between requirements and design according to the specific project needs.

> Design elements may be defined as all the relevant application choices in the project that are someway influenced by the strategic high-level goals or that are able to satisfy stakeholders needs and that include the technical-applicative aspect *and* the format, the procedures, the workflow, the activities of users, etc.

[45] See also Paragraph 3.6.
[46] See also Botturi & Belfer [2003].

For instance, in the L@E case, five design categories were taken into account:
- static components, i.e. the "bricks" the experience was composed by;
- dynamic components, i.e. how static components were assembled in a workflow;
- transversal components being both static and dynamic or no one of the two;
- educational materials, i.e. contents of the educational experience and
- testing materials, i.e. all the elements used to measure the educational impact of the experience.

TRAMA calls this kind of design, "experience design" because the overall user experience is considered in all its aspects. More analytically, an experience design may be composed by the following elements:
- *Conceptual elements*, i.e. the traditional conceptual design elements: content and structure of content, navigation architecture, access paths, operations, pages and layout.
- *Contextual settings*, e.g. the technical equipment, the place where the application is used, the physical disposition of machines in this place, etc.;
- *Organisational elements*, e.g. in L@E, how different use sessions are organised during a week, which activities are implies in the use of the application, etc.;
- *Other accessorial elements*, e.g. study material needed to use the application (in a educational system), etc.

As described before, from this point of view the difference between such a design element and a very low-level requirement could be seen as rather undefined or unclear. In certain phases of the project, some arbitrary choices are pragmatically turned into requirements why one decides that these choices are no more under discussion, they become "untouchables" in future steps of the project; sometimes these choices are solutions successfully experimented in previous projects. The border between a requirement and an arbitrary choice is sometimes fuzzy: in general, a requirement is perceived as a static and unchangeable point, a given element of the problem. Therefore, there is a certain fluidity between requirements and design choices. It is anyway important that at a certain point of the project the difference between static and changeable choices becomes clear and explicitly defined.

(iii) Design sources

The word "sources" indicates here the reasons because a specific design solution has been adopted. Design sources may of course be from one hand requirements-related information, i.e. visions, goals and requirements, as it has been discussed below. However, some experiences in the field [Bolchini et al., 2005; Randazzo 2005a] show that requirements are not the only motivation for a design element; in the referenced cases, more than 50% of design has not requirements related motivations but it is borrowed from designer expertise or from a specific understanding of the domain. From this point of view, design sources may be between the following:
- *the designer expertise*, i.e. particular "good design" principles that are part of the designer's skills and that she/he applies in any case;
- *a specific understanding of the domain*, i.e. recurrent good solutions in a domain that the designer applies because she/he learnt it by other cases in the same domain;

- *a particular constraint*, e.g. budget limitations, time, technology limitations, etc.;
- *a law obligation*, e.g. copyright issues, personal data treatment, etc.
- *a requirements-related information*, i.e. a vision, a goal, a requirements, etc.
- *an arbitrary choice*, i.e. a choice without particular reasons, usually a single detail that should anyway be set in a way or another, e.g. in L@E the structure of a game in three steps (instead of four or two).

(b) Relationships

Relationships investigate different tracing approaches concerning the traces that are suggested to be captured/maintained. According to von Knethen & Paech [2002], this concept consider the four aspect of: kind, direction, setting and attributes.

- *Kind* – TRAMA supports three kind of relationships to be traced:
  - from requirements to design elements, tracing the impact of requirements on the design;
  - from design to requirements, tracing the justification of design solutions;
  - from design to its sources, tracing the motivations for design choices.
- *Direction* – TRAMA is a post-Requirements Specification Traceability method and supports forward traceability from requirements to design elements and backward traceability from design elements to requirements or to other motivations.
- *Setting* – Due to the its own nature, TRAMA allows only explicit relationships; the method does not include any implicit or automatic generation of traces.
- *Attributes* – According to the rich traceability approach [Dick, 2002] TRAMA traces does not mean anything *per se* but they have to be completed and commented through notes highlighting the rationale of the relationship. E.g. a trace between a design element and a requirement may have as attribute the reason why such an element has been conceived to fit with the requirement and how that element answers to the needs expressed by the same requirement. In addition, TRAMA traces may have as attribute a dependence with another trace and a priority value; e.g. a relationships may highlight if its source can be considered as the main and major elements that fits with the trace's target.

As shown in Figure 33, TRAMA relationships can be surjective or not but never injective. In mathematics, injections, surjections and bijections are classes of functions distinguished by the manner in which arguments (input expressions) and images (output expressions) are related. A function is *injective (one-to-one)* if each image is mapped to by at most one element of the domain. A function is *surjective (onto)* if every element of the co-domain is mapped to by some element of the domain. A function is *bijective (one-to-one and onto)* if and only if it is both injective and surjective. Equivalently, every element of the co-domain is mapped to by exactly one element of the domain. A one-to-one *function* is injective, but may fail to be surjective, while a one-to-one *correspondence* is both injective and surjective. In TRAMA, requirements-to-design relationships are not injective (more requirements may have impact on the same design elements) and not surjective (some design elements may not derive from requirements). In the same way, design-to-motivations relationships are not injective (more design elements can have the same kind of source) and not surjective (in a project only a part of possible kind of sources may be applicable as motivation for design choices). On the other hand, the inverse motivations-to-design relationship is a surjective function because there are not design elements without a source (since one of the kind of sources is "arbitrary choices").

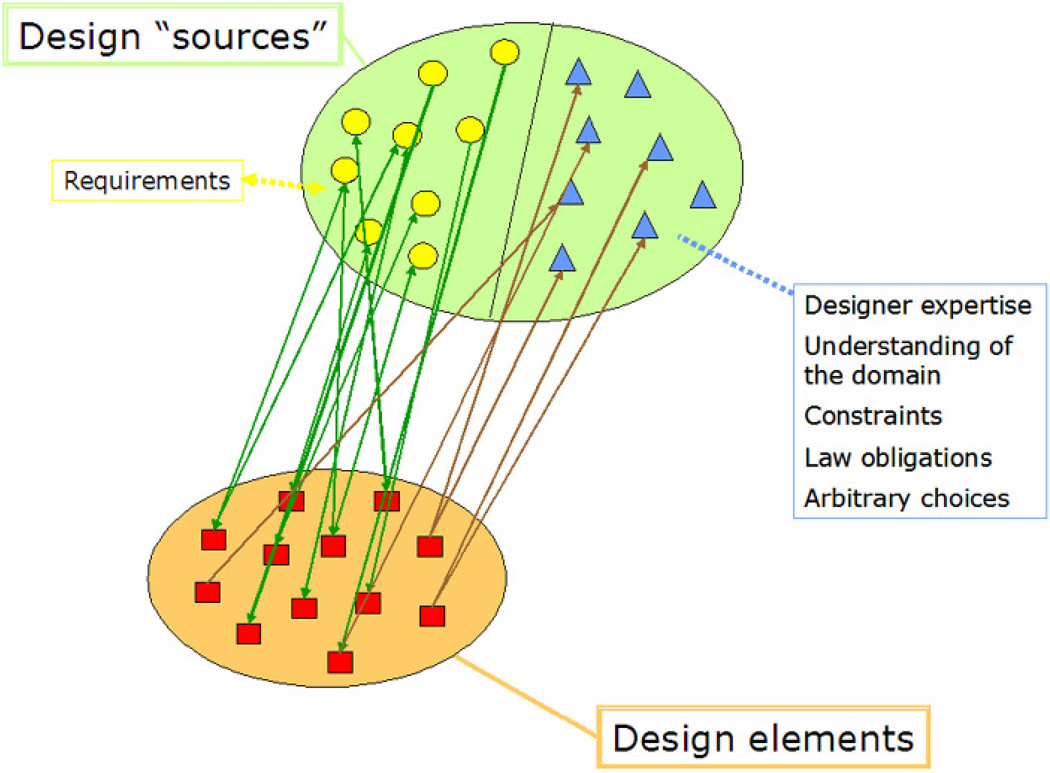


Figure 33. The TRAMA relationships between design and its sources

4.3.4. Tools

As it has been described in section 2, current methodologies have two main open problems from the tools point of view:

- while it is difficult maintaining the huge mass of dependences among the many objects produced by a large software system development effort, some current approaches require the use of a software tool to become usable and manageable; current tools have problems in maintaining relationships concerning artefacts expressed in natural language, often ambiguous, or artefact created independently by non-interoperable tools and that evolves autonomously;
- current methodologies focus on traceability of singles objects and on the use of conceptual tools that emphasise a "punctiform" view, but they do not consider a more global picture; traceability of single objects is not very important but in some minor cases, which can be considered very particular and specifics. Since a TRAMA analysis produces a complex picture under control, a method emphasising this global picture as a whole is needed.

For these reasons, TRAMA does not require the use of a software tool to be applied. The production of a software to support the TRAMA tracing activity could anyway be a future development of this research[47]. Furthermore, the managing of the traceability chain and of the "global picture" is assured by the use of matrices as conceptual tools. A second advantage of the matrix representation is that it is easy to understand and it provides a format that can be discussed by stakeholders with different backgrounds.

TRAMA provides two main tools that allows to discuss, analyse, access and present the traced information. These two conceptual tools are based on the use of matrices. According to Wieringa [1995] a matrix is a simple way to represent links between items, in which the horizontal and vertical dimension list the items that can be linked, and the entries in the matrix represent links between these items and traces attributes. The TRAMA method includes two representation tools: (i) RIM and (ii) DMM. The analysis approach for both models consist in one or more matrices representing traces between two families of objects (e.g. requirements and design topics). Each matrix can be used as a kind of checklist supporting the traceability analyst in considering the relevance and the mean of each possible pair of objects.

(i) RIM, Requirements Impact Matrix/Model[48]

This matrix list vertically requirements-related information and horizontally all the design elements. Table 3 shows a possible template for a RIM matrix and Figure 34 provides an example of how crosses may be filled. The model highlights the impact on design of visions, stakeholders goals, users motivations and requirements. The horizontal dimension of the matrix provides information on how single requirements are taken into account into the design; the vertical dimension shows how a single design element satisfy the project requirements.

		DESIGN ELEMENTS			
		Content 1	Content 2	Access path 1	…
REUQIREMENTS-RELATED INFORMATION	**VISIONS**				
	Vision 1				
	Vision 2				
	…				
	GOALS				
	Goal 1				
	Goal 2				
	…				
	REQUIREMENTS				
	Requirement 1				
	Requirement 2				

Table 3. A template for the RIM matrix

[47] See also section 6

[48] While RIM and DMM represent a conceptual trace model for a specific application, names refer to it both as matrices or models.

Therefore, this matrix can be filled in two ways:

- Vertically, according to the designer point of view, design elements by design elements, analysing which of the requirements listed fits to a design element; in fact, when several designers work on the same application, each designer develops just a single part of the entire system. The question that designers with the help of the traceability expert try to answer is: "Taking into account a single design element, how does it fit with requirements?".
- Horizontally, according to the client point of view, requirement by requirement, analysing which design element has impact on a requirement and how a need is satisfied by the application; the real impact of requirements on the design has been therefore analysed.

TRAMA let the analysts free to consider first the horizontal and then the vertical dimension of the RIM matrix or the inverse or both dimensions in a mixed way. The method suggest anyway to keep in mind and to analyse both points of view, i.e. how single requirements are taken into account into the design, and how a single design element satisfies the project requirements.

	DYNAMIC COMPONENTS	
	D5 In-the-large sequence	D6 In-the-small sequence
REQUIREMENTS		
R01 The experience have to include the use of collaborative 3D worlds		Designed to use 3D features
R02 The experience have to include the use of tools for asynchronous collaboration	Sequence forces use (and time to use) Async. Tools	
R03 The experience have to include the teachers' active role	More autonomy to teachers	
R04 The educational activities have to involve the whole class	In-the-large sequence must maximize segmentation and involvement: planning of activities	YES: planning of activities maximizes involvement
R05 The activities have to be modularised in order to facilitate class segmentation		YES: planning of activities in view of segmentation

Figure 34. A detail of a RIM cross in the L@E project

(ii) DMM, Design Motivations Model/Matrix

This matrix list vertically the design elements of the application and horizontally their kind of motivations, i.e. design sources. Table 4 shows a possible DMM matrix template and Figure 35 provides an example of how crosses may be filled. Traces between design elements and their motivations are not just the opposite of requirements-design relationships: in fact, the model highlight the justification of the design, that may be motivated by specific requirements or goals, by visions, by an understanding of the specific domain, by the expertise of the designer, by constraints or by arbitrary choices. The matrix can be filled only horizontally, trying to answer the "why" question design element by design element. "Negative" design elements can be also listed in this matrix; in this case, relationships rationale and comments inside crossed reports the "why not" answer, i.e. why the negative element were rejected or eliminated.

		DESIGN MOTIVATIONS						
		Visions	Goals / Requirements	Designer expertise	Understan-ding of the domain	Constraints	Law obligations	Arbitrary choices
DESING ELEMENTS	Content 1							
	Content 2							
	Access path 1							
	…							
	Negative design element 1							
	Negative design element 2							

Table 4. A template for the DMM matrix

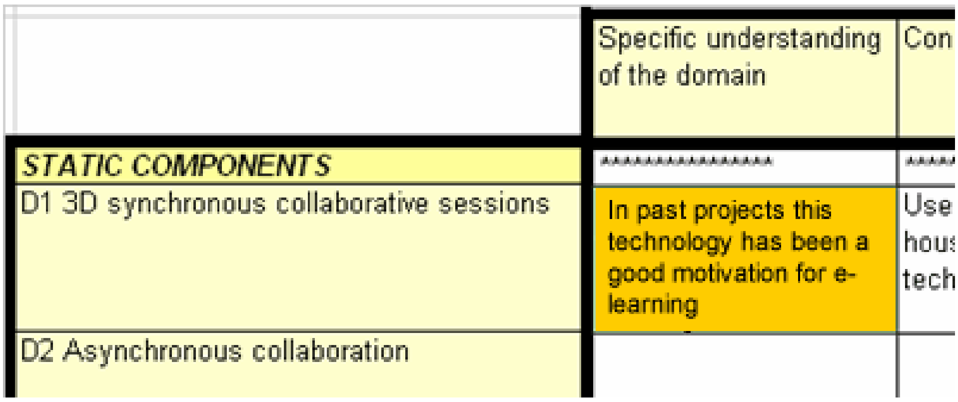


Figure 35. A detail of a DMM cross in the L@E project

4.4. A tracing discipline?

As described in paragraph 4.3.1, TRAMA can be profitably used by designers (as main users class) but also by project managers and requirements analysts. All these stakeholders can take advantage from the information raised out from a TRAMA analysis; however, in most cases they are not the real direct users of the method: a traceability expert is needed.

As stated in section 2, current practices consider traceability as a part of the requirements analysis process. The TEC-Lab's[49] and HOC's[50] experience and research in the field seems to show that traceability can be considered as a self-standing activity instead. In fact, if requirements are the strategy to satisfy stakeholders' goals and the design is how the application have to behave, tracing can be see as the activity of arguing why design solutions satisfy requirements. Therefore, the traceability expert is not a requirement analyst or a designer but he/she needs specific competences and skills; besides, due to

[49] TEC-Lab is the Technology Enhanced Communication Laboratory at the University of Lugano (Switzerland).
[50] HOC is the Hypermedia Open Centre at the Politecnico di Milano (Italy).

psychological issues, an analyst or a designer cannot easily perform the self-observation activities included in the tracing process. Furthermore, in the case studies conducted for this research, traceability appeared as a self-standing activity due to organisational problems (e.g. resources allocation).
All these reasons brought project managers of the cases considered for TRAMA in considering the traceability analyst as a different role than designer or requirement analyst. A traceability expert/analyst is in fact a facilitator in project meetings, with specific competences in eliciting and understanding why a certain decision is being or has been taken.
This fact suggest a strong hypothesis, that could be the topic of a future research: is traceability a self-standing discipline? As well as requirements analysis has been considered for a long time a part of the design process, traceability is now considered as part of a requirements management phase. The hypothesis that I propose as a suggestion is that traceability can have the dignity of a discipline in the wider engineering field, a discipline distinct and separate from RE.

4.5.Benefits

The main benefits brought by the use of the TRAMA method can be taken from the experiences and the projects where it was already applied. In particular, TRAMA brings to the projects benefits and advantages in terms of: (i) communication with clients, (ii) internal communication, (iii) project documentation, (iv) design and requirements consistency, (v) reverse requirements engineering, (vi) design tuning and revision and (vii) system re-engineering.

(i) Communication with clients
TRAMA is a powerful communication mean to show to the clients that all their requirements have been considered and how, and that there are not unmotivated elements in the design. The traceability activity helps in make clearer the relationships between the different application parts and their strategic role, providing a useful tool to express the main benefits of the project. This analysis helps in better identifying a proper justification for all the elements in the design as a preventive activity allowing the final application to be accepted for publication.

(ii) Internal communication
Traceability documents can be used by project managers to communicate the project status to all the team members, i.e. to designers and engineers that implemented the system and that have just a partial understanding of the project, limited to what they did and developed. The traceability specification may be therefore used as a consistent and up-to-date document reporting requirements, design and interdependencies between the two, i.e. to communicate the project status to the overall work-team.

(iii) Project documentation
More in general, TRAMA is a powerful tool for a project knowledge re-organisation; the knowledge "normalisation" for requirements and design provides a standard and structured set of concepts that can easily related each other; the models used in the "normalisation" allow expressing a big set of concepts in a few elements; requirements and design specification are re-produced in a consistent form enabling

reasoning and confrontation between them. Discussions about requirements and design with the project responsible allows also to update these documents consistently with the actual state of the experience.

(iv) Design and requirements consistency

TRAMA is a structured practice to check design consistency for revision, surfacing missing design elements and missing requirements. The different kind of traces provided by the method allow to highlight requirements that are not considered in the design, useless features in the application or missing knowledge (goals, visions) that motivates existing elements in the system.

(v) Reverse requirements engineering

TRAMA supports reverse requirements engineering. In some projects requirements are never clearly stated; a lot of documentation describing the project and its benefits is always produced, but it is sometimes not very clear e.g. what strategic goals and what visions are. Therefore, each team member has her/his own opinion about what it is "untouchable" and what it is changeable in the application. Applying TRAMA in these projects, forces to a more structured vision of this knowledge, re-organising and refining requirements and surfacing missing information, fundamentals to understand the project but never explicitly documented.

(vi) Design tuning and revision

TRAMA is an advanced tool to tune up and re-align design in maintenance phase and to assign priorities to design elements. In some projects design is described in discursive documents but never represented as structured components. TRAMA forces designers to distinguish between details and base elements of the application; this practice allows surface missing design components and re-align the design with the project state-of-the-art. In other projects TRAMA can be applied to improve a first version of the design. In fact this method helps designers in identify mandatory design elements, i.e. those elements that are related to a main goal or requirement, and in understand which parts can be changed instead. The weak elements can be identify and improved as well. TRAMA allows to identify problems and inconsistencies included in the design; some goals may be not properly supported by the application and some motivations may not be completely clear. The analysis highlights also some aspects that are overemphasized respect to the high-level goals, and some other aspects that are relegated in a secondary plan and that should be considered differently.

(vii) System re-engineering

TRAMA specifications provide a complete project knowledge summary of requirements-related information, of design elements and of relationships between them, as vital information allowing an effective system reengineering, workflow organisation and more focused verification procedures to be performed. The understanding of which elements were designed to answer to a specific need or according to a specific vision and of which ones were designed for other reasons helps in identifying those elements that could be slightly modified with no effect on the overall application quality in terms of requirements coverage. The TRAMA analysis highlights also the application weak points that help in formulating some suggestions for further improvements.

4.6. Limits

As far as TRAMA has been applied in the case studies presented in section 3, problems related to the method have been highlighted and a solution to these problems have been conceived in order to improve the method itself. The current version of TRAMA, described in this section, includes all the improvements developed in this empirical way. Anyway, TRAMA still includes some limits, which could be the focus of a new step in this research. As a kind of self-criticism, I have identified two major limits in this approach: (i) a lack of a massive set of cases in which TRAMA has been applied and (ii) some organisational problems in maintaining the RIM and DMM matrices up-to-date.

(i) Experimentations
TRAMA has been currently applied in six different projects included as case studies in this dissertation[51]. Taking into account time and resources needed for the analysis of large projects such as those considered in this research, it was not possible to face with more cases. These experiences have been useful to develop a stable and usable version of the method. Anyway, a massive experimentation phase is now needed to check and validate TRAMA in industrial cases. From an academic point of view, TRAMA is being taught in some courses (both of bachelor and master level) at University of Lugano and at Politecnico di Milano; students involved in these courses are almost 1000.

(ii) Maintenance
When major or minor modifications are applied to a project, the traceability specifications have to be updated accordingly. The problems is that the project manager have to establish who is responsible to keep the TRAMA matrices up-to-date, still keeping the global picture under control. This problems surfaced in the L@E project, when new modifications have been introduced after the last experimentation phase. In that project, the problem has been solved in a creative way:

- Each designer has been coupled with a requirement: the designer is responsible to monitor the development and the status of its requirement. This role is called *requirement watcher*.
- Anyone who intervenes in any way on one or more design elements, have to check on the RIM matrix which requirement(s) is related to those elements. Then, he/she has to communicate to the specific requirement watcher what modification has been performed and why.
- The requirement watcher involved is responsible to update the traceability documentation and, in case, the requirements specification.

[51] See also Section 3.

5.Teaching TRAMA

<<Between thought and expression / there lies a lifetime.>>
Bob Dylan

Abstract
This section presents the modules, the activities and the courses conceived to teach the TRAMA method to different targets and in different situations. In particular, three modules and four courses are provided.

The modules composing the teaching framework are:
- Module 1: Introduction to Tracing;
- Module 2: The TRAMA method;
- Module 3: TRAMA in practice.

These modules, combined in different ways, allow to create academic and industrial courses:
- TRAMA: traceability for interactive application (full-size course)
- TRAMA take-away (1 hour course)
- TRAMA crash course (1 day course)
- TRAMA: requirements traceability in practice (3 days course)

5.1. Instructional Design method

In order to effectively shape, formalize and communicate the instructional design of TRAMA courses, a very simplified version of E^2ML method [Botturi, 2004] has been used. E^2ML – Educational Environment Modelling Language – is a visual modelling language for the design of educational environments in Higher Education. It is useful for representing the product of the design process: the educational activity or activities performed into an educational environment.

The TRAMA teaching approach is divided in learning modules that allow teachers to compose ad-hoc courses depending on time resources, target and situation of use. Each module is presented using the same conceptual structure. In particular, the structure used includes: (i) a goals representation, (ii) an activity flow diagram and (iii) a resources list.

(i) Goals Representation

First of all, the goals of each module have been illustrated in a very detailed manner using an adapted version of the table provided by E^2ML method.

GOAL STATEMENT						
TAG	TEACHING STRATEGY ID	STATEMENT	TARGET	APPROACH	ASSESSMENT	IMPORTANCE
<Goal Tag>	<ID of learning approach>	<Statement of the learning goal>	<Learners target of the learning strategy>	<Learning approach>	<Strategy for assessing the goal's achievement>	<Goal's relative importance>

Table 5. The detailed table used for presenting the learning goals

(ii) Activity flow

By each detailed table the activity flow for teaching and learning the module is presented. The activity flow diagram is the chronological way for teaching the module. It is important to underline that a module could have many flows (it is possible to have different chronological paths in order to teach the module). The diagram also shows the strategies for assessing the goals' achievement.

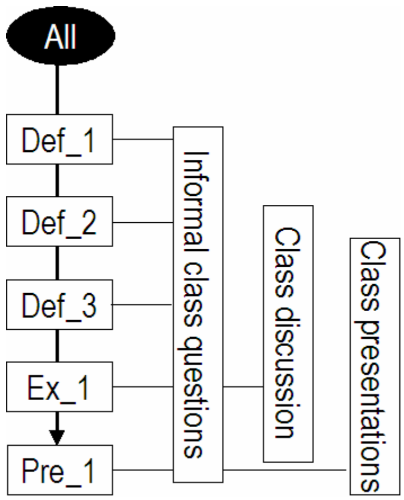


Figure 36. Example of activity flow diagram

(iii) Resources list

For each module the list of the learning resources has been provided.

RESOURCES			
NAME	DESCRIPTION	TYPE	USE LOCATION
<Resource’s name>	<Description of the resource>	<Type of resource>	<Where to use the resource>

Table 6. The detailed table used for presenting the learning resources

5.2. Modules to teach TRAMA

In this section I will present the learning modules related with the TRAMA teaching plan. Each module is independent and self-standing: a single module may be considered as a mini-course on its specific aspect of the TRAMA framework. However, the whole modules represent a meta-architecture that allows creating a complete course taking into account the type of learners and the time at disposal (examples of these courses are provided on section 5.3). The modules composing the teaching framework are three:

- Module 1: Introduction on Traceability;
- Module 2: The TRAMA method;
- Module 3: TRAMA in practice.

5.2.1. Module 1: Introduction on Traceability

TRAMA module 1 is an introduction that wants to create the background to better understand the method. Traceability is not a known practice and, on the other hand, the term is used in other fields (e.g. alimentation) to define other kind of activities that could cause ambiguity problems or misunderstandings. For this reason, the module focus on defining traceability in the Requirements Engineering field, highlighting its possible uses and purposes. To accomplish these objectives, the main learning topics are:

- Defining traceability (G1_M1, Goal1_Module1);
- Focusing on Requirements Traceability (G2_M1);
- Showing possible traceability approaches: purposes, processes, models and tools (G3_M1);
- Introducing examples of traceability practices (G4_M1);
- Providing a short review of traceability software tools (G5_M1).

(i) Goals table

GOAL STATEMENT						
TAG	TEACHING STRATEGY ID	STATEMENT	TARGET	APPROACH	ASSESSMENT	IMP.
G1_M1	Def_1	Define the concept of traceability in software development projects	All	Definitions	Informal class questions	2
G2_M1	Def_2	Distinct the specific characteristics of Requirements Traceability	All	Definitions	Informal class questions	3
G3_M1	Def_3	Be aware of possible uses and benefits of RT practices	All	Definitions	Informal class questions	5
G4_M1	Ex_1	Develop a positive attitude towards RT	All	Examples of TR practices	Informal class questions + Class discussion	4
G5_M1	Pre_1	Develop a critical approach towards software-based solutions	All	Presentations of TR tools	Class presentations + Informal class questions	2

Table 7. Detailed explanation of Module 1 goals

(ii) Activity flow

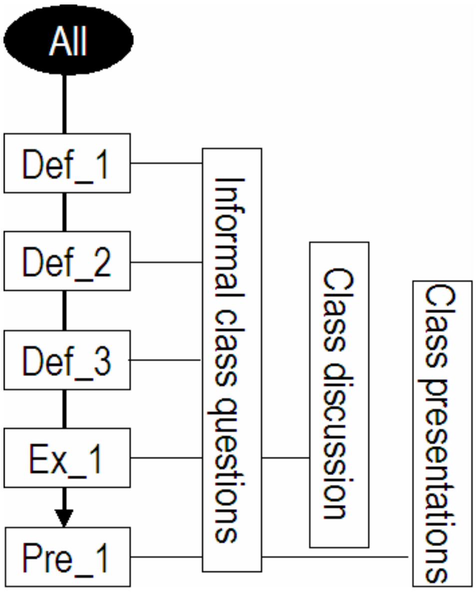


Figure 37. The activity flow of Module 1

(iii) Resources list

RESOURCES			
NAME	DESCRIPTION	TYPE	USE LOCATION
Slides pack 1	A set of slides presenting the definitions of TR and of RT and introducing possible TR approaches	Content	Classroom + Home
Examples of TR practices	Several examples of concrete practices and uses of TR in industrial projects	Concrete examples	Classroom
Review of TR software tools	Several examples of software tools enabling TR	Concrete examples	Classroom
Readings on traceability	• Gotel, O. & Finkelstein, A. (1994). An analysis of the Requirements Traceability Problem. • Hull, E. (2003). Requirements Engineering. Chap.7 & 9.	Content	Home

Table 8. Module 1 list of resources

5.2.2. Module 2: The TRAMA method

TRAMA module 2 focus on teaching how the method can be applied, highlighting its main concepts and tools. The TRAMA activities workflow is then studied in detail, with a particular use of examples for each step of the TRAMA analysis practice. To accomplish these objectives, the main learning topics are:

- Introducing TRAMA basic features and concepts (G1_M2);
- Presenting TRAMA activities workflow (G2_M2);
- Detailing activity 1: preliminary plan (G3_M2);
- Detailing activity 2: information re-organisation and normalisation (G4_M2);
- Detailing activity 3: elicitation and analysis (G5_M2);
- Detailing activity 4: specification and validation (G6_M2);
- Introducing examples where TRAMA has been applied (G7_M1);
- Wrap-up the main concepts (G8_M1).

(i) Goals table

GOAL STATEMENT						
TAG	TEACHING STRATEGY ID	STATEMENT	TARGET	APPROACH	ASSESSMENT	IMP.
G1_M2	Def_4	Define the basic TRAMA concepts and introducing the main features	All	Definitions	Informal class questions	5
G2_M2	Def_5	Understand the global workflow picture, identifying the different steps and their relationships	All	Definitions	Informal class questions	5
G3_M2	Dex_1	Understand how to develop a TRAMA preliminary plan	All	Definitions + Examples	Informal class questions	3
G4_M2	Dex_2	Understand how to develop the TRAMA information re-organisation and normalisation activity	All	Definitions + Examples	Informal class questions	4
G5_M2	Dex_3	Understand how to develop the TRAMA Elicitation and analysis activity	All	Definitions + Examples	Informal class questions	5
G6_M2	Dex_4	Understand how to document and validate TRAMA results	All	Definitions + Examples	Informal class questions	4
G7_M2	Ex_2	Develop a positive approach towards TRAMA, understanding its benefits through examples.	All	Examples of TRAMA analysis	Informal class questions	4
G8_M2	Ass_1	Assess the TRAMA approach	All	Recapitulation	Informal class questions and/or Written exam	3

Table 9. Detailed explanation of Module 2 goals

(ii) Activity flow

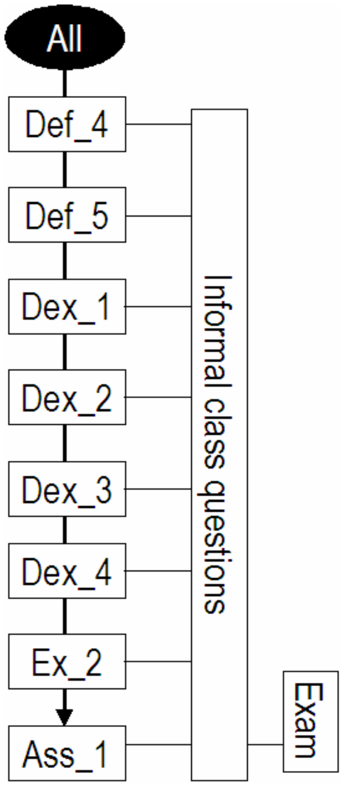


Figure 38. The activity flow of Module 2

(iii) Resources list

RESOURCES			
NAME	DESCRIPTION	TYPE	USE LOC.
Slides pack 2	A set of slides presenting the definitions of TRAMA basic concepts and of the TRAMA activities workflow	Content	Classroom + Home
Slides pack 3	A set of slides presenting in detail each step of the TRAMA activities workflow with examples from concrete projects	Content + Examples	Classroom + Home
TRAMA ex.	Several examples of concrete TRAMA analysis cases	Ex.	Classroom
Exercise 1	An exercise related to the RIM and DMM analysis	Exercise	Home
Readings on requirements, design and traceability	• Bolchini, D., Randazzo, G., Paolini, P. (2005). Vision-Driven Requirements Analysis for Communication Design. • Bolchini, D., Piccinotti, N,, Randazzo, G., Gobbetti, D. (2005). IDM - A User-Centred Model Shaping User Interaction as a Dialogue. • Randazzo, G. (2005). Museum of non-European cultures: a Design Traceability Case Study adopting the TRAMA approach for Int. Appl.	Content	Classroom + Home

Table 10. Module 1 list of resources

5.2.3. Module 3: TRAMA in practice

After the understand of the main concepts and the workflow related to the TRMA method, Module 3 is dedicated to its practice. The learners experiment gradually traceability activities, problems, communication-related issues, etc. To accomplish these objectives, the main learning topics are:

- Analysing and documenting traceability: practical tips (G1_M3);
- Individual training (G2_M3);
- Group training (G3_M3).

(i) Goals table

GOAL STATEMENT						
TAG	TEACHING STRATEGY ID	STATEMENT	TARGE T	APPROACH	ASSESSMENT	IMP.
G1_M3	Gui_1	Be aware of how TRAMA can be applied	All	Definitions + Guidelines + Ex.	Informal class questions	4
G2_M3	Pra_1	To be trained on TRAMA method + Check the understand on the approach	Single student	Individual case study (simple didactical projects)	Individual questions + Tutoring	4
G3_M3	Pra_2	To be trained on a complex TRAMA analysis + Check the understand on the approach	All (by group)	Group case study (verisimilar work-team situations)	Group questions + Tutoring + Analysis Presentation (written & oral)	5

Table 11. Detailed explanation of Module3 goals

(ii) Activity flow

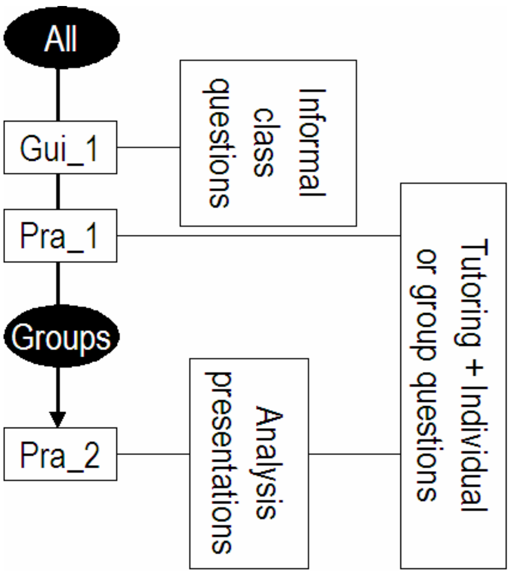


Figure 39. The activity flow of Module 2

(iii) Resources list

RESOURCES			
NAME	DESCRIPTION	TYPE	USE LOCATION
Slides pack 4	A set of slides presenting the TRAMA analysis tips and guidelines	Content	Classroom + Home
Examples of TR reports	Examples of good traceability reports	Content + Examples	Classroom + Home
List of projects	A list and a brief description of projects for individual and groups case studies	Exercise	Home
TRAMA reports	[Reports] L@E, Pompei, Munch in Berlin	Content	Classroom + Home

Table 12. Module 2 list of resources

5.3. Courses

The modules described in paragraph 5.2 can be combined and used in different ways, depending on time, target and purposes of the course. For TRAMA, four different kind of courses have been planned, considering different possible cases that may occur both on the academic and on the industrial arena. These courses have been entitled:

- TRAMA: traceability for interactive application (full-size course)
- TRAMA take-away (1 hour course)
- TRAMA crash course (1 day course)
- TRAMA: requirements traceability in practice (3 days course)

Each course is presented highlighting the target towards which it is directed, time and resources needed as well as the list of modules, learning goals and activity flow. A possible scheduling for each course is also provided.

5.3.1. TRAMA: traceability for interactive application

(i)Target and resources needed

The "full-size" TRAMA course has been conceived for university students at a master level as well as for practitioners. The course do not need an existing experience in requirements analysis and in design of interactive applications. The course length is from 50 to 60 hours for ex-cathedra lectures and tutoring and from 25 to 35 hours for the individual case study and the group work.

(ii) Selecting modules, learning goals and activity flow

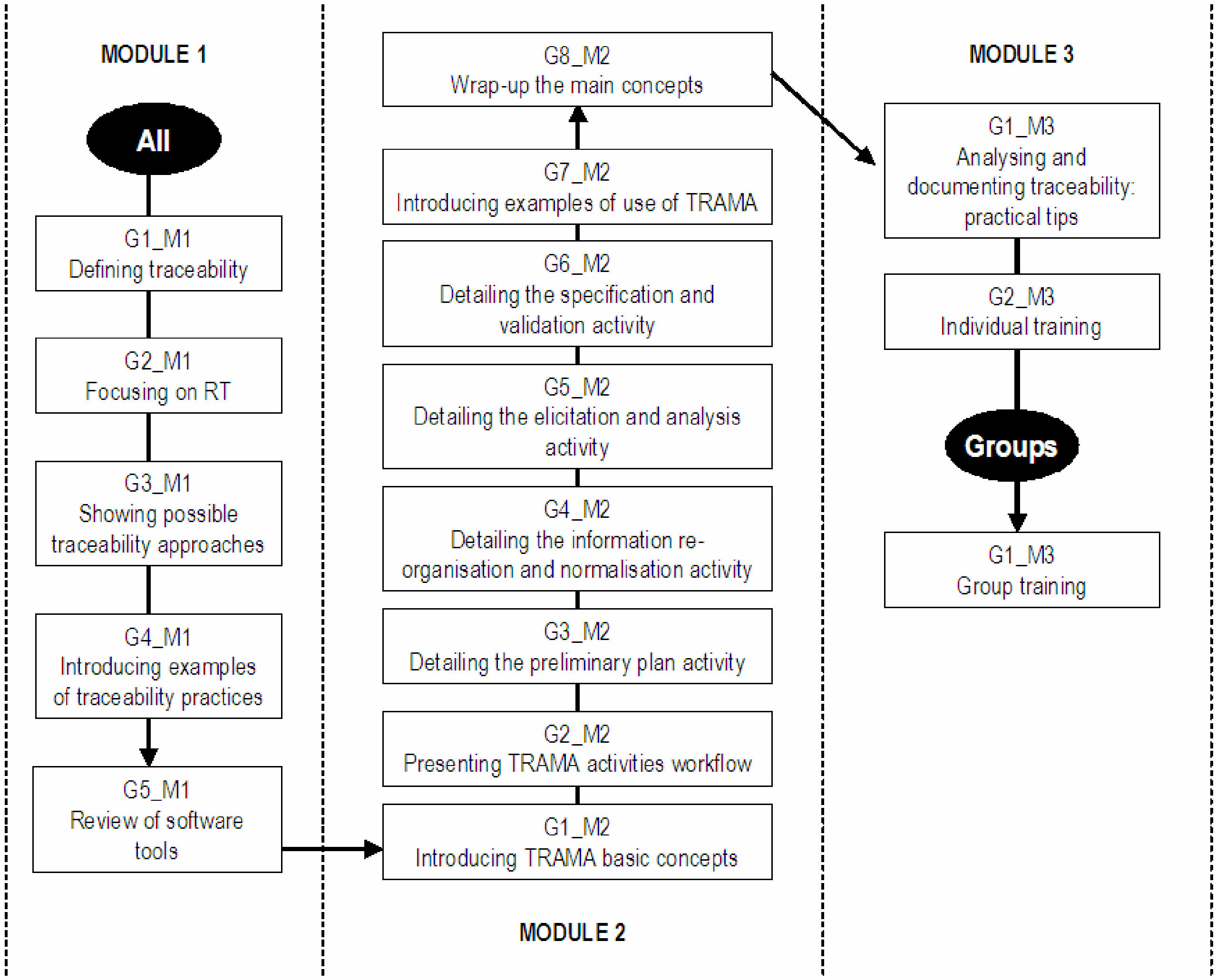


Figure 40. Activity flow of TRAMA full-size course

(iii) Scheduling

#	Hours	Activity type	Topic	Homework
1	4	Lecture	Course intro & traceability foundations	Assignment "software tools" (individual)
2	4	Lecture	Traceability approaches and examples	
3	3	Presentation	Software tools presentations	Delivery (in class) "Software tools" + Assignment "Traceability analysis 1" (individual)
4	4	Lecture	TRAMA basic concepts	
5	4	Lecture	TRAMA workflow 1: preliminary plan & information re-organisation & normalisation	
6	6	Tutoring	Individual tutoring	
7	2	Tutoring	Class questions & answers	
8	4	Lecture	TRAMA workflow 2: Elicitation and analysis	Delivery "Traceability analysis 1" + Assignment "Traceability analysis 2" (teamwork)
9	4	Tutoring	Group tutoring 1	
10	4	Lecture	TRAMA workflow 3: Specification and validation + Analysis examples	
11	4	Tutoring	Group tutoring 2	
12	2	Lecture	TRAMA documentation examples	
13	4	Lecture + Tutoring	Course wrap-up + Class q&a	
14	8	Presentation	Group analysis presentations	Delivery "Traceability analysis 2"

Table 13. Detailed scheduling of TRAMA full-size course

5.3.2.TRAMA take-away

(i)Target and resources needed

The shorter TRAMA course has been conceived for learners with an existing experience in requirements analysis and in design of interactive applications. In 1 hour this course gives to professionals or university students the basic concepts to perform an high-level traceability analysis.

(ii) Selecting modules, learning goals and activity flow

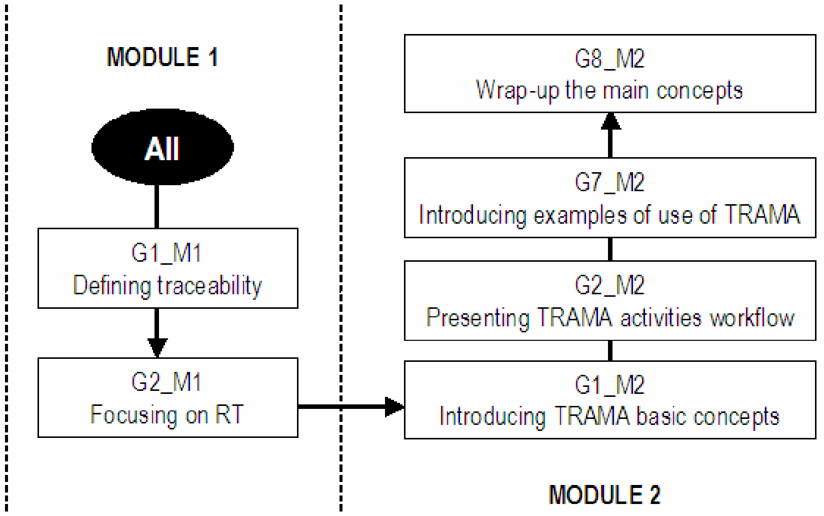


Figure 41. Activity flow of TRAMA 1-hour course

(iii)Scheduling

#	Time	Activity type	Topic
1	5'	Lecture	Traceability foundations
	15'	Lecture	TRAMA basic concepts
	10'	Lecture	TRAMA workflow 1
			Coffee break
	10'	Lecture	TRAMA workflow 2
	10'	Lecture	A short example of TRAMA analysis: Munch in Berlin
	5'	Lecture	Class questions & answers
	5'	Lecture	Course wrap-up

Table 14. Detailed scheduling of TRAMA 1-hour course

5.3.3. TRAMA crash course

(i)Target and resources needed

This TRAMA course has been conceived for university students at a master level as well as for practitioners. The course need an existing experience in requirements analysis and in design of interactive applications. The course length is of 1 day (8 hours) including ex-cathedra lectures, tutoring and the group work.

(ii) Selecting modules, learning goals and activity flow

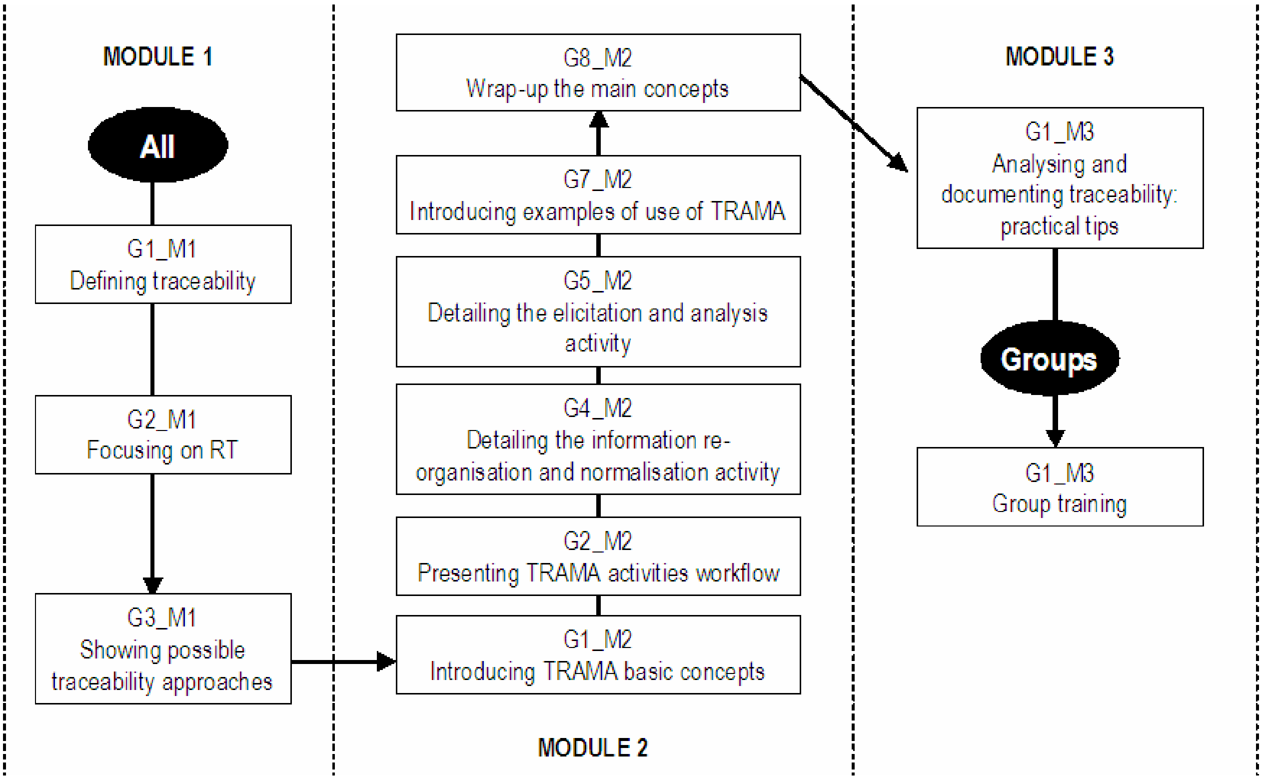


Figure 42. Activity flow of TRAMA crash course

(iii)Scheduling

#	Time	Hours	Activity type	Topic
1	15	08:00-08:15	Lecture	Course intro & traceability foundations
	15	08:15-08:30	Lecture	Traceability approaches and examples
	60	08:30-09:30	Lecture	TRAMA basic concepts
	30	09:30-10:00	Lecture	TRAMA workflow 1: information re-organisation
	30	*10:00-10:30*		*Coffee break*
	30	10:30-11:00	Lecture	TRAMA workflow 2: information normalisation
	60	11:00-12:00	Lecture	TRAMA workflow 2: Elicitation and analysis
	30	12:00-12:30	Lecture	TRAMA examples
	60	*12:30-13:30*		*Launch break*
	30	13:30-14:00	Lecture	Wrap-up + Class questions & answers
	120	14:00-16:00	Tutoring	Group tutoring
	30	*16:00-16:30*		*Coffee break*
	90	16:30-18:00	Presentations	Group analysis short presentations

Table 15. Detailed scheduling of TRAMA crash course

5.3.4. TRAMA: requirements traceability in practice

(i) Target and resources needed

This TRAMA course is particularly conceived for professionals and industrial practitioners and need an existing experience in requirements analysis and in design of interactive applications. The learners will achieve particular competences in managing traceability activities, in facilitating traceability meetings and in analysing and documenting these kind of information. The course length can be calculated as follows: from x to y hours for ex-cathedra lectures and tutoring and from z to k hours for the individual case study.

(ii) Selecting modules, learning goals and activity flow

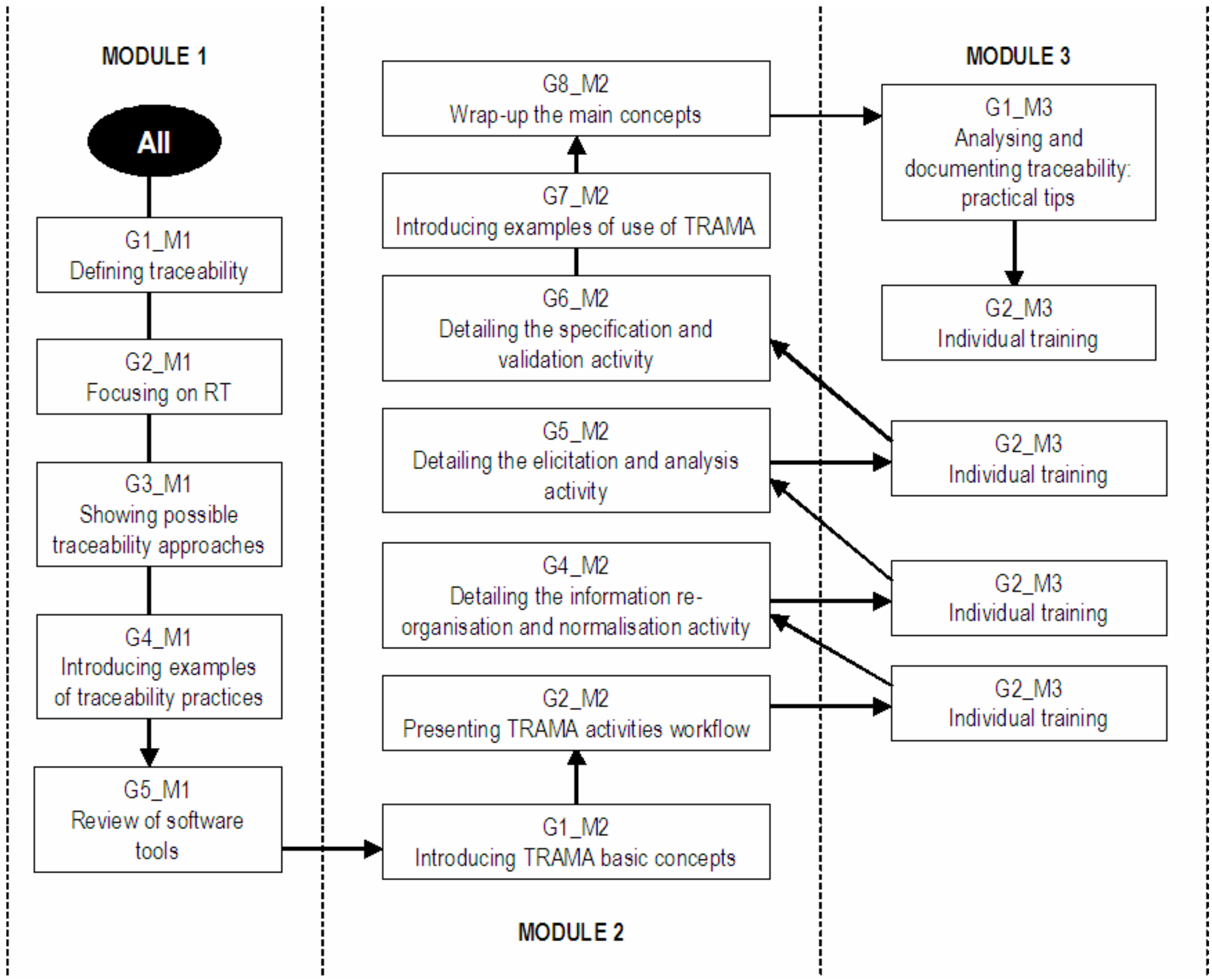


Figure 43. Activity flow of TRAMA practical course

(iii) Scheduling

#	Time	Hours	Activity type	Topic
1	1h	08:00-09:00	Lecture	Course intro & traceability foundations
	1.5h	09:00-10:30	Lecture	Traceability approaches, tools and examples
		10:30-10:45		*Coffee break*
	2h	10:45-12:45	Lecture	TRAMA basic concepts
		12:45-14:00		*Launch break*
	0.5h	14:00-14:30	Lecture	Example 1: Munch in Berlin
	0.5h	14:30-15:00	Discussion	Class discussion about the example
	2h	15:00-15:30	Exercise	Simple analysis exercise (individual)
		15:30-15:45		*Coffee break*
		15:45-17:15	Tutoring	Individual tutoring
	0.5h	17:15-17:45	Lecture + Tutoring	Day wrap-up + Class questions & answers
2	1h	08:00-09:00	Presentation	Presentation of the analysis exercise
	1h	09:00-10:00	Tutoring	Class discussion about the exercise
		10:00-10:15		*Coffee break*
	1h	10:15-11:15	Lecture	TRAMA workflow 1: information re-organisation & normalisation
	2.5h	11:15-12:30	Exercise	Information re-organisation and normalisation exercise (individual)
		12:30-13:30		*Launch break*
		13:30-14:45	Tutoring	Individual tutoring
	0.5h	14:45-15:15	Presentation	Presentation of the information re-organisation exercise (1)
		15:15-15:30		*Coffee break*
	1.5h	15:30-17:00	Presentation	Presentation of the information re-organisation exercise (2)
	0.5h	17:00-17:30	Lecture + Tutoring	Day wrap-up + Class q&a
3	1h	08:00-09:00	Lecture	TRAMA workflow 2: Elicitation and analysis
	1h	09:00-10:00	Lecture	TRAMA analysis examples
		10:00-10:15		*Coffee break*
	0.5h	10:15-10:45	Lecture	TRAMA workflow 3: Specification and validation
	0.5h	10:45-11:15	Lecture	TRAMA documentation examples
	2.5h	11:15-12:30	Exercise	Advanced analysis exercise (individual)
		12:30-13:30		*Launch break*
		13:30-14:45	Tutoring	Individual tutoring
	1h	14:45-15:45	Presentation	Presentation of the analysis exercise (1)
		15:45-16:00		*Coffee break*
	1h	16:00-17:00	Presentation	Presentation of the analysis exercise (2)
	0.5h	17:00-17:30	Lecture + Tutoring	Course wrap-up + Class q&a

Table 16. Detailed scheduling of TRAMA practical course

6.Conclusions & Future Works

<<Computer in the future may weigh no more than 1.5 tons.>>
Popular Mechanics, 1949

6.1. What is TRAMA?

This dissertation proposes a design traceability method supporting post-Requirements Specification Traceability in both a forward and backward direction. The method, called TRAMA - TRaceability Analysis Method for (interactive) Applications – was conceived for being used on interactive application projects, but its tools and concepts can be applied in wider domains, including information systems, knowledge management systems, educational application, etc. TRAMA bases on the use of structured matrices to facilitate the meetings with stakeholders and to analyse the surfaced information. The method helps find both impact relationships between requirements and design elements (forward traceability) and motivation relationships between design solutions and its sources (backward traceability).

6.2. The problem

A common opinion in the Requirements Traceability field is that solution design, i.e. the design of the application solutions, may be derived directly from requirements refinement; some works [Pohl et al., 1994; Egyed et al., 2000] consider design as the result of a requirements refinement process and try to record this process establishing a requirements-to-design traceability. TRAMA proposes a different thesis: the previous position cannot be taken as granted, because different experiences conducted for this research (and described in section 3) seems to invalidate it. TRAMA take into account that *at least in some cases* the design process is not a fully rational and explicit sequence of actions; designers keep requirements in mind as a background knowledge, and they build up the application architecture almost from scratch, as a result of a an inductive and in part intuitive activity. According to Arciszewsky [Arciszewsky et al., 1995], design is an intuition and induction process more than a derivation one. Some requirements remains implicit at the beginning of the project but they are considered in design; often, at the end of the project, designers do not remember the actual reason for that choices. This problem make very hard to verify, to evaluate, to revision and to reuse efficiently design solutions in relation with high-level requirements.

TRAMA do not take into account the mental process that brings from general requirements to concrete design solutions; on the contrary, the tracing activity tries to move intuition and induction to more rational cause-effect motivations. This kind of analysis must not repress or stiffen the design process, but it helps in better understanding the reasons for design choices and it forces to better make explicit requirements that are both implicit or unexpressed.

6.3. Non-original concepts

In the last 15 years, Requirements Traceability (RT) has been identified in the literature as a quality factor, i.e. a characteristic a system should possess and include as a non-functional requirement. Major concepts and suggestions that TRAMA borrows from literature are:

- the requirements to design tracing activity, formerly studied for the methodologies PRO-ART [Pohl et al., 1994] and CBPS [Egyed et al., 2000];

- the idea of record the reasons for design decisions as in the Potts and Bruns model [Potts & Bruns, 1988];
- the idea of explicitly representing different design choices and the reasons for choosing one of them as in what I call the "Xerox approach" [McLean et al., 1989];
- the concept of design rationale as in [Arkley et al., 2002];
- the concept of rich traceability as in [Dick, 2002];
- the idea of Design Versioning, inspired by the Contributions Structures approach [Gotel & Finkelstein, 1995].

6.4. Hypothesis

Current methodologies focus on the traceability of singles objects and on the use of conceptual tools that emphasise a "punctiform" view, but they do not consider a more global picture. **Hypothesis 1** was that design structure and requirements structure are not (or at least not always) homogeneous: in some cases, a requirement does not impact on a single design element but on a number of design elements and on their interactions. The case studies presented in section 3 show that *at least in these cases* the hypothesis can be validated. For this reason, the TRAMA traceability process produces a project's complex picture under control, adopting matrices as conceptual tools allowing both a detailed analysis and a global picture management.

The case studies conducted for this research, highlighted an interesting factor: requirements are not the only motivation or justification for a design element, i.e. sometimes the reason why a certain application choice has been taken, cannot be referenced to a particular goal or requirement. **Hypothesis 2** states that in some cases the motivations for design choices can be found not only in requirements but also in other elements such as a wider knowledge about the specific application domain, some project constraints, "good design" and usability principles, designer skills, etc. TRAMA takes therefore in consideration as design sources the designer expertise, specific understandings of the domain, particular constraints, law obligations, requirements-related information and arbitrary choices.

The same experiences shown in section 3, highlighted a further interesting element: due to organisational problems (e.g. resources allocation) and psychological issues (e.g. problems in self-observation), the traceability expert is viewed as having a different role than a designer or a requirement analyst; a traceability expert is n fact a facilitator in project meetings, with specific competences in eliciting and understanding why a certain decision is being or has been taken. This element seems to validate the **hypothesis 3**. TRAMA considers therefore the traceability activity forward to and backward from design elements as an argumentation activity about why design solutions satisfy requirements. This aptitude is mirrored in the TRAMA activity workflow, structured as a self-standing process composed by: a preliminary plan, an information re-organisation activity, an information "normalisation" activity, an elicitation phase, the analysis, a specification activity and the results validation with clients.

6.5. Characteristics of TRAMA

TRAMA does not base on the use of a specific software tool. The state-of-the-art review performed in section 2 shows that research efforts in the field of RT share the use of formal graphs or formal languages to represent the relevant entities and the traces between them, as well as the idea that the huge mass of traceability information produced need a software tool to be managed. In some cases, the use of this kind of tools suggests the possibility of an automation of the tracing process. Some practices have therefore access problems for the user (communication problems); methodologies are often not clear, not complete or too formal. TRAMA does not want to depend on the use of a software tool to be applied; the method is conceived to be usable also with general-purposes tools (e.g. Excel) or simply drawing on paper (e.g. on the flipchart during meetings).

In current methodologies and practices, the impact of strategic goals and requirements is analysed considering as target technical design features or application elements, in terms of pieces of contents and functionalities, UML-classes, pieces of code or objects. Sometimes (e.g. in educational systems) it is impossible to understand the impact of requirements-related elements into the application considering only the software elements of the system; in other words, it is impossible to understand the project solutions without considering contextual information. TRAMA adopts therefore a wider definition of "design", called "experience design" because the overall user experience is considered in all its aspects: conceptual elements, contextual settings, organisational elements, and other accessorial elements (e.g. study material)

As said before, the TRAMA traceability process produces a global complex picture, taking under control by the use of matrices as conceptual tools. In particular, TRAMA make use of matrices that cross requirements with design in a forward direction and design with its sources (requirements, motivations, constraints, etc.) in a backward direction. Requirements-to-Design matrix called RIM (Requirements Impact Model/Matrix) can be filled and read both horizontally, highlighting how single requirements are taken into account into the design, and vertically, showing how a single design element satisfies the project requirements. Design-to-Sources matrix called DMM (Design Motivations Model/Matrix) traces back single design elements to the motivation why a certain decision is relevant for the project, e.g. satisfying a requirements, fulfilling a constraint, allowing more usability in the system, etc.

Last trends in RT (e.g. [Dick, 2002]) focus on adding explicit semantics to relationships, in particular for traces that involve conceptual design elements or pieces of the application (codes, classes, use cases, etc.). According to this approach, TRAMA traces does not mean anything *per se* but they have to be completed and commented through notes highlighting the rationale of the relationships.

6.6. Benefits and limits

Benefits in the use of TRAMA are mainly the following:

- TRAMA is a powerful communication mean to show to the clients that all their requirements have been considered and how, and that there are not unmotivated elements in the design;
- TRAMA is a structured practice to check requirements and design consistency for revision, surfacing missing design elements and missing requirements; the method supports reverse requirements engineering;
- TRAMA is an advanced tool to tune up and re-align design in maintenance phase and to assign priorities to design elements;
- TRAMA specifications provide a complete project knowledge summary of requirements-related information, of design elements and of relationships between them, as vital information allowing an effective system reengineering, workflow organisation and more focused verification procedures to be performed.

TRAMA unfortunately includes some limits, which could be the focus of a new step in this research. As a kind of self-criticism, I have identified two major limits in this approach:

- a lack of a massive set of cases in which TRAMA has been applied
- some organisational problems in maintaining the RIM and DMM matrices up-to-date.

6.7. Future works

Current evolution of TRAMA still needs further experimentation to be validated on a large-scale basis. So far, positive feedback has been received from those practitioners and scholars with whom projects have been carried out in the academic and industrial arena. TRAMA has been introduced also in academic classes (at University of Lugano and Politecnico di Milano) focusing on requirements and design for web and multichannel applications. These courses (involving overall more than 300 students a year) are targeted not only to people with a technological background, but also to students who studied communication sciences, tourism, cultural heritage, and humanities in general. Future research will focus on consolidating the method by applying these concepts to projects in other domains. Finally, the design of specific support tools is also being considered to facilitate the efficient documentation of the analysis material and the corresponding communication activity.

References & Bibliography

AAVV (2002). Rational Requisite Pro, Product Description [on line]. URL: http://www.rational.com/products/reqpro/index.jsp [19.08.2002].

Alexander, I. (2003). Semi-Automatic Tracing of Requirement Versions to Use Cases. ASE2003:http://www.soi.city.ac.uk/%7Egespan/paper8.pdf 2nd International Workshop on Traceability in Emerging Forms of Software Engineering. October 8-10, Montreal, Canada.

Ambler, S. (1999). Tracing your Design. In: Software Development, April 1999 [on line]. URL: http://www.sdmagazine.com/documents/s=815 /sdm9904d/ [19.08.2002].

Antón, A. (1996). Goal-Based Requirements Analysis. In: International Conference on Requirements Engineering (ICRE `96), Colorado Springs, Colorado, USA, April 1996, pp. 136-144.

Antón, A. (1997). Goal Identification and Refinement in the Specification of Software-Based Information Systems, Ph.D. Thesis, Georgia Institute of Technology, Atlanta, GA.

Antón, A., Dempster, J. & Siege, D. (2000). Deriving goals from a use case based requirement specification for an electronic commerce system. In: Proceedings from RE01 - Fifth IEEE International Symposium on Requirements engineering. April 25, 2001 Imperial College, London.

Arciszewski, T., Michalski, R. S. & Wnek, J. (1995). Constructive induction: the key to design creativity. In: Proceedings of the Third International Round-Table Conference on Computational Models of Creative Design, Heron Island, Queensland, Australia, December 3-7, 1995.

Arkley, P., Mason, P. & Riddle, S. (2002). Position Paper: Enabling Traceability. TEFSE 2002: 1st International Workshop on Traceability in Emerging Forms of Software Engineering. September 28th, Edinburgh, U.K.

Armani, J, Botturi, L., Cantoni, I., Di Benedetto, M. & Garzotto, F. (2004). Integrating Instructional Design and Hypermedia Design. ED-MEDIA 2004, 21-26 June, Lugano, Switzerland

Arnold, R. & Bohner, S. (1993). Impact Analysis. Towards A Framework for Comparison. In: Proceeding from the Conference on Software Maintenance. IEEE CS Press, Los Alamitos, California, pp. 292-301.

Bedocs Jr., J. (1999). A Data Architecture for DOORS Projects, InDOORS 1999 [on line]. URL: http://www2.telelogic.com/doors/index.cfm [19.08.2002].

Boehm, B.W. (1991). Software Risk Management: Principles and Practices. In: IEEE Software. 8(1): 32-41.

Boerstler, J. & Janning, T. (1991). Bridging the gap between Requirements Engineering and Design. In: Proceedings of 15th International Computer Software & Application Conference.

Boerstler, J. & Janning, T. (1992). Traceability Between Requirements and Design: A Transformational Approach. In: Proceedings of16th International Computer Software & Application Conference, pp. 362-368.

Bohner, S. & Arnold, R. (1996). Software Change Impact Analysis. IEEE Computer Society Press. Los Alamitos, California.

Bohner, S. (1991). Software Change Impact Analysis for Design Evolution. In: Proceedings of 8th International Conference on Software Maintenance and Re-engineering.

Bolchini, D., Paolini, P. & Randazzo, G. (2003). Adding Hypermedia Requirements to Goal-Driven Analysis. RE03 - International Conference on Requirements Engineering, Monterey, California.

Bolchini, D., Piccinotti, N,, Randazzo, G., Gobbetti, D. (2005a). IDM – A User-Centred Model Shaping User Interaction as a Dialogue. In: Proceedings on HCII05, International Conference on Human-Computer Interaction, 22-27 July, Las Vegas, Nevada.

Bolchini, D., Randazzo, G., Paolini, P. (2005b). Vision-Driven Requirements Analysis for Communication Design. RE05 - International Conference on Requirements Engineering, Paris, France.

Botturi, L. (2004). E2ML: A Visual Instructional Design Language. AECT 2004 Convention, Chicago, Illinois, USA [Proceedings Vol 2, 70-79].

Botturi, L., & Belfer, K. (2003). Pedagogical Patterns for Online Learning. *World Conference on E-Learning in Corp., Govt., Health., & Higher Ed. 2003*(1), 881-884.

CADRE. (1992). Teamwork/RqT, Marketing Brochure, CADRE Technologies, Inc.

Cantoni, L., Di Blas, N. & Bolchini, D. (2003). Comunicazione, qualità, usabilità. Apogeo, Lugano.

Carlsharmre, P. & Regnell, B. (2000). Requirements Lifecycle Management and Release Planning in Market-Driven Requirements Engineering Processes. In: 13th International Conference on Database and Expert Systems Applications - DEXA 2002, pp. 961-965.

Carlsharmre, P. & Regnell, B. (2000). Requirements Lifecycle Management and Release Planning in Market-Driven Requirements Engineering Processes. DEXA, pp. 961-965.

Casotto, A. (1993). Run-Time Requirements Tracing. In: Proceedings of IEEE/ACM international conference on Computer-aided design 1993, Santa Clara, California, United States.

Casotto, A., Newton, R., Sangiovanni-Vincentelli, R. (1990). Design management based on design traces. In: Conference proceedings of 27th ACM/IEEE design automation conference 1990. Orlando, Florida, United States.

Castro, J., Kolp, M.& Mylopoulos, J. (2002). Towards Requirements-Driven Software Development Methodology: The Tropos Project. Information Systems, June.

Christel, M. G. & Kang, K. C. (1992). Issues in Requirements elicitation. Technical Report CMU/SEI-929TR-012. ESC-TR-92-012.

Conklin, J. (1989). Design rationale and maintainability. In: Proceedings of 22nd International Conference on System Science, Vol. 2, pp. 533-539.

Corriveau, J.-P. (1996). Traceability Process for Large OO Projects. In : IEEE Computer, Vol. 29, No. 9, pp. 63-68.

Dardenne, A., van Lamsweerde, A. & Fickas, S. (1993). Goal-directed Requirements Acquisition. In: Science of Computer Programming, vol. 20, pp. 3-50.

Darimont, R. & van Lamsweerde, A. (1996). Formal refinement Patterns for Goals-driven requirements elaboration. In: Proceedings 4th ACM Symposium on the foundations of software engineering (FSE4), San Francisco, Oct. 1996, pp. 179-190.

Di Blas, N., Paolini, P., Poggi, C. (2005). 3D Worlds for Edutainment: Educational, Relational and Organizational Principles. In: Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications, 8-12 March 2005, Kauai Island, HI, USA.

Dick, J. (2002). Rich Traceability. In: Proceedings Automated Software Engineering, 23-27 Sept. 2002**.**

Dömges, R. & Pohl, K. (1998). Adapting Traceability Environments to Project-Specific Needs. In: Communications of the ACM, Vol. 41, No. 12 (dec 1998), pp. 54-62.

Edwards, M. & Howell, S. (1991). A methodology for system requirements specification and traceability for large real-time complex systems. Technical report, Naval Surface Warfare Center.

Egyed, A., Gruenbacher, P. & Medvidovic, N. (2000). Refinement and Evolution Issues in Bridging Requirements and Architectures. Technical Report of USCCSE.

Farncombe, A. (2004). Process Flexibility and the 'Missing Levels' Problem. Proceedings from IEE - Systems Engineering for the Railway Industry, 26th May 2004.

Finholt, T. A., Horn, D. & Thomé, S. (2003). NEESgrid requirements Traceability Matrix. Technical Report NEESgrid-2003-13.

Garzotto, F., Mainetti, L. & Paolini, P. (1993). HDM2: Extending the E-R Approach to Hypermedia Application Design. In: Proceedings of the 12th International Conference on the Entity-Relationship Approach: Entity-Relationship Approach. Springer-Verlag Ltd. London.

Goguen, J. (1997). Formal Methods: Promises and problems. In: IEEE Software, January 1997.

Goguen, J. (2000). Parameterised programming and software Architecture. University of California, San Diego.

Gotel, O. & Finkelstein, A. (1994). An Analysis of the Requirements Traceability Problem. In: Proceedings of International Conference on Requirements Engineering (ICRE), IEEE CS Press, pp. 94-101.

Gotel, O. & Finkelstein, A. (1995). Contribution Structures. In: Proceedings of 2nd International Symposium on Requirements Engineering (RE '95), York, UK, pages 100-107.

Gotel, O. & Finkelstein, A. (1996). Making requirements Elicitation traceable. Workshop Introduction, Department of Computing, Imperial College of Science, Technology and Medicine.

Gotel, O. & Finkelstein, A. (1997). Extended requirements traceability—Results of an industrial case study. In: Proceedings of 3rd International Symposium on Requirements Engineering. (Annapolis, Md., Jan. 1997), pp.169–178.

Grünbacher, P., Egyed, A. & Medvidovic, N. (2001). Reconciling Software Requirements and Architectures – The CBSP Approach. In: Proceedings of the 5th IEEE International Symposium on Requirements Engineering (RE01). Springer-Verlag, Toronto, Canada, pp. 202-211.

Grundy, J., Hosking, J. & Mugridge, W. (1998). Inconsistency Management for Multiple-View Software Development Environments. In: IEEE Transactions on Software Engineering, Vol. 24, No. 11, pp. 960-981.

Hamilton, V. & Beeby, M. (1991). Issues of Traceability in Integrating Tools. In: IEE. Tools and Techniques for Maintaining Traceability During Design, IEE Colloquium, Computing and Control Division, Professional Group C1, Digest No.: 1991/180., pp. 4/1-4/3.

Hammond, J., Rawlings, R. & Hall, H. (2001). Will it work?. RE01 - 5th IEEE International Symposium on Requirements Engineering. April 25, London, UK.

Haumer, P., Pohl, K. & Weidenhaupt, K. (1998). Requirements Elicitation and Validation with Real world Scenes. CREWS Report 98-16. In: IEEE Transactions on Software Engineering, Vol. 24, No. 12 (dec 1998). Special Issue on Scenario Management.

Hayhurst, K. J. (1997). A comprehensive Look at the Guidance and Control Software (GCS) Project. NASA Langley Research Center Report.

Hilton, A. (2003). Engineering Software Systems for Customer Acceptance. Report of Praxis High Integrity Systems.

Holmer, M. (2001). A glimpse at Requirements Traceability. In: Hauptseminar requirements Engineering (12.10.2001) – Presentation slides.

IEEE Std 830-1993. IEEE Recommended practice for Software Requirements Specifications. Dec 2 1993

IEEE/EIA 12207. Standard for Information Technology.

Institute of Electrical and Electronics Engineers. IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries. New York.

International Council on Systems Engineering (2001). SE Tools Taxonomy –Requirements Traceability Tools [on line]. URL: http://www.incose.org/tools/tooltax/reqtrace_tools.html [20.08.2002]

ISO/IEC. Information Technology - Information Resource Dictionary Systems (IRDS) - ISO/IEC Framework 10027.

Jarke, M. (1998). Requirements Tracing. In: Communications of the ACM, Vol. 41, No. 12 (dec 1998), pp. 32-36.

Jarke, M., Pohl, K., Rolland, C. & Schmitt, J. (1994). Experience-based method evaluation and improvement: a process modelling approach. In: IFIP WG8.1 Conference CRIS'94.

Jones, D., Kar, P., van Gaasbeck, J., Hollenbach, F., Bell, M. & Ellinger, R. (1997). Interfacing Requirements Management tools in the requirements management process - a first look. In: 7th Int. Symposium of the INCOSE, Vol.2 (aug 1997) [on line]. URL: http://incose.org/rwg/97_paper_inter/inter_rmt.html [19.08.2002].

Kean, L. (2002). Argument-Based Design Rationale Capture Methods for Requirements Tracing. In: Software technology review. URL: http://www.sei.cmu.edu/str/descriptions/argument.html [20.08.2002].

Kean, L. (2002). Requirements tracing – An overview. In: Software technology review [on line]. URL: www.sei.cmu.edu/str/descriptions /reqtracing_body.html [20.08.2002].

Kenny, C. (1996) Requirements Traceability. Cmpt 856 Project.

Kirkman, D. (1998). Requirement Decomposition and Traceability. In: Journal Requirements Engineering, Vol. 3, No. 2, pp. 107-114.

Langford, D. (1991). PORC 0.41:Outline Description of Enhancements and Changes. BT Internal Report, Ipswich, 28 June.

Lavazza, L. & Valetto, G. (1998). Enhancing Requirements and change management through process modelling and measurement. CEFRIEL and Politecnico di Milano.

Lee, J. (1991). Extending the Potts and Bruns Model for Recording Design Rationale. ICSE91: 13th International Conference on Software Engineering, 13-16 May, Austin, Texas, USA.

Leffingwell, D. & Widrig, D. (2000). Managing Software Requirements. A Unified Approach. Addison Wesley.

Leon, M. (2000). Staying on Track. In: Intelligent Enterprise, pp. 54-57.

Lindvall, M. & Sandahl, K. (1996). Practical Implications of Traceability. In: Software—Practice and Experience, Vol. 26, No.10 (1996), pp. 1161–1180.

Lindvall, M. (1994). A Study of Traceability in Object-Oriented Systems Development. Licentiate thesis Linköping Studies in Science and Technology No 462, Linköping University, Institute of Technology, Sweden.

Lowe, D. (2003). Web System Requirements: An Overview. In: Requirements Engineering, July, vol. 8, n.2. Springer-Verlag Ltd, London.

Macfarlane, I. & Reilly, I. (1995). Requirements Traceability in an Integrated Development Environment. In: Proceedings of International Symposium on Requirements Engineering, pp. 116-123.

MacLean, A., Young, R. M. & Moran, T. P. (1989). Design Rationale: the Argument behind the Artifact. CHI89: International Conference on Computer-Human Interaction, 30 April-4 May. Austin, TX, USA.

Madhavji, N. (1994). Environment evolution: the Prism model of changes. In: IEEE Transactions on Software Engineering, Vol. 15, No.5, pp. 380-392.

Maletic, J. I., Munson, E. V., Marcus, A. & Nguyen, T. N. (2003). Using a Hypertext Model for Traceability Link Conformance Analysis. TEFSE03 - 2nd International Workshop on Traceability in Emerging Forms of Software Engineering. Montreal, Quebec, Canada. Oct. 7, 2003.

Nejmeh, B., Dickey, T. & Wartik, S. (1989). Traceability technology at the software productivity consortium. G.X. In: Riter, (Ed). Information Processing '89. Elsevier Science Publishers, New York.

Nuseibeh, N. (2000). Requirements engineering : A Roadmap. In: Conference at Imperial College, London.

OMG (1999). Unified Modelling Language. Technical report, Rational Software Corporation, Version 1.3 [on line]. URL: http://www.rational.com/uml [19.08.2002].

Palmer, J. (1997). Traceability. In: Thayer, R. & Dorfman, M. (Eds). Software Requirements Engineering. IEEE Computer Society Press, New York.

Parnas, D. & Madey, J. (1991). Functional Documentation for Computer Systems Engineering. CRL Report 237, McMaster University, Hamilton, Ontario, Canada.

Pinheiro, F. & Goguen, J. (1996). An object-oriented tool for tracing requirements. In: IEEE Software., Vol. 13, No. 2 (mar 1996), pp. 52–64.

Pinheiro, F. (1996). Design of a Hyper-Environment for Tracing Object-Oriented Requirements. Wolfson Collage, Dissertation, University of Oxford.

Pinheiro, F. (1997). Formal and Informal Aspects of Requirements Tracing. In: Sthepen, D. Trends in Software Engineering [on line]. URL: http://sern.ucalgary.ca/~daves/SENG693/SENG693_20010123-Pinheiro.html [14.08.2002].

Pohl, K. (1996). Process-Centred Requirements Engineering. Research Studies Press Ltd. England.

Pohl, K. (1999). Requirements Traceability: Models of the Development World. In: Jarke, M., Rolland, C., Sutcliffe, A. & Dömges, R. (Eds.). The NATURE of Requirements Engineering. Shaker Verlag, pp. 117-146.

Pohl, K., Brandenburg, K. & Gülich, A. (2001). Integrating Requirement and Architecture Information: A Scenario and Meta-Model Based Approach. In: Seventh International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ, June 2001).

Pohl, K., Dömges, R. & Jarke, M. (1994). PRO –ART. PROcess based Approach to Requirements Traceability. Technical report, Informatik V, RWTH-Aachen, Ahornstrass 55, 52056 Aachen, Germany.

Pohl, K., Dömges, R. & Jarke, M. (1997). Towards selective, model-driven trace capture. In: Proceedings of 7th International Conference on Advanced Information Systems Engineering (Barcelona).

Potts, C. & Bruns, G. (1988). Recording the reasons for design decisions. 10th international conference on Software engineering. 11-15 April, Singapore.

Praxis Critical Systems (2001). REVEAL - A Keystone of Modern Systems Engineering. [on line] http://www.praxis-his.com/reveal/index.htm.

Ramamoorthy, C. V., Garg, V. and Prakash, A. (1988). Support for Reusability in Genesis. *IEEE Transactions on Software Engineering*, SE-14, Number 7, July, pp. 1145-1153.

Ramamoorthy, C., Usuda, Y., Prakash, A. & Tsai, W. (1990). The evolution support environment system. In: IEEE Transactions on Software Engineering, Vol. 16., No. 11, pp. 1225-1234.

Ramesh, B. & Dhar, V. (1992). Supporting Systems Development by Capturing Deliberations During Requirements Engineering. In: IEEE Transactions on Software Engineering, Vol. 18, No. 6, pp. 498-510.

Ramesh, B. & Jarke M. (2001). Towards Reference Models for Requirements Traceability. In: IEEE Transactions in Software Engineering, Vol. 27, No. 1, pp. 58-93.

Ramesh, B. (1998). Factors influencing requirements traceability practice. In: Communications of the ACM, Vol. 41, No.12 (dec 1998), pp. 37-44.

Ramesh, B. (2002), Acquisition Knowledge Management with traceability. Interim Report founded by the external acquisition research program, Naval Postgraduate School.

Ramesh, B., Harrington, G., Rondeau, K. & Edwards, M (1993). A model of requirements traceability to support systems development. Technical Report NPS-SM93-017 Naval Surface Warfare Center, Dahlgren Division, 10901 New Hampshire Avenue, Silver Spring, Maryland 20903-50000 U.S.A.

Ramesh, B., Powers, T., Stubbs, C. & Edwards, M. (1995). Implementing Requirements Traceability: A Case Study. In: Proceedings of International Symposium on Requirements Engineering, IEEE Computer Society Press, pp. 89-95.

Ramesh, B., Powers, T., Stubbs, C., & Edwards, M. (1993). A study of current practices of requirements traceability in systems development. Technical Report NPS-SM93-018 Naval Surface Warfare Center, Dahlgren Division, 10901 New Hampshire Avenue, Silver Spring, Maryland 20903-50000 U.S.A.

Ramesh, B., Stubbs, C., Powers, T., and Edwards, M. (1997). Requirements traceability —Theory and practice. In: Annals of Software Engineering, Vol. 3 (1997), pp. 397–415.

Randazzo, G. (2001). User models in the customised document delivery on the Web: a Semantic Web approach. Memoria di licenza, Università della Svizzera italiana, Facoltà di Scienze della comunicazione, a.a. 2000/2001 - (on the Web site of the Biblioteca Universitaria di Lugano).

Randazzo, G. (2002). Traceability survey. A state-of-the-art on requirements traceability. TEC-lab report TR02.6, Lugano, September 2002.

Randazzo, G. (2004). TRAMA: Traceability Analysis Methodologies for (Web) Applications. RE04 - International Conference on Requirements Engineering, Kyoto, Japan.

Randazzo, G. (2005a). Requirements to Design: an Effective Traceability Method to Elicit Knowledge about Requirements Impact and Design Motivations. To be published.

Randazzo, G. (2005b). "Museum of non-European cultures": a Design Traceability Case Study adopting the TRAMA approach for Interactive Applications. SCOMS – Studies in Communication Sciences, vol., num., Lugano, Switzerland.

Randazzo, G., Alonzo, F. & Paolini, P. (2003). Opendrama - The digital heritage of opera in the open network environment. ICNEE03 - The International Conference on New Educational Environment, May 26-28 2003, Luzern, Switzerland.

Randazzo, G., Triacca, L. & Speroni, M. (2003). A systematic web requirements approach for a more effective usability evaluation plan. INTERACT03 - IFIP TC13 International conference on human-computer interaction, September 1-5 2003, Zürich, Switzerland.

Reigeluth, C.M. (ed.) (1999). Instructional-Design Theories and Models Volume II – A New Paradigm of Instructional Theory. Mahwah, NJ: Lawrence Erlbaum Associates Publishers.

Sherba, S.A., Anderson, K. M. & Faisal, M. (2003). A framework for mapping traceability relationships. ASE2003:http://www.soi.city.ac.uk/%7Egespan/paper8.pdf 2nd International Workshop on Traceability in Emerging Forms of Software Engineering. October 8-10, Montreal, Canada.

Silva, A. & Marcocci, L. (1998). Improvement of the testing process exploiting requirements traceability. Process Improvement Experiment Final Report, VASIE, http://www.esi.es.

Smith, B. (1999). The Application of a Tool Like RDD-100 to Accommodate Traceability from Requirements to Test Results [on line]. URL: http://www.alc.com/archive.shtml [20.08.2002].

Sodhi, J. (1991). Software Engineering: Methods, Management, and CASE Tools, McGraw-Hill.

Song, X., Hasling, B., Mangla, G. & Sherman, B. (1998). Lessons Learned From Building A Web-Based Requirements Tracing System. In: Proceedings of International Conference on Requirements Engineering, pp. 41-50.

Spanoudakis G. (2003). Supporting the Construction of Decision Models for Software Selection: A Probabilistic Traceability Approach. ASE2003: http://www.soi.city.ac.uk/%7Egespan/ paper8.pdf. 2nd International Workshop on Traceability in Emerging Forms of Software Engineering. October 8-10, Montreal, Canada.

Spanoudakis, G. (2002). Plausible and Adaptive requirement Traceability Structures. In: 14th International Conference on Software Engineering and Knowledge Engineering, Ischia, Italy.

Spence, I. & Probasco, L. (1998). Traceability Studies for Managing Requirements with Use Cases [on line]. URL: http://www.rational.com/ products/whitepapers/022701.jsp [19.08.2002].

Stark, G. , Oman, P. , Skillicorn, A. & Ameele, A. (1999). An Examination of the Effects of Requirements Change on Software Maintenance Releases. In: Journal of Software Maintenance: Research and Practice, No. 11, pp. 293-309.

Stout, G. (2001). Requirements Traceability and the effect on the System Development Lifecycle (SDLC). Research Paper 2, DISS 725: System Development Process, Spring Cluster.

Strasunskas, D. (2003) Traceability in Collaborative Systems Development from Lifecycle Perspective: A Position Paper. 6th International Workshop on Termination. June 13-14, Valencia, Spain.

Takeda, N., Shiomi, A., Kawai, K. and Ohiwa, H. (1993). Requirements Analysis by the KJ Editor. In: Proceedings of the IEEE International Symposium on Requirements Engineering, San Diego, California, January 4-6, pp. 98-101.

Texas Instruments. (1988). A Guide to Information Engineering Using the IEF: Computer- Aided Planning, Analysis, and Design. Second Edition.

Triacca, L. (2001). Requirements and evaluation for E-commerce Web sites. Memoria di licenza, Università della Svizzera Italiana, Lugano, Switzerland.

Triacca, L. (2004). Enhancing the Effectiveness of Methodologies and Improving their Communication Features: the U2 Approach. Internal report, Università della Svizzera Italiana, Lugano.

Triacca, L., Randazzo, G., Bolchini, D. (2005). Improving prototype-BASED usability: lessons-learned from the Opendrama case study. EVA 2005, Florence, Italy.

Tsang, K. & Chan, E. (1998). Test Design using DOORS. InDOORS'98 [on line]. URL: http://www2.telelogic.com/doors/index.cfm [19.08.2002].

Tvete, B. & Sundnes, P (1999). Requirements Management in Alcatel Telecom Norway Defence Communications Division. Process Improvement Experiment Final Report, VASIE, http://www.esi.es.

Van Brink, M. R. & Kuebler, S. (2004). Traceability Matrix: Taking the Guesswork out of Fulfilling User Requirements. Solvay Pharmaceuticals Report.

van Lamsweerde, A. (2000). Requirements Engineering in the Year 00: A research perspective. In: Proceedings on 22nd International Conference on Software engineering, Limericjk, June 2000, ACM Press.

van Lamsweerde, A. (2001). Building formal requirements models for reliable software. In: Reliable Software Technologies, Ada-Europe 2001, Springer-Verlag notes in computer sciences, LNCS 2043, May 2001.

van Lamsweerde, A. (2001). Goal-Oriented Requirements Engineering: A guided tour. In: Proceedings RE01, 5th IEEE International Symposium on requirements Engineering, Toronto, August 2001, pp. 249-163.

van Lamsweerde, A., Darimont, R. & Massonet, P. (2001). Goal-directed elaboration of requirements for a meeting scheduler: problems and lessons learnt. In: Proceeding RE95, 2nd IEEE Symposium on requirements engineering, York, March 1995, pp. 195-203.

van Lamsweerde, A., Darmont, R. & Letier, E. (1998). Managing Conflicts in Goal-Driven Requirements Engineering. In: IEEE Transactions on Software Engineering, Vol. 24, No. 11 (nov 1998), pp. 908-926.

van Lamsweerde, A., Tran Van, H., Massonet, P., Ponsard, Ch. (2004). Goal-Oriented Requirements Animation. RE04 - International Conference on Requirements Engineering, Kyoto, Japan.

von Knethen, A. & Paech, B. (2002). A Survey on Tracing Approaches in Practice and Research. IESE-Report No. 095.01/E, Fraunhofer Institut Experimentelles Software Engineering.

von Knethen, A. (1992). Change-Oriented Requirements Traceability. Support for Evolution of Embedded Systems. PhD Theses in Experimental Software Engineering, Vol. 9, Fraunhofer IRB.

von Knethen, A. (2002). Automatic Change Support based on a Trace Model. TEFSE 2002: 1st International Workshop on Traceability in Emerging Forms of Software Engineering. September 28th, Edinburgh, U.K.

Watkins, R. & Neal, M. (1994). Why and how of Requirements Tracing. In: IEEE Software, Vol. 11, No. 4, pp. 104-106.

Weidenhaupt, K., Pohl, K., Jarke, M. & Haumer, P. (1998). Scenarios in System Development: Current Practice. In: IEEE Software, Vol. 15, No. 2 (mar-apr 1998), pp.34-45.

WIC Program (2004). Functional Requirements Document for a Model WIC Information System. WIC official documentation. [on line] http://www.fns.usda.gov/wic/.

Wiegers, K. (1999). Automating Requirements Management. In: Software Development (July 1999).

Wieringa, R. (1995). An introduction to requirements traceability. Technical Report IR-389, Faculty of Mathematics and Computer Science, Vrije Universiteit, Amsterdam.

Wieringa, R. (1997). Traceability and Modularity in Software Design. Proceedings of the 9th international workshop on Software specification and design. Ise-shima, Japan April 16-18.

Woodruff, W. (1997). Requirements Management for Small Organisation. Rational Software Corporation Report. [on line] http://home.jtan.com/~wayne/reqmgmt.html.

Woukeu, A., Carr, L., Wills, G. and Hall, W. (2003). Rethinking Web Design Models: Requirements for Addressing the Content. Technical Report ECSTR-IAM03 -002, Department of Electronics and Comnputer Science, University of Southampton.

Yu, E. (1993). Modeling Organizations for Information Systems Requirements Engineering. In: Proc. of the First International Symposium on Requirements Engineering, RE'93, San Jose, USA.

TRAMA: A Traceability Analysis Method for (Interactive) Applications

ANNEXES

A dissertation presented by
Giovanni Randazzo

Supervised by
Prof. Paolo Paolini

Submitted to the
Faculty of Communication Sciences
University of Lugano

For the degree of
Ph.D. in Communication Sciences

October 2005

ANNEX 1
Educational Material

TRAMA: Traceability Analysis Method for (interactive) Applications

TRAMA take-away

Giovanni Randazzo

23 slides

Traceab... what?

- Traceability is the degree to which a relationship can be established between two or more products of the development process [IEEE, 1990]
- Forward traceability – What is the impact of each requirement on the application?

Requirements → Design

- Backward traceability – What is the motivation of the presence of each design element in the application?

Requirements ← Design

TRAMA: a traceability analysis method

- TEC-Lab, Unversità della Svizzera Italiana, Lugano (Switzerland)
- HOC – Hypermedia Open Center, Politecnico di Milano (Italy)
- Design traceability method supporting both backward and forward traceability
- Provides to designer an effective tool conceived to discuss and analyse the design choices after they have been taken, in order to refine it according to the main requirements and in order to eliminate unmotivated elements

TRAMA: purposes (1)

- Compliance checking
  - Check the compliance of design elements with requirements, understanding if a particular design element answers to one or more stakeholders' needs
- Requirements and Design "tuning"
  - Correct and refine requirements and design according to strategic goals
- Reverse requirements engineering
  - Understand requirements from design and motivations.
- Impact analysis
  - Evaluate the impact of a requested change

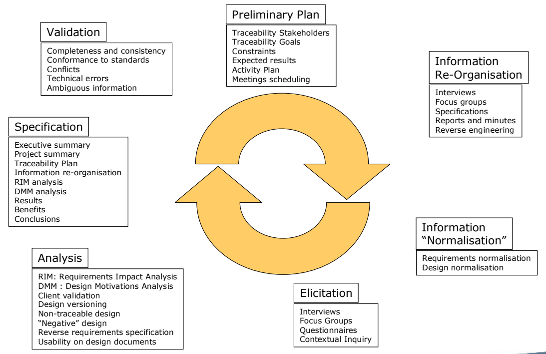
TRAMA: purposes (2)

- "Negative" design tracing
  - Keep trace of choices that for any reason have been rejected or eliminated from the application
- Solutions patterns
  - Keep a library of effective need-solution pairs
- Workflow management
  - For a better control of the project global picture
- Communication
  - Keep the "fil rouge" of the decisions taken during the project and provide to clients arguments and evidences of the project quality in terms of satisfaction of goals, needs and expectations

TRAMA: main concepts

- **The design of the application solutions may not derive directly from requirements refinement**
- **Designing is an intuition and induction process more than a derivation one**
- **The TRAMA tracing activity tries to move intuition and induction to more rational cause-effect motivations**
- **The method forces to better make explicit requirements that are both implicit or unexpressed**
- **TRAMA is based on structured matrices that cross requirements with design in a forward direction and design with its sources (requirements, motivations, constraints, etc.) in a backward direction**

TRAMA Analysis Process



Preliminary plan

- Who are the traceability stakeholders?
- Which are the goals that this activity is intended to reach?
- Which are the time and budget constraint related to this phase?
- Which are the expected results of the traceability analysis?
- A setup of the tracing activity is also needed.

Information re-organisation

- It is needed to have structured and ordered elements both form the requirements and from the design side
- In the real world:
  - The requirements specification may be unstructured or incomplete
  - The requirements specification may be absent
  - The design documentation may be absent or incomplete
- Re-organise = Understanding requirements and design from documents or from interviews with designers and organise it in terms of structured specifications

Information "normalisation"

- Structuring requirements and design information in "normal" terms, base on a strong methodology
- A normal form is a representative element within an equivalence class, which is a simples or most manageable or otherwise tidiest and most desirable form, in terms of structure or syntax
- Requirements normalisation
  - requirements information should be transformed in a more manageable form in order to be traced towards the design
  - e.g. AWARE
- Design normalisation
  - Transforming the design knowledge gathered during the information re-organisation activity in terms of structured design
  - e.g. IDM

Elicitation and analysis

- Elicitation: surfacing relationships between requirements and design in terms of
  - impact of requirements on the design ("How did you considered this requirements in the design?")
  - motivations for design choices ("Why did you adopted this solution?").
- Analysis
  - taking all the information surfaced by the different elicitation practices performed (interviews, focus groups, etc.)
  - gathering all this knowledge in a structured and analytical picture

RIM – Requirements Impact Matrix

- Lists vertically requirements-related information and horizontally all the design elements
- Highlights the impact on design of visions, stakeholders goals, users motivations and requirements
- Vertically
  - information on how single requirements are taken into account into the design
  - "Taking into account a single design element, how does it fit with requirements?"
- Horizontally
  - how a single design element satisfy the project requirements
  - "Taking into account a single requirements, how has it considered in the design?"

		DESING ELEMENTS			
		Content 1	Content 2	Access path 1	...
REQUIREMENTS-RELATED INFORMATION	VISIONS				
	Vision 1				
	Vision 2				
	...				
	GOALS				
	Goal 1				
	Goal 2				
	...				
	REQUIREMENTS				
	Requirement 1				
	Requirement 2				
	....				

Requirements-related information

- Visions
  - correspond to a strategic insight of a stakeholder in the domain
  - provide a way for modelling the assumptions of a stakeholder which dictate her “weltanschaung” on the project
- Goals
  - a wished state of affairs for the main stakeholders
  - a wished experience or an expectation for a class of users
- Requirements
  - sufficiently high-level descriptions of properties or functionalities of the application as input for the design activity

TRAMA take-away 13

Design elements

- Conceptual elements
  - the traditional conceptual design elements
  - content and structure of content, navigation architecture, access paths, operations, pages and layout, etc.
- Contextual settings
  - e.g. the technical equipment, the place where the application is used, the physical disposition of machines in this place, etc.
- Organisational elements
  - e.g. how different use sessions are organised during a week, which activities are implies in the use of the application, etc.
- Other accessorial elements
  - e.g. study material needed to use the application (in a educational system), etc.

TRAMA take-away 14

DMM – Design Motivations Matrix

- Lists vertically the design elements of the application and horizontally their kind of motivations
- It is not just the opposite of RIM
- “Negative” design elements can be also listed in this matrix
- Horizontally: “Why did you adopted this solution?”, “Why did you rejected this solution?”

		DESING MOTIVATIONS						
		Visions	Goals/Requirements	Designer expertise	Understanding of the domain	Contraints	Law obbligations	Arbitrary choices
DESIGN ELEMENTS	Content 1							
	Content 2							
	Access path 1							
	...							
	Negative design element 1							
	Negative design element 2							
	...							

TRAMA take-away 15

Design sources

- The designer expertise
  - i.e. particular “good design” principles that are part of the designer’s skills and that she/he applies in any case
- A specific understanding of the domain
  - i.e. recurrent good solutions in a domain that the designer applies because she/he learnt it by other cases in the same domain
- A particular constraint
  - e.g. budget limitations, time, technology limitations, etc.
- A law obligation
  - e.g. copyright issues, personal data treatment, etc
- A requirements-related information, i.e. a vision, a goal, a requirements, etc
- An arbitrary choice
  - i.e. a choice without particular reasons, usually a single detail that should anyway be set in a way or another, e.g. the structure of a game in three steps (instead of four or two).

TRAMA take-away 16

Specification and validation

- Specification:
  - present all the results in a structured document
  - documenting stakeholders, goals and analysis results.
- Validation
  - checking the results with requirements analysts, designers, project managers and clients

TRAMA take-away 17

Example: Munch in Berlin

- Web site for the “Munch un Berlin exhibition” at the *Staatliche Museen zu Berlin* (Germany)
- It represents the first practical result of the WED approach based upon a linguistic approach considering the interaction of a user with a web site as a dialogue
- The web site is optimized for visually impaired people, where the interaction is more natural, like in an oral dialogue

TRAMA take-away 18

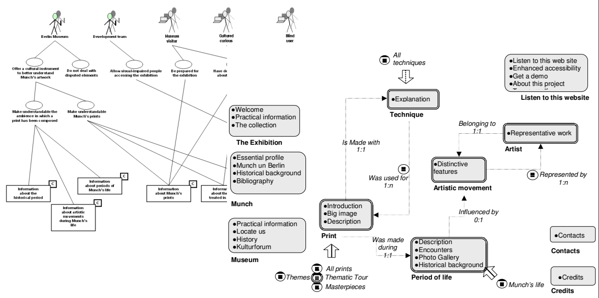
Munch: Preliminary Plan

- Before to put the application on-line, a consistency check have been requested to "adjust" the last elements and to fix an up-to-date documentation of the overall project.
- TEC-Lab performed a first traceability analysis focusing on the conciseness and on the understandability of the documentation to provide.
- A further traceability phase has been conducted in February 2005 to cope with new and refined project goals
  - design a website which might work also as a fixed information kiosk in the museum
  - make the website more usable by visually-impaired users (refining the WED approach)
  - promote knowledge and awareness about a temporary exhibition being hosted at the Museum (Munch's prints and drawings).
- Traceability was here performed to evaluate the impact of changing requirements and of proposed new solutions on the application.

TRAMA take-away 19

Munch: information normalisation

- With AWARE and IDM



TRAMA take-away 20

Munch: elicitation and analysis (RIM)

DESIGN ELEMENTS	CONTENTS	Prints	Techniques	Periods of life	Artistic movements	Artists	Munch	The museum	The exhibition	Contacts	Credits	Listen to this Website	RELATIONSHIPS BETWEEN CONTENTS	From a print to the technique it was made with	From a technique to the prints made with that technique	From a print to the period of life during which the print was made	From a period of life to the artistics movements actives in Germany during that period	From an artistic movement to the artists that represent the movement	From an artist to the artistic movement he belonged to	ACCESS PATHS TO CONTENTS	A list of all the prints	A group of masterpieces	Some thematic tour of prints grouped by theme	A list of Munch's periods of life	A list of all the techniques
VISIONS																									
Frame works of art within historical background				X	X	X										X	X	X						X	
MOTIVATIONS																									
Be prepared for visiting the exhibition		X	X				X	X	X	X				X	X						X	X	X		X
Study Munch and his art		X	X	X	X	X	X							X	X	X	X	X	X		X	X	X	X	X
Appreciate the artworks in the exhibition		X	X									X													
GOALS																									
Design site & kiosk																									
Make site accessible		X										X													
Raise awareness on Munch's prints		X						X	X													X	X		
Provide leading themes				X	X	X										X	X	X						X	
Raise awareness on art. movement					X												X	X							
Raise awareness on polit. background				X	X	X										X	X	X						X	
Understanding what's interesting in the exhibition																						X	X		
Finding information about the paintings in the exhibition		X	X											X	X										
Finding information about Munch's life				X			X																	X	
Finding historical information about Munch				X	X	X	X									X	X	X	X					X	
Finding detailed information about Munch's work and art		X	X											X	X						X	X	X		X
Understanding the relevance of Munch in the history of art					X	X											X	X	X						
Finding information about the techniques used in the paintings			X											X											X
Efficiently accessing the exhibition's topics		X																				X	X		
Understanding the site structure and the browsing capabilities												X													

TRAMA take-away 21

Munch: elicitation and analysis (DMM)

CHOICES SOURCE	Visions	Motivations	Goals	Specific understanding of the domain	Designer expertise	Technology constraints	"Graphic" constraints	Budget constraints	Laws obligations
CONTENTS									
Prints	X	X	X		X				
Techniques		X	X						
Periods of life		X	X						
Artistic movements	X		X						
Artists	X		X						
Munch	X		X						
The museum		X							
The exhibition		X	X						
Contacts					X				
Credits					X				
Listen to this Website			X						
RELATIONSHIPS BETWEEN CONTENTS									
From a print to the technique it was made with					X				
From a technique to the prints made with that technique					X				
From a print to the period of life during which the print was made					X				
From a period of life to the artistic movements actives in Germany during that period	X								
From an artistic movement to the artists that represent the movement					X				
From an artist to the artistic movement he belonged to					X				
ACCESS PATHS TO CONTENTS									
A list of all the prints					X				
A group of masterpieces		X	X						
Some thematic tour of prints grouped by theme			X	X					
A list of Munch's periods of life					X				
A list of all the techniques					X				

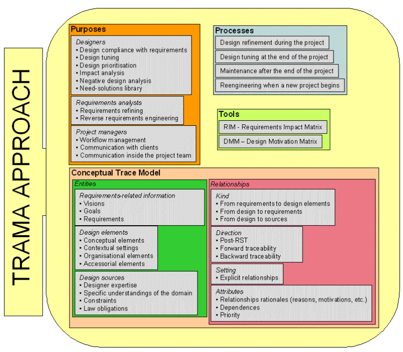
TRAMA take-away 22

TRAMA: pros and cons

- **Benefits**
  - **a powerful communication mean to show to the clients that all their requirements have been considered and how, and that there are not unmotivated elements in the design;**
  - **a structured practice to check design consistency for revision;**
  - **an advanced tool to tune up design in maintenance phase;**
  - **a complete project knowledge summary of requirements, of design elements and of relationships between them, as vital information allowing an effective system reengineering**
- **Limits**
  - **Maintenance problems**
  - **Solution: the requirement watcher**

TRAMA take-away 23

Wrap-up



TRAMA take-away 24

TRAMA: Traceability Analysis Method for (interactive) Applications

TRAMA crash course

Giovanni Randazzo

1 day / 100 slides

Summary

- AM (08:00-12:30)
  - Traceability foundations
  - Approaches and examples
  - TRAMA: basic concepts
  - TRAMA activity workflow
  - Information re-organisation and normalisation
  - Elicitation and analysis
  - Examples
- PM (13:30-18:00)
  - Q & A
  - Group Work
  - Group Presentations

Summary

- AM (08:00-12:30)
  - **Traceability foundations**
  - Approaches and examples
  - TRAMA: basic concepts
  - TRAMA activity workflow
  - Information re-organisation and normalisation
  - Elicitation and analysis
  - Examples
- PM (13:30-18:00)
  - Q & A
  - Group Work
  - Group Presentations

Traceab... what?

- Traceability is
  - the degree to which a relationship can be established between two or more products of the development process [IEEE, 1990]
  - the ability to explicitly trace and document relationships between the different phases of a project's life-cycle
- Requirements Traceability
  - is the ability to determine which documentation entities of a software system are related to which other documentation entities according to specific relationships
  - helps ascertain how and why system development products satisfy stakeholder requirements

Forward traceability

Requirements → Design

- What is the impact of each requirement on the application?
- Forward to requirements
  - maps stakeholder needs, visions and goals to the requirements, so that the analyst can determine the impact to requirements as needs change
- Forward from requirements
  - assigns responsibility for fulfilling a requirement to the design or to the various system components that will implement it, letting the responsible ensure that each requirement is fulfilled

Backward traceability

Requirements ← Design

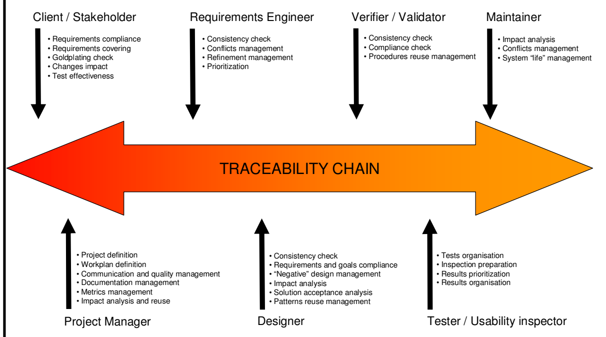
- What is the motivation of the presence of each design element in the application?
- Backward from requirements
  - lets the analyst verify that the system meets the user community's needs
  - allow to understand the source of requirements
- Backward to requirements
  - verifies compliance of design, software or tests built to requirements

Summary

- AM (08:00-12:30)
  - Traceability foundations
  - **Approaches and examples**
  - TRAMA: basic concepts
  - TRAMA activity workflow
  - Information re-organisation and normalisation
  - Elicitation and analysis
  - Examples
- PM (13:30-18:00)
  - Q & A
  - Group Work
  - Group Presentations

TRAMA crash course 7

Traceability approaches: purposes



TRAMA crash course 8

Traceability approaches: processes

- Define "entities"
  - elicit and define with stakeholders the objects to keep related each other, e.g. requirements, design elements, test procedures, etc.
- Capture traces
  - trace the relationships between the different elements of the trace model.
- Analyse traces
  - interpret the relationships and highlight problems or weaknesses raised out from traceability
- Represent traces
  - provide tools, procedures, checklists, etc. helping stakeholders and analysts in document, illustrate and display the traceability knowledge; summarise the results in a traceability report
- Maintain traces
  - keep tracing information up-to-date as far as new decisions are taken or any change is made to the system status.

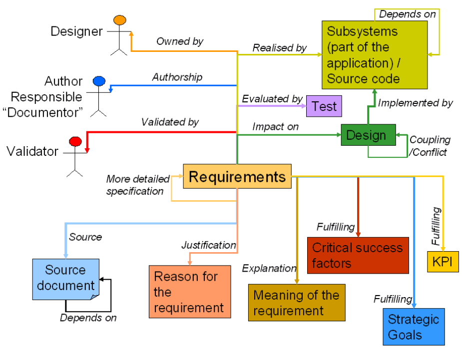
TRAMA crash course 9

Traceability approaches: entities

- Kind
  - Requirements
  - Goals
  - Design elements
  - Classes
  - Code
  - Test cases
- Attributes
  - effort
  - priority (determined by the customer)
  - source
  - status
    - proposed/approved/designed/incorporated/validated
    - captured/specified/planned/realised
    - new/assigned/classified/selected/applied/rejected
    - optional/mandatory/deleted/desirable

TRAMA crash course 10

Traceability approaches: relationships



TRAMA crash course 11

Tools

- Conceptual tools
  - Traceability matrices
  - Cross-references
  - ER models
  - Graphical models
  - Tracing languages
- Software tools
  - General-purpose tools
  - Special-purpose tools
  - Workbenches
  - Environments and beyond

TRAMA crash course 12

Examples

- Contribution structures
  - define relationships between a project artefact and its author/contributor/responsible
- PRO-ART
  - a tool-based requirements engineering environment
  - the model tries to identify relationships between requirements and application architecture on the base of scenarios
- CBPS
  - Component, Bus, System, Property approach
  - helps refining requirements to an initial architecture, supports development with evolving requirements and architecture and facilitates the elicitation of architectural information out of requirements
- The Potts and Bruns model
  - a generic model for representing design deliberations and the relation between deliberations and the generation of method-specific artefacts
  - delineates the generic elements of software design rationale, such as artefacts, issues, positions, justifications, and the relations among them

Summary

- AM (08:00-12:30)
  - Traceability foundations
  - Approaches and examples
  - **TRAMA: basic concepts**
  - TRAMA activity workflow
  - Information re-organisation and normalisation
  - Elicitation and analysis
  - Examples
- PM (13:30-18:00)
  - Q & A
  - Group Work
  - Group Presentations

TRAMA: a traceability analysis method

- TEC-Lab, Unversità della Svizzera Italiana, Lugano (Switzerland)
- HOC – Hypermedia Open Center, Politecnico di Milano (Italy)
- Design traceability method supporting both backward and forward traceability
- Provides to designer an effective tool conceived to discuss and analyse the design choices after they have been taken, in order to refine it according to the main requirements and in order to eliminate unmotivated elements

TRAMA: main concepts

- **The design of the application solutions may not derive directly from requirements refinement**
- **Designing is an intuition and induction process more than a derivation one**
- **The TRAMA tracing activity tries to move intuition and induction to more rational cause-effect motivations**
- **The method forces to better make explicit requirements that are both implicit or unexpressed**

TRAMA: purposes for designers

- Compliance checking
  - in order to check the compliance of design elements with requirements
  - with this method one can understand if a particular element of the design answers to one or more stakeholders' needs
- Design "tuning"
  - one can base on compliance checking and on design motivations analysis to correct and refine or to reengineer the design according to strategic goals
- Design prioritisation
  - in order to evaluate the relative weight and the effort request for a design element according to requirements compliance, simplifying or enriching that element
- Impact analysis
  - one can evaluate the impact of a requested change, analysing dependences between design elements and requirements, constraints, visions, etc.
- "Negative" design tracing
  - in order to keep trace of choices that for any reason have been rejected or eliminated from the application, avoiding to discuss again these solutions in future development of the project
- Solutions patterns
  - one can keep a library of effective need-solution pairs

TRAMA: purposes for Requirements analysts

- Requirements refinement
  - in order to refine the requirements specification
  - some requirements are sometimes let implicit or they are not recorded in a document, even if they are not obvious or trivial ones
  - key requirements that have not been explicitly discussed in the analysis phase may surface in a TRAMA analysis considering design motivations
- Reverse requirements engineering
  - if requirements have not been documented in previous analysis phases, the requirements analyst can use the TRAMA information for a reverse engineering activity, understanding requirements from design and motivations

TRAMA: purposes for project managers

- Workflow management
  - for a better control of the overall project
  - the method provides a global picture of the entire project, highlighting the relationships between its different pieces and the reasons why those decision were formerly taken
  - these elements are crucial to organize efficiently the time plan, giving priorities to the development of the core elements of the application and avoiding useless or superfluous features
- Communication with clients
  - TRAMA is a huge communication mean with clients, providing to project managers arguments and evidences of the project quality in terms of satisfaction of goals, needs and expectations
- Communication inside the project team
  - TRAMA is a powerful communication tool for project managers and for designers that work on different elements of the application
  - while each designer develops a single application feature, a wider understanding of how requirements and educational goals are considered in the design is needed to refine and improve these solutions
  - in order to keep the "fil rouge" of the decisions taken during the project, understanding which elements cannot be modified and which ones may be altered in the revision process
- Documentation "tuning"
  - the TRAMA analysis allows complete and refined documentation and specifications
  - the traceability chain provides a preferential way to order, link and organize each document or deliverable

TRAMA crash course 19

TRAMA: processes (1)

- Refinement
  - a first design attempt is usually shaped by designer taking in account requirements as background information
  - in this phase, solutions are not conceived in a full explicit way and requirements are not considered one by one
  - this practice is not necessarily negative, since the results are often quite good in relation to stakeholders needs
  - TRAMA may be applied during the project after a first design has been produced
  - TRAMA tries to trace *ex-post* relationships surfacing motivations for design choices
  - the method may be applied to understand the explicit or implicit, conscious or unconscious reasons for solutions proposed in the design
  - these traces help designers in the refinement activity, i.e. in adjusting the first design attempt according to requirements and priorities in a explicit and structured way

TRAMA crash course 20

TRAMA: processes (2)

- Tuning
  - Even if the main design effort is done in the design phase, some adjustment must be always be performed during the entire project's life-cycle because of
    - technology limitations
    - business or resources constraints
    - new requirements
    - changes in requirements
  - TRAMA helps designer in the tuning activity
  - It keeps traces of old design solutions and of reasons for changes
  - Design to requirements relationships allow designers to understand the impact of a changed requirement into the application

TRAMA crash course 21

TRAMA: processes (3)

- Maintenance
  - After an application production project is ended, a continuous maintenance activity is needed to "keep alive" the application through the years
  - In particular, a real life use of the product by the final users and its effects on the organisation and on the business of the company, make clear if all the solution proposed were actually good and effective solutions
  - If this is not the case or if some changes occurs in the company (e.g. new constraints, new requirements, etc.), the application needs to be revised, updated or changed
  - TRAMA helps project managers and designers in adjusting weak solutions or in conceiving new solutions for old or new needs, understanding the impact of these changes on the overall application and on its compliance with requirements

TRAMA crash course 22

TRAMA: processes (4)

- Reengineering
  - A new project in order to insert major modifications in the application or in order to develop a new application
  - Because of
    - new needs
    - new requirements or
    - more in general, new relevant elements for the company
  - In these cases, the old application may be simply tuned-up or completely reengineered
  - TRAMA helps to understand or to remember why certain solutions have been formerly adopted even if some years have passed and if there is a new project team
  - TRAMA allows also to organise the redesign activity according to old dependencies with requirements, identifying the elements that can be improved and the "untouchable" elements linked to still valid goals and that should not be changed

TRAMA crash course 23

TRAMA analysis tools

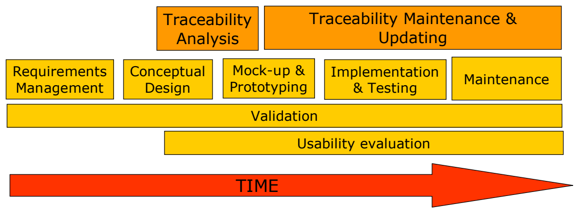
- TRAMA is based on traceability matrices
  - cross requirements with design in a forward direction
  - cross design with its sources (requirements, motivations, constraints, etc.) in a backward direction
- RIM (Requirements Impact Model/Matrix)
  - Requirements-to-Design matrix
  - can be filled and read horizontally, highlighting how single requirements are taken into account into the design
  - can be filled and read vertically, showing how a single design element satisfies the project requirements
- DMM (Design Motivations Model/Matrix)
  - Design-to-Sources matrix
  - traces back single design elements to the motivation why a certain decision is relevant for the project
  - e.g. satisfying a requirements, fulfilling a constraint, allowing more usability in the system, etc.

TRAMA crash course 24

Traceability phase

- In which moment of the project life-cycle?
- relevant information can be surfaced after a first version of the design is produced
- a detailed design is possibly needed to profitably trace relationships towards high-level requirements
- Suggestion:
  - perform a tracing activity after the first design phase
  - a continuous activity during the rest of the project is then needed to maintain the traceability specification up-to-date

Traceability in the project's life-cycle



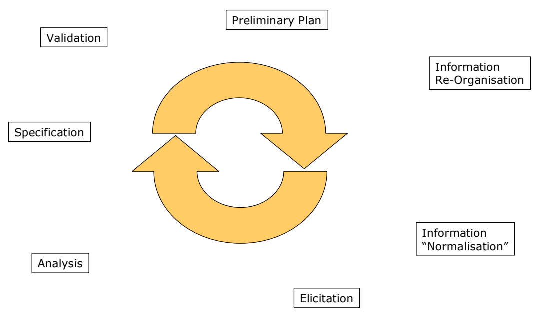
Summary

- AM (08:00-12:30)
  - Traceability foundations
  - Approaches and examples
  - TRAMA: basic concepts
  - **TRAMA activity workflow**
  - Information re-organisation and normalisation
  - Elicitation and analysis
  - Examples
- PM (13:30-18:00)
  - Q & A
  - Group Work
  - Group Presentations

TRAMA components

- The approach consists in
  - a structured analysis process
  - a general conceptual model of entities and relationships to trace
  - a set of conceptual tools supporting traces inquiry, analysis and documentation

TRAMA Analysis Process



TRAMA workflow (1)

- Preliminary plan
  - understanding which the stakeholders of the traceability analysis, the traceability goals, the constraints (time and budget, related to ROI) and the expected results are
- Information re-organisation
  - understanding requirements and design from documents or from interviews with designers and organise it in terms of structured specifications
- Information "normalisation"
  - structuring requirements and design information in "normal" terms, base on a strong methodology (e.g. AWARE for requirements and IDM for design)

TRAMA workflow (2)

- Elicitation
  - surfacing relationships between requirements and design in terms of impact of requirements on the design (“How did you considered this requirements in the design?”) and of motivations for design choices (“Why did you adopted this solution?”).
- Analysis
  - tracing relationship and developing the Requirements Impact and the Design Motivations Matrices (RIM and DMM).
- Specification
  - documenting stakeholders, goals and analysis results
- Validation
  - checking the results with requirements analysts, designers, project managers and clients.

Summary

- AM (08:00-12:30)
  - Traceability foundations
  - Approaches and examples
  - TRAMA: basic concepts
  - TRAMA activity workflow
  - **Information re-organisation and normalisation**
  - Elicitation and analysis
  - Examples
- PM (13:30-18:00)
  - Q & A
  - Group Work
  - Group Presentations

Information re-organisation (1)

- TRAMA aims at discovering relationships between requirements and design and between design and its motivation
- To clearly discuss about this relationships with stakeholders and to avoid misunderstandings, it is needed to have structured and ordered elements both form the requirements and from the design side
- In a perfect world
  - requirements information are explicitly organised and recorded during the project analysis phase
  - this specification is continuously updated during the project development
  - design is step-by-step documented in formal schemes
  - it is always keep aligned with the actual application implemented

We do not live in a perfect world... (1)

- The requirements specification may be unstructured or incomplete
  - the penetration degree of requirements management approaches in industrial practices is very low
  - in most of the cases, unstructured and informal approaches are used to record the information raised out form the firsts operative meetings
  - sometimes there is not a clear and univocal perception of what a “requirement” or what a “goal” is
    - technical details of the applications?
    - high-level visions related to a topic?
    - business-related expectations?
    - application-related desires?

We do not live in a perfect world... (2)

- The requirements specification may be absent
  - In the worst cases, the requirements specification is not only confused or unstructured, but completely absent
  - In some projects the first recorded sign of what goals and requirements were, is the description of how the application is made
  - In frequent cases, the requirements specification is not used as a base to design the application, but it is an ex-post documentation used to describe the backgrounds of an existing product

We do not live in a perfect world... (3)

- The design documentation may be absent or incomplete
  - Often it is not clear what a design for an interactive application is
    - it should describe all the technical implementation details?
    - it should be a conceptual picture of the applications contents, functionalities, navigation, etc.?
  - Sometimes this kind of specification is completely absent
    - Just a technical documentation of how the application has been programmed is provided
  - In other cases, an unstructured specification of the elements of the application design is produced
    - but it includes a mix of indistinct contents, operations, navigation capabilities, organisation elements, roles, etc.

Information re-organisation (2)

- Sometimes, of course, requirements and design specification are recorded with scientific and formal approaches
- Anyway, the TRAMA method cannot take this eventuality for granted but it should consider all the possibilities that can be encountered in the real world
- TRAMA can therefore be applied anyway, no matter if there is previous documentation or not.

- Information re-organisation consists in understanding requirements and design information before to start the tracing process
- The traceability analysts has somehow to understand what the goals, the requirements, etc. of the project and what the designed contents, functionalities, etc. of the application were
- She/he has to "pick up" and to organise these elements in a requirements specification and in a design document.

37

Information sources

- Specific interviews or focus groups with requirements, analysts, designers, project responsible or other members of the work-team

- Existing documents, specifications, reports, minutes or annotations of some project meeting or activity

- A reverse engineering activity, extracting the design form the actual application or (more difficult) inferring requirements from the design

38

Information "normalisation"

- The knowledge gathered during the information re-organisation activity can be documented pragmatically using no matter what approach
- The approach adopted have to answer to the needs of clarity, simplicity and correctness in terms of information structure

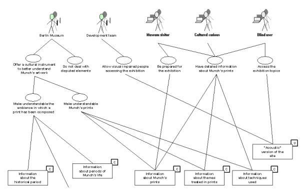
- Normalisation = Structuring requirements and design information in "normal" terms, base on a strong methodology
- A normal form is a representative element within an equivalence class, which is a simples or most manageable or otherwise tidiest and most desirable form, in terms of structure or syntax

- TRAMA distinguish between the sub-activities of
  - requirements normalisation and
  - design normalisation.

39

Requirements normalisation

- Structuring the requirements-related information in a "normal" form
- Requirements information should be transformed in a more manageable form in order to be traced towards the design
- A goal-oriented methodology is suggested
  - structure the knowledge in terms of goals, goals refinement and requirements
- e.g. AWARE



40

AWARE: a definition

- AWARE: Analysis of Web Application Requirements

- A goal-oriented methodology supporting the requirements analysis and requirements documentation for web projects

- Representation and understanding of relevant website stakeholders and their goals is key element for successful design

41

AWARE is

- Stakeholder-centered
  - Websites are made by people for people
- Goal-oriented
  - High-level objectives come before the solutions
- Scenario-based
  - Reflection on contexts of use help requirements surface
- Project-driven
  - Goals and domain knowledge is mediated within the scope of the project
- Tool-independent
  - Flexible notation not constrained by a proprietary platform
- Web-specific
  - but extendable to other domains

42

AWARE: general concepts

- Stakeholder
- Goal
- Goal Refinement
- Requirement
- Scenario

Stakeholders

- Those who have a direct interest in the success of the website are called stakeholders.
- Stakeholders may include the users, the clients who finance the web site, and other people involved in the project (e.g. sponsor, developers, and representatives of the organization departments, etc.).
- Stakeholders are either individuals or placeholders for an organization's or institution's interests.
- They may be "typed" (e.g. the secretary) or "single" (e.g. the director of bank x)

Goals

- A stakeholder may own one or more goals with respect to the website-to-be.
- A goal is defined as a high-level target of achievement for a stakeholder.
- It may represent a wished state of affairs for the main stakeholders ("Increase customer loyalty"), but also a wished experience or an expectation for a class of users ("Find suitable funds").
- Goals vary in abstraction level and granularity.

Goal refinement

- Goals are analysed by decomposing them into subgoals, according to an ad-hoc refinement process
- The refinement process consists in:
  - Detailing the goals
  - Deciding which and how upper goals may be satisfied - according to the constraints, the obstacles met and resource available – and highlight possible alternatives
  - Defining requirements contributing to accomplish the goals
- The refinement process is mainly top-down but highly iterative

Requirements

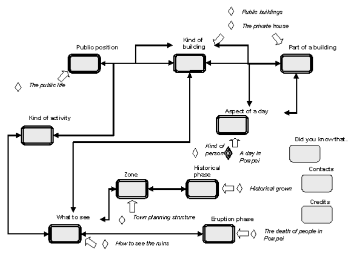
- The outcome of the goal decomposition is a set of requirements, which represent the actual input for the design activity
- A requirement is a sufficiently high-level descriptions of the a property or functionality of the website meaningful for one or more stakeholders (e.g. "provide up-to-date fund information")
- Requirements address a variety of design dimensions (content, navigation, access, operations, etc.)

Scenarios

- The elicitation and refinement process may be supported by envioisioning salient episodes of use of the website, called scenarios (e.g. "an enrolled student looks for information about a specific course he is not attending….")
- Scenarios can help uncover overlooked stakeholders, surface and exemplify goals and requirements, justify, validate or invalidate decisions
- Scenarios provoke stakeholders to reflect on requirements in view of more concrete and vivid artifacts (e.g. pieces of design, prototypes, stories)

Design normalisation

- Transforming the design knowledge gathered during the information re-organisation activity in terms of structured design
- e.g. IDM

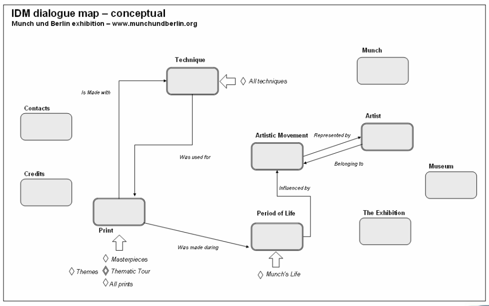


IDM

- Interactive Dialogue Model
- A dialogue-based design model to shape interactive applications
- Can represent both sketched ideas or fully developed solutions
- The graphic representation of these structures is very readable, compact and expressed in a conceptually simple way
- Easy to use for brainstorming
- Good as elicitation tool
- Tailored to master multichannel applications

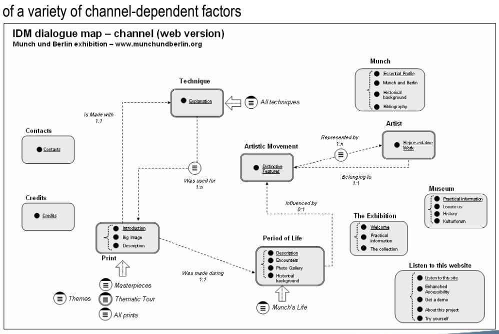
Conceptual design (C-IDM)

- A conceptual schema, of an interactive application, must convey all the necessary "dialogue strategies", without (and before) digging into details depending on technical issues



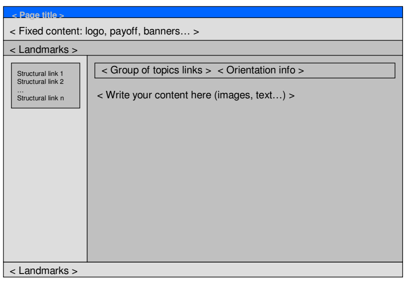
Logical Design (L-IDM)

- It can be seen as a detailed version of the conceptual design, where details are decided on the basis of a variety of channel-dependent factors



Page design (P-IDM)

- Defining the elements to be communicated to the user in a single dialogue act
- Crafting the actual pages containing the necessary elements to sustain the dialogue



Summary

- AM (08:00-12:30)
  - Traceability foundations
  - Approaches and examples
  - TRAMA: basic concepts
  - TRAMA activity workflow
  - Information re-organisation and normalisation
  - **Elicitation and analysis**
  - Examples
- PM (13:30-18:00)
  - Q & A
  - Group Work
  - Group Presentations

Elicitation

- Elicitation is to call forth or draw out as information or a response something latent or potential
- Elicitation is the process of identifying needs and bridging the disparities among the involved communities for the purpose of defining and distilling requirements to meet the constraints of these communities
- Elicitation is the activity of surfacing relationships between requirements and design in terms of impact of requirements on the design ("How did you considered this requirements in the design?") and of motivations for design choices ("Why did you adopted this solution?")

Elicitation techniques

- Create an environment where stakeholders feel at their ease and are able to demonstrate ideas
- Combine different techniques:
  - Interviews
  - Focus groups
  - Questionnaires
  - Direct observation

Interviews (1)

- Very common for this kind of activity because they allow a "live" contact with a person that could be a source of information
- Here everything depends on the interviewer's skills and on the right selection of people to talk with
- In large projects where many people are involved, this activity could take a lot of time

Interviews (2)

- Benefits
  - When "few" people each know a "Lot"
  - Gather RICH information
  - Insights about stakeholder's perspectives
  - Insights about the culture and the domain
- Tips
  - Allow people showing material, examples and demonstrating their ideas
  - Trade-off between listening, guiding and intrusion
- Drawbacks
  - Time consuming
  - Miss interaction between stakeholders

Focus groups (1)

- Discussion meetings between the traceability expert and the project's work-team
- It is possible a "live" contact with people working on the project
- It is not so focused as in an interview
- New knowledge may raised out from group discussion
- A single meeting or a couple of meetings do not take so much time

Focus groups (2)

- Benefits
  - New knowledge from discussions and interaction
  - Good both for brainstorming and focus groups
  - Everybody need to explain ideas for other to understand
- Tips
  - 3-20 stakeholders in one room
  - Analysty offers issues and questions
  - Every one should feel accepted and involved in
- Drawbacks
  - Difficult to fit in the stakehoders' agenda
  - Only "public" opinion emerge
  - Risk to be conflict-driven

Questionnaires (1)

- Can be used as a preliminary step in focus groups or interviews
- To set up the discussion agenda
- Where too much people are involved in the project
  - interviews for the two or three project main responsible
  - questionnaires for all the other project workers

Questionnaires (2)

- Benefits
  - Quantify and compare data
  - Large sample at low cost
  - Appear scientific due to statistical data
- Tips
  - Should be short
  - Alternate open and close questions
- Drawbacks
  - No time for explanation, solve misunderstanding and provoke "habit change"
  - No human touch
  - Focussed aswers to specific questions only
  - Short time causes poor reflection and knowledge evocation

Direct observation (1)

- Following the entire project form the beginning can be an option in case of high budgets and large projects
- Here a traceability expert follow the different project's phases as an internal observer and debrief step-by-step the motivations why the application is designed in a certain way
- This technique presupposes many time and resources to be performed

Direct observation (2)

- Benefits
  - Stakeholders are observed while doing their job
  - Insight about actual process, work context and time
  - Elicit tacit knowledge and automatic processes
- Tips
  - Be as passive as possible
- Drawbacks
  - **Hawthorne effect**: people aware of being watched act differently than they do when unobserved

Meetings set-up

- Place
  - A large meetings room with a table and some chairs
  - The room should have some free walls in order to hang up the papers with the matrices
- Tools
  - Coloured pencils
  - Blackboard/flipcharts/Papers
  - Self-stick wall pads
- Roles
  - the discussants, e.g. in focus groups the project work-team that animate the meeting
  - a facilitator, i.e. a traceability expert in charge to address the discussion in a right direction, provoking answers, asking critical questions, etc.
  - a wall writer, drawing the matrices on the wall papers and filling the crosses with the traceability information raised out from discussion
  - a secretary, recording and writing notes (on a PC) about the meeting
  - a chair officer, e.g. the project manager coordinating the overall meeting

Some biases in elicitation

- Cognitive biases
- Overconfidence
- Faulty reasoning
- Communication problems
- Motivational biases

Cognitive biases

- **Easy of recall**: events that are vivid and emotional or happened recently are easier to recall by the stakeholders, but they are not actually likely to occur.
- Stak. "it is very important that the user might be able to find that information"
- User "I really liked the home page of that site"
- Strategy:
  - Directed questions:
    - "how many timed does it happen in the last month?"
    - "what if the same goals is achieved by different means"?
    - "Why" questions

67

Overconfidence

- **Overconfidence**: Analysts are optimistic about their understanding of stakeholders' goals. Requirements gathering process risk terminating too soon.
- An. "…I see what you need, that is enough for me"
- Strategy:
  - Scenario reflection: revealing knowledge being used rather than assumed
  - Direct prompting: using the ideas of another stakeholders as counter-arguments for causing reflection
  - What other kind of solution could you imagine?
  - "why questions"

68

Faulty reasoning

- **Faulty reasoning**: stakeholders might do illogical inferences in supporting their beliefs.
- "In the site, products must be organized by storing categories because our product catalogue – as you can see – is organized in this way. Also our supplier presents information by similar categories, so…"
- Strategy:
  - Devil's advocate
  - Scenario reflection

69

Communication problems (1)

- Different Background
  - tech vs manag
- Different Domain Knowledge
  - ad extra – ad intra
- Different Language
  - system specific vs domain specific
- Different Goals
  - efficiency and easy of maintainance vs maximum functionality

70

Communication problems (2)

- Strategy: Pre-elicitation conditioning
  - Discuss the purpose of the meeting
  - What the analyst will be asking
  - What stakeholder will need to provide
  - Explain key terms
  - Explain how information will be used
  - Making stakeholders aware of potential biases

71

Motivational biases (1)

- Stakeholders are unwilling to provide accurate requirements because:
  - Organizational policy
    - Fear of being evaluated by others
    - Don't know who will know what they say
    - Fear of offending someone or break balances
    - Self-protection, self-preservation
    - Bias on domains of other stakeholders
  - Don't know what analyst needs
  - Don't know other stakeholders already met

72

Motivational biases (2)

Strategy: Pre-elicitation conditioning
- Explain how information elicited will benefit both
- Explain how information elicited will be used
- State that everyone's opinion is valued
- Tell other stakeholders already met
- Assure responses are kept confidential

73

Analysis

- Taking all the information surfaced by the different elicitation practices performed (interviews, focus groups, etc.)
- Gathering all this knowledge in a structured and analytical picture
- Different points of view have to be integrated:
  - The designer's point of view
  - The client/customer's point of view

74

The designer's point of view

- Each designer develop different parts and different functionalities of a same application
- His/her perception of the project is often limited to a "vertical" view on how these parts and functionalities answers to the strategic needs
- The traceability analysis have to gather all these partial views, showing how the entire application fits with requirements through the inter-action of its different parts.

75

The client/customer's point of view

- Often this point of view is mediated by the project manager
- The focus here is how a single requirement has been taken into account in the application development
- The analyst have therefore to consider all the information gatherer from an "horizontal" point of view, documenting the impact that all the strategic needs (expressed by goals and requirements) have on the application design

76

TRAMA analysis aspects

- Designer's and client's points of view are mirrored in two aspects taken into account by the TRAMA analysis:
  - the justification or motivation of the design (designer's point of view), that can comes from requirements or from other kind of sources (an understand of the specific domain, the expertise of the designer, a constraint, etc.);
    - these traces are called Design Motivations Model (DMM)
  - the impact on design (client's point of view) of: visions, stakeholders-goals, users-motivations, domain issues, scenarios, constraints and requirements
    - these traces form the Requirements Impact Model (RIM)

77

RIM – Requirements Impact Matrix

- Lists vertically requirements-related information and horizontally all the design elements
- Highlights the impact on design of visions, stakeholders goals, users motivations and requirements
- Vertically
  - information on how single requirements are taken into account into the design
  - "Taking into account a single design element, how does it fit with requirements?"
- Horizontally
  - how a single design element satisfy the project requirements
  - "Taking into account a single requirements, how has it considered in the design?"

		DESING ELEMENTS			
		Content 1	Content 2	Access path 1	...
REQUIREMENTS-RELATED INFORMATION	VISIONS				
	Vision 1				
	Vision 2				
	...				
	GOALS				
	Goal 1				
	Goal 2				
	...				
	REQUIREMENTS				
	Requirement 1				
	Requirement 2				
	....				

78

Requirements-related information

- Visions
  - correspond to a strategic insight of a stakeholder in the domain
  - provide a way for modelling the assumptions of a stakeholder which dictate her "weltanschaung" on the project
- Goals
  - a wished state of affairs for the main stakeholders
  - a wished experience or an expectation for a class of users
- Requirements
  - sufficiently high-level descriptions of properties or functionalities of the application as input for the design activity

Design elements

- Conceptual elements
  - the traditional conceptual design elements
  - content and structure of content, navigation architecture, access paths, operations, pages and layout, etc.
- Contextual settings
  - e.g. the technical equipment, the place where the application is used, the physical disposition of machines in this place, etc.
- Organisational elements
  - e.g. how different use sessions are organised during a week, which activities are implies in the use of the application, etc.
- Other accessorial elements
  - e.g. study material needed to use the application (in a educational system), etc.

DMM – Design Motivations Matrix

- Lists vertically the design elements of the application and horizontally their kind of motivations
- It is not just the opposite of RIM
- "Negative" design elements can be also listed in this matrix
- Horizontally: "Why did you adopted this solution?", "Why did you rejected this solution?"

		DESING MOTIVATIONS						
		Visions	Goals/Requirements	Designer expertise	Understanding of the domain	Contraints	Law obbligations	Arbitrary choices
DESIGN ELEMENTS	Content 1							
	Content 2							
	Access path 1							
	...							
	Negative design element 1							
	Negative design element 2							
	...							

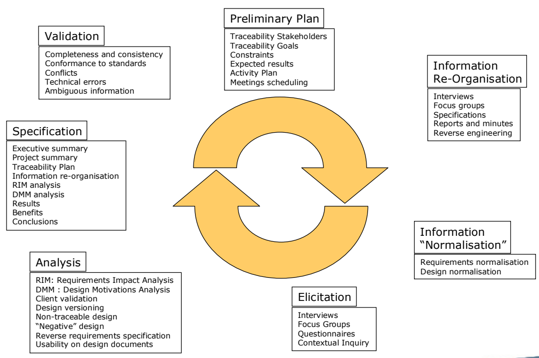
Design sources

- The designer expertise
  - i.e. particular "good design" principles that are part of the designer's skills and that she/he applies in any case
- A specific understanding of the domain
  - i.e. recurrent good solutions in a domain that the designer applies because she/he learnt it by other cases in the same domain
- A particular constraint
  - e.g. budget limitations, time, technology limitations, etc.
- A law obligation
  - e.g. copyright issues, personal data treatment, etc
- A requirements-related information, i.e. a vision, a goal, a requirements, etc
- An arbitrary choice
  - i.e. a choice without particular reasons, usually a single detail that should anyway be set in a way or another, e.g. the structure of a game in three steps (instead of four or two).

Specification and validation

- Specification:
  - present all the results in a structured document
  - documenting stakeholders, goals and analysis results.
- Validation
  - checking the results with requirements analysts, designers, project managers and clients

TRAMA Analysis Process



Summary

- AM (08:00-12:30)
  - Traceability foundations
  - Approaches and examples
  - TRAMA: basic concepts
  - TRAMA activity workflow
  - Information re-organisation and normalisation
  - Elicitation and analysis
  - **Examples**
- PM (13:30-18:00)
  - Q & A
  - Group Work
  - Group Presentations

#	Time	Hours	Activity type	Topic
1	15	08:00-08:15	Lecture	Course intro & traceability foundations
	15	08:15-08:30	Lecture	Traceability approaches and examples
	60	08:30-09:30	Lecture	TRAMA basic concepts
	30	09:30-10:00	Lecture	TRAMA workflow 1: information re-organisation
	30	*10:00-10:30*		*Coffee break*
	30	10:30-11:00	Lecture	TRAMA workflow 2: information normalisation
	60	11:00-12:00	Lecture	TRAMA workflow 2: Elicitation and analysis
	30	12:00-12:30	Lecture	TRAMA examples
	60	*12:30-13:30*		*Launch break*
	30	13:30-14:00	Lecture	Wrap-up + Class questions & answers
	120	14:00-16:00	Tutoring	Group tutoring
	30	*16:00-16:30*		*Coffee break*
	90	16:30-18:00	Presentations	Group analysis short presentations

Example 1: Munch in Berlin

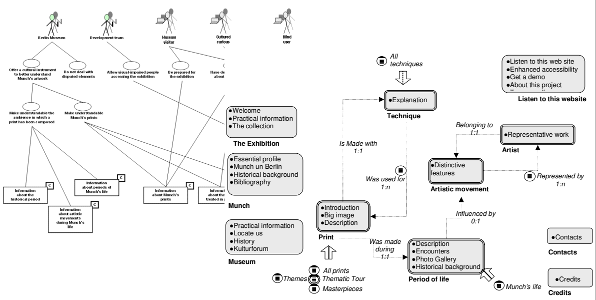
- Web site for the "Munch un Berlin exhibition" at the *Staatliche Museen zu Berlin* (Germany)
- It represents the first practical result of the WED approach based upon a linguistic approach considering the interaction of a user with a web site as a dialogue
- The web site is optimized for visually impaired people, where the interaction is more natural, like in an oral dialogue

Munch: Preliminary Plan

- Before to put the application on-line, a consistency check have been requested to "adjust" the last elements and to fix an up-to-date documentation of the overall project.
- TEC-Lab performed a first traceability analysis focusing on the conciseness and on the understandability of the documentation to provide.
- A further traceability phase has been conducted in February 2005 to cope with new and refined project goals
  - design a website which might work also as a fixed information kiosk in the museum
  - make the website more usable by visually-impaired users (refining the WED approach)
  - promote knowledge and awareness about a temporary exhibition being hosted at the Museum (Munch's prints and drawings).
- Traceability was here performed to evaluate the impact of changing requirements and of proposed new solutions on the application.

Munch: information normalisation

- With AWARE and IDM



Munch: elicitation and analysis (RIM)

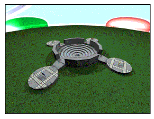
DESIGN ELEMENTS	CONTENTS	Prints	Techniques	Periods of life	Artistic movements	Artists	Munch	The museum	The exhibition	Contacts	Credits	Listen to this Website	RELATIONSHIPS BETWEEN CONTENTS	From a print to the technique it was made with	From a technique to the prints made with that technique	From a print to the period of life during which the print was made	From a period of life to the artistics movements actives in Germany during that period	From an artistic movement to the artists that represent the movement	From an artist to the artistic movement he belonged to	ACCESS PATHS TO CONTENTS	A list of all the prints	A group of masterpieces	Some thematic tour of prints grouped by theme	A list of Munch's periods of life	A list of all the techniques
VISIONS																									
Frame works of art within historical background				X	X	X										X	X	X						X	
MOTIVATIONS																									
Be prepared for visiting the exhibition		X	X				X	X	X	X				X	X						X	X	X		X
Study Munch and his art		X	X	X	X	X	X							X	X	X	X	X	X		X	X	X	X	X
Appreciate the artworks in the exhibition		X	X									X													
GOALS																									
Design site & kiosk																									
Make site accessible		X										X													
Raise awareness on Munch's prints		X						X	X													X	X		
Provide leading themes				X	X	X										X	X	X						X	
Raise awareness on art. movement					X												X	X							
Raise awareness on polit. background				X	X	X										X	X	X						X	
Understanding what's interesting in the exhibition																						X	X		
Finding information about the paintings in the exhibition		X	X											X	X										
Finding information about Munch's life				X			X																	X	
Finding historical information about Munch				X	X	X	X									X	X	X	X					X	
Finding detailed information about Munch's work and art		X	X											X	X						X	X	X		X
Understanding the relevance of Munch in the history of art					X	X											X	X	X						
Finding information about the techniques used in the paintings			X											X											X
Efficiently accessing the exhibition's topics		X																				X	X		
Understanding the site structure and the browsing capabilities												X													

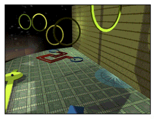
Munch: elicitation and analysis (DMM)

CHOICES SOURCE	Visions	Motivations	Goals	Specific understanding of the domain	Designer expertise	Technology constraints	"Graphic" constraints	Budget constraints	Laws obligations
CONTENTS									
Prints	X	X	X		X				
Techniques		X	X						
Periods of life		X	X						
Artistic movements	X		X						
Artists	X		X						
Munch	X		X						
The museum		X							
The exhibition		X	X						
Contacts					X				
Credits					X				
Listen to this Website			X						
RELATIONSHIPS BETWEEN CONTENTS									
From a print to the technique it was made with					X				
From a technique to the prints made with that technique					X				
From a print to the period of life during which the print was made					X				
From a period of life to the artistic movements actives in Germany during that period	X								
From an artistic movement to the artists that represent the movement					X				
From an artist to the artistic movement he belonged to					X				
ACCESS PATHS TO CONTENTS									
A list of all the prints					X				
A group of masterpieces		X	X						
Some thematic tour of prints grouped by theme			X	X					
A list of Munch's periods of life					X				
A list of all the techniques					X				

Example 2: Learning @ Europe

- Educational project aiming at fostering the development of a "European Identity"
- Educational approach novel in several respects
  - advanced content
  - technology-enhanced e-Learning
  - multicultural experience
  - engaging "games"
  - cultural competition





L@E: preliminary plan (1)

- In a first full experimentation year, between 2004 and 2005, 48 classes from 6 European countries (Belgium, France, Italy, Norway, Poland and Spain), nearly 60 teachers and 1,000 students were involved
- A new advanced experimentation year, between 2005 and 2006, will bring the project at an industrial stage. Before this new experimentation, a complete revision of the whole setting of the experience will be performed.
- A traceability analysis has been requested to facilitate this revision activity

L@E: preliminary plan (2)

- Goals
  - reorganize the complex and various material describing and designing the experience
  - pave the grounds for a reengineering activity

  - internal communication, to communicate the project status to all the team members
  - reverse requirements engineering, re-organizing and refining requirements and surfacing missing information, fundamentals to understand the project but never explicitly documented
  - design tuning, surfacing missing design components and re-aligning the design with the project state-of-the-art
  - design revision, to facilitate the project revision before a new experimentation period

L@E: information "normalisation" (Goals)

- General goals
  - Offering to schools a collaborative learning experience based on new technologies
  - Basing the experience on historical contents
  - Basing the experience on a multicultural approach
  - Allowing the educational impact to be measurable
  - Allowing to participate classes and pupils of every level and kind, not only the best classes in the best schools
  - Minimizing the internal management costs of the experience
- Educational goals
  - Knowledge (i.e. teaching a "know what" to students)
    - About local (national) history
    - About other countries' history
    - About general historical concepts and processes
  - Skills (i.e. teaching a "know how" to students)
    - Use of "professional" English (as a tool to work)
    - Use of technological tools for synchronous or asynchronous collaboration (3D worlds, forums, online communities, etc.)
    - Group work (face to face collaboration)
    - Collaborative work (remote collaboration)
  - B3 Attitudes (i.e. provoke an habit change to students)
    - Sense of curiosity for history
    - National identities are the result of a process: multiple cultures / multiple identities
    - Improved attitude towards history
    - Critical thinking towards knowledge: truth appears through a variety of opinions
    - Different attitude towards knowledge, different learning modality (e-learning)

L@E: information "normalisation" (Visions)

- Integration in schools' curricula
  - Convenient quantity of commitment
    - For students. The project aims at support schools as they are and not to subvert the internal organisation; the experience have to involve an entire class (12 to 25 students) and have to be guided by teachers with active and directive roles. The project may help teachers in managing different class segments.
    - For teachers. The project aims do not include that teachers learn something about technology. The experience does not base on teachers' technological skills.
  - Convenient use of infrastructural resources. The project must not requests to school an excessive use of laboratories or a too sophisticated technological equipment.
  - The educational benefits have to be related with the general educational goals of schools and of their curricula. Teachers must be able to justify the time and organisational effort spent for participating in this experience.
- Characteristics of and educational competition
  - It has to be a motivation for students in learning; it has to be a "true" competition and repay the commitment. Therefore the competition should be:
    - Open: motivation should remain active for everyone until the end, also for micro-sessions
    - Serious: it should repay different skills and valorise a deep understanding but it should not be frustrating
  - It has not to be frustrating: participants should not be demotivated by difference of results with the others. This characteristic have to be balanced with the previous one.
  - Engaging but not an end in itself; e.g. the access to cultural questions (the "serious" part) could be win with games involving "physical" or technical skills (the engaging part).

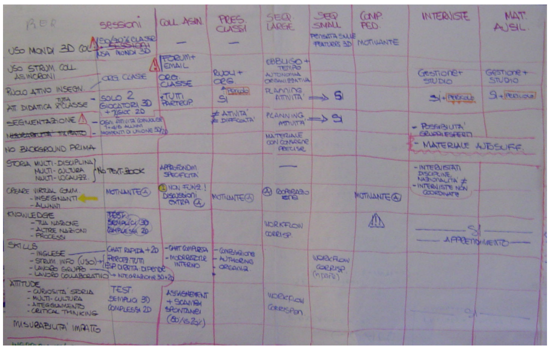
L@E: information "normalisation" (Requirements)

- The experience have to include the use of collaborative 3D worlds
- The experience have to include the use of tools for asynchronous collaboration
- The experience have to include the teachers' active role
- The educational activities have to involve the whole class
- The activities have to be modularized in order to facilitate class segmentation
- The activities must require to students a minimum background knowledge
- The activities must not presuppose that teachers know how to use technologies
- The applications must allow to participate with a low technology level and include a degraded mode of use for low connections
- The historical contents have to highlight multiple opinions, disciplines, localizations and cultures involved in the topic
- The experience have to support the creation of a virtual communities of students, also after the end of the project
- The experience have to support the creation of a virtual communities of teachers, also after the end of the project

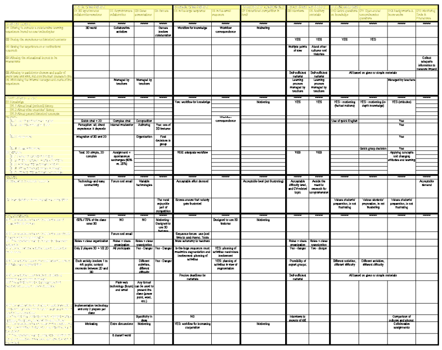
L@E: information "normalisation" (Design)

- Static components
  - 3D synchronous collaborative sessions
  - Asynchronous collaboration (forum/email)
  - Class presentations
  - Games
- Dynamic components
  - In-the-large sequence: succession of sessions, asynchronous sessions and off-line activities during the experience
  - In-the-small sequence : succession of the activities, contents and tests in a session
- Transversal components
  - Educational competition in itself
- Educational materials
  - Interviews (extended and simplified)
  - Auxiliary materials
- Testing materials
  - Quick questions on knowledge, "matter of fact" about local history, about other countries' history and about general historical concepts
  - Open-ended comprehension questions about local history, about other countries' history and about general historical concepts
  - Assignments & home-works (to apply the knowledge)
  - Monitoring Tools & Procedures

L@E: Meetings



L@E: elicitation and analysis (RIM)



L@E: elicitation and analysis (RIM)

	STATIC COMPONENTS				DYNAMIC COMPONENTS
	D1 3D synchronous collaborative sessions	D2 Asynchronous collaboration	D3 Class presentations	D4 Games	D5 In-the-large sequence
EDUCATIONAL GOALS					
B1 Knowledge					Yes: workflow for knowledge
B1.1 About local (national) history					
B1.2 About other countries' history					
B1.3 About general historical concepts					
B2 Skills					
B2.1 Use of "professional" English	Quick chat + 2D	Complex chat	Composition		
B2.2 Use of technological tools	Perception: all; direct experience: it depends	Internal moderator	Authoring	Yes: use of 3D features	
B2.3 Group work	Integration of 3D and 2D		Organization	Fast decisions in group	
B2.4 Collaborative work					
B3 Attitudes	Test: 3D simple, 2D complex	Assignment + spontaneous exchanges (80% vs. 20%)			YES: adequate workflow
B3.1 Sense of curiosity for history					
B3.2 Multiple cultures / multiple identities					
B3.3 Improved attitude towards history					
B3.4 Critical thinking towards knowledge					
B3.5 Different attitude towards knowledge					
VISIONS					
V1 Integration in schools' curricula	Technology and easy connectivity	Forum and email	Variable technologies		Acceptable effort demand
V2 Characteristics of and educational competition				The most enjoyable part of competition	Scores ensure that nobody gets frustrated
REQUIREMENTS					
R01 The experience have to include the use of collaborative 3D worlds	50% / 70% of the class uses 3D	NO	NO	Motivating - Designed to use 3D features	
R02 The experience have to include the use of tools for asynchronous collaboration		Forum and email			Sequence forces use (and time to use) Async. Tools

L@E: elicitation and analysis (DMM)

	Visions	Requirements	Designer expertise	Specific understanding of the domain	Constraints	Law obbligations
STATIC COMPONENTS	###############	###############	###############	###############	###############	###############
D1 3D synchronous collaborative sessions		Offering to schools a collaborative learning experience based on new technologies		In past projects this technology has been a good motivation for e-learning	Use of the in-house 3D technology	
D2 Asynchronous collaboration		Create a community	Forums and emails are common tools to build up an asynchronous collaboration			
D3 Class presentations			Just to add an activity in the first (introductory) session			
D4 Games				Another good idea from past projects as a motivation for traditional learning		
DYNAMIC COMPONENTS	###############	###############	###############	###############	###############	###############
D5 In-the-large sequence	Acceptable effort demand	Workflow adequate to educational goals				
D6 In-the-small sequence			Workflow correspondance. Each session has not to be too "boring" or long or simple/difficult, etc.			
TRANSVERSAL COMPONENTS	###############	###############	###############	###############	###############	###############
D7 Educational competition in itself	It has to be a motivation for students in learning: open and serious			In past projects it has been a good motivation for e-learning		
EDUCATIONAL MATERIALS	###############	###############	###############	###############	###############	###############
D8 Interviews			This way to present information has worked very well in past projects			
D9 Auxiliary materials			This material go with interviews as background explaination (maps, biographies, etc.)			
TESTING MATERIALS	###############	###############	###############	###############	###############	###############
D10 Quick questions on knowledge	Values students' preparation, is not frustrating					
D11 Open-ended comprehension questions						
D12 Assignments & home-works		Improve knowledge, skills and attitudes				
D13 Monitoring Tools & Procedures		The educational impact has to be measurable				

Summary

- AM (08:00-12:30)
  - Traceability foundations
  - Approaches and examples
  - TRAMA: basic concepts
  - TRAMA activity workflow
  - Information re-organisation and normalisation
  - Elicitation and analysis
  - Examples
- **PM (13:30-18:00)**
  - Q & A
  - Group Work
  - Group Presentations

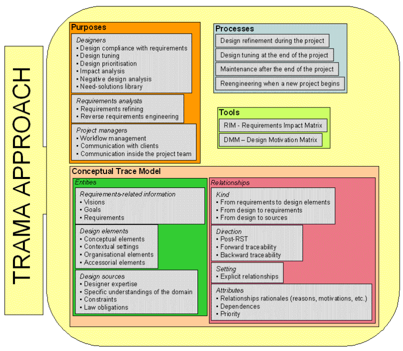
103

TRAMA: pros and cons

- **Benefits**
  - **a powerful communication mean to show to the clients that all their requirements have been considered and how, and that there are not unmotivated elements in the design;**
  - **a structured practice to check design consistency for revision;**
  - **an advanced tool to tune up design in maintenance phase;**
  - **a complete project knowledge summary of requirements, of design elements and of relationships between them, as vital information allowing an effective system reengineering**
- **Limits**
  - **Maintenance problems**
  - **Solution: the requirement watcher**

104

Wrap-up



105

TRAMA: Traceability Analysis Method for (interactive) Applications

TRAMA in practice

Giovanni Randazzo

3 days / 185 slides

Goals

- Understand benefits of requirements traceability
- Learn TRAMA, an advanced traceability method
- Shortly understand AWARE to represent requirements
- Shortly understand IDM to represent design elements
- Apply the method in practice
- Manage traceability in work-teams

Scheduling – Day 1

#	Time	Hours	Activity type	Topic
1	1h	08:00-09:00	Lecture	Course intro & traceability foundations
	1.5h	09:00-10:30	Lecture	Traceability approaches, tools and examples
		10:30-10:45		*Coffee break*
	2h	10:45-12:45	Lecture	TRAMA basic concepts
		12:45-14:00		*Launch break*
	0.5h	14:00-14:30	Lecture	Example 1: Munch in Berlin
	0.5h	14:30-15:00	Discussion	Class discussion about the example
	2h	15:00-15:30	Exercise	Simple analysis exercise (individual)
		15:30-15:45		*Coffee break*
		15:45-17:15	Tutoring	Individual tutoring
	0.5h	17:15-17:45	Lecture + Tutoring	Day wrap-up + Class quenstion & answers

Scheduling – Day 2

#	Time	Hours	Activity type	Topic
2	1h	08:00-09:00	Presentation	Presentation of the analysis exercise
	1h	09:00-10:00	Tutoring	Class discussion about the exercise
		10:00-10:15		*Coffee break*
	1h	10:15-11:15	Lecture	TRAMA workflow 1: information re-organisation & normalisation
	2.5h	11:15-12:30	Exercise	Information re-organisation and normalisation exercise (individual)
		12:30-13:30		*Launch break*
		13:30-14:45	Tutoring	Individual tutoring
	0.5h	14:45-15:15	Presentation	Presentation of the information re-organisation exercise (1)
		15:15-15:30		*Coffee break*
	1.5h	15:30-17:00	Presentation	Presentation of the information re-organisation exercise (2)
	0.5h	17:00-17:30	Lecture + Tutoring	Day wrap-up + Class q&a

Scheduling – Day 3

#	Time	Hours	Activity type	Topic
3	1h	08:00-09:00	Lecture	TRAMA workflow 2: Elicitation and analysis
	1h	09:00-10:00	Lecture	TRAMA analysis examples
		10:00-10:15		*Coffee break*
	0.5h	10:15-10:45	Lecture	TRAMA workflow 3: Specification and validation
	0.5h	10:45-11:15	Lecture	TRAMA documentation examples
	2.5h	11:15-12:30	Exercise	Advanced analysis exercise (individual)
		12:30-13:30		*Launch break*
		13:30-14:45	Tutoring	Individual tutoring
	1h	14:45-15:45	Presentation	Presentation of the analysis exercise (1)
		15:45-16:00		*Coffee break*
	1h	16:00-17:00	Presentation	Presentation of the analysis exercise (2)
	0.5h	17:00-17:30	Lecture + Tutoring	Course wrap-up + Class q&a

Summary – Day 1

- Traceability foundations
- Approaches, tools and examples
- TRAMA: basic concepts
- Example 1
- Exercise 1

Summary – Day 1

- **Traceability foundations**
- Approaches, tools and examples
- TRAMA: basic concepts
- Example 1
- Exercise 1

Traceab... what?

- Traceability is
  - the degree to which a relationship can be established between two or more products of the development process [IEEE, 1990]
  - the ability to explicitly trace and document relationships between the different phases of a project's life-cycle
- Requirements Traceability
  - is the ability to determine which documentation entities of a software system are related to which other documentation entities according to specific relationships
  - helps ascertain how and why system development products satisfy stakeholder requirements

Pre-Requirements Specification traceability (1)

- Pre-RST is concerned with those aspects of a requirement's life prior to its inclusion in the requirements specification
  - Requirements production and refinement
- It is a technique that attempts to document the rationale and socio-political context from which requirements emerge, thus linking the business world with that of information technology
- Serves to answer questions that arise during the project's life-cycle, including:
  - "Who is responsible for including this requirement?"
  - "To whom should I refer to for more information?"
  - "Who was responsible for copying this information into this document?"
  - "Was this requirement a result of a meeting of stakeholders or just one individual?"
- Pre-RST facilitates the reopening of previously closed specifications, tracing back to the sources of requirements, and then the (possible) reworking of a specification in the forward direction

Pre-Requirements Specification traceability (2)

- Sources of requirements may be the following:
  - Stakeholder Visions: stakeholders are those who have a direct interest in the success of the website (e.g. clients, sponsors, representatives, opinion makers, etc.); stakeholder visions are the assumptions of a stakeholder which dictate her "weltanshaung" on the project
  - User Motivations: they shape the emotional, psychological, social or individual elements which can trigger a person (a final user) to use an interactive application
  - Goals: they are defined as high-level targets of achievement for a user or a stakeholder; goals may represent a wished state of affairs (for main stakeholders) or a wished experience (for users) and may arise from visions or motivations
  - Constraints: they are defined as those elements that implicate a restriction on the degree of freedom the requirement analyst have in providing a solution; constraints can be economic, political, technical, or environmental and pertain to project resources, schedule, target environment, or to the system itself.

Post-Requirements Specification traceability (1)

- Post-RST is concerned with those aspects of a requirement's life that result from its inclusion in the RS
  - requirement deployment and use
- This kind of traceability provides a way to elicit and discover the impact of requirements and how requirements have been taken into account on the subsequent project elements

Post-Requirements Specification traceability (2)

- Targets of requirements may be the following:
  - conceptual design: high-level definition of the information structure, of the features and of the services/capabilities that the application will own;
  - technical design: in-detail definition of the software (and/or hardware) components the application will be made of;
  - experience design: definition of all the elements contributing in building the user experience, including organisational concerns, technical set-up and use scenarios;
  - implementation: it's the "tangible" part of the application, i.e. classes, routines, lines of code, interfaces, etc.
  - tests: including technical test verifying if the application works properly, usability tests and accessibility test.

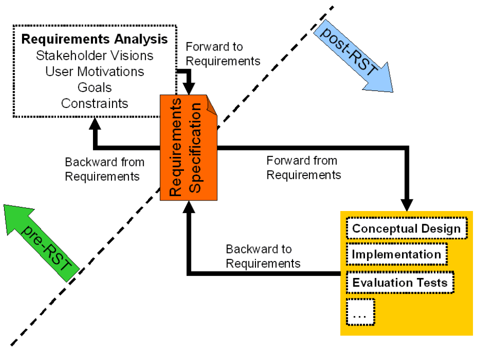
Forward traceability

- What is the impact of each requirement on the application?
- Forward to requirements
  - maps stakeholder needs, visions and goals to the requirements, so that the analyst can determine the impact to requirements as needs change
- Forward from requirements
  - assigns responsibility for fulfilling a requirement to the design or to the various system components that will implement it, letting the responsible ensure that each requirement is fulfilled

Backward traceability

- What is the motivation of the presence of each design element in the application?
- Backward from requirements
  - lets the analyst verify that the system meets the user community's needs
  - allow to understand the source of requirements
- Backward to requirements
  - verifies compliance of design, software or tests built to requirements

Traceability types



Quality of Service

- *ad intra*
  - quality is considered by mean of the intrinsic characteristics of the application (e.g. performance, accuracy, up-to-date);
- *ad extra*
  - quality is the correspondence between services offered and stakeholders' goals; it can see as the combination of the quality of the user experience, the user satisfaction and the main stakeholder' satisfaction

Software Quality

- the totality of features and characteristics of a software product that bears on its ability to satisfy given needs, for example to conform to specifications
- the degree to which software possesses a desired combination of attributes
- the degree to which a customer or user perceives that software meets his or her composite expectations
- the composite characteristic of software that determine the degree to which the software in use will meet the expectations of the customer

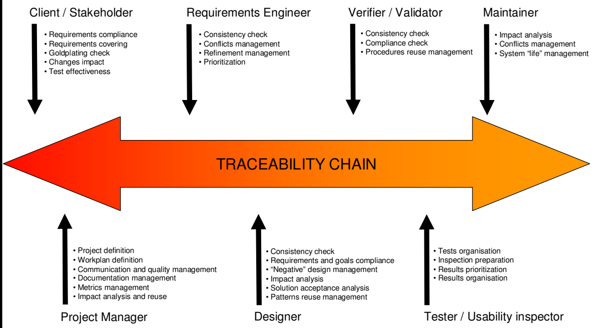
Traceability as element of SQ

- Quality is a multifaceted characteristic of an application
- The quality degree of a project may depend on
  - services and features provided
  - user satisfaction and context of use
  - customer and main stakeholders satisfaction
  - compliance with strategic goals
  - impact on the organisation
- It becomes crucial to keep in a global picture the relationships between these elements
- Traceability can improve the quality of the systems development process.

Summary – Day 1

- Traceability foundations
- **Approaches, tools and examples**
- TRAMA: basic concepts
- Example 1
- Exercise 1

Traceability approaches: purposes



Client/Customer/Stakeholder (1)

- They have a certain number of problems in evaluating the quality and the effectiveness of a software application a priori
  - before its effects have been produced
- There is a knowledge and understanding gap between stakeholders and the development team
- Clients hardly can see how and where the applications provided may fit to their needs and goals
- Traceability analysts can guide these people in evaluating such applications
- Traceability is a communication “bridge” between
  - a client (usually with marketing or economics background)
  - and a software house, a web agency or anyway the internal development team (with engineering or informatics background)

Client/Customer/Stakeholder (2)

- Requirements compliance
  - Traceability can shows the relationships between strategic goals, requirements and solutions in the application, allowing clients evaluating the compliance degree of the product with their needs
  - The overall quality of the application can be understood without any need to consider single technical or software details
- Requirements covering
  - Relationships between requirements and elements or pieces of the application may highlight the progress state of the project
  - Clients can understand which percentage of the stated requirements are met and which part of the job is completed
  - A thorough traceability analysis may also provide stakeholders that all the strategic goals have been satisfied and how the application will address to their needs
- Goldplating check
  - Goldplating is the presence of features that are not motivated by any explicit reason
  - Traceability analysis highlights goldplating by linking all the application features with their motivations

Client/Customer/Stakeholder (3)

- Changes impact
  - It is not unusual to observe that after the end of a project, clients may ask to developers further changes to the applications
  - Reasons can be identify in lack of proper needs analysis or in lack of proper communication to the client
  - Traceability analysts can help clients in evaluating the consequences of their requests, i.e. the impact of a requested change on the entire application and on the way it meets stakeholders goals
- Tests effectiveness
  - If the tracking information system records which requirements are satisfied by which parts of the implementation, and which tests must be performed to ascertain the “presence” of a requirement, then clients can better understand the value, the results and the implications of technical tests and usability evaluations
  - In addition, acceptance testing can refer directly to the user requirements being tested for, making it relevant from a stakeholder point of view

Project manager and project planner (1)

- Project definition
  - An early traceability analysis during the work definition allows project managers to control that the work team and the client have the same perception of the project
  - This includes the delivered ant the not delivered artefacts, how much does it costs, who will perform the work, how the work will be done and which benefits will be achieved
- Workplan definition, development and managing
  - Matching goals with design elements is crucial to organize efficiently the time plan, giving priorities to the development of the core elements of the application and avoiding useless or superfluous features
  - Project managers can prevent conflicts and check the progresses of the different tasks related each other, with test procedures and with the main strategic goals
  - Conflicts between requirements can be discovered earlier and unexpected product delays avoided
- Communication and quality management
  - Traceability is a powerful communication mean with clients, providing to project managers arguments and evidences of the project quality in terms of satisfaction of goals, needs and expectations

Project manager and project planner (2)

- Documentation management
  - Traceability analysis allows complete and refine documentation and specifications
  - The traceability chain provides a preferential way to order, link and organize each document or deliverable
- Metrics management
  - All the relationships traced between parts of the application, features and services on one hand, and test procedures on the other hand, becomes crucial to give to project managers quantitative data to identify trends, support decisions and as indication of the good health of the project
- Impact analysis and reuse
  - Project planners use a tracing approach to perform impact analysis
  - Requirements can be tracked to determine the impact of a required change on the entire project, on the workplan, on other feature of the application, on goals, etc.
  - Requirements not yet satisfied by the implementation can be collected, and the work to be done to satisfy these remaining requirements can be estimated
  - Future systems will have reduced development time and effort because past implementation decisions can be reused

Requirements engineer (1)

- Requirements engineers keep and elicit visions, strategic goals, constraints, user profiles, etc. from stakeholders and motivations, user goals, etc. from users
- A pre-requirements specification traceability analysis is needed to keep these relationships between stakeholders and goals between users and goals, between goals and sub-goals in the refinement process and between sub-goals and requirements

Requirements engineer (2)

- Consistency check
  - Traceability analysis is used by requirements engineers to keep the consistency between the different information they consider, ad in particular between requirements as indications for the design from one hand and goals and constraints as source and motivations for requirements form the other hand
- Conflicts management
  - Conflicts between goals are usual, in particular between stakeholders goals and user goals
  - Traceability helps the analyst in finding a good compromise between conflicting goals, considering the relevance of stakeholders that own such goals and evaluating the impact that changes may have on other goals, sub-goals or requirements

Requirements engineer (3)

- Refinement management
  - During goals refinement activities it is crucial to keep all the relationships between high-level goals and derived or refined sub-goals
  - Traceability may also help in keeping an history of all the refinement changes performed in different moments of the project life-cycle and for different reasons (technology changes or constraints, budget constraints, timing, etc.)
- Prioritization
  - The traceability chain links as in a flow, stakeholders with goals and requirements
  - If all the relations are kept and updated, the requirements analyst can give a relative priority to each requirement or to groups of requirements that meet the needs of a certain stakeholder; requirements related to more relevant stakeholder should be considered with higher priority respect to others

Designer (1)

- Designers of software products are responsible to shape the information architecture of the application, considering the content structure, transitions between pieces of contents, interactive features, access to contents and features and navigation architecture
- To keep the consistency of the entire project, designers take in considerations goals and requirements highlighted during the requirements analysis
- Nevertheless, a major part of the final design has other motivations than requirements:
  - e.g. some elements could have pure technical reasons or being just based on "good design" principles
- Usually, part of these reasons are not recorded and part are not explicitly perceived or understood
- A traceability analysis allows eliciting hidden or unconscious knowledge and helps designers to show that the elements indicated in the conceptual design are not unusual, unnecessary or unmotivated

Designer (2)

- Consistency check
  - A tracking information system should record the results of design, the justification of the results, alternatives considered, and the assumptions made in a decision; therefore a traceability analysis prevents from consistency problems between different parts of the project and may help in solve inconsistencies with technical implementation or with strategic goals
- Requirements and goals compliance
  - Designers use traceability to understand dependencies between the requirements and to check whether all requirements are considered by the design
  - Therefore, they can more easily verify that a design satisfies the requirements or not. If a design element is not directly liked to a specific requirement, they can find arguments in traceability documents to justify their decisions in a more general relation with strategic goals or with non-functional requirements
  - A traceability approach force designer to ask themselves the "why" question before it is put by the client

Designer (3)

- "Negative" design management
  - With "negative" design I refer to the design elements that for any reason have been rejected or eliminated from the application
  - In most of the cases, the knowledge of which are these elements and why they have been deleted is crucial to measure their impact on the project
  - Traceability analysis support designers in keeping these kind of "design history", avoiding time-consuming features that for the same reasons would be rejected and considering alternate solutions for other similar cases
- Impact analysis
  - Traces between the different elements of a project allow designers to evaluate possible consequences for changing a design feature in terms of compliance with requirements and goals or in terms of needed changes in implemented prototypes and applications
  - From another point of view, designers can understand the impact on the design of a change in requirements and take consequent decisions
  - Designers can use traceability information also to estimate the impact of a change in available implementation technology on the design assumptions and hence on the design alternatives
- Solutions acceptance analysis
  - Starting from traceability documents, designers can understand the reasons why a certain design was accepted and another rejected, even when the design was produced long time ago by a designer not present anymore
  - These reasons may relate design decisions to non-functional requirements, to unexpressed constraints or to more general stakeholder visions

TRAMA in practice 31

Designer (4)

- Patterns reuse management
  - A traceability chain relates a specific need with a certain design solution
  - If the design is accepted, such a solution can be considered as a good one at least from a stakeholder point of view
  - Therefore, designers may reuse design components for similar needs in other projects because the assumptions under which the component will work are recorded in the traceability report
  - Besides, the tracking information system may become a kind of "corporate memory", i.e. a library of solutions patterns and a way to refers to specific solutions in a fast and direct way
- Design revision
  - Traceability documents keep the knowledge about the relationships between requirements and design in a structured way
  - If there is a need to tune up or to revise a former project, designer can understand and/or remember previous decisions taken and properly "adjust" the application

TRAMA in practice 32

Verifier / validator

- Verifiers in large projects provides a further consistency check of the final application
- They base their job on traceability information to verify that all the strategic goals have been properly satisfied, all the requirements have been taken into account, design doesn't have goldplating, software meets with design specifications and the application have been properly tested
- Validators use traceability relationships between requirements and test plans to prove that the system "completely" meets the needs of the customer
- In addition, test procedures can be identified that should be rerun to validate an implemented change
- This saves test resources and allows the schedule to be streamlined.

TRAMA in practice 33

Tester / usability inspector

- Testers perform a detail evaluation of the system technical performances
  - the application should not "crack" or generate errors in any condition of use
  - they can perform their tests in a more systematic way; e.g. they can test features in relevance order or organize tests grouping features by stakeholder or by goal they meet
  - in case of problems surfaced during the tests, they can indicate which exactly are the pieces of software or the design elements to review
  - they can also suggest a priority order for these problems based on the impact they have on the satisfaction of strategic goals
- Usability inspectors are concerned with the application "easy of use"
  - they check that the declared goals can be reached by users by the mean of the application in a efficient end effective way
  - they have to taken into account high-level goals of the product, evaluating it according to its real scope
  - they can also use entire parts of the traceability analysis to plan and prepare their evaluation: in fact, inspectors need to know dependencies between user profiles, goals and features in the application to properly test the usability of that solutions
  - as for the testers case, to usability problems can be assign a priority and the inspectors can indicate on which element of the project they have an impact.

TRAMA in practice 34

Maintainer

- Maintainers "keep alive" the application
- This is particularly true for interactive and web-based applications, where being up-to-date and always adapt the communication and business channels to new user or stakeholder needs are key success factors
- Maintainers use the traceability information to decide how a required and accepted change will affect a system, i.e., which modules are directly affected and which other modules will experience residual effects
- Documenting an engineer's design rationale helps the maintainer to understand the system
- If a required change is implemented, understanding the existing solution structure helps to prevent the system from degrading
- A maintainer can this way estimate the impact of a change in requirements on other requirements, discover conflicts dependencies, estimate the impact of a change in requirements on the implementation and estimate the permissibility of a change in implementation with respect to (unchanged) requirements

TRAMA in practice 35

Traceability approaches: processes

- Define "entities"
  - elicit and define with stakeholders the objects to keep related each other, e.g. requirements, design elements, test procedures, etc.
- Capture traces
  - trace the relationships between the different elements of the trace model.
- Analyse traces
  - interpret the relationships and highlight problems or weaknesses raised out from traceability
- Represent traces
  - provide tools, procedures, checklists, etc. helping stakeholders and analysts in document, illustrate and display the traceability knowledge; summarise the results in a traceability report
- Maintain traces
  - keep tracing information up-to-date as far as new decisions are taken or any change is made to the system status.

TRAMA in practice 36

Traceability approaches: entities (1)

- Kind
  - Requirements
  - Goals
  - Design elements
  - Classes
  - Code
  - Test cases
- Direction
  - backward and forward
  - pre- and post-RST

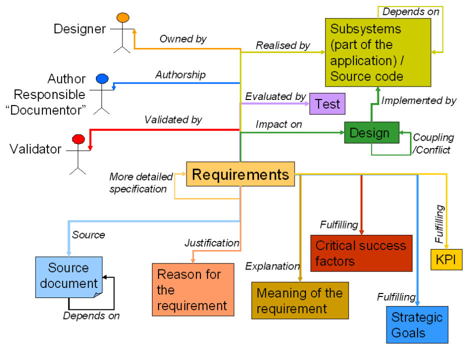
Traceability approaches: entities (2)

- Attributes
  - effort
  - priority (determined by the customer)
  - source
  - status
    - proposed/approved/designed/incorporated/validated
    - captured/specified/planned/realised
    - new/assigned/classified/selected/applied/rejected
    - optional/mandatory/deleted/desirable

Traceability approaches: entities (3)

- Setting
  - implicit relationships - links that do not require manual setting, e.g. name tracing, where if names and abbreviations are used in the same way and are meant to denote the same things in two documents, then a degree of traceability between them may be established
  - explicit relationships - they are manually implemented references between documentation entities and came from external considerations supplied by the developers; so, for example, the linkage, or relationship, between a textual requirement and a use case that describes the requirement is determined solely by the decision of the developers that such a relationship has meaning

Traceability approaches: relationships



Examples

- Contribution structures
  - define relationships between a project artefact and its author/contributor/responsible
- PRO-ART
  - a tool-based requirements engineering environment
  - the model tries to identify relationships between requirements and application architecture on the base of scenarios
- CBPS
  - Component, Bus, System, Property approach
  - helps refining requirements to an initial architecture, supports development with evolving requirements and architecture and facilitates the elicitation of architectural information out of requirements
- The Potts and Bruns model
  - a generic model for representing design deliberations and the relation between deliberations and the generation of method-specific artefacts
  - delineates the generic elements of software design rationale, such as artefacts, issues, positions, justifications, and the relations among them

Tools

- Conceptual tools
  - Traceability matrices
  - Cross-references
  - ER models
  - Graphical models
  - Tracing languages
- Software tools
  - General-purpose tools
  - Special-purpose tools
  - Workbenches
  - Environments and beyond

Conceptual tools (1)

- Traceability matrices
  - the horizontal and vertical dimension list the items that can be linked
  - the entries in the matrix represent links between these items
  - only binary links between items can be represented
  - easy to understand

43

Conceptual tools (2)

- Cross-references
  - among parts of the document
  - across different documents
  - links between documentation entities are embedded as pointers (e.g. hyperlinks or embedding phrases like "see section x") in a text
  - entities may be an informal natural language text or a formal specification
  - cross-references allow the related documents to be navigated through
  - the use is simple to understand
  - software tools that maintains cross-references and that produce reports about them can be implemented easily
  - is useful for written specifications but not for a concise representation of links such as can be done with matrices
  - cross-references are always binary links, so that links of higher arity cannot be easily represented

44

Conceptual tools (3)

- ER models
  - the linked items are entities, the links are relationship instances
  - links with arity higher than 2 can be represented
  - an ER model of links can be implemented using any database technology
    - ad hoc query and reporting facilities are easily available

45

Conceptual tools (4)

- Graphical models
  - documentation entities are represented by entities
  - relationships between them are represented by relationships
  - graphical notation (e.g. UML)

- Tracing languages
  - include DB query languages (as SQL) and regular expressions

46

Software tools (1)

- The current generation of commercially available traceability tools typically provides the following functionality:
  - storage of links between items; the items may be requirements, design items, explanations, etc. and they may be represented as fixed format database records or free format text; links may be annotated, e.g. with degree of strength;
  - storage of links between texts; the texts may be requirements, documents, design documents, etc.;
  - storage of requirements in free text format with a hierarchical numbering scheme;
  - reporting facilities; examples are keyword searches, the traversal of links, producing cross-reference lists, producing traceability matrices, etc.

47

Software tools (2)

- General-purpose tools
  - include hypertext editors, word processors, spreadsheets, database management systems and prototyping tools
  - they can be hand-configured to allow previously manual and paper-based requirements traceability tasks to be carried out on-line

- Special-purpose tools
  - A number of tools support single and well-defined activities related to requirements engineering
  - Of these, some achieve restricted types of requirements traceability
  - Although there may be a limited degree of explicit control and guidance, support is generally implicit in the use of the tool, which automates any mundane and repetitive tasks needed to provide this requirements traceability

48

Software tools (3)

- Workbenches
  - Typically centred around a database management system of some form, these software types comprise dedicated tools for documenting, parsing, editing, interlinking, organising, and managing requirements
  - They often provide facilities to help assess and carry out any changes made to these requirements
- Environments and beyond
  - Requirements traceability can potentially be provided throughout a project's life if tools supporting all aspects of development are integrated
  - The basis used for internal integration tends to define how requirements traceability is established: through the use of a common language
  - Those with the flexibility to incorporate third-party environments tend to provide requirements traceability support through the use of powerful repositories and underlying database management systems

TRAMA in practice 49

Examples (1)

- **Analyst Pro by Goda Software, Inc. (http://www.analysttool.com)**
  - Analyst Pro uses a requirements management methodology that covers the entire life-cycle including, from the initial requirements-gathering phase through the separation phase where requirements and non-requirements are set apart. Analyst Pro utilizes a Configuration Management methodology that enables the development staff to analyze the impact of change on requirements and component assets. Analyst Pro incorporates the following features:
    - Importing Requirements - Analyst Pro allows users to import requirements from existing documents from various formats (doc, html and text).
    - Requirements Sharing - Analyst Pro allows users to share and trace requirements across projects.
    - Requirements Change Management - Analyst Pro automatically records and lists any changes to your project, when the changes were made and who made the changes.
    - Requirements Assignment - Users can assign requirements to team members and track its status.
    - Requirements Graphs - Users can create pie and bar graphs with a number of requirements versus attributes. The attributes include priority, version, status and source.

TRAMA in practice 50

Examples (2)

- **CaliberRM by Borland (http://www.borland.com/caliber/index.html)**
  - Caliber-RM is a collaborative, Web-based requirements management system that facilitates communication among project teams by providing centralized requirement data to distributed team members and allowing documented discussions about requirements as well as allowing project teams to fully define, manage and communicate changing application or system requirements. Changes made to requirement data such as traceability, document references, status, user responsibility and more are recorded in Caliber-RM's central repository. CaliberRM keeps team members up to date on changes made to requirements by automatically notifying responsible individuals of the changes. CaliberRM also enables team members to quickly identify potential requirement problems by highlighting ambiguous and commonly used terms defined in a shared glossary. The latest version of CaliberRM provides LiveLink integration with Caliber-RBT so that requirements in Caliber-RM can be associated with corresponding cause-effect graph files in Caliber-RBT. CaliberRM allows project teams to provide input on requirements via standard browsers and remote clients can access the system through an Internet connection.

TRAMA in practice 51

Examples (3)

- **DOORS/ERS by Telelogic (http://www.telelogic.com/products/doorsers)**
  - DOORS (Dynamic Object Oriented Requirements System) is an Information Management and Traceability (IMT) tool. Requirements are handled within DOORS as discrete objects. Each requirement can be tagged with an unlimited number of attributes allowing easy selection of subsets of requirements for specialist tasks. DOORS includes an on-line change proposal and review system that lets users submit proposed changes to requirements, including a justification. DOORS offers unlimited links between all objects in a project for full multi-level traceability. Impact and traceability reports as well as reports identifying missing links are all available across all levels or phases of a project life cycle. Verification matrices can be produced directly or output in any of the supported formats including RTF for MS-Word, Interleaf and FrameMaker. The DOORS Extension Language (DXL) is a high level C-like language that provides access to virtually all DOORS functions for user extensions and customization. DOORS includes the following functionality:
    - Control of data model for process management allows user to manage the relationship between data fully including its direction, type and even whether a relationship is allowed.
    - Improved security control through the use of passwords, and timeouts which "lock up" DOORS after a specified period of inactivity.
    - New templates to make document generation easier have been added to the DOORS template library. New templates include ISO 12207, ISO 6592 and IEEE software standards.

TRAMA in practice 52

Examples (4)

- **IRqA (Integral Requisite Analyzer) by TCP Sistemas e Ingeniería (http://www.irqaonline.com)**
  - IRqA is a state-of-the-art Requirements Engineering (RE) tool specifically designed to provide an integral support to the complete Requirements Engineering process. In IRqA the complete specification cycle is supported via standard models:
    - Requirements Capture
    - Requirements Management
    - Requirements Analysis
    - System Specification building
    - Specification validation (specification vs requirements)
    - Acceptance Tests management
    - Requirements Organization & Classification

TRAMA in practice 53

Examples (5)

- **Rational RequisitePro by Rational Software (http://www.rational.com/products/reqpro/index.jsp)**
  - RequisitePro is a requirements management tool designed for multi-user environments. It features integration of Microsoft Word and a requirements database. Software project teams can gather, enter and manage requirements "in situ" (within your documents) or in a database. Automated traceability tracks requirements and changes through implementation and testing. Related requirements can be linked together, so that as changes occur to one requirement users can easily see its impact on other related requirements. RequisitePro includes templates to simplify production of requirements documents. Rational RequisitePro supports a choice of databases (Oracle, Microsoft SQL Server, and Microsoft Access) which allow users to organize, prioritize, and trace relationships between requirements. Version 2001A includes the ability to treat linked files as a requirement and trace other requirements to your linked files.
  - RequisitePro also provides various views to enhance traceability. One of those views is the Traceability Matrix. This matrix displays requirements in a matrix format for easier coverage viewing. The matrix will provide visual feedback about what system requirements were derived from which customer requirements. Using the matrix, it is also easy to check coverage and make sure that all of the customer requirements were broken down into system requirements. Another useful view provided by RequisitePro is the Traceability Tree view. This view shows the requirements in a hierarchical fashion. The benefit of this view is in graphically showing relationships between requirements. If a requirement is modified, added or deleted, the user can visually see all of the other affected requirements. The affected requirements can then be properly scrutinized and modified to accommodate the original requirement change. This helps maintain a cohesive set of requirements by eliminating orphaned requirements and also by preventing outdated requirements from being left in the set.
  - RequisitePro also offers cross project traceability. Often times, especially with legacy systems, a number of projects will spawn off of a central project. These new projects will share a significant number of requirements with its parent and sibling projects. RequisitePro allows traceability of requirements to span cross-project. This greatly increases requirement reuse which can in turn foster design, code, and test reuse.

TRAMA in practice 54

Examples (6)

- **RDT (Requirements Design & Traceability) by Igatech (http://www.igatech.com)**
  - RDT supports several mechanisms to aid the user in requirements analysis and identification. These include a parser that imports text documents then identifies requirements by key words and structure. The tool provides functionality for deriving, allocating and assigning requirements and acceptance test procedures. Requirements can be traced from top level requirements down to the lowest level requirements. The tool is able to classify/categorize requirements during identification using requirements attributes. In addition the tool provides capabilities to capture architecture, functional decomposition and WBS in graphical format and display data as a tree view of requirements. RDT is able to generate documentation directly into MS Word, including requirements and test specifications, requirement allocation matrices, parent-child relationships and design documents. New features incorporated in version 3 include:
    - The ability to share data between different sites, and the facility to collate this data back to the master database.
    - Revision control, which allows users to look at all changes made to data, and when and by whom these changes were made.
    - An RDT AxiomSys Bridge exists that allows the bi-directional transfer of requirements and tests between any part of the project database in RDT, and the software or system model(s) in AxiomSys 6.0.

55

Examples (7)

- **RTM (Requirements Traceability Management) by Integrated Chipware Inc. (http://www.chipware.com)**
  - RTM supports multiple users working on the same requirements at the same time by implementing locking control on a requirement-by-requirement basis. RTM's toolset supports the ability to capture graphical information as traceable requirements objects. The tool utilizes the native tool, which created the graphics object. A class definition tool is included that allows the user to model any type of hierarchical project data (requirement document, hierarchies, system element structure and WBS). Once the hierarchy is defined generic relationships can also be established to allow cross-reference link information to be established between any active data item. Version 5.3 of RTM includes the following capabilities:
    - An information modelling capability allows users to design change records or problem reports and associate them with specific requirements data.
    - A complete test management solution including information concerning schedules, resources, test verification and results versus requirements.
    - User defined forms to allow users to view information in familiar layouts.
    - Change request capability allows users to propose and review changes to the current baseline requirements from within RTM.

56

Summary – Day 1

- Traceability foundations
- Approaches, tools and examples
- **TRAMA: basic concepts**
- Example 1
- Exercise 1

57

TRAMA: a traceability analysis method

- TEC-Lab, Unversità della Svizzera Italiana, Lugano (Switzerland)
- HOC – Hypermedia Open Center, Politecnico di Milano (Italy)

- Design traceability method supporting both backward and forward traceability

- Provides to designer an effective tool conceived to discuss and analyse the design choices after they have been taken, in order to refine it according to the main requirements and in order to eliminate unmotivated elements

58

TRAMA: main concepts

- **The design of the application solutions may not derive directly from requirements refinement**
- **Designing is an intuition and induction process more than a derivation one**

- **The TRAMA tracing activity tries to move intuition and induction to more rational cause-effect motivations**
- **The method forces to better make explicit requirements that are both implicit or unexpressed**

59

TRAMA: purposes for designers

- Compliance checking
  - in order to check the compliance of design elements with requirements
  - with this method one can understand if a particular element of the design answers to one or more stakeholders' needs
- Design "tuning"
  - one can base on compliance checking and on design motivations analysis to correct and refine or to reengineer the design according to strategic goals
- Design prioritisation
  - in order to evaluate the relative weight and the effort request for a design element according to requirements compliance, simplifying or enriching that element
- Impact analysis
  - one can evaluate the impact of a requested change, analysing dependences between design elements and requirements, constraints, visions, etc.
- "Negative" design tracing
  - in order to keep trace of choices that for any reason have been rejected or eliminated from the application, avoiding to discuss again these solutions in future development of the project
- Solutions patterns
  - one can keep a library of effective need-solution pairs

60

TRAMA: purposes for Requirements analysts

- Requirements refinement
  - in order to refine the requirements specification
  - some requirements are sometimes let implicit or they are not recorded in a document, even if they are not obvious or trivial ones
  - key requirements that have not been explicitly discussed in the analysis phase may surface in a TRAMA analysis considering design motivations
- Reverse requirements engineering
  - if requirements have not been documented in previous analysis phases, the requirements analyst can use the TRAMA information for a reverse engineering activity, understanding requirements from design and motivations

TRAMA in practice 61

TRAMA: purposes for project managers

- Workflow management
  - for a better control of the overall project
  - the method provides a global picture of the entire project, highlighting the relationships between its different pieces and the reasons why those decision were formerly taken
  - these elements are crucial to organize efficiently the time plan, giving priorities to the development of the core elements of the application and avoiding useless or superfluous features
- Communication with clients
  - TRAMA is a huge communication mean with clients, providing to project managers arguments and evidences of the project quality in terms of satisfaction of goals, needs and expectations
- Communication inside the project team
  - TRAMA is a powerful communication tool for project managers and for designers that work on different elements of the application
  - while each designer develops a single application feature, a wider understanding of how requirements and educational goals are considered in the design is needed to refine and improve these solutions
  - in order to keep the “fil rouge” of the decisions taken during the project, understanding which elements cannot be modified and which ones may be altered in the revision process
- Documentation “tuning”
  - the TRAMA analysis allows complete and refined documentation and specifications
  - the traceability chain provides a preferential way to order, link and organize each document or deliverable

TRAMA in practice 62

TRAMA: processes (1)

- Refinement
  - a first design attempt is usually shaped by designer taking in account requirements as background information
  - in this phase, solutions are not conceived in a full explicit way and requirements are not considered one by one
  - this practice is not necessarily negative, since the results are often quite good in relation to stakeholders needs
  - TRAMA may be applied during the project after a first design has been produced
  - TRAMA tries to trace *ex-post* relationships surfacing motivations for design choices
  - the method may be applied to understand the explicit or implicit, conscious or unconscious reasons for solutions proposed in the design
  - these traces help designers in the refinement activity, i.e. in adjusting the first design attempt according to requirements and priorities in a explicit and structured way

TRAMA in practice 63

TRAMA: processes (2)

- Tuning
  - Even if the main design effort is done in the design phase, some adjustment must be always be performed during the entire project’s life-cycle because of
    - technology limitations
    - business or resources constraints
    - new requirements
    - changes in requirements
  - TRAMA helps designer in the tuning activity
  - It keeps traces of old design solutions and of reasons for changes
  - Design to requirements relationships allow designers to understand the impact of a changed requirement into the application

TRAMA in practice 64

TRAMA: processes (3)

- Maintenance
  - After an application production project is ended, a continuous maintenance activity is needed to “keep alive” the application through the years
  - In particular, a real life use of the product by the final users and its effects on the organisation and on the business of the company, make clear if all the solution proposed were actually good and effective solutions
  - If this is not the case or if some changes occurs in the company (e.g. new constraints, new requirements, etc.), the application needs to be revised, updated or changed
  - TRAMA helps project managers and designers in adjusting weak solutions or in conceiving new solutions for old or new needs, understanding the impact of these changes on the overall application and on its compliance with requirements

TRAMA in practice 65

TRAMA: processes (4)

- Reengineering
  - A new project in order to insert major modifications in the application or in order to develop a new application
  - Because of
    - new needs
    - new requirements or
    - more in general, new relevant elements for the company
  - In these cases, the old application may be simply tuned-up or completely reengineered
  - TRAMA helps to understand or to remember why certain solutions have been formerly adopted even if some years have passed and if there is a new project team
  - TRAMA allows also to organise the redesign activity according to old dependencies with requirements, identifying the elements that can be improved and the “untouchable” elements linked to still valid goals and that should not be changed

TRAMA in practice 66

TRAMA: entities (1)

- Requirements-related information
  - Visions
    - correspond to a strategic insight of a stakeholder in the domain
    - provide a way for modelling the assumptions of a stakeholder which dictate her "weltanschaung" on the project
  - Goals
    - a wished state of affairs for the main stakeholders
    - a wished experience or an expectation for a class of users
  - Requirements
    - sufficiently high-level descriptions of properties or functionalities of the application as input for the design activity

67

TRAMA: entities (2)

- Design elements
  - Conceptual elements
    - the traditional conceptual design elements
    - content and structure of content, navigation architecture, access paths, operations, pages and layout, etc.
  - Contextual settings
    - e.g. the technical equipment, the place where the application is used, the physical disposition of machines in this place, etc.
  - Organisational elements
    - e.g. how different use sessions are organised during a week, which activities are implies in the use of the application, etc.
  - Other accessorial elements
    - e.g. study material needed to use the application (in a educational system), etc.

68

TRAMA: entities (3)

- Design sources
  - The designer expertise
    - i.e. particular "good design" principles that are part of the designer's skills and that she/he applies in any case
  - A specific understanding of the domain
    - i.e. recurrent good solutions in a domain that the designer applies because she/he learnt it by other cases in the same domain
  - A particular constraint
    - e.g. budget limitations, time, technology limitations, etc.
  - A law obligation
    - e.g. copyright issues, personal data treatment, etc
  - A requirements-related information
    - i.e. a vision, a goal, a requirements, etc
  - An arbitrary choice
    - i.e. a choice without particular reasons, usually a single detail that should anyway be set in a way or another, e.g. the structure of a game in three steps (instead of four or two).

69

TRAMA: relationships

- Kind
  - from requirements to design elements, tracing the impact of requirements on the design
  - from design to requirements, tracing the justification of design solutions
  - from design to its sources, tracing the motivations for design choices
- Direction
  - TRAMA is a post-Requirements Specification Traceability method
  - supports forward traceability from requirements to design elements and backward traceability from design elements to requirements or to other motivations
- Setting
  - Due to the its own nature, TRAMA allows only explicit relationships; the method does not include any implicit or automatic generation of traces
- Attributes
  - rationale of the relationship
  - dependence with another trace
  - priority value

70

TRAMA: analysis tools

- TRAMA is based on traceability matrices
  - cross requirements with design in a forward direction
  - cross design with its sources (requirements, motivations, constraints, etc.) in a backward direction
- RIM (Requirements Impact Model/Matrix)
  - Requirements-to-Design matrix
  - can be filled and read horizontally, highlighting how single requirements are taken into account into the design
  - can be filled and read vertically, showing how a single design element satisfies the project requirements
- DMM (Design Motivations Model/Matrix)
  - Design-to-Sources matrix
  - traces back single design elements to the motivation why a certain decision is relevant for the project
  - e.g. satisfying a requirements, fulfilling a constraint, allowing more usability in the system, etc.

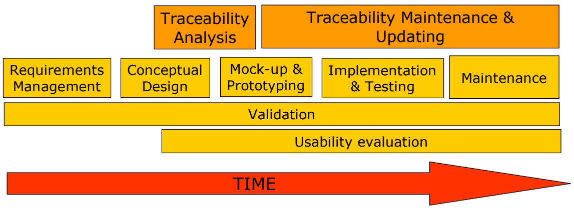
71

Traceability phase

- In which moment of the project life-cycle?
- relevant information can be surfaced after a first version of the design is produced
- a detailed design is possibly needed to profitably trace relationships towards high-level requirements
- Suggestion:
  - perform a tracing activity after the first design phase
  - a continuous activity during the rest of the project is then needed to maintain the traceability specification up-to-date

72

Traceability in the project's life-cycle



TRAMA in practice 73

Summary – Day 1

- Traceability foundations
- Approaches, tools and examples
- TRAMA: basic concepts
- **Example 1**
- Exercise 1

TRAMA in practice 74

Example 1: Munch in Berlin

- Web site for the "Munch un Berlin exhibition" at the *Staatliche Museen zu Berlin* (Germany)
- It represents the first practical result of the WED approach based upon a linguistic approach considering the interaction of a user with a web site as a dialogue
- The web site is optimized for visually impaired people, where the interaction is more natural, like in an oral dialogue

TRAMA in practice 75

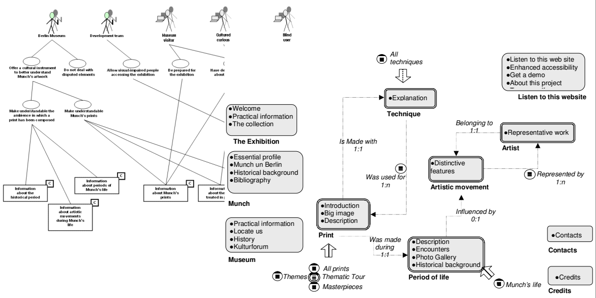
Munch: Preliminary Plan

- Before to put the application on-line, a consistency check have been requested to "adjust" the last elements and to fix an up-to-date documentation of the overall project.
- TEC-Lab performed a first traceability analysis focusing on the conciseness and on the understandability of the documentation to provide.
- A further traceability phase has been conducted in February 2005 to cope with new and refined project goals
  - design a website which might work also as a fixed information kiosk in the museum
  - make the website more usable by visually-impaired users (refining the WED approach)
  - promote knowledge and awareness about a temporary exhibition being hosted at the Museum (Munch's prints and drawings).
- Traceability was here performed to evaluate the impact of changing requirements and of proposed new solutions on the application.

TRAMA in practice 76

Munch: information normalisation

- With AWARE and IDM



TRAMA in practice 77

Munch: elicitation and analysis (RIM)

DESIGN ELEMENTS	CONTENTS	Prints	Techniques	Periods of life	Artistic movements	Artists	Munch	The museum	The exhibition	Contacts	Credits	Listen to this Website	RELATIONSHIPS BETWEEN CONTENTS	From a print to the technique it was made with	From a technique to the prints made with that technique	From a print to the period of life during which the print was made	From a period of life to the artistics movements actives in Germany during that period	From an artistic movement to the artists that represent the movement	From an artist to the artistic movement he belonged to	ACCESS PATHS TO CONTENTS	A list of all the prints	A group of masterpieces	Some thematic tour of prints grouped by theme	A list of Munch's periods of life	A list of all the techniques
VISIONS																									
Frame works of art within historical background				X	X	X										X	X	X						X	
MOTIVATIONS																									
Be prepared for visiting the exhibition		X	X				X	X	X	X				X	X						X	X	X		X
Study Munch and his art		X	X	X	X	X	X							X	X	X	X	X	X		X	X	X	X	X
Appreciate the artworks in the exhibition		X	X									X													
GOALS																									
Design site & kiosk																									
Make site accessible		X										X													
Raise awareness on Munch's prints		X						X	X													X	X		
Provide leading themes				X	X	X										X	X	X						X	
Raise awareness on art. movement					X												X	X							
Raise awareness on polit. background				X	X	X										X	X	X						X	
Understanding what's interesting in the exhibition																						X	X		
Finding information about the paintings in the exhibition		X	X											X	X										
Finding information about Munch's life				X			X																	X	
Finding historical information about Munch				X	X	X	X									X	X	X	X					X	
Finding detailed information about Munch's work and art		X	X											X	X						X	X	X		X
Understanding the relevance of Munch in the history of art					X	X											X	X	X						
Finding information about the techniques used in the paintings			X											X											X
Efficiently accessing the exhibition's topics		X																				X	X		
Understanding the site structure and the browsing capabilities												X													

TRAMA in practice 78

Munch: elicitation and analysis (DMM)

CHOICES SOURCE	Visions	Motivations	Goals	Specific understanding of the domain	Designer expertise	Technology constraints	"Graphic" constraints	Budget constraints	Laws obligations
CONTENTS									
Prints	X	X	X		X				
Techniques		X	X						
Periods of life		X	X						
Artistic movements	X		X						
Artists	X		X						
Munch	X		X						
The museum		X							
The exhibition		X	X						
Contacts					X				
Credits					X				
Listen to this Website			X						
RELATIONSHIPS BETWEEN CONTENTS									
From a print to the technique it was made with					X				
From a technique to the prints made with that technique					X				
From a print to the period of life during which the print was made					X				
From a period of life to the artistic movements actives in Germany during that period	X								
From an artistic movement to the artists that represent the movement					X				
From an artist to the artistic movement he belonged to					X				
ACCESS PATHS TO CONTENTS									
A list of all the prints					X				
A group of masterpieces		X	X						
Some thematic tour of prints grouped by theme			X	X					
A list of Munch's periods of life					X				
A list of all the techniques					X				

79

Example 1: Munch und Berlin

- Class discussion

80

Summary – Day 1

- Traceability foundations
- Approaches, tools and examples
- TRAMA: basic concepts
- Example 1
- **Exercise 1**

81

Exercise 1

- Simple analysis exercise
- Individual
- Choose a topic and try to analyse it

82

Day 1 – Wrap up

- Traceability foundations
- Approaches, tools and examples
- TRAMA: basic concepts
- Example 1
- Exercise 1

83

Summary – Day 2

- Presentations
- TRAMA activity workflow
- Information re-organisation and normalisation
- Exercise 2
- Presentations

84

Summary – Day 2

- **Presentations**
- TRAMA activity workflow
- Information re-organisation and normalisation
- Exercise 2
- Presentations

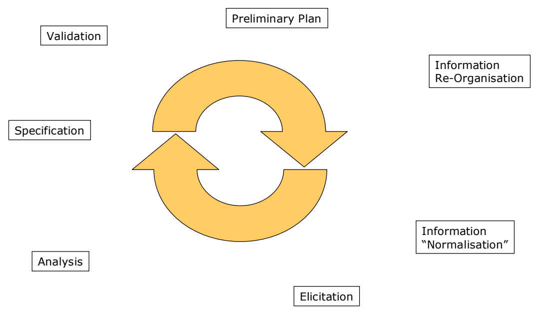
Summary – Day 2

- Presentations
- **TRAMA activity workflow**
- Information re-organisation and normalisation
- Exercise 2
- Presentations

TRAMA components

- The approach consists in
  - a structured analysis process
  - a general conceptual model of entities and relationships to trace
  - a set of conceptual tools supporting traces inquiry, analysis and documentation

TRAMA Analysis Process



TRAMA workflow (1)

- Preliminary plan
  - understanding which the stakeholders of the traceability analysis, the traceability goals, the constraints (time and budget, related to ROI) and the expected results are
- Information re-organisation
  - understanding requirements and design from documents or from interviews with designers and organise it in terms of structured specifications
- Information "normalisation"
  - structuring requirements and design information in "normal" terms, base on a strong methodology (e.g. AWARE for requirements and IDM for design)

TRAMA workflow (2)

- Elicitation
  - surfacing relationships between requirements and design in terms of impact of requirements on the design (“How did you considered this requirements in the design?”) and of motivations for design choices (“Why did you adopted this solution?”).
- Analysis
  - tracing relationship and developing the Requirements Impact and the Design Motivations Matrices (RIM and DMM).
- Specification
  - documenting stakeholders, goals and analysis results
- Validation
  - checking the results with requirements analysts, designers, project managers and clients.

Summary – Day 2

- Presentations
- TRAMA activity workflow
- **Information re-organisation and normalisation**
- Exercise 2
- Presentations

Information re-organisation (1)

- TRAMA aims at discovering relationships between requirements and design and between design and its motivation
- To clearly discuss about this relationships with stakeholders and to avoid misunderstandings, it is needed to have structured and ordered elements both form the requirements and from the design side
- In a perfect world
  - requirements information are explicitly organised and recorded during the project analysis phase
  - this specification is continuously updated during the project development
  - design is step-by-step documented in formal schemes
  - it is always keep aligned with the actual application implemented

We do not live in a perfect world... (1)

- The requirements specification may be unstructured or incomplete
  - the penetration degree of requirements management approaches in industrial practices is very low
  - in most of the cases, unstructured and informal approaches are used to record the information raised out form the firsts operative meetings
  - sometimes there is not a clear and univocal perception of what a “requirement” or what a “goal” is
    - technical details of the applications?
    - high-level visions related to a topic?
    - business-related expectations?
    - application-related desires?

We do not live in a perfect world... (2)

- The requirements specification may be absent
  - In the worst cases, the requirements specification is not only confused or unstructured, but completely absent
  - In some projects the first recorded sign of what goals and requirements were, is the description of how the application is made
  - In frequent cases, the requirements specification is not used as a base to design the application, but it is an ex-post documentation used to describe the backgrounds of an existing product

We do not live in a perfect world... (3)

- The design documentation may be absent or incomplete
  - Often it is not clear what a design for an interactive application is
    - it should describe all the technical implementation details?
    - it should be a conceptual picture of the applications contents, functionalities, navigation, etc.?
  - Sometimes this kind of specification is completely absent
    - Just a technical documentation of how the application has been programmed is provided
  - In other cases, an unstructured specification of the elements of the application design is produced
    - but it includes a mix of indistinct contents, operations, navigation capabilities, organisation elements, roles, etc.

Information re-organisation (2)

- Sometimes, of course, requirements and design specification are recorded with scientific and formal approaches
- Anyway, the TRAMA method cannot take this eventuality for granted but it should consider all the possibilities that can be encountered in the real world
- TRAMA can therefore be applied anyway, no matter if there is previous documentation or not.

- Information re-organisation consists in understanding requirements and design information before to start the tracing process
- The traceability analysts has somehow to understand what the goals, the requirements, etc. of the project and what the designed contents, functionalities, etc. of the application were
- She/he has to "pick up" and to organise these elements in a requirements specification and in a design document.

Information sources

- Specific interviews or focus groups with requirements, analysts, designers, project responsible or other members of the work-team
- Existing documents, specifications, reports, minutes or annotations of some project meeting or activity
- A reverse engineering activity, extracting the design form the actual application or (more difficult) inferring requirements from the design

Information "normalisation"

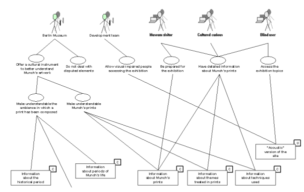
- The knowledge gathered during the information re-organisation activity can be documented pragmatically using no matter what approach
- The approach adopted have to answer to the needs of clarity, simplicity and correctness in terms of information structure

- Normalisation = Structuring requirements and design information in "normal" terms, base on a strong methodology
- A normal form is a representative element within an equivalence class, which is a simples or most manageable or otherwise tidiest and most desirable form, in terms of structure or syntax

- TRAMA distinguish between the sub-activities of
  - requirements normalisation and
  - design normalisation.

Requirements normalisation

- Structuring the requirements-related information in a "normal" form
- Requirements information should be transformed in a more manageable form in order to be traced towards the design
- A goal-oriented methodology is suggested
  - structure the knowledge in terms of goals, goals refinement and requirements
- e.g. AWARE



AWARE: a definition

- AWARE: Analysis of Web Application Requirements
- A goal-oriented methodology supporting the requirements analysis and requirements documentation for web projects
- Representation and understanding of relevant website stakeholders and their goals is key element for successful design

AWARE is

- Stakeholder-centered
  - Websites are made by people for people
- Goal-oriented
  - High-level objectives come before the solutions
- Scenario-based
  - Reflection on contexts of use help requirements surface
- Project-driven
  - Goals and domain knowledge is mediated within the scope of the project
- Tool-independent
  - Flexible notation not constrained by a proprietary platform
- Web-specific
  - but extendable to other domains

AWARE: general concepts

- Stakeholder
- Goal
- Goal Refinement
- Requirement
- Scenario

Stakeholders

- Those who have a direct interest in the success of the website are called stakeholders.
- Stakeholders may include the users, the clients who finance the web site, and other people involved in the project (e.g. sponsor, developers, and representatives of the organization departments, etc.).
- Stakeholders are either individuals or placeholders for an organization's or institution's interests.
- They may be "typed" (e.g. the secretary) or "single" (e.g. the director of bank x)

Goals

- A stakeholder may own one or more goals with respect to the website-to-be.
- A goal is defined as a high-level target of achievement for a stakeholder.
- It may represent a wished state of affairs for the main stakeholders ("Increase customer loyalty"), but also a wished experience or an expectation for a class of users ("Find suitable funds").
- Goals vary in abstraction level and granularity.

Goal refinement

- Goals are analysed by decomposing them into subgoals, according to an ad-hoc refinement process
- The refinement process consists in:
  - Detailing the goals
  - Deciding which and how upper goals may be satisfied - according to the constraints, the obstacles met and resource available – and highlight possible alternatives
  - Defining requirements contributing to accomplish the goals
- The refinement process is mainly top-down but highly iterative

Requirements

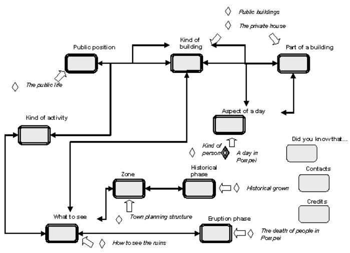
- The outcome of the goal decomposition is a set of requirements, which represent the actual input for the design activity
- A requirement is a sufficiently high-level descriptions of the a property or functionality of the website meaningful for one or more stakeholders (e.g. "provide up-to-date fund information")
- Requirements address a variety of design dimensions (content, navigation, access, operations, etc.)

Scenarios

- The elicitation and refinement process may be supported by envioisioning salient episodes of use of the website, called scenarios (e.g. "an enrolled student looks for information about a specific course he is not attending….")
- Scenarios can help uncover overlooked stakeholders, surface and exemplify goals and requirements, justify, validate or invalidate decisions
- Scenarios provoke stakeholders to reflect on requirements in view of more concrete and vivid artifacts (e.g. pieces of design, prototypes, stories)

Design normalisation

- Transforming the design knowledge gathered during the information re-organisation activity in terms of structured design
- e.g. IDM

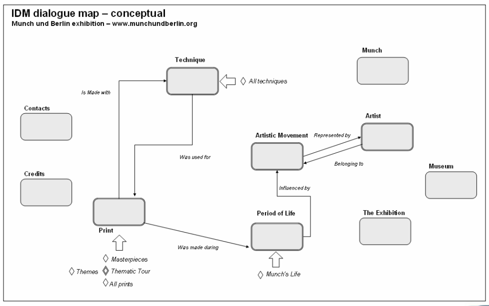


IDM

- Interactive Dialogue Model
- A dialogue-based design model to shape interactive applications
- Can represent both sketched ideas or fully developed solutions
- The graphic representation of these structures is very readable, compact and expressed in a conceptually simple way
- Easy to use for brainstorming
- Good as elicitation tool
- Tailored to master multichannel applications

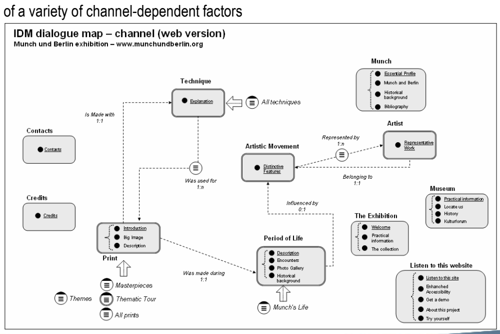
Conceptual design (C-IDM)

- A conceptual schema, of an interactive application, must convey all the necessary "dialogue strategies", without (and before) digging into details depending on technical issues



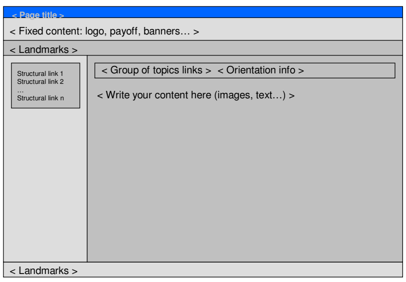
Logical Design (L-IDM)

- It can be seen as a detailed version of the conceptual design, where details are decided on the basis of a variety of channel-dependent factors



Page design (P-IDM)

- Defining the elements to be communicated to the user in a single dialogue act
- Crafting the actual pages containing the necessary elements to sustain the dialogue



Summary – Day 2

- Presentations
- TRAMA activity workflow
- Information re-organisation and normalisation
- **Exercise 2**
- Presentations

Exercise 2

- Information re-organisation and normalisation exercise
- Individual
- Use the material provided by the teacher to re-organise the knowledge in terms of AWARE and IDM

Summary – Day 2

- Presentations
- TRAMA activity workflow
- Information re-organisation and normalisation
- Exercise 2
- **Presentations**

Day 2 – Wrap-up

- Presentations
- TRAMA activity workflow
- Information re-organisation and normalisation
- Exercise 2
- Presentations

Summary – Day 3

- Elicitation and analysis
- Example 2
- Specification and validation
- Example 3
- Exercise 3
- Presentations
- Wrap-up

Summary – Day 3

- **Elicitation and analysis**
- Example 2
- Specification and validation
- Example 3
- Exercise 3
- Presentations
- Wrap-up

Elicitation

- Elicitation is to call forth or draw out as information or a response something latent or potential
- Elicitation is the process of identifying needs and bridging the disparities among the involved communities for the purpose of defining and distilling requirements to meet the constraints of these communities
- Elicitation is the activity of surfacing relationships between requirements and design in terms of impact of requirements on the design ("How did you considered this requirements in the design?") and of motivations for design choices ("Why did you adopted this solution?")

Elicitation techniques

- Create an environment where stakeholders feel at their ease and are able to demonstrate ideas
- Combine different techniques:
  - Interviews
  - Focus groups
  - Questionnaires
  - Direct observation

Interviews (1)

- Very common for this kind of activity because they allow a "live" contact with a person that could be a source of information
- Here everything depends on the interviewer's skills and on the right selection of people to talk with
- In large projects where many people are involved, this activity could take a lot of time

Interviews (2)

- Benefits
  - When "few" people each know a "Lot"
  - Gather RICH information
  - Insights about stakeholder's perspectives
  - Insights about the culture and the domain
- Tips
  - Allow people showing material, examples and demonstrating their ideas
  - Trade-off between listening, guiding and intrusion
- Drawbacks
  - Time consuming
  - Miss interaction between stakeholders

Focus groups (1)

- Discussion meetings between the traceability expert and the project's work-team
- It is possible a "live" contact with people working on the project
- It is not so focused as in an interview
- New knowledge may raised out from group discussion
- A single meeting or a couple of meetings do not take so much time

Focus groups (2)

- Benefits
  - New knowledge from discussions and interaction
  - Good both for brainstorming and focus groups
  - Everybody need to explain ideas for other to understand
- Tips
  - 3-20 stakeholders in one room
  - Analysty offers issues and questions
  - Every one should feel accepted and involved in
- Drawbacks
  - Difficult to fit in the stakehoders' agenda
  - Only "public" opinion emerge
  - Risk to be conflict-driven

Questionnaires (1)

- Can be used as a preliminary step in focus groups or interviews
- To set up the discussion agenda
- Where too much people are involved in the project
  - interviews for the two or three project main responsible
  - questionnaires for all the other project workers

Questionnaires (2)

- Benefits
  - Quantify and compare data
  - Large sample at low cost
  - Appear scientific due to statistical data
- Tips
  - Should be short
  - Alternate open and close questions
- Drawbacks
  - No time for explanation, solve misunderstanding and provoke "habit change"
  - No human touch
  - Focussed aswers to specific questions only
  - Short time causes poor reflection and knowledge evocation

Direct observation (1)

- Following the entire project form the beginning can be an option in case of high budgets and large projects
- Here a traceability expert follow the different project's phases as an internal observer and debrief step-by-step the motivations why the application is designed in a certain way
- This technique presupposes many time and resources to be performed

Direct observation (2)

- Benefits
  - Stakeholders are observed while doing their job
  - Insight about actual process, work context and time
  - Elicit tacit knowledge and automatic processes
- Tips
  - Be as passive as possible
- Drawbacks
  - **Hawthorne effect**: people aware of being watched act differently than they do when unobserved

Meetings set-up

- Place
  - A large meetings room with a table and some chairs
  - The room should have some free walls in order to hang up the papers with the matrices
- Tools
  - Coloured pencils
  - Blackboard/flipcharts/Papers
  - Self-stick wall pads
- Roles
  - the discussants, e.g. in focus groups the project work-team that animate the meeting
  - a facilitator, i.e. a traceability expert in charge to address the discussion in a right direction, provoking answers, asking critical questions, etc.
  - a wall writer, drawing the matrices on the wall papers and filling the crosses with the traceability information raised out from discussion
  - a secretary, recording and writing notes (on a PC) about the meeting
  - a chair officer, e.g. the project manager coordinating the overall meeting

Some biases in elicitation

- Cognitive biases
- Overconfidence
- Faulty reasoning
- Communication problems
- Motivational biases

Cognitive biases

- **Easy of recall**: events that are vivid and emotional or happened recently are easier to recall by the stakeholders, but they are not actually likely to occur.
- Stak. "it is very important that the user might be able to find that information"
- User "I really liked the home page of that site"
- Strategy:
  - Directed questions:
    - "how many timed does it happen in the last month?"
    - "what if the same goals is achieved by different means"?
    - "Why" questions

Overconfidence

- **Overconfidence**: Analysts are optimistic about their understanding of stakeholders' goals. Requirements gathering process risk terminating too soon.
- An. "…I see what you need, that is enough for me"
- Strategy:
  - Scenario reflection: revealing knowledge being used rather than assumed
  - Direct prompting: using the ideas of another stakeholders as counter-arguments for causing reflection
  - What other kind of solution could you imagine?
  - "why questions"

Faulty reasoning

- **Faulty reasoning**: stakeholders might do illogical inferences in supporting their beliefs.
- "In the site, products must be organized by storing categories because our product catalogue – as you can see – is organized in this way. Also our supplier presents information by similar categories, so…"
- Strategy:
  - Devil's advocate
  - Scenario reflection

Communication problems (1)

- Different Background
  - tech vs manag
- Different Domain Knowledge
  - ad extra – ad intra
- Different Language
  - system specific vs domain specific
- Different Goals
  - efficiency and easy of maintainance vs maximum functionality

Communication problems (2)

- Strategy: Pre-elicitation conditioning
  - Discuss the purpose of the meeting
  - What the analyst will be asking
  - What stakeholder will need to provide
  - Explain key terms
  - Explain how information will be used
  - Making stakeholders aware of potential biases

Motivational biases (1)

- Stakeholders are unwilling to provide accurate requirements because:
  - Organizational policy
    - Fear of being evaluated by others
    - Don't know who will know what they say
    - Fear of offending someone or break balances
    - Self-protection, self-preservation
    - Bias on domains of other stakeholders
  - Don't know what analyst needs
  - Don't know other stakeholders already met

Motivational biases (2)

Strategy: Pre-elicitation conditioning
- Explain how information elicited will benefit both
- Explain how information elicited will be used
- State that everyone's opinion is valued
- Tell other stakeholders already met
- Assure responses are kept confidential

139

Analysis

- Taking all the information surfaced by the different elicitation practices performed (interviews, focus groups, etc.)
- Gathering all this knowledge in a structured and analytical picture
- Different points of view have to be integrated:
  - The designer's point of view
  - The client/customer's point of view

140

The designer's point of view

- Each designer develop different parts and different functionalities of a same application
- His/her perception of the project is often limited to a "vertical" view on how these parts and functionalities answers to the strategic needs
- The traceability analysis have to gather all these partial views, showing how the entire application fits with requirements through the inter-action of its different parts.

141

The client/customer's point of view

- Often this point of view is mediated by the project manager
- The focus here is how a single requirement has been taken into account in the application development
- The analyst have therefore to consider all the information gatherer from an "horizontal" point of view, documenting the impact that all the strategic needs (expressed by goals and requirements) have on the application design

142

TRAMA analysis aspects

- Designer's and client's points of view are mirrored in two aspects taken into account by the TRAMA analysis:
  - the justification or motivation of the design (designer's point of view), that can comes from requirements or from other kind of sources (an understand of the specific domain, the expertise of the designer, a constraint, etc.);
    - these traces are called Design Motivations Model (DMM)
  - the impact on design (client's point of view) of: visions, stakeholders-goals, users-motivations, domain issues, scenarios, constraints and requirements
    - these traces form the Requirements Impact Model (RIM)

143

Client validation

- To set up a structured argumentation to show to the client that all the needs have been taken into consideration
- This activity is supported by a proper traceability approach
  - in a forward direction
    - showing which requirements have been taken into account in the design and how
    - following evolving requirements in design
    - checking consistency and feasibility of requirements
    - estimating the impact of a change in requirements on the design
  - in a backward direction
    - finding arguments to justify design decisions
    - checking whether all requirements are considered by the design
    - and estimating the effect of a required design change
- In TRAMA the RIM matrix allows project managers and designers to map each goal and each requirements into design solutions, providing a powerful communication tool to show that everything (every strategic goal, etc.) has been considered in the application

144

Design versioning

- To highlight different design areas for different stakeholders.
- A proper backward traceability approach allows understanding which parts of the design are relevant for which stakeholder
- The design-requirements-goals-stakeholders chain helps creating different versions of the design documentation, addressed to specific targets

- If requirements are normalised with a proper goal-oriented methodology (e.g. with AWARE), each goal is linked to the stakeholder(s) who owns it
- Goals-to-design relationships in the RIM matrix allows to identify the application elements that satisfy the goals of a specific stakeholder

Non-traceable design

- To document the motivations of design elements that do not derive from requirements

- A big part of the design elements are not motivated by a requirement-related information
- Most of the choices come from usability or "good design" principles or are just due to the designer's expertise

- The DMM matrix allows to distinguish the different motivations for design elements, relating design with its sources types and answering to the "why" question ("Why this design element has been placed into the application? Why in this way?")

"Negative" design

- With "negative" design I mean those design objects that have been eliminated or modified during the project life-cycle
- Proposed elements in the application may become part of the negative design
  - because of a direct rejection
  - because of a change in related objects
  - because of business, technology or law constraints
- Keeping trace of old design versions and understand and remember former design decision is useful to
  - remember why a decision and not another has been taken
  - validate negative decisions with stakeholders
  - understand why a design decision has been rejected
  - show the "negative" impact of a specific constraint or requirement on design

- Rejected design choices can be (separately) listed in the DMM matrix
- The crosses with the different sources types answer to the "why not" question

Reverse requirements specification (1)

- Check the consistency between design and requirements
- "Tune" requirements specification according to the real stakeholders' goals
- Extract consistent requirements specification from design.

- Sometimes requirements specifications are written after design or after implementation phase, just for documentation
- In these cases, a proper traceability approach may help in producing an effective requirements specification according to the real stakeholders' goals and requirements

Reverse requirements specification (2)

- "Ex-post" traces are anyway useful to
  - check the consistency between design and requirements
  - tune up existing requirements specification according to the actual application
  - extract consistent requirements specification from design
- Such a reverse requirements specification is a beautiful tool to
  - keep trace of strategic decisions
  - provide design decisions with argumentations
  - collect information and material for a consistent usability test

- This activity is supported by the RIM matrix that force analysts in surfacing consistency or inconsistency traces.

Usability on design documents

- Select the elements in the design involved for a specific task
- Evaluating the quality of the product with respect to the high-level goals
- Identifying test procedures that should be rerun to validate an implemented design change

- The RIM matrix allows usability experts to perform inspections on specific design areas, properly considering the strategic goals that should be fit by those inspected elements

Analysis checklist

- TRAMA provides to analysts a set of pragmatic questions
- They can be used as a guide or as a checklist to properly consider all the aspects involved in a relationship between project elements
- For each cross in a matrix the traceability expert should ask himself:
  - “Which design element fits with the needs of this stakeholder?”; “If I had to present the project to this stakeholder, which part of the design should I highlight?”;
  - “Which design element fits with this goal?”; “Which is the impact of this goal into the design?”;
  - “Which design element better fits with the needs of this user?”; “How can I arguing design choices to show that this user is considered in it?”
  - “Which strategy is set-up in the design to fit with this user motivation?”;
  - “Which is the (positive or negative) impact of this constraint into the design?”;
  - “Which are the design elements that fit with this requirement?”; “How can I show that this requirement has been properly taken into account in the design?”;
  - “Why the designer chose to put this element into the design?”; “How can I show that this element is not an extra-feature in the design?”;
  - “Why this element has been rejected or modified in the current design?”; “What is the impact of this choice into the project consistency with strategic goals?”

TRAMA in practice 151

RIM – Requirements Impact Matrix

- Lists vertically requirements-related information and horizontally all the design elements
- Highlights the impact on design of visions, stakeholders goals, users motivations and requirements
- Vertically
  - information on how single requirements are taken into account into the design
  - “Taking into account a single design element, how does it fit with requirements?”
- Horizontally
  - how a single design element satisfy the project requirements
  - “Taking into account a single requirements, how has it considered in the design?”

		DESING ELEMENTS			
		Content 1	Content 2	Access path 1	...
REQUIREMENTS-RELATED INFORMATION	VISIONS				
	Vision 1				
	Vision 2				
	...				
	GOALS				
	Goal 1				
	Goal 2				
	...				
	REQUIREMENTS				
	Requirement 1				
	Requirement 2				
	....				

TRAMA in practice 152

DMM – Design Motivations Matrix

- Lists vertically the design elements of the application and horizontally their kind of motivations
- It is not just the opposite of RIM
- “Negative” design elements can be also listed in this matrix
- Horizontally: “Why did you adopted this solution?”, “Why did you rejected this solution?”

		DESING MOTIVATIONS						
		Visions	Goals/Requirements	Designer expertise	Understanding of the domain	Contraints	Law obbligations	Arbitrary choices
DESIGN ELEMENTS	Content 1							
	Content 2							
	Access path 1							
	...							
	Negative design element 1							
	Negative design element 2							
	...							

TRAMA in practice 153

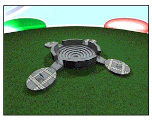
Summary – Day 3

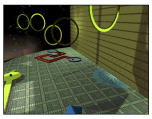
- Elicitation and analysis
- **Example 2**
- Specification and validation
- Example 3
- Exercise 3
- Presentations
- Wrap-up

TRAMA in practice 154

Example 2: Learning @ Europe

- Educational project aiming at fostering the development of a “European Identity”
- Educational approach novel in several respects
  - advanced content
  - technology-enhanced e-Learning
  - multicultural experience
  - engaging “games”
  - cultural competition





TRAMA in practice 155

L@E: preliminary plan (1)

- In a first full experimentation year, between 2004 and 2005, 48 classes from 6 European countries (Belgium, France, Italy, Norway, Poland and Spain), nearly 60 teachers and 1,000 students were involved
- A new advanced experimentation year, between 2005 and 2006, will bring the project at an industrial stage. Before this new experimentation, a complete revision of the whole setting of the experience will be performed.
- A traceability analysis has been requested to facilitate this revision activity

TRAMA in practice 156

L@E: preliminary plan (2)

- Goals
  - reorganize the complex and various material describing and designing the experience
  - pave the grounds for a reengineering activity

  - internal communication, to communicate the project status to all the team members
  - reverse requirements engineering, re-organizing and refining requirements and surfacing missing information, fundamentals to understand the project but never explicitly documented
  - design tuning, surfacing missing design components and re-aligning the design with the project state-of-the-art
  - design revision, to facilitate the project revision before a new experimentation period

L@E: information "normalisation" (Goals)

- General goals
  - Offering to schools a collaborative learning experience based on new technologies
  - Basing the experience on historical contents
  - Basing the experience on a multicultural approach
  - Allowing the educational impact to be measurable
  - Allowing to participate classes and pupils of every level and kind, not only the best classes in the best schools
  - Minimizing the internal management costs of the experience

- Educational goals
  - Knowledge (i.e. teaching a "know what" to students)
    - About local (national) history
    - About other countries' history
    - About general historical concepts and processes
  - Skills (i.e. teaching a "know how" to students)
    - Use of "professional" English (as a tool to work)
    - Use of technological tools for synchronous or asynchronous collaboration (3D worlds, forums, online communities, etc.)
    - Group work (face to face collaboration)
    - Collaborative work (remote collaboration)
  - B3 Attitudes (i.e. provoke an habit change to students)
    - Sense of curiosity for history
    - National identities are the result of a process: multiple cultures / multiple identities
    - Improved attitude towards history
    - Critical thinking towards knowledge: truth appears through a variety of opinions
    - Different attitude towards knowledge, different learning modality (e-learning)

L@E: information "normalisation" (Visions)

- Integration in schools' curricula
  - Convenient quantity of commitment
    - For students. The project aims at support schools as they are and not to subvert the internal organisation; the experience have to involve an entire class (12 to 25 students) and have to be guided by teachers with active and directive roles. The project may help teachers in managing different class segments.
    - For teachers. The project aims do not include that teachers learn something about technology. The experience does not base on teachers' technological skills.
  - Convenient use of infrastructural resources. The project must not requests to school an excessive use of laboratories or a too sophisticated technological equipment.
  - The educational benefits have to be related with the general educational goals of schools and of their curricula. Teachers must be able to justify the time and organisational effort spent for participating in this experience.

- Characteristics of and educational competition
  - It has to be a motivation for students in learning; it has to be a "true" competition and repay the commitment. Therefore the competition should be:
    - Open: motivation should remain active for everyone until the end, also for micro-sessions
    - Serious: it should repay different skills and valorise a deep understanding but it should not be frustrating
  - It has not to be frustrating: participants should not be demotivated by difference of results with the others. This characteristic have to be balanced with the previous one.
  - Engaging but not an end in itself; e.g. the access to cultural questions (the "serious" part) could be win with games involving "physical" or technical skills (the engaging part).

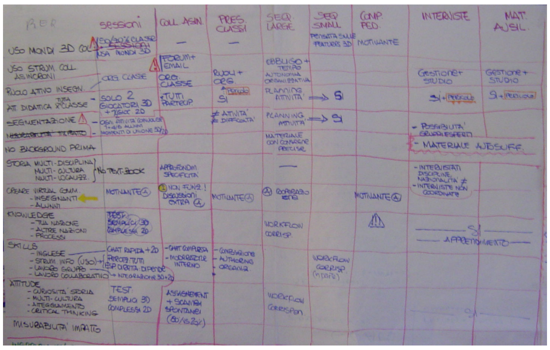
L@E: information "normalisation" (Requirements)

- The experience have to include the use of collaborative 3D worlds
- The experience have to include the use of tools for asynchronous collaboration
- The experience have to include the teachers' active role
- The educational activities have to involve the whole class
- The activities have to be modularized in order to facilitate class segmentation
- The activities must require to students a minimum background knowledge
- The activities must not presuppose that teachers know how to use technologies
- The applications must allow to participate with a low technology level and include a degraded mode of use for low connections
- The historical contents have to highlight multiple opinions, disciplines, localizations and cultures involved in the topic
- The experience have to support the creation of a virtual communities of students, also after the end of the project
- The experience have to support the creation of a virtual communities of teachers, also after the end of the project

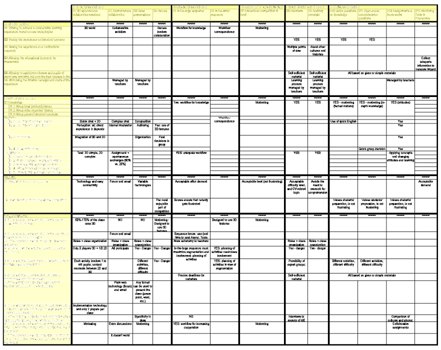
L@E: information "normalisation" (Design)

- Static components
  - 3D synchronous collaborative sessions
  - Asynchronous collaboration (forum/email)
  - Class presentations
  - Games
- Dynamic components
  - In-the-large sequence: succession of sessions, asynchronous sessions and off-line activities during the experience
  - In-the-small sequence : succession of the activities, contents and tests in a session
- Transversal components
  - Educational competition in itself
- Educational materials
  - Interviews (extended and simplified)
  - Auxiliary materials
- Testing materials
  - Quick questions on knowledge, "matter of fact" about local history, about other countries' history and about general historical concepts
  - Open-ended comprehension questions about local history, about other countries' history and about general historical concepts
  - Assignments & home-works (to apply the knowledge)
  - Monitoring Tools & Procedures

L@E: Meetings



L@E: elicitation and analysis (RIM)



L@E: elicitation and analysis (RIM)

	STATIC COMPONENTS				DYNAMIC COMPONENTS
	D1 3D synchronous collaborative sessions	D2 Asynchronous collaboration	D3 Class presentations	D4 Games	D5 In-the-large sequence
EDUCATIONAL GOALS					
B1 Knowledge					Yes: workflow for knowledge
B1.1 About local (national) history					
B1.2 About other countries' history					
B1.3 About general historical concepts					
B2 Skills					
B2.1 Use of "professional" English	Quick chat + 2D	Complex chat	Composition		
B2.2 Use of technological tools	Perception: all; direct experience: it depends	Internal moderator	Authoring	Yes: use of 3D features	
B2.3 Group work	Integration of 3D and 2D		Organization	Fast decisions in group	
B2.4 Collaborative work					
B3 Attitudes	Test: 3D simple, 2D complex	Assignment + spontaneous exchanges (80% vs. 20%)			YES: adequate workflow
B3.1 Sense of curiosity for history					
B3.2 Multiple cultures / multiple identities					
B3.3 Improved attitude towards history					
B3.4 Critical thinking towards knowledge					
B3.5 Different attitude towards knowledge					
VISIONS					
V1 Integration in schools' curricula	Technology and easy connectivity	Forum and email	Variable technologies		Acceptable effort demand
V2 Characteristics of and educational competition				The most enjoyable part of competition	Scores ensure that nobody gets frustrated
REQUIREMENTS					
R01 The experience have to include the use of collaborative 3D worlds	50% / 70% of the class uses 3D	NO	NO	Motivating - Designed to use 3D features	
R02 The experience have to include the use of tools for asynchronous collaboration		Forum and email			Sequence forces use (and time to use) Async. Tools

L@E: elicitation and analysis (DMM)

	Visions	Requirements	Designer expertise	Specific understanding of the domain	Constraints	Law obbligations
STATIC COMPONENTS						
D1 3D synchronous collaborative sessions		Offering to schools a collaborative learning experience based on new technologies		In past projects this technology has been a good motivation for e-learning	Use of the in-house 3D technology	
D2 Asynchronous collaboration		Create a community	Forums and emails are common tools to build up an asynchronous collaboration			
D3 Class presentations			Just to add an activity in the first (introductory) session			
D4 Games				Another good idea from past projects as a motivation for traditional learning		
DYNAMIC COMPONENTS						
D5 In-the-large sequence	Acceptable effort demand	Workflow adequate to educational goals				
D6 In-the-small sequence			Workflow correspondance. Each session has not to be too "boring" or long or simple/difficult, etc.			
TRANSVERSAL COMPONENTS						
D7 Educational competition in itself	It has to be a motivation for students in learning: open and serious			In past projects it has been a good motivation for e-learning		
EDUCATIONAL MATERIALS						
D8 Interviews			This way to present information has worked very well in past projects			
D9 Auxiliary materials			This material go with interviews as background explaination (maps, biographies, etc.)			
TESTING MATERIALS						
D10 Quick questions on knowledge	Values students' preparation, is not frustrating					
D11 Open-ended comprehension questions						
D12 Assignments & home-works		Improve knowledge, skills and attitudes				
D13 Monitoring Tools & Procedures		The educational impact has to be measurable				

Summary – Day 3

- Elicitation and analysis
- Example 2
- **Specification and validation**
- Example 3
- Exercise 3
- Presentations
- Wrap-up

Specification

- After the analysis has been properly conducted, the traceability expert have to present all the results in a structured document
- This document is a real traceability specification, reporting
  - which the stakeholders and the goals of the tracing activity were
  - which kind of activities have been actually performed
  - which information have been surfaced
  - which the consequences of these results may be
- The role of a traceability documentation is resuming the elements surfaced during the analysis and organising it in a structured way
- This kind of specification is able to summarise all the project components and the relationships between them, allowing a compact but complete understanding of the project status

Typical specification structure (1)

1. Executive summary.
2. Project summary
   - highlighting its goals, people involved, current status, etc.
3. Traceability preliminary plan
   - summarising the goals, the stakeholders and the expected results of this activity
4. Information re-organisation and normalisation section
   - presenting how the project knowledge have been structure to allow a proper tracing activity
   - this section provides a general view of the current project status, a compact requirements specification and a compact design schema

Typical specification structure (2)

5. RIM matrix
   - with comments, highlighting the relationships between requirements and design
6. DMM matrix
   - with comments, highlighting the relationships between design and its motivations
7. Summary of the results achieved by the analysis
8. Benefits
   - section highlighting the benefits that traceability brings or will bring to the project.
9. Conclusion
   - reporting the reasons why the tracing activity has been performed and the main consequences for the project, in terms of design areas to review, features to better implements, requirements to re-consider, etc.

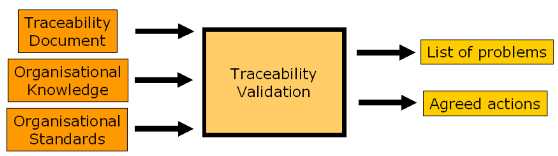
Validation (1)

- Traceability validation is the activity to check the analysis results with requirements analysts, designers, project managers and clients
- The specification is written for these people, so it must be written in a language which they can understand
- Furthermore, the results should be written so that they may be verified
- Validation works with a final draft of the traceability document, i.e. with negotiated and agreed information, after each meeting with project managers and designers
- The validation phase is therefore a "transversal" activity that should be run and re-run continuously during elicitation and analysis, as well as after the specification has been written

Validation (2)

- Validation certifies that the traceability document is an acceptable description of the overall project, in terms of:
  - Completeness and consistency of all the information reported.
  - Conformance to standards adopted in the project and in the company (reports structure, responsibilities, etc.).
  - Conflicts between traceability stakeholders' goals, e.g. between designers ("all our choices were strongly motivated") and clients ("some elements could be improved").
  - Technical errors in the description of how the design is of what the requirements are, from a designers and requirements analysts point of view.
  - Ambiguous information, expressed not clearly or using terms, schemas or other elements that in that particular project or in the company have other meaning

The validation process (1)



The validation process (2)

- Inputs
  - the traceability document
    - should be a complete version of the document, not an unfinished draft, formatted and organised according to organisational standards
  - organisational knowledge
    - knowledge, often implicit, of the organisation which may be used to judge the realism of the results
  - organisational standards
    - local standards e.g. for the organisation of the specification documents

The validation process (3)

- Outputs:
  - list of problems
    - a list of discovered problems in the traceability document
  - agreed actions
    - a list of agreed actions (that can be several or none) in response to problems discovered

Summary – Day 3

- Elicitation and analysis
- Example 2
- Specification and validation
- **Example 3**
- Exercise 3
- Presentations
- Wrap-up

175

Example 3: L@E

- Documentation example
- Learning @ Europe: document provided by teacher

176

Summary – Day 3

- Elicitation and analysis
- Example 2
- Specification and validation
- Example 3
- **Exercise 3**
- Presentations
- Wrap-up

177

Exercise 3

- Advanced analysis exercise
- Individual
- Choose a project and analyse it applying TRAMA

178

Summary – Day 3

- Elicitation and analysis
- Example 2
- Specification and validation
- Example 3
- Exercise 3
- **Presentations**
- Wrap-up

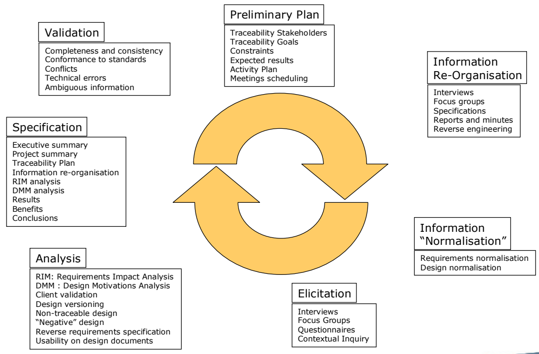
179

180

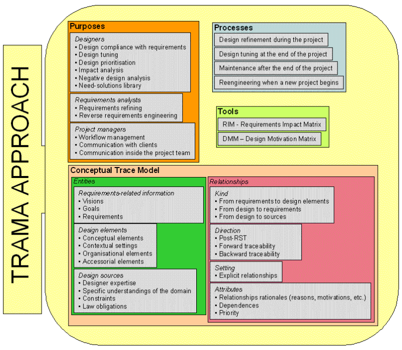
Summary – Day 3

- Elicitation and analysis
- Example 2
- Specification and validation
- Example 3
- Exercise 3
- Presentations
- **Wrap-up**

TRAMA Analysis Process



TRAMA approach



TRAMA: pros and cons

- **Benefits**
  - **a powerful communication mean to show to the clients that all their requirements have been considered and how, and that there are not unmotivated elements in the design;**
  - **a structured practice to check design consistency for revision;**
  - **an advanced tool to tune up design in maintenance phase;**
  - **a complete project knowledge summary of requirements, of design elements and of relationships between them, as vital information allowing an effective system reengineering**
- **Limits**
  - **Maintenance problems**
  - **Solution: the requirement watcher**

Course wrap-up

- Traceability foundations
- Approaches, tools and examples
- TRAMA: basic concepts
- Example 1
- Exercise 1
  - Presentations
  - TRAMA activity workflow
  - Information re-organisation and normalisation
  - Exercise 2
  - Presentations
    - Elicitation and analysis
    - Example 2
    - Specification and validation
    - Example 3
    - Exercise 3
    - Presentations
    - Wrap-up

TRAMA: Traceability Analysis Method for (interactive) Applications

Slides Pack 1

Giovanni Randazzo

65 slides

Slides pack 1

- Defining traceability (G1_M1, Goal1_Module1);
- Focusing on Requirements Traceability (G2_M1);
- Showing possible traceability approaches: purposes, processes, models and tools (G3_M1);
- Introducing examples of traceability practices (G4_M1);
- Providing a short review of traceability software tools (G5_M1).

Slides pack 1

- **Defining traceability (G1_M1, Goal1_Module1);**
- Focusing on Requirements Traceability (G2_M1);
- Showing possible traceability approaches: purposes, processes, models and tools (G3_M1);
- Introducing examples of traceability practices (G4_M1);
- Providing a short review of traceability software tools (G5_M1).

Traceab... what? (1)

- Traceability is
  - the degree to which a relationship can be established between two or more products of the development process [IEEE, 1990]
  - the ability to explicitly trace and document relationships between the different phases of a project's life-cycle

Quality of Service

- *ad intra*
  - quality is considered by mean of the intrinsic characteristics of the application (e.g. performance, accuracy, up-to-date);
- *ad extra*
  - quality is the correspondence between services offered and stakeholders' goals; it can see as the combination of the quality of the user experience, the user satisfaction and the main stakeholder' satisfaction

Software Quality

- the totality of features and characteristics of a software product that bears on its ability to satisfy given needs, for example to conform to specifications
- the degree to which software possesses a desired combination of attributes
- the degree to which a customer or user perceives that software meets his or her composite expectations
- the composite characteristic of software that determine the degree to which the software in use will meet the expectations of the customer

Traceability as element of SQ

- Quality is a multifaceted characteristic of an application
- The quality degree of a project may depend on
  - services and features provided
  - user satisfaction and context of use
  - customer and main stakeholders satisfaction
  - compliance with strategic goals
  - impact on the organisation
- It becomes crucial to keep in a global picture the relationships between these elements
- Traceability can improve the quality of the systems development process.

Slides pack 1

- Defining traceability (G1_M1, Goal1_Module1);
- **Focusing on Requirements Traceability (G2_M1);**
- Showing possible traceability approaches: purposes, processes, models and tools (G3_M1);
- Introducing examples of traceability practices (G4_M1);
- Providing a short review of traceability software tools (G5_M1).

Traceab... what? (2)

- Requirements Traceability
  - is the ability to determine which documentation entities of a software system are related to which other documentation entities according to specific relationships
  - helps ascertain how and why system development products satisfy stakeholder requirements

Pre-Requirements Specification traceability (1)

- Pre-RST is concerned with those aspects of a requirement's life prior to its inclusion in the requirements specification
  - Requirements production and refinement
- It is a technique that attempts to document the rationale and socio-political context from which requirements emerge, thus linking the business world with that of information technology
- Serves to answer questions that arise during the project's life-cycle, including:
  - "Who is responsible for including this requirement?"
  - "To whom should I refer to for more information?"
  - "Who was responsible for copying this information into this document?"
  - "Was this requirement a result of a meeting of stakeholders or just one individual?"
- Pre-RST facilitates the reopening of previously closed specifications, tracing back to the sources of requirements, and then the (possible) reworking of a specification in the forward direction

Pre-Requirements Specification traceability (2)

- Sources of requirements may be the following:
  - Stakeholder Visions: stakeholders are those who have a direct interest in the success of the website (e.g. clients, sponsors, representatives, opinion makers, etc.); stakeholder visions are the assumptions of a stakeholder which dictate her "weltanshaung" on the project
  - User Motivations: they shape the emotional, psychological, social or individual elements which can trigger a person (a final user) to use an interactive application
  - Goals: they are defined as high-level targets of achievement for a user or a stakeholder; goals may represent a wished state of affairs (for main stakeholders) or a wished experience (for users) and may arise from visions or motivations
  - Constraints: they are defined as those elements that implicate a restriction on the degree of freedom the requirement analyst have in providing a solution; constraints can be economic, political, technical, or environmental and pertain to project resources, schedule, target environment, or to the system itself.

Post-Requirements Specification traceability (1)

- Post-RST is concerned with those aspects of a requirement's life that result from its inclusion in the RS
  - requirement deployment and use
- This kind of traceability provides a way to elicit and discover the impact of requirements and how requirements have been taken into account on the subsequent project elements

Post-Requirements Specification traceability (2)

- Targets of requirements may be the following:
  - conceptual design: high-level definition of the information structure, of the features and of the services/capabilities that the application will own;
  - technical design: in-detail definition of the software (and/or hardware) components the application will be made of;
  - experience design: definition of all the elements contributing in building the user experience, including organisational concerns, technical set-up and use scenarios;
  - implementation: it's the "tangible" part of the application, i.e. classes, routines, lines of code, interfaces, etc.
  - tests: including technical test verifying if the application works properly, usability tests and accessibility test.

Forward traceability

Requirements → Design

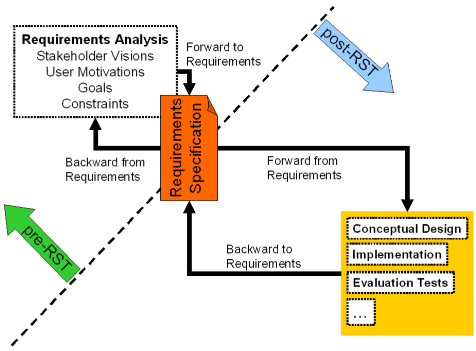
- What is the impact of each requirement on the application?
- Forward to requirements
  - maps stakeholder needs, visions and goals to the requirements, so that the analyst can determine the impact to requirements as needs change
- Forward from requirements
  - assigns responsibility for fulfilling a requirement to the design or to the various system components that will implement it, letting the responsible ensure that each requirement is fulfilled

Backward traceability

Requirements ← Design

- What is the motivation of the presence of each design element in the application?
- Backward from requirements
  - lets the analyst verify that the system meets the user community's needs
  - allow to understand the source of requirements
- Backward to requirements
  - verifies compliance of design, software or tests built to requirements

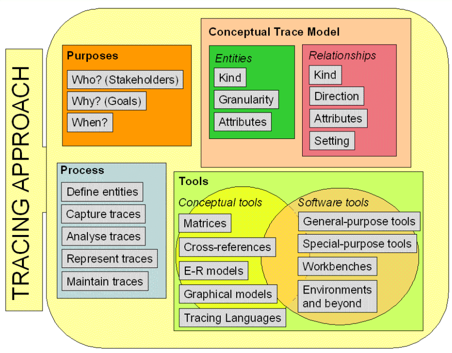
Traceability types



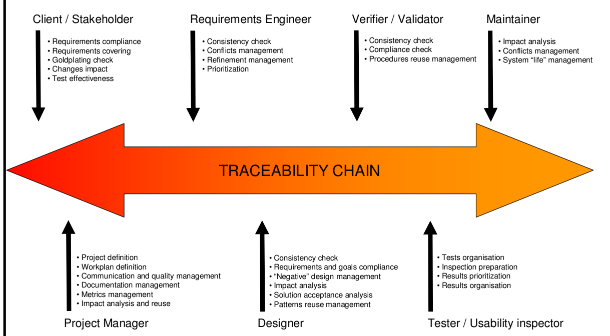
Slides pack 1

- Defining traceability (G1_M1, Goal1_Module1);
- Focusing on Requirements Traceability (G2_M1);
- **Showing possible traceability approaches: purposes, processes, models and tools (G3_M1);**
- Introducing examples of traceability practices (G4_M1);
- Providing a short review of traceability software tools (G5_M1).

A tracing meta-approach



Purposes



Slides pack 1 19

Client/Customer/Stakeholder (1)

- They have a certain number of problems in evaluating the quality and the effectiveness of a software application a priori
  - before its effects have been produced
- There is a knowledge and understanding gap between stakeholders and the development team
- Clients hardly can see how and where the applications provided may fit to their needs and goals
- Traceability analysts can guide these people in evaluating such applications
- Traceability is a communication “bridge” between
  - a client (usually with marketing or economics background)
  - and a software house, a web agency or anyway the internal development team (with engineering or informatics background)

Slides pack 1 20

Client/Customer/Stakeholder (2)

- Requirements compliance
  - Traceability can shows the relationships between strategic goals, requirements and solutions in the application, allowing clients evaluating the compliance degree of the product with their needs
  - The overall quality of the application can be understood without any need to consider single technical or software details
- Requirements covering
  - Relationships between requirements and elements or pieces of the application may highlight the progress state of the project
  - Clients can understand which percentage of the stated requirements are met and which part of the job is completed
  - A thorough traceability analysis may also provide stakeholders that all the strategic goals have been satisfied and how the application will address to their needs
- Goldplating check
  - Goldplating is the presence of features that are not motivated by any explicit reason
  - Traceability analysis highlights goldplating by linking all the application features with their motivations

Slides pack 1 21

Client/Customer/Stakeholder (3)

- Changes impact
  - It is not unusual to observe that after the end of a project, clients may ask to developers further changes to the applications
  - Reasons can be identify in lack of proper needs analysis or in lack of proper communication to the client
  - Traceability analysts can help clients in evaluating the consequences of their requests, i.e. the impact of a requested change on the entire application and on the way it meets stakeholders goals
- Tests effectiveness
  - If the tracking information system records which requirements are satisfied by which parts of the implementation, and which tests must be performed to ascertain the “presence” of a requirement, then clients can better understand the value, the results and the implications of technical tests and usability evaluations
  - In addition, acceptance testing can refer directly to the user requirements being tested for, making it relevant from a stakeholder point of view

Slides pack 1 22

Project manager and project planner (1)

- Project definition
  - An early traceability analysis during the work definition allows project managers to control that the work team and the client have the same perception of the project
  - This includes the delivered ant the not delivered artefacts, how much does it costs, who will perform the work, how the work will be done and which benefits will be achieved
- Workplan definition, development and managing
  - Matching goals with design elements is crucial to organize efficiently the time plan, giving priorities to the development of the core elements of the application and avoiding useless or superfluous features
  - Project managers can prevent conflicts and check the progresses of the different tasks related each other, with test procedures and with the main strategic goals
  - Conflicts between requirements can be discovered earlier and unexpected product delays avoided
- Communication and quality management
  - Traceability is a powerful communication mean with clients, providing to project managers arguments and evidences of the project quality in terms of satisfaction of goals, needs and expectations

Slides pack 1 23

Project manager and project planner (2)

- Documentation management
  - Traceability analysis allows complete and refine documentation and specifications
  - The traceability chain provides a preferential way to order, link and organize each document or deliverable
- Metrics management
  - All the relationships traced between parts of the application, features and services on one hand, and test procedures on the other hand, becomes crucial to give to project managers quantitative data to identify trends, support decisions and as indication of the good health of the project
- Impact analysis and reuse
  - Project planners use a tracing approach to perform impact analysis
  - Requirements can be tracked to determine the impact of a required change on the entire project, on the workplan, on other feature of the application, on goals, etc.
  - Requirements not yet satisfied by the implementation can be collected, and the work to be done to satisfy these remaining requirements can be estimated
  - Future systems will have reduced development time and effort because past implementation decisions can be reused

Slides pack 1 24

Requirements engineer (1)

- Requirements engineers keep and elicit visions, strategic goals, constraints, user profiles, etc. from stakeholders and motivations, user goals, etc. from users
- A pre-requirements specification traceability analysis is needed to keep these relationships between stakeholders and goals between users and goals, between goals and sub-goals in the refinement process and between sub-goals and requirements

Slides pack 1 25

Requirements engineer (2)

- Consistency check
  - Traceability analysis is used by requirements engineers to keep the consistency between the different information they consider, ad in particular between requirements as indications for the design from one hand and goals and constraints as source and motivations for requirements form the other hand
- Conflicts management
  - Conflicts between goals are usual, in particular between stakeholders goals and user goals
  - Traceability helps the analyst in finding a good compromise between conflicting goals, considering the relevance of stakeholders that own such goals and evaluating the impact that changes may have on other goals, sub-goals or requirements

Slides pack 1 26

Requirements engineer (3)

- Refinement management
  - During goals refinement activities it is crucial to keep all the relationships between high-level goals and derived or refined sub-goals
  - Traceability may also help in keeping an history of all the refinement changes performed in different moments of the project life-cycle and for different reasons (technology changes or constraints, budget constraints, timing, etc.)
- Prioritization
  - The traceability chain links as in a flow, stakeholders with goals and requirements
  - If all the relations are kept and updated, the requirements analyst can give a relative priority to each requirement or to groups of requirements that meet the needs of a certain stakeholder; requirements related to more relevant stakeholder should be considered with higher priority respect to others

Slides pack 1 27

Designer (1)

- Designers of software products are responsible to shape the information architecture of the application, considering the content structure, transitions between pieces of contents, interactive features, access to contents and features and navigation architecture
- To keep the consistency of the entire project, designers take in considerations goals and requirements highlighted during the requirements analysis
- Nevertheless, a major part of the final design has other motivations than requirements:
  - e.g. some elements could have pure technical reasons or being just based on “good design” principles
- Usually, part of these reasons are not recorded and part are not explicitly perceived or understood
- A traceability analysis allows eliciting hidden or unconscious knowledge and helps designers to show that the elements indicated in the conceptual design are not unusual, unnecessary or unmotivated

Slides pack 1 28

Designer (2)

- Consistency check
  - A tracking information system should record the results of design, the justification of the results, alternatives considered, and the assumptions made in a decision; therefore a traceability analysis prevents from consistency problems between different parts of the project and may help in solve inconsistencies with technical implementation or with strategic goals
- Requirements and goals compliance
  - Designers use traceability to understand dependencies between the requirements and to check whether all requirements are considered by the design
  - Therefore, they can more easily verify that a design satisfies the requirements or not. If a design element is not directly liked to a specific requirement, they can find arguments in traceability documents to justify their decisions in a more general relation with strategic goals or with non-functional requirements
  - A traceability approach force designer to ask themselves the “why” question before it is put by the client

Slides pack 1 29

Designer (3)

- “Negative” design management
  - With “negative” design I refer to the design elements that for any reason have been rejected or eliminated from the application
  - In most of the cases, the knowledge of which are these elements and why they have been deleted is crucial to measure their impact on the project
  - Traceability analysis support designers in keeping these kind of “design history”, avoiding time-consuming features that for the same reasons would be rejected and considering alternate solutions for other similar cases
- Impact analysis
  - Traces between the different elements of a project allow designers to evaluate possible consequences for changing a design feature in terms of compliance with requirements and goals or in terms of needed changes in implemented prototypes and applications
  - From another point of view, designers can understand the impact on the design of a change in requirements and take consequent decisions
  - Designers can use traceability information also to estimate the impact of a change in available implementation technology on the design assumptions and hence on the design alternatives
- Solutions acceptance analysis
  - Starting from traceability documents, designers can understand the reasons why a certain design was accepted and another rejected, even when the design was produced long time ago by a designer not present anymore
  - These reasons may relate design decisions to non-functional requirements, to unexpressed constraints or to more general stakeholder visions

Slides pack 1 30

Designer (4)

- Patterns reuse management
  - A traceability chain relates a specific need with a certain design solution
  - If the design is accepted, such a solution can be considered as a good one at least from a stakeholder point of view
  - Therefore, designers may reuse design components for similar needs in other projects because the assumptions under which the component will work are recorded in the traceability report
  - Besides, the tracking information system may become a kind of "corporate memory", i.e. a library of solutions patterns and a way to refers to specific solutions in a fast and direct way
- Design revision
  - Traceability documents keep the knowledge about the relationships between requirements and design in a structured way
  - If there is a need to tune up or to revise a former project, designer can understand and/or remember previous decisions taken and properly "adjust" the application

31

Verifier / validator

- Verifiers in large projects provides a further consistency check of the final application
- They base their job on traceability information to verify that all the strategic goals have been properly satisfied, all the requirements have been taken into account, design doesn't have goldplating, software meets with design specifications and the application have been properly tested
- Validators use traceability relationships between requirements and test plans to prove that the system "completely" meets the needs of the customer
- In addition, test procedures can be identified that should be rerun to validate an implemented change
- This saves test resources and allows the schedule to be streamlined.

32

Tester / usability inspector

- Testers perform a detail evaluation of the system technical performances
  - the application should not "crack" or generate errors in any condition of use
  - they can perform their tests in a more systematic way; e.g. they can test features in relevance order or organize tests grouping features by stakeholder or by goal they meet
  - in case of problems surfaced during the tests, they can indicate which exactly are the pieces of software or the design elements to review
  - they can also suggest a priority order for these problems based on the impact they have on the satisfaction of strategic goals
- Usability inspectors are concerned with the application "easy of use"
  - they check that the declared goals can be reached by users by the mean of the application in a efficient end effective way
  - they have to taken into account high-level goals of the product, evaluating it according to its real scope
  - they can also use entire parts of the traceability analysis to plan and prepare their evaluation: in fact, inspectors need to know dependencies between user profiles, goals and features in the application to properly test the usability of that solutions
  - as for the testers case, to usability problems can be assign a priority and the inspectors can indicate on which element of the project they have an impact.

33

Maintainer

- Maintainers "keep alive" the application
- This is particularly true for interactive and web-based applications, where being up-to-date and always adapt the communication and business channels to new user or stakeholder needs are key success factors
- Maintainers use the traceability information to decide how a required and accepted change will affect a system, i.e., which modules are directly affected and which other modules will experience residual effects
- Documenting an engineer's design rationale helps the maintainer to understand the system
- If a required change is implemented, understanding the existing solution structure helps to prevent the system from degrading
- A maintainer can this way estimate the impact of a change in requirements on other requirements, discover conflicts dependencies, estimate the impact of a change in requirements on the implementation and estimate the permissibility of a change in implementation with respect to (unchanged) requirements

34

Processes

- Define "entities"
  - elicit and define with stakeholders the objects to keep related each other, e.g. requirements, design elements, test procedures, etc.
- Capture traces
  - trace the relationships between the different elements of the trace model.
- Analyse traces
  - interpret the relationships and highlight problems or weaknesses raised out from traceability
- Represent traces
  - provide tools, procedures, checklists, etc. helping stakeholders and analysts in document, illustrate and display the traceability knowledge; summarise the results in a traceability report
- Maintain traces
  - keep tracing information up-to-date as far as new decisions are taken or any change is made to the system status.

35

Models: entities (1)

- Kind
  - Requirements
  - Goals
  - Design elements
  - Classes
  - Code
  - Test cases
- Direction
  - backward and forward
  - pre- and post-RST

36

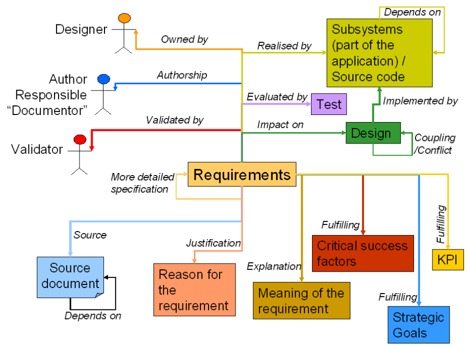
Models: entities (2)

- Attributes
  - effort
  - priority (determined by the customer)
  - source
  - status
    - proposed/approved/designed/incorporated/validated
    - captured/specified/planned/realised
    - new/assigned/classified/selected/applied/rejected
    - optional/mandatory/deleted/desirable

Models: entities (3)

- Setting
  - implicit relationships - links that do not require manual setting, e.g. name tracing, where if names and abbreviations are used in the same way and are meant to denote the same things in two documents, then a degree of traceability between them may be established
  - explicit relationships - they are manually implemented references between documentation entities and came from external considerations supplied by the developers; so, for example, the linkage, or relationship, between a textual requirement and a use case that describes the requirement is determined solely by the decision of the developers that such a relationship has meaning

Models: relationships



Tools

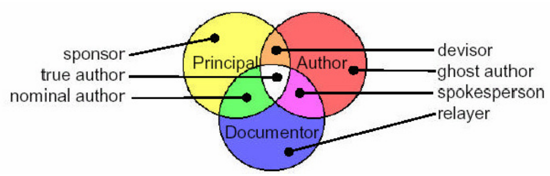
- Conceptual tools
  - Traceability matrices
  - Cross-references
  - ER models
  - Graphical models
  - Tracing languages
- Software tools
  - General-purpose tools
  - Special-purpose tools
  - Workbenches
  - Environments and beyond

Slides pack 1

- Defining traceability (G1_M1, Goal1_Module1);
- Focusing on Requirements Traceability (G2_M1);
- Showing possible traceability approaches: purposes, processes, models and tools (G3_M1);
- **Introducing examples of traceability practices (G4_M1);**
- Providing a short review of traceability software tools (G5_M1).

Contribution structures

- O. Gotel & A. Finkelstein, 1995
- Methodology to define relationships between a project artefact and its author/contributor/responsible
- According to socio-linguistic theories, the contributors can have different roles



KAOS - 1

- A. Van Lamsweerde, 1998
- Goal hierarchies express system goals and the requirements that support the achievement of system goals

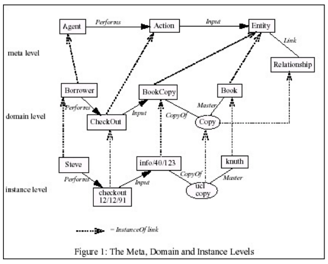


Figure 1: The Meta, Domain and Instance Levels

KAOS - 2

- The language provides a rich ontology for capturing requirements in terms of goals, constraints, objects, actions, agents, etc.
- Links between requirements are represented to capture refinements, conflicts, operationalisation, responsibility, assignments, etc.

Goal Achieve [ProgressWhenGoSignal]
FormalDef ∀ tr: Train, b: Block
On (tr, b) ∧ *Go[b+1]* ⇒ ◊ On (tr, b+1)

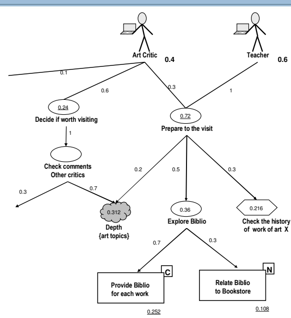
Goal Achieve [SignalSetToGo]
FormalDef ∀ tr: Train, b: Block
On (tr, b) ⇒ ◊ Go[b+1]

Distributed Intentionality (i*)

- E. Yu, 1993
- Organizational modelling framework
- Captures the intentional structure of a software process and its embedding organization
  - in terms of dependency relationships among stakeholders
- Stakeholders are represented as (social) actors who depend on each other for goals to be achieved, tasks to be performed, and resources to be furnished
- I* uses the notions of actor, goal and (actor) dependency, as a foundation to analyse high-level goals together with non-functional requirements and to model architectural and detailed design
- The i* framework includes the strategic dependency model for describing the network of relationships among actors, as well as the strategic rationale model for describing and supporting the reasoning that each actor goes through concerning its relationships with other actors
- A strategic dependency model is a graph involving actors who have strategic dependences among each other.
- A dependency describes an "agreement" between two actors; the type of the dependency describes the nature of the agreement.
- A strategic rationale graph captures the relationship between the goals of each actor and the dependencies through which the actor expects these dependencies to be fulfilled

AWARE

- D. Bolchini, 2003
- Stakeholder-based approach
- Relationships between stakeholder and goals
- Refinement traces between goals and sub-goals
- Operationalisation relationships between goals and requirements

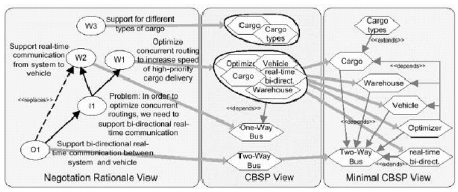


PRO-ART

- K. Pohl et al., 1994
- Process and RepOsitory based Approach for Requirements Traceability
- a tool-based requirements engineering environment
- a "focused traceability" approach that supports change integration and integrates requirements with architecture information
- The model tries to identify relationships between requirements and application architecture on the base of scenarios
- The model is conceived to define first generic traces and to specialise the most relevant ones in a second time
- The use of scenarios should facilitate the representation of user requirements, reduce the complexity, support communication with customer and interrelate requirements with architecture.
- Another element of this approach stands in the use of meta-models describing artefacts, structuring requirements information and interrelating structured information.
- The PRO-ART tool is based on three main contributions:
  - (i) a three-dimensional framework for requirements engineering which defines the kind of information to be recorded;
  - (ii) a trace-repository for structuring the trace information and enabling selective trace retrieval;
  - (iii) a novel tool interoperability approach which enables (almost) automated trace capture.

CBPS

- A. Egyed, 2000
- Component, Connector-Bus, System, Property
- Refining requirements to an initial architecture
- Identifying artefacts relevant for architecture, specifying interdependencies among artefacts and classifying it into various CBSP categories



The Potts and Bruns model

- C. Potts & G. Bruns, 1988
- An attempt to delineate the generic elements of software design rationale
  - artefacts, issues, positions, justifications, and the relations among them
- Provides a simple representation that can be tailored to different design specific methods and used for representing the process of design deliberation as well as the artefacts that result from such deliberations
- A rule-based, semi-structured, hypertext system that helps the user to examine and record design rationales easily
- A design history is kept as a network structure linking the nodes representing the different elements of their model
- A design history is regarded as a network consisting of artefacts and deliberation nodes
  - Artefacts represent specifications or design documents
  - Deliberation nodes represent issues, alternatives or justifications
- Existing artefacts, including requirements documents, give rise to issues about the evolving design.
- J. Lee, 1991
- an extension to the Potts and Bruns model consisting of enriching the internal structure of justification in the original model by making explicit the goals presupposed by arguments, the relations among arguments, and the first-class nature of these relations
- a language and a system supporting this extension of the model is also proposed

Rich Traceability

- J. Dick, 2002
- an extension to the idea of recording the rationale of a traceability relationship
- the approach encourages the use of a deeper semantics in the traceability relationship
- the author suggests to define a possible approach to richer traceability relationships, making use of textual rationale and propositional logic in the construction of traceability arguments
- the underlying logic allows other, deeper kinds of analysis to be performed
- the same structures can be applied to the management of requirements for product families as well
- the use of “exclusive or” in rich traceability provides a way of
- representing alternative ways of meeting sets of requirements
- it can therefore be used to represent the variance in system requirements addressed by different configurations of a product range

Conceptual tools (1)

- Traceability matrices
  - the horizontal and vertical dimension list the items that can be linked
  - the entries in the matrix represent links between these items
  - only binary links between items can be represented
  - easy to understand

Conceptual tools (2)

- Cross-references
  - among parts of the document
  - across different documents
  - links between documentation entities are embedded as pointers (e.g. hyperlinks or embedding phrases like "see section x") in a text
  - entities may be an informal natural language text or a formal specification
  - cross-references allow the related documents to be navigated through
  - the use is simple to understand
  - software tools that maintains cross-references and that produce reports about them can be implemented easily
  - is useful for written specifications but not for a concise representation of links such as can be done with matrices
  - cross-references are always binary links, so that links of higher arity cannot be easily represented

Conceptual tools (3)

- ER models
  - the linked items are entities, the links are relationship instances
  - links with arity higher than 2 can be represented
  - an ER model of links can be implemented using any database technology
    - ad hoc query and reporting facilities are easily available

Conceptual tools (4)

- Graphical models
  - documentation entities are represented by entities
  - relationships between them are represented by relationships
  - graphical notation (e.g. UML)
- Tracing languages
  - include DB query languages (as SQL) and regular expressions

Slides pack 1

- Defining traceability (G1_M1, Goal1_Module1);
- Focusing on Requirements Traceability (G2_M1);
- Showing possible traceability approaches: purposes, processes, models and tools (G3_M1);
- Introducing examples of traceability practices (G4_M1);
- **Providing a short review of traceability software tools (G5_M1).**

Slides pack 1 55

Software tools (1)

- The current generation of commercially available traceability tools typically provides the following functionality:
  - storage of links between items; the items may be requirements, design items, explanations, etc. and they may be represented as fixed format database records or free format text; links may be annotated, e.g. with degree of strength;
  - storage of links between texts; the texts may be requirements, documents, design documents, etc.;
  - storage of requirements in free text format with a hierarchical numbering scheme;
  - reporting facilities; examples are keyword searches, the traversal of links, producing cross-reference lists, producing traceability matrices, etc.

Slides pack 1 56

Software tools (2)

- General-purpose tools
  - include hypertext editors, word processors, spreadsheets, database management systems and prototyping tools
  - they can be hand-configured to allow previously manual and paper-based requirements traceability tasks to be carried out on-line
- Special-purpose tools
  - A number of tools support single and well-defined activities related to requirements engineering
  - Of these, some achieve restricted types of requirements traceability
  - Although there may be a limited degree of explicit control and guidance, support is generally implicit in the use of the tool, which automates any mundane and repetitive tasks needed to provide this requirements traceability

Slides pack 1 57

Software tools (3)

- Workbenches
  - Typically centred around a database management system of some form, these software types comprise dedicated tools for documenting, parsing, editing, interlinking, organising, and managing requirements
  - They often provide facilities to help assess and carry out any changes made to these requirements
- Environments and beyond
  - Requirements traceability can potentially be provided throughout a project's life if tools supporting all aspects of development are integrated
  - The basis used for internal integration tends to define how requirements traceability is established: through the use of a common language
  - Those with the flexibility to incorporate third-party environments tend to provide requirements traceability support through the use of powerful repositories and underlying database management systems

Slides pack 1 58

Examples (1)

- **Analyst Pro by Goda Software, Inc. (http://www.analysttool.com)**
  - Analyst Pro uses a requirements management methodology that covers the entire life-cycle including, from the initial requirements-gathering phase through the separation phase where requirements and non-requirements are set apart. Analyst Pro utilizes a Configuration Management methodology that enables the development staff to analyze the impact of change on requirements and component assets. Analyst Pro incorporates the following features:
    - Importing Requirements - Analyst Pro allows users to import requirements from existing documents from various formats (doc, html and text).
    - Requirements Sharing - Analyst Pro allows users to share and trace requirements across projects.
    - Requirements Change Management - Analyst Pro automatically records and lists any changes to your project, when the changes were made and who made the changes.
    - Requirements Assignment - Users can assign requirements to team members and track its status.
    - Requirements Graphs - Users can create pie and bar graphs with a number of requirements versus attributes. The attributes include priority, version, status and source.

Slides pack 1 59

Examples (2)

- **CaliberRM by Borland (http://www.borland.com/caliber/index.html)**
  - Caliber-RM is a collaborative, Web-based requirements management system that facilitates communication among project teams by providing centralized requirement data to distributed team members and allowing documented discussions about requirements as well as allowing project teams to fully define, manage and communicate changing application or system requirements. Changes made to requirement data such as traceability, document references, status, user responsibility and more are recorded in Caliber-RM's central repository. CaliberRM keeps team members up to date on changes made to requirements by automatically notifying responsible individuals of the changes. CaliberRM also enables team members to quickly identify potential requirement problems by highlighting ambiguous and commonly used terms defined in a shared glossary. The latest version of CaliberRM provides LiveLink integration with Caliber-RBT so that requirements in Caliber-RM can be associated with corresponding cause-effect graph files in Caliber-RBT. CaliberRM allows project teams to provide input on requirements via standard browsers and remote clients can access the system through an Internet connection.

Slides pack 1 60

Examples (3)

- **DOORS/ERS by Telelogic (http://www.telelogic.com/products/doorsers)**
  - DOORS (Dynamic Object Oriented Requirements System) is an Information Management and Traceability (IMT) tool. Requirements are handled within DOORS as discrete objects. Each requirement can be tagged with an unlimited number of attributes allowing easy selection of subsets of requirements for specialist tasks. DOORS includes an on-line change proposal and review system that lets users submit proposed changes to requirements, including a justification. DOORS offers unlimited links between all objects in a project for full multi-level traceability. Impact and traceability reports as well as reports identifying missing links are all available across all levels or phases of a project life cycle. Verification matrices can be produced directly or output in any of the supported formats including RTF for MS-Word, Interleaf and FrameMaker. The DOORS Extension Language (DXL) is a high level C-like language that provides access to virtually all DOORS functions for user extensions and customization. DOORS includes the following functionality:
    - Control of data model for process management allows user to manage the relationship between data fully including its direction, type and even whether a relationship is allowed.
    - Improved security control through the use of passwords, and timeouts which "lock up" DOORS after a specified period of inactivity.
    - New templates to make document generation easier have been added to the DOORS template library. New templates include ISO 12207, ISO 6592 and IEEE software standards.

Slides pack 1 61

Examples (4)

- **IRqA (Integral Requisite Analyzer) by TCP Sistemas e Ingeniería (http://www.irqaonline.com)**
  - IRqA is a state-of-the-art Requirements Engineering (RE) tool specifically designed to provide an integral support to the complete Requirements Engineering process. In IRqA the complete specification cycle is supported via standard models:
    - Requirements Capture
    - Requirements Management
    - Requirements Analysis
    - System Specification building
    - Specification validation (specification vs requirements)
    - Acceptance Tests management
    - Requirements Organization & Classification

Slides pack 1 62

Examples (5)

- **Rational RequisitePro by Rational Software (http://www.rational.com/products/reqpro/index.jsp)**
  - RequisitePro is a requirements management tool designed for multi-user environments. It features integration of Microsoft Word and a requirements database. Software project teams can gather, enter and manage requirements "in situ" (within your documents) or in a database. Automated traceability tracks requirements and changes through implementation and testing. Related requirements can be linked together, so that as changes occur to one requirement users can easily see its impact on other related requirements. RequisitePro includes templates to simplify production of requirements documents. Rational RequisitePro supports a choice of databases (Oracle, Microsoft SQL Server, and Microsoft Access) which allow users to organize, prioritize, and trace relationships between requirements. Version 2001A includes the ability to treat linked files as a requirement and trace other requirements to your linked files.
  - RequisitePro also provides various views to enhance traceability. One of those views is the Traceability Matrix. This matrix displays requirements in a matrix format for easier coverage viewing. The matrix will provide visual feedback about what system requirements were derived from which customer requirements. Using the matrix, it is also easy to check coverage and make sure that all of the customer requirements were broken down into system requirements. Another useful view provided by RequisitePro is the Traceability Tree view. This view shows the requirements in a hierarchical fashion. The benefit of this view is in graphically showing relationships between requirements. If a requirement is modified, added or deleted, the user can visually see all of the other affected requirements. The affected requirements can then be properly scrutinized and modified to accommodate the original requirement change. This helps maintain a cohesive set of requirements by eliminating orphaned requirements and also by preventing outdated requirements from being left in the set.
  - RequisitePro also offers cross project traceability. Often times, especially with legacy systems, a number of projects will spawn off of a central project. These new projects will share a significant number of requirements with its parent and sibling projects. RequisitePro allows traceability of requirements to span cross-project. This greatly increases requirement reuse which can in turn foster design, code, and test reuse.

Slides pack 1 63

Examples (6)

- **RDT (Requirements Design & Traceability) by Igatech (http://www.igatech.com)**
  - RDT supports several mechanisms to aid the user in requirements analysis and identification. These include a parser that imports text documents then identifies requirements by key words and structure. The tool provides functionality for deriving, allocating and assigning requirements and acceptance test procedures. Requirements can be traced from top level requirements down to the lowest level requirements. The tool is able to classify/categorize requirements during identification using requirements attributes. In addition the tool provides capabilities to capture architecture, functional decomposition and WBS in graphical format and display data as a tree view of requirements. RDT is able to generate documentation directly into MS Word, including requirements and test specifications, requirement allocation matrices, parent-child relationships and design documents. New features incorporated in version 3 include:
    - The ability to share data between different sites, and the facility to collate this data back to the master database.
    - Revision control, which allows users to look at all changes made to data, and when and by whom these changes were made.
    - An RDT AxiomSys Bridge exists that allows the bi-directional transfer of requirements and tests between any part of the project database in RDT, and the software or system model(s) in AxiomSys 6.0.

Slides pack 1 64

Examples (7)

- **RTM (Requirements Traceability Management) by Integrated Chipware Inc. (http://www.chipware.com)**
  - RTM supports multiple users working on the same requirements at the same time by implementing locking control on a requirement-by-requirement basis. RTM's toolset supports the ability to capture graphical information as traceable requirements objects. The tool utilizes the native tool, which created the graphics object. A class definition tool is included that allows the user to model any type of hierarchical project data (requirement document, hierarchies, system element structure and WBS). Once the hierarchy is defined generic relationships can also be established to allow cross-reference link information to be established between any active data item. Version 5.3 of RTM includes the following capabilities:
    - An information modelling capability allows users to design change records or problem reports and associate them with specific requirements data.
    - A complete test management solution including information concerning schedules, resources, test verification and results versus requirements.
    - User defined forms to allow users to view information in familiar layouts.
    - Change request capability allows users to propose and review changes to the current baseline requirements from within RTM.

Slides pack 1 65

TRAMA: Traceability Analysis Method for (interactive) Applications

Slides Pack 2

Giovanni Randazzo

25 slides

Slides pack 2

- Introducing TRAMA basic features and concepts (G1_M2);
- Presenting TRAMA activities workflow (G2_M2);

Slides pack 2

- **Introducing TRAMA basic features and concepts (G1_M2);**
- Presenting TRAMA activities workflow (G2_M2);

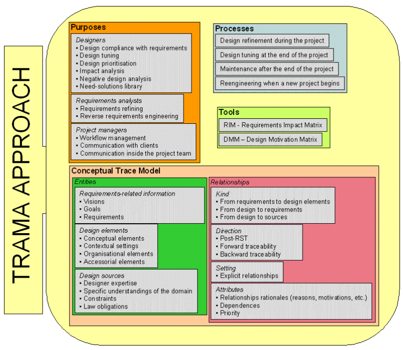
TRAMA: a traceability analysis method

- TEC-Lab, Unversità della Svizzera Italiana, Lugano (Switzerland)
- HOC – Hypermedia Open Center, Politecnico di Milano (Italy)
- Design traceability method supporting both backward and forward traceability
- Provides to designer an effective tool conceived to discuss and analyse the design choices after they have been taken, in order to refine it according to the main requirements and in order to eliminate unmotivated elements

TRAMA: main concepts

- **The design of the application solutions may not derive directly from requirements refinement**
- **Designing is an intuition and induction process more than a derivation one**
- **The TRAMA tracing activity tries to move intuition and induction to more rational cause-effect motivations**
- **The method forces to better make explicit requirements that are both implicit or unexpressed**

TRAMA approach



TRAMA: purposes for designers

- Compliance checking
  - in order to check the compliance of design elements with requirements
  - with this method one can understand if a particular element of the design answers to one or more stakeholders' needs
- Design "tuning"
  - one can base on compliance checking and on design motivations analysis to correct and refine or to reengineer the design according to strategic goals
- Design prioritisation
  - in order to evaluate the relative weight and the effort request for a design element according to requirements compliance, simplifying or enriching that element
- Impact analysis
  - one can evaluate the impact of a requested change, analysing dependences between design elements and requirements, constraints, visions, etc.
- "Negative" design tracing
  - in order to keep trace of choices that for any reason have been rejected or eliminated from the application, avoiding to discuss again these solutions in future development of the project
- Solutions patterns
  - one can keep a library of effective need-solution pairs

Slides pack 2 7

TRAMA: purposes for Requirements analysts

- Requirements refinement
  - in order to refine the requirements specification
  - some requirements are sometimes let implicit or they are not recorded in a document, even if they are not obvious or trivial ones
  - key requirements that have not been explicitly discussed in the analysis phase may surface in a TRAMA analysis considering design motivations
- Reverse requirements engineering
  - if requirements have not been documented in previous analysis phases, the requirements analyst can use the TRAMA information for a reverse engineering activity, understanding requirements from design and motivations

Slides pack 2 8

TRAMA: purposes for project managers

- Workflow management
  - for a better control of the overall project
  - the method provides a global picture of the entire project, highlighting the relationships between its different pieces and the reasons why those decision were formerly taken
  - these elements are crucial to organize efficiently the time plan, giving priorities to the development of the core elements of the application and avoiding useless or superfluous features
- Communication with clients
  - TRAMA is a huge communication mean with clients, providing to project managers arguments and evidences of the project quality in terms of satisfaction of goals, needs and expectations
- Communication inside the project team
  - TRAMA is a powerful communication tool for project managers and for designers that work on different elements of the application
  - while each designer develops a single application feature, a wider understanding of how requirements and educational goals are considered in the design is needed to refine and improve these solutions
  - in order to keep the "fil rouge" of the decisions taken during the project, understanding which elements cannot be modified and which ones may be altered in the revision process
- Documentation "tuning"
  - the TRAMA analysis allows complete and refined documentation and specifications
  - the traceability chain provides a preferential way to order, link and organize each document or deliverable

Slides pack 2 9

TRAMA: processes (1)

- Refinement
  - a first design attempt is usually shaped by designer taking in account requirements as background information
  - in this phase, solutions are not conceived in a full explicit way and requirements are not considered one by one
  - this practice is not necessarily negative, since the results are often quite good in relation to stakeholders needs
  - TRAMA may be applied during the project after a first design has been produced
  - TRAMA tries to trace *ex-post* relationships surfacing motivations for design choices
  - the method may be applied to understand the explicit or implicit, conscious or unconscious reasons for solutions proposed in the design
  - these traces help designers in the refinement activity, i.e. in adjusting the first design attempt according to requirements and priorities in a explicit and structured way

Slides pack 2 10

TRAMA: processes (2)

- Tuning
  - Even if the main design effort is done in the design phase, some adjustment must be always be performed during the entire project's life-cycle because of
    - technology limitations
    - business or resources constraints
    - new requirements
    - changes in requirements
  - TRAMA helps designer in the tuning activity
  - It keeps traces of old design solutions and of reasons for changes
  - Design to requirements relationships allow designers to understand the impact of a changed requirement into the application

Slides pack 2 11

TRAMA: processes (3)

- Maintenance
  - After an application production project is ended, a continuous maintenance activity is needed to "keep alive" the application through the years
  - In particular, a real life use of the product by the final users and its effects on the organisation and on the business of the company, make clear if all the solution proposed were actually good and effective solutions
  - If this is not the case or if some changes occurs in the company (e.g. new constraints, new requirements, etc.), the application needs to be revised, updated or changed
  - TRAMA helps project managers and designers in adjusting weak solutions or in conceiving new solutions for old or new needs, understanding the impact of these changes on the overall application and on its compliance with requirements

Slides pack 2 12

TRAMA: processes (4)

- Reengineering
  - A new project in order to insert major modifications in the application or in order to develop a new application
  - Because of
    - new needs
    - new requirements or
    - more in general, new relevant elements for the company
  - In these cases, the old application may be simply tuned-up or completely reengineered
  - TRAMA helps to understand or to remember why certain solutions have been formerly adopted even if some years have passed and if there is a new project team
  - TRAMA allows also to organise the redesign activity according to old dependencies with requirements, identifying the elements that can be improved and the “untouchable” elements linked to still valid goals and that should not be changed

13

TRAMA: entities (1)

- Requirements-related information
  - Visions
    - correspond to a strategic insight of a stakeholder in the domain
    - provide a way for modelling the assumptions of a stakeholder which dictate her “weltanschaung” on the project
  - Goals
    - a wished state of affairs for the main stakeholders
    - a wished experience or an expectation for a class of users
  - Requirements
    - sufficiently high-level descriptions of properties or functionalities of the application as input for the design activity

14

TRAMA: entities (2)

- Design elements
  - Conceptual elements
    - the traditional conceptual design elements
    - content and structure of content, navigation architecture, access paths, operations, pages and layout, etc.
  - Contextual settings
    - e.g. the technical equipment, the place where the application is used, the physical disposition of machines in this place, etc.
  - Organisational elements
    - e.g. how different use sessions are organised during a week, which activities are implies in the use of the application, etc.
  - Other accessorial elements
    - e.g. study material needed to use the application (in a educational system), etc.

15

TRAMA: entities (3)

- Design sources
  - The designer expertise
    - i.e. particular “good design” principles that are part of the designer’s skills and that she/he applies in any case
  - A specific understanding of the domain
    - i.e. recurrent good solutions in a domain that the designer applies because she/he learnt it by other cases in the same domain
  - A particular constraint
    - e.g. budget limitations, time, technology limitations, etc.
  - A law obligation
    - e.g. copyright issues, personal data treatment, etc
  - A requirements-related information
    - i.e. a vision, a goal, a requirements, etc
  - An arbitrary choice
    - i.e. a choice without particular reasons, usually a single detail that should anyway be set in a way or another, e.g. the structure of a game in three steps (instead of four or two).

16

TRAMA: relationships

- Kind
  - from requirements to design elements, tracing the impact of requirements on the design
  - from design to requirements, tracing the justification of design solutions
  - from design to its sources, tracing the motivations for design choices
- Direction
  - TRAMA is a post-Requirements Specification Traceability method
  - supports forward traceability from requirements to design elements and backward traceability from design elements to requirements or to other motivations
- Setting
  - Due to the its own nature, TRAMA allows only explicit relationships; the method does not include any implicit or automatic generation of traces
- Attributes
  - rationale of the relationship
  - dependence with another trace
  - priority value

17

TRAMA: analysis tools

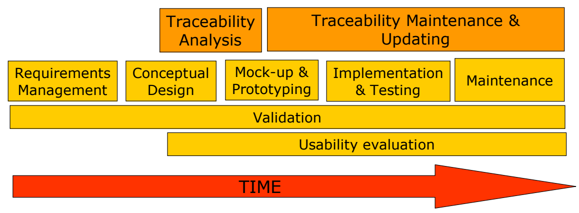
- TRAMA is based on traceability matrices
  - cross requirements with design in a forward direction
  - cross design with its sources (requirements, motivations, constraints, etc.) in a backward direction
- RIM (Requirements Impact Model/Matrix)
  - Requirements-to-Design matrix
  - can be filled and read horizontally, highlighting how single requirements are taken into account into the design
  - can be filled and read vertically, showing how a single design element satisfies the project requirements
- DMM (Design Motivations Model/Matrix)
  - Design-to-Sources matrix
  - traces back single design elements to the motivation why a certain decision is relevant for the project
  - e.g. satisfying a requirements, fulfilling a constraint, allowing more usability in the system, etc.

18

Traceability phase

- In which moment of the project life-cycle?
- relevant information can be surfaced after a first version of the design is produced
- a detailed design is possibly needed to profitably trace relationships towards high-level requirements
- Suggestion:
  - perform a tracing activity after the first design phase
  - a continuous activity during the rest of the project is then needed to maintain the traceability specification up-to-date

Traceability in the project's life-cycle



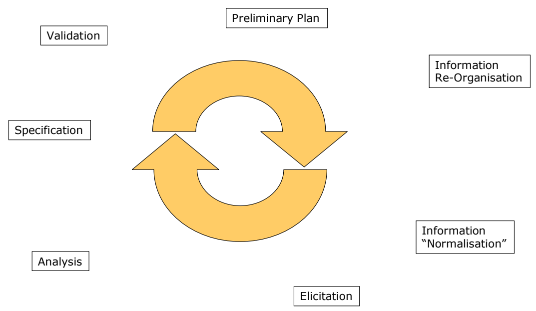
Slides pack 2

- Introducing TRAMA basic features and concepts (G1_M2);
- **Presenting TRAMA activities workflow (G2_M2);**

TRAMA components

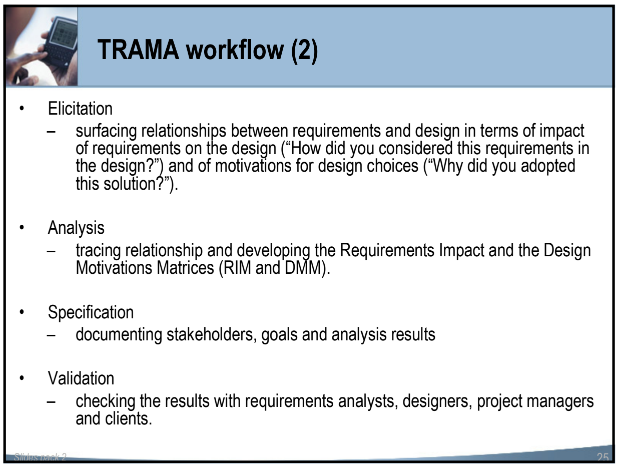
- The approach consists in
  - a structured analysis process
  - a general conceptual model of entities and relationships to trace
  - a set of conceptual tools supporting traces inquiry, analysis and documentation

TRAMA Analysis Process



TRAMA workflow (1)

- Preliminary plan
  - understanding which the stakeholders of the traceability analysis, the traceability goals, the constraints (time and budget, related to ROI) and the expected results are
- Information re-organisation
  - understanding requirements and design from documents or from interviews with designers and organise it in terms of structured specifications
- Information "normalisation"
  - structuring requirements and design information in "normal" terms, base on a strong methodology (e.g. AWARE for requirements and IDM for design)


TRAMA workflow (2)
• Elicitation
– surfacing relationships between requirements and design in terms of impact of requirements on the design ("How did you considered this requirements in the design?") and of motivations for design choices ("Why did you adopted this solution?").
• Analysis
– tracing relationship and developing the Requirements Impact and the Design Motivations Matrices (RIM and DMM).
• Specification
– documenting stakeholders, goals and analysis results
• Validation
– checking the results with requirements analysts, designers, project managers and clients.

TRAMA: Traceability Analysis Method for (interactive) Applications

Slides Pack 3

Giovanni Randazzo

70 slides

Slides pack 3

- Detailing activity 1: preliminary plan (G3_M2);
- Detailing activity 2: information re-organisation and normalisation (G4_M2);
- Detailing activity 3: elicitation and analysis (G5_M2);
- Detailing activity 4: specification and validation (G6_M2);
- Introducing examples where TRAMA has been applied (G7_M1);
- Wrap-up the main concepts (G8_M1).

Slides pack 3

- **Detailing activity 1: preliminary plan (G3_M2);**
- Detailing activity 2: information re-organisation and normalisation (G4_M2);
- Detailing activity 3: elicitation and analysis (G5_M2);
- Detailing activity 4: specification and validation (G6_M2);
- Introducing examples where TRAMA has been applied (G7_M1);
- Wrap-up the main concepts (G8_M1).

Preliminary plan

- Tracing requirements and design is a complex set of activities
  - Can have very different purposes
  - For a number of actors in a project's life-cycle
- The kind of information recorded and the analysis results differ due to
  - actors considered
  - objectives considered
- From this point of view, a kind of "requirements analysis" for the traceability phase is required
- Understanding which are
  - the stakeholders of the traceability analysis
  - the traceability goals
  - the constraints (time and budget, related to ROI)
  - the expected results

Preliminary plan activities

- Stakeholders
  - the specification of who the stakeholders of this phase are
- Goals
  - a precise definition of all the goals that this activity is intended to reach,
- Constraints
  - all the time and budget constraint related to this phase
- Expected results
  - which are the expected results of the traceability analysis.
- Scheduling
  - a setup of the tracing activity

Stakeholders

- In the context of the traceability preliminary plan, a stakeholder can be defined as any actor of the application development-related activities which has a specific interest or goal in the results of the traceability analysis
- The analyst discuss with a decision maker (the client or the project manager) about who are the people to involve in the tracing activity, both as sources of goals and opinion about the activity or as sources of useful information to surface traces between requirements and design elements
- Stakeholders may therefore be the client itself, the project manager, the project planner, the requirement analysts, the designers, the verifiers and/or the maintainers of the application
- TRAMA has been tested with success for three particular kind of people:
  - project managers, who use traceability information to control project progress and as communication tool with the client;
  - requirements analysts, who use traceability information to check, refine and update the requirements specification;
  - designers, who use traceability information to keep the consistency between design and requirements and to check the design compliance with strategic goals.

Goals

- As for every phase of a project's life-cycle, a precise definition of which the goals and the needs to fit are, is an essential element for the success of the phase itself
- In this case, this is even more so true because of the variety of the possible purposes of a tracing activity
- The analyst has here the responsibility to highlight what it can be done and what it cannot be done with such an analysis
- During a first meeting with stakeholders, the analyst have to elicit the objectives of this activity, and have to help in selecting the aspects that could be more relevant for the stakeholders' needs
- An "all purposes" analysis is not realistic in any case:
  - first, time and budget could be serious constraints that limit the possible actions to perform during this phase;
  - second, the experience shows that the more the traceability analysis' goals are focused, the more that analysis may be effective in terms of ROI
- Pragmatically, the "magic number" of goals for this activity should be included between 2 and 4
- More than four different goals risks to cause an activity overload and a negative costs/benefits balance.

Constraints

- Needs and desires often have to face with the actual resources provided for a certain activity
- Tracing is not an exception: there will be always limitations of time and of budget in order to perform traceability in projects where the money spent is under strict control and where time-to-market is a quality measure of the production process
- The preliminary plan have to define precise terms the effort needed for this phase, detailing the expected number of man/months, the number of days planned and the estimated cost for a traceability action in the project
- Other possible constraints that the analyst have to preliminarily consider, may be particular law obligations (e.g. privacy issues) or other organisational elements

Expected results

- A central element of a traceability preliminary plan is to define the expectation of the stakeholders about which benefits would the analysis bring to the project
- The analyst should manage carefully these expectations, discussing it precisely in order to reach a common vision about what the tracing activity would give to the project
- Different expectations about the results are usually the reason of a different perception of the success of this activity

Scheduling

- Definition and setup of the subsequent tracing activity
- Once defined the people to talk with to get and give information, their goals and expectations and the constraints included, the actual traceability phase can be performed in a structured way
- According to the TRAMA method, this activity has to be carried on in strict collaboration with the different stakeholders
- In particular, some meetings with project managers and designers have to be planned to elicit traceability information
- These meeting can be planned only at this point because their number and duration depends on the activity's goals and expectations
- Therefore, a complete activity plan have to be defined and described, including
  - a meetings calendar
  - the main analysis phases
  - milestones, time, effort and costs for each phase

Slides pack 3

- Detailing activity 1: preliminary plan (G3_M2);
- **Detailing activity 2: information re-organisation and normalisation (G4_M2);**
- Detailing activity 3: elicitation and analysis (G5_M2);
- Detailing activity 4: specification and validation (G6_M2);
- Introducing examples where TRAMA has been applied (G7_M1);
- Wrap-up the main concepts (G8_M1).

Information re-organisation (1)

- TRAMA aims at discovering relationships between requirements and design and between design and its motivation
- To clearly discuss about this relationships with stakeholders and to avoid misunderstandings, it is needed to have structured and ordered elements both form the requirements and from the design side
- In a perfect world
  - requirements information are explicitly organised and recorded during the project analysis phase
  - this specification is continuously updated during the project development
  - design is step-by-step documented in formal schemes
  - it is always keep aligned with the actual application implemented

We do not live in a perfect world... (1)

- The requirements specification may be unstructured or incomplete
  - the penetration degree of requirements management approaches in industrial practices is very low
  - in most of the cases, unstructured and informal approaches are used to record the information raised out form the firsts operative meetings
  - sometimes there is not a clear and univocal perception of what a "requirement" or what a "goal" is
    - technical details of the applications?
    - high-level visions related to a topic?
    - business-related expectations?
    - application-related desires?

We do not live in a perfect world... (2)

- The requirements specification may be absent
  - In the worst cases, the requirements specification is not only confused or unstructured, but completely absent
  - In some projects the first recorded sign of what goals and requirements were, is the description of how the application is made
  - In frequent cases, the requirements specification is not used as a base to design the application, but it is an ex-post documentation used to describe the backgrounds of an existing product

We do not live in a perfect world... (3)

- The design documentation may be absent or incomplete
  - Often it is not clear what a design for an interactive application is
    - it should describe all the technical implementation details?
    - it should be a conceptual picture of the applications contents, functionalities, navigation, etc.?
  - Sometimes this kind of specification is completely absent
    - Just a technical documentation of how the application has been programmed is provided
  - In other cases, an unstructured specification of the elements of the application design is produced
    - but it includes a mix of indistinct contents, operations, navigation capabilities, organisation elements, roles, etc.

Information re-organisation (2)

- Sometimes, of course, requirements and design specification are recorded with scientific and formal approaches
- Anyway, the TRAMA method cannot take this eventuality for granted but it should consider all the possibilities that can be encountered in the real world
- TRAMA can therefore be applied anyway, no matter if there is previous documentation or not.

- Information re-organisation consists in understanding requirements and design information before to start the tracing process
- The traceability analysts has somehow to understand what the goals, the requirements, etc. of the project and what the designed contents, functionalities, etc. of the application were
- She/he has to "pick up" and to organise these elements in a requirements specification and in a design document.

Information sources

- Specific interviews or focus groups with requirements, analysts, designers, project responsible or other members of the work-team

- Existing documents, specifications, reports, minutes or annotations of some project meeting or activity

- A reverse engineering activity, extracting the design form the actual application or (more difficult) inferring requirements from the design

Information "normalisation"

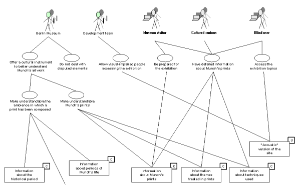
- The knowledge gathered during the information re-organisation activity can be documented pragmatically using no matter what approach
- The approach adopted have to answer to the needs of clarity, simplicity and correctness in terms of information structure

- Normalisation = Structuring requirements and design information in "normal" terms, base on a strong methodology
- A normal form is a representative element within an equivalence class, which is a simples or most manageable or otherwise tidiest and most desirable form, in terms of structure or syntax

- TRAMA distinguish between the sub-activities of
  - requirements normalisation and
  - design normalisation.

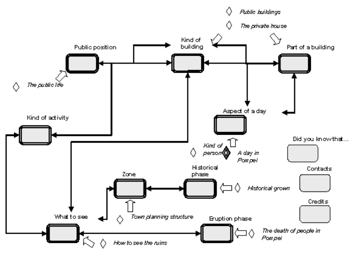
Requirements normalisation

- Structuring the requirements-related information in a "normal" form
- Requirements information should be transformed in a more manageable form in order to be traced towards the design
- A goal-oriented methodology is suggested
  - structure the knowledge in terms of goals, goals refinement and requirements
- e.g. AWARE



Design normalisation

- Transforming the design knowledge gathered during the information re-organisation activity in terms of structured design
- e.g. IDM



Slides pack 3

- Detailing activity 1: preliminary plan (G3_M2);
- Detailing activity 2: information re-organisation and normalisation (G4_M2);
- **Detailing activity 3: elicitation and analysis (G5_M2);**
- Detailing activity 4: specification and validation (G6_M2);
- Introducing examples where TRAMA has been applied (G7_M1);
- Wrap-up the main concepts (G8_M1).

Elicitation

- Elicitation is to call forth or draw out as information or a response something latent or potential
- Elicitation is the process of identifying needs and bridging the disparities among the involved communities for the purpose of defining and distilling requirements to meet the constraints of these communities
- Elicitation is the activity of surfacing relationships between requirements and design in terms of impact of requirements on the design ("How did you considered this requirements in the design?") and of motivations for design choices ("Why did you adopted this solution?")

Elicitation techniques

- Create an environment where stakeholders feel at their ease and are able to demonstrate ideas
- Combine different techniques:
  - Interviews
  - Focus groups
  - Questionnaires
  - Direct observation

Interviews (1)

- Very common for this kind of activity because they allow a "live" contact with a person that could be a source of information
- Here everything depends on the interviewer's skills and on the right selection of people to talk with
- In large projects where many people are involved, this activity could take a lot of time

Interviews (2)

- Benefits
  - When "few" people each know a "Lot"
  - Gather RICH information
  - Insights about stakeholder's perspectives
  - Insights about the culture and the domain
- Tips
  - Allow people showing material, examples and demonstrating their ideas
  - Trade-off between listening, guiding and intrusion
- Drawbacks
  - Time consuming
  - Miss interaction between stakeholders

Focus groups (1)

- Discussion meetings between the traceability expert and the project's work-team
- It is possible a "live" contact with people working on the project
- It is not so focused as in an interview
- New knowledge may raised out from group discussion
- A single meeting or a couple of meetings do not take so much time

Focus groups (2)

- Benefits
  - New knowledge from discussions and interaction
  - Good both for brainstorming and focus groups
  - Everybody need to explain ideas for other to understand
- Tips
  - 3-20 stakeholders in one room
  - Analysty offers issues and questions
  - Every one should feel accepted and involved in
- Drawbacks
  - Difficult to fit in the stakehoders' agenda
  - Only "public" opinion emerge
  - Risk to be conflict-driven

Questionnaires (1)

- Can be used as a preliminary step in focus groups or interviews
- To set up the discussion agenda
- Where too much people are involved in the project
  - interviews for the two or three project main responsible
  - questionnaires for all the other project workers

Questionnaires (2)

- Benefits
  - Quantify and compare data
  - Large sample at low cost
  - Appear scientific due to statistical data
- Tips
  - Should be short
  - Alternate open and close questions
- Drawbacks
  - No time for explanation, solve misunderstanding and provoke "habit change"
  - No human touch
  - Focussed aswers to specific questions only
  - Short time causes poor reflection and knowledge evocation

Direct observation (1)

- Following the entire project form the beginning can be an option in case of high budgets and large projects
- Here a traceability expert follow the different project's phases as an internal observer and debrief step-by-step the motivations why the application is designed in a certain way
- This technique presupposes many time and resources to be performed

Direct observation (2)

- Benefits
  - Stakeholders are observed while doing their job
  - Insight about actual process, work context and time
  - Elicit tacit knowledge and automatic processes
- Tips
  - Be as passive as possible
- Drawbacks
  - **Hawthorne effect**: people aware of being watched act differently than they do when unobserved

31

Analysis

- Taking all the information surfaced by the different elicitation practices performed (interviews, focus groups, etc.)
- Gathering all this knowledge in a structured and analytical picture
- Different points of view have to be integrated:
  - The designer's point of view
  - The client/customer's point of view

32

The designer's point of view

- Each designer develop different parts and different functionalities of a same application
- His/her perception of the project is often limited to a "vertical" view on how these parts and functionalities answers to the strategic needs
- The traceability analysis have to gather all these partial views, showing how the entire application fits with requirements through the inter-action of its different parts.

33

The client/customer's point of view

- Often this point of view is mediated by the project manager
- The focus here is how a single requirement has been taken into account in the application development
- The analyst have therefore to consider all the information gatherer from an "horizontal" point of view, documenting the impact that all the strategic needs (expressed by goals and requirements) have on the application design

34

TRAMA analysis aspects

- Designer's and client's points of view are mirrored in two aspects taken into account by the TRAMA analysis:
  - the justification or motivation of the design (designer's point of view), that can comes from requirements or from other kind of sources (an understand of the specific domain, the expertise of the designer, a constraint, etc.);
    - these traces are called Design Motivations Model (DMM)
  - the impact on design (client's point of view) of: visions, stakeholders-goals, users-motivations, domain issues, scenarios, constraints and requirements
    - these traces form the Requirements Impact Model (RIM)

35

Client validation

- To set up a structured argumentation to show to the client that all the needs have been taken into consideration
- This activity is supported by a proper traceability approach
  - in a forward direction
    - showing which requirements have been taken into account in the design and how
    - following evolving requirements in design
    - checking consistency and feasibility of requirements
    - estimating the impact of a change in requirements on the design
  - in a backward direction
    - finding arguments to justify design decisions
    - checking whether all requirements are considered by the design
    - and estimating the effect of a required design change
- In TRAMA the RIM matrix allows project managers and designers to map each goal and each requirements into design solutions, providing a powerful communication tool to show that everything (every strategic goal, etc.) has been considered in the application

36

Design versioning

- To highlight different design areas for different stakeholders.
- A proper backward traceability approach allows understanding which parts of the design are relevant for which stakeholder
- The design-requirements-goals-stakeholders chain helps creating different versions of the design documentation, addressed to specific targets

- If requirements are normalised with a proper goal-oriented methodology (e.g. with AWARE), each goal is linked to the stakeholder(s) who owns it
- Goals-to-design relationships in the RIM matrix allows to identify the application elements that satisfy the goals of a specific stakeholder

Non-traceable design

- To document the motivations of design elements that do not derive from requirements

- A big part of the design elements are not motivated by a requirement-related information
- Most of the choices come from usability or “good design” principles or are just due to the designer’s expertise

- The DMM matrix allows to distinguish the different motivations for design elements, relating design with its sources types and answering to the “why” question (“Why this design element has been placed into the application? Why in this way?”)

“Negative” design

- With “negative” design I mean those design objects that have been eliminated or modified during the project life-cycle
- Proposed elements in the application may become part of the negative design
  - because of a direct rejection
  - because of a change in related objects
  - because of business, technology or law constraints
- Keeping trace of old design versions and understand and remember former design decision is useful to
  - remember why a decision and not another has been taken
  - validate negative decisions with stakeholders
  - understand why a design decision has been rejected
  - show the “negative” impact of a specific constraint or requirement on design

- Rejected design choices can be (separately) listed in the DMM matrix
- The crosses with the different sources types answer to the “why not” question

Reverse requirements specification (1)

- Check the consistency between design and requirements
- “Tune” requirements specification according to the real stakeholders’ goals
- Extract consistent requirements specification from design.

- Sometimes requirements specifications are written after design or after implementation phase, just for documentation
- In these cases, a proper traceability approach may help in producing an effective requirements specification according to the real stakeholders’ goals and requirements

Reverse requirements specification (2)

- “Ex-post” traces are anyway useful to
  - check the consistency between design and requirements
  - tune up existing requirements specification according to the actual application
  - extract consistent requirements specification from design
- Such a reverse requirements specification is a beautiful tool to
  - keep trace of strategic decisions
  - provide design decisions with argumentations
  - collect information and material for a consistent usability test

- This activity is supported by the RIM matrix that force analysts in surfacing consistency or inconsistency traces.

Usability on design documents

- Select the elements in the design involved for a specific task
- Evaluating the quality of the product with respect to the high-level goals
- Identifying test procedures that should be rerun to validate an implemented design change

- The RIM matrix allows usability experts to perform inspections on specific design areas, properly considering the strategic goals that should be fit by those inspected elements

Analysis checklist

- TRAMA provides to analysts a set of pragmatic questions
- They can be used as a guide or as a checklist to properly consider all the aspects involved in a relationship between project elements
- For each cross in a matrix the traceability expert should ask himself:
  - "Which design element fits with the needs of this stakeholder?"; "If I had to present the project to this stakeholder, which part of the design should I highlight?";
  - "Which design element fits with this goal?"; "Which is the impact of this goal into the design?";
  - "Which design element better fits with the needs of this user?"; "How can I arguing design choices to show that this user is considered in it?"
  - "Which strategy is set-up in the design to fit with this user motivation?";
  - "Which is the (positive or negative) impact of this constraint into the design?";
  - "Which are the design elements that fit with this requirement?"; "How can I show that this requirement has been properly taken into account in the design?";
  - "Why the designer chose to put this element into the design?"; "How can I show that this element is not an extra-feature in the design?";
  - "Why this element has been rejected or modified in the current design?"; "What is the impact of this choice into the project consistency with strategic goals?"

RIM – Requirements Impact Matrix

- Lists vertically requirements-related information and horizontally all the design elements
- Highlights the impact on design of visions, stakeholders goals, users motivations and requirements
- Vertically
  - information on how single requirements are taken into account into the design
  - "Taking into account a single design element, how does it fit with requirements?"
- Horizontally
  - how a single design element satisfy the project requirements
  - "Taking into account a single requirements, how has it considered in the design?"

		DESING ELEMENTS			
		Content 1	Content 2	Access path 1	...
REQUIREMENTS-RELATED INFORMATION	VISIONS				
	Vision 1				
	Vision 2				
	...				
	GOALS				
	Goal 1				
	Goal 2				
	...				
	REQUIREMENTS				
	Requirement 1				
	Requirement 2				
	....				

DMM – Design Motivations Matrix

- Lists vertically the design elements of the application and horizontally their kind of motivations
- It is not just the opposite of RIM
- "Negative" design elements can be also listed in this matrix
- Horizontally: "Why did you adopted this solution?", "Why did you rejected this solution?"

		DESING MOTIVATIONS						
		Visions	Goals/Requirements	Designer expertise	Understanding of the domain	Contraints	Law obbligations	Arbitrary choices
DESIGN ELEMENTS	Content 1							
	Content 2							
	Access path 1							
	...							
	Negative design element 1							
	Negative design element 2							
	...							

Slides pack 3

- Detailing activity 1: preliminary plan (G3_M2);
- Detailing activity 2: information re-organisation and normalisation (G4_M2);
- Detailing activity 3: elicitation and analysis (G5_M2);
- **Detailing activity 4: specification and validation (G6_M2);**
- Introducing examples where TRAMA has been applied (G7_M1);
- Wrap-up the main concepts (G8_M1).

Specification

- After the analysis has been properly conducted, the traceability expert have to present all the results in a structured document
- This document is a real traceability specification, reporting
  - which the stakeholders and the goals of the tracing activity were
  - which kind of activities have been actually performed
  - which information have been surfaced
  - which the consequences of these results may be
- The role of a traceability documentation is resuming the elements surfaced during the analysis and organising it in a structured way
- This kind of specification is able to summarise all the project components and the relationships between them, allowing a compact but complete understanding of the project status

Typical specification structure (1)

1. Executive summary.
2. Project summary
   - highlighting its goals, people involved, current status, etc.
3. Traceability preliminary plan
   - summarising the goals, the stakeholders and the expected results of this activity
4. Information re-organisation and normalisation section
   - presenting how the project knowledge have been structure to allow a proper tracing activity
   - this section provides a general view of the current project status, a compact requirements specification and a compact design schema

Typical specification structure (2)

5. RIM matrix
   – with comments, highlighting the relationships between requirements and design
6. DMM matrix
   – with comments, highlighting the relationships between design and its motivations
7. Summary of the results achieved by the analysis
8. Benefits
   – section highlighting the benefits that traceability brings or will bring to the project.
9. Conclusion
   – reporting the reasons why the tracing activity has been performed and the main consequences for the project, in terms of design areas to review, features to better implements, requirements to re-consider, etc.

49

Validation (1)

- Traceability validation is the activity to check the analysis results with requirements analysts, designers, project managers and clients
- The specification is written for these people, so it must be written in a language which they can understand
- Furthermore, the results should be written so that they may be verified
- Validation works with a final draft of the traceability document, i.e. with negotiated and agreed information, after each meeting with project managers and designers
- The validation phase is therefore a “transversal” activity that should be run and re-run continuously during elicitation and analysis, as well as after the specification has been written

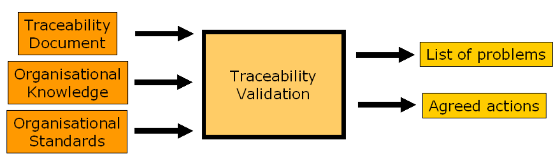
50

Validation (2)

- Validation certifies that the traceability document is an acceptable description of the overall project, in terms of:
  – Completeness and consistency of all the information reported.
  – Conformance to standards adopted in the project and in the company (reports structure, responsibilities, etc.).
  – Conflicts between traceability stakeholders’ goals, e.g. between designers (“all our choices were strongly motivated”) and clients (“some elements could be improved”).
  – Technical errors in the description of how the design is of what the requirements are, from a designers and requirements analysts point of view.
  – Ambiguous information, expressed not clearly or using terms, schemas or other elements that in that particular project or in the company have other meaning

51

The validation process (1)



52

The validation process (2)

- Inputs
  – the traceability document
    - should be a complete version of the document, not an unfinished draft, formatted and organised according to organisational standards
  – organisational knowledge
    - knowledge, often implicit, of the organisation which may be used to judge the realism of the results
  – organisational standards
    - local standards e.g. for the organisation of the specification documents

53

The validation process (3)

- Outputs:
  – list of problems
    - a list of discovered problems in the traceability document
  – agreed actions
    - a list of agreed actions (that can be several or none) in response to problems discovered

54

Slides pack 3

- Detailing activity 1: preliminary plan (G3_M2);
- Detailing activity 2: information re-organisation and normalisation (G4_M2);
- Detailing activity 3: elicitation and analysis (G5_M2);
- Detailing activity 4: specification and validation (G6_M2);
- **Introducing examples where TRAMA has been applied (G7_M1);**
- Wrap-up the main concepts (G8_M1).

Slides pack 3 55

Example 1: Munch in Berlin

- Web site for the "Munch un Berlin exhibition" at the *Staatliche Museen zu Berlin* (Germany)
- It represents the first practical result of the WED approach based upon a linguistic approach considering the interaction of a user with a web site as a dialogue
- The web site is optimized for visually impaired people, where the interaction is more natural, like in an oral dialogue

Slides pack 3 56

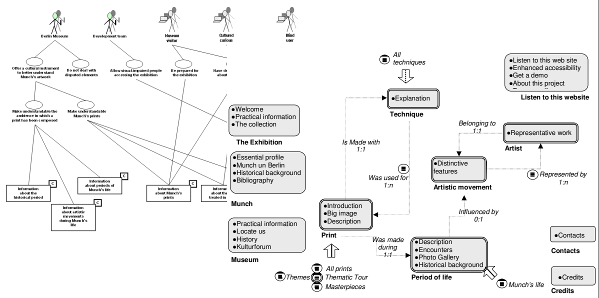
Munch: Preliminary Plan

- Before to put the application on-line, a consistency check have been requested to "adjust" the last elements and to fix an up-to-date documentation of the overall project.
- TEC-Lab performed a first traceability analysis focusing on the conciseness and on the understandability of the documentation to provide.
- A further traceability phase has been conducted in February 2005 to cope with new and refined project goals
  - design a website which might work also as a fixed information kiosk in the museum
  - make the website more usable by visually-impaired users (refining the WED approach)
  - promote knowledge and awareness about a temporary exhibition being hosted at the Museum (Munch's prints and drawings).
- Traceability was here performed to evaluate the impact of changing requirements and of proposed new solutions on the application.

Slides pack 3 57

Munch: information normalisation

- With AWARE and IDM



Slides pack 3 58

Munch: elicitation and analysis (RIM)

DESIGN ELEMENTS	CONTENTS	Prints	Techniques	Periods of life	Artistic movements	Artists	Munch	The museum	The exhibition	Contacts	Credits	Listen to this Website	RELATIONSHIPS BETWEEN CONTENTS	From a print to the technique it was made with	From a technique to the prints made with that technique	From a print to the period of life during which the print was made	From a period of life to the artistics movements actives in Germany during that period	From an artistic movement to the artists that represent the movement	From an artist to the artistic movement he belonged to	ACCESS PATHS TO CONTENTS	A list of all the prints	A group of masterpieces	Some thematic tour of prints grouped by theme	A list of Munch's periods of life	A list of all the techniques
VISIONS																									
Frame works of art within historical background				X	X	X										X	X	X						X	
MOTIVATIONS																									
Be prepared for visiting the exhibition		X	X				X	X	X	X				X	X						X	X	X		X
Study Munch and his art		X	X	X	X	X	X							X	X	X	X	X	X		X	X	X	X	X
Appreciate the artworks in the exhibition		X	X									X													
GOALS																									
Design site & kiosk																									
Make site accessible		X										X													
Raise awareness on Munch's prints		X						X	X													X	X		
Provide leading themes				X	X	X										X	X	X						X	
Raise awareness on art. movement					X												X	X							
Raise awareness on polit. background				X	X	X										X	X	X						X	
Understanding what's interesting in the exhibition																						X	X		
Finding information about the paintings in the exhibition		X	X											X	X										
Finding information about Munch's life				X			X																	X	
Finding historical information about Munch				X	X	X	X									X	X	X	X					X	
Finding detailed information about Munch's work and art		X	X											X	X						X	X	X		X
Understanding the relevance of Munch in the history of art					X	X											X	X	X						
Finding information about the techniques used in the paintings			X											X											X
Efficiently accessing the exhibition's topics		X																				X	X		
Understanding the site structure and the browsing capabilities												X													

Slides pack 3 59

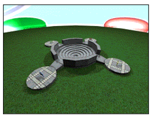
Munch: elicitation and analysis (DMM)

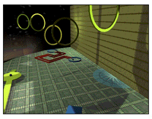
CHOICES SOURCE	Visions	Motivations	Goals	Specific understanding of the domain	Designer expertise	Technology constraints	"Graphic" constraints	Budget constraints	Laws obligations
CONTENTS									
Prints	X	X	X		X				
Techniques		X	X						
Periods of life		X	X						
Artistic movements	X		X						
Artists	X		X						
Munch	X		X						
The museum		X							
The exhibition		X	X						
Contacts					X				
Credits					X				
Listen to this Website			X						
RELATIONSHIPS BETWEEN CONTENTS									
From a print to the technique it was made with					X				
From a technique to the prints made with that technique					X				
From a print to the period of life during which the print was made					X				
From a period of life to the artistic movements actives in Germany during that period	X								
From an artistic movement to the artists that represent the movement					X				
From an artist to the artistic movement he belonged to					X				
ACCESS PATHS TO CONTENTS									
A list of all the prints					X				
A group of masterpieces		X	X						
Some thematic tour of prints grouped by theme			X	X					
A list of Munch's periods of life					X				
A list of all the techniques					X				

Slides pack 3 60

Example 2: Learning @ Europe

- Educational project aiming at fostering the development of a “European Identity”
- Educational approach novel in several respects
  - advanced content
  - technology-enhanced e-Learning
  - multicultural experience
  - engaging “games”
  - cultural competition





Slides pack 3 61

L@E: preliminary plan (1)

- In a first full experimentation year, between 2004 and 2005, 48 classes from 6 European countries (Belgium, France, Italy, Norway, Poland and Spain), nearly 60 teachers and 1,000 students were involved
- A new advanced experimentation year, between 2005 and 2006, will bring the project at an industrial stage. Before this new experimentation, a complete revision of the whole setting of the experience will be performed.
- A traceability analysis has been requested to facilitate this revision activity

Slides pack 3 62

L@E: preliminary plan (2)

- Goals
  - reorganize the complex and various material describing and designing the experience
  - pave the grounds for a reengineering activity
  - internal communication, to communicate the project status to all the team members
  - reverse requirements engineering, re-organizing and refining requirements and surfacing missing information, fundamentals to understand the project but never explicitly documented
  - design tuning, surfacing missing design components and re-aligning the design with the project state-of-the-art
  - design revision, to facilitate the project revision before a new experimentation period

Slides pack 3 63

L@E: information “normalisation” (Goals)

- General goals
  - Offering to schools a collaborative learning experience based on new technologies
  - Basing the experience on historical contents
  - Basing the experience on a multicultural approach
  - Allowing the educational impact to be measurable
  - Allowing to participate classes and pupils of every level and kind, not only the best classes in the best schools
  - Minimizing the internal management costs of the experience
- Educational goals
  - Knowledge (i.e. teaching a “know what” to students)
    - About local (national) history
    - About other countries’ history
    - About general historical concepts and processes
  - Skills (i.e. teaching a “know how” to students)
    - Use of “professional” English (as a tool to work)
    - Use of technological tools for synchronous or asynchronous collaboration (3D worlds, forums, online communities, etc.)
    - Group work (face to face collaboration)
    - Collaborative work (remote collaboration)
  - B3 Attitudes (i.e. provoke an habit change to students)
    - Sense of curiosity for history
    - National identities are the result of a process: multiple cultures / multiple identities
    - Improved attitude towards history
    - Critical thinking towards knowledge: truth appears through a variety of opinions
    - Different attitude towards knowledge, different learning modality (e-learning)

Slides pack 3 64

L@E: information “normalisation” (Visions)

- Integration in schools’ curricula
  - Convenient quantity of commitment
    - For students. The project aims at support schools as they are and not to subvert the internal organisation; the experience have to involve an entire class (12 to 25 students) and have to be guided by teachers with active and directive roles. The project may help teachers in managing different class segments.
    - For teachers. The project aims do not include that teachers learn something about technology. The experience does not base on teachers’ technological skills.
  - Convenient use of infrastructural resources. The project must not requests to school an excessive use of laboratories or a too sophisticated technological equipment.
  - The educational benefits have to be related with the general educational goals of schools and of their curricula. Teachers must be able to justify the time and organisational effort spent for participating in this experience.
- Characteristics of and educational competition
  - It has to be a motivation for students in learning; it has to be a “true” competition and repay the commitment. Therefore the competition should be:
    - Open: motivation should remain active for everyone until the end, also for micro-sessions
    - Serious: it should repay different skills and valorise a deep understanding but it should not be frustrating
  - It has not to be frustrating: participants should not be demotivated by difference of results with the others. This characteristic have to be balanced with the previous one.
  - Engaging but not an end in itself; e.g. the access to cultural questions (the “serious” part) could be win with games involving “physical” or technical skills (the engaging part).

Slides pack 3 65

L@E: information “normalisation” (Requirements)

- The experience have to include the use of collaborative 3D worlds
- The experience have to include the use of tools for asynchronous collaboration
- The experience have to include the teachers’ active role
- The educational activities have to involve the whole class
- The activities have to be modularized in order to facilitate class segmentation
- The activities must require to students a minimum background knowledge
- The activities must not presuppose that teachers know how to use technologies
- The applications must allow to participate with a low technology level and include a degraded mode of use for low connections
- The historical contents have to highlight multiple opinions, disciplines, localizations and cultures involved in the topic
- The experience have to support the creation of a virtual communities of students, also after the end of the project
- The experience have to support the creation of a virtual communities of teachers, also after the end of the project

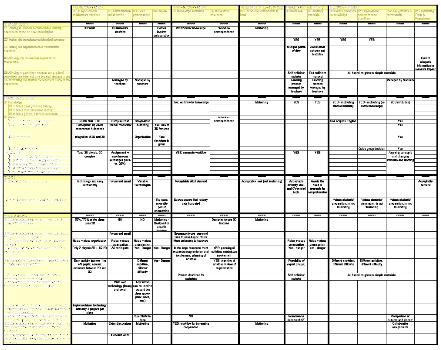
Slides pack 3 66

L@E: information "normalisation" (Design)

- Static components
  - 3D synchronous collaborative sessions
  - Asynchronous collaboration (forum/email)
  - Class presentations
  - Games
- Dynamic components
  - In-the-large sequence: succession of sessions, asynchronous sessions and off-line activities during the experience
  - In-the-small sequence : succession of the activities, contents and tests in a session
- Transversal components
  - Educational competition in itself
- Educational materials
  - Interviews (extended and simplified)
  - Auxiliary materials
- Testing materials
  - Quick questions on knowledge, "matter of fact" about local history, about other countries' history and about general historical concepts
  - Open-ended comprehension questions about local history, about other countries' history and about general historical concepts
  - Assignments & home-works (to apply the knowledge)
  - Monitoring Tools & Procedures

L@E: Meetings

L@E: elicitation and analysis (RIM)



L@E: elicitation and analysis (RIM)

	STATIC COMPONENTS				DYNAMIC COMPONENTS
	D1 3D synchronous collaborative sessions	D2 Asynchronous collaboration	D3 Class presentations	D4 Games	D5 In-the-large sequence
EDUCATIONAL GOALS					
B1 Knowledge					Yes: workflow for knowledge
B1.1 About local (national) history					
B1.2 About other countries' history					
B1.3 About general historical concepts					
B2 Skills					
B2.1 Use of "professional" English	Quick chat + 2D	Complex chat	Composition		
B2.2 Use of technological tools	Perception: all; direct experience: it depends	Internal moderator	Authoring	Yes: use of 3D features	
B2.3 Group work	Integration of 3D and 2D		Organization	Fast decisions in group	
B2.4 Collaborative work					
B3 Attitudes	Test: 3D simple, 2D complex	Assignment + spontaneous exchanges (80% vs. 20%)			YES: adequate workflow
B3.1 Sense of curiosity for history					
B3.2 Multiple cultures / multiple identities					
B3.3 Improved attitude towards history					
B3.4 Critical thinking towards knowledge					
B3.5 Different attitude towards knowledge					
VISIONS					
V1 Integration in schools' curricula	Technology and easy connectivity	Forum and email	Variable technologies		Acceptable effort demand
V2 Characteristics of and educational competition				The most enjoyable part of competition	Scores ensure that nobody gets frustrated
REQUIREMENTS					
R01 The experience have to include the use of collaborative 3D worlds	50% / 70% of the class uses 3D	NO	NO	Motivating - Designed to use 3D features	
R02 The experience have to include the use of tools for asynchronous collaboration		Forum and email			Sequence forces use (and time to use) Async. Tools

L@E: elicitation and analysis (DMM)

	Visions	Requirements	Designer expertise	Specific understanding of the domain	Constraints	Law obbligations
STATIC COMPONENTS	###############	###############	###############	###############	###############	###############
D1 3D synchronous collaborative sessions		Offering to schools a collaborative learning experience based on new technologies		In past projects this technology has been a good motivation for e-learning	Use of the in-house 3D technology	
D2 Asynchronous collaboration		Create a community	Forums and emails are common tools to build up an asynchronous collaboration			
D3 Class presentations			Just to add an activity in the first (introductory) session			
D4 Games				Another good idea from past projects as a motivation for traditional learning		
DYNAMIC COMPONENTS	###############	###############	###############	###############	###############	###############
D5 In-the-large sequence	Acceptable effort demand	Workflow adequate to educational goals				
D6 In-the-small sequence			Workflow correspondance. Each session has not to be too "boring" or long or simple/difficult, etc.			
TRANSVERSAL COMPONENTS	###############	###############	###############	###############	###############	###############
D7 Educational competition in itself	It has to be a motivation for students in learning: open and serious			In past projects it has been a good motivation for e-learning		
EDUCATIONAL MATERIALS	###############	###############	###############	###############	###############	###############
D8 Interviews			This way to present information has worked very well in past projects			
D9 Auxiliary materials			This material go with interviews as background explaination (maps, biographies, etc.)			
TESTING MATERIALS	###############	###############	###############	###############	###############	###############
D10 Quick questions on knowledge	Values students' preparation, is not frustrating					
D11 Open-ended comprehension questions						
D12 Assignments & home-works		Improve knowledge, skills and attitudes				
D13 Monitoring Tools & Procedures		The educational impact has to be measurable				

Slides pack 3

- Detailing activity 1: preliminary plan (G3_M2);
- Detailing activity 2: information re-organisation and normalisation (G4_M2);
- Detailing activity 3: elicitation and analysis (G5_M2);
- Detailing activity 4: specification and validation (G6_M2);
- Introducing examples where TRAMA has been applied (G7_M1);
- **Wrap-up the main concepts (G8_M1).**

TRAMA: Traceability Analysis Method for (interactive) Applications

Slides Pack 4

Giovanni Randazzo

30 slides

Slides pack 4

- Analysing and documenting traceability: practical tips (G1_M3);

Tip 1

- Use AWARE to normalise requirements

AWARE: a definition

- AWARE: Analysis of Web Application Requirements
- A goal-oriented methodology supporting the requirements analysis and requirements documentation for web projects
- Representation and understanding of relevant website stakeholders and their goals is key element for successful design

AWARE is

- Stakeholder-centered
  - Websites are made by people for people
- Goal-oriented
  - High-level objectives come before the solutions
- Scenario-based
  - Reflection on contexts of use help requirements surface
- Project-driven
  - Goals and domain knowledge is mediated within the scope of the project
- Tool-independent
  - Flexible notation not constrained by a proprietary platform
- Web-specific
  - but extendable to other domains

AWARE: general concepts

- Stakeholder
- Goal
- Goal Refinement
- Requirement
- Scenario

Stakeholders

- Those who have a direct interest in the success of the website are called stakeholders.
- Stakeholders may include the users, the clients who finance the web site, and other people involved in the project (e.g. sponsor, developers, and representatives of the organization departments, etc.).
- Stakeholders are either individuals or placeholders for an organization’s or institution’s interests.
- They may be “typed” (e.g. the secretary) or “single” (e.g. the director of bank x)

7

Goals

- A stakeholder may own one or more goals with respect to the website-to-be.
- A goal is defined as a high-level target of achievement for a stakeholder.
- It may represent a wished state of affairs for the main stakeholders (“Increase customer loyalty”), but also a wished experience or an expectation for a class of users (“Find suitable funds”).
- Goals vary in abstraction level and granularity.

8

Goal refinement

- Goals are analysed by decomposing them into subgoals, according to an ad-hoc refinement process
- The refinement process consists in:
  - Detailing the goals
  - Deciding which and how upper goals may be satisfied - according to the constraints, the obstacles met and resource available – and highlight possible alternatives
  - Defining requirements contributing to accomplish the goals
- The refinement process is mainly top-down but highly iterative

9

Requirements

- The outcome of the goal decomposition is a set of requirements, which represent the actual input for the design activity
- A requirement is a sufficiently high-level descriptions of the a property or functionality of the website meaningful for one or more stakeholders (e.g. “provide up-to-date fund information”)
- Requirements address a variety of design dimensions (content, navigation, access, operations, etc.)

10

Scenarios

- The elicitation and refinement process may be supported by envisioning salient episodes of use of the website, called scenarios (e.g. “an enrolled student looks for information about a specific course he is not attending….”)
- Scenarios can help uncover overlooked stakeholders, surface and exemplify goals and requirements, justify, validate or invalidate decisions
- Scenarios provoke stakeholders to reflect on requirements in view of more concrete and vivid artifacts (e.g. pieces of design, prototypes, stories)

11

Tip 2

- Use IDM to normalise design

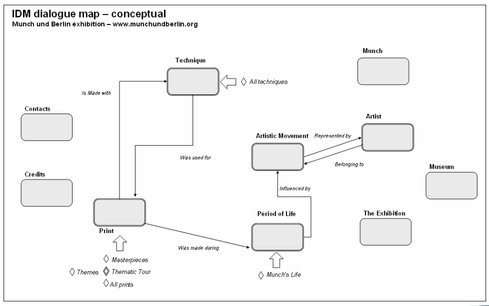
12

IDM

- Interactive Dialogue Model
- A dialogue-based design model to shape interactive applications
- Can represent both sketched ideas or fully developed solutions
- The graphic representation of these structures is very readable, compact and expressed in a conceptually simple way
- Easy to use for brainstorming
- Good as elicitation tool
- Tailored to master multichannel applications

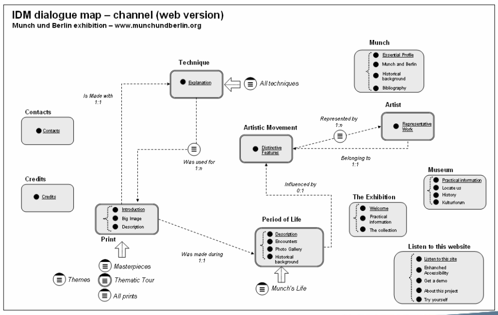
Conceptual design (C-IDM)

- A conceptual schema, of an interactive application, must convey all the necessary "dialogue strategies", without (and before) digging into details depending on technical issues



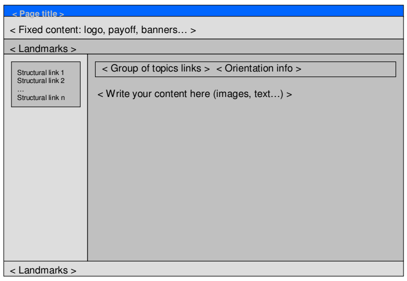
Logical Design (L-IDM)

- It can be seen as a detailed version of the conceptual design, where details are decided on the basis of a variety of channel-dependent factors



Page design (P-IDM)

- Defining the elements to be communicated to the user in a single dialogue act
- Crafting the actual pages containing the necessary elements to sustain the dialogue



Tip 3

- Meetings set-up

Meetings set-up

- Place
  - A large meetings room with a table and some chairs
  - The room should have some free walls in order to hang up the papers with the matrices
- Tools
  - Coloured pencils
  - Blackboard/flipcharts/Papers
  - Self-stick wall pads
- Roles
  - the discussants, e.g. in focus groups the project work-team that animate the meeting
  - a facilitator, i.e. a traceability expert in charge to address the discussion in a right direction, provoking answers, asking critical questions, etc.
  - a wall writer, drawing the matrices on the wall papers and filling the crosses with the traceability information raised out from discussion
  - a secretary, recording and writing notes (on a PC) about the meeting
  - a chair officer, e.g. the project manager coordinating the overall meeting

Tip 4

- Communication Management in Elicitation Meetings

19

Some biases in elicitation

- Cognitive biases
- Overconfidence
- Faulty reasoning
- Communication problems
- Motivational biases

20

Cognitive biases

- **Easy of recall**: events that are vivid and emotional or happened recently are easier to recall by the stakeholders, but they are not actually likely to occur.

- Stak. “it is very important that the user might be able to find that information”
- User “I really liked the home page of that site”

- Strategy:
  - Directed questions:
    - “how many timed does it happen in the last month?”
    - “what if the same goals is achieved by different means”?
    - “Why” questions

21

Overconfidence

- **Overconfidence**: Analysts are optimistic about their understanding of stakeholders’ goals. Requirements gathering process risk terminating too soon.

- An. “…I see what you need, that is enough for me”

- Strategy:
  - Scenario reflection: revealing knowledge being used rather than assumed
  - Direct prompting: using the ideas of another stakeholders as counter-arguments for causing reflection
  - What other kind of solution could you imagine?
  - “why questions”

22

Faulty reasoning

- **Faulty reasoning**: stakeholders might do illogical inferences in supporting their beliefs.

- “In the site, products must be organized by storing categories because our product catalogue – as you can see – is organized in this way. Also our supplier presents information by similar categories, so…”

- Strategy:
  - Devil’s advocate
  - Scenario reflection

23

Communication problems (1)

- Different Background
  - tech vs manag

- Different Domain Knowledge
  - ad extra – ad intra

- Different Language
  - system specific vs domain specific

- Different Goals
  - efficiency and easy of maintainance vs maximum functionality

24

Communication problems (2)

- Strategy: Pre-elicitation conditioning
  - Discuss the purpose of the meeting
  - What the analyst will be asking
  - What stakeholder will need to provide
  - Explain key terms
  - Explain how information will be used
  - Making stakeholders aware of potential biases

25

Motivational biases (1)

- Stakeholders are unwilling to provide accurate requirements because:
  - Organizational policy
    - Fear of being evaluated by others
    - Don't know who will know what they say
    - Fear of offending someone or break balances
    - Self-protection, self-preservation
    - Bias on domains of other stakeholders
  - Don't know what analyst needs
  - Don't know other stakeholders already met

26

Motivational biases (2)

Strategy: Pre-elicitation conditioning
  - Explain how information elicited will benefit both
  - Explain how information elicited will be used
  - State that everyone's opinion is valued
  - Tell other stakeholders already met
  - Assure responses are kept confidential

27

Tip 5

- A set of questions during analysis

28

Analysis checklist

- For each cross in a matrix the traceability expert should ask himself:
  - "Which design element fits with the needs of this stakeholder?"; "If I had to present the project to this stakeholder, which part of the design should I highlight?";
  - "Which design element fits with this goal?"; "Which is the impact of this goal into the design?";
  - "Which design element better fits with the needs of this user?"; "How can I arguing design choices to show that this user is considered in it?"
  - "Which strategy is set-up in the design to fit with this user motivation?";
  - "Which is the (positive or negative) impact of this constraint into the design?";
  - "Which are the design elements that fit with this requirement?"; "How can I show that this requirement has been properly taken into account in the design?";
  - "Why the designer chose to put this element into the design?"; "How can I show that this element is not an extra-feature in the design?";
  - "Why this element has been rejected or modified in the current design?"; "What is the impact of this choice into the project consistency with strategic goals?"

29

Tip 6

- The structure of a good traceability document

30

Typical specification structure (1)

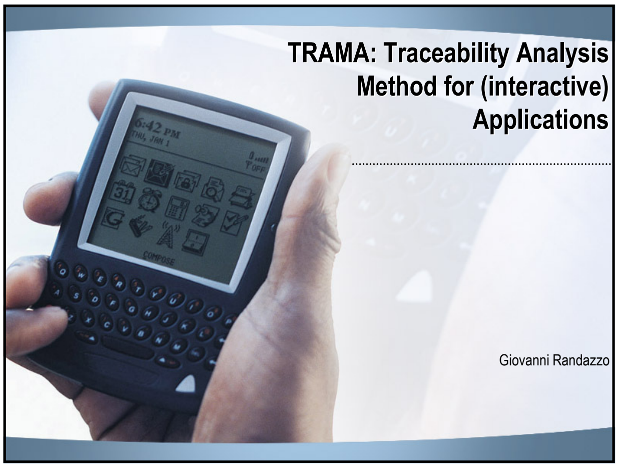
1. Executive summary.
2. Project summary
   - highlighting its goals, people involved, current status, etc.
3. Traceability preliminary plan
   - summarising the goals, the stakeholders and the expected results of this activity
4. Information re-organisation and normalisation section
   - presenting how the project knowledge have been structure to allow a proper tracing activity
   - this section provides a general view of the current project status, a compact requirements specification and a compact design schema

31

Typical specification structure (2)

5. RIM matrix
   - with comments, highlighting the relationships between requirements and design
6. DMM matrix
   - with comments, highlighting the relationships between design and its motivations
7. Summary of the results achieved by the analysis
8. Benefits
   - section highlighting the benefits that traceability brings or will bring to the project.
9. Conclusion
   - reporting the reasons why the tracing activity has been performed and the main consequences for the project, in terms of design areas to review, features to better implements, requirements to re-consider, etc.

32



ANNEX 2 TRAMA in a Nutshell

TRAMA in a nutshell

Traceability is the degree to which a relationship can be established between two or more products of the development process [IEEE, 1990]. According to Palmer [1997], RT helps ascertain how and why system development products satisfy stakeholder requirements:

- in a backward direction: backward traceability records information and data on the past history of the product, providing knowledge about the sources of a specific element (e.g. a requirement, a design element or a piece of code) and about the reasons of a specific decision in the previous project items (e.g. in goals, requirements, etc.);
- in a forward direction: forward traceability maps stakeholder needs, visions and goals to the requirements, so that the analyst can determine the impact to requirements as needs change, and assign responsibility for fulfilling a requirement to the design or to the various system components that will implement it, letting the responsible ensure that each requirement is fulfilled.

A common opinion in the Requirements Traceability field is that solution design, i.e. the design of the application solutions, may be derived directly from requirements refinement; according to current industrial practices and to some specific experiences, TRAMA proposes a different thesis: the design process is not a fully rational and explicit sequence of actions; designers keep requirements in mind as a background knowledge, and they build up the application architecture almost from scratch, as a result of a an inductive and in part intuitive activity. Since requirements are understood as base information about how the application should be and why, skilled designers are able to draw a design that satisfy in a certain measure those requirements. In common industrial cases these relationships are anyway still not explicitly specified; this problem make very hard to verify, to evaluate, to revision and to reuse efficiently design solutions in relation with high-level requirements.

TRAMA, a TRaceability Analysis Method for (interactive) Applications is a design traceability method supporting both backward and forward traceability. The TRAMA tracing activity tries to move intuition and induction to more rational cause-effect motivations, forcing to better make explicit requirements that are both implicit or unexpressed.
The method provides to designer an effective tool conceived to discuss and analyse the design choices after they have been taken, in order to refine it according to the main requirements and in order to eliminate unmotivated elements. TRAMA is based on structured matrices that cross requirements with design in a forward direction and design with its sources (requirements, motivations, constraints, etc.) in a backward direction.
As a kind of self-standing process, the TRAMA activity workflow is structured as follows:

- Preliminary plan: understanding which the stakeholders of the traceability analysis, the traceability goals, the constraints (time and budget, related to ROI1) and the expected results are.
- Information re-organisation: understanding requirements and design from documents or from interviews with designers and organise it in terms of structured specifications.
- Information "normalisation": structuring requirements and design information in "normal" terms, base on a strong methodology (e.g. AWARE for requirements and IDM for design).
- Elicitation: surfacing relationships between requirements and design in terms of impact of requirements on the design ("How did you considered this requirements in the design?") and of motivations for design choices ("Why did you adopted this solution?").

[1] Return on investment (ROI) is a straightforward financial tool that measures the economic return of a project or investment. ROI measures the effectiveness of the investment by calculating how many times the net benefits (benefits from investment minus initial and ongoing costs) recover the original investment [from: http://www.odellion.com].

- Analysis: tracing relationship and developing the Requirements Impact and the Design Motivations Matrices (RIM and DMM).
- Specification: documenting stakeholders, goals and analysis results.
- Validation: checking the results with requirements analysts, designers, project managers and clients.

Since a TRAMA analysis produces a complex picture under control, a method emphasising this global picture as a whole is needed. The managing of this "global picture" is assured by the use of matrices as conceptual tools. A second advantage of the matrix representation is that it is easy to understand and it provides a format that can be discussed by stakeholders with different backgrounds. TRAMA provides two main tools that allows to discuss, analyse, access and present the traced information:

(i) RIM, Requirements Impact Matrix
This matrix list vertically requirements-related information and horizontally all the design elements. Figure 1 shows a possible template for a RIM matrix. The horizontal dimension of the matrix provides information on how single requirements are taken into account into the design; the vertical dimension shows how a single design element satisfy the project requirements.

		DESING ELEMENTS			
		Content 1	Content 2	Access path 1	...
REQUIREMENTS-RELATED INFORMATION	VISIONS				
	Vision 1				
	Vision 2				
	...				
	GOALS				
	Goal 1				
	Goal 2				
	...				
	REQUIREMENTS				
	Requirement 1				
	Requirement 2				
	....				

Figure 1. A template for the RIM matrix

(ii) DMM, Design Motivations Model
This matrix list vertically the design elements of the application and horizontally their kind of motivations, i.e. design sources. Figure 2 shows a possible DMM matrix template. Traces between design elements and their motivations are not just the opposite of requirements-design relationships: in fact, the model highlight the justification of the design, that may be motivated by specific requirements or goals, by visions, by an understanding of the specific domain, by the expertise of the designer, by constraints or by arbitrary choices. "Negative" design elements can be also listed in this matrix; in this case, relationships rationale and comments inside crossed reports the "why not" answer, i.e. why the negative element were rejected or eliminated.

		DESING MOTIVATIONS						
		Visions	Goals/Requirements	Designer expertise	Understanding of the domain	Contraints	Law obbligations	Arbitrary choices
DESIGN ELEMENTS	Content 1							
	Content 2							
	Access path 1							
	...							
	Negative design element 1							
	Negative design element 2							
	...							

Figure 2. A template for the DMM matrix

Requirements-related information are:

- Visions, which correspond to a strategic insight of a stakeholder in the domain, and which provide a way for modelling the assumptions of a stakeholder which dictate her "weltanschaung" on the project.
- Goals, which are a wished state of affairs for the main stakeholders, but also a wished experience or an expectation for a class of users.
- Requirements, which are sufficiently high-level descriptions of properties or functionalities of the application as input for the design activity.

Design elements may be:

- Conceptual elements, i.e. the traditional conceptual design elements: content and structure of content, navigation architecture, access paths, operations, pages and layout.
- Contextual settings, e.g. the technical equipment, the place where the application is used, the physical disposition of machines in this place, etc.;
- Organisational elements, e.g. how different use sessions are organised during a week, which activities are implies in the use of the application, etc.;
- Other accessorial elements, e.g. study material needed to use the application (in a educational system), etc.

Design sources are:

- the designer expertise, i.e. particular "good design" principles that are part of the designer's skills and that she/he applies in any case;
- a specific understanding of the domain, i.e. recurrent good solutions in a domain that the designer applies because she/he learnt it by other cases in the same domain;
- a particular constraint, e.g. budget limitations, time, technology limitations, etc.;
- a law obligation, e.g. copyright issues, personal data treatment, etc.
- a requirements-related information, i.e. a vision, a goal, a requirements, etc.
- an arbitrary choice, i.e. a choice without particular reasons, usually a single detail that should anyway be set in a way or another, e.g. the structure of a game in three steps (instead of four or two).

Benefits of this method are providing designers with:

- a powerful communication mean to show to the clients that all their requirements have been considered and how, and that there are not unmotivated elements in the design;
- a structured practice to check design consistency for revision;
- an advanced tool to tune up design in maintenance phase;
- a complete project knowledge summary of requirements, of design elements and of relationships between them, as vital information allowing an effective system reengineering

REFERENCES

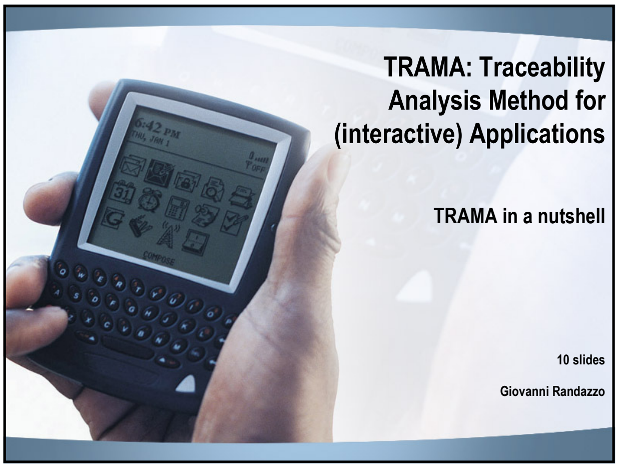
Dick, J. (2002). Rich Traceability. In: Proceedings Automated Software Engineering, 23-27 Sept. 2002

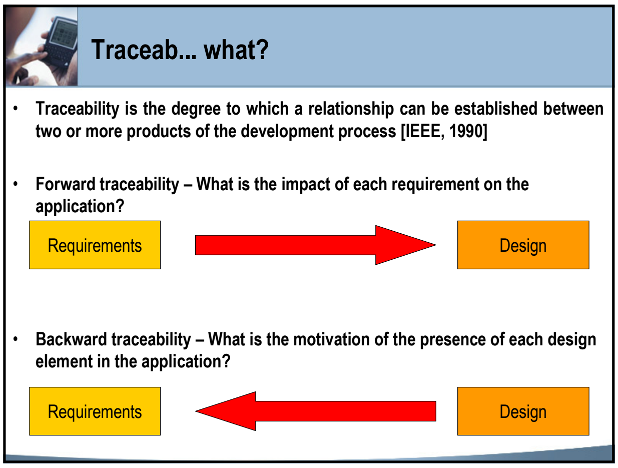
Gotel, O. & Finkelstein, A. (1994). An Analysis of the Requirements Traceability Problem. In: Proceedings of International Conference on Requirements Engineering (ICRE), IEEE CS Press, pp. 94-101.

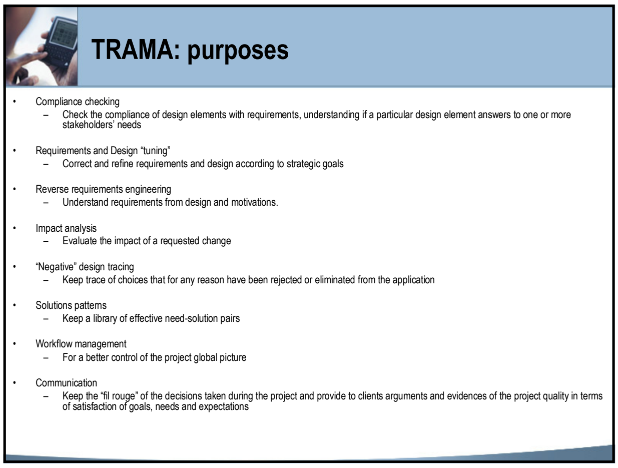
IEEE Std 830-1993. IEEE Recommended practice for Software Requirements Specifications Dec 2 1993

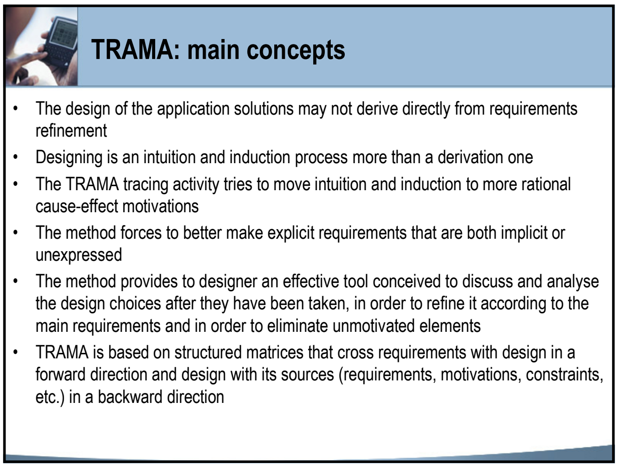
Palmer, J. (1997). Traceability. In: Thayer, R. & Dorfman, M. (Eds). Software Requirements Engineering. IEEE Computer Society Press, New York.

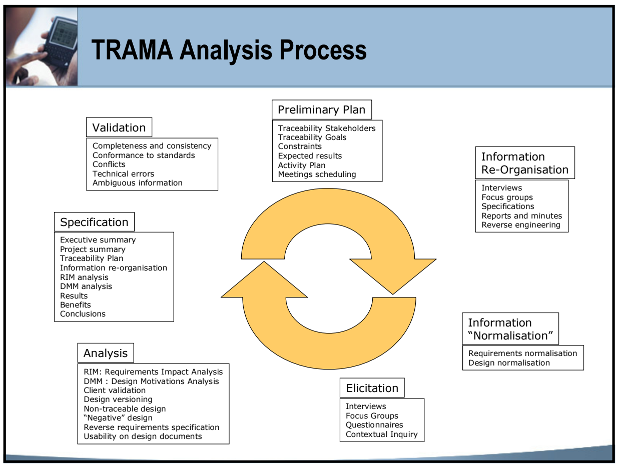
Potts, C. & Bruns, G. (1988). Recording the reasons for design decisions. 10th international conference on Software engineering. 11-15 April, Singapore.

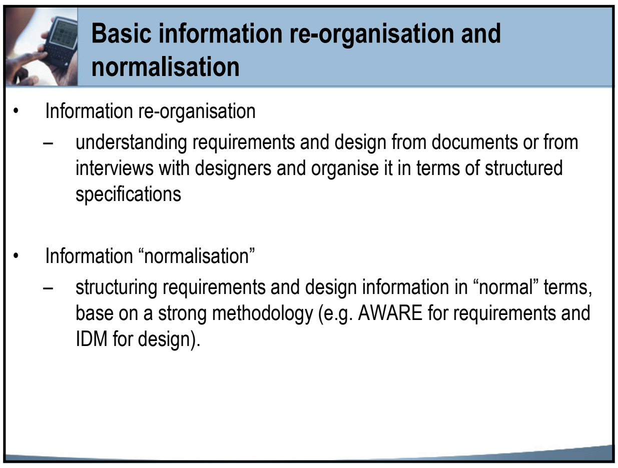

TRAMA: Traceability Analysis Method for (interactive) Applications
TRAMA in a nutshell
10 slides
Giovanni Randazzo


Traceab... what?
Traceability is the degree to which a relationship can be established between two or more products of the development process [IEEE, 1990]
Forward traceability – What is the impact of each requirement on the application?
Requirements
Design
Backward traceability – What is the motivation of the presence of each design element in the application?
Requirements
Design


TRAMA: purposes
Compliance checking
– Check the compliance of design elements with requirements, understanding if a particular design element answers to one or more stakeholders' needs
Requirements and Design "tuning"
– Correct and refine requirements and design according to strategic goals
Reverse requirements engineering
– Understand requirements from design and motivations.
Impact analysis
– Evaluate the impact of a requested change
"Negative" design tracing
– Keep trace of choices that for any reason have been rejected or eliminated from the application
Solutions patterns
– Keep a library of effective need-solution pairs
Workflow management
– For a better control of the project global picture
Communication
– Keep the "fil rouge" of the decisions taken during the project and provide to clients arguments and evidences of the project quality in terms of satisfaction of goals, needs and expectations


TRAMA: main concepts
The design of the application solutions may not derive directly from requirements refinement
Designing is an intuition and induction process more than a derivation one
The TRAMA tracing activity tries to move intuition and induction to more rational cause-effect motivations
The method forces to better make explicit requirements that are both implicit or unexpressed
The method provides to designer an effective tool conceived to discuss and analyse the design choices after they have been taken, in order to refine it according to the main requirements and in order to eliminate unmotivated elements
TRAMA is based on structured matrices that cross requirements with design in a forward direction and design with its sources (requirements, motivations, constraints, etc.) in a backward direction


TRAMA Analysis Process
Preliminary Plan
Traceability Stakeholders
Traceability Goals
Constraints
Expected results
Activity Plan
Meetings scheduling
Information Re-Organisation
Interviews
Focus groups
Specifications
Reports and minutes
Reverse engineering
Information "Normalisation"
Requirements normalisation
Design normalisation
Elicitation
Interviews
Focus Groups
Questionnaires
Contextual Inquiry
Analysis
RIM: Requirements Impact Analysis
DMM : Design Motivations Analysis
Client validation
Design versioning
Non-traceable design
"Negative" design
Reverse requirements specification
Usability on design documents
Specification
Executive summary
Project summary
Traceability Plan
Information re-organisation
RIM analysis
DMM analysis
Results
Benefits
Conclusions
Validation
Completeness and consistency
Conformance to standards
Conflicts
Technical errors
Ambiguous information


Basic information re-organisation and normalisation
Information re-organisation
– understanding requirements and design from documents or from interviews with designers and organise it in terms of structured specifications
Information "normalisation"
– structuring requirements and design information in "normal" terms, base on a strong methodology (e.g. AWARE for requirements and IDM for design).

Elicitation and analysis: RIM

- Requirements Impact Matrix
- Requirements-related information:
  - Visions
  - Goals
  - Requirements
- Design elements:
  - Conceptual (hypermedia) elements
  - Contextual settings
  - Organisational elements
- Vertically: "Taking into account a single design element, how does it fit with requirements?"
- Horizontally: "Taking into account a single requirements, how has it considered in the design?"

		DESING ELEMENTS			
		Content 1	Content 2	Access path 1	...
REQUIREMENTS-RELATED INFORMATION	VISIONS				
	Vision 1				
	Vision 2				
	...				
	GOALS				
	Goal 1				
	Goal 2				
	...				
	REQUIREMENTS				
	Requirement 1				
	Requirement 2				
	....				

Elicitation and analysis: DMM

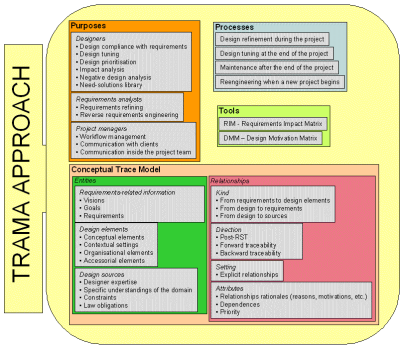
- Design Motivations Matrix
- Design elements + "Negative" design
- Horizontally: "Why did you adopted this solution?", "Why did you rejected this solution?"
- Design sources:
  - designer expertise
  - specific understanding of the domain
  - particular constraints
  - law obligations
  - requirements-related information
  - arbitrary choices

		DESING MOTIVATIONS						
		Visions	Goals/Requirements	Designer expertise	Understanding of the domain	Contraints	Law obbligations	Arbitrary choices
DESIGN ELEMENTS	Content 1							
	Content 2							
	Access path 1							
	...							
	Negative design element 1							
	Negative design element 2							
	...							

TRAMA: pros and cons

- Benefits
  - a powerful communication mean to show to the clients that all their requirements have been considered and how, and that there are not unmotivated elements in the design;
  - a structured practice to check design consistency for revision;
  - an advanced tool to tune up design in maintenance phase;
  - a complete project knowledge summary of requirements, of design elements and of relationships between them, as vital information allowing an effective system reengineering
- Limits
  - Maintenance problems
  - Solution: the requirement watcher

Wrap-up



TRAMA: Traceability Analysis Method for (interactive) Applications

Giovanni Randazzo

ANNEX 3
Case studies reports

Traceability Report 1

SEE: Shrine Educational Experience

Date: November, 2002
Target: Overall project team
Goals: Document the rationale of the entire project

Introduction

SEE is an innovative environment providing a unique experience to classes of all kinds of schools, of all the countries, for students aged between 13 to 19 years. The experience is about the Dead Sea Scrolls (the scrolls found in eleven caves, near the archaeological site of Qumran by the Dead Sea), the sect that probably wrote the scrolls, the Shrine of The Book (the section of the Israel Museum where the scrolls are preserved), the beliefs of the sect, Judaism, Early Christianity, the Bible (being the scrolls the oldest version of many parts of it), and all the different cultural, religious, historical, social themes that may originate from the scrolls and the sect. In addition, participating classes are induced to relate all the different themes to their culture, their environment, their religion, their social experience, etc. Content for the educational experience is provided in two ways: basic material for an initial background and interviews to leading experts (of different cultures and religions), providing state-of-the-art points of view about Qumran, the scrolls and their relevance for our contemporary cultures. Over time the body of interviews will become the cornerstone of a multicultural activity of dissemination concerning Qumran, related interpretations and its correspondence to worldwide cultures. The interviews (and related educational material, of various nature) currently focus on the Shrine of the Book, the Dead Sea Scrolls, the community of Qumran (their life's style, their beliefs, their rituals), the Bible and the relation between Qumran and Christianity. We plan to develop our "patrimony" adding new topics, year after year.

The above-described content is the background for a unique online experience. 4 classes meet together "in real time", through standard Internet Browsers, in a cooperative 3D world; participating students, under the direction of a museum guide, will discuss the issues, play games, answer questions, receive explanations, etc. The 3D shared environment is based upon an original methodology and technique (WebTalk-Cube) developed by Politecnico di Milano, that overcomes most limitations of similar environments, namely the idleness and the lack of real actions. Participating students are continuously engaged into action and "forced" to exploit technical tools to their limits, interacting and cooperating with each other. The educational benefits for the students can be therefore synthesized as follows:

A. Increased knowledge about Qumran and related issues (religion, history, anthropology, etc.).
B. Possibility of intercultural exchanges with students of different countries/cultures
C. Possibility of practicing an innovative and engaging form of interaction, using virtual
D. environments and set ups. The games students are invited to perform also have the role of
E. consolidating "team-ship", creating relationships and ties among different schools.
F. Possibilities of getting acquainted with state-of-the-art Communication Technologies, modern
G. multimedia, graphic, Web and Internet technologies.

The Shrine Experience includes 4 online cooperative sessions, distributed through a period of 6/7 weeks, and also a set of offline activities taking place in the schools in the intervals between a session and the following During the experience there is a week's forum called "ask the expert": one of the interviewed expert will be available, in a specially dedicated forum, to answer questions concerning the interviews and her/his own work. In a typical cooperative experience, four classes of students (ranging from 13 to 19 years of age) from different geographical areas (e.g. Israel, Italy, U.S., Australia) join a Museum guide in a set of 3D virtual environments. Some of them reproduce the Israel Museum's Shrine of the Book (see Figure 1), where the Scrolls are preserved, while some other environments are artificial settings, designed for favouring users' interaction. Participants meet together and with a guide in the shared online 3D environment, where, however, only 9 avatars (i.e. graphical human-like representations of users) will be visible: 8 students plus the guide. Two students per class are connected (and represented by an avatar), moving and acting in the shared space, while the

others support them, by suggesting answers, writing messages to the other players, and cheering passionately! Connected students can perform (hyper)-movements, manipulate objects, chat, fly and use other interaction devices.
In order to avoid the typical "idleness" of 3D worlds, where users end up with hanging around aimlessly, the Shrine Experience has been structured in detail, through a sort of "storyboard": very slot of time in a cooperative session is dedicated to a very precise activity. Actions allowed (or forbidden) to users in any situation are defined in advance. Students are never left "idle": the guide coordinates them, invites them to perform activities, to cooperate with each other and to interact with the virtual environment. The guide is provided with extended powers, which may be used in order to maintain discipline, to "move on" in the session and to assist avatars that encounter technical problems. At the beginning of each cooperative session (after a short "welcome"), a short lecture is given by the guide, who then invites students to move around and explore the environment. Cultural Games are the core of the experience: "Quiz", "Treasure Hunt", "Olympic games", offer students an engaging experience, at the same time requiring previous knowledge about the topic dealt with. A rich set of introductory material is offered to students and teachers, so that they may prepare before the experience and exploit at best the online-shared time. This material is composed of:

- Interviews to leading experts, about Qumran, the Scrolls and related issues.
- Editorial insets, explaining in detail some issues, events or characters mentioned by the experts in the interviews.
- Anthologies collecting all the excerpts from the Scrolls, the Bible, historical sources or other texts mentioned or quoted in the interviews.
- Auxiliary educational resources, providing background information on historical and geographical issues that may be obvious to some parts of the audience, but obscure to others (e.g. Israeli students know quite well where the Dead Sea is; for students of faraway countries, however, this might not be obvious at all).

Key concepts from the introductory material are then recalled during the online experience with the help of "boards" (pop-up browser windows). The rationale behind SEE, confirmed by initial trials in schools, is that teachers appreciate this innovative way of learning and that students are motivated to study and recall what they've learnt by the excitement of the game and of the competition. Collaboration among participants is not limited to the shared virtual space: it extends also to off-line activities; students are requested to work with their remote colleagues, researching on what they have learnt: they should try for example to relate ancient rituals to signs and traditions of their own present culture. The virtual museum environment is thus not only a space where an antique culture is discussed and reflected upon: it becomes a lively setting where people from different cultures are confronted with each other and with bi-millenary - yet still topical - issues.

Traceability goals

A traceability analysis has been requested for the SEE project after the first two years of experimentation. Stakeholders and goals of this analysis are listed in the following lines:

- The Israel Museum is interested to this analysis to verify if its goals have benne reached by the experience
- Designers of the experience want to verify its consistency and completeness of their choices and they need a tool to show it to the Israel Museum
- Software developers want to verify the consistency of their work in relation with deisng and they need a support to the impact analysis fro design modifications
- The educational institutions involved need to verify the consistency of the experience with their study plan.

Entities and relationships

In this project a full requirements documentation is missing. To find trace entities a short analysis of stakeholders, goals and requirements has been conducted. The following lines summarise the results.

Stakeholders

- SH01: The Israel Museum and in particular its "Shrine of the Book" section is the owner of the precious Dead See Scrolls. The Museum is the project sponsor, thanks to money received from the Dorothy Foundation.
- SH02: The Politecnico di Milano's rectorate is interested in the project because it mix technological research and cultural diffusion; this is part of the strategic goals of the Politecnico.
- SH03: Designers are those that manage the project and its development. They are interested in the success of the application for further future economical possibilities.
- SH04: Technical developers, from SOPHIE company, that owns the 3D technology for virtual cooperative worlds.
- SH05: The scientific committee is an expert group that aim at stimulating ideas and at facilitating the communication with the Israel Museum as authoritative cultural bridge for the application contents.
- SH06: Educational institutions involved in the project are secondary schools, high schools and technical institutes.
- SH07: Teachers from educational institution involved may stimulate ideas in focus groups and have the important role of educational guide for students.
- SH08: Students are the final users of the application, young people aged 10 to 18.
- SH09: Other museums may be interested in the project to replicate the experience.

Goals

- GL01: Promote institutionally the name of the Israel Museum
- GL02: Spread the knowledge about the Dead Sea Scrolls
- GL03: Declare an historical presence of Judaic populations in Palestine
- GL04: Promote institutionally the name of Politecnico di Milano as culture spreading institution and not only as technical school.
- GL05: Success of the project and its application to other domains (economic return)
- GL06: Experiment design techniques for 3D modules related to edutainment
- GL07: Transmit cultural contents
- GL08: Apply implementation technologies for cooperative virtual spaces

- GL09: Give visibility to the proprietary platform for 3D spaces (economic return)
- GL10: Be promoted through the involvement in an advanced project
- GL11: Promote the remote interaction
- GL12: Promote the meeting of different cultures
- GL13: Understand the potentialities of a new technology
- GL14: Transmit contents related to the course plan
- GL15: Experiment the edutainment approach
- GL16: Remote cultural interaction
- GL17: Observe and experience to replicate it in another context

Sub-goals
- SG01: The meeting of cultures should be supported by the participation of different schools.
- SG02: The application is not intended to be limited at the technological chat-like gadget.
- SG03: Stimulate curiosity and provoke the user in better understand and study the contents
- SG04: Transmit complex content and face authority problems about a culturally "hot" topic
- SG05: For the first year the application has to be tested in Italy and Israel
- SG06: From the second year the application should be tested in Europe and in America.
- SG07: Content have to be inter-cultural and inter-disciplinary
- SG08: Attract potential user towards 3D
- SG09: The interaction style of the user should simulate a real interaction with objects in a museum.
- SG10: The application should transmit the idea that the designers are kind of cultural "missioners"
- SG11: Teachers must feel to be up-to-date and be convinced that the experience is not a lose of time or a retard in their institutional program
- SG12: Support cooperation outside the session time.
- SG13: Keep the current teachers and attract new ones
- SG14: Show possible cultural links with institutional programs
- SG15: Teachers should be reassured on their central and active educational role.
- SG16: Contextualise the scrolls as archaeological finding.

Functional Requirements
- RQ01: The application is composed by a traditional web site (2D) for huge textual contents.
- RQ02: The 2D site is free access.
- RQ03: The 2D contents introduce the context and the application functionalities.
- RQ04: The application preview a game space.
- RQ05: The game space is a virtual 3D cooperative space.
- RQ06: The 3D site is not free.
- RQ07: The graphic spaces must reproduce the cultural atmosphere of the Museum.
- RQ08: The cooperative space must be used by eight users at the same time.
- RQ09: Each class must have two avatars.
- RQ10: Classes must be allow to compose two teams, each one with four avatars
- RQ11: The interaction is promoted by competitive games that or their completion request the interchange of information between different users
- RQ12: The experience is guided by an agent that provoke and guide the interaction.
- RQ13: The users receive symbolic prizes for the competition.
- RQ14: The experience is divided in three meetings.
- RQ15: Meeting 0 is a cultural contextualisation introduction
- RQ16: Meeting 1 aims at the content study in detail
- RQ17: Meeting 2 aims at the comparison between contents with personal on other's culture
- RQ18: Each session is composed by four parts.
- RQ19: Part 1 is an introduction.

- RQ20: Part 2 is a cultural game.
- RQ21: Part 3 is a quiz.
- RQ22: Part 4 is a wrap-up of the experience where home-works for the next session are assigned as well.
- RQ23: Textual contents are presented in a traditional way (on the web site or on 2D panels).
- RQ24: Part of the contents is dedicated to Qumran in itself.
- RQ25: Part of the contents has educational goals (questions and teacher's kit).
- RQ26: Textual contents are whitepapers about a single topic.
- RQ27: Textual contents are interviews with experts in the filed.
- RQ28: Interviews do not exceed the 10 pages and are completed by a few pages summary.
- RQ29: Contents must be available in Italian.
- RQ30: Contents must be available in Hebraic.
- RQ31: Contents must be available in English.
- RQ32: Contents must be inter-disciplinary.
- RQ33: Contents must be inter-cultural.

Non-functional requirements
- NF01: 3D experience must have a beautiful and attractive aspect.
- NF02: The application must be highly usable for non expert users.
- NF03: The application must allow rich interaction possibilities.
- NF04: Contents must be readable and interesting with no lack of richness.
- NF05: Contents must be understandable for users from different cultures.
- NF06: Graphic spaces have to support the educational plot and its activities.
- NF07: Graphic spaces are a visual content in itself: in the 3D world objects related with the topic treated are represented.
- NF08: The user must be allow to control its avatar in a rich and attractive way.
- NF09: Controls must be simple and intuitive.
- NF10: The application must fast react to controls, giving a real-time impression.
- NF11: Each world must be fast downloadable.
- NF12: Avatars are sketched graphical elements.
- NF13: Backgrounds are created by the use of textures.

Constraints
- CO01: The application cannot support more than 10 avatars in a single session.
- CO02: Resources are insufficient to produce ex-novo contents.
- CO03: Each world can have a minimum of 500kb and a maximum of 700kb.

Since there were no clear methodologies to design a 3D world in a structured way, the project team applied a in-house formalism to represent design objects, including sessions, activities, rendering, games and quizzes. All the elements are listed here below:

Sessions
- DO01: Museum Session. Introduction to the Shrine of the Book and to the Qumran community.
- DO02: Topic Session. Details wit specific references to the Qumran world.
- DO03: Topic in the world Session. Students present their home works and compare the Qumran habits with their own cultures.

Activities
- DO04: Arrival.
- DO05: Introduction.

- DO06: Passage to Shrine-inside.
- DO08: Introduction to the Qumran community.
- DO10: Introduction to the Treasure Hunt.
- DO11: Technical training
- DO13: Discussion A.
- DO14: Discussion B.
- DO17: Wrap up.
- DO18: Welcome.
- DO19: Topic introduction
- DO21: Object exhibition.
- DO22: Homework assignment.
- DO23: Guided exhibition 1.
- DO24: Guided exhibition 2.
- DO25: Introduction to the game: find your avatar.

Rendering
- DO07: Rendering of the corridor space.
- DO09: Rendering of the game space.
- DO15: Rendering of the quiz space.
- DO20: Rendering of the vault space.

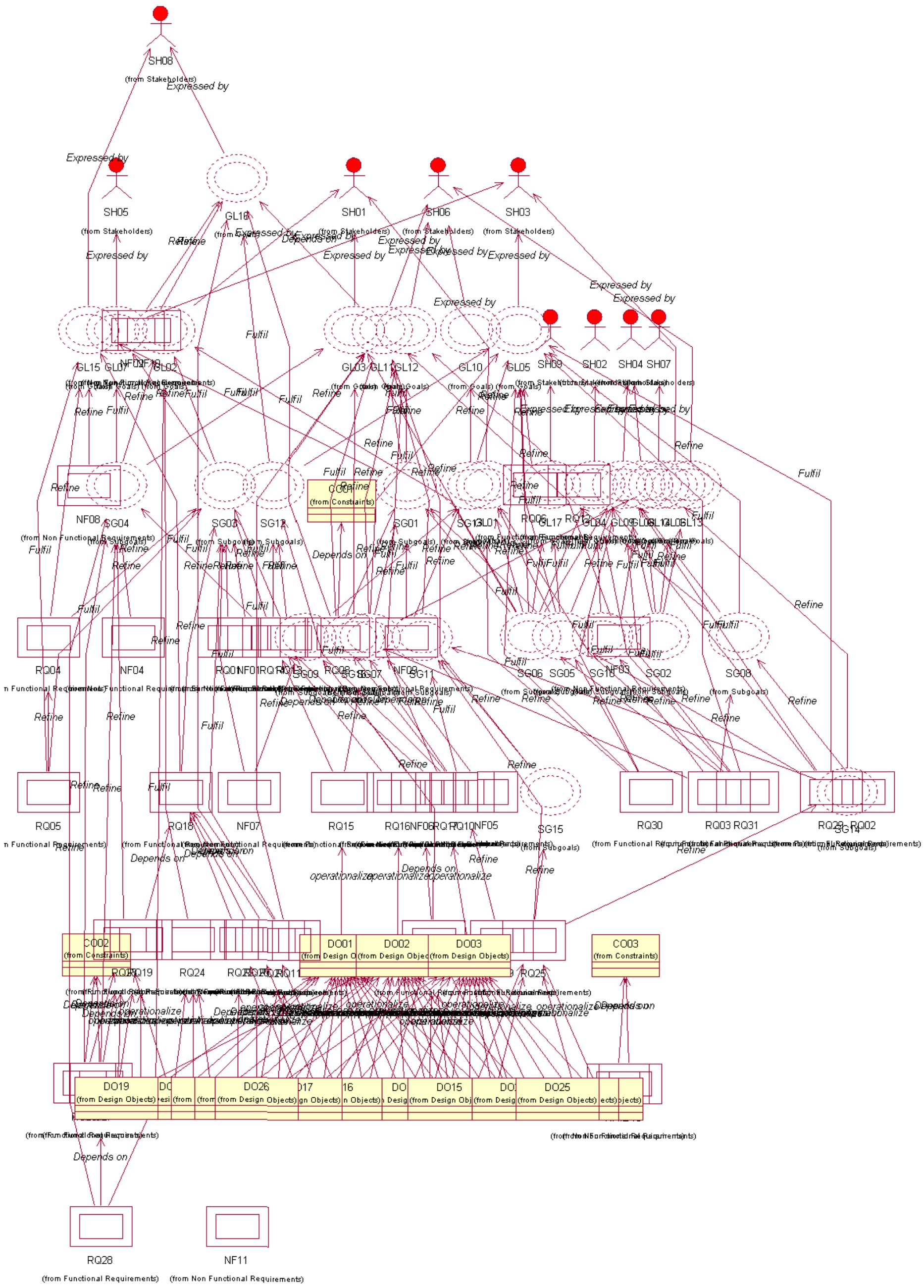
Games
- DO12: Treasure hunt.
- DO26: Find your avatar.

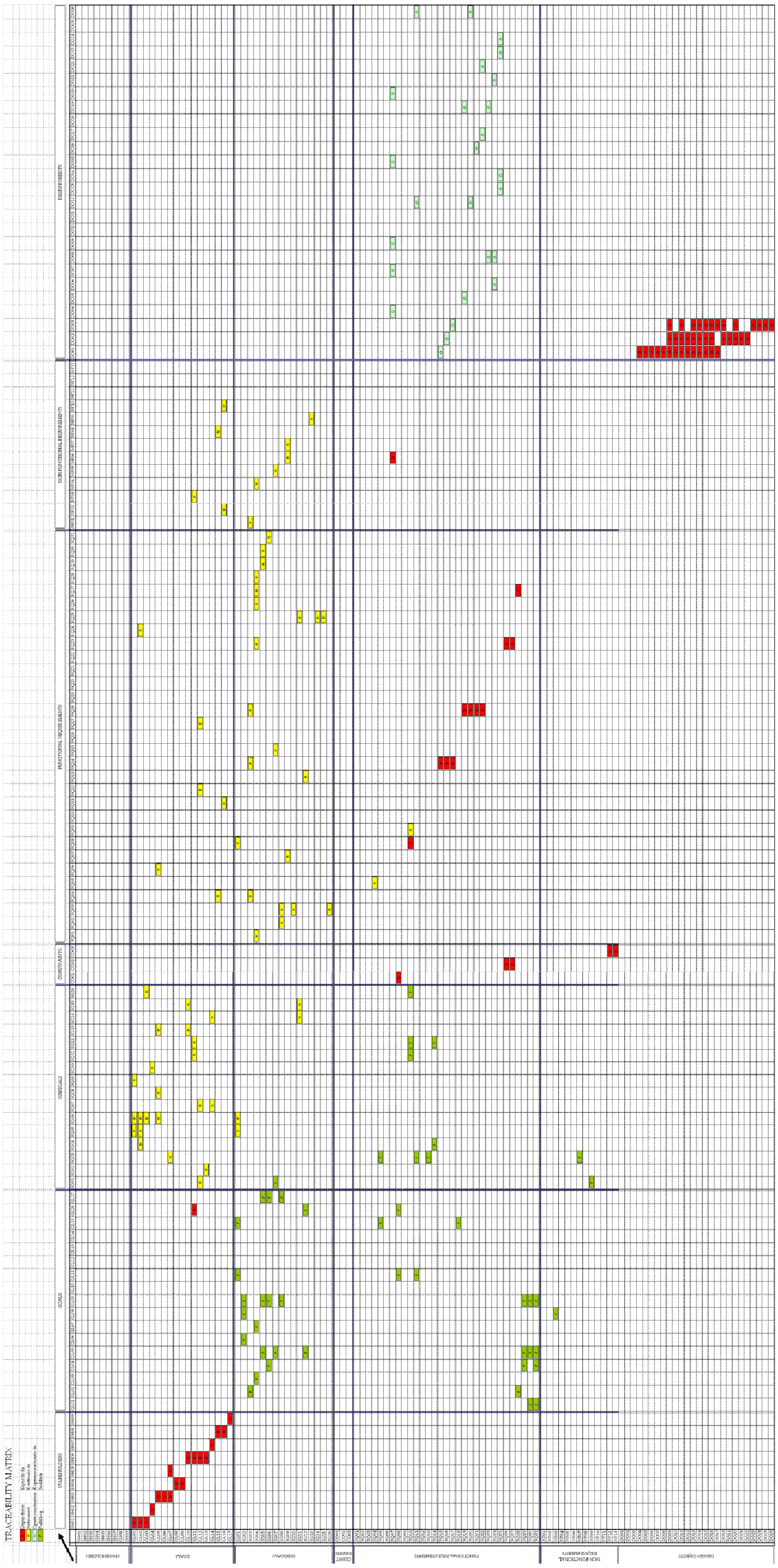
Quizzes
- DO16: Quiz.

The goal is to find relationships between all these elements, both requirements-based or design-based. Traces that have been found have the following semantics: refinement, dependency, "operationalisation", expressions and fulfilment.

Traces representations

Relationships between the entities described in the previous section have been represented using a double conceptual tool: first, a simple matrix, with all the entities listed horizontally and vertically and different colours and letters to define the semantics described before; then, a UML-like graph resuming entities and relationships in a compact picture.





Traceability Report 2
Munch un Berlin Exhibition – version 1

Date: April, 2003
Target: Project Manager
Goals: Check the design compliance with requirements

Executive summary

The main goals highlighted in the requirements phase where offering a cultural instrument to better understand Munch's artwork and allowing visually impaired people accessing the exhibition. An important element that the Museum wanted to take into consideration was not to deal with disputed elements in the scientific and artistic research about Munch.

The need to have a site accessible for blind users had no specific impact in single elements of the design, but it had an overall impact in clearly organising and structuring the conceptual map. We can identify just one particular decision coming directly from accessibility consideration, i.e. the dialogic acts in the "print" topic: the "big image" act is isolated (also from a navigational point of view) t give to blind people a descriptive introduction to the print and not frustrating it by providing from the beginning a page with just an image.

The main content structure of the site fits with the cultural need of the Museum, that wants more understanding and informed visitors. For this reason, the focus of the application is on Munch's prints exhibit in the Museum, but also in techniques used and periods of Munch's life. The site contains also some elements about artistic movements and artists active during Munch's life.

Some navigational or access possibilities have been eliminated from the site because of the need to not deal with disputed relationships between Munch and contemporary artists or movements. For this reason there are no relationship between the topic "artistic movement" and "period of life" and between "artist" and "period of life"; for the same reason, there are no direct accesses to artistic movements or to artists in the site.

“Munch und Berlin” exhibition web site

This analysis is concerned with the development of a web site for the “Munch un Berlin exhibition at the Staatliche Museen zu Berlin. The exhibition has taken place from April the 12th to july the 13th 2003 and was curated by Dr. Sigrid Achenbach. The website has been developped as part of Help project (partially funded by the European Commission). It was the result of a joint effort by Staatliche Museen zu Berlin - Preußischer Kulturbesitz (Germany), HOC-LAB of Politecnico di Milano (Italy) and TEC-LAB of University of Italian Switzerland (Lugano, Switzerland).

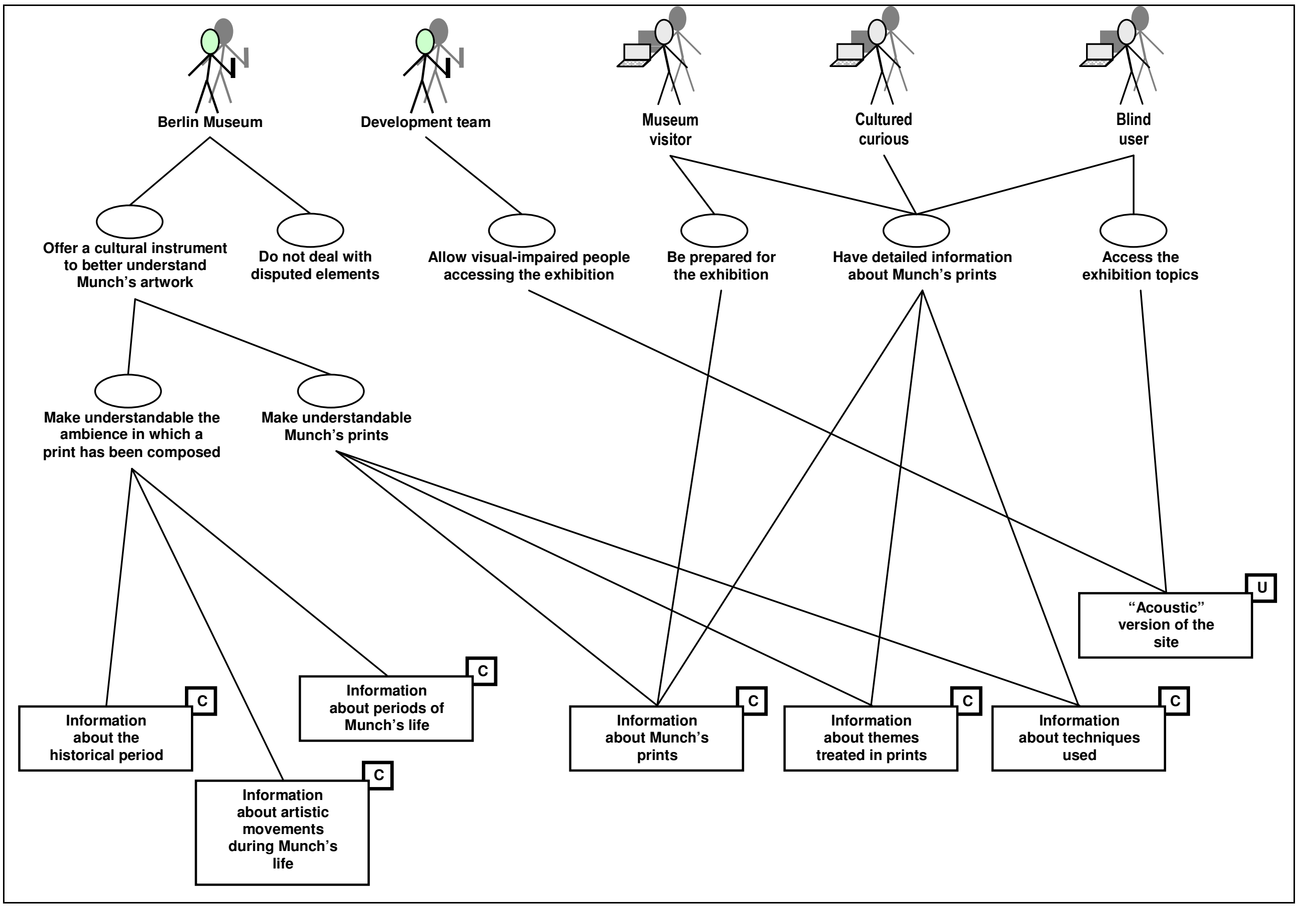
The project has developed an innovative technology that allows overcoming most of the limitations of the W3C accessibility guidelines for visually impaired users. The technology is based upon a linguistic approach to the web called WED (WEb as Dialogue), developed by TEC-Lab and HOC. The WED approach considers the interaction of a user with a web site as a dialogue. Based on this assumption, WED is an innovative design methodology which makes the designer think to the web site not just as an informative tool but as a partner (i.e. the teacher) of a didactic dialogue with its user (the pupil). HELP exploited the WED approach for designing a cultural web site optimized for visually impaired people, where the interaction is more natural, like in an oral dialogue. The success of the WED approach is showed in the challenging HELP case study: the Munch’s Exhibition web site in Berlin.

The design of the website www.munchundberlin.org represents the first practical result of the WED approach. It complies with almost all the accessibility rules of W3C; apart from being accessible, the web site presents some features that make it optimized for visually impaired users. An example is the page schema, a short summary (orally read but invisible in the page) of the basic sections of the page that the screen reader reads before reading any other content. The page schema enhances accessibility under two aspects: it gives the user the possibility to decide which section s/he’s interested in and it helps memorizing the page structure, being based on consistent templates which facilitate the user navigation and orientation.

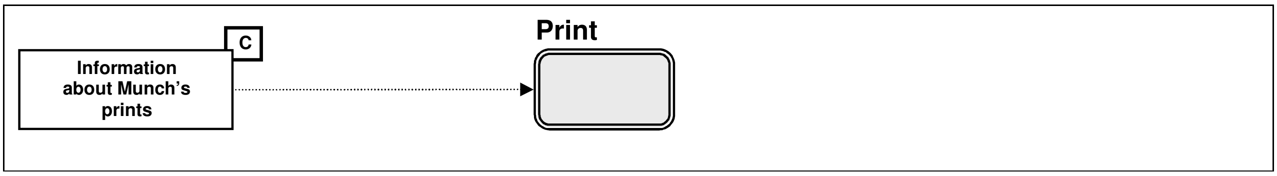
The positive feedback received by visually impaired people are encouraging. Furthermore, the methodology turned out to be useful for designing web sites for sighted people too, improving usability and user satisfaction. The results obtained are offering innovative cues and ideas for new outlooks. Future work has the overall goal of making the man-machine dialogues (such as those of a user with the web) closer to human-human dialogues (such as those of a user talking with a expert), and their effectiveness.

Requirements analysis

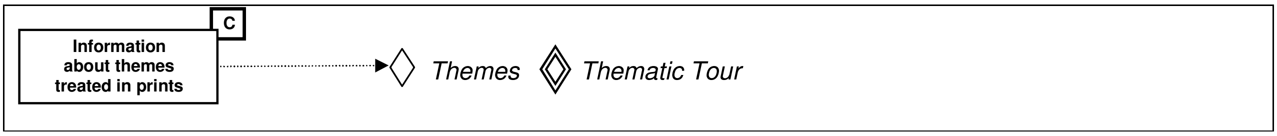
The following schema shows a synthetic view of the high-level goals, stakeholders and requirements related to the Munch exhibition web site.



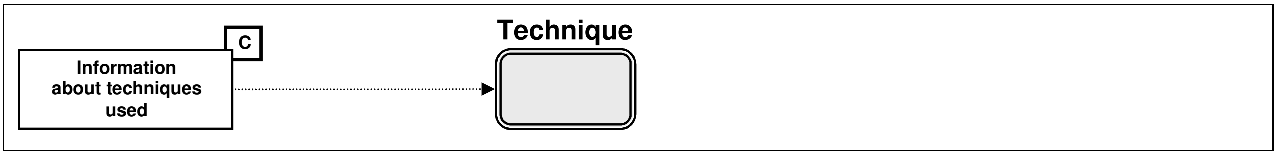
In the next pages a synthesis of how these requirements have been taken into account in the design of the web site will be presented.



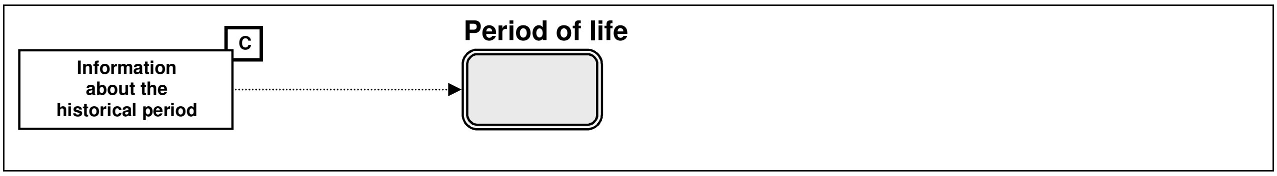
The topic "print" fits with the main need of having information about Munch's prints: it's the centre of the application (we talk about a print's exhibition).



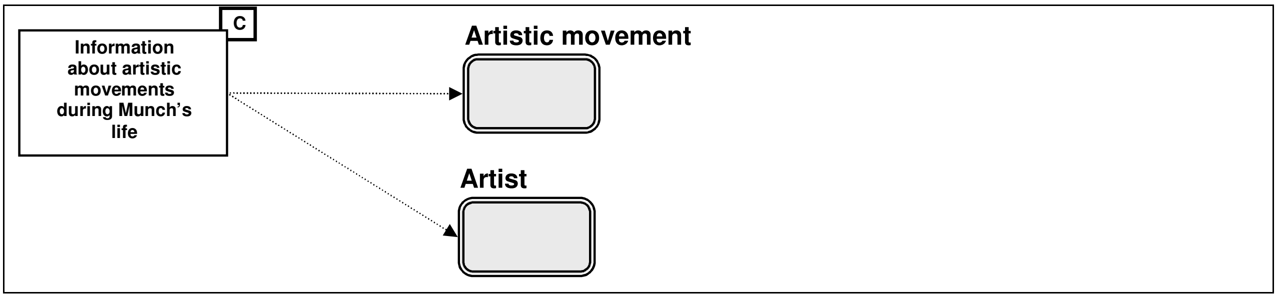
Provide information about themes treated in prints is a requirements fo the application, but contents about themes are poor. Therefore we decided to give to this content a low relevance rate, showing themes only through a collection of prints by theme.



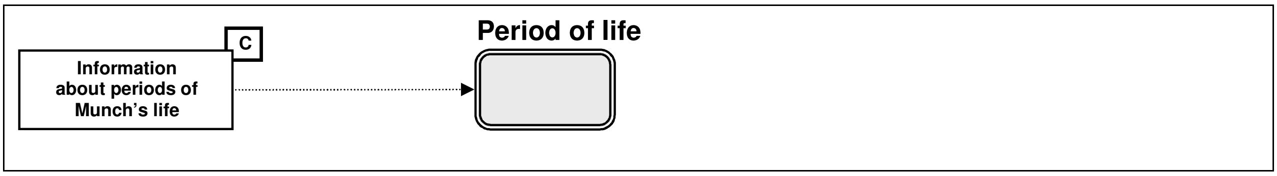
The main goal of the Museum is to offer a cultural instrument to better understand Munch's artwork; this means that the site have to give information about relevant topics of Munch's work, such as techniques used.



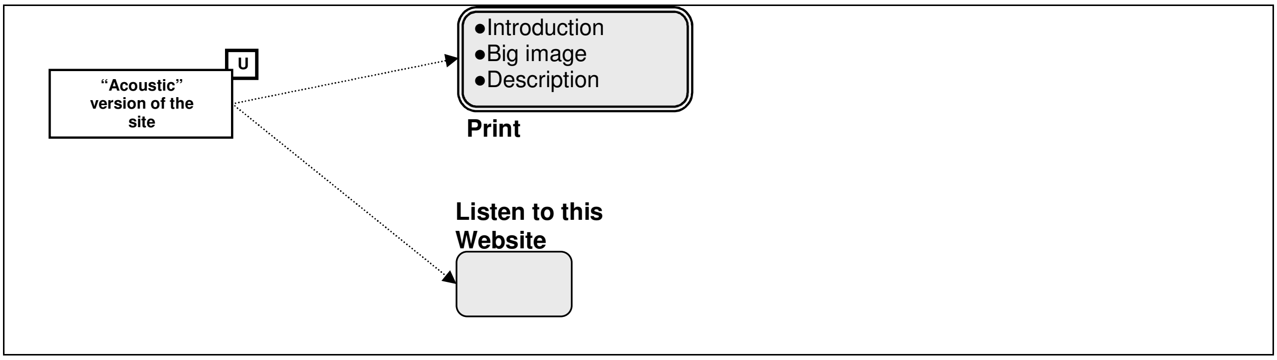
Part of the information about the historical period are in the topic "period of life" that describe a period of Munch's life. The application seems a little weak on this point.



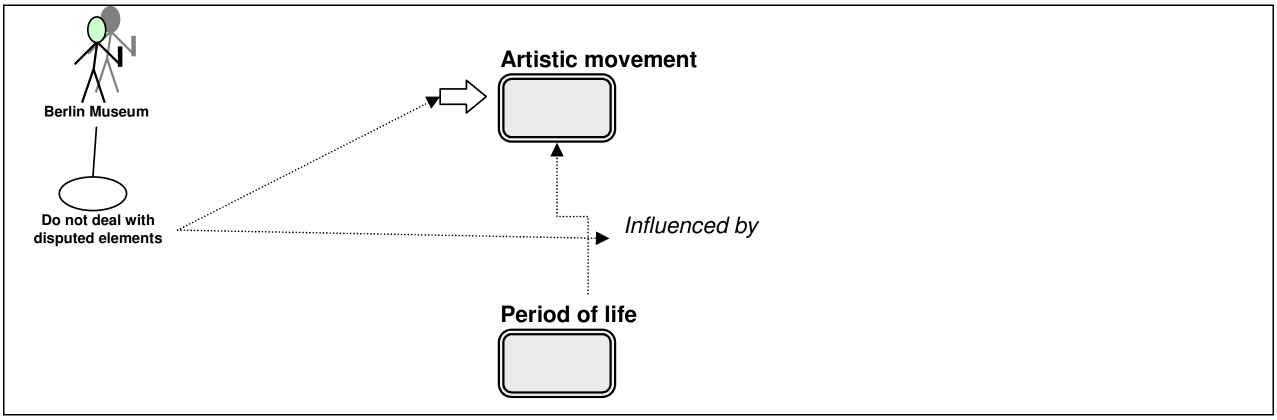
Information about artistic movements make more understandable the environment in which an artwork was born. Therefore, the design decision of providing elements about artistic movements and its representative artist goes in the direction of contextualise the period in which Munch worked.



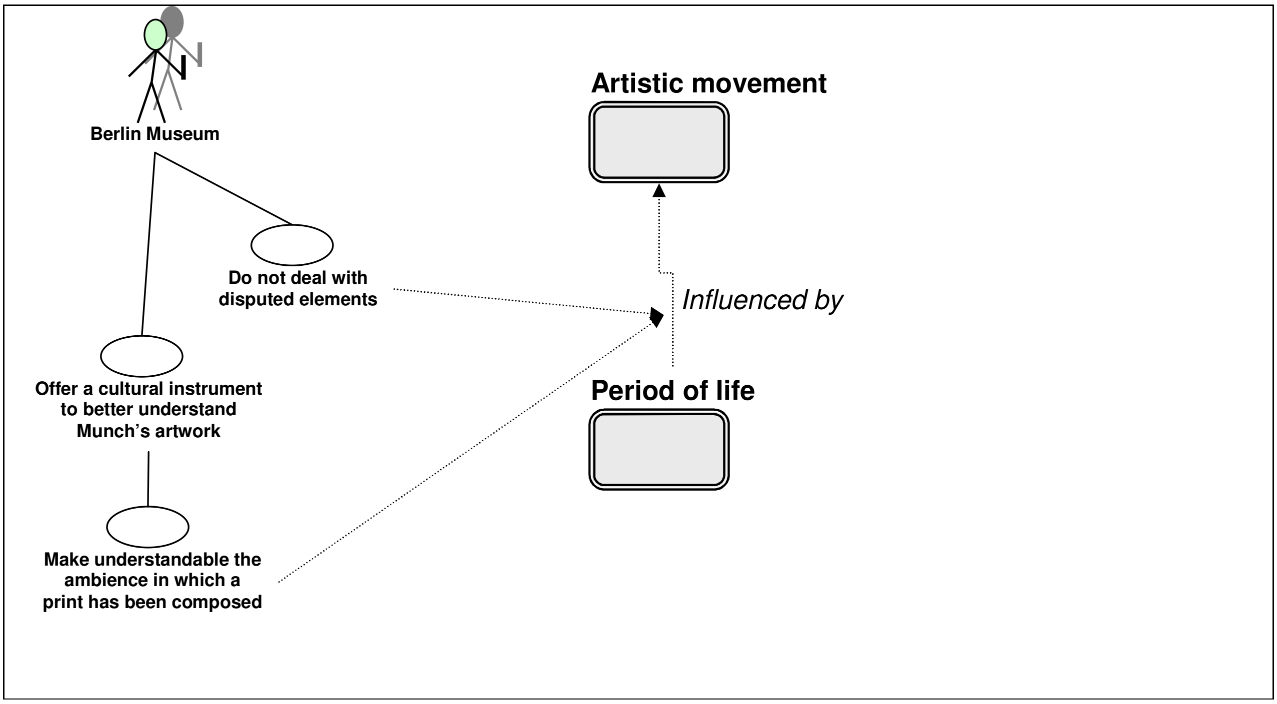
Information about periods of Munch's life make more understandable both the artworks and the social environment in which an artwork was born.



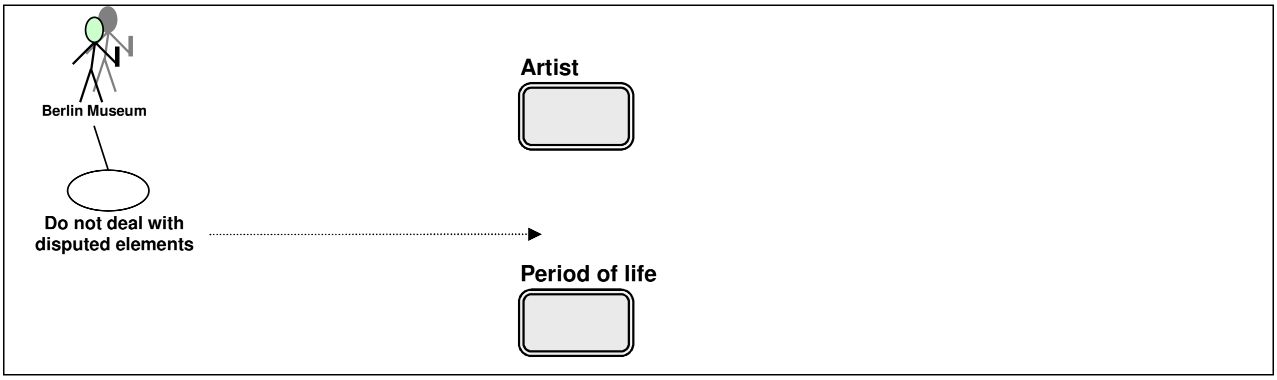
The need to produce an acoustic version of the site had a general impact on the design (that should be clearly structured) and on the implementation technique. A specific work on contents has not be done. The only "concrete" elements visible in the design are the service promotional section (Listen to this website) and the fact that the big image has been separated from the rest of the print's navigation; this last decision has been taken to provide an acoustic description of the print to blind users.



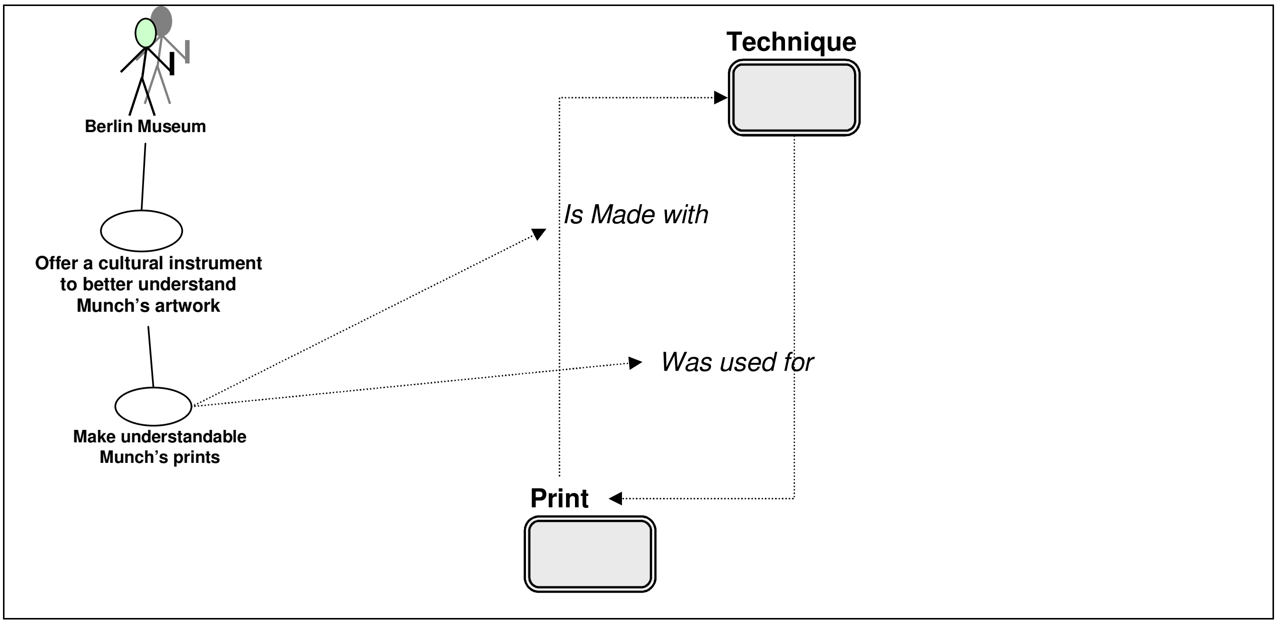
There is not a relation between artistic movements and periods of Munch's life: this has been done to do not highlight a doubtful artistic relationship. Similarly, relationships between artistic movements and Munch's artworks are disputed; therefore there are not direct accesses to that topic.



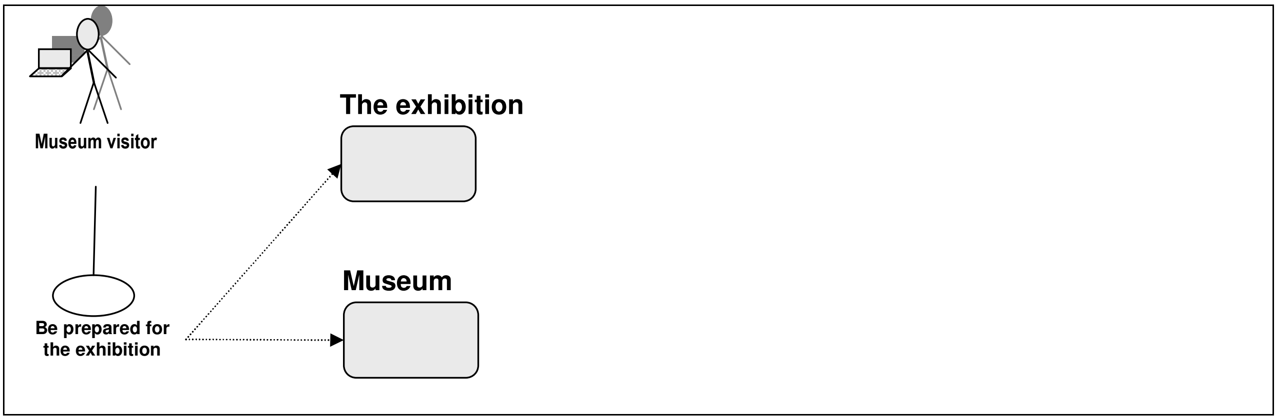
These two goals are in opposition: the relationships between periods of life an artistic movements is disputed, but the Museum wants anyway to provide deep information making more understandable the historical period in which Munch has worked. Therefore we decided to put an ambiguous semantic to this relationship; in other terms we say that in a given period of Munch's life were active some artistic movements, but there is no information about possible artistic "contaminations".



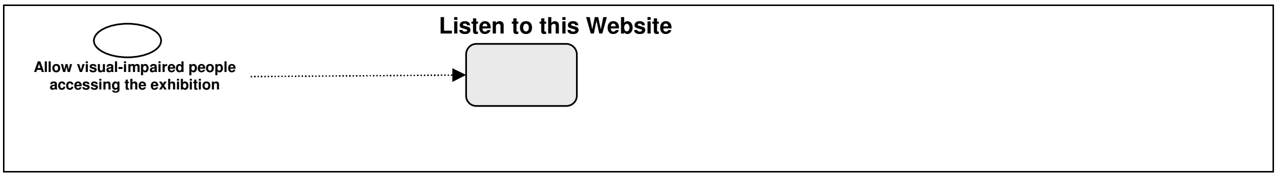
There is no relationships between periods of Munch's life and artists because of the need to do not highlight a disputed artistic influence.



These relationships help in better understanding the artwork, even if this is a reason understood "a posteriori": the true source of the decision was the designer.

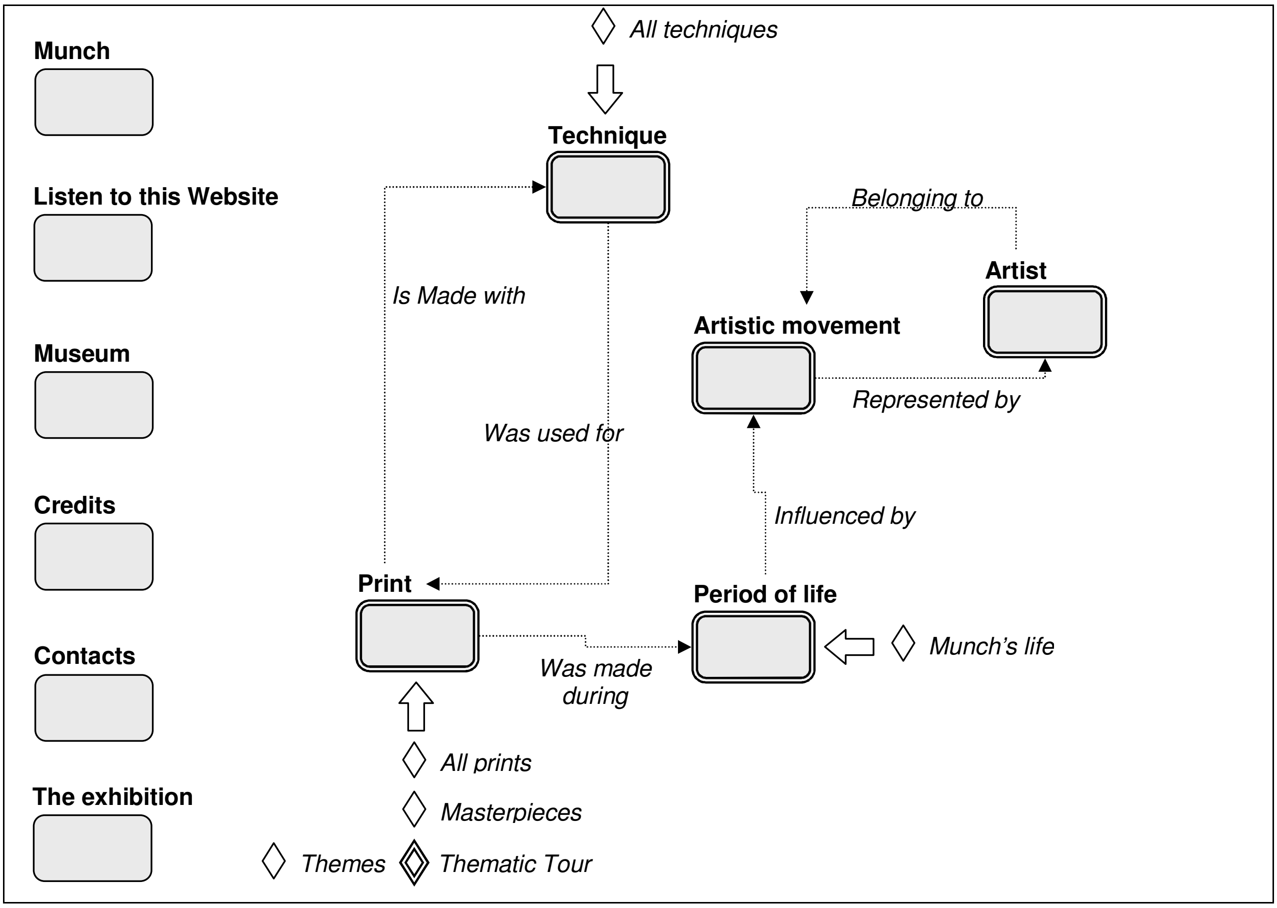


In this way the user can find contact information, can understand how to reach the Museum and the opening hours of the exhibition. Anyway, these are information that we had to put in the site because of the Museum in which the exhibition has taken place was supposed to have a visibility in the site.

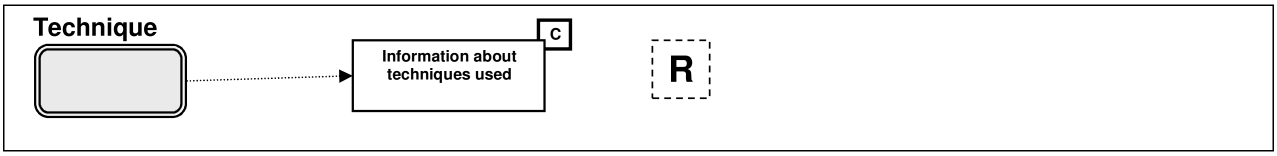


This topic is more a promotional content than a content tailored for blind users (who anyway access the acoustic site from the beginning…). Maybe is more an unexpressed requirement.

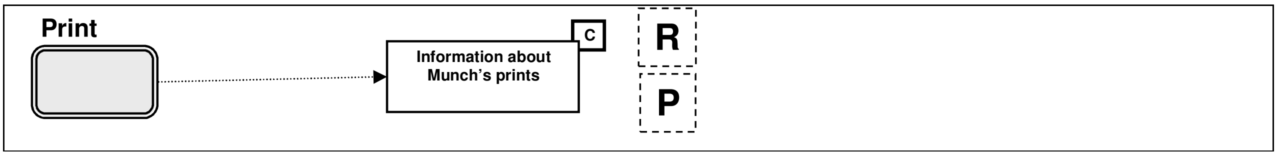
IDM Conceptual Map



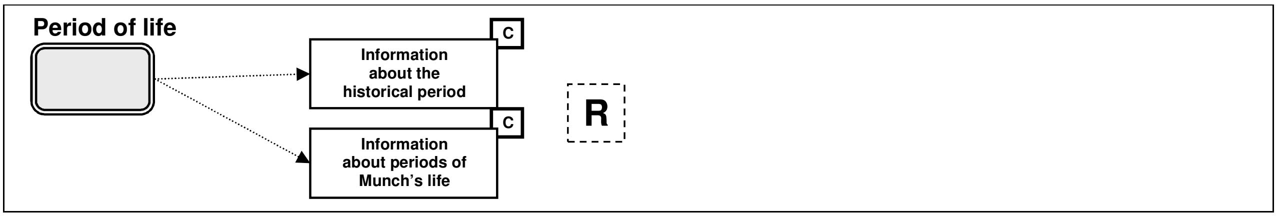
The following pages shape the relationships between design decision (for the conceptual map) and the reasons for which these decision has been taken. To indicate the source of the decision a specific taxonomy will be used: [R] to indicate that the design artefact fits with a specific requirement, [P] to indicate that the design artefacts comes from a project designer's choose and [D] to indicate that the design artefact comes from a particular understand of the application domain.



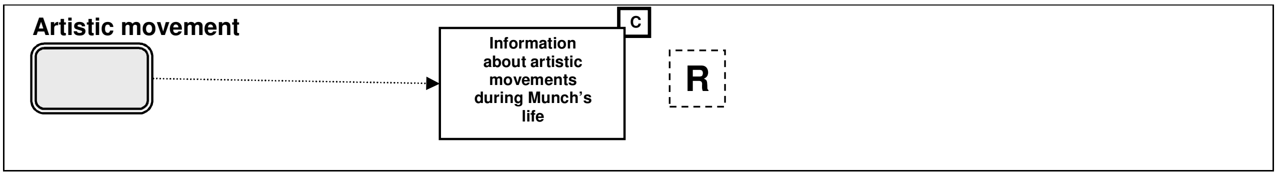
Information about techniques used fit with a specific need of the Museum: offer a cultural instrument to better understand Munch's artwork.



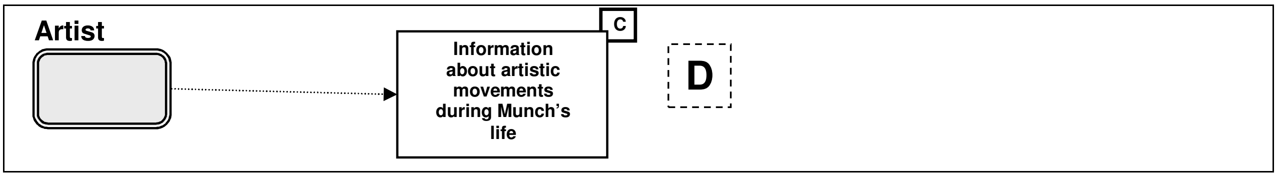
Prints are of course required as the centre of the entire application. anyway, there is not a specific indication to give such a relevance to prints, but project designer have think that in a print's exhibition site, prints should have a main position.



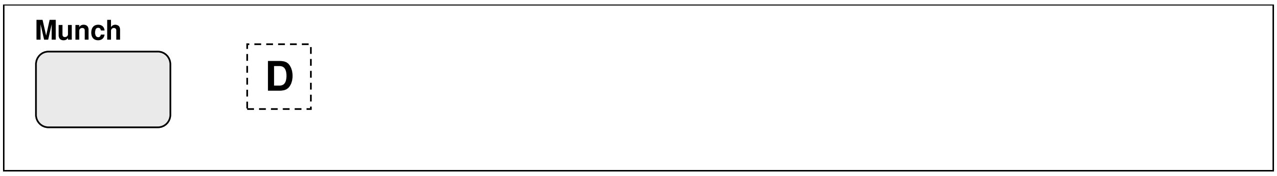
Information about periods of life make more understandable both Munch's artwork and the historical period in which Munch has worked.



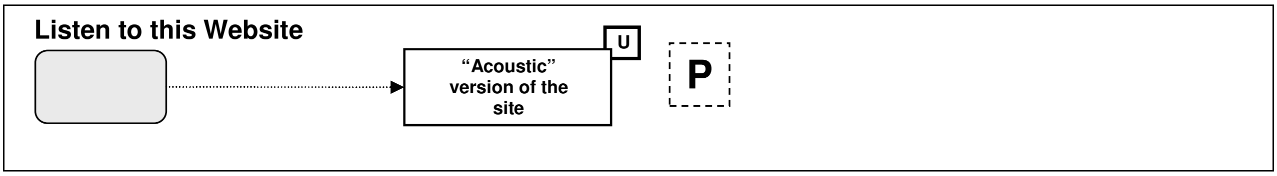
Artistic movements help in better understanding the environment in which an artwork was born.



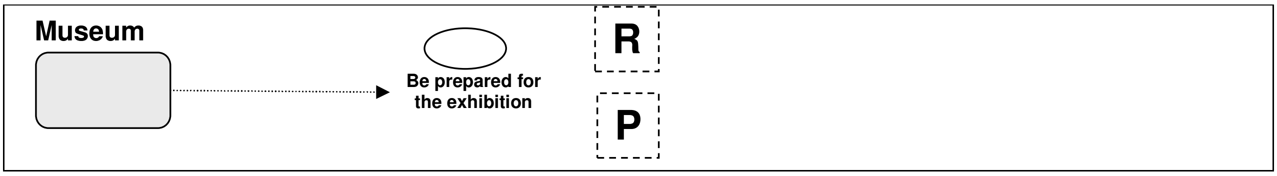
The project designer, in his understanding of the domain, has decided that information about artists are useful in better understand the artistic movements that were active during Munch's life.



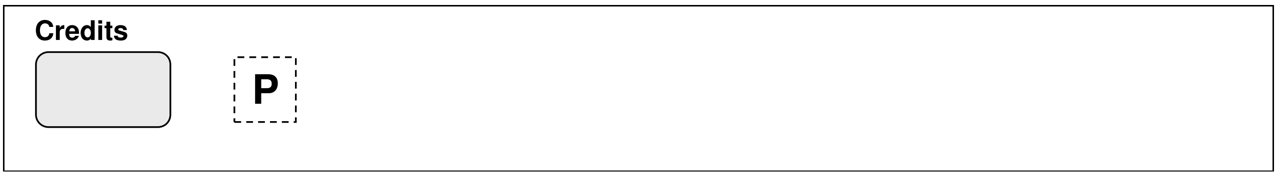
Information about Munch's life and work seem a good introduction, at least expected by anyone who think to such a web site.



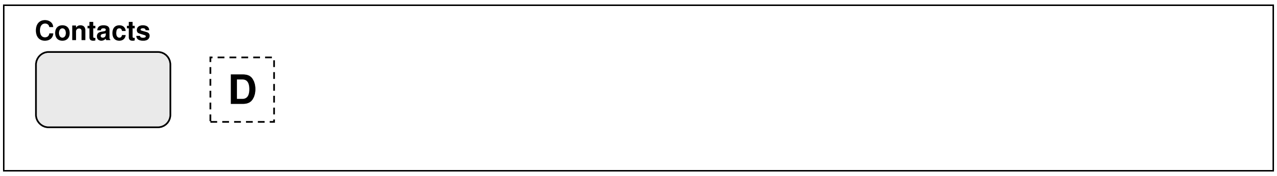
This is a promotional content, more dedicated to other research institution curious about the technology used, than to blind people (who listen the site from the beginning).



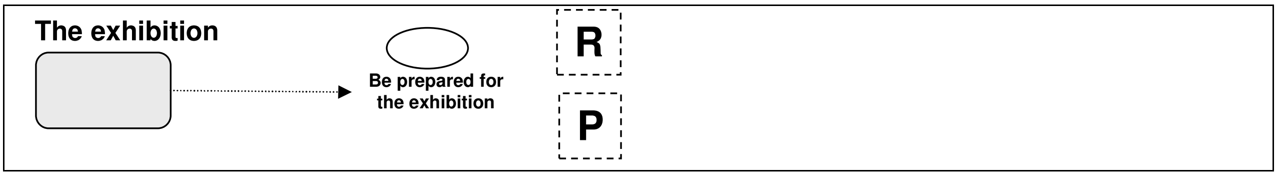
In these contents the user can find contact information and how to reach the museum.



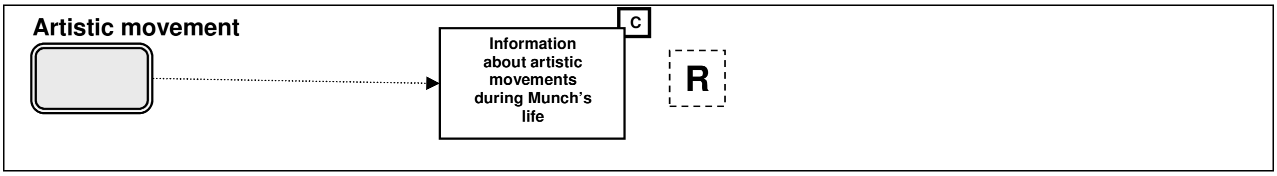
This is a "to be" content. Due to the particular technology used for the site, this could be useful for other research institution active in accessibility. This could be an "ex post" requirement: "promoting the project team".



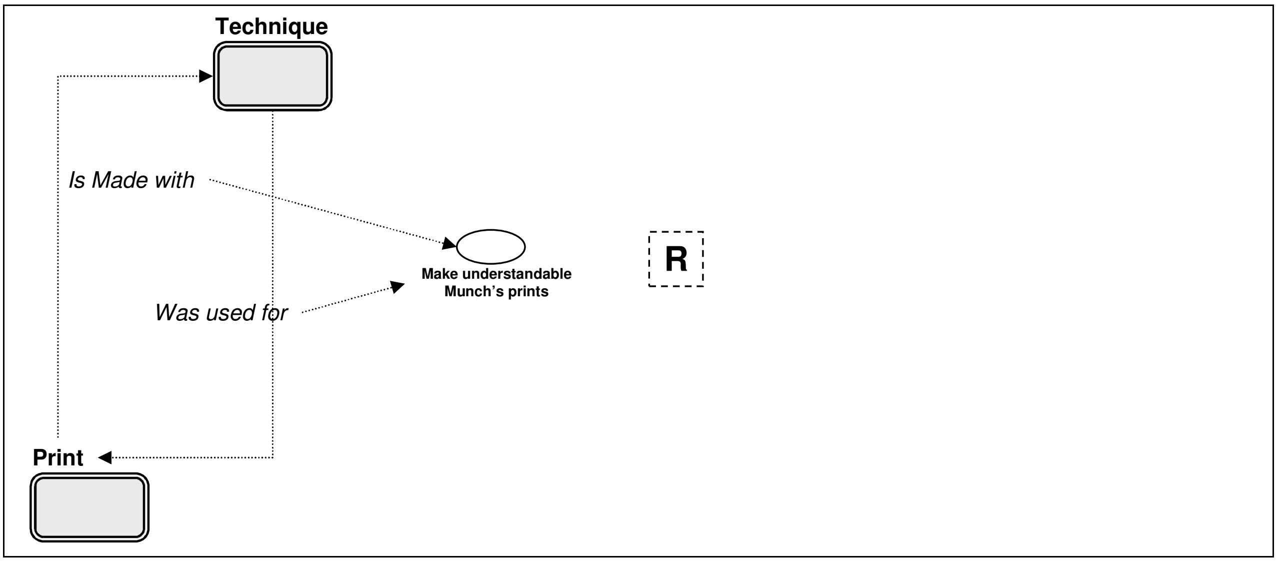
These are information that are expected by the user to be there.



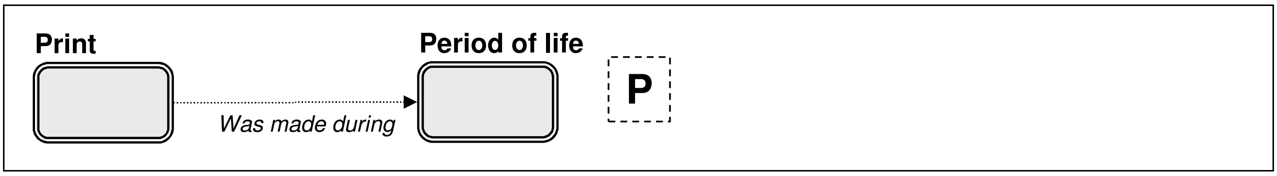
Here can be find useful information such as the opening hours. Anyway, some information about the exhibition in itself are expected in a web site of an exhibition.



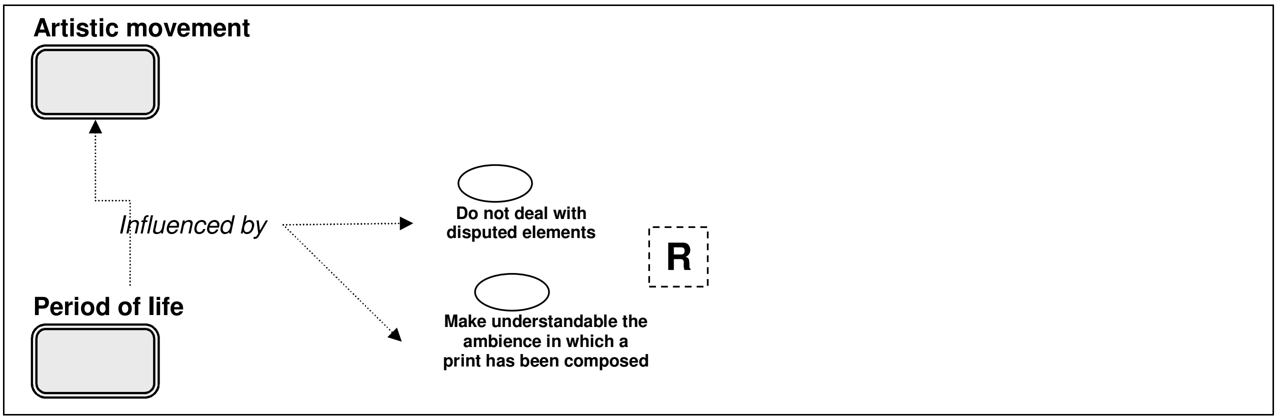
Artistic movements help in better understanding the environment in which an artwork was born.



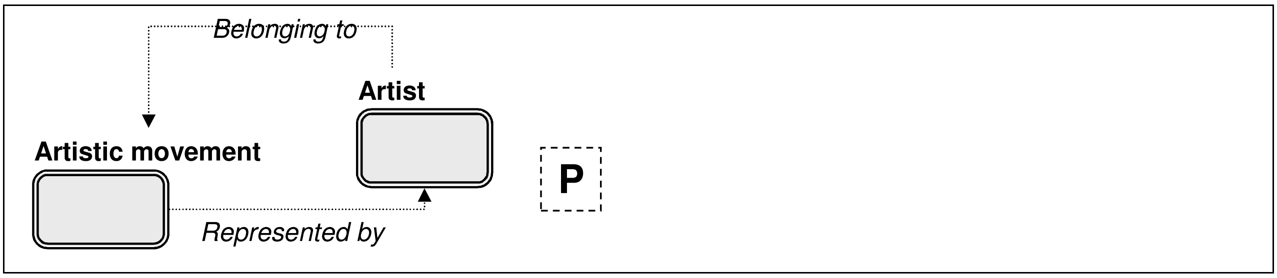
This relationship help in better understand a print. Anyway, we understood that the relationship would fit that requirement after the project designer proposed to make it (just according to good design principles).



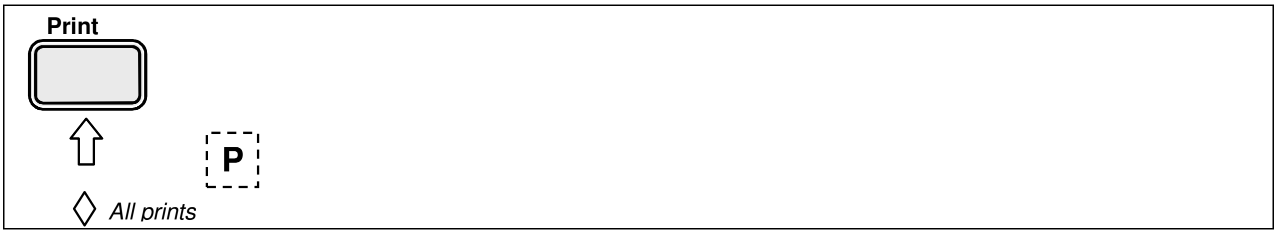
The relationship helps in better understand the context in which a print was born.



The relationship fits with two opposites goals: it is needed to make understandable the ambience in which a print has been composed but the Museum do not want to deal with a disputed element such as the artistic influence of an artistic movement on Munch's artwork. The solution is to let this relationship a little bit ambiguous, the artistic movements have just a "co presence in time" relation with a period of Munch's life.

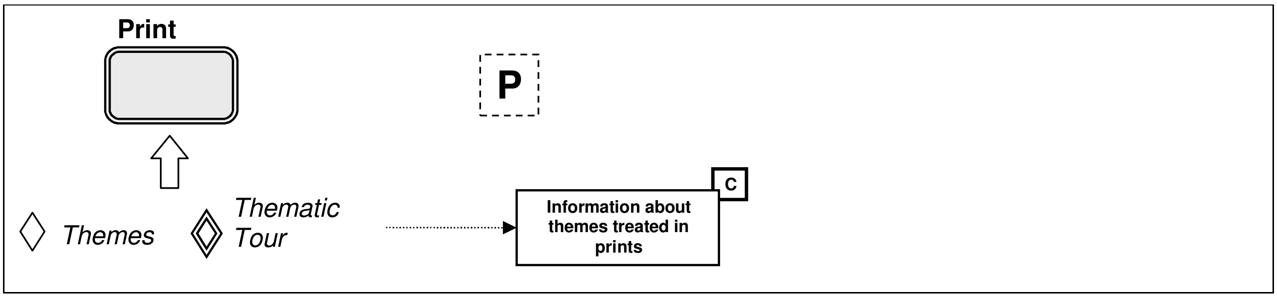


After the decision of giving information about artistic movements and artist, these relationships seemed following good design principles.

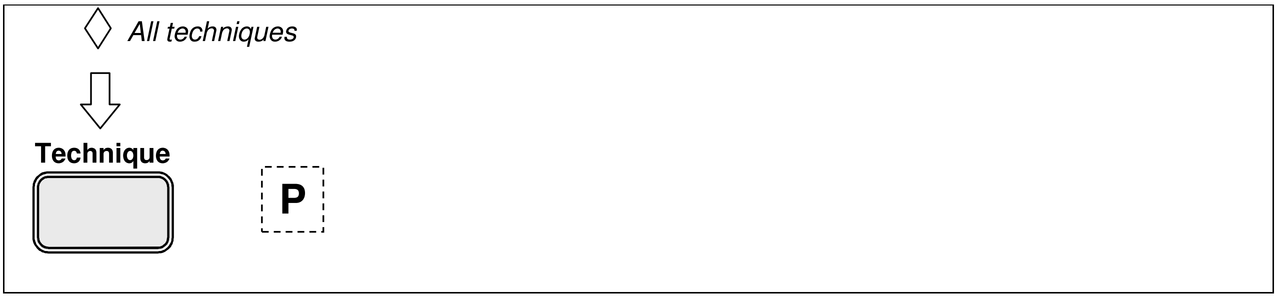


This is a “classic” collection, decided by the project designer.

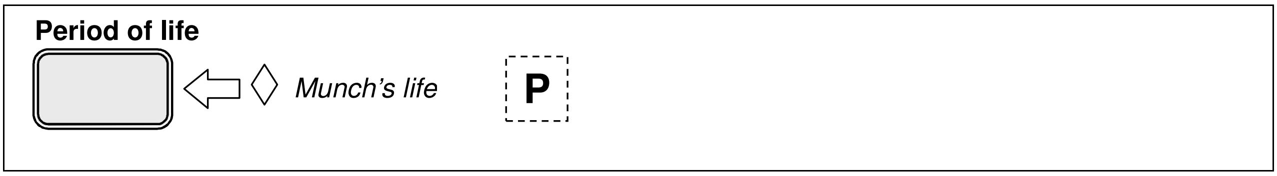
This is a “classic” collection, decided by the project designer.



Thematic tours are a Museum’s idea, but at implementation time contents needed for each theme where not available.

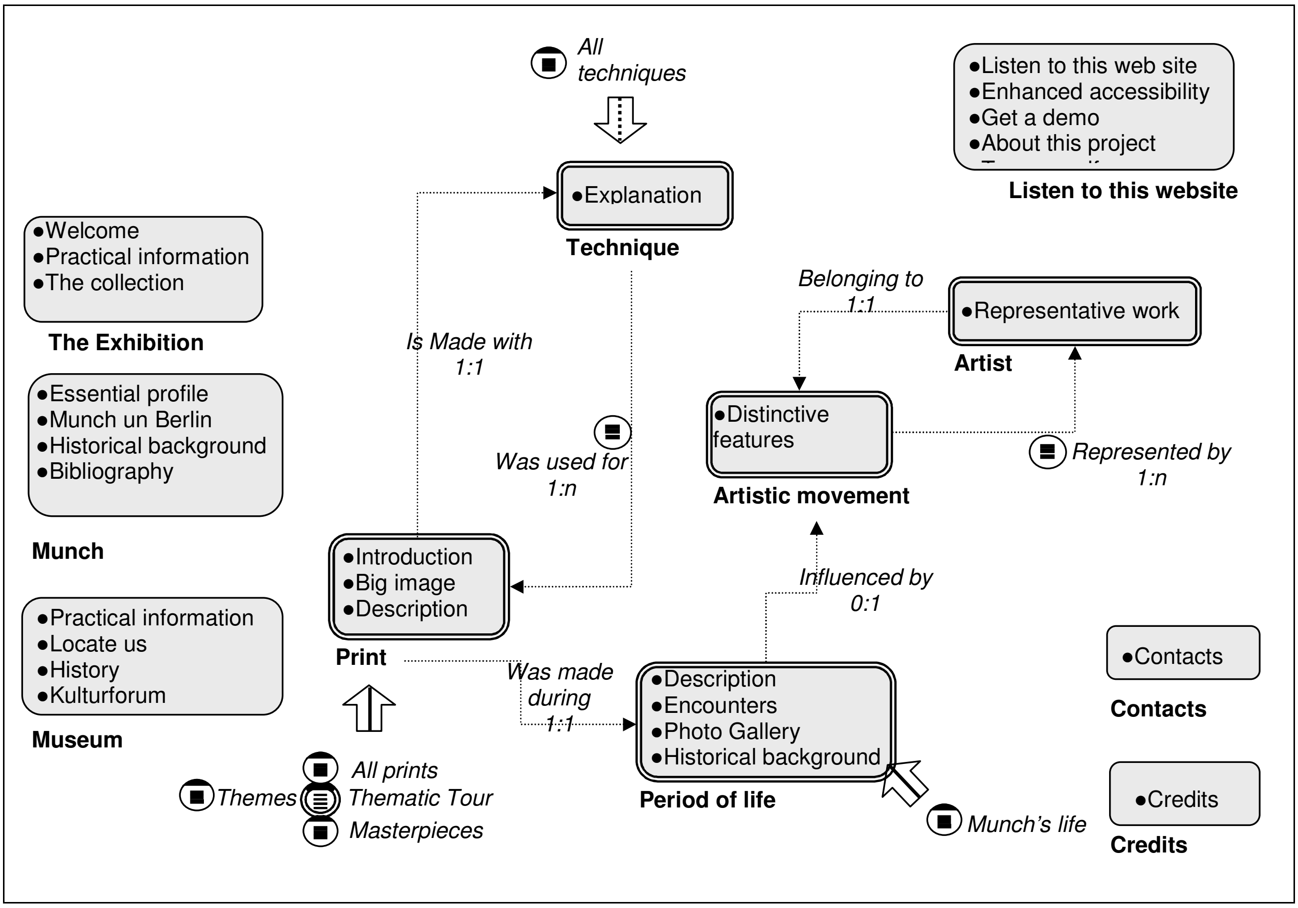


Techniques are just seven: too few for everything but this group.

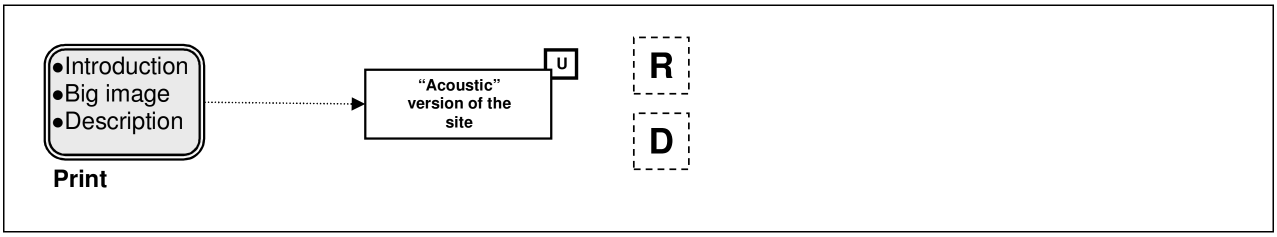


Periods of life are just 6: the project designer decided to put it all in a unique group.

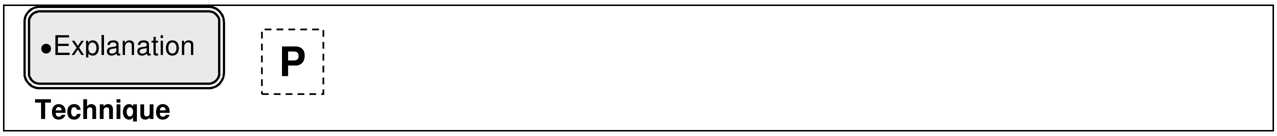
IDM Channel Map: Web



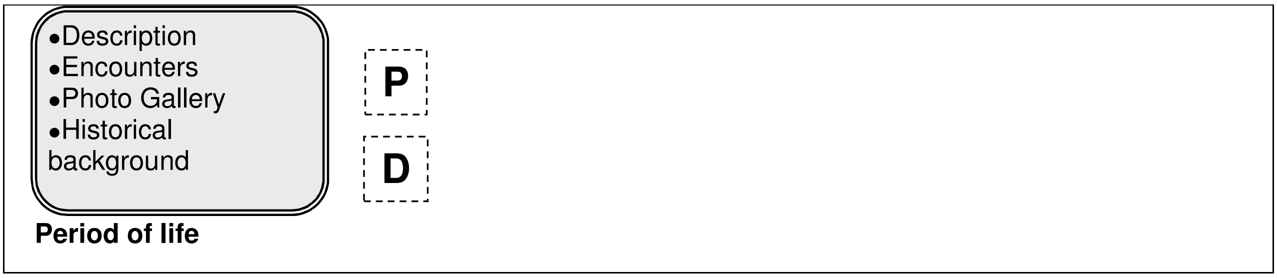
The following pages shape the relationships between design decision (for the channel map) and the reasons for which these decision has been taken. To indicate the source of the decision a specific taxonomy will be used: [R] to indicate that the design artefact fits with a specific requirement, [P] to indicate that the design artefacts comes from a project designer's choose and [D] to indicate that the design artefact comes from a particular understand of the application domain.



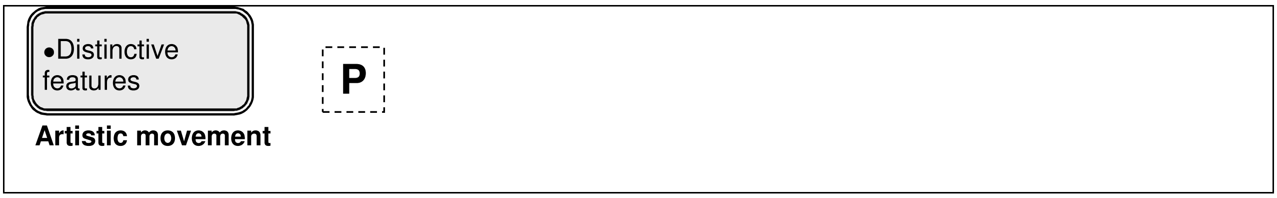
The decision of dividing the big image from the introduction and not to put an initial page with a big image has been done considering the need to offer to blind users a descriptive introduction of the print.
description and introduction were supposed to be two distinct things, but content providers didn't maintain a stylistic coherence so that this decision is now not fully understandable.



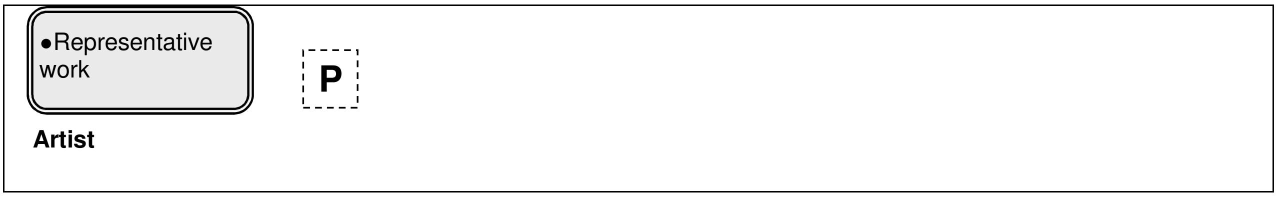
There were few contents for each technique.



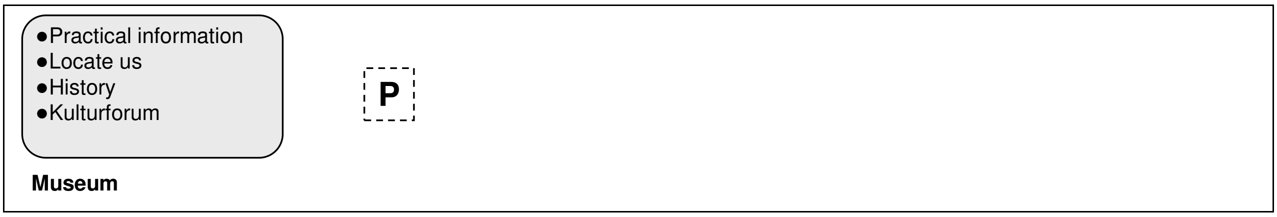
According to good design principles, it is better to separate a general description from the historical background and from a photo gallery. There is no relationships between period of life and artists; anyway an understanding of the domain has taken to the decision of highlight the artists that Munch encountered in a particular period, with no focus on a particular artistic influence.



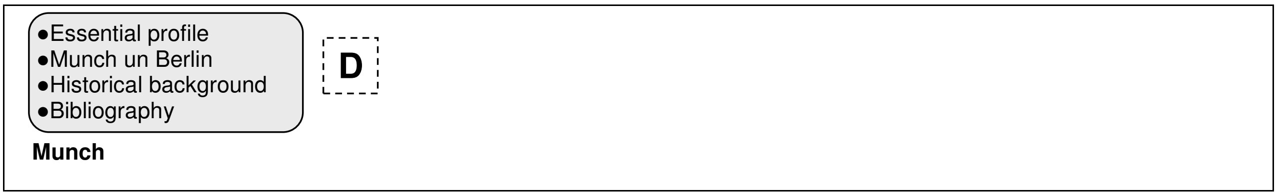
There were few contents for each movement.



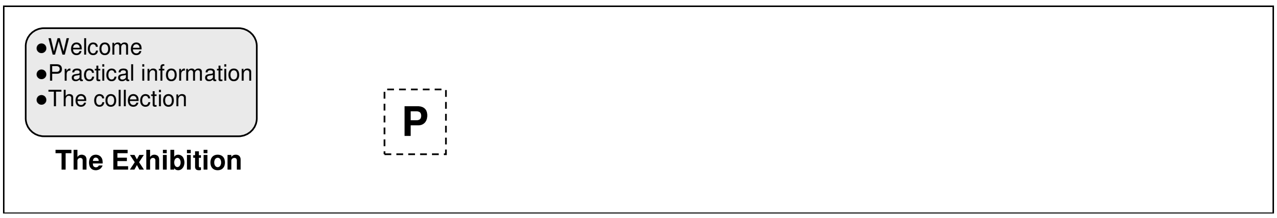
There were few contents for each artist.



The division in dialogic acts is due to clear design principles, each act supporting an activity or an goal.



These contents comes from the need of scientific strictness and from a will of completeness.



The project designer has divided the contents by kind of goal: what is this exhibition, when it will take place and what does it exhibit.

- Listen to this web site
- Enhanced accessibility
- Get a demo
- About this project
- Try yourself

P

Listen to this website

The project designer separated the contents in this topic giving to the user the opportunity to choose to download a demo or not: this is a good design principle.

- Contacts

P

Contacts

There were few contents for contacts.

- Credits

P

Credits

There were few contents for credits.

Conclusions

During the analysis two main stakeholders have been highlighted: the Berlin Museum (with cultural goals) and the Developer's team (with research goals). The users that are target of the application are the following:

- potential visitors, German and cultured
- cultured curious (not a visitor)
- blind users

The goals identified have let the development team almost free to do his work; the Museum had goals just related to the cultural "shape" given on the site about the exhibition. Therefore, the main goals were the following:

- offer cultural information to enrich the exhibition contents
- attract the users in attending the exhibition
- let blind users accessing the exhibition contents

The need of the Museum was to contextualise in history the Munch's artwork; the Museum wanted description of periods of Munch's life, but it did not want to say anything about influences of artistic movements or artists in Munch's prints. Therefore there is no explanation about the relationship between Munch and, for instance, the German Expressionism: the site highlight a simple coincidence in time.

After the design was ready, the Museum wanted to put it in informative points inside the exhibition, but the artistic vision of the Museum, even if formally correct, is not so proactive; in fact, in the site there are no storytelling, no linear contents. Therefore this site does not fit with a fruition in electronic points in the Museum.

The design has been developed taking into account the following requirements:

- do not link the artistic movements with periods of Munch's life
- do not emphasise the artistic movements
- model the application to be accessed at home

There are no explanation contents about themes that are the main access to prints. This decision is not fully understandable but has been maintained for a Museum will.
The site do not works very well in highlighting the historical periods; in fact there is a big emphasis on prints (this is of course normal) but there is a poor historical contextualisation.
Blind users can access very comfortably to the site structure, but there are some problems in contents: there are no good description of prints and the distinction between presentation and description is incoherent and stylistically unclear.

We can highlight the following suggestion for a redesign:

- prints could be organised also by period of Munch's life
- themes should be introduced in a more communicative way
- the "print" topic should be reorganised in two ways:
    - a main dialogue act, with the big image and a description for blind users
    - another dialogue act with a stylistic comment of the print

Traceability Report 3
Pompei Archaeological Site

Date: December, 2003
Target: Project Manager and Designers
Goals: Refine and align requirements and design

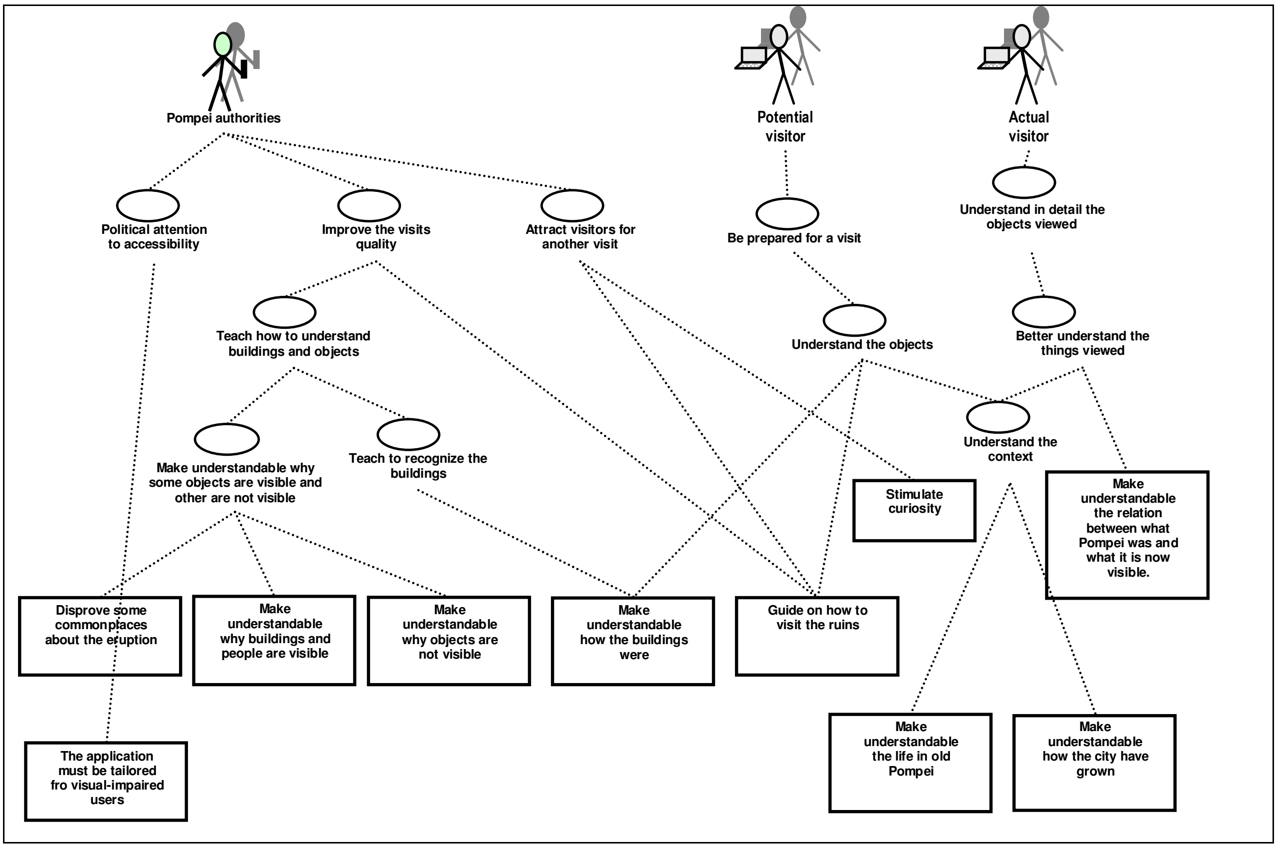
Pompei archaeological site on line

This analysis is concerned with the development of a web application about the Pompei archaeological site; the prototype application is being developed by the Hypermedia Open Centre team at Politecnico di Milano for the ministerial authorities in charge to manage the Pompei heritage. An encyclopedic and more institutional web application is currently online and should not be replaced or replicated. The application that is the subject of this report aims not at describing analytically the archaeological site but it should be more "applicative", enhancing the quality and the number of the visits in Pompei.
The main objectives of this new application are therefore twofold: from one hand, it should allow the user to visit Pompei "consciously", i.e. understanding better and in a more detailed way what she/he is going to see or what she/he has just yet visited. Contexts of use are therefore the house of the users, before or after a visit. Some computers and kiosks will be placed in the park as well, just for demonstration: this solution is poorly functional but strongly promotional, in a web marketing perspective.
The application should mainly present to (potential) visitors a different key to understand the archaeological park throughout thematic paths and provide in a clear and simple way information about what Pompei was before the Vesuvio eruption, in order to attract the user in visiting it. As a subordinate goal, the application should attract in visiting also the wider vesuvian area around Pompei.
The original characteristic of this application is its attention to the accessibility problem: the web site is being developing with a novel technology that go behind the current approach enabling a more involving access experience for visual-impaired users.

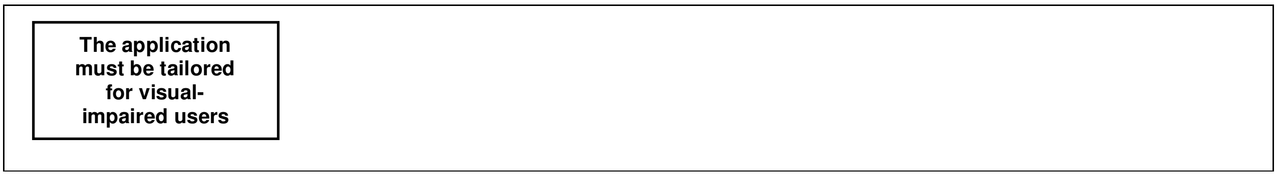
The traceability analysis' objectives are twofold: (i) refine and align the requirements and the design documentation and (ii) pave the grounds for refining and correcting the design in a stakeholders and goals-oriented perspective.

Requirements traceability – Requirements satisfaction model

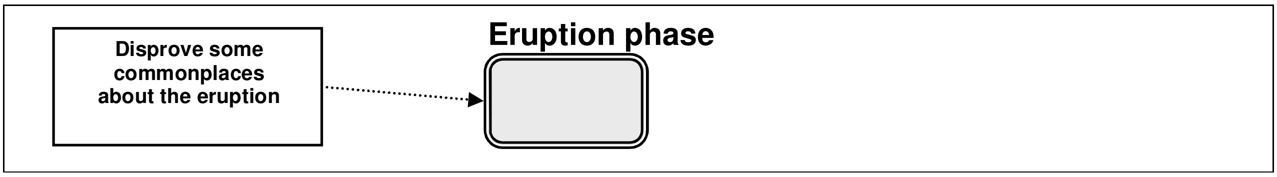
The following schema shows a synthetic view of the high-level goals, stakeholders and requirements related to the Pompei web site.



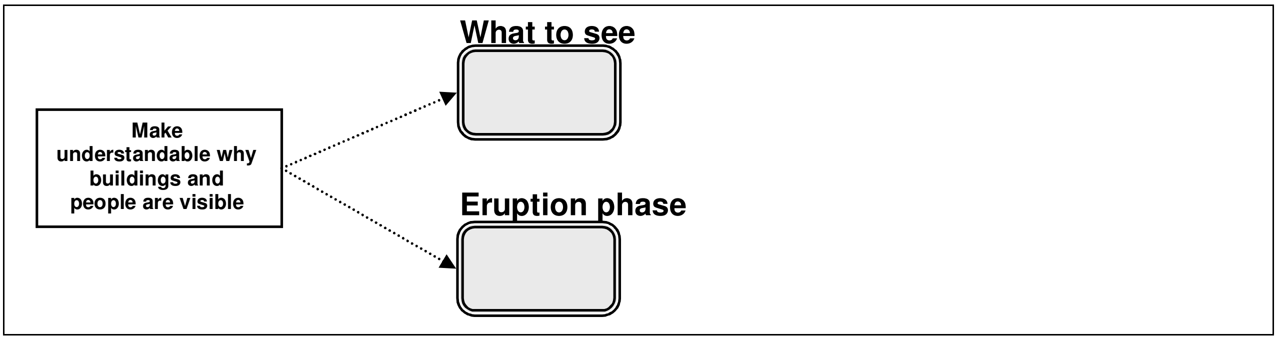
In the next pages a synthesis of how these requirements have been taken into account in the design of the web site will be presented.



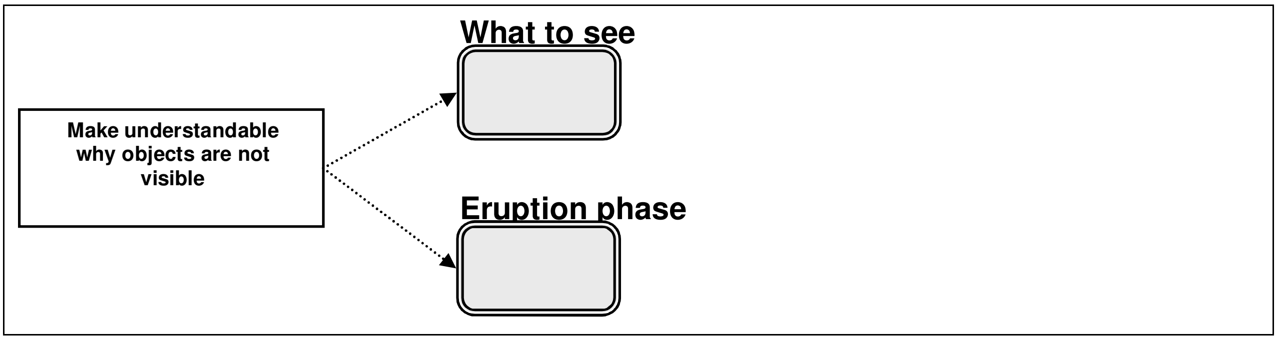
IMPACT: For the moment it is not possible identify specific accessibility elements in the design.
TAXONOMY: No relation
RELEVANCE: High – This requirement must be strongly taken into consideration



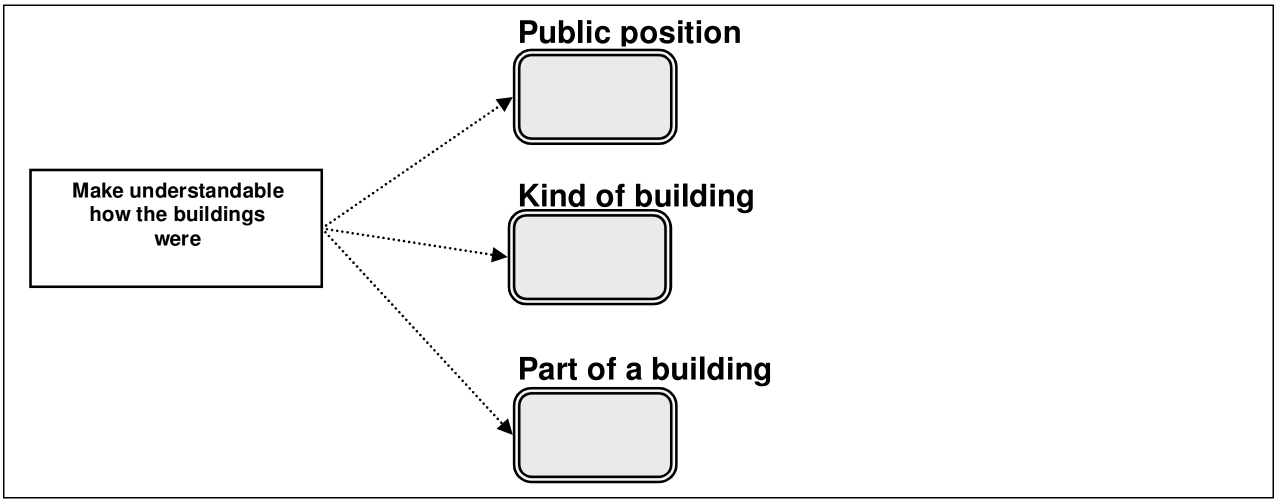
IMPACT: Commonplaces are mainly related to how the eruption happens; this topic helps in making understandable why objects and people are now placed in a certain site and in a certain way.
TAXONOMY: Relation linked to an understanding of the domain.
RELEVANCE: Medium



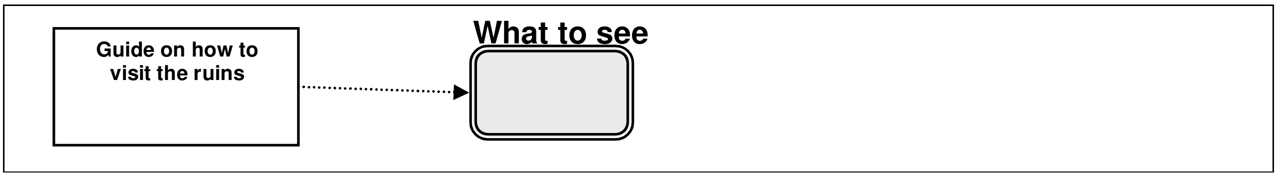
IMPACT: The reason why some buildings remains in a certain way is explained by both the eruption phases and in the context of what to see.
TAXONOMY: Relation linked to an understanding of the domain.
RELEVANCE: Medium



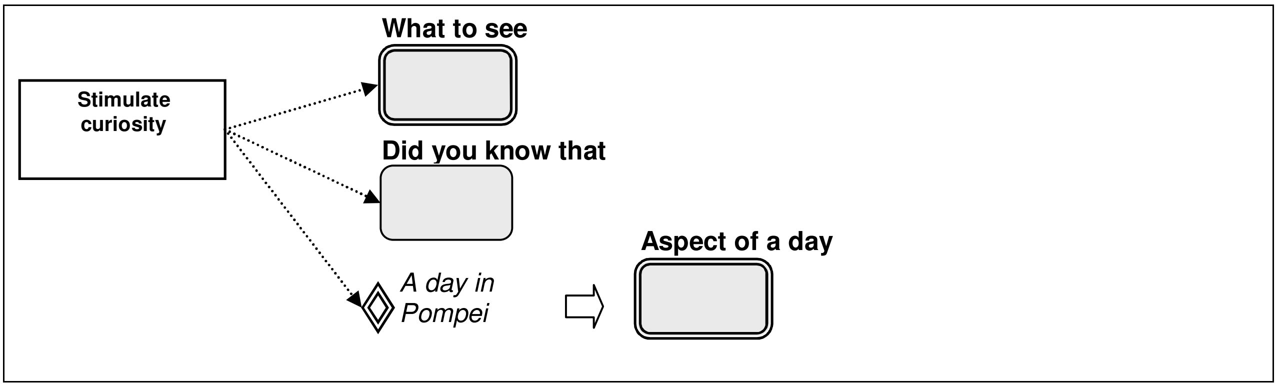
IMPACT: In some cases the eruption phases make us understandable why some objects remained, as well as the fact that some objects are in the museum is referred in “what to see”.
TAXONOMY: Relation linked to an understanding of the domain.
RELEVANCE: Medium



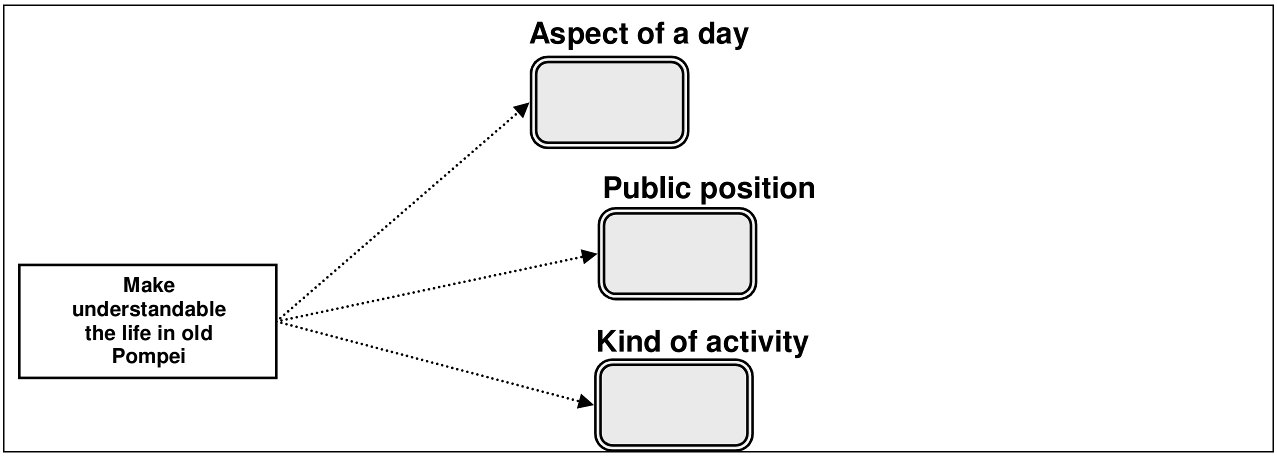
IMPACT: more than a plain encyclopedic list of all the buildings, the explanation of buildings typologies (and in more difficult cases of buildings parts) helps in understand what is visible. Single buildings are used as examples. To understand a building an understanding of who lived there and of what use made of it is needed.
TAXONOMY: Relation linked to an understanding of the domain.
RELEVANCE: Medium



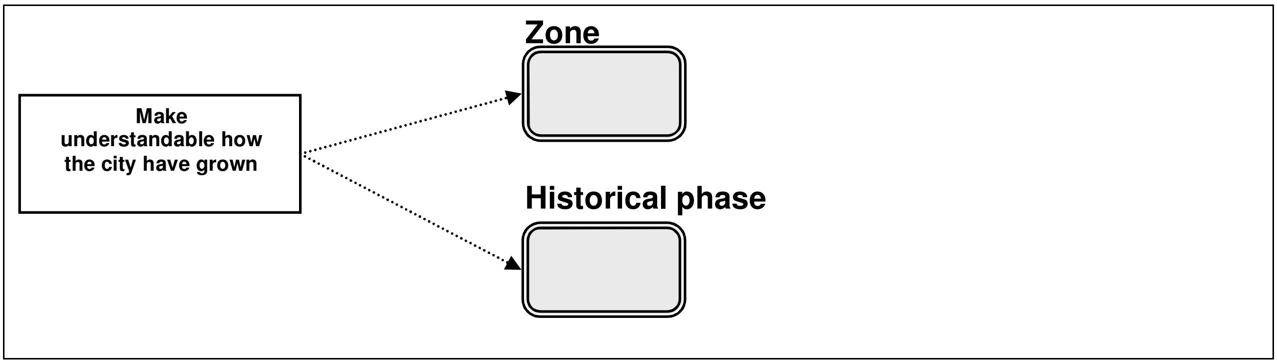
IMPACT: The simpler solution includes indications about what to see. This point seems to be a little weak: one can conceive guided tours between the contents of the site, structured as a typical day in Pompei.
TAXONOMY: Cause-effect relation.
RELEVANCE: High



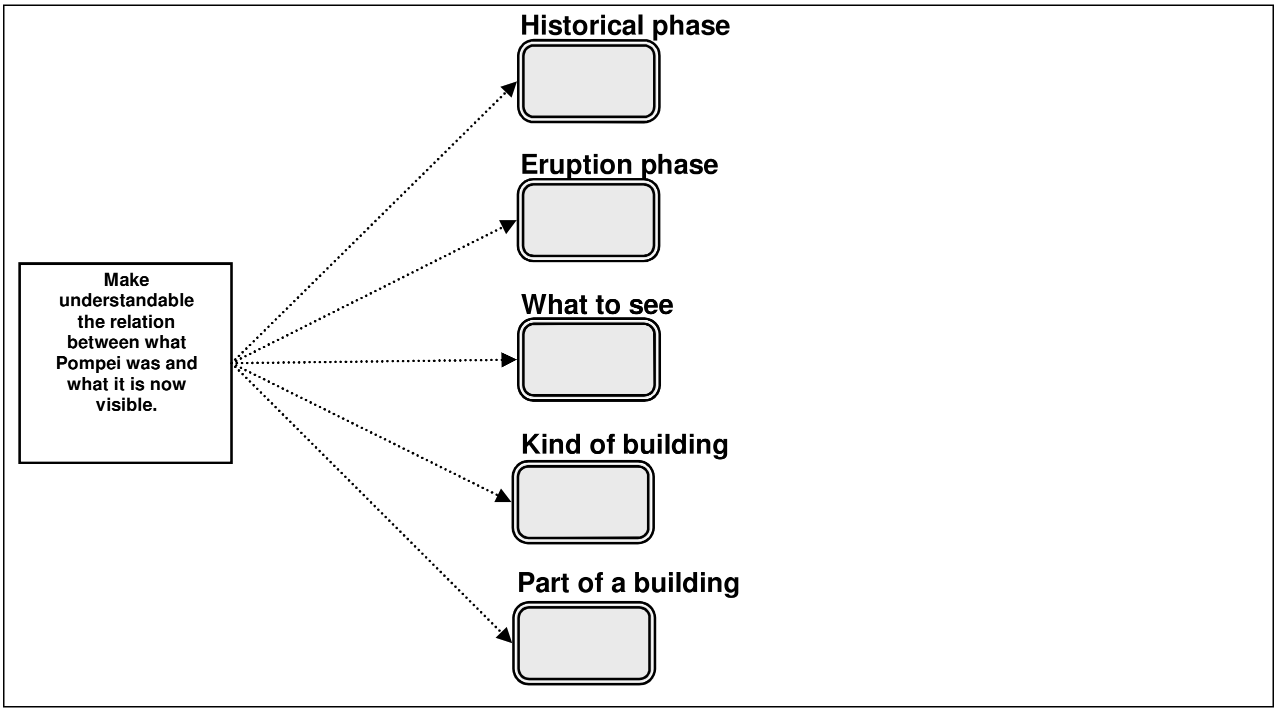
IMPACT: Three strategies to stimulate curiosity: a series of curious and attractive anecdotes (did you know that), a structured review of interesting thins to watch (what to see) and guided tours that reproduce the aspects of a day in Pompei.
TAXONOMY: Relation linked to a communication strategy
RELEVANCE: High



IMPACT: In this case too, different strategies allow to understand the everyday life in Pompei, through activities, aspects of a day and people that lived there.
TAXONOMY: Relation linked to an understanding of the domain.
RELEVANCE: Medium

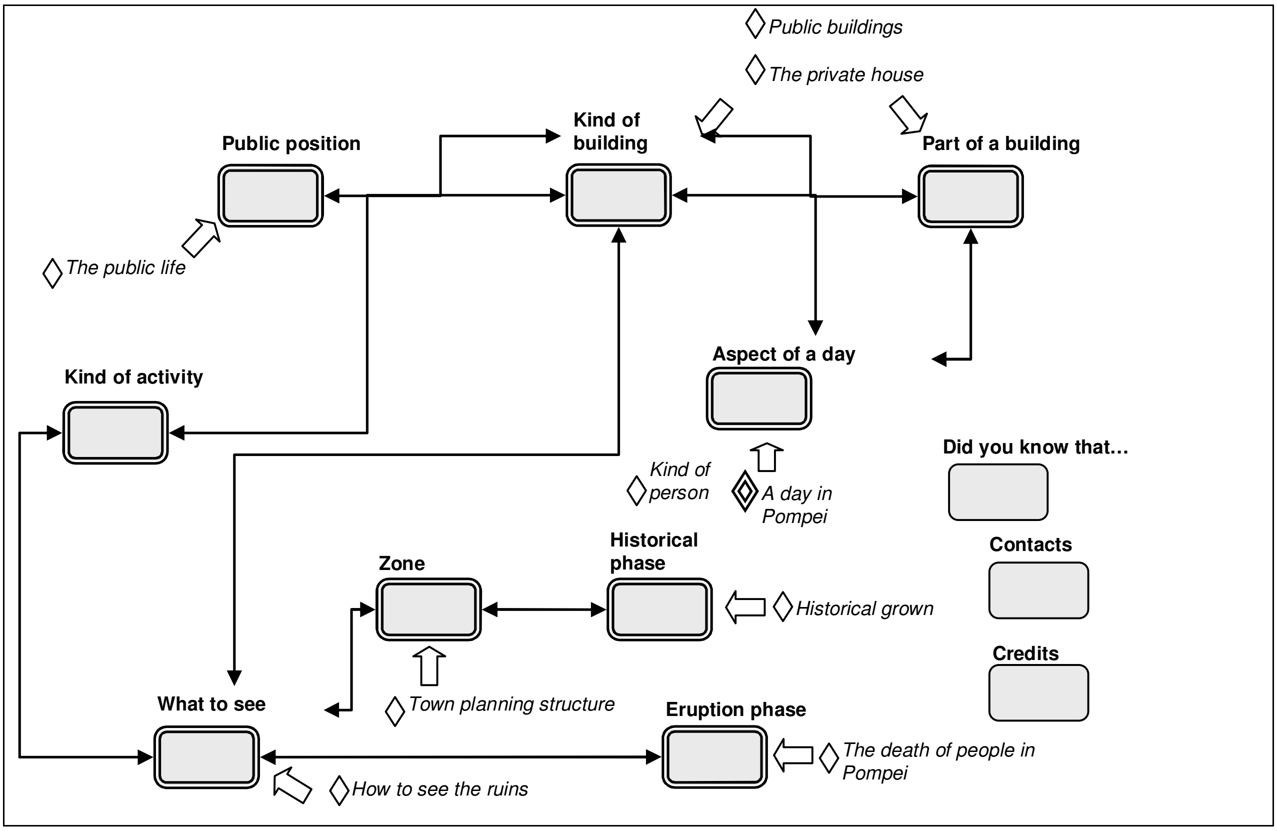


IMPACT: The town grown help in understand what is visible in Pompei. The need is filled by the description of the historical grown of the city and of the different zones in which the town is divided.
TAXONOMY: Relation linked to an understanding of the domain.
RELEVANCE: Medium

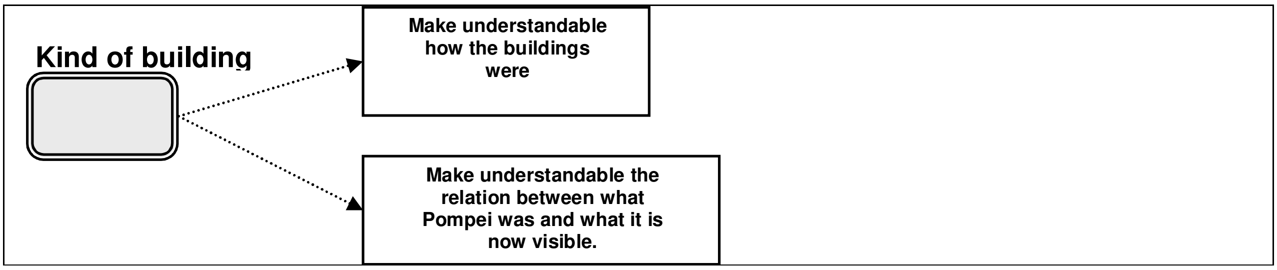


IMPACT: One try to use different strategies, discussing the historical city grown and the eruption phases (what the eruption modified), highlighting the differences in what to see or making understandable how a building was and how it is now, with examples of kind or of parts of building.
TAXONOMY: Relation linked to an understanding of the domain.
RELEVANCE: High

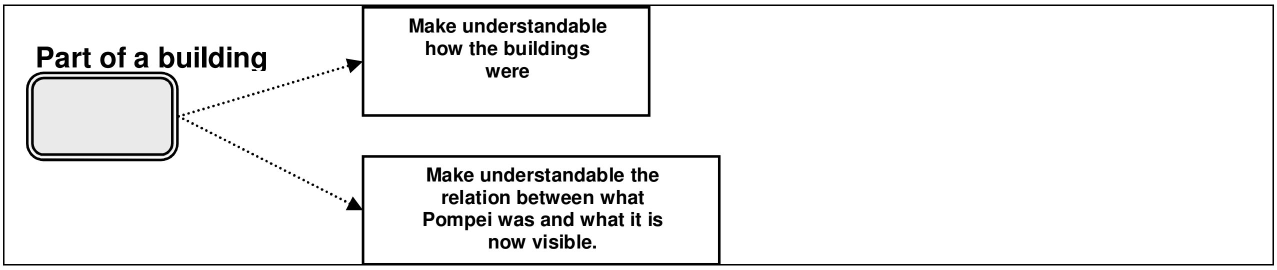
Design traceability – Design justification model



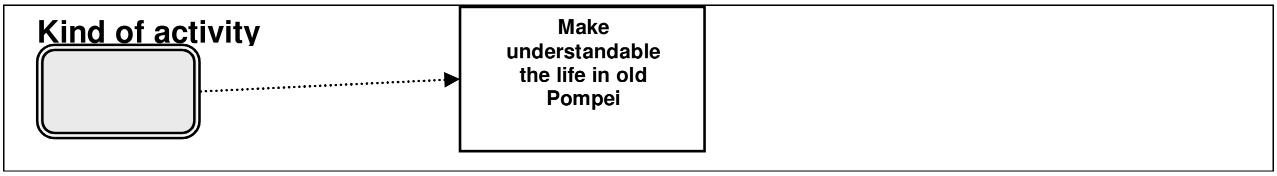
The following pages shape the relationships between design decision (for the conceptual map) and the reasons for which these decision has been taken. To indicate the source of the decision a specific taxonomy will be used: [R] to indicate that the design artifact fits with a specific requirement, [P] to indicate that the design artifacts comes from a project designer's choose and [D] to indicate that the design artifact comes from a particular understand of the application domain.



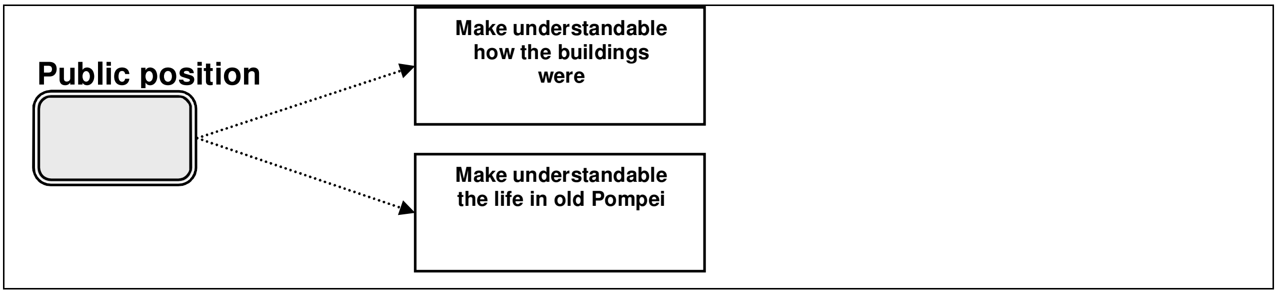
R >> The goals is not to show all the houses but it is to teach in recognize and understand the buildings. One use actual data as example. The site supports the visit but it is not encyclopedic. It is conceived to make understandable how and why a building has been made in a certain way, how it was and how it is now.



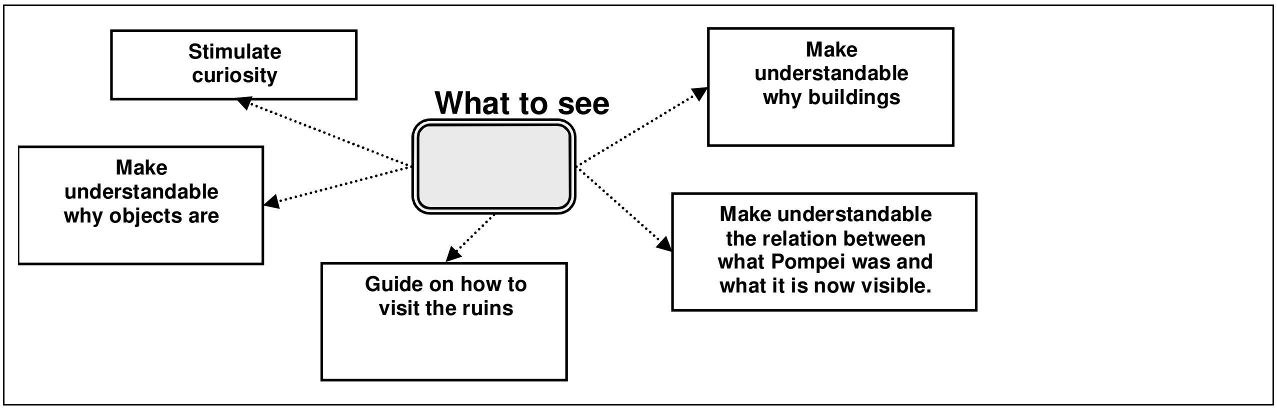
R >> To some more complex buildings, a further detail level is useful to explain the functionalities of each part of it.



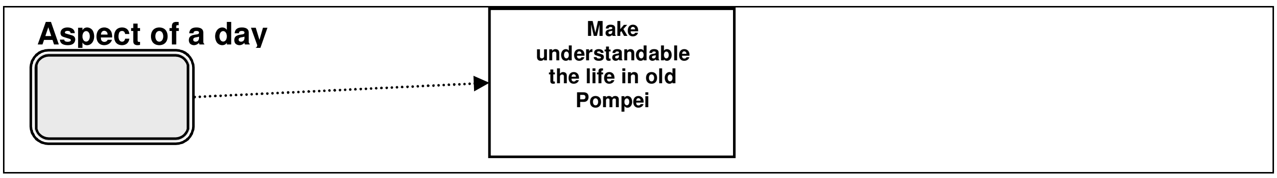
R >> It helps in understand what it is visible, why building have been made in a certain way, etc.



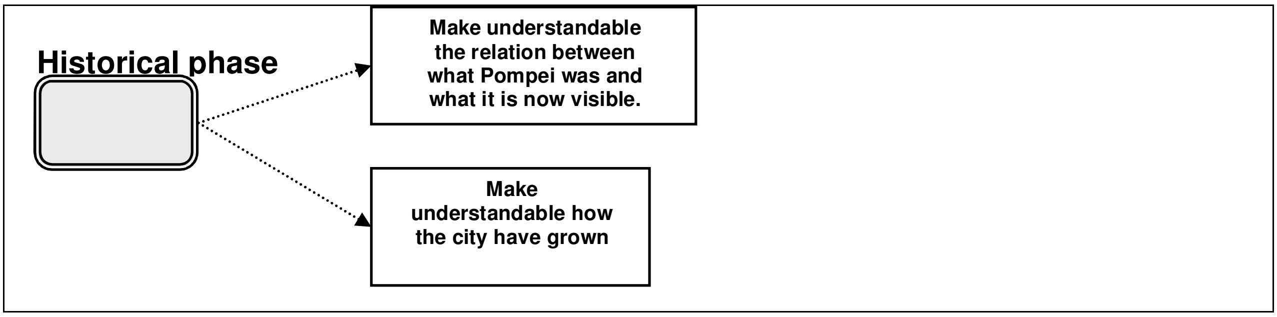
R >> It makes understandable who lived the buildings placed there and why are they made in such a way.



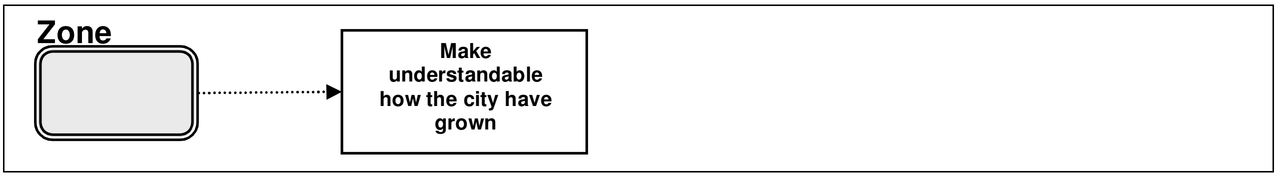
R >> It helps in attracting the visitor about some details to see and in clarify why something is visible or not. Furthermore, the “what” to see may be the source for good advices about elements to focus on during the visit, highlighting their relationship with the past.



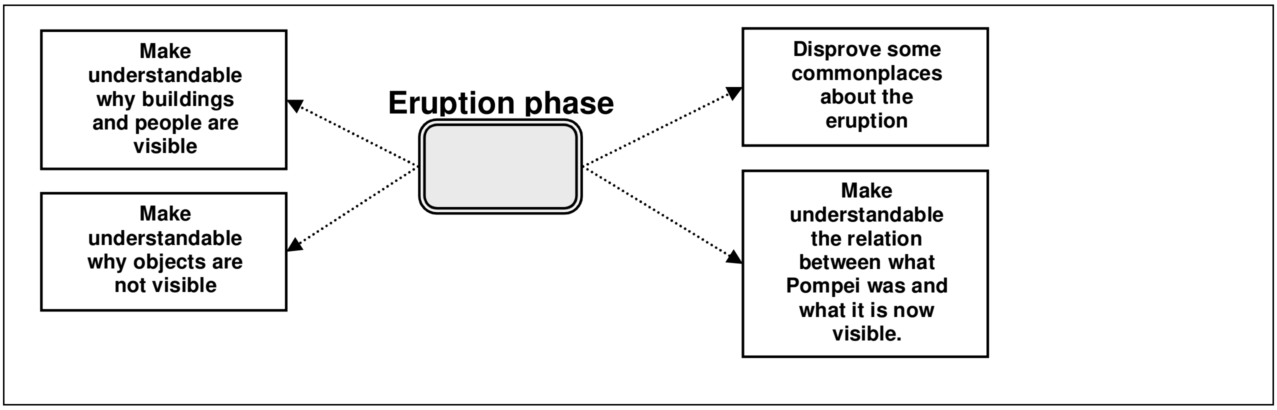
R >> It helps in understand the life in old Pompei, clarifying why some buildings and some spaces are made in a certain way.



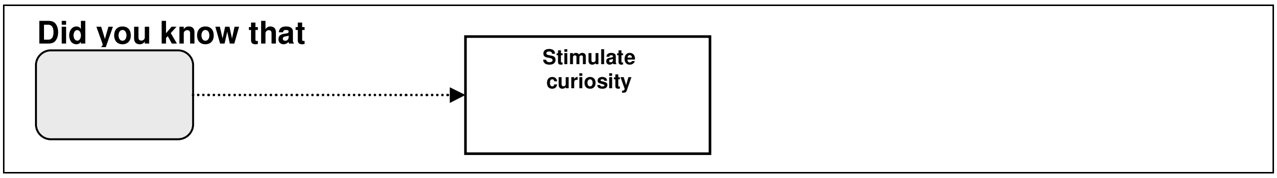
R >> The town grown helps in understand what is visible in Pompei and what is its relationship with how the city was.



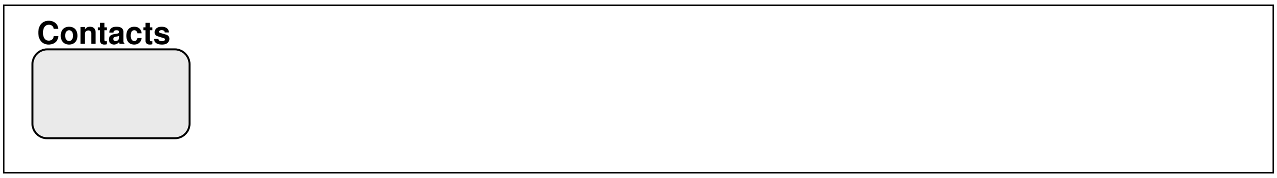
R >> The town grown is supported by the description of the city zones.
P >> To describe the zones helps the user in the general understanding of what is visible.



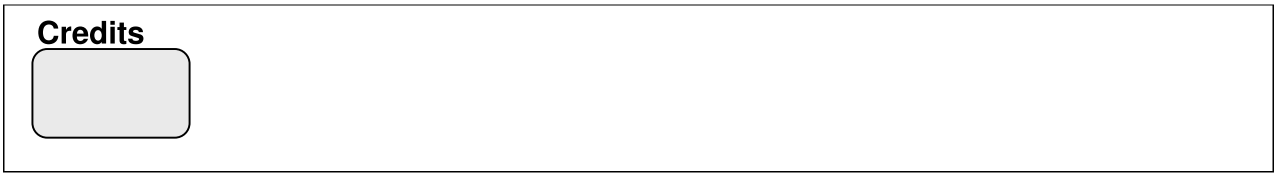
P >> The town has not been covered by the lava. There are many commonplaces related to Pompei, mainly related to the iconography about the eruption. To well understand how it happened helps in understand why something is visible, why objects remains in a certain way and which relationships they have with what there was.



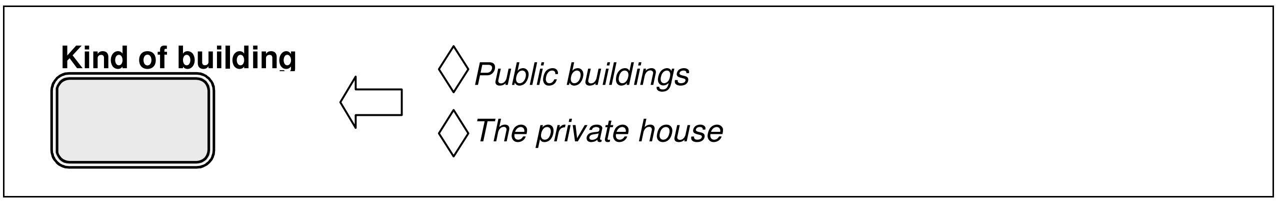
R >> It is useful mainly to attract users through curious or not well known facts.



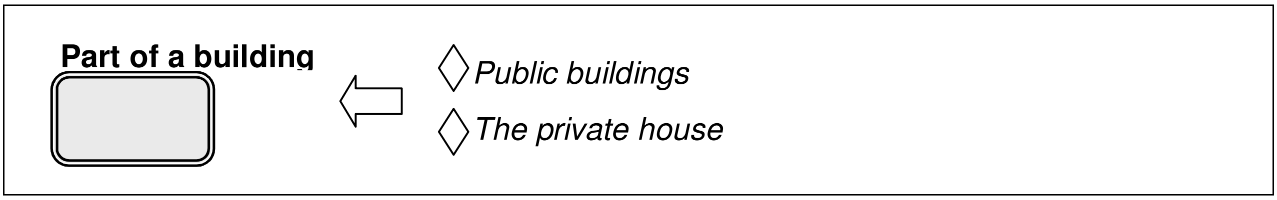
P >> Partially to support future visits, but mainly because everyone expects something like that.



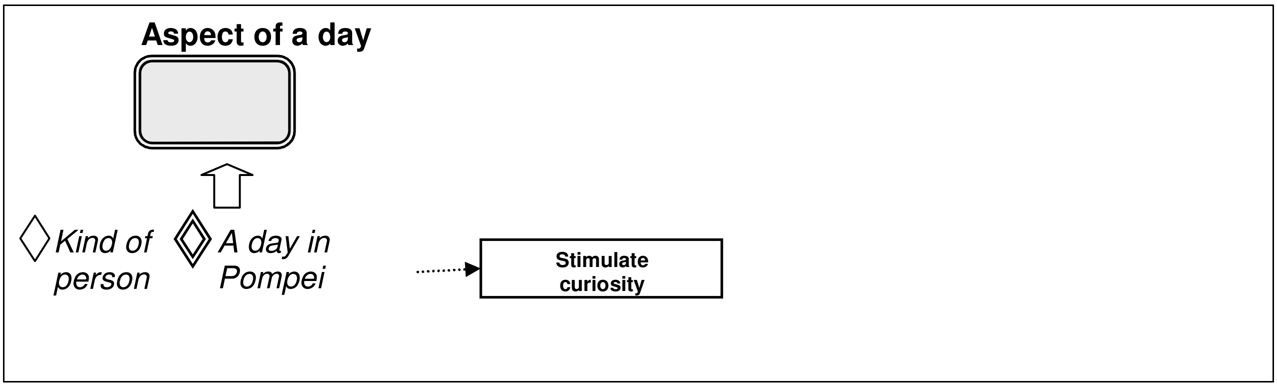
P >> "This things are placed in every web site…"



P >> The access schema is simple because cardinalities are low.

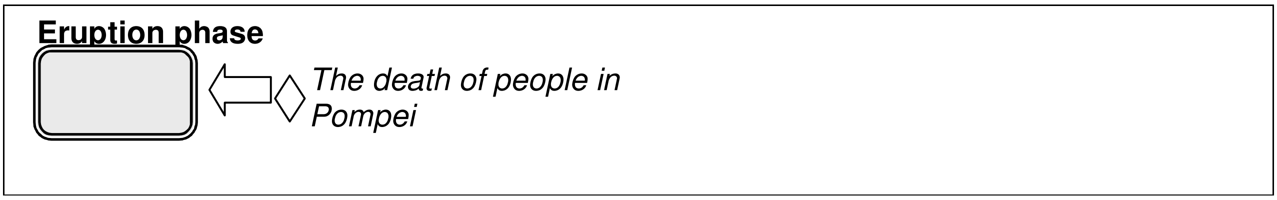


P >> The access schema is simple because cardinalities are low.



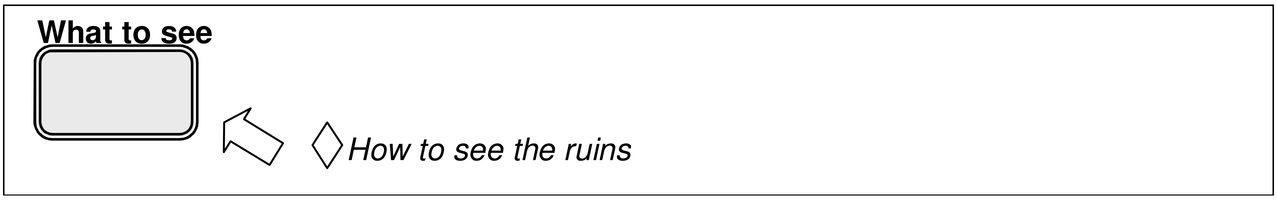
P >> Guided tours? They guide suggestively a non expert user, so they aggregate meaningfully the contents. Idea: a day in Pompei.

P >> The access schema is simple because cardinalities are low.

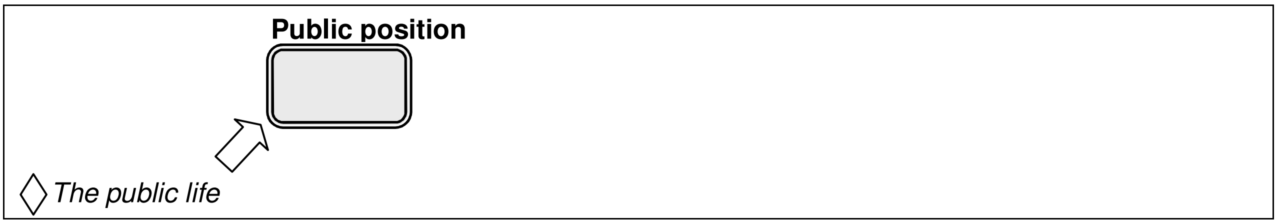


P >> The access schema is simple because cardinalities are low.

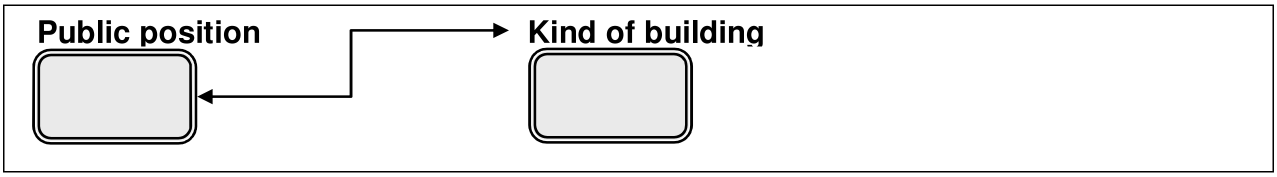
P >> The access schema is simple because cardinalities are low.



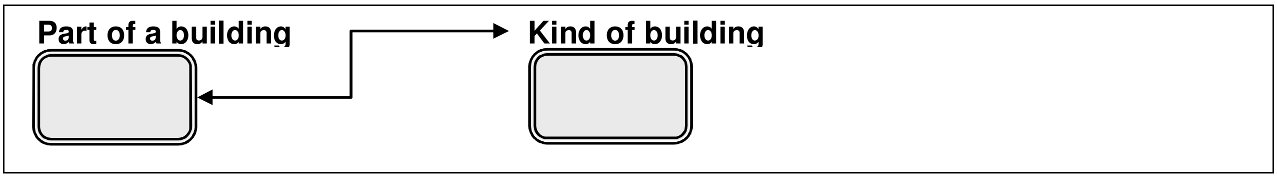
P >> The access schema is simple because cardinalities are low.



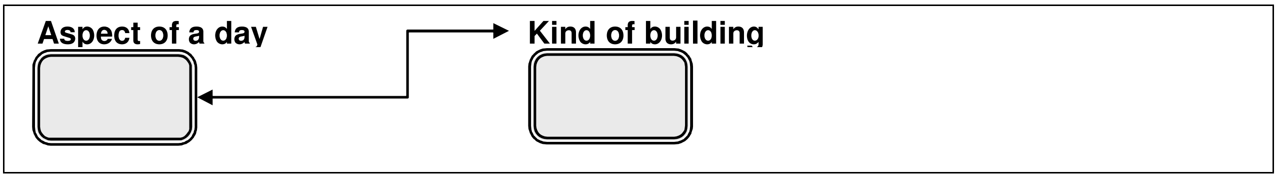
P >> The access schema is simple because cardinalities are low.



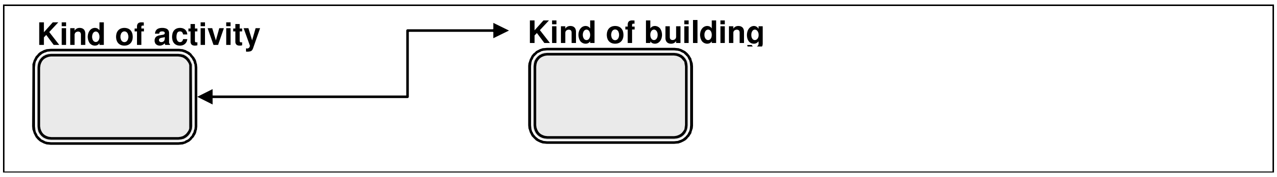
P >> This relationships make better understandable what can be watch in Pompei; it does not derive from a requirement, it is part of the designers' skills.



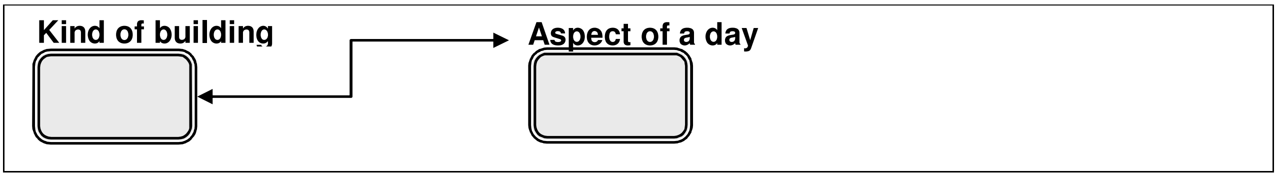
P >> This relationships make better understandable what can be watch in Pompei; it does not derive from a requirement, it is part of the designers' skills.



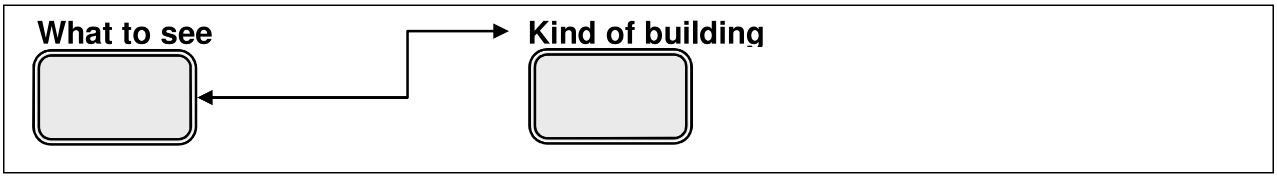
P >> This relationships make better understandable what can be watch in Pompei; it does not derive from a requirement, it is part of the designers' skills.



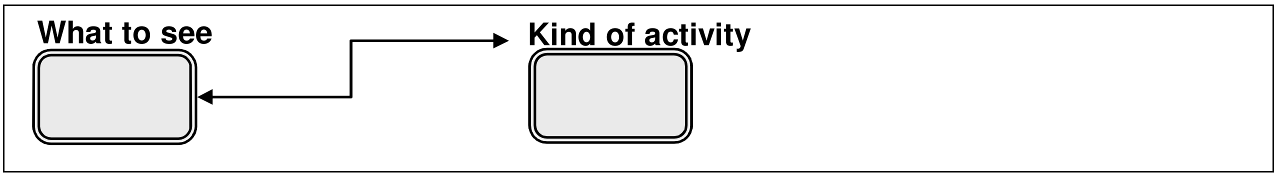
P >> This relationships make better understandable what can be watch in Pompei; it does not derive from a requirement, it is part of the designers' skills.



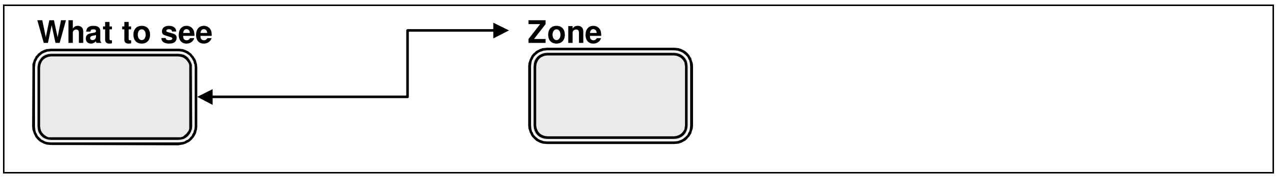
P >> This relationships make better understandable what can be watch in Pompei; it does not derive from a requirement, it is part of the designers' skills.



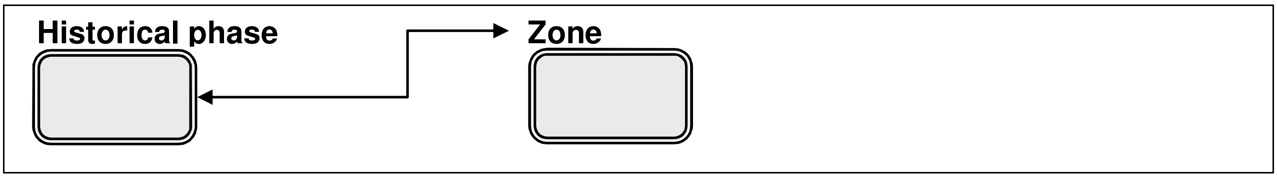
P >> This relationships make better understandable what can be watch in Pompei; it does not derive from a requirement, it is part of the designers' skills.



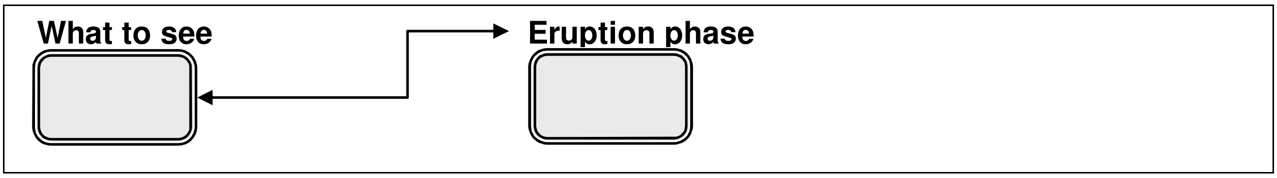
P >> This relationships make better understandable what can be watch in Pompei; it does not derive from a requirement, it is part of the designers' skills.



P >> This relationships make better understandable what can be watch in Pompei; it does not derive from a requirement, it is part of the designers' skills.



P >> This relationships make better understandable what can be watch in Pompei; it does not derive from a requirement, it is part of the designers' skills.



P >> This relationships make better understandable what can be watch in Pompei; it does not derive from a requirement, it is part of the designers’ skills.

Conclusions

In this report the relationships between the design choices and the requirements expressed for a web application about the Pompei archeological site have been highlighted. Some strength and weak points may be reported.

Strength points:

- The application answer very well to the cultural needs of the users; in particular, the user can understand the buildings visited (or to visit) and the kind of these buildings; she/he may understand also the kind of life in the I century d.C. and how the everyday activities were organized: this make the structure of buildings more understandable.
- The application make also more understandable why some buildings are in a certain way and the use of some parts of these buildings in specific moments of the day.
- The site helps also in disprove the commonplace about the Pompei eruption: lava was not there!

Weak points and possible improvements:

- The application do not support very well the user in how to visit the ruins: the argument is subjective and it is difficult to organize a visit in a hypermedial way. One can conceive guided tours between the contents of the site, structured as a typical day in Pompei.
- The application do not explain very clearly why some objects are exhibit in the park and other in the museum, which kind of object remains, which ones have been lost and why. This point have to be enhanced to make more understandable how the houses where in old Pompei.
- The application has a lack of focus on people founded death in Pompei, on where and how and why have they been founded in a certain way.
- Curiosities and anecdotes are difficult to structure in a organic way, but this is a content that need more emphasis. One may transform the topic "did you know that" in a multiple topic.
- The description of the relationship between Pompei nowadays and Pompei in the I century may be enhanced, e.g. with virtual tours or 3D reconstructions of the city.

Traceability Report 4
Museum of Non-European Cultures

Date: November, 2004
Target: Designers
Goals: Refine the design

Introduction

The Museum of non-European cultures ("Museo delle Culture Extraeuropee") in Lugano opened in 1989. It houses approximately 600 objects donated to the city by Serge and Graziella Brignoni in 1980. Serge Brignoni, an accomplished and recognized painter in his own right, dedicated many years of his life to assembling the collection of objects from Oceania, Africa and India.

Although the collection is culturally significant, due to poor management and lack of promotional activities on the part of museum and city officials, it was virtually unknown in the local community. As a result, the museum received very few visitors, which led the city of Lugano to propose closing the building in 2003. Objects in the collection were to be sold or loaned to other ethnographic museums in Europe.

A local citizen group successfully challenged this proposal and, in 2004, the city agreed to reappraise the museum's situation. Following this reappraisal, the city is now planning to invest money and resources to re-launch the museum. A permanent curator will be appointed in the coming months. In addition, they are considering developing a website and other interactive applications to support the re-launch.

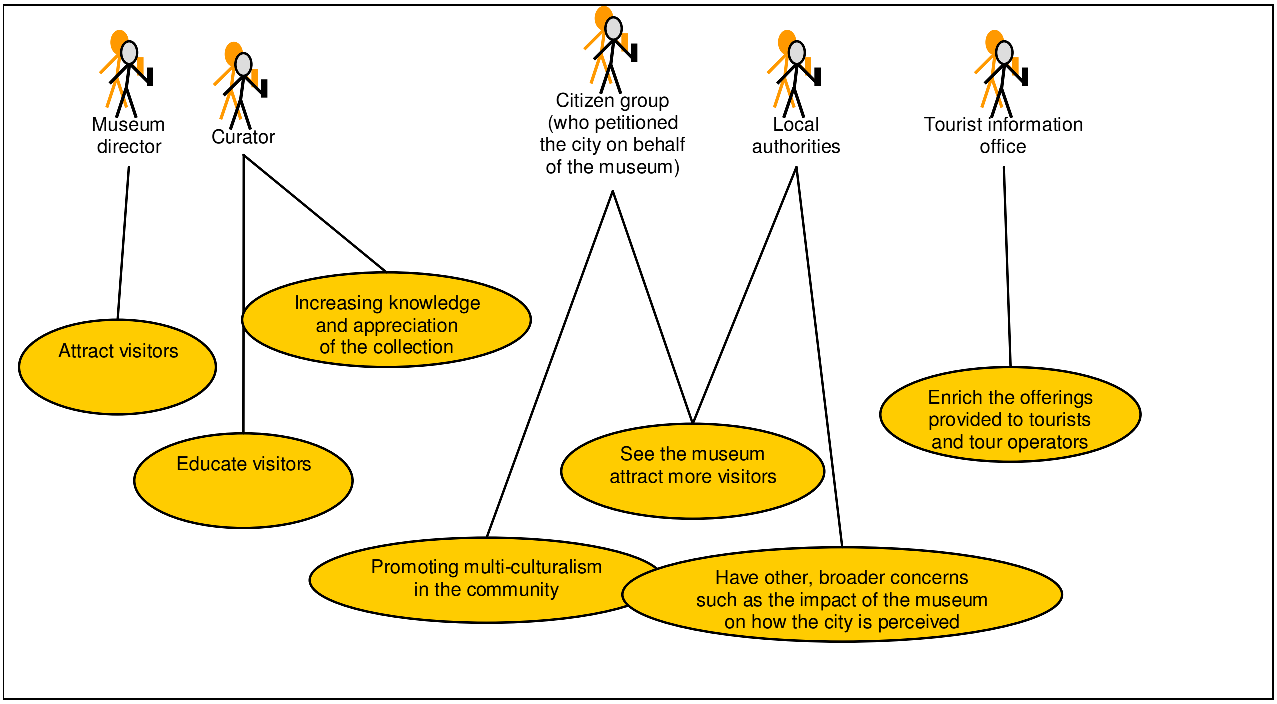
TEC-Lab and the Master in Technology-Enhanced Communication for Cultural Heritage (TEC-CH) received the task to design a general purpose website for the museum. As present no website exists and the only information available online is a QuickTime VR tour of the gallery which is located on the city of Lugano site.

The traceability study presented in this report has as goal the refinement of the first design produced by two participants of the TEC-CH Master.

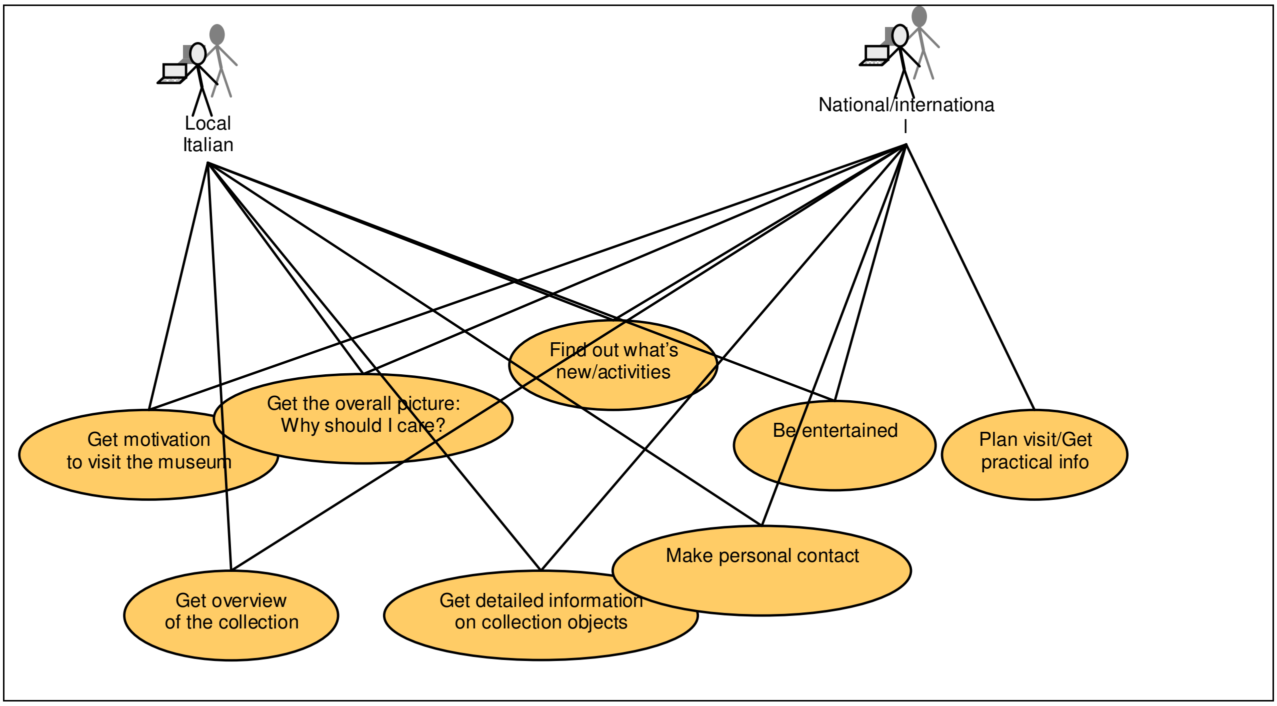
Requirements and design normalisation

Visions and assumptions

Stakeholders and goals



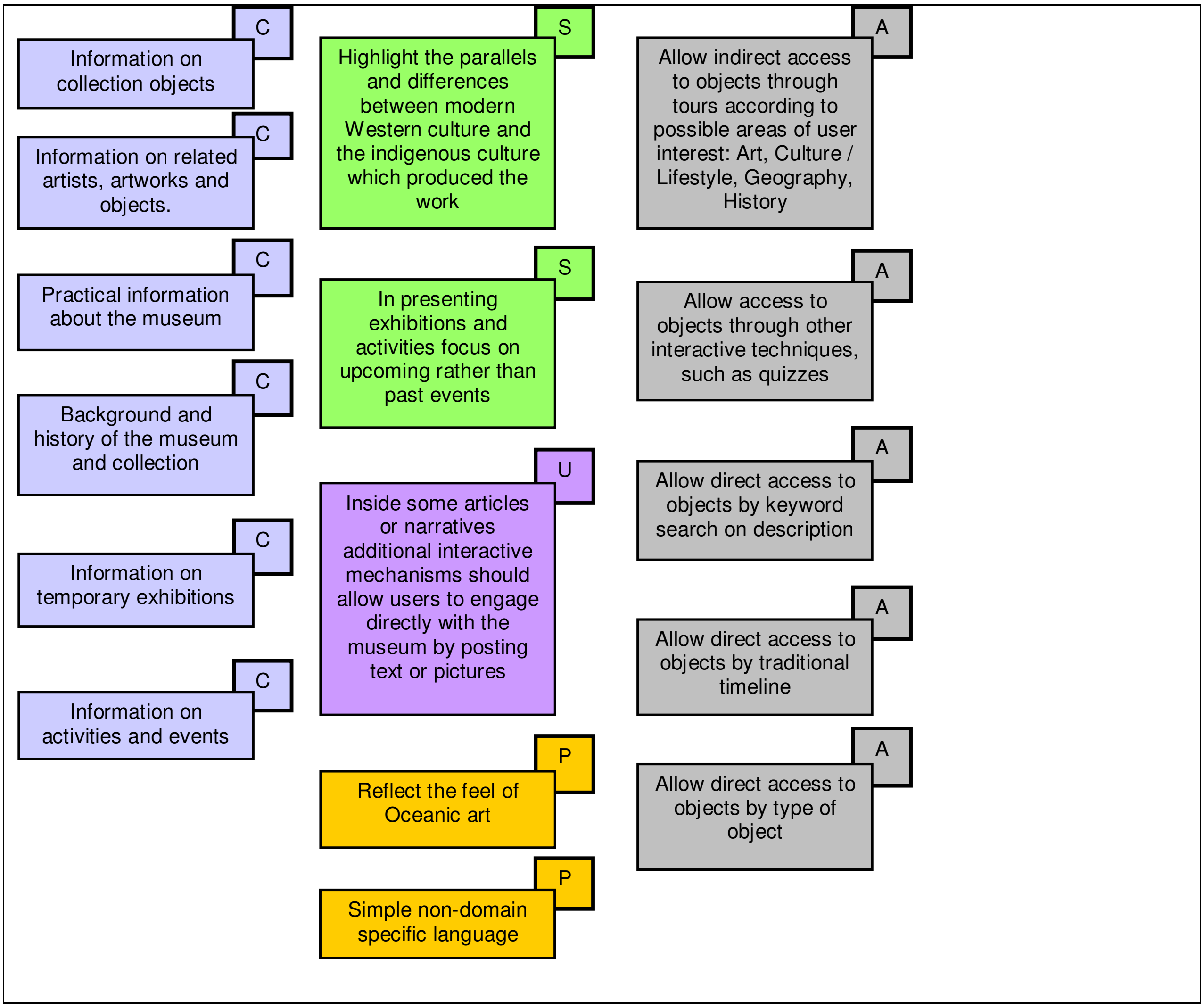
Users and motivations



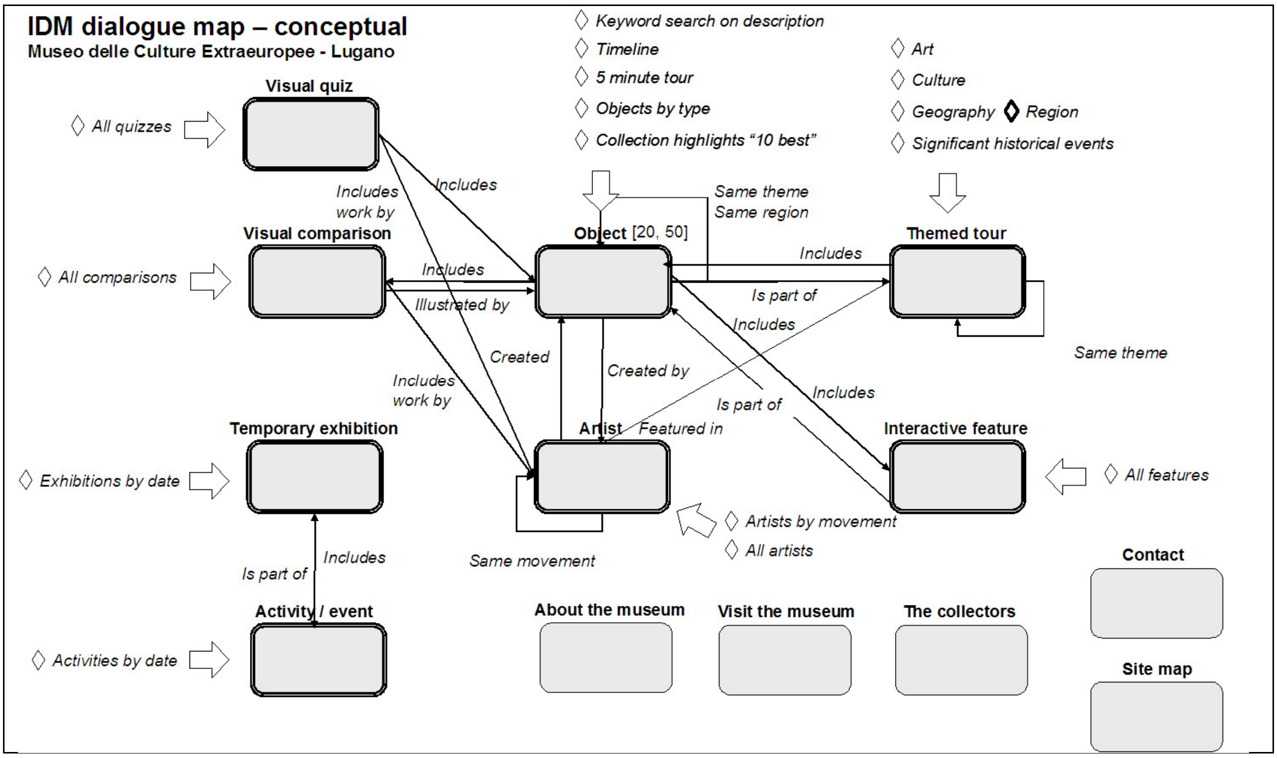
National/international non Italian speakers are:

- Swiss German tourists
- Domain experts from universities and other cultural institutions residents of Lugano

Requirements



Design model



Traceability Model 1 – Requirements Impact Model

Visions - Design

		TOPICS													RELEVANT RELATIONS																	GROUP OF TOPICS															
		Object	Themed tour	Artist	Interactive feature	Visual quiz	Visual comparison	Temporary exhibition	Activity / Event	About the museum	Visit the museum	The collectors	Contact	Site map	Visual quiz INCLUDES Object	Visual comparison ILLUSTRATED BY Object	Object INCLUDES Visual comparison	Visual comparison INCLUDES WORK BY Artist	Visual quiz INCLUDES WORK BY Artists	Object IS PART OF Themed tour	Themed tour INCLUDES Object	Object SAME THEME Object	Object SAME REGION Object	Object CREATED BY Artist	Artist CREATED Object	Artist SAME MOVEMENT Artists	Object INCLUDES Interactive feature	Interactive feature IS PART OF Object	Themed tour SAME THEME Themed tour	Temporary exhibition INCLUDES Activity / Event	Activity / Event IS PART OF Temporary exhibition	All quizzes	All comparisons	Exhibitions by date	Activities by date	Keyword search on description	Timeline	5 minute tour	Objects by type	Collection highlights "10 best"	Art	Culture	Geography << Region	Significant historical events	Artists by movement	All artists	All features
VISIONS	Stakeholders are essentially united in the desire to see the museum stay open																																														
	We have to support the re-launch of the museum								X	X	X																							X	X												
	Users will have very little motivation to visit our website or the museum				X																							X																		X
	Our target audience has little or no interest in Oceanic or Extra-European art				X	X	X								X	X		X	X													X	X					X		X	X	X	X	X	X		X

Stakeholders - Design

		TOPICS													RELEVANT RELATIONS																	GROUP OF TOPICS															
		Object	Themed tour	Artist	Interactive feature	Visual quiz	Visual comparison	Temporary exhibition	Activity / Event	About the museum	Visit the museum	The collectors	Contact	Site map	Visual quiz INCLUDES Object	Visual comparison ILLUSTRATED BY Object	Object INCLUDES Visual comparison	Visual comparison INCLUDES WORK BY Artist	Visual quiz INCLUDES WORK BY Artists	Object IS PART OF Themed tour	Themed tour INCLUDES Object	Object SAME THEME Object	Object SAME REGION Object	Object CREATED BY Artist	Artist CREATED Object	Artist SAME MOVEMENT Artists	Object INCLUDES Interactive feature	Interactive feature IS PART OF Object	Themed tour SAME THEME Themed tour	Temporary exhibition INCLUDES Activity / Event	Activity / Event IS PART OF Temporary exhibition	All quizzes	All comparisons	Exhibitions by date	Activities by date	Keyword search on description	Timeline	5 minute tour	Objects by type	Collection highlights "10 best"	Art	Culture	Geography << Region	Significant historical events	Artists by movement	All artists	All features
STAKEHOLDERS	Museum director		X		X	X			X	X	X		X						X											X	X	X		X	X	X				X						X	X
	Curator	X	X	X		X	X	X		X	X	X			X	X	X	X		X	X	X	X	X	X	X	X	X	X				X	X		X	X	X	X	X	X	X	X	X	X	X	X
	Citizen group (who petitioned the city on behalf of the museum)							X	X	X	X																							X	X												
	Local authorities							X	X	X		X	X																							X							X			X	
	Tourist information office							X	X		X		X																	X	X			X	X												

Goals - Design

		TOPICS													RELEVANT RELATIONS																	GROUP OF TOPICS															
		Object	Themed tour	Artist	Interactive feature	Visual quiz	Visual comparison	Temporary exhibition	Activity / Event	About the museum	Visit the museum	The collectors	Contact	Site map	Visual quiz INCLUDES Object	Visual comparison ILLUSTRATED BY Object	Object INCLUDES Visual comparison	Visual comparison INCLUDES WORK BY Artist	Visual quiz INCLUDES WORK BY Artists	Object IS PART OF Themed tour	Themed tour INCLUDES Object	Object SAME THEME Object	Object SAME REGION Object	Object CREATED BY Artist	Artist CREATED Object	Artist SAME MOVEMENT Artists	Object INCLUDES Interactive feature	Interactive feature IS PART OF Object	Themed tour SAME THEME Themed tour	Temporary exhibition INCLUDES Activity / Event	Activity / Event IS PART OF Temporary exhibition	All quizzes	All comparisons	Exhibitions by date	Activities by date	Keyword search on description	Timeline	5 minute tour	Objects by type	Collection highlights "10 best"	Art	Culture	Geography << Region	Significant historical events	Artists by movement	All artists	All features
GOALS	Attract visitors							X	X		X																			X	X			X	X			X		X							
	Educate visitors		X		X	X	X								X	X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X				X	X		X	X	X	X	X	X		X
	Increasing knowledge and appreciation of the collection	X	X		X	X	X								X	X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X				X	X	X	X	X	X	X	X	X		X
	Promoting multi-culturalism in the community						X		X							X	X	X												X	X		X								X	X	X	X			
	See the museum attract more visitors							X	X		X		X																					X	X												
	Have other, broader concerns such as the impact of the museum on how the city is perceived									X																																					
	Enrich the offerings provided to tourists and tour operators							X	X		X																							X	X												

Users - Design

TOPICS

		Object	Themed tour	Artist	Interactive feature	Visual quiz	Visual comparison	Temporary exhibition	Activity / Event	About the museum	Visit the museum	The collectors	Contact	Site map
USERS	Local Italian speakers	X	X	X	X	X	X	X	X	X	X	X	X	X
USERS	National/international non Italian speakers	X	X	X	X	X	X	X	X	X	X	X	X	X

RELEVANT RELATIONS

		Visual quiz INCLUDES Object	Visual comparison ILLUSTRATED BY Object	Object INCLUDES Visual comparison	Visual comparison INCLUDES WORK BY Artist	Visual quiz INCLUDES WORK BY Artists	Object IS PART OF Themed tour	Themed tour INCLUDES Object	Object SAME THEME Object	Object SAME REGION Object	Object CREATED BY Artist	Artist CREATED Object	Artist SAME MOVEMENT Artists	Object INCLUDES Interactive feature	Interactive feature IS PART OF Object	Themed tour SAME THEME Themed tour	Temporary exhibition INCLUDES Activity / Event	Activity / Event IS PART OF Temporary exhibition
USERS	Local Italian speakers	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
USERS	National/international non Italian speakers	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

GROUP OF TOPICS

		All quizzes	All comparisons	Exhibitions by date	Activities by date	Keyword search on description	Timeline	5 minute tour	Objects by type	Collection highlights "10 best"	Art	Culture	Geography << Region	Significant historical events	Artists by movement	All artists	All features
USERS	Local Italian speakers	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
USERS	National/international non Italian speakers	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Motivations - Design

TOPICS

		Object	Themed tour	Artist	Interactive feature	Visual quiz	Visual comparison	Temporary exhibition	Activity / Event	About the museum	Visit the museum	The collectors	Contact	Site map
MOTIVATIONS	Get motivation to visit the museum		X					X	X					
MOTIVATIONS	Get the overall picture: Why should I care?		X					X	X					
MOTIVATIONS	Get overview of the collection							X						
MOTIVATIONS	Find out what's new/activities							X	X					X
MOTIVATIONS	Get detailed information on collection object	X						X				X		
MOTIVATIONS	Make personal contact												X	
MOTIVATIONS	Be entertained		X		X	X								
MOTIVATIONS	Plan visit/Get practical info							X			X		X	

RELEVANT RELATIONS

		Visual quiz INCLUDES Object	Visual comparison ILLUSTRATED BY Object	Object INCLUDES Visual comparison	Visual comparison INCLUDES WORK BY Artist	Visual quiz INCLUDES WORK BY Artists	Object IS PART OF Themed tour	Themed tour INCLUDES Object	Object SAME THEME Object	Object SAME REGION Object	Object CREATED BY Artist	Artist CREATED Object	Artist SAME MOVEMENT Artists	Object INCLUDES Interactive feature	Interactive feature IS PART OF Object	Themed tour SAME THEME Themed tour	Temporary exhibition INCLUDES Activity / Event	Activity / Event IS PART OF Temporary exhibition
MOTIVATIONS	Get motivation to visit the museum																	
MOTIVATIONS	Get the overall picture: Why should I care?																	
MOTIVATIONS	Get overview of the collection																	
MOTIVATIONS	Find out what's new/activities																X	X
MOTIVATIONS	Get detailed information on collection object								X	X	X	X	X					
MOTIVATIONS	Make personal contact																	
MOTIVATIONS	Be entertained																	
MOTIVATIONS	Plan visit/Get practical info																	

GROUP OF TOPICS

		All quizzes	All comparisons	Exhibitions by date	Activities by date	Keyword search on description	Timeline	5 minute tour	Objects by type	Collection highlights "10 best"	Art	Culture	Geography << Region	Significant historical events	Artists by movement	All artists	All features
MOTIVATIONS	Get motivation to visit the museum							X		X	X	X	X	X			
MOTIVATIONS	Get the overall picture: Why should I care?							X		X	X	X	X	X			
MOTIVATIONS	Get overview of the collection						X	X		X							
MOTIVATIONS	Find out what's new/activities			X	X					X							
MOTIVATIONS	Get detailed information on collection object						X		X						X	X	
MOTIVATIONS	Make personal contact																
MOTIVATIONS	Be entertained							X		X							X
MOTIVATIONS	Plan visit/Get practical info			X	X												

Requirements - Design

TOPICS

			Object	Themed tour	Artist	Interactive feature	Visual quiz	Visual comparison	Temporary exhibition	Activity / Event	About the museum	Visit the museum	The collectors	Contact	Site map
REQUIREMENTS	content	Information on collection objects	X	X					X						
REQUIREMENTS	content	Information on related artists, artworks and objects			X			X							
REQUIREMENTS	content	Practical information about the museum										X			
REQUIREMENTS	content	Background and history of the museum and collection									X		X		
REQUIREMENTS	content	Information on temporary exhibitions							X						
REQUIREMENTS	content	Information on activities and events								X					
REQUIREMENTS	structure of content	Highlight the parallels and differences between modern Western culture and the indigenous culture which produced the work						X							
REQUIREMENTS	structure of content	In presenting exhibitions and activities focus on upcoming rather than past events													
REQUIREMENTS	user interaction	Inside some articles or narratives additional interactive mechanisms should allow users to engage directly with the museum by posting text or pictures				X									
REQUIREMENTS	presentation	Reflect the feel of Oceanic art		X											
REQUIREMENTS	presentation	Simple non-domain specific language													
REQUIREMENTS	access paths to content	Allow indirect access to objects through tours according to possible areas of user interest: Art, Culture / Lifestyle, Geography, History		X											
REQUIREMENTS	access paths to content	Allow access to objects through other interactive techniques, such as quizzes					X								
REQUIREMENTS	access paths to content	Allow direct access to objects by keyword search on description													
REQUIREMENTS	access paths to content	Allow direct access to objects by traditional timeline													
REQUIREMENTS	access paths to content	Allow direct access to objects by type of object													

RELEVANT RELATIONS

			Visual quiz INCLUDES Object	Visual comparison ILLUSTRATED BY Object	Object INCLUDES Visual comparison	Visual comparison INCLUDES WORK BY Artist	Visual quiz INCLUDES WORK BY Artists	Object IS PART OF Themed tour	Themed tour INCLUDES Object	Object SAME THEME Object	Object SAME REGION Object	Object CREATED BY Artist	Artist CREATED Object	Artist SAME MOVEMENT Artists	Object INCLUDES Interactive feature	Interactive feature IS PART OF Object	Themed tour SAME THEME Themed tour	Temporary exhibition INCLUDES Activity / Event	Activity / Event IS PART OF Temporary exhibition
REQUIREMENTS	content	Information on collection objects																	
REQUIREMENTS	content	Information on related artists, artworks and objects			X	X	X			X		X		X					
REQUIREMENTS	content	Practical information about the museum																	
REQUIREMENTS	content	Background and history of the museum and collection																	
REQUIREMENTS	content	Information on temporary exhibitions																	
REQUIREMENTS	content	Information on activities and events																	
REQUIREMENTS	structure of content	Highlight the parallels and differences between modern Western culture and the indigenous culture which produced the work		X	X	X		X	X										
REQUIREMENTS	structure of content	In presenting exhibitions and activities focus on upcoming rather than past events																	
REQUIREMENTS	user interaction	Inside some articles or narratives additional interactive mechanisms should allow users to engage directly with the museum by posting text or pictures																	
REQUIREMENTS	presentation	Reflect the feel of Oceanic art																	
REQUIREMENTS	presentation	Simple non-domain specific language																	
REQUIREMENTS	access paths to content	Allow indirect access to objects through tours according to possible areas of user interest: Art, Culture / Lifestyle, Geography, History																	
REQUIREMENTS	access paths to content	Allow access to objects through other interactive techniques, such as quizzes	X													X			
REQUIREMENTS	access paths to content	Allow direct access to objects by keyword search on description																	
REQUIREMENTS	access paths to content	Allow direct access to objects by traditional timeline																	
REQUIREMENTS	access paths to content	Allow direct access to objects by type of object																	

GROUP OF TOPICS

			All quizzes	All comparisons	Exhibitions by date	Activities by date	Keyword search on description	Timeline	5 minute tour	Objects by type	Collection highlights "10 best"	Art	Culture	Geography << Region	Significant historical events	Artists by movement	All artists	All features
REQUIREMENTS	content	Information on collection objects																
REQUIREMENTS	content	Information on related artists, artworks and objects																
REQUIREMENTS	content	Practical information about the museum																
REQUIREMENTS	content	Background and history of the museum and collection																
REQUIREMENTS	content	Information on temporary exhibitions																
REQUIREMENTS	content	Information on activities and events																
REQUIREMENTS	structure of content	Highlight the parallels and differences between modern Western culture and the indigenous culture which produced the work		X								X	X	X	X			
REQUIREMENTS	structure of content	In presenting exhibitions and activities focus on upcoming rather than past events			X													
REQUIREMENTS	user interaction	Inside some articles or narratives additional interactive mechanisms should allow users to engage directly with the museum by posting text or pictures																X
REQUIREMENTS	presentation	Reflect the feel of Oceanic art									X			X		X		
REQUIREMENTS	presentation	Simple non-domain specific language					X											
REQUIREMENTS	access paths to content	Allow indirect access to objects through tours according to possible areas of user interest: Art, Culture / Lifestyle, Geography, History										X	X	X	X			
REQUIREMENTS	access paths to content	Allow access to objects through other interactive techniques, such as quizzes	X															
REQUIREMENTS	access paths to content	Allow direct access to objects by keyword search on description					X											
REQUIREMENTS	access paths to content	Allow direct access to objects by traditional timeline						X										
REQUIREMENTS	access paths to content	Allow direct access to objects by type of object								X								

Requirements (content) - Design

TOPICS

		Object	Themed tour	Artist	Interactive feature	Visual quiz	Visual comparison	Temporary exhibition	Activity / Event	About the museum	Visit the museum	The collectors	Contact	Site map
content	Information on collection objects	X	X					X						
	Information on related artists, artworks and objects			X			X							
	Practical information about the museum										X			
	Background and history of the museum and collection									X		X		
	Information on temporary exhibitions							X						
	Information on activities and events								X					

RELEVANT RELATIONS

		Visual quiz INCLUDES Object	Visual comparison ILLUSTRATED BY Object	Object INCLUDES Visual comparison	Visual comparison INCLUDES WORK BY Artist	Visual quiz INCLUDES WORK BY Artists	Object IS PART OF Themed tour	Themed tour INCLUDES Object	Object SAME THEME Object	Object SAME REGION Object	Object CREATED BY Artist	Artist CREATED Object	Artist SAME MOVEMENT Artists	Object INCLUDES Interactive feature	Interactive feature IS PART OF Object	Themed tour SAME THEME Themed tour	Temporary exhibition INCLUDES Activity / Event	Activity / Event IS PART OF Temporary exhibition
content	Information on collection objects																	
	Information on related artists, artworks and objects			X	X	X			X		X		X					
	Practical information about the museum																	
	Background and history of the museum and collection																	
	Information on temporary exhibitions																	
	Information on activities and events																	

GROUP OF TOPICS

		All quizzes	All comparisons	Exhibitions by date	Activities by date	Keyword search on description	Timeline	5 minute tour	Objects by type	Collection highlights "10 best"	Art	Culture	Geography << Region	Significant historical events	Artists by movement	All artists	All features
content	Information on collection objects																
	Information on related artists, artworks and objects																
	Practical information about the museum																
	Background and history of the museum and collection																
	Information on temporary exhibitions																
	Information on activities and events																

Requirements (structure of content) - Design

TOPICS

		Object	Themed tour	Artist	Interactive feature	Visual quiz	Visual comparison	Temporary exhibition	Activity / Event	About the museum	Visit the museum	The collectors	Contact	Site map
structure of content	Highlight the parallels and differences between modern Western culture and the indigenous culture which produced the work						X							
	In presenting exhibitions and activities focus on upcoming rather than past events													

RELEVANT RELATIONS

		Visual quiz INCLUDES Object	Visual comparison ILLUSTRATED BY Object	Object INCLUDES Visual comparison	Visual comparison INCLUDES WORK BY Artist	Visual quiz INCLUDES WORK BY Artists	Object IS PART OF Themed tour	Themed tour INCLUDES Object	Object SAME THEME Object	Object SAME REGION Object	Object CREATED BY Artist	Artist CREATED Object	Artist SAME MOVEMENT Artists	Object INCLUDES Interactive feature	Interactive feature IS PART OF Object	Themed tour SAME THEME Themed tour	Temporary exhibition INCLUDES Activity / Event	Activity / Event IS PART OF Temporary exhibition
structure of content	Highlight the parallels and differences between modern Western culture and the indigenous culture which produced the work		X	X	X		X	X										
	In presenting exhibitions and activities focus on upcoming rather than past events																	

GROUP OF TOPICS

		All quizzes	All comparisons	Exhibitions by date	Activities by date	Keyword search on description	Timeline	5 minute tour	Objects by type	Collection highlights "10 best"	Art	Culture	Geography << Region	Significant historical events	Artists by movement	All artists	All features
structure of content	Highlight the parallels and differences between modern Western culture and the indigenous culture which produced the work		X								X	X	X	X			
	In presenting exhibitions and activities focus on upcoming rather than past events			X													

Requirements (user operation) - Design

TOPICS

		Object	Themed tour	Artist	Interactive feature	Visual quiz	Visual comparison	Temporary exhibition	Activity / Event	About the museum	Visit the museum	The collectors	Contact	Site map
user interaction	Inside some articles or narratives additional interactive mechanisms should allow users to engage directly with the museum by posting text or pictures				X									

RELEVANT RELATIONS

		Visual quiz INCLUDES Object	Visual comparison ILLUSTRATED BY Object	Object INCLUDES Visual comparison	Visual comparison INCLUDES WORK BY Artist	Visual quiz INCLUDES WORK BY Artists	Object IS PART OF Themed tour	Themed tour INCLUDES Object	Object SAME THEME Object	Object SAME REGION Object	Object CREATED BY Artist	Artist CREATED Object	Artist SAME MOVEMENT Artists	Object INCLUDES Interactive feature	Interactive feature IS PART OF Object	Themed tour SAME THEME Themed tour	Temporary exhibition INCLUDES Activity / Event	Activity / Event IS PART OF Temporary exhibition
user interaction	Inside some articles or narratives additional interactive mechanisms should allow users to engage directly with the museum by posting text or pictures																	

GROUP OF TOPICS

		All quizzes	All comparisons	Exhibitions by date	Activities by date	Keyword search on description	Timeline	5 minute tour	Objects by type	Collection highlights "10 best"	Art	Culture	Geography << Region	Significant historical events	Artists by movement	All artists	All features
user interaction	Inside some articles or narratives additional interactive mechanisms should allow users to engage directly with the museum by posting text or pictures																X

Requirements (presentation) - Design

TOPICS

		Object	Themed tour	Artist	Interactive feature	Visual quiz	Visual comparison	Temporary exhibition	Activity / Event	About the museum	Visit the museum	The collectors	Contact	Site map
presentation	Reflect the feel of Oceanic art		X											
	Simple non-domain specific language													

RELEVANT RELATIONS

		Visual quiz INCLUDES Object	Visual comparison ILLUSTRATED BY Object	Object INCLUDES Visual comparison	Visual comparison INCLUDES WORK BY Artist	Visual quiz INCLUDES WORK BY Artists	Object IS PART OF Themed tour	Themed tour INCLUDES Object	Object SAME THEME Object	Object SAME REGION Object	Object CREATED BY Artist	Artist CREATED Object	Artist SAME MOVEMENT Artists	Object INCLUDES Interactive feature	Interactive feature IS PART OF Object	Themed tour SAME THEME Themed tour	Temporary exhibition INCLUDES Activity / Event	Activity / Event IS PART OF Temporary exhibition
presentation	Reflect the feel of Oceanic art																	
	Simple non-domain specific language																	

GROUP OF TOPICS

		All quizzes	All comparisons	Exhibitions by date	Activities by date	Keyword search on description	Timeline	5 minute tour	Objects by type	Collection highlights "10 best"	Art	Culture	Geography << Region	Significant historical events	Artists by movement	All artists	All features
presentation	Reflect the feel of Oceanic art									X			X		X		
	Simple non-domain specific language					X											

Requirements (access) - Design

TOPICS

		Object	Themed tour	Artist	Interactive feature	Visual quiz	Visual comparison	Temporary exhibition	Activity / Event	About the museum	Visit the museum	The collectors	Contact	Site map
access paths to content	Allow indirect access to objects through tours according to possible areas of user interest: Art, Culture / Lifestyle, Geography, History		X											
	Allow access to objects through other interactive techniques, such as quizzes					X								
	Allow direct access to objects by keyword search on description													
	Allow direct access to objects by traditional timeline													
	Allow direct access to objects by type of object													

RELEVANT RELATIONS

		Visual quiz INCLUDES Object	Visual comparison ILLUSTRATED BY Object	Object INCLUDES Visual comparison	Visual comparison INCLUDES WORK BY Artist	Visual quiz INCLUDES WORK BY Artists	Object IS PART OF Themed tour	Themed tour INCLUDES Object	Object SAME THEME Object	Object SAME REGION Object	Object CREATED BY Artist	Artist CREATED Object	Artist SAME MOVEMENT Artists	Object INCLUDES Interactive feature	Interactive feature IS PART OF Object	Themed tour SAME THEME Themed tour	Temporary exhibition INCLUDES Activity / Event	Activity / Event IS PART OF Temporary exhibition
access paths to content	Allow indirect access to objects through tours according to possible areas of user interest: Art, Culture / Lifestyle, Geography, History																	
	Allow access to objects through other interactive techniques, such as quizzes	X													X			
	Allow direct access to objects by keyword search on description																	
	Allow direct access to objects by traditional timeline																	
	Allow direct access to objects by type of object																	

GROUP OF TOPICS

		All quizzes	All comparisons	Exhibitions by date	Activities by date	Keyword search on description	Timeline	5 minute tour	Objects by type	Collection highlights "10 best"	Art	Culture	Geography << Region	Significant historical events	Artists by movement	All artists	All features
access paths to content	Allow indirect access to objects through tours according to possible areas of user interest: Art, Culture / Lifestyle, Geography, History										X	X	X	X			
	Allow access to objects through other interactive techniques, such as quizzes	X															
	Allow direct access to objects by keyword search on description					X											
	Allow direct access to objects by traditional timeline						X										
	Allow direct access to objects by type of object								X								

"Negative" Requirements

Negative content requirement: 3D animated tour. The current page of the museum of the City of Lugano website contains a 3D animated tour which should not be included in the new website. In its current form it is not an effective tool to encourage visitors to come to the museum. While some form of 3D tour of the building may be useful, it is not essential to the promotional or educational goals of the current project.

Traceability Model 2 – Design Motivations Model

Topics - Motivations

		Visions	Stakeholders	Goals	Users	Motivations	Requirements	Specific understanding of the domain	Designer expertise	Technology constraints	"Graphic" constraints	Budget constraints	Laws obligations
TOPICS	Object						X						
	Themed tour			X			X						
	Artist						X						
	Interactive feature			X					X				
	Visual quiz								X				
	Visual comparison			X				X					
	Temporary exhibition		X	X			X						
	Activity / Event		X	X			X						
	About the museum		X						X				
	Visit the museum				X	X							
	The collectors		X										
	Contact				X	X			X				
	Site map								X				

Relevant relations - Motivations

		Visions	Stakeholders	Goals	Users	Motivations	Requirements	Specific understanding of the domain	Designer expertise	Technology constraints	"Graphic" constraints	Budget constraints	Laws obligations
RELEVANT RELATIONS	Visual quiz INCLUDES Object							X					
	Visual comparison ILLUSTRATED BY Object							X					
	Object INCLUDES Visual comparison							X					
	Visual comparison INCLUDES WORK BY Artist							X					
	Visual quiz INCLUDES WORK BY Artists							X					
	Object IS PART OF Themed tour							X					
	Themed tour INCLUDES Object							X					
	Object SAME THEME Object							X					
	Object SAME REGION Object							X					
	Object CREATED BY Artist							X					
	Artist CREATED Object							X					
	Artist SAME MOVEMENT Artists							X					
	Object INCLUDES Interactive feature								X				
	Interactive feature IS PART OF Object								X				
	Themed tour SAME THEME Themed tour							X					
	Temporary exhibition INCLUDES Activity / Event		X						X				
	Activity / Event IS PART OF Temporary exhibition		X						X				

Group of topics - Motivations

		Visions	Stakeholders	Goals	Users	Motivations	Requirements	Specific understanding of the domain	Designer expertise	Technology constraints	"Graphic" constraints	Budget constraints	Laws obligations
GROUP OF TOPICS	All quizzes								X				
	All comparisons								X				
	Exhibitions by date		X						X				
	Activities by date		X						X				
	Keyword search on description		X										
	Timeline							X					
	5 minute tour								X				
	Objects by type		X										
	Collection highlights "10 best"								X				
	Art							X					
	Culture							X					
	Geography << Region							X					
	Significant historical events							X					
	Artists by movement		X										
	All artists								X				
	All features								X				

"Negative" Design

Topic: "Kind of object"
Motivation: The dialogue risked to become very complex for a non-expert users; we preferred to add a short introduction to the group of topic "Object by type".

Conclusions

The main results of this analysis can be summarised as follows:

- the goal "enrich the offerings provided to tourists and tour operator" is poorly supported by the design
- the motivation "make personal contact" of the user is not supported by the design; the goal is now considered only in the contact information, but this element is insufficient to answer to this (possible) user need; this aspect could be emphasized as means to fulfil a stakeholder's goal
- the big quantity of relevant relations risks to overemphasize the navigation possibilities on the site and to disorient the user; in fact, the majority of these relationships are designer choices and they do not come from a precise goal; a reduction of the relations should be discussed

Traceability Report 5
Munch un Berlin Exhibition – version 2

Date: February, 2005
Target: Project Manager
Goals: Evaluate the impact of changing requirements

Executive summary

This report concerns the development of the web site for the “Munch und Berlin” exhibition at the Berlin State Museum in Germany (“Staatliche Museen zu Berlin”). The exhibition hosted Munch’s prints and drawings and took place from April the 12th to July the 13th 2004. The website for the exhibition has been developed as part of the HELP project (partially funded by the European Commission) and included an innovative aspect (which is not the central theme of this paper): the development of a design technique enabling overcoming most of the usability problems experienced by visually impaired users using the web.

The requirements analysis activity has been performed partially during the project and partially after the publication of the website. During the design process, the analysis has taken into account the curator of the exhibition as main stakeholder, eliciting its visions about the application and the strategic goals of the site. At the end of the project, a traceability analysis has been performed to link the requirements material with the design solutions and to point out indications for improving the application.

A previous traceability analysis were conducted in April 2004. This new traceability phase is needed by new and refined project goals. Even if the exhibition is now finished, the project team is keeping alive this web site for educational purposes.

"Munch und Berlin" exhibition web site

This analysis is concerned with the development of a web site for the "Munch un Berlin exhibition at the Staatliche Museen zu Berlin. The exhibition has taken place from April the 12th to july the 13th 2003 and was curated by Dr. Sigrid Achenbach. The website has been developped as part of Help project (partially funded by the European Commission). It was the result of a joint effort by Staatliche Museen zu Berlin - Preußischer Kulturbesitz (Germany), HOC-LAB of Politecnico di Milano (Italy) and TEC-LAB of University of Italian Switzerland (Lugano, Switzerland).

The project has developed an innovative technology that allows overcoming most of the limitations of the W3C accessibility guidelines for visually impaired users. The technology is based upon a linguistic approach to the web called WED (WEb as Dialogue), developed by TEC-Lab and HOC. The WED approach considers the interaction of a user with a web site as a dialogue. Based on this assumption, WED is an innovative design methodology which makes the designer think to the web site not just as an informative tool but as a partner (i.e. the teacher) of a didactic dialogue with its user (the pupil). HELP exploited the WED approach for designing a cultural web site optimized for visually impaired people, where the interaction is more natural, like in an oral dialogue. The success of the WED approach is showed in the challenging HELP case study: the Munch's Exhibition web site in Berlin.

The design of the website www.munchundberlin.org represents the first practical result of the WED approach. It complies with almost all the accessibility rules of W3C; apart from being accessible, the web site presents some features that make it optimized for visually impaired users. An example is the page schema, a short summary (orally read but invisible in the page) of the basic sections of the page that the screen reader reads before reading any other content. The page schema enhances accessibility under two aspects: it gives the user the possibility to decide which section s/he's interested in and it helps memorizing the page structure, being based on consistent templates which facilitate the user navigation and orientation.

The positive feedback received by visually impaired people are encouraging. Furthermore, the methodology turned out to be useful for designing web sites for sighted people too, improving usability and user satisfaction. The results obtained are offering innovative cues and ideas for new outlooks. Future work has the overall goal of making the man-machine dialogues (such as those of a user with the web) closer to human-human dialogues (such as those of a user talking with a expert), and their effectiveness.

Requirements analysis

The requirements analysis activity has been performed partially during the project and partially after the publication of the website. During the design process, the analysis has taken into account the curator of the exhibition as main stakeholder, eliciting its visions about the application and the strategic goals of the site. At the end of the project, a traceability analysis has been performed to link the requirements material with the design solutions and to point out indications for improving the application.

Stakeholder's visions

During the meetings with the museum curator (one of the main stakeholders) the following goals for the Munch und Berlin website emerged:

1. design a website which might work also as a fixed information kiosk in the museum;
2. make the website usable by visually-impaired users;
3. promote knowledge and awareness about a temporary exhibition being hosted at the Museum (Munch's prints and drawings).

The first goal aims at offering to the user a multi-channel interaction, i.e. a similar interactive experience on different channels. On the website (at home) and on the info kiosk (in the museum) different content and services will be offered to the user, but the same look & feel should be kept. The second goal has to do with a growing concern: accessibility. Visually-impaired users can surf the web through special software, called "screen readers". To enable visually-impaired users to use satisfactorily a website, designers should optimize their site design to be read by screen readers in an effective way. The third goal represents the overall mission of the website, which is the reason why the application has been designed.

If we carefully consider these goals, they seem quite general and almost stakeholder-independent: they are objectives which may be easily shared and agreed upon by many museums curators. Going deeper in the requirements analysis (after some design iterations), and trying to understand how to shape the presentation of Munch's collection on the website, we discovered that the curator was putting particular emphasis on the historical and social surrounding of Munch's life. He was strongly committed to make the audience understand the historical period in which Munch lived and worked to his drawing. The curator insisted to provide accurate content on Munch's different stages of life (Childhood and youth in Norway, The beginning of his artistic career, the Berlin period, Success and crisis, and so on) and on the corresponding historical events happening in those years (ca. 1890) in Europe (beginning of Imperialism, political movements in Norway, ecc.). We realized that the amount of content about these themes was becoming considerable, and actually enriched a simple presentation of Munch's drawings and prints.

Deriving goals from stakeholder's visions

Once understood, a vision may bring to formulate a set of corresponding goals. A possible line of inquiry for make goals surface from a vision is the following: How does the application embody this vision? Considering the curator's vision (works of art need to be framed within their historical background to be properly understood and appreciated), a number of new goals for the website emerged, which were not considered before:

- a1. Encourage understanding of Munch's works by leading themes, bound to the historical and social context of that period.
- a2. Create awareness on the artistic movements which influenced Munch's style.

a3. Create awareness on the social and political background characterizing the periods in which Munch worked at his prints.

User Motivations

Communication-intensive websites should be targeted to specific users, who may be driven by different factors to visit the application. User motivations are general reasons that bring a particular user type to make use of the application. These motivations are to be taken into account in the design to find solutions that are meaningful for the user we are addressing to. In our case, user motivations have been elicited with the museum curator by envisioning some user scenarios or "success stories" for the website. Here we describe 3 salient scenarios that emerged:

S1. A German man, 40 years old, wants to visit the exhibition next week. He has a good education about visual arts, but he is not very experienced with Munch's artworks. This potential visitor accesses the site to be prepared about what he will see at the exhibition. He browses around looking for information that allows him understanding the exhibition itself and for practical info.

S2. An Italian, 35-years old woman has a passion for visual arts, but she doesn't know Munch's works very well. She will never go at the exhibition but she is curious about the information in the site. She would like to study Munch more in depth and see what's important and interesting in this collection.

S3. A visual-impaired user access the site to enjoy Munch's artworks. He looks for interesting paintings and for information that could help him understand the beauty of the artworks. The user wants not only be able to physically access the content but also to have a nice experience on the site.

These "stories about use" are high-level scenarios, each one highlighting a specific user motivation. Namely, we have elicited three corresponding main motivations behind these scenarios:

M1. Be prepared for visiting the exhibition: the user wants to arrive at the Museum knowing what he/she will see and being able to understand the artworks exhibited.

M2. Study Munch and his art: the user wants to enrich his/her knowledge about Munch and about his paintings and prints.

M3. Appreciate the artworks in the exhibition: the user wants to be able to enjoy and appreciate Munch's art through the website.

Deriving Goals from User Motivations

Since user motivations describe the reasons why a user should use the application, it is possible to derive a proper set of user goals from this knowledge. In particular, from motivation **M1** we understand that a potential visitor may have the following goals:

- See what is worth visiting in the exhibition, the best artworks exhibited and the "must-see" paintings;
- gather basic information about the set of works exhibited in its whole and its artistic importance;
- know the basics about Munch and his historical context;

From the motivation **M2**, the following goals may be specified for a "curious" non-visitor:

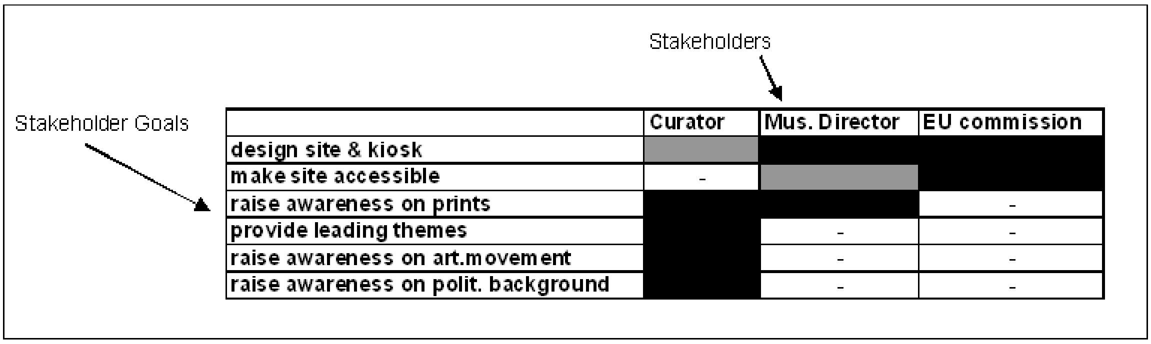
- finding historical information about Munch, his life, the encounters, his influences, etc.;

- finding detailed information about Munch's work and art, his style and the kind of artworks he did;
- finding information about the techniques used in the paintings;

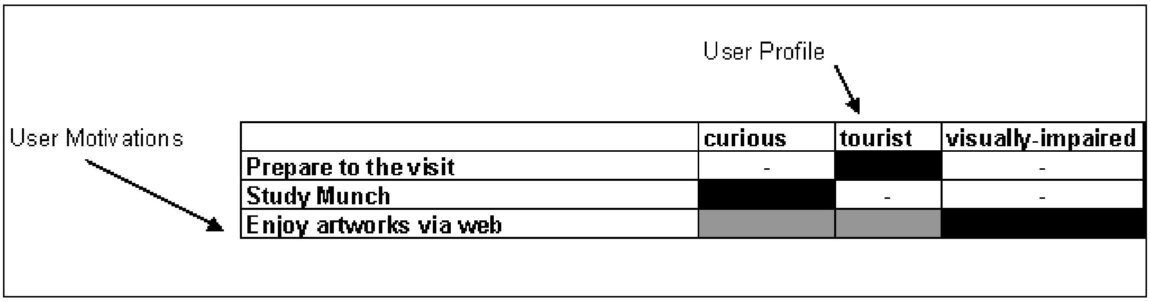
From the motivation **M3**, we can detail the following goals for a visually-impaired user:

- efficiently accessing the exhibition's topics, understanding the site structure and the browsing capabilities on each page;
- understanding Munch's paintings in the exhibition and what they represent;
- finding information about Munch, his life, and his style.

Requirements Matrices



Stakeholder-goal matrix



User profiles and their motivations

User Motivations

Stakeholder Goals

	Prepare to the visit	Study Munch	Enjoy artworks via web
design site & kiosk	promote kiosk services on the site	-	-
make site accessible	-	provide 100% text equivalent content	enable VII understand the topics and techniques
raise awareness on Munch's prints	educate to appreciate the value of Munch's prints wrt his paintings	promote the unicitiy of the exhibition	provide high-resolution and large pictures of the prints and drawings
provide leading themes	enable to create personalized thematic	organize prints by theme	-
raise awareness on art.movement	enable gather elements to interpret munch's style	understand who influences Munch's style	-
raise awareness on polit. backgroun	enable gather elements to interpret munch's themes	understand political events which influenced Munch's life	-

Intersecting stakeholder goals with user motivations

Requirements

Content

promote kiosk services on the website
provide elements to appreciate the value of Munch's prints wrt painting
enable gather elements to interpret munch's style
enable gather elements to interpret munch's themes
provide 100% text equivalent content
promote the unicitiy of the exhibition
provide elements to understand who influences Munch's style
understand political events which influenced Munch's life
enable VII understand the topics and techniques
provide high-resolution and large pictures of the prints and drawings
provide historical context
provide information about the different techniques used
provide information about the various artistic movements

Structure

highlight must see
provide collection overview

Navigation

enable to create personalized path by theme
relate prints to the technique
relate prints to artistic movement

Requirements set

Requirements Impact Model (RIM)

DESIGN ELEMENTS	CONTENTS	Prints	Techniques	Periods of life	Artistic movements	Artists	Munch	The museum	The exhibition	Contacts	Credits	Listen to this Website	RELATIONSHIPS BETWEEN CONTENTS	From a print to the technique it was made with	From a technique to the prints made with that technique	From a print to the period of life during which the print was made	From a period of life to the artistics movements actives in Germany during that period	From an artistic movement to the artists that represent the movement	From an artist to the artistic movement he belonged to	ACCESS PATHS TO CONTENTS	A list of all the prints	A group of masterpieces	Some thematic tour of prints grouped by theme	A list of Munch's periods of life	A list of all the techniques
VISIONS																									
Frame works of art within historical background				X	X	X										X	X	X						X	
MOTIVATIONS																									
Be prepared for visiting the exhibition		X	X				X	X	X	X				X	X						X	X	X		X
Study Munch and his art		X	X	X	X	X	X							X	X	X	X	X	X		X	X	X	X	X
Appreciate the artworks in the exhibition		X	X									X													
GOALS																									
Design site & kiosk																									
Make site accessible		X										X													
Raise awareness on Munch's prints		X						X	X													X	X		
Provide leading themes				X	X	X										X	X	X						X	
Raise awareness on art. movement					X												X	X							
Raise awareness on polit. background				X	X	X										X	X	X						X	
Understanding what's interesting in the exhibition																						X	X		
Finding information about the paintings in the exhibition		X	X											X	X										
Finding information about Munch's life				X			X																	X	
Finding historical information about Munch				X	X	X	X									X	X	X	X					X	
Finding detailed information about Munch's work and art		X	X											X	X						X	X	X		X
Understanding the relevance of Munch in the history of art					X	X											X	X	X						
Finding information about the techniques used in the paintings			X											X											X
Efficiently accessing the exhibition's topics		X																				X	X		
Understanding the site structure and the browsing capabilities												X													

As we have discussed, the curator's vision was to Frame Munch's works within their historical background. This general vision has been taken into account in the application on one hand by providing descriptions of the periods of Munch's life, of artistic movements and of the corresponding artists, and, on the other, by supporting the structuring of this content through a proper navigational architecture.

The goal to have an application also running on a kiosk inside the Museum was not taken into account due to the nature of content provided: the Museum produced content which is formally correct but not at all proactive or engaging for a scenario in which the user is standing in front of a fixed information point. The content provided by the museum had no storytelling, it supported very poor interactivity, and was too much linear. As a consequence, due to budgets constraints, new content for the kiosk version has not been produced.

The need to deliver an acoustic version of the site for visually-impaired users had a general impact on the design (which must be consistently structured for facilitating the listening experience of a blind user) and on the implementation technique. A specific work on providing equivalent acoustic content for all the content pieces of the site has not been done for resource constraints. The only concrete elements visible in the design is the "promotional" section "Listen to this website" and the fact that the print's big images have been separated from the rest of the print's navigation; this was meant to provide an acoustic description of the print to blind users.

Design Motivations Model (DMM)

CHOICES SOURCE	Visions	Motivations	Goals	Specific understanding of the domain	Designer expertise	Technology constraints	"Graphic" constraints	Budget constraints	Laws obbligations
CONTENTS									
Prints	X		X						
Techniques		X	X						
Periods of life		X	X						
Artistic movements	X		X						
Artists	X		X						
Munch	X		X						
The museum		X							
The exhibition		X	X						
Contacts					X				
Credits					X				
Listen to this Website			X						
RELATIONSHIPS BETWEEN CONTENTS									
From a print to the technique it was made with					X				
From a technique to the prints made with that technique					X				
From a print to the period of life during which the print was made					X				
From a period of life to the artistics movements actives in Germany during that period	X								
From an artistic movement to the artists that represent the movement					X				
From an artist to the artistic movement he belonged to					X				
ACCESS PATHS TO CONTENTS									
A list of all the prints					X				
A group of masterpieces		X	X						
Some thematic tour of prints grouped by theme			X	X					
A list of Munch's periods of life					X				
A list of all the techniques					X				

Prints are of course considered as the core content of the entire website. Even if there was not a specific and explicit indication to give such a prominence to prints, all visions, motivations, goals and even the specific designer expertise brought to this decision. From the curator's vision it was clear that the relationships between Munch's periods of life and the artistic movements were important to make understandable the ambience in which a print was composed. However, a major problem which emerged clearly during traceability analysis was that the curator did not want to deal with historically disputed elements such as the artistic influence of an artistic movement on Munch's artwork. In fact, it is still very controversial among art experts which artistic movement most influenced Munch and why. The curator did not want to get into this debate and therefore decided for a rather neutral position. The design solution has been to leave the semantics of this relationship a bit "ambiguous": the artistic movements have just a "co-presence in time" relation with a period of Munch's life.

Traceability Report 6
Learning at Europe

Date: May, 2005
Target: Overall project team
Goals: Communicate the project status and tune-up the design

Executive summary

LearningAtEurope is an educational project aiming at fostering the development of a "European Identity" for the new generations of European students. L@E proposes an educational approach novel in several respects: advanced content, technology-enhanced eLearning, a multicultural experience, coupled with engaging "games" and a cultural competition among different European classes.

This report documents a traceability analysis for this project, whose main goal is to reorganize the complex and various material describing and designing the experience, to pave the grounds for a reengineering activiy.

L@E team uses this analysis for the following reasons and profiting by the following benefits:
- Internal communication, to communicate the project status to all the team members
- Reverse requirements engineering, re-organizing and refining requirements and surfacing missing information, fundamentals to understand the project but never explicitly documented
- Design tuning, surfacing missing design components and re-aligning the design with the project state-of-the-art
- Design revision, to facilitate the project revision before a new experimentation period

Introduction

LearningAtEurope (L@E) is an educational project by Politecnico di Milano (Italy) with the contribution of Accenture Foundation. The subject of L@E is "the birth of modern states in Europe", intending all the complex social, political, economic, religious and cultural factors that have lead to the arising of the modern states, as we know today in Europe. The educational approach of L@E is based upon a seven week experience: students (high schools) intermix traditional study (with downloaded traditional material and interviews to experts and scholars of different countries, different cultural approaches and different disciplines) with "on-line meetings", e-Forums, homework, etc. Four classes from different regions of Europe take part in the same experience, with a cultural competition (two against two) among them. During these "on-line meetings" (three for each experience), students meet in a shared 3D virtual space, accessible via Internet. Each student (two per each class), connected with the environment, is visualized (in the world itself) as an "avatar". In the virtual space, under the guidance of an "educator", students "walk around", find objects, interact, chat, "fly" and play games. Discussions along the way are used both to clarify difficult aspects in the subject and also to provoke the exposition of similarities and differences among different European regions. A final homework provides to each class the opportunity of digging in its own context (and past) and to expose it to the other classes. An electronic forum is used to allow students, participating in the same experience, to keep in touch, continue their discussions, cooperate for the homework, exchange documents, etc. A shared discussion space is also used to organize "meet the expert" sessions, where students are allowed to directly interact (for a week) with the experts of some of the aspects involved in their experience.

In a first full experimentation year, between 2004 and 2005, 48 classes from 6 European countries (Belgium, France, Italy, Norway, Poland and Spain), nearly 60 teachers and 1,000 students were involved. A new advanced experimentation year, between 2005 and 2006, will bring the project at an industrial stage. Before this new experimentation, a complete revision of the whole setting of the experience will be performed. A traceability analysis has been requested to facilitate this revision activity.

The tracing analysis process for the L@E project, has been structured in (a) a preliminary plan, (b) a basic information re-organization, (c) two elicitation and analysis meetings and (d) a specification activity.

Preliminary plan

In the L@E traceability was not perceived by project managers as part of the traceability process but as a self-standing activity. From this point of view, an important preliminary activity has been to understand why traceability was performed and which benefits would it bring to the project.

For L@E, stakeholders of the traceability analysis were the project manager and the experience designers.

While each designer developed a single feature of the experience, a wider understanding of how requirements and educational goals were considered in the design were needed to refine and improve their solutions.

A second goal were to keep the “fil rouge” of the decisions taken during the project, understanding which elements cannot be modified and which ones may be altered in the revision process.

The expected results were to obtain a global picture of this complex project, highlighting the relationships between its different pieces and the reasons why those decision were formerly taken.

These aspects were discussed during a first short meeting and a preliminary plan was produced, clearly summarizing these goals and setting up the subsequent activity. Two meetings of four hours each were established: a first one with the aim of bring together the various elements of the project and to start tracing the first relationships between the set of goals/requirements and the design elements; a second one, the day after the first, with the aim to refine the analysis considering one by one the motivations of the design elements.

Basic information re-organization

A main needed activity during the first meeting were to re-organize requirements and design in a structured way. Requirements and Design documents were been produced, but not in a organized way: business or research goals were not distinguished from educational goals, technical elements of the application were mixed with the experience organization elements, etc. In the TRAMA method, this activity is called requirements and design "normalization"; in fact, the approach take into account the very frequent cases where the design specification is absent or incomplete, the requirements specification is absent or the requirements specification is unstructured or incomplete. In the L@E project, the requirements "normalization" activity consisted in structuring the previous knowledge in terms of general goals, educational goals, visions and requirements.

General goals, i.e. research or business goals at the base of the entire project was the following:

G1 Offering to schools a collaborative learning experience based on new technologies
G2 Basing the experience on historical contents
G3 Basing the experience on a multicultural approach
G4 Allowing the educational impact to be measurable
G5 Allowing to participate classes and pupils of every level and kind, not only the best classes in the best schools
G6 Minimizing the internal management costs of the experience

Educational goals, i.e. educational benefits for students needed for the L@E experience was the following:

B1 Knowledge (i.e. teaching a "know what" to students)
B1.1 About local (national) history
B1.2 About other countries' history
B1.3 About general historical concepts and processes
B2 Skills (i.e. teaching a "know how" to students)
B2.1 Use of "professional" English (as a tool to work)
B2.2 Use of technological tools for synchronous or asynchronous collaboration (3D worlds, forums, online communities, etc.)
B2.3 Group work (face to face collaboration)
B2.4 Collaborative work (remote collaboration)
B3 Attitudes (i.e. provoke an habit change to students)
B3.1 Sense of curiosity for history
B3.2 National identities are the result of a process: multiple cultures / multiple identities
B3.3 Improved attitude towards history
B3.4 Critical thinking towards knowledge: truth appears through a variety of opinions
B3.5 Different attitude towards knowledge, different learning modality (e-learning)

Visions corresponds to a generic and project-independent opinion of stakeholders towards how to do or how to understand something; according to Bolchini et al., they are strategic insights of stakeholders in the domain. In the L@E case, stakeholders had two visions that impacted on the design:

V1 Integration in schools' curricula
V1.1 Convenient quantity of commitment

- For students. The project aims at support schools as they are and not to subvert the internal organisation; the experience have to involve an entire class (12 to 25

students) and have to be guided by teachers with active and directive roles. The project may help teachers in managing different class segments.
- For teachers. The project aims do not include that teachers learn something about technology. The experience does not base on teachers' technological skills.

V1.2 Convenient use of infrastructural resources. The project must not requests to school an excessive use of laboratories or a too sophisticated technological equipment.

V1.3 The educational benefits have to be related with the general educational goals of schools and of their curricula. Teachers must be able to justify the time and organisational effort spent for participating in this experience.

V2 Characteristics of and educational competition

V2.1 It has to be a motivation for students in learning; it has to be a "true" competition and repay the commitment. Therefore the competition should be:
- Open: motivation should remain active for everyone until the end, also for micro-sessions
- Serious: it should repay different skills and valorise a deep understanding but it should not be frustrating

V2.2 It has not to be frustrating: participants should not be demotivated by difference of results with the others. This characteristic have to be balanced with the previous one.

V2.3 Engaging but not an end in itself; e.g. the access to cultural questions (the "serious" part) could be win with games involving "physical" or technical skills (the engaging part).

Requirements, i.e. consequences of goals and visions and indications for the design was the following:

R01 The experience have to include the use of collaborative 3D worlds
R02 The experience have to include the use of tools for asynchronous collaboration
R03 The experience have to include the teachers' active role
R04 The educational activities have to involve the whole class
R05 The activities have to be modularized in order to facilitate class segmentation
R06 The activities must require to students a minimum background knowledge
R07 The activities must not presuppose that teachers know how to use technologies
R08 The applications must allow to participate with a low technology level and include a degraded mode of use for low connections
R09 The historical contents have to highlight multiple opinions, disciplines, localisations and cultures involved in the topic
R10 The experience have to support the creation of a virtual communities of students, also after the end of the project
R11 The experience have to support the creation of a virtual communities of teachers, also after the end of the project

A further activity was the design "normalization". In this project was impossible to understand requirements impact considering only the software application. In cases like information systems or, more again, like educational applications, it is impossible to understand the project solutions without considering contextual information: it is needed to consider as design the technical-applicative aspect and the format, the procedures, the workflow, the activities of users, etc. In fact, specific design languages1 have been developed in the e-learning community to shape what they call the "educational environments". In this case a simplified model has been used to represent the design.
For L@E five design categories were taken into account: static components, i.e. the "bricks" the experience is composed by; dynamic components, i.e. how static components are assembled in

[1] Cfr. Botturi & Belfer [2003].

a workflow; transversal components being both static and dynamic or no one of the two; educational materials, i.e. contents of the educational experience; testing materials, i.e. all the elements used to measure the educational impact of the experience.

The **design** was then detailed as follows:

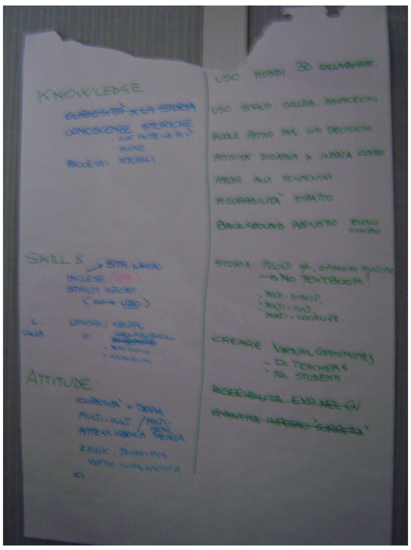
Static components
D1 3D synchronous collaborative sessions
D2 Asynchronous collaboration (forum/email)
D3 Class presentations
D4 Games

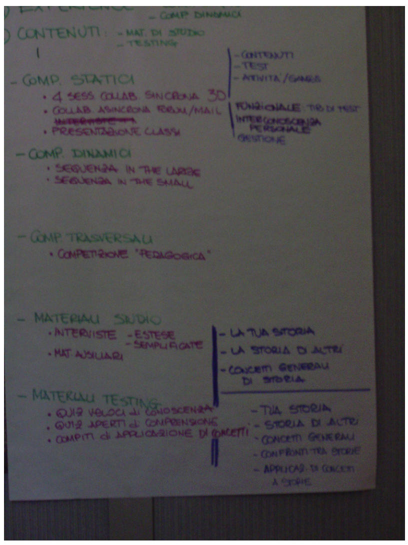
Dynamic components
D5 In-the-large sequence: succession of sessions, asynchronous sessions and off-line activities during the experience
D6 In-the-small sequence : succession of the activities, contents and tests in a session

Transversal components
D7 Educational competition in itself

Educational materials
D8 Interviews (extended and simplified)
D9 Auxiliary materials

Testing materials
D10 Quick questions on knowledge, "matter of fact" about local history, about other countries' history and about general historical concepts
D11 Open-ended comprehension questions about local history, about other countries' history and about general historical concepts
D12 Assignments & home-works (to apply the knowledge)
D13 Monitoring Tools & Procedures





Analysis of the Requirements Impact

	STATIC COMPONENTS				DYNAMIC COMPONENTS		TRANSVERSAL COMPONENTS	EDUCATIONAL MATERIALS		TESTING MATERIALS			
	D1 3D synchronous collaborative sessions	D2 Asynchronous collaboration	D3 Class presentations	D4 Games	D5 In-the-large sequence	D6 In-the-small sequence	D7 Educational competition in itself	D8 Interviews	D9 Auxiliary materials	D10 Quick questions on knowledge	D11 Open-ended comprehension questions	D12 Assignments & home-works	D13 Monitoring Tools & Procedures
GENERAL GOALS	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******
G1 Offering to schools a collaborative learning experience based on new technologies	3D world	Collaborative activities		Games involves collaboration	Workflow for knowledge	Workflow correspondance	Motivating						
G2 Basing the experience on historical contents								YES	YES	YES	YES		
G3 Basing the experience on a multicultural approach								Multiple points of view	About other cultures and histories				
G4 Allowing the educational impact to be measurable													Collect adequate information to measure impact
G5 Allowing to participate classes and pupils of every level and kind, not only the best classes in the								Self-sufficient material	Self-sufficient material	All based on given or simple materials			
G6 Minimising the internal management costs of the experience		Managed by teachers	Managed by teachers					Learning process managed by teachers	Learning process managed by teachers			Managed by teachers	
EDUCATIONAL GOALS	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******
B1 Knowledge					Yes: workflow for knowledge		Motivating	YES	YES	YES - motivating (factual notions)	YES - motivating (in-depth knowledge)	YES (attitudes)	
B1.1 About local (national) history													
B1.2 About other countries' history													
B1.3 About general historical concepts													
B2 Skills						Workflow correspondance							
B2.1 Use of "professional" English	Quick chat + 2D	Complex chat	Composition							Use of quick English		Yes	
B2.2 Use of technological tools	Perception: all; direct experience: it depends	Internal moderator	Authoring	Yes: use of 3D features								Yes	
B2.3 Group work	Integration of 3D and 2D		Organization	Fast decisions in group								Yes	
B2.4 Collaborative work											Quick group decision	Yes	
B3 Attitudes	Test: 3D simple, 2D complex	Assignment + spontaneous exchanges (80% vs. 20%)			YES: adequate workflow			YES	YES			Applying concepts and changing attitudes and learning	
B3.1 Sense of curiosity for history													
B3.2 Multiple cultures / multiple identities													
B3.3 Improved attitude towards history													
B3.4 Critical thinking towards knowledge													
B3.5 Different attitude towards knowledge													
VISIONS	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******
V1 Integration in schools' curricula	Technology and easy connectivity	Forum and email	Variable technologies		Acceptable effort demand		Acceptable level (not frustrating)	Acceptable difficulty level, and CV-related topic	Avoids the need to research for comprehension				Acceptable demand
V2 Characteristics of and educational competition				The most enjoyable part of competition	Scores ensure that nobody gets frustrated					Values students' preparation, is not frustrating	Values students' preparation, is not frustrating	Values students' preparation, is not frustrating	
REQUIREMENTS	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******	*******
R01 The experience have to include the use of collaborative 3D worlds	50% / 70% of the class uses 3D	NO	NO	Motivating - Designed to use 3D features		Designed to use 3D features	Motivating						
R02 The experience have to include the use of tools for asynchronous collaboration		Forum and email			Sequence forces use (and time to use) Async. Tools								
R03 The experience have to include the teachers' active role	Roles + class organization	Roles + class organization	Roles + class organization		More autonomy to teachers			Roles + class organization	Roles + class organization				
R04 The educational activities have to involve the whole class	Only 2 players 3D + 1/2 2D	All participate	Yes - Danger	Yes - Danger	In-the-large sequence must maximize segmentation and involvement: planning of activities	YES: planning of activities maximizes involvement		Yes - danger	Yes - danger				
R05 The activities have to be modularised in order to facilitate class segmentation	Each activity involves 1 to 4/5 pupils; contact moments between 2D and 3D		Different activities, different difficulty	Yes - Danger		YES: planning of activities in view of segmentation		Possibility of expert groups;		Different activities, different difficulty	Different activities, different difficulty		
R06 The activities must require to students a minimum background knowledge					Precise deadlines for materials			Self-sufficient material		All based on given or simple materials			
R07 The activities must not presuppose that teachers know how to use technologies		Plain web technology (forum) and email	Any format can be used to present the class (power point, word, etc.)										
R08 The applications must allow to participate with a low technology level and include a degraded mode of use for low connections	Implementation technology and only 2 players per class												
R09 The historical contents have to highlight multiple opinions, disciplines, localisations and cultures			Specificity in deep		NO			Interviews to experts of diff.				Comparison of cultures and places;	
R10 The experience have to support the creation of a virtual communities of students, also after the end of the project	Motivating	Extra discussions	Motivating		YES: workflow for increasing cooperation		Motivating					Collaborative assignments	
R11 The experience have to support the creation of a virtual communities of teachers, also after the end of the project		It doesn't work!											

RIM – Requirements Impact Matrix

The RIM matrix here reported lists vertically all the requirements-related information, i.e. general goals, educational goals, visions and requirements, and horizontally all the design components, i.e. static components, dynamic components, transversal components, educational materials and testing materials. The crosses represents relationships between these elements and each cell can hold a comment about the "rationale", the reason and the meaning of the relationship.

In L@E, cross cells have been filled according to two directions representing two points of view. The first one is the designers point of view, that considers the matrix vertically, design elements by design elements, because each designer developed just a single part of the entire application. The question that designers with the help of the traceability expert have tried to answer was: "Taking into account a single design element, how does it fit with requirements?". For instance, we can consider the element D1. Here the matrix has worked as a kind of checklist: each cross of the D1 column with requirements information has been the subject of a discussion: were the 3D synchronous collaborative sessions a solution or part of a solution for that specific goal or vision or requirement? Yes or no? If yes, how and why?

The result can be observed in the RIM matrix: 3D sessions allow collaboration based on new technologies (G1) since 50% to 70% of the class uses it (R01); this element improve the use of professional English (B2.1) thanks to the quick chat that players have to use to talk each other and provides to all the students at least the perception of a technological tool, while only two players per class can use it (B2.2); this component allows students to work in groups (B2.3) and improves their attitudes towards history and knowledge (B3), since the collaborative sessions can be performed after homework and traditional learning have been preliminary done; 3D technology used allow a good integration in schools (V1) and the participation of every kind of class, even with a low technology equipment (R08); this element motivates the creation of a virtual community of students (R10) and facilitate modularization of activities (R05); anyway, roles and class organization is let to the responsibility of teachers (R03) that keep in this way their main role in the learning process.

After the designers point of view has been considered, the RIM matrix has been review according to a more "client-centered" point of view: the real impact of requirements on the design has been therefore analyzed. The project manager with the help of the traceability expert has re-considered each cell of the matrix in a horizontal way, requirement by requirement. An example of a result for this kind of analysis is represented in the RIM matrix: how the requirement R03 (teachers' active role) have been taken into account in the design? Here again the matrix has been used as a kind of checklist, discussing the impact of R03 for each listed design element. In this case, while the requirement was that the experience had to include the teachers' active role, design gives to teachers the management and the organization of roles and groups in the class; in particular, this responsibility is done to teachers in the 3D sessions (D1), in the asynchronous collaboration (D2) and in the class presentations (D3); furthermore, the traditional learning activity in D2 (using the material D08 and D09) is completely managed by teachers. In this track, the L@E project manager has indicated that also in the sequence in-the-large (D5) more autonomy have been assigned to teachers in terms of time and activities management.

Analysis of the Design Motivations

	Visions	Requirements	Designer expertise	Specific understanding of the domain	Constraints	Law obbligations
STATIC COMPONENTS	****************	****************	****************	****************	****************	****************
D1 3D synchronous collaborative sessions		Offering to schools a collaborative learning experience based on new technologies		In past projects this technology has been a good motivation for e-learning	Use of the in-house 3D technology	
D2 Asynchronous collaboration		Create a community	Forums and emails are common tools to build up an asynchronous collaboration			
D3 Class presentations			Just to add an activity in the first (introductory) session			
D4 Games				Another good idea from past projects as a motivation for traditional learning		
DYNAMIC COMPONENTS	****************	****************	****************	****************	****************	****************
D5 In-the-large sequence	Acceptable effort demand	Workflow adequate to educational goals				
D6 In-the-small sequence			Workflow correspondance. Each session has not to be too "boring" or long or simple/difficult, etc.			
TRANSVERSAL COMPONENTS	****************	****************	****************	****************	****************	****************
D7 Educational competition in itself	It has to be a motivation for students in learning: open and serious			In past projects it has been a good motivation for e-learning		
EDUCATIONAL MATERIALS	****************	****************	****************	****************	****************	****************
D8 Interviews			This way to present information has worked very well in past projects			
D9 Auxiliary materials			This material go with interviews as background explaination (maps, biographies, etc.)			
TESTING MATERIALS	****************	****************	****************	****************	****************	****************
D10 Quick questions on knowledge	Values students' preparation, is not frustrating					
D11 Open-ended comprehension questions						
D12 Assignments & home-works		Improve knowledge, skills and attitudes				
D13 Monitoring Tools & Procedures		The educational impact has to be measurable				

DMM – Design Motivations Matrix

During the second meeting, a more detailed analysis of the motivations for design decisions has been performed, using the DMM matrix. This tool has helped the team of L@E in understanding explicitly why certain choices have been taken for the experience and where they were allowed to perform changes. In fact, the DMM matrix highlights the design "sources" i.e. the arguments that justify the design.

The matrix lists vertically the design elements of the application and horizontally their motivations (sources). The word "source" indicates here the reason because a specific design solution has been adopted and it can be of course related to visions and requirements for a part, but for another part they can be related to the following:

- the designer expertise, i.e. particular "good design" principles that are part of the designer's skills and that she/he applies in any case;
- a specific understanding of the domain, i.e. recurrent good solutions in a domain that the designer applies because she/he learnt it by other cases in the same domain;
- a particular constraint, e.g. budget limitations, time, technology limitations, etc.;
- a law obligation, e.g. copyright issues, personal data treatment, etc.

DMM matrix shows the sources for L@E design; in this case approximately 50% of design elements do not comes from requirements but from designer expertise or a specific understanding of the domain.
The only constraint here was the use of the in-house 3D technology developed for another similar project (the SEE, Shrine Educational Experience). No law obligations have determined design choices.

The matrix has been filled horizontally, trying to answer the "why" question, design element by design element. In this way some interesting information have been surfaced. For instance, the in-the-small sequence of each experience session or activity (D6) has that given structure just to correspond with the overall workflow; the motivation of designers was just to be not too boring or difficult or simple or long, etc. Another example is related to the class presentations (D3): why asking to classes to present itself during the first 3D meeting? As a side effect, this solution helps classes knowing each other and adds a small brick to build a community. But the true reason was that the first session in the virtual world was too "passive" for students, so the designer added a simple activity. These two elements (the class presentations and the sequence in-the-small) are therefore not strictly related to a requirement, they're not "untouchable" and they can be changed or improved saving the overall goals compliance of the experience.

Conclusions

This report documents the strategic goals of the experience, its educational goals, the stakeholders' visions that influenced it, the application requirements and the impact that al these elements had on the design; the document recalls also the main design components and the reasons why these solutions have been adopted. This traceability specification have been then used to validate the traces, i.e. to check the results with the project team and finally to keep this knowledge in a explicit and communicative document.

L@E team uses this analysis for the following reasons and profiting by the following benefits:

Internal communication
Traceability documents has been used by L@E project manager to communicate the project status to all the team members, i.e. to designers and engineers that implemented the application and that had just a partial understanding of the project, limited to what they did and developed.

Reverse requirements engineering
In L@E goals and requirements was never clearly stated. A lot of documentation about educational benefits and elements of the project was produced, but it was not clear e.g. what strategic goals and what visions were; therefore, each team member had her/his own opinion about what it was "untouchable" and what it was changeable in the application. Applying TRAMA in this project, forced to a more structured vision of this knowledge, re-organizing and refining requirements and surfacing missing information, fundamentals to understand the project but never explicitly documented.

Design tuning
L@E design was always described in discursive documents but never represented in structured components. TRAMA forced designers to distinguish between details and base elements of the application; this practice allowed surface missing design components and re-align the design with the project state-of-the-art.

Design revision
According to the main goals of this activity for L@E, traceability was performed to facilitate the project revision before a new experimentation period. TRAMA provided all the information useful for this activity, highlighting the relationships between the project components and their priority related to requirement compliance. Furthermore, this analysis helped designers in identify mandatory design elements, i.e. those elements that are related to a main goal or requirement, and in understand which parts could be changed instead. Some weak elements have been identified as well; e.g. the support for the creation of virtual communities of teachers that the project should provide does not work properly and new solutions to this need should be adopted.