A Survey of Issue Resolution on the Incremental Refinement of the System Scope in Web System Development

Norazlin Yusop
University of Technology, Sydney
PO Box 123
Broadway, NSW
+61-2-95144446
Norazlin@it.uts.edu.au

David Lowe
University of Technology, Sydney
PO Box 123
Broadway, NSW
+61-2-95142526
David.Lowe@uts.edu.au

Didar Zowghi
University of Technology, Sydney
PO Box 123
Broadway, NSW
+61-2-95141860
Didar@it.uts.edu.au

ABSTRACT

Increasingly, IT systems and Web systems in particular, fall into the class of applications where the scope of the application to be developed cannot be clearly articulated a priori. This is for a range of reasons, but is at least in part due to the complex inter-dependencies which exist between different aspects of the problem domain. While Whereas the core system requirements might be clear, the domain inter-dependencies lead to ancillary requirements which are only identified as the emerging system and its relationship to the domain are understood. A key mechanism in supporting the development of this understanding is the exploration of application pilots or prototypes, and the subsequent identification of solution or problem domain "issues". The resolution of these issues will often play a crucial role in supporting the development of domain understanding and hence project scope. In this paper we explore this phenomenon by analysing issue handling across a range of case studies. We describe a model of the issue resolution process and highlight those issues which are most likely to assist in clarification of project scope.

Categories and Subject Descriptors

D.2.9 [**Software Engineering**]: Management – life cycle, software process models. D.2.2 [**Design Tools and Techniques**]

General Terms

Management, Documentation, Performance, Design, Security, Human Factors

Keywords

Issues resolution, Scope Refinement, Web system, Business domain issues, project issues, system technical issues

ICWE'06, July 11-14, 2006, Palo Alto, California, USA.
ACM 1-59593-352-2/06/0007.

1. INTRODUCTION

The organisational and system related issues that exist in Web system[1] development are complicated due to the inter-dependencies of the system and the business domain. Such inter-dependencies complicate the resolution of issues, particularly when resolution of these issues is critical. Issues that arise during web development have to be addressed immediately for the Web developer to articulate and refine the system scope better.

It is postulated here that in Web system development, both the system design and business domain (business processes, workflow, organisational issues) co-evolve, such that changes in characteristics of the design will inadvertently change and impact characteristics of the business domain. A recent study in Web systems impacts report on such characteristics and the nature of these impacts [20]. This (co-evolution of Web systems development and business domain) is particularly a challenge in Web systems as developers often need to build the system with a limited knowledge of the business domain.

This in turn impacts on the definition and refinement of the system scope as the scope can not be identified clearly at the initial stage of the development. The Software Engineering Institute [15] states that scoping is an activity that bounds the system or set of systems by defining those behaviours or aspects that are "in" and those behaviours that are "out". There has been very little empirical evidence on the way Web developers define and incrementally refine the system scope.

It is with this premise that this paper aims to guide Web developers in refining the system scope through resolving issues that emerge in developing a Web system. In achieving this aim, a conceptual model of issue resolution and scope refinement process is developed from a review of the literature. The conceptual model shows that resolution of an issue typically leads to building up knowledge of the domain which in turn leads to the refinement of the scope.

A survey of issue resolution and scope refinement was conducted to validate this conceptual model. The survey findings can be used

[1] In this paper when we use the term *Web system* we are referring to those system which utilise web technologies as an integral element of a functionally complex system which typically incorporates interfaces beyond the organisational boundaries.

as indicators of warning signs early in the system analysis phase to alert developers regarding their potential impact of issues on the system scope. It also assists Web developers to rigorously derive knowledge about issue resolution and proactively identify how it can help in the refinement of the system scope. In this paper, we ascertain that issue resolution during the design and implementation of the system is fundamental to the clarification of the system scope.

2. RELATED WORK

There are very few related work in studying issue resolutions and scope refinement in Web systems development. This does not imply that study in this area is unimportant but because the problem in definition here is complex and involves an in-depth study of Web development. The first step in reviewing related work is to study the management of issue tracking and issue resolution in Web system development. Several topics are investigated such as identifying Web system development characteristics, processes and practices. This investigation resulted in the discovery of several sub-topics, such as impacts in Web system projects, domain modeling, and issue resolution, which contributes in articulating the context encompassing resolving issues in Web systems development. This literature review aims at providing a theoretical foundation for the study and development of the approaches and techniques. The findings from the literature review have been used to formulate the research aims, questions and the variables for the survey.

1.1 Web system development and Domain Knowledge

Jakob Nielsen [in 14] ascertains that the Web is different to conventional systems in many ways. The most important difference between a Web design and traditional interface design may not be a design issue but an organisational one. Web developers need to project their usability findings and design to huge number of content providers, most of whom have no understanding of user-centred design and poor understanding of the scope. The diversity of the Web system paralyses system developers with uncertainty, particularly when given the task to design Websites that meet diverse needs [14]. Lowe [10] states that with web-based systems, the technology, development skills, business models and competing systems change rapidly which results in the domain not only being poorly understood but also constantly evolving.

In an empirical study conducted by France and Horton [4], the authors learned that domain analysis process is iterative not only within major stages but also across the process stages. A significant amount of iteration is needed to obtain clear and precise models of the domain. In defining the domain, the scoping activity proves to be the hardest. This is due to an initial lack of understanding of the domain and suggests that "*as a deeper understanding of the domain and its relationship with other domains is developed, a better determination of the scope will be achieved*". In conventional systems, domain analysis and modeling can be done at the initial stage of the development but in Web development it is not possible due to uncertainty of the system design and complexity of system inter-dependencies [16].

1.2 Issue Tracking and Issue Resolution

The objective of having an issue tracking process includes "prioritizing activities, identifying cross-project influences and maintaining visibility on key issues so that they are driven to resolution" [1]. Previous knowledge or what the author refers as "lessons learned" in resolving issues are added to the knowledge base, which is used to assist with future similar situations and prevent team members from revisiting closed issues [1,5,17].

There are three categories of issues [1]. Firstly an issue is defined as a "technical or business situation with no known solution that is negatively affecting a project". Secondly, as a "technical or business situation that is negatively affecting a project for which there is a proposed solution that hasn't been fully implemented yet". Thirdly, as a "technical or business situation that is out of your control and negatively affecting a project". An issue is a concern that arises during development, which relates to integrating new technologies and implementing new ideas [18]. Issues that arise can have a direct impact on cost, schedule, scope and/or limitations of the project. Kuntz and Rittel [8] identify issues as a problem, or a question for which a decision can be made that will advance the project.

Web development processes need to facilitate a system's continual refinement and evolution based on feedback from end-users. This can be achieved by end-user involvement through integration of feedback and issue reporting mechanisms into Web systems [13]. Lehman [9] states that evolutionary processes are multi-level, multi-loop and multi-agent feedback systems. The use of such a feedback system generates new issues, which leads to changes in the system scope.

Issue reporting mechanisms are important in that they allow Web developers to resolve issues with their team members, by using sources of knowledge such as design artifacts, documents, consultations with stakeholders and solutions from past projects [6]. Web developers treat issue resolution with urgency but not necessarily in a structured manner [1,5,13,16]. This is particularly due to the characteristics of the Web systems development whereby the systems are developed within a short time frame, incrementally and with the consultation of a multi-disciplinary team [10, 20].

1.3 Approaches in Issue Resolution

Issue based approaches and processes are complex and current models and approaches do not provide guidelines as to how issues are resolved. An issue based approach and method, IBIS (Issue Based Information System) is widely discussed by Kunz and Rittel [8] and several other approaches were later developed as an extension to IBIS. IBIS is a method and an issue-based model for capturing design rationale on large, complex, design problems The models and tools that are derived and extended from the IBIS model are IWEB [3], HyperIBIS [7], gIBIS) [2], rIBIS and REMAP [2]. Among these, only IWEB offers a resolution approach, albeit conceptually.

In summary, users of IBIS-like models are critical of these models, primarily due to incorrect expectations [8]. From the review of various literature, it was inferred that there were several factors that made issue resolution a problem for developers and project managers alike. Chin [1] highlights the several reasons including that issues tend to lead to more issues, are often interrelated, issues are a reason why projects fall behind schedule and issues needed visibility until they are resolved. Hence, it is crucial that issues are tracked and resolved in an effective manner as they emerge.

1.4 Issue Resolution and Scope Refinement

From the analysis of the literature, it is ascertained that the definition of the system scope is problematic and a challenging task in Web development projects [5,16]. There are also various organisational and system technical issues that emerge during a Web project. In order to resolve such issues developers use paper prototypes and organise quick sessions with users to seek their assistance in making urgent decisions [5]. Such issue resolution aids the Web developer in identifying the finer system scope as specific decisions are not discussed when the initial requirements are given to the Web developers. So far, there has been no empirical evidence looking at the way Web developers incrementally refine the system scope

2. AIMS

In a Web development project, the client presents the project brief, which contains a brief description of the system scope [5,11,16]. The system scope is rarely identified clearly upfront due to uncertainty and complexity of the system inter-dependencies. The Web developer then develops a system prototype (capturing the design). While developing the prototype, the developer discovers issues that are then resolved through a knowledge source, such as other developers, the client, archived documents of past projects, design artifacts and the project brief. Currently, this approach is not done in a systematic way. It is also not supported in a meaningful way by design tools nor recognised in formal models of the process. The system scope is clarified from the issue resolution process and by providing support for this process; Web developers will be able to improve their understanding of issue resolutions and scope refinement.

The aim of this study is to identify the way issue resolution is supported and how resolution of issues fundamentally informs and modifies the understanding of the scope of a project. The research statement for this study is:

"Ongoing issue resolution during the design and implementation of the system is a crucial element which informs and changes the scope of the system"

The main research question that addresses this statement is:

"In what ways does issue resolution assist in the incremental refinement of the known system scope?"

3. IDENTIFYING PROCESS IN ISSUE RESOLUTION AND SCOPE REFINEMENT

In order to have a better understanding of the system scope, a conceptual model showing the process of the issue resolution and scope refinement is developed based on findings from literature [5,7,8,16,19]. Figure 1 shows that the Web developer goes through several stages in resolving issues and discover a problem when developing the Web system. The Web developer then contacts the client to discuss resolutions for the issue. The problem is then flagged as an issue with an indication that the issue needs to be resolved with other developers. The Web developer then evaluates the issue by identifying the origins of the issue and the level of affect it has on the system. At this stage, the Web developers go through several steps, which include reviewing design artifacts such as UML diagrams, structured analysis diagrams, and other documentations such as email archives, project brief and functional specifications. Throughout these activities and discussions with the client and development team, the Web developer is able to resolve the issue and also clarify the system scope better.

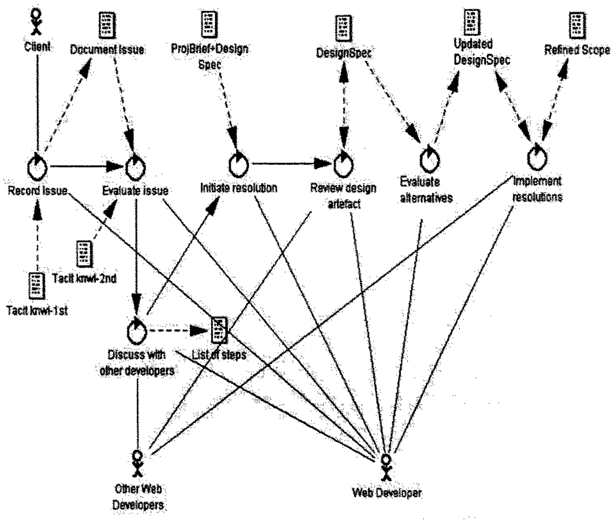


Figure 1: Conceptual model of issue resolution process

Figure 1 shows that issues are clarified by going through several activities and articulation of various sources of knowledge such as the tacit knowledge ("tacit knwl"), resolution steps and by referring to documents such as Project Brief and Design Specifications. Throughout this process, the documentations are updated as new knowledge is derived through various activities. In this paper, we aim to identify if this process is occurring in commercial practices and how issue resolution helps in the clarification of the scope. In order to identify this, we undertake a survey, where questions about issue resolutions and scope refinement were asked. The aim of this survey is to evaluate if the findings from the survey is consistent with the conceptual model.

4. METHODOLOGY

This study begins with a critical analysis of the scholarly literature and the survey of commercial tools. Research statement and sub-questions are then developed based on performing content analysis of both. The survey is then conducted to investigate the rationale, and reasoning behind issue resolution and particularly in examining in what way issue resolution refines the scope of the system.

4.1 Data Collection Techniques: Content Analysis and Survey

Content analysis technique is used for making valid inferences from the identified research literature [12]. The focus is on analysing the data qualitatively in that the objective is to investigate the study and practices of managing issues and issue resolution in Web system development.

The findings from the literature review are used to formulate the research question. The survey's objective is to investigate the rationale, and reasoning behind issue resolution and particularly in examining in what way issue resolution refines the scope of the system. A questionnaire[2] consisting of closed and open-ended questions are used for the survey. The type of survey used for this study is an online questionnaire and face-to face interviews.

The participants for the survey are Web developers (within Australian industry) who within the last 3 years have developed a Web system for their company or client. 45 Web developers were contacted via electronic mail to participate in the survey and 23 Web developers later participated. Those who responded were a representative sample of the 45 Web developers contacted. The participants have had experienced in various Web development projects from a broad spectrum of development areas (involving the internet, intranet and extranet), application domain (educational institutions, hospitality, government and software vendors) and company sizes.

When responding to the survey, the participants were asked to refer to a particular Web development project in which they had experience in tracking and resolving issues. In addition to providing technical input to the project, the developers also liaised with the client (or relevant stakeholders) to advice on the required changes and processes to accommodate these changes.

For the online survey, the participants were given an URL to submit their responses. The responses were analysed and followed up by an e-mail or telephone contact (according to participants' preference of contact mode). They were contacted to further clarify some of their responses.

In both survey types, the responses given were rich in information as participants shared their experiences and knowledge in dealing with issues in developing Web systems. Some participants also offered valuable suggestions as to how issues could be better handled in future, if they were to develop such similar Web system project. For the face-to-face interviews, the sessions were recorded on an audio tape and transcribed on a document. Although there was a relatively small sample of 23 Web developers that participated in this survey, the aim of the survey is to investigate potential factors and reasoning rather than aiming to ensure statistically significant results.

4.2 Analysis of Results

The data collected from the survey was analysed both qualitatively and quantitatively. Once the data was collected, the responses from the questionnaires were tabulated in a spreadsheet and analysed with basic statistical methods such as descriptive frequency. For open-ended responses, the data was analysed qualitatively to identify the common themes issues and to compare for similarity and differences.

5. FINDINGS

To further analyse and discuss the role that issue resolution play on the incremental refinement of the system scope, relevant questions from the survey questionnaire were extracted and the results analysed. We will discuss the participants' comments and structure a pattern from these responses. The issues discussed can be used as indicators and be taken as warning signs early in the system analysis phase to alert developers regarding their potential impact on the system scope. It can also be considered as a methodology that Web developers can use to help them rigorously derive the knowledge about issue resolution and identify how it can help in the refinement of the system scope.

We **firstly** look at sources of information that Web developers use in getting explanation about the system scope. **Secondly**, we discuss how the scope of the system is being refined throughout the Web system project and at what stages of the project was the scope being refined. **Thirdly**, we discuss in what way resolving issues lead to the refinement if the system scope. **Fourthly**, we evaluate which of the issues relating to business domain; project development and system technical when resolved lead to the refinement of the system scope. **Finally**, we discuss which issues the Web developers consider most commonly lead to the refinement of the system scope, when resolved.

5.1 Sources of information about scope

Participants were asked to select the information sources where they looked for explanation about the system scope. The options given were *project specifications document (or project brief), requirements document, design document, e-mail archives, contact project sponsors/stakeholders, look at similar projects already developed* and *others*.

14 out of 23 participants referred to the Project Brief for explanation about the system scope. The participants referred to the Project Brief as the initial source of information about the scope because it contained the main outline of the system they need to develop.

Most of the participants (16 out of 23) referred to Requirements Document for some general idea about the scope and stated that the document helped little, contained very few functional specifications and list of constraints for the system. From these specifications and constraints they formed their own understanding about the scope.

The Design Document was used by 11 participants as a source of information about scope because in most of the projects, participants could not articulate details about the system scope without first having some ideas and planning about the design. E-mail archives were used by 6 participants because it contained discussions between the client and the participants and between the development team.

Finally, 10 participants contacted project sponsors and stakeholders. The participants firstly discussed the scope of the system with their own development team because in those projects, the scope was vague. Clients were mostly contacted at the beginning and during project review meetings.

All the participants used more than one source of information for knowledge about the system scope. This was because, firstly, the participants were not able to obtain a detailed description of the scope from one single source. Secondly, each of these sources explained certain aspect of the scope, though in an abstract form.

5.2 Stages in Web development where scope is being refined

Out of 23 participants, 1 participant stated that the scope was not refined throughout the development while 2 participants stated that the scope was being refined at the initial stage of the

[2] Due to space brevity, the questionnaire is not shown here. Interested readers may refer to the questionnaire at this URL: http://surveys.uts.edu.au/index.cfm?surveyid=1112

development where the requirements for the system were written. 5 participants stated that the scope was refined during the design phase when prototypes were produced and in some circumstances when the design artifacts such as storyboards, screenshots, UML diagrams were developed. 3 participants stated the scope was being refined during the testing stage and 1 during the system implementation. A high number of 11 participants stated that the scope is being refined throughout the entire Web system project. Indicated below are some of their comments:

> "*We tried to define the scope at the beginning of the project when we did the proposal...and talked to the Project Manager. We refine the scope throughout the project due to different expectations... During review meetings, an issue was raised which wasn't thought about and because client's needs have changed... it had immediate concerns and the scope was refined.*"

Another Web developer explained the reason that the scope of the system was refined in his project was due to a new enhancement that led to changes in the project. Such enhancement came in the form of a new function, which client perceived as a better idea. In addition to enhanced features and functions, another Web developer also commented that the requirement for additional and revised content also led to the change and refinement of scope:

> "*The scope of the project changed throughout because the content that needed to be included got bigger and the customer wanted updated information each time. The contents got bigger each time. So an agreed 20 page website became a few hundreds.*"

The participants were also asked when the scope was refined during the project. Majority of the participants stated that the scope was refined during informal review meetings with their development team and with the client.

There were other participants who also commented that the scope was refined during design phase when the prototype was presented to the client. The client would request for changes in the functions, either by requesting additional functions or removing certain functions entirely. Some of these requests such as additional databases functions led to the change in scope. During this stage, participants and design team could have a better understanding of the system scope because functional specifications were better captured and issues such as data integration and data migration were resolved.

5.3 Relationship between Issue Resolution and Scope Refinement

The participants were asked, "*In what way does resolving issues lead to the refinement of the system scope*". At the beginning of the questionnaire, descriptions of the types of issues under the category of business domain, project development and system technical issues were explained. There were 6 options for this question: *it defines the system scope clearly, it identifies aspects of the system scope that were previously implicit, it leads to the resolution of other business domain issue, it leads to the resolution of other project development issues, it leads to the resolution of other system technical issue* and *other*. The participants could select more than one option for this question, as there might be a few ways which resolution of issues could lead to the refinement of the system scope.

13 out of 23 participants stated that resolution of issues helped in defining the scope clearly. A brief description of the system scope was specified before the development, however it was unclear as certain functional specifications and design related issues could not be articulated clearly. The detailed explanation about these types of issues will be discussed in the next sub-section.

13 participants also stated that the resolution of issues helped in identifying aspects of the system scope that were previously implicit. The system functions and constraints were stated briefly in the system scope and were thought to be implicit. However by resolving issues, such aspects of the system were identified to be more critical than first expected:

> "*Scope changes because we wanted the system to do this and that on top of what we wanted to have earlier. We later realised that we couldn't do this because of the constraints...so we made a few changes and gave a new proposal with a refined scope section*"

9 participants stated that by the resolving business domain issues, it led to the resolution of other business domain issues as well. One of the participants commented about his project, which was developing an online enrolment system for a University where one of the requirements was to align the business processes:

> "*..the existing business issues resolved other issues. For example, when a student dropped a unit, there were other business issues related to it, such as data-related issues, which were part of the scope. Therefore when it was resolved, it led to the refinement of the scope as well.*"

10 participants stated that resolution of system technical issues led to the resolution of other system technical issues as well. One of the participants commented about his project developing an online system, which handled reporting, and allocation of maintenance task. In this project, the participant developed a database, which stored maintenance staff's details and the type of maintenance work he did. It was initially a fairly basic database system but there were issues (eg. to include data that contained personal information) that required immediate resolution. The participant also encountered a data security issues about staff access level, which needed to be resolved before the database issues. Therefore by resolving the data security issues, the database system issues were also resolved. Due to this the scope of the system was therefore more understood and refined.

5.4 Types of Issues Resolved

The participants were asked "*Which of the business domain issues (project development issues and system technical issues) when resolved lead to the refinement of the system scope*?" The participants may select more than one option as they may encounter more than one issue during their Web project. Due to space constraints, we do not present the charts or graphs to show the frequency of responses. As an alternative, we present textual descriptions of responses and explain in detail those that have significant implications.

For the question on **Business domain issues**, the options were *compliance with government regulations industry, alignment of the system with business processes, migration of data (migrating data from remote system to Web system), organisational policy, staff data confidentiality* and *staff privacy*. 15 out of 23 participants stated that resolving issues pertaining to *alignment of* *the system with business processes* led to the refinement of the system scope.

This is consistent with the responses that the participants gave in the other questions. Aligning the business process with the system is integral to the Web system development project. One of the participants state:

> *"What affects the scope is defining the process. When you can understand that process, then you can refine the scope and understand what the client need. Defining what the customer needs affect the scope and understanding the scope leads to its refinement"*

For the question on **Project Development issues**, the options were *Requirements, Change request, Task progress* and *Time delay*. Most of the participants considered resolving *change request* (17 out of 23 participants) and *requirements-related* issues (13 out of 23 participants) led to the refinement of the system scope. Some of the participants stated that change requests became an issue because client requested for some functions to be changed which were going to be out of scope if implemented. In certain instances, clients claimed those requested changes would still be within the scope. Therefore negotiations of such change requests led to better understanding and clarification of system scope between the Web developers and clients.

In addition, another participant commented the following about requirements and change request:

> *"...when the system was half way developed, the client changed the requirements ... During implementation, the client asked for change requests primarily because business requirements have changed by the time the project was implemented. So requirements and change request issues were mainly governed by the client"*

Such change requests did not entirely changed the scope of the system but it actually helped in refining the scope because aspects such as change in the business requirements for this project could not be anticipated before the system is implemented. From the above comments, it was clear that change request and requirements issues assisted the Web developers to have a better clarification of the system scope.

For question on **System Technical issues**, the options given were *Data Security, System Security, Network Security, Data Integration, Data Migration, User Interface Design, Bugs, Performance* and *System Integrity.* 11 out of 23 participants commented on the data migration issues:

> *"There was an issue when we tried to feed the data from the old systems to the new Web systems and there were too much exception entries in the report for example when extracting staff information from old systems to the new Web systems. The old system could accept and store staff entry without [inputting] their department and title but then the new Web system requires such entry details... then during the system data migration, the new Web system rejected the data since the data didn't have the staff's department and title"*

From the above comments, the participant explained about a *data migration issue* that he encountered in a Web project, which was developing an online system for a government agency. The participant reported that during the requirements capture and system analysis stage, such data migration issues were not discussed in detail. The scope stated that the Web system should facilitate the entry of staff details. Such description was deemed to be satisfactory during that stage, however its impact was only known after the system was implemented. In this instance, such data migration issues when resolved led to the refinement of the system scope.

5.5 Issues Most Commonly Help With the Refinement of the System Scope

The participants were asked the issues that most commonly helped with the refinement of the system scope. The three issues that most commonly helped with the refinement of the system scope were "*alignment of the system with business processes*", "*change request*" and "*requirements-related issue*".

The participants were concerned with business process related issues as they did not necessarily have a good understanding of the client's organisation and operations. This is consistent with the discussions from the literature which state that business processes-related issues include issues involving the streamlining of business operations, structure and communication processes between the organisation's employees [17]. Such issues are complicated as it relates to other issues such as Internet security, change of application platform, change of operations, integration of workflow and information updating [17].

As such, it is critical that Web developers focus on such issues by discussing them during the stage when the system is scoped. However, this task could not be done effectively due to the inter-dependencies of the system and business impact and also because such issues were mostly design related issues which could not be anticipated at the early stage of the development.

One of the participants considered **business domain and project development issues** (and not system technical issues) as the types of issues commonly assisted in the refinement of the system scope. The reasons are as follow:

> *"The system issues did not change the scope during the project unless we discovered for some reason that it was technically impossible to do what we thought we could do. A change in requirement also led to the change in scope as well. They looked at the business process and said we need to get more data or info and this may affect the scope too. If bugs, it wouldn't really affect because it is just something which we could repair and fix"*

6. Discussion

From the survey results, we are able to get an in depth information because the participants provided deeper insights of the process they went through in resolving issues and how that helped them in the refinement of the system scope. Explained below are the discussions of the findings and in terms of how it relates to the conceptual model in Figure 1.

6.1 Issues are resolved using various sources of information

Survey results showed that various sources of information were used to resolve the issues, such as Project Brief, Requirements Document, Design Document, contacting project sponsors and stakeholders. The survey also showed that each of these documents provided different types of information. The Project

Brief for example, provided initial information about the scope as it contained main outline of the system they need to develop. The Requirements Document and the Design Document, on the other hand, provided information relating to design knowledge and certain aspects of the domain knowledge. By contacting the project sponsors and client, the participants could get a better clarification of the domain knowledge. Some of the issues clarified with the client included issues relating to business process, workflow, and business operations, providing security access to employees, data management and data handling issues.

The participants ascertained that by having a good clarification of the domain, they would then be able to have a better understanding of the scope. As such, discussions with the participants led to deeper understanding of this complex problem, which the literature (and conceptual model) did not clarify.

We found that the information the participants collected provided a rich source of knowledge. Furthermore, the participants used various information sources in certain stages of communication with the client and the Web development team. Most of the knowledge started with the articulation of tacit knowledge, whereby the participant first attempted at resolving issue through ideas or knowledge based on their own experience at resolving issues in past projects.

6.2 Stages in Web development where scope is being refined

The conceptual model does not show the stages (and related artifacts used) in Web system development where the scope is being refined. In the survey, the participants identified the stages and types of artifacts used in the refinement of the scope in each of the stages. Such information is pertinent as it guides the Web developer in predicting aspects of the scope to be refined during various stages of the Web system development.

6.3 Relationship between Issue Resolution and Scope Refinement

In the survey, the participants explained the types of issues that when resolved led to the refinement of the system scope. The conceptual model depicted this; however it does not state the reasoning and the communication details between the Web developer and the client and between the Web developer and his own development team. From discussions with the participants, it was highlighted that communication issues was also one of the main issues that they faced during the development. The participants commented that communication between different parties was a complex task as it involved people from different groups with different expectations. There were also many situations that the participants thought that they and the client (or system users) had a same understanding of a particular matter but which in fact they did not. This in turn later led to confusions and conflicts. This had great implication on the project as resolution of such communication issues might have an impact on the scope refinement.

In addition, the commercial practices showed that discussing a particular issue typically involved more than one person and through several documents revisions. It was an iterative process. This is however not shown in the conceptual model.

Resolution of issues helped developers in defining the scope more clearly. In addition, resolution of one type of issue also led to the resolution of other issues as well. This information is valuable for Web developers to identify how resolution of one issue can affect the other issues and how it helps them to understand the scope better. The interdependencies between issues should become an important aspect of the conceptual model. Such details are not currently included in the conceptual model.

7. RECOMMENDATIONS AND FUTURE WORK

There are several recommendations for Web developers that could be drawn from this study. Firstly, it is imperative that Web developers refer to past projects in order to have a better understanding about the scope. Importantly, to study the kinds of issues resolved, how they are resolved, how the resolution of such issues impacted on the scope. Secondly, during each deliverable stage, Web developers need to check if the design and changes requested are within the current scope, if not the Web developer needs to analyse or predict its impact on the completion time, budget, requirement on developers' knowledge and expertise and pertinently if it will result in or affect other issues as well. As such, the Web developers need to examine the connection between the issues and scope carefully.

The outcomes of this research highlight a number of key areas for future work. The first is to develop a clearer process that can be used to guide developers in using issue identification and resolution in the project scoping exercise. This requires clearer understanding on a number of key questions. If we have a set of unresolved issues then what type of prototype might best assist in resolution of those issues? Conversely, what sort of prototype is most likely to lead to the greatest clarification of project scope?

We are also currently developing a prototype tool which links an issue tracking system and a domain knowledge repository, so that the domain models emerge naturally (and possible automatically) as a consequence of the natural dialog between developers and clients, particular driven by issue resolution.

8. CONCLUSION

From the conceptual model and results discussed from the survey findings, it is ascertained that ongoing issue resolution during the design and implementation of the system is a crucial element, which informs and changes the scope of the system. Issue resolution and scope refinement are both very complex activities. However, by rationalizing its problems, identifying its origins and evaluating the types of knowledge needed, the Web developer will be able to articulate the resolution of issues more effectively. Understanding the rationale, and reasoning behind issue resolution and examining the way issue resolution refines the scope of the system will assist developers to resolve issues in a more systematic and structured way. This also leads to the ability in predicting the occurrences of issues and their role in the refinement of the tem scope.

There are several benefits that Web developers can derive from this study. Firstly, it helps to clearly visualise the different types of issues that are resolved during the Web system development. Secondly, the issues discussed can be used as warning signs early in the system analysis phase to alert developers regarding their potential impact on the system scope. Thirdly, it guides Web developers in identifying the precise process for issue resolution and defining functions in resolving issues that helps in the clarification and refinement of the system scope.

9. ACKNOWLEDGMENTS

The authors wish to acknowledge the support provided by the Australian Research Council under grant no. DP05598362 and the Fraunhofer IESE for providing us with the Spearmint™ tool.

10. REFERENCES

[1] Chin, G. *Agile Project Management: How to succeed in the face of changing project requirements*, AMACOM Publisher, New York, USA, 2003

[2] Dhar. V and Ramesh, B. Supporting systems development by capturing deliberations during requirements engineering. *IEEE Transactions on Software Engineering.* 18(6), 498-510, 1992

[3] Dutoit, A. Bruegge. B. and Coyne, R. Using an issue-based model in a team-based software engineering course, *6th Working Conference in SE Education & Practice*, Dunedin, New Zealand, 1996

[4] France, R.B. and Horton, T.B. Applying domain analysis and modeling: an industrial experience, *Proceedings of the 1995 Symposium on Software reusability*, 206-214, 1995

[5] Heller, D., Krenzelok, L and Orr, J. Webtop: realities in designing a Web-application platform, *Proceedings of the 2003 conference on Designing for user experiences*, Pleasonton, CA, 1 – 15, 2003

[6] Hitchins, D. K. In Systems Integration: Principles and Practice", *IEE Colloquium*, 3/1-3/12, 1990

[7] Isenmann, S. and Reuter, W.D. IBIS—a convincing concept...but a lousy instrument?", *Proceedings of the Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, 163-172, 1997

[8] Kuntz, W. and Rittel, H. *Issues as elements of information systems*, Working Paper No. 131, Inst. Of Urban and Regional Development, UC at Berkeley, California, 1997

[9] Lehman, M.M, Rules and Tools for Software Evolution Planning and Management", *Annals of Software Engineering*, 11(1). 15-44, 2001

[10] Lowe, D. Web Requirements An Overview", R*equirements Engineering Journal*, 8, (2), 102-113, 2003

[11] Lowe, D.B. and Eklund, J. Client needs and the design process in web projects', *Journal of Web Engineering*, 1(1), 23-36, 2002

[12] Michael, S. and Lewis, B. *Research practice: International handbooks of quantitative application in social sciences*, Sage, 1994

[13] Ramler, R., Wolmaier, K. and Weippl, E. From Maintenance to Evolutionary Development of Web Applications: A Pragmatic Approach", Proceedings of ICWE 2004, *Lecture Notes in Computing Science* 3140, 287-299, 2004

[14] Schneiderman, B., Nielsen, J., Levi. M., Conrad, F. Is the Web really different from everything else?", *Proceedings of Conference on Human Factors in Computing Systems*, 92-93, 1998

[15] Software Engineering Institute (SEI), Available at http://www.sei.cmu/productlines/frame_report/productLS.htm [Accessed 25th March 2005]

[16] Sood, S., Gilligan, R., Chandler, C. and Slack, S. Disneyworld.com redesign", Conference on Human Factors in Computing Systems, Vienna, Austria, 869 – 884, 2004

[17] Tang, S. M. An impact factor model of Intranet adoption: an exploratory and empirical research', *Journal of Systems and Software*, 51(3), 157-173, 2000

[18] Temponi, C. and Lambert, T. Joint product development implementation: an empirical study" in *Engineering Management Conference, 2002. IEMC '02. 2002 IEEE International*, 2(2), 725-730, 2002

[19] Yakemovic, B. and Conklin, J. Report on a development project use of an issue-based information system", Proceedings of the CSCW, Los Angeles, CA, October 7-10, 105-118, 1990

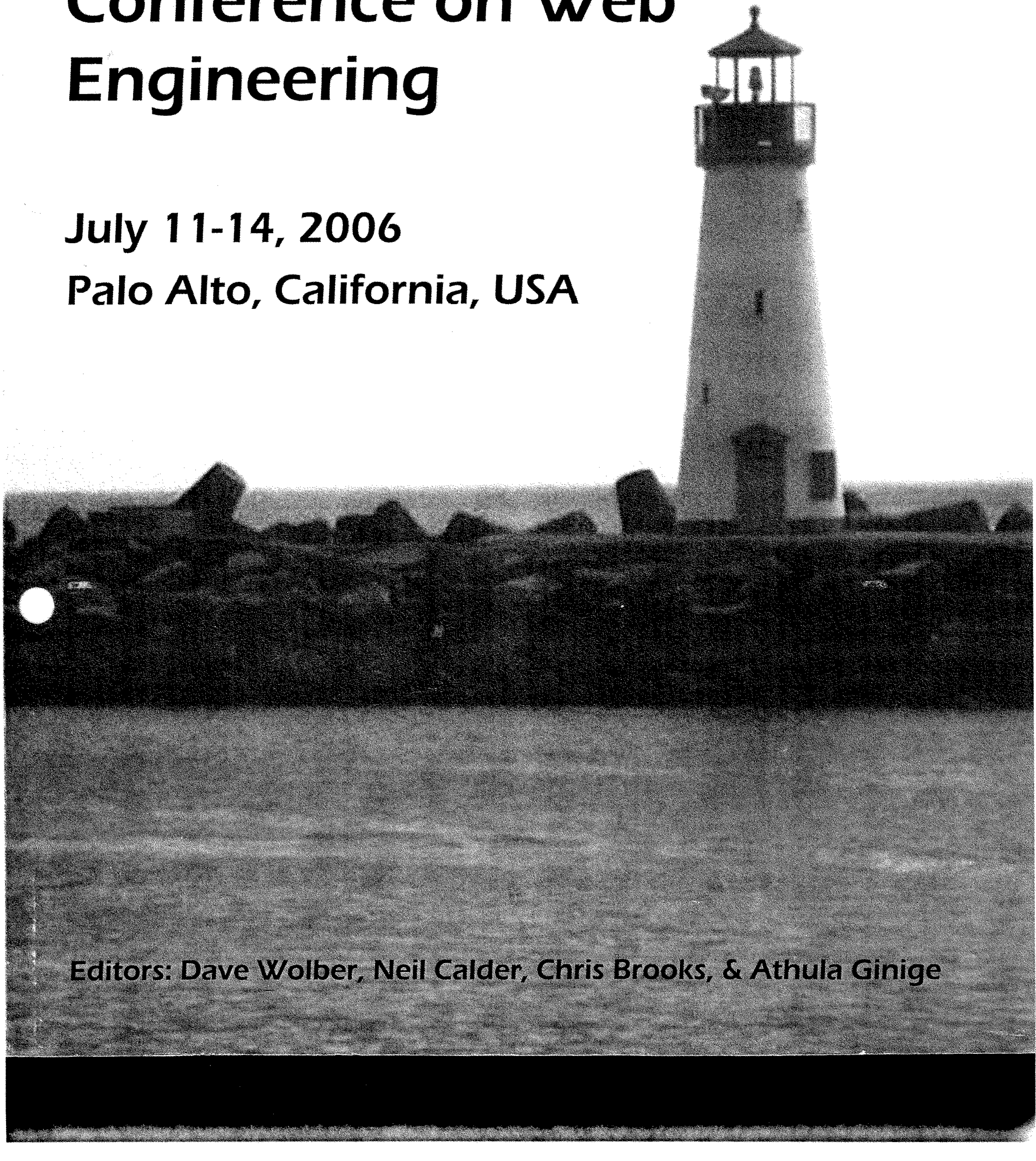
[20] Yusop, N., Lowe, D., Zowghi, D. A domain framework for representation of Web system impacts", Proceedings of the Fifth International Conference on Web Information Systems Engineering, Brisbane, Australia, Nov 2004

ICWE'06

The Sixth International Conference on Web Engineering

July 11-14, 2006

Palo Alto, California, USA



Editors: Dave Wolber, Neil Calder, Chris Brooks, & Athula Ginige

The Association for Computing Machinery
1515 Broadway
New York, New York 10036

ISBN: 1-59593-352-2

Printed in the USA