An Evaluation of Inquiry-Based Requirements Analysis for an Internet Service[1]

Colin Potts[a], Kenji Takahashi[b], Jeffrey Smith[b], Kenji Ota[b]

[a]College of Computing, Georgia Institute of Technology, [b]NTT Software Laboratories, Nippon Telegraph and Telephone Corporation

[a]potts@cc.gatech.edu, [b]{kt,sumisu,ota}@nttspe.slab.ntt.jp

Abstract

The Inquiry Cycle is a generic process model for conducting requirements elaboration. It consists of three activities: requirements expression, discussion and commitment. This paper describes an evaluation of the Inquiry Cycle model and a support tool, Tuiqiao, in the requirements analysis phase of a real project, a proposed commercial consumer information service based on Internet. The extent to which the discussion of requirements follows the Inquiry Cycle is analyzed using quantitative measures and qualitative classification schemes, and effects of adopting an inquiry-based strategy on patterns of collaboration among the team members are analyzed. The project gradually shifted from synchronous meetings to asynchronous and individual work patterns. A shift also occurred from an emphasis on discussion to an emphasis on commitment and expression. The requirements elaboration process was consistent with most aspects of the Inquiry Cycle, but analysts did not record reasons for requirements. The paper concludes with some recommendations for tool support.

---

[1]This paper is a revised and extended version of a paper published in the Proceedings of the Second International Symposium on Requirements Engineering, (York, United Kingdom, April, 1995).

1. Introduction

1.1 Requirements Analysis

Requirements analysis in practice is always an exploratory, *learning* experience. Because of this, the most significant practical problems faced by design teams during requirements analysis concern the creation, communication and management of informal ideas.

A survey of 23 development organizations [7] showed that projects in many applications have similar problems with requirements, including:

(a) Organizational solutions (e.g. responsibility assignments and meeting facilitation) are more pertinent than technological solutions (e.g. CASE tools).

(b) General-purpose technology, not structured CASE technology, is customized to keep track of informal requirements-related information.

(c) Organizations under-invest in support and education, so that analysts cannot effectively use the tools they have.

(d) Many requirements are invented during product design; for consumer products and software services, there is no definitive source of customer requirements.

Most of these problems concern managing and communicating informal information, not demonstrating consistency, completeness, feasibility or well-formedness of a specification.

Waltz et al [12] studied a single project, the goal of which was to develop a an object server for a development environment. The team's interactions over a four-month period were dominated by domain-independent communication and knowledge acquisition problems. Team members forgot issues they had discussed or remembered them differently, and made different assumptions. Meetings lacked facilitation and direction. The requirements process "shut down" when the project ran out of time.

1.2 Inquiry-Based Requirements Analysis

Based on these empirical findings, we have developed a requirements analysis process model and strategy, the *Inquiry Cycle* (Figure 1) according to which three activities repeat under the control of a strategy.

(a) *Expression* is the proposing or writing of requirements-related information. This includes requirements documents and domain-specific information, scenarios, and enterprise goals.

(b) *Discussion* includes formal meetings and circulation of comments and individual annotation of requirements.

(c) *Commitment* includes making decisions resulting from discussion, such as change requests, agreements about terminology, commitments to seek missing information, etc.

The Inquiry Cycle applies to many requirements methods. Previous papers [9,10] discuss possible instantiations with natural language text and scenario descriptions for expression and an extension of IBIS (cf. [2]) for discussions. Other instantiations are also possible.

The cycle is driven by scrutinizing expressed requirements, so technology that allows hypertextual attachment of annotations to parts of documents should form the central feature of any tool support. Such a technology, *Tuiqiao*, has been designed and prototyped [11]. Tuiqiao is a single-user tool for annotating requirements. Annotations are usually *questions*, which in turn lead to possible *answers*. Answers may be justified with *reasons* and may lead to *change requests*. Tuiqiao stores all these types of discussion contributions, the links between them, and links to the requirements. Links may be traversed using several browsing strategies, and the designer may quickly retrieve lists and networks of open questions or pending commitments. Tuiqiao is implemented in Hypercard.

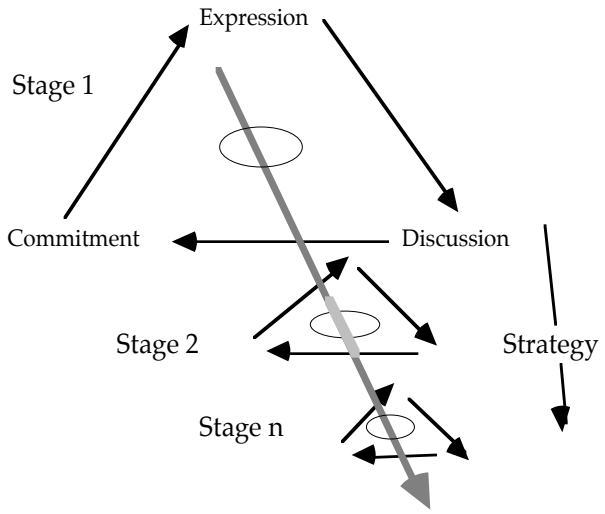

Figure 1: The Inquiry Cycle.

1.3 Empirical Investigation of Inquiry-Based Requirements Analysis

This paper describes an exploratory study to investigate an inquiry-based strategy during the requirements analysis for a commercial Internet-based consumer information service. It is not a controlled experiment because of the unfeasibility of controlling the many extraneous factors that affect real projects. Our goal was to observe the practices and experience of analysis and report descriptive quantitative data where appropriate, in particular the kinds and number of questions raised over time, their resolution, and the effects on the resulting requirements. We also wanted to assess the impact on the collaborative process of consciously following an inquiry-based strategy.

Three sub-teams participated in the project: *Analysts*, *Customers*, and *Implementors*. The four analysts, the authors, were working in an office in suburban Tokyo and spoke Japanese and English. The four representatives of the Customers were in downtown Tokyo and spoke Japanese. They really represent the development organization and acted as marketing consultants. The two implementors were in Mountain View, California, and spoke Japanese and English.

2. Consumer Information Service (CIS) Project

2.1 The Application

The application (CIS) is a commercial Internet service to augment paper-based consumer information sources (e.g. directories, promotional literature, guidebooks).

2.2 Project Chronology

An *application* of the Inquiry Cycle consists of a number of 'unwound cycles,' each with an expression, discussion and commitment activity. The chronology of the CIS project is shown in this way in Figure 2. Rectangles represent documents, circles represent discussions and triangles represent decisions. Solid arrows show direct mappings; broken arrows show indirect mappings.

2.3 Documents (Expression)

The CIS documents were informal. Four types were produced:

(1) *Customer notes (CN)*: Notes produced during and after meetings with customers.

(2) *Function lists (FL)*. Lists of phrases, each of which described a CIS function.

(3) *Goal hierarchy (GH)*. A goal refinement method derived from [4] was used.

(4) *Requirements document (RD)*: The requirements document was an eight-page natural language document written in English and translated into Japanese for customer approval. It was derived from the goal hierarchy by: (a) linearizing the outline into text; (b) describing the functions in detail; and (c) reordering and polishing the presentation for stylistic reasons.

The documents are summarized in Table 1. The sizes of the documents and are given in the table. For each type of document, the size monotonically increased from one version to the next.

Counting functions, instead of words also shows a monotonic increase from the first function list (FL_1) through the requirements document (RD_1). The number of functions is plotted in Figure 3.

2.4 Types of Interaction and Collaborative Tools (Discussions and Commitments)

The interactions that occurred during the project are summarized in Table 2.

Thirteen interactions occurred, ten of which were meetings and three of which were e-mail conversations. Tuiqiao was available in the electronic meeting room on all participant's screens. When Tuiqiao was used, one team member took notes on behalf of the team. In the case of two of the meetings (those discussing GH_2 and GH_3) an outline processor was used in the electronic meeting room, and the discussion notes were entered into Tuiqiao subsequently.

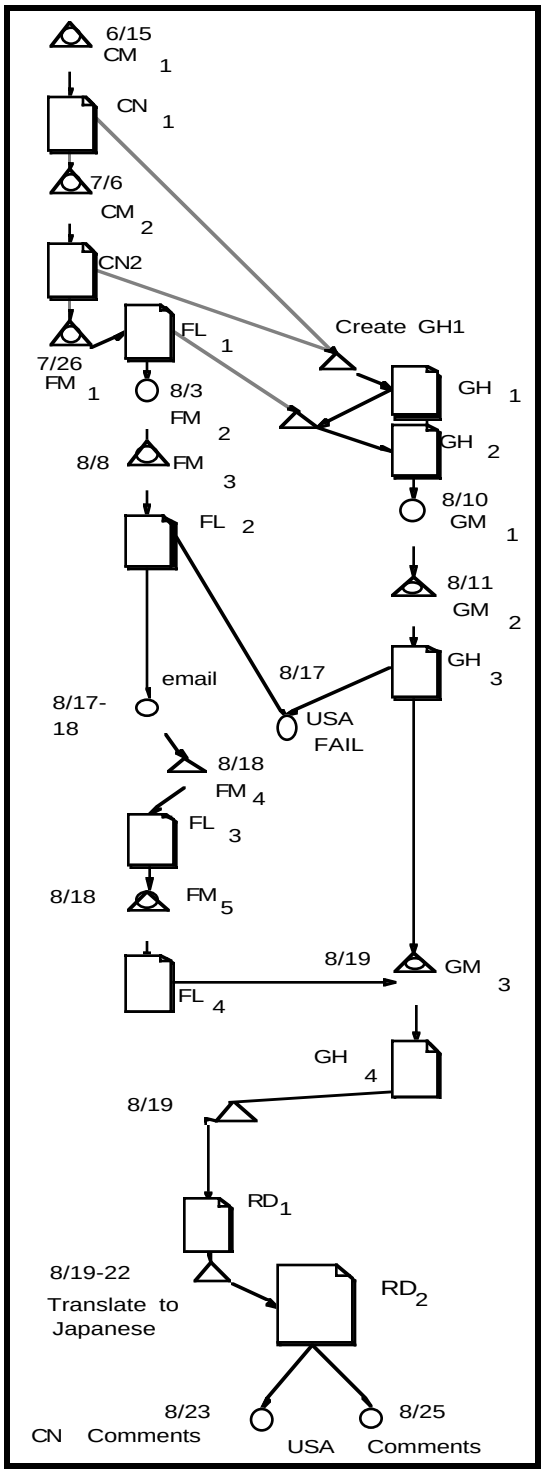

Figure 2: Case study chronology.

Ref.	*Size (words)*	*Details*
CN_1	974	Notes taken during and after first meeting with customer representatives. Produced using word processor.
CN_2	1,393	Notes taken during and after second meeting with customer representatives. Produced using word processor.
FL_1	93	Incomplete function list containing single phrases and words denoting desirable functions. Produced using word processor and copied into Tuiqiao.
FL_2	380	Revision of FL1 to include functions not discussed in first meeting. Produced using word processor and copied into Tuiqiao.
FL_3	991	Function list in which the top-level functions were each decomposed into several functions or operations that must be provided by CIS. Produced by editing Tuiqiao copy of FL_2.
FL_4	1,016	Function list with completed decomposition of top-level functions. Produced by editing Tuiqiao copy of FL_3.
GH_1	n.a.	Initial goal hierarchy produced using outline processor.
GH_2	417	Goal hierarchy with all functions from FL_2 added as leaf nodes. Produced by editing GH_1 using outline processor.
GH_3	538	Revision of GH_2 taking account of discussion. Produced by editing GH_2 using outline processor during discussion and commitment meeting.
GH_4	1,326	Revision of GH_3 incorporating decomposed functions from FL_4. Produced using outline processor during discussion and commitment meeting.
RD_1	2,081	Draft requirements document. Produced using word processor by: (1) reorganizing so that high-level goals do not appear; (2) removing goals operationalized by user behavior; (3) adding English explanation of the decomposed goals; (4) reordering and polishing for style.
RD_2	n.a.	Japanese translation of RD_1. Produced using Japanese word processor.

Table 1: Documents produced during case study.

To discuss requirements by e-mail, the analysis sub-team adopted a convention for message headers that made it obvious to recipients what type of discussion contribution was contained in the e-mail message and to what it referred.

The electronic meeting room used was NTT's *Cogent Room* [6], a state-of-the-art electronic meeting room. We used the Cogent Room in restricted ways. For example, Tuiqiao and the outline processor are single-user tools, not collaborative editors, and so when these were used, one team member would "drive" while others could watch the display on their screens.

No special decision making tools or facilitation strategies were adopted. Because it was small, the team was able to decide on changes to the documents by consensus.

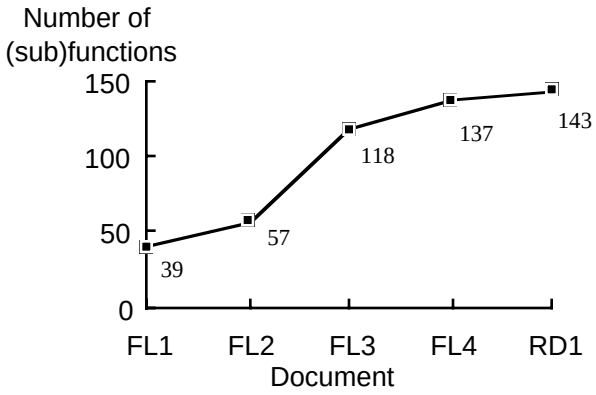

Figure 3: Size of successive function lists.

2.5 Approaches and Methods (Strategy)

The initial phase of the project involved gathering data on the application from the customer sub-team. No structured method appeared appropriate during this phase, and we simply took notes of their meetings with the customers. Following that phase, we used brainstorming to produce a decomposed list of CIS functions.

In parallel, a goal refinement strategy was adopted to help understand the context for the requirements. The objective was to identify missing requirements.

2.6 Data Gathered

All documents, discussions and commitments were stored in Tuiqiao. The discussions were analyzed into questions, answers and reasons. All meetings taking place in the Cogent Room were videotaped.

3. Requirements Discussions

3.1 Synchronous and asynchronous discussions

Not surprisingly, the project started with synchronous discussions (meetings), and then moved increasingly to asynchronous and individual activities. More work effort became devoted to commitment and expression activities (deciding and re-writing) than discussion activities, which dominated initially.

3.2 Discussion elements

Interactions differed in the numbers of questions raised and resolved. Some interactions (e.g. FM_4) had the purpose of raising questions; others (e.g. GM_3) had the purpose of resolving them (making commitments); while still others (e.g. CM_1 and GM_1) were opportunities to present ideas or requirements for subsequent discussion.

Ref.	*Date*	*Duration*	*Details*
CM_1	June 15	1:05	Initial meeting with representatives of customer organization.
CM_2	July 6	1:30 + 20 min. current product demo.	Second meeting with representatives of customer organization.
FM_1	July 26	1:07	Brainstorming to produce function list.
FM_2	Aug 3	1:35	Explanation of function list and further brainstorming
FM_3	Aug 8	1:43	Partial decomposition of function list.
FM_4	Aug 17-18	(e-mail)	Informal review of function list.
FM_5	Aug 18	> 2 hours	Completion of function list decomposition.
GM_1	Aug 10	1:30	Explanation and walk through of the goal hierarchy.
GM_2	Aug 11	1:11	Continued explanation and walk through of the goal hierarchy.
GM_3	Aug 19	> 2 hours	Discussion of incorporation of detailed functions from FL_4 into goal hierarchy GH_3.
USA fail	Aug 17	0	Attempted video conference to discuss goal refinement approach. Abandoned because of video equipment failure. (Administrative discussion continued on audio for c. 1 hour).
USA comments	Aug 23	(e-mail)	Implementation sub-team's comments on RD_2.
CN comments	Aug 25	(e-mail)	Customer sub-team's comments on RD_2.

Table 2: Interactions during the case study

Discussion contributions were usually easy to categorize as questions or answers, but sometimes the differences were not as clear cut. A common ambiguity was that between answers and reasons. Some answers trivially answered yes/no questions, and so most of the text of the answer was really an elaboration that could easily have been categorized as a reason. For example:

FM_2/FM_3 A14: This has two advantages against conventional [system]: (1) you can jump to related entries, (2) this links to related entries (the conventional one links only to synonyms).

Some discussion contributions contained questions, answers *and* reasons. For example:

FM_2/FM_3 Q1: How to record anchors to items? URL is not appropriate because still readers have to find out the right one in the page corresponding to the URL.

3.3 Discussion topics

Types of discussion topics: A tenet of the Inquiry Cycle model is that requirements discussions are discussions *about* requirements. That is, discussion contributions are attached to a requirements document. But some

questions were so general that the topic was not obvious. In such cases, the team attached the question to the most general section in the document that seemed appropriate.

Other questions were more specific, but about topics not mentioned in the document. Questions about the requirements *process* fall into this category.

Finally, a question could refer to a topic mentioned in several places. A question the team returned to repeatedly asked about the presentation of geographical information. The word 'geographical' sometimes meant general information about locations (e.g. neighborhood, distance from consumer's address, etc.), but elsewhere it meant cartographic information. Questions about 'maps' and 'geographical information' sprang up throughout the document without coordination. The logical response would be to attach a question to several requirements (and Tuiqiao's annotation mechanism supports this). In practice, though, this did not happen. It was too much trouble to establish extra links.

Discussion / artifact ambiguity: Another tenet of the Inquiry Cycle is the difference between expression and discussion; there are documents and there are meetings to discuss documents. Things are not so simple. Some discussions *became* artifacts, and some artifacts *mediated* discussions. The first function list exemplifies a document starting out in a discussion. We brainstormed the potential functions by answering the question: "What functions are desired?" One could view this process as the proposal of multiple answers followed by multiple commitments (a change request to add a function for every selected answer) but this is long-winded; what really happened was that the discussion produced a document as it went along.

Another example of ambiguity between documents and discussions occurred when one of the team sketched possible screen layouts on the whiteboard. He was trying to clarify the difference between map and geographic information. Thus the sketch was supporting material or part of an answer to a question concerning the difference between these two concepts. But that same sketch, polished and refined, was included in the Japanese translation of the requirements document. It was now part of a document.

3.4 Short Cuts

Sometimes we answered a question that was too obvious to record. Sometimes we changed the document without recording why. These *short cuts* are a natural aspect of requirements discussions [9], and the CIS project underlines the need to support them.

The implementation sub-team, which was almost cut off from the discussions because of time zone differences and technical problems, restricted its contributions to proposed changes. Its members must have discussed these requests before suggesting them, but they did not record the discussions. This reveals the significance of collocation and frequency of interaction. Team members who have been part of a discussion can easily create notes summarizing

the reasoning so far. Because they have participated closely in the discussion, they can allude to ideas knowing that they will be understood. Team members who are more isolated in space or time require more comprehensive and carefully structured notes that often still require verbal explanation.

3.5 Closure

Not all questions can or should be answered immediately, and not all commitments are acted upon immediately.

Resolution of Questions: Over one-third of the questions raised during the requirements analysis activity remained open by the time of the production of the draft requirements document. There are several reasons for questions remaining open.

Some questions remained open because they were about the implementation, and could not be resolved until later. For example:

FM4 Q7: Is this possible, because the features are on the client-side - not the server side?

Other requirements, especially some about preferences and priorities, had to be deferred until the customer organization had been consulted. For example:

FM4 Q5: How high a priority is this and at what stage does this facility need to be included?

Some questions could not be answered without further work, such as a technical feasibility study, market research, performance analysis, checking regulations, etc. For example:

FM4 Q16: Can we/do we provide a mechanism for making comparisons based on certain attributes (e.g. price) which assist the purchase decision making process [Requires consultation of organizational regulations or procedures.]

Two questions remained open because they depended on the resolution of other open questions. For example:

FM4 Q15: Does this [Reqt. 223] include the purchase mechanism? [Depends on Q16 above].

There were no cases of questions that remained open because they had been forgotten. Of course, there still may have been questions raised during verbal discussion or which team members thought of but failed to record. The breakdown of the resolved and open questions is given below in Table 3.

Status of question	*Number*	*Percentage of total (open)*
Resolved	26	63
Open (implementation)	*5*	*(33)*
Open (customer input)	*5*	*(33)*
Open (forgotten)	*0*	*(0)*
Open (dependent on others)	*2*	*(14)*
Open (require analysis)	*3*	*(20)*
Total open questions	15	37
Total questions	**41**	**100**

Table 3: Status of questions.

Two of the open questions referred to one requirement concerning evaluation of services (such as restaurant or movie reviews). This had been a major sticking point, because of possible legal ramifications of providing third-party evaluations as part of the required services. There were no other "hot spots" in the requirements, however; the remaining questions were fairly evenly spread through the document.

Enactment of commitments: Most of the commitments made during the course of the requirements activity were acted upon. There are several reasons for failing to act upon a commitment. Some answers were indirect and did not translate into easily stated change requests. For example:

FM_2/FM_3 Q3: Can we identify who accesses an entry? And can we know how many times?

FM_2/FM_3 A17: Yes. We can identify who accesses an entry each time.

Some commitments remained pending by the time RD_1 was completed. The reasons for commitments remaining pending were similar to the reasons that some questions remained open. A commitment might require further input from the customer before it could be acted upon or further analysis or other work before the requirements can be changed. For example:

FM_4 Q11: What accuracy is needed for the coordinates? Would information about which district the [service or product] location was in be sufficient (i.e. no need for coordinates)?

FM_4 A12: One of the goals is to use as a filter. So we want to use coordinates or addresses to determine proximity. If we are going to generate maps automatically, we need greater precision, but if the maps are drawn individually for each entry (store) then [that is] not necessary.

Some commitments are too detailed to act upon until the project gets further downstream, while other commitments may be difficult to act on until other commitments, upon which they are dependent, have been finalized.

Table 4 lists the numbers of the kinds of pending and enacted commitments at the time of production of RD_1.

Commitment status	*Number*	*%age total pending)*
Acted on	98	83
Indirect answers	10	8
Pending (too detailed)	*4*	*(37)*
Pending (customer input)	*1*	*(12)*
Pending (require analysis)	*4*	*(37)*
Pending (dependent)	*1*	*(12)*
Total pending	10	8
Total	**118**	**100**

Table 4: Status of commitments

3.6. Requirements Quality

Discussions and commitments can improve the expression of requirements in numerous ways. In Table 5 we show the number of commitments reached that improve the completeness, consistency and economy of the requirements. Completeness improvements add previously missing requirements. Consistency improvements remove contradictory statements. Finally, economy is the absence of 'gold plating' or unnecessary requirements.

Quality improvement	*Source = analysis subteam*	*Source = customer sub-team*	*Total*
Completeness	85	5	90
Consistency	5	0	5
Economy	1	2	3
Total	**91**	**7**	**98**

Table 5: Effects of commitments on requirements qualities.

Most of the commitments made reflected the recognition of previously missing requirements, most of these by the analysis sub-team. Unsurprisingly, of the three instances of gold plating two were found by the customer sub-team.

4. Effects of Inquiry-Based Analysis and Tools on Process

4.1 Analysis effort

Team members spent their time jointly and individually as shown in Table 6. This figure includes all analysis work, but excludes translation of English documents into Japanese.

Team member	*Joint effort*	*Individ. effort*	*Total effort*
1	9.5	11.5	20.0
2	9.0	11.0	20.0
3	12.5	9.5	22.0
4	12.5	3.0	15.5

Table 6: Case study effort.

In other words, we spent about half a person-week each over a period of about a month on this project (although, the initial interaction CM_1 occurred much earlier than this). Since the requirements activities of this project are not yet complete, and more detailed functional specifications and prototypes are likely to be produced, it is not appropriate to compare this analysis as if it were finished with the effort estimates for comparable requirements practices [10].

We consciously adopted the Inquiry Cycle as a strategy for our work. Most of the time team members knew what they were discussing and kept track of open questions diligently. This is not normal practice. While there were many cases of meetings containing discussions that were tangential to the explicit topic, we seldom had discussions without clear purpose. Normally, we knew how resolved or open the topic remained.

One explanation is that the note taker had more influence than other participants over meeting discussions. The note taker selected the appropriate Tuiqiao browsing view and presented that to everyone's screen. Other team members modulated the note taker's influence by suggesting topics for discussion, discussing the categorization of discussion contributions, and generally making explicit the inquiry-based nature of the discussion. Note-taking rotated among two team members.

4.2 Coordination overhead

Coordination overheads, delays or extra effort that are required not to help the analysis project make progress directly so much as to maintain and service the tools and methods used, were not the problem we expected. Some were encountered, such as a meeting between the implementors and the analysts that was abandoned when the conferencing system failed.

4.3 Technology Use

Noticeable shifts in technology use occurred during the 10 weeks of the project. Although many types of tools were useful throughout this period, the emphasis shifted over time, an emphasis that reflects the tasks being performed during expression, discussion and commitment work. Table 7 provides details about the task and the technological aids found most useful in the different phases. Figure 4 summarizes the process followed and the main

phases and the sequence followed. (This is not meant to imply that the phases were followed absolutely in the sequence followed or that they should be.)

Phase	*Tasks*	*Typical properties*	*Tools*
Famili-arization (CM_1, CM_2 / CN_1, CN_2)	Information gathering. Note taking.	Collab. but collab. tools would be too intrusive.	Pen & paper. Word processor.
Idea exploration (FM_1, FM_2, FM_3 / FL_1, FL_2)	Brain-storming Inquiry Exploration.	Collab., very informal, largely synchronous.	Mtg. room. Tuiqiao.
Systematic analysis (GM_1, GM_2, FM_5 / GH_1, GH_2, GH_3, GH_4, FL_3, FL_4)	Refinement. Scenario analysis.	Collab. but including individual work.	Mtg. room. Outline processor.
Informal review (FM_4, USA fail, USA comments, CN)	Inquiry. Scenario analysis.	Exclusively collab. Most effective and efficient when asynchronous.	E-mail.
Document preparation (RD_1, RD_2)	Writing.	Exclusively individual.	Word/ document processor.

Table 7: Summary of activity phases.

4.4 Assumptions about Mode of Use

Some of the tools required users to adopt specific processes. Tuiqiao requires an assumption that discussion *is* the annotation of a document. In practice this means that the structure of the document drives the discussion. For most of the time this ensured that discussions stayed on track and the requirements were analyzed systematically, but the assumption broke down during the period between the production of FL_1 and GH_1. FL_1, a document produced as the result of an informal brainstorming exercise, became the driving force behind the discussion of the requirements. Because FL_1 had not been produced using a systematic method, there was a danger that many fundamental issues would be overlooked.

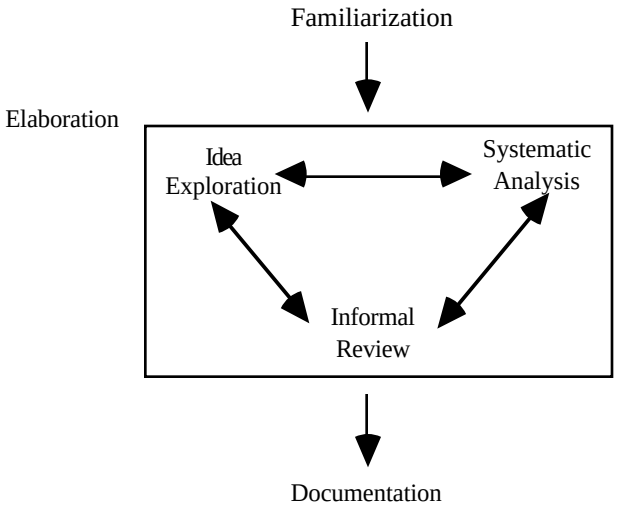

Figure 4: Summary of activity phases.

Tuiqiao supports short cuts, such as the attachment of an answer or a change request to a requirement without the need for an intermediate discussion. This is very valuable. If users are forced to classify their discussion contributions as "speech acts" then it essential that they do not also have to obey an arbitrary speech act grammar. One weakness of Tuiqiao, however, is that it does not support questions that follow from a discussion. Nor does it handle contributions that have no obvious anchoring requirement. There is a need for free-floating discussion contributions. This is related to Constantine's idea of a *parts bin*, ideas that are not yet thought out, but may be useful later [3].

5. Discussion and Future Work

5.1 Patterns of collaboration

During the requirements activity there was a trend from work that was largely dominated by discussion to work that was equally dominated by commitment and expression tasks. Early discussions were free-floating and led to the production of draft documents. Later discussions tended to be grounded in the documents already produced. Simultaneously, there was a trend from synchrony towards asynchrony and individual work.

These trends affect the types of technology chosen and the implications for technology designed to support the tasks. The tools used during this project supported our activities well, but the transition from one phase to the next was not sudden or recognizable at the time it occurred, and the often poorly integrated tools therefore introduced discontinuities into the process and made it difficult to keep track of information effectively.

Asynchronous and individual work generally require more careful coordination than synchronous meetings, and large specifications in projects with formal change control procedures do need to be controlled as rigorously as source code. Not all individual or asynchronous requirements work introduces uncontrollable inconsistencies,

however, and where documents are informal the team must coordinate its efforts manually. In the CIS project, it was necessary to embed all the functions in the goal hierarchy and analyze the composite document to check whether anything was missing, but this required domain knowledge and was *not* a consistency checking activity.

5.2 Structuring of discussions

The CIS project demonstrates that structures and systems can aid requirements analysis long before requirements can be expressed formally. But not all structuring procedures are appropriate, timely or helpful. For example, the eventual purpose of an interaction (especially meetings) was not always the purpose intended in advance. It is a mistake, therefore, to insist that interactions be pre-structured or pre-planned as part of a complex program of work.

Structuring discussion *content* can also be taken too far. We distinguished three types of discussion contribution: questions, answers, and reasons. Questions and answers (or statements) are intuitively different speech acts, but reasons are not obviously distinct from answers. Nor are they obviously worth recording. We did not record reasons, which is consistent with the finding that very few questions raised during requirements meetings are 'why' questions [5]. Insisting that discussants justify their arguments simply conflicts with reasonable practice.

We conclude that only questions and statements (i.e. answers to questions or assumptions about requirements) are easily discriminated speech acts, and that tools supporting discussion should not *require* (although they could provide) a more elaborate discussion schema than one providing these two types of discussion contribution. This contrasts with the many attempts to refine IBIS model to make it more formally expressive.

Not all discussion contributions are about requirements, so the annotation metaphor of Tuiqiao needs to be extended; but neither are all discussions disembodied, as the network metaphor of gIBIS [2] suggests, so it is also necessary to ground discussions in the topic discussed. This contradiction implies that discussion and commitment support tools should permit general, free-floating discussion contributions, contributions that annotate parts of existing documents, and contributions that qualify existing discussions. Networks, annotations and lists are all needed in different situations, and the analyst should not be forced to choose a tool that only provides one.

While careful categorization of the type of a discussion contribution may not be as valuable proactively (i.e. to the analyst) as it is retroactively (i.e. to a later implementor or the researcher), categorizing the *status* of a contribution is proactively useful. Classifying open questions by the reasons why they are still open directly suggests actions that team members might perform. The types of questions can therefore be used as reminders of outstanding work.

We have emphasized textual expression of requirements, but we observed above how informal sketches help mediate discussions while also acting as nascent documents. Pictorial representations of requirements 'talk back' [8] to the analyst more effectively than textual descriptions, and they blur the distinction between expression and

discussion. The ambivalent role of scenarios in earlier work [10] also stems from the concrete portrayal of system behavior being at once a medium for discussion and a description to be discussed. More work is needed to study patterns of discussion and elaboration when scenarios, mockups and other informal sketches play a central role in the definition of requirements.

5.3 Requirements completeness

Simply keeping track of the number of functions specified is a valuable indicator of completeness. Although a requirements document cannot be shown to be complete, analysts can become more confident about completeness if the number of functions grows asymptotically. When further discussion ceases to yield many new insights (i.e. new or decomposed functions) it is time to express and review the requirements more formally. We are investigating the use of goal refinement and scenario analysis in conjunction with the inquiry-based approach as a way to estimate completeness of a set of requirements against the enterprise goals [1].

5.4. Ongoing and Recent Evaluation projects

Unlike experimental studies, case studies, especially those in which researchers are their own subjects, fail to control for contextual variables that occur in real-world projects. Thus, to reap the benefits of the case study methodology (scale, complexity, and organizational context), one must repeat case studies in different settings. A previous paper [10] dealt with a demonstration of the Inquiry Cycle rather than a pure case study. Other more formal case studies are now under way.

One is a collaborative requirements annotation tool being designed by the authors. This project also involves Tuiqiao and meeting room technologies. However, because much of the analysis is taking place at long distance, greater use will be made of remote conferencing tools and e-mail.

Finally, another much larger project in a US engineering company is also being studied by the first author, using a participant observation technique. This project is designing a software infrastructure for a range of future communication products.

References

[1] Anton, A., M. McCracken and C. Potts (1994), "Goal Decomposition and Scenario Analysis in Business Process Reengineering," *Proc. CAiSE,* (Utrecht, Netherlands), Springer-Verlag.

[2] Conklin, J. and M. Begeman (1988), "gIBIS: A Hypertext Tool for Exploratory Policy Discussion," *ACM Transactions on Office Information Systems*, 6(4):303-331.

[3] Constantine, L. (1993), "Work Organization: Paradigms for Project Management and Organization," *Communications ACM*, 36(10): 34-43.

[4] Dardenne, A., S. Fickas, and A. Van Lamsweerde (1991), "Goal-directed Concept Acquisition in Requirements Engineering," *Proceedings Sixth International Workshop on Software Specification and Design* (Como, Italy), IEEE Computer Society Press, pp. 14-21.

[5] Kuwana, E. and J. Herbsleb (1993), "Representing Knowledge in Requirements Engineering: An Empirical Study of What Software Engineers Need to Know," *Proc. RE'93: IEEE Int. Symp. Requirements Eng.*, January 4-6, 1993, (San Diego, CA), IEEE Computer Society Press, pp. 273-276.

[6] Kuwana, E., Y. Nakamura, Y. Sakamoto, T. Masuo, E. Yana, T. Kitayama, M. Adachi and K. Wada (1994), "Computer-Supported Meeting Environments for an Intellectual Teamwork: COGENT" *Proc. Information Processing Soc. Japan SIG Groupware, 5-2,* pp. 9-16 (in Japanese).

[7] Lubars, M., C. Potts and C. Richter (1993), "A Review of the State of the Practice in Requirements Modeling," *Proc. RE'93: IEEE Int. Symp. Requirements Eng.*, January 4-6, 1993, (San Diego, CA), IEEE Computer Society Press, pp. 2-14.

[8] Potts, C. (1993) "Choices and Assumptions in Requirements Definition: Position Paper for Panel 'Keeping Track and Recording Rationale and Assumptions,'" *Proc. RE'93: IEEE Int. Symp. Requirements Eng.*, January 4-6, 1993, San Diego, CA, pp. 2-14, IEEE Computer Society Press.

[9] Potts, C. and K. Takahashi (1993), "An Active Hypertext Model for System Requirements," *Proceedings Seventh International Workshop on Software Specification and Design* (Redondo Beach, CA), IEEE Computer Society Press.

[10] Potts, C., K. Takahashi and A. Anton (1994), "Inquiry-Based Requirements Analysis," *IEEE Software*, March 1994, pp. 21-32.

[11] Takahashi, K. and C. Potts (1993), *Tuiqiao: A Hypertext Tool for Requirements Analysis,* Georgia Institute of Technology, College of Computing Technical Report, GIT-94-107,

[12] Waltz, D., J. Elam and B. Curtis (1993), "Inside a Software Design Team: Knowledge Acquisition, Sharing and Integration," *Communications ACM 36(10)*: 62-77.