

**On the Use of Scenarios in
Requirements Acquisition**

Anne Dardenne

**August 1993
CIS-TR-93-17**

**Department of Computer and Information Science
University of Oregon**

On the Use of Scenarios in Requirements Acquisition

Anne Dardenne

University of Oregon
Computer and Information Science Dept.
Eugene, OR 97403 (USA)
ada@cs.uoregon.edu

August 4, 1993

Abstract

This report will focus on the use of scenarios in requirements acquisition. Requirements acquisition is the very first step of the software life-cycle. It is now well established that it has a major impact on subsequent steps. There are several ways in which users can describe requirements. We will focus on scenarios. We are interested in scenarios for requirements acquisition because it seems that it is a very natural way for people to describe what they want a system to do. We plan to build a tool that will help acquire scenarios from users. The tool functionality is outlined in the report.

1. Introduction

This report will focus on the use of scenarios in requirements acquisition. Requirements acquisition is the very first step of the software life-cycle. There are several ways in which users can describe requirements. We will focus on scenarios. We are interested in scenarios for requirements acquisition because it seems that it is a very natural way for people to describe what they want a system to do. We plan to build a tool that will help acquire scenarios from users. An outline of the tool functionality will be presented. The report will be organized as follows. A study about how people describe requirements is presented in Section 2, together with its findings. A scenario representation scheme is described in Section 3. TAMS, a tool for acquiring and merging scenarios, is presented in Section 4. Some related work and conclusions follow in Sections 5 and 6.

2. A Study on Requirements Acquisition

2.1. Description of the Study

A study was conducted in April 1993 by the author to see how people describe requirements, and more specifically if they use scenarios in describing requirements. A total of eight acquisition sessions have been held, each with a different subject. (See Appendix for a complete transcription

of the acquisition sessions.¹⁾ Each subject has been presented with the same hypothetical situation. This situation was that the Computer and Information Science Department of the University of Oregon was considering implementing a meeting scheduler system and making it available to everybody in the department. The subject would be a potential user of the meeting scheduler. The subject was asked, as a potential user, to explain what kind of meeting scheduler s/he would like to have at the Department. The acquisition sessions were not guided in any way. The subjects were free to describe their requirements the way they wanted. They were not asked questions unless something they said was unclear. Especially, nothing was said about scenarios before or during the acquisition sessions, so as not to influence the subjects about their way of describing requirements: if they were to use scenarios, it would be spontaneously.

2.2 Findings from the Study

Below I describe several interesting findings related to the use of scenarios. In the context of this study, a scenario is defined as any combination of two or more actions.

2.2.1. Frequency of Use of Scenarios

The first finding is the fact that *all subjects used scenarios* in their requirements descriptions. Some of the subjects used more scenarios than others, but they all used some. The average was 3.6 scenarios per subject. As the subjects were not told anything about scenarios, they used scenarios spontaneously. Also, as the application considered was a meeting scheduler, it was not a system that led to scenario-centered requirements the way reactive systems often do [Kel91]. The application in itself did not suggest that the subjects use scenarios. This systematic use of scenarios by every subject of the study backs up the idea of focusing on scenarios for requirements acquisition. Scenarios seem to be a natural way of describing requirements.

2.2.2. Characterization of Scenarios

The second finding is about the characterization of the scenarios. There are three main characteristics that are defined for every scenario. Some other characteristics have been identified that are relevant only for some of the scenarios. The main characteristics are presented first.

LENGTH

The first main characteristic is the length of the scenarios. We define the length of a scenario as the number of actions it is a combination of. All scenarios used in the study were short. The minimum length is 2, the maximum length is 7, and the average length is 3. This illustrates the fact that when people use scenarios in describing requirements, they don't use long scenarios showing the global functionality of the system. Rather, they use short scenarios to illustrate specific aspects of the system. It implies that a method for merging several scenarios will be needed.

POSITIVE/NEGATIVE

The second main characteristic is that every scenario is either positive or negative. A *positive scenario* describes a situation that the subject would like to see taking place when using the meeting scheduler. The following scenario is an example of positive scenario. "I think it might be nice if you could send someone a schedule request. So you could say, well, I

1. Wherever excerpts from the transcription are used in this report, they are followed by two numbers referring to the first and last line of the excerpt as they are numbered in the Appendix.

need to meet with you, take a look at my schedule, and schedule yourself in, where there's a free spot. And then maybe you could automatically update someone else's schedule based on that request.⁴²⁻⁴⁴ A *negative scenario* describes a situation that the users would not want to see taking place when using the meeting scheduler. The following scenario is an example of negative scenario. "I don't want to quit the program and then start using phone, e-mail or whatever to talk to people about the meeting and just arrange or schedule it.⁴⁴⁷⁻⁴⁴⁹" The fact that both positive and negative scenarios have been used by the subjects suggests that a method of integration of positive and negative scenarios will be needed.

AUTOMATED/MANUAL

The third main characteristic is that every scenario is either a scenario of using the meeting scheduler (such scenarios will be called "automated") or a scenario of how things are done without the meeting scheduler (such scenarios will be called "manual"). More precisely, an *automated scenario* is a combination of automated actions. An *automated action* is an action that is executed by the system (e.g., "...give me the five best times this meeting can be held.¹⁷⁶"), or by the user while using the system (e.g., "...I can pick which one of those was the best time to hold the meeting.¹⁷⁸⁻¹⁷⁹"). Manual scenarios can also be more precisely defined. A *manual scenario* is a combination of manual actions. A *manual action* is an action that is executed manually, without the help of the system. (e.g., "...you line up all these forms and you try to see where there is a vacant slot for meeting.²³¹⁻²³²") Most scenarios were automated scenarios, only 2 out of 29 were manual ones. It seems to make sense that, when describing requirements for a new application, you would focus on how the application will be used. Nevertheless, it is interesting to note that sometimes manual scenarios are used too, to explain what is now done manually and should be done automatically by the system. The integration of manual scenarios in requirements might be subtle because they do not explicitly describe what the system should do, they describe it implicitly by referring to the current manual situation. Merging automated and manual scenarios might be challenging.

Among the scenarios of the study, there was one that was a combination of automated and manual actions: "...quit the program and then start using phone, e-mail or whatever to talk to people about the meeting and just arrange or schedule it.⁴⁴⁷⁻⁴⁴⁸" Such a scenario should be considered as automated because it says something about the border between the system (automated action: "quit the system") and its environment (manual actions: "using phone, e-mail"). It seems that such mixed scenarios should receive special attention, but the study did not yield enough of them to analyze them. They will not be considered in the present stage of our research.

Other characteristics that were relevant for some but not all scenarios include the following ones:

INITIAL STATE

The description of an initial state that has to be true for the combination of actions to be executed is a characteristic that was part of two scenarios. An example of a scenario with an initial state follows, where the initial state is in italic. "So that *if they don't [have their schedules on line]*, then it can send them a message, you know, and ask them about a particular time or something and then they could answer.³³³⁻³³⁴" The fact that subjects used initial states in some scenarios will justify the definition of an initial state characteristic for modelling scenarios (see Section 3.2).

JUSTIFICATION OF RELEVANCE

A justification of relevance of a scenario was used once, in the following scenario. The justification is in *italic*. "If I have a class on Wednesday at ten thirty, there are thirty people involved, and I cannot even think of trying to move it around, *because of problems of thirty people's schedules.*"¹⁰²⁻¹⁰³" In this specific case, the scenario could be re-expressed by replacing the justification by an action. A possible transformation yields the following description "...If I try to move it [the class] around, thirty people's schedules must be checked." It seems reasonable to assume that it will not always be possible to make such transformations of justifications into additional actions. An explicit characteristic will therefore be available for describing justifications of relevance.

[GENERIC/INSTANTIATED]

Some scenarios were generic, some were instantiated, and yet others were partly generic and partly instantiated. As the border between generic and instantiated was not clear, such a characteristic has not been considered for modelling scenarios.

3. Describing Scenarios

3.1 The KAOS Meta-model

As a model for describing scenarios, we decided to use the KAOS meta-model [Dar93]. The KAOS meta-model is a conceptual model for acquiring and structuring requirements. The conceptual model provides a number of domain-independent abstractions in terms of which requirements are acquired. It is thus a meta-model. Its set of abstractions allows to capture objectives of the system under consideration, constraints that make such objectives operational, agents that control the system's behavior according to such constraints, events that cause the application of actions on entities, etc. The meta-model can be represented as a conceptual graph where nodes represent abstractions (called *meta-concepts*) and edges represent structuring links (called *meta-relationships*). Figure 1 illustrates a portion of this graph that is relevant for modelling scenarios. Examples of meta-concepts are Agent, Action, Goal, Object, Scenario, etc. Examples of meta-relationships are Performs, Responsibility, Combination, Cause, etc.

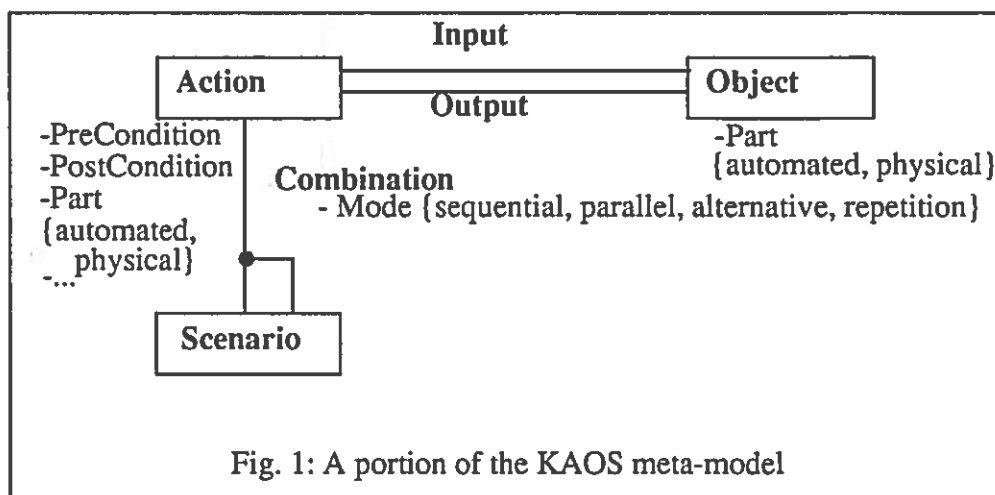


Fig. 1: A portion of the KAOS meta-model

Requirements components are described as domain-specific instances of KAOS meta-concepts, linked by instances of meta-relationships, characterized by instances of meta-attributes. Domain-level concepts are specific to the application domain. For example, for the Library management domain, the “Borrower” concept is an instance of the Agent meta-concept. We will now describe the meta-concepts, meta-attributes and meta-relationships that are relevant to the modelling of scenarios.

Every KAOS meta-concept has an optional meta-attribute called “InformalDef”. InformalDef provides an informal definition of the concept it is characterizing.

An *Object* is a thing of interest. Instances of objects can evolve from state to state. The Object meta-concept has a specific “Part” meta-attribute for describing if it is part of the system (value of Part is “automated”), or part of the system’s environment (value of Part is “physical”).

An *Action* is an input-output relation over objects. The Action meta-concept has several meta-attributes which include PreCondition, PostCondition, TriggerCondition, StopCondition and Duration. Only PreCondition and PostCondition are mandatory. Actions define state transitions. The pair (PreCondition, PostCondition) captures the state transitions produced by application of the action. The Action meta-concept is linked to the Object meta-concept through the Input/Output meta-relationships. Similarly to the Object meta-concept, the Action meta-concept has a “Part” meta-attribute that can take two values: “automated” if the action is executed in the context of the system, and “physical” if the action is executed in the system’s environment.

A *Scenario* is a combination of actions that may be structured into a combination of scenarios. The Scenario meta-concept is linked to itself and to the Action meta-concept through the Combination meta-relationship. This meta-relationship supports four modes for combining actions or scenarios: sequential, parallel, alternative and repetitive. The Scenario meta-concept does not have any specific meta-attributes.

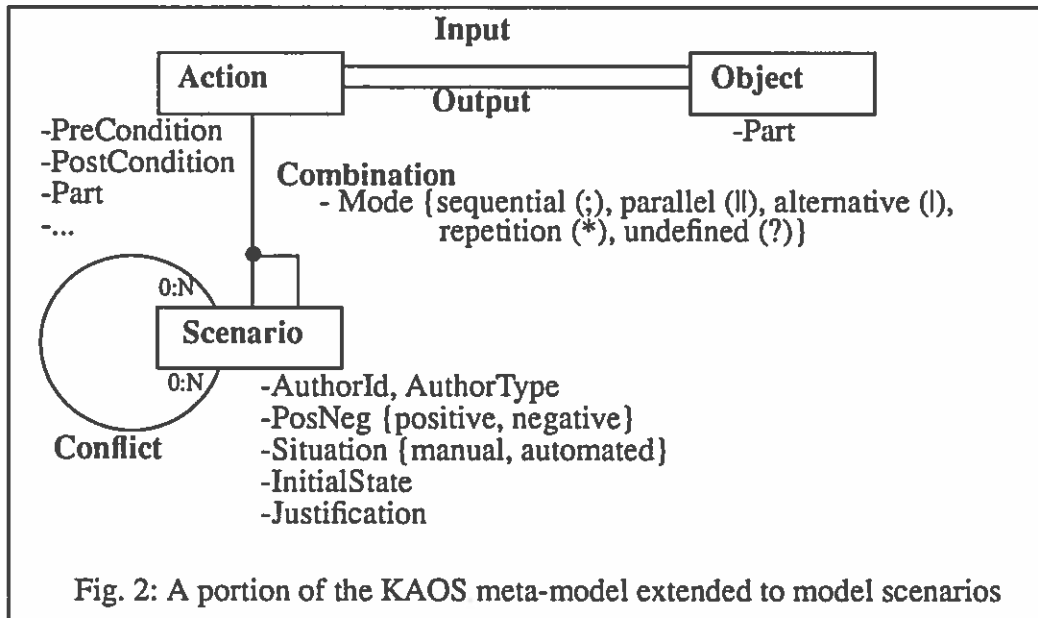
3.2. Extending the KAOS Meta-model

The current KAOS meta-model does not allow us to represent all the useful information about scenarios that was identified during the study (Section 2.2.2). An extension of the KAOS meta-model is proposed below that allows to represent that information. (See Fig. 2.)

One more value has been added to the domain of the Combination mode. This additional value is “undefined”. This value is necessary when a scenario’s author does not want to specify a specific combination mode. In such a situation, leaving the combination mode undefined will allow more flexibility during the merging of scenarios. The symbols following each combination mode value in Fig. 2 are used in the language instead of the corresponding mode names. (See descriptions of scenarios in Section 3.3.)

A reflexive Conflict meta-relationship has been defined on the Scenario meta-concept. It will allow to keep track of conflicting scenarios (see Section 4.3). A scenario can be in conflict with zero, one or several other scenarios.

Several meta-attributes have been added to the Scenario meta-concept to be able to represent the characteristics identified in Section 2.2.2.



- The AuthorId and AuthorType meta-attributes describe the user who elaborated the scenario. The AuthorId and AuthorType meta-attributes are mandatory. Those attributes are necessary to get back to the user if there is a problem during the merging of scenarios. AuthorId takes as value the identification of the user. AuthorType takes as value the type of the user. Examples of types are: management, student, secretary. Types might be useful in the process of merging scenarios because, depending on the type of their author, some scenarios might be considered as more important than others.

- The PosNeg meta-attribute describes if a scenario is positive or negative. The PosNeg meta-attribute is mandatory. It can take two values: "positive" or "negative".

- The Situation meta-attribute describes if a scenario is automated or manual. The Situation meta-attribute is mandatory. It can take two values: "automated" or "manual".

- The InitialState meta-attribute describes an initial state that has to be true for the combination of actions of a scenario to take place. The InitialState meta-attribute is optional. InitialState takes as value a predicate. If "true" is the value of InitialState, it is equivalent to not having any initial state. See examples of initial states in Section 3.3.

- The Justification meta-attribute describes a justification of the relevance of a scenario. The Justification attribute is optional. A justification of relevance can be described in any form: informal text, predicate, etc.

3.3. A Language for Describing Scenarios

As requirements components are described as domain-specific instances of meta-concepts, the meta-model determines the structure of the requirements language. A complete description of the language corresponding to the latest version of the KAOS meta-model can be found in [Dar92]. This language will have to be modified in order to cover the extensions proposed in Section 3.2.

Below are two examples of descriptions of scenarios that focus on the part of the language corresponding to those extensions.

I'd really like it to [...] give me the five best times this meeting can be held, who would be left out of the meeting if we hold it at these times and who would be included in the list. And [...] show me some graphic representation that shows how those meetings would fit at each of those five times, so that I can pick which one of those was the best time to hold the meeting.176-179

Scenario 6

Author Subject#3

PosNeg positive

Situation automated

CombinationOf

((Find5BestMeetingTimes ? FindLeftOutForTimes ?
FindIncludedForTimes ? ShowGraphicForTimes); ChooseTime)

I'd like to have a kind of more automatic mode, where I can specify the people who are involved [...] in the meeting, the time, the date, and maybe the location, and some kind of comment which the other people see [...] I'd like to get feedback, so who is able to join the meeting, who is not able to join the meeting, and who would like to rearrange the meeting.422-427

Scenario 33

Author Subject#8

PosNeg positive

Situation automated

InitialState System.OperatingMode = "automatic"

CombinationOf

((GiveListofPeople ? GiveDate&Time ? GiveLocation ? GiveComment) |
(GiveListofPeople ? GiveDate&Time ? GiveComment));
(GetWhoCanJoin ? GetWhoCannotJoin ? GetWhoWantsReschedule)

4. A Tool for Scenario Acquisition and Merging

4.1. Overall Presentation of TAMS

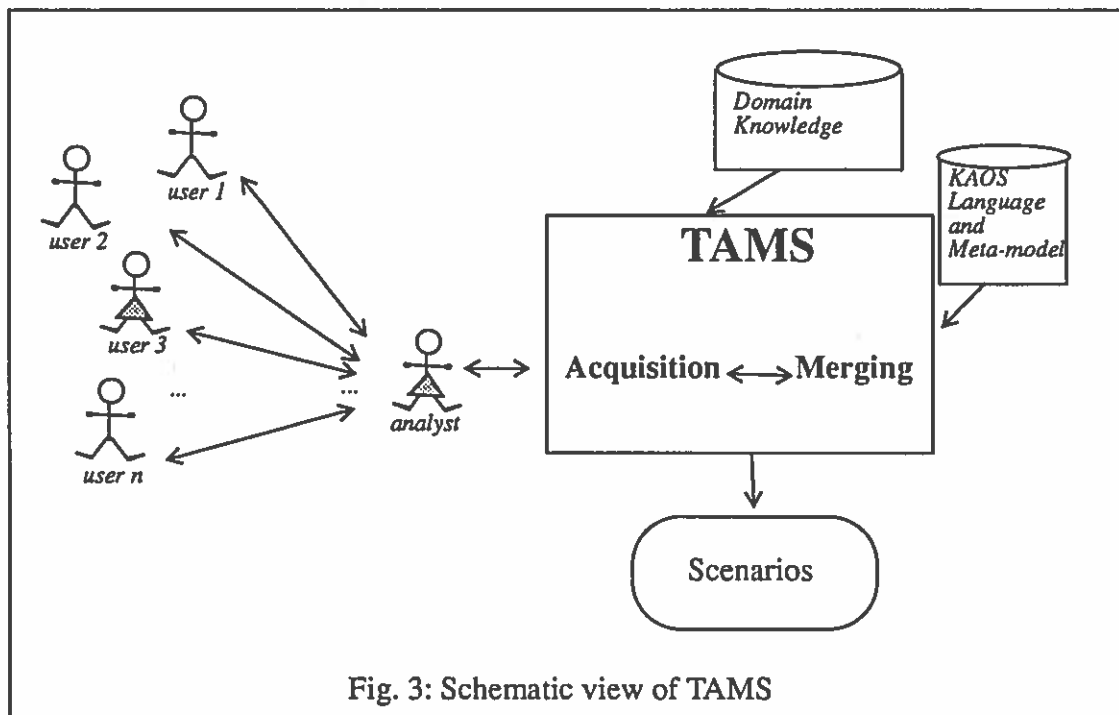
From the findings of the study described earlier (Section 2.2), it seems that a tool could be proposed to help people use scenarios to describe requirements. Such a tool would be used for acquiring and merging scenarios. Our tool is called TAMS (Tool for Acquiring and Merging Scenarios). Before describing what TAMS would do, we will mention what it will *not* do.

- TAMS will not be a tool to be used directly by real users. By "real users", we mean potential end-users of the system being designed. Analysts will use the tool and will interact with real users. In the following, we will refer to users of TAMS as "analysts" and to end-users of the system being designed as "users".

- TAMS will not help acquiring complete requirements. The focus of the tool would be only scenarios. Even if scenarios are not the only part of requirements necessary to make requirements complete, the other information needed would not be acquired by TAMS.

- TAMS will not deal with natural language issues. Natural language will not be considered as the form in which scenarios are described. Instead, users of TAMS will use a structured description language made out of the KAOS language syntax and a palette of actions and predicates to choose from (see more details about this in Section 4.2).

TAMS is schematically presented in Figure 3. The two main ideas behind TAMS are as follows: (i) people spontaneously use scenarios when describing requirements; (ii) all the scenarios acquired for one system's requirements need to be merged together. The study presented in Section 2 backs up those two ideas. TAMS will have two main functions: acquire and merge scenarios.



TAMS will be a domain-specific tool because the use of domain knowledge for acquiring and merging scenarios is critical. The application domain we have chosen is personal calendars and meeting scheduling. In a first phase, this domain contains enough interesting features to illustrate the capabilities of TAMS and test its limitations. Later on, a wider domain might be considered. The domain knowledge will include descriptions of typical domain-specific actions, predicates, objects, characteristics, and scenarios.

4.2. Acquiring Scenarios with TAMS

Users will interact with analysts while elaborating their scenarios. In order to avoid natural language issues, analysts will not be allowed to provide TAMS with scenarios written in natural language. Instead, users will choose from a palette of actions and predicates. The palette will contain typical actions of the application domain, and typical predicates for describing initial states of scenarios. The actions will be described according to the KAOS language. Predicates will also be described according to the KAOS language. Analysts will have to know the KAOS language and underlying meta-model in order to help users elaborate their scenarios. Each action and predicate of the palette will have an associated informal description (InformalDef meta-attribute). The pal-

ette is part of the domain knowledge of TAMS.

Two alternatives might be considered for the palette.

(i) The palette is fixed before users start working on a new system, and it cannot be changed while working on the system. After studying the resulting system requirements, the analyst might decide to extend the palette by adding some new actions and/or predicates to the domain knowledge.

(ii) The palette can be customized to the system under consideration by being extended with user-defined actions and/or predicates. Customizing could happen as follows. If a user cannot find an action or predicate s/he needs in the palette, s/he will be allowed to define his/her own action or predicate. User-defined actions/predicates will automatically be part of the palette of the system. They will therefore also be available to other users working on requirements for the same system. As all users working on the same system have access to the same palette, user-defined actions/predicates must have an informal description attached to them so that each user is able to understand the semantics of actions/predicates added by other users. User-defined actions/predicates will not be automatically included in the domain knowledge. They will not automatically be made available for acquiring scenarios about other systems. The analyst would have to decide whether to add them or not in the domain knowledge.

The first alternative will be the one used in our research.

Choosing actions (for the combination) and predicates (for the initial state) from a palette is not enough to describe a scenario. The actions of a scenario should be combined with the available combination modes. The different combination modes of actions will be available in a menu for the user to choose from. The AuthorId and AuthorType characteristics will be automatically given values by TAMS thanks to the login information provided by the user at the beginning of a session with TAMS. The PosNeg and Situation characteristics of scenarios should also be given values. For each of those two characteristics, the user will have to select from a menu one out of two possible values.

We can now summarize what TAMS will need for the acquisition phase: a palette of actions and predicates that are part of the domain knowledge, and a description of the KAOS language and meta-model. Both pieces of knowledge will be presented in a user-friendly way, in order to ease the collaborative work between analysts and users.

4.3. Merging Scenarios with TAMS

As we have seen in the findings from the study, it will be necessary to merge several scenarios in the context of the requirements of a system. Those scenarios may be provided by different users. Also, each user may provide several scenarios. Different problems will have to be taken into account for the merging phase:

(i) There might be different kinds of conflicts: between the actions of scenarios; between the combination modes of scenarios (parallel, sequential, etc.).

There might also be conflicting initial states, but it seems that those would be less problematic than the first two types of conflicts.

How should TAMS get back to the user if there is a conflict detected between one of her/his scenarios and other scenario(s)? Should TAMS ask yes/no questions to the user, or provide the user with counter-scenarios to elicit more information from her/him? Should TAMS try to find com-

promises and submit them to users' approval? Detecting and solving scenario conflicts is a research focus in itself. (see [Rob93].)

(ii) Merging positive and negative scenarios will not be easy. Should all scenarios be made positive before being merged? The merging of positive and negative scenarios could be a research focus in itself.

(iii) Dealing with automated and manual scenarios will also make the merging phase more complex. Should manual scenarios be transformed into equivalent automated scenarios before being merged? The merging of manual and automated scenarios could also be a research focus in itself.

A lot of questions about the merging phase remain open. They will be the focus of future research.

5. Related Work

Related work for TAMS can be found in two different areas, namely (i) the use of scenarios and (ii) the integration of several persons' requirements.

5.1. Scenarios

In the domain of reactive systems, two systems for requirements acquisition based on the use of scenarios have been developed.

Kelly [Kel91] developed an automated acquisition environment (WATSON) that reads and interprets informal natural language scenarios and induces a plausible formal specification. WATSON relies on domain knowledge to correct routine omissions and errors in scenarios, constrain the space of possible scenario generalizations, and plan queries to the user about variant scenarios. Recent follow-on work on WATSON (KITSS) [Kel93] is not involved with requirements acquisition anymore. Rather, it is about cleaning up scenarios from mistakes and operationalizing them for use as tests. There is also a large natural language processing component.

Hall [Hal93] has been working on scenario generalization. The idea is that end-users describe functional requirements of reactive systems as concrete behavior scenarios. An apprentice (ISAT) helps the human developer in acquiring and maintaining a knowledge-based specification consistent with the scenarios. ISAT generalizes concrete scenarios into classes of scenarios. ISAT uses a modified version of EBG to show the user where to focus attention during the requirements acquisition process.

The ARIES Simulation Component (ASC) [Ben 93] relies also on scenarios. ASC supports the formalization of an analyst's focus, and the automated execution of a specification at various levels of abstraction. This allows ASC to execute a specification even if inconsistency, incompleteness and ambiguity are present, while also ensuring that the execution is an accurate predictor of behaviors in the original specification. Scenarios are used among other things to formalize validation questions asking the user if some desirable or undesirable system behavior is or is not present. ASC uses these scenarios to guide an abstraction process to extract from the current specification an executable specification.

5.2. Requirements Integration

Cooperation in elaborating requirements has been addressed in several systems.

Finkelstein [Fin89] has been working on cooperative elaboration of requirements based on viewpoints. A formal model of how specifications can be constructed from multiple viewpoints has been elaborated. The model of specification from multiple viewpoints treats the development of a specification as a dialogue in which the viewpoints negotiate, establish responsibilities and cooperatively construct an overall specification. Tools have been built to support this approach.

Robinson [Rob93] has been working on automated negotiated design integration. A collaborative design tool (Oz) has been built that allows individuals to independently create designs achieving their own goals. Oz then collectively and interactively derives a single unified design using automated negotiation techniques. The selfish interests of individuals are productively used to derive robust collaborative designs through the automated negotiation of their conflicts.

6. Conclusion

We have described a study that was conducted to observe if people use scenarios when describing requirements. All the subjects involved in the study used scenarios. This finding, together with the characteristics of the scenarios provided, has been at the basis of the description of a tool for acquiring and merging scenarios. The tool (TAMS) has been briefly described. A language for describing scenarios has also been described. Open issues have been identified for future research.

Acknowledgments

The author wishes to thank Steve Fickas for productive discussions and useful comments on an earlier draft of this report. Thanks also to Patricia Fierro for preparing transcripts of the acquisition sessions.

References

- [Ben93] K. Benner, "The ARIES Simulation Component (ASC)", to appear in *Proceedings of Knowledge-Based Software Engineering Conference*, 1993.
- [Dar92] A. Dardenne, *KAOS Acquisition Language*, KAOS Report No. 22, Institut d'Informatique, FUNDP-Namur (Belgium), November 1992.
- [Dar93] A. Dardenne, A. van Lamsweerde, S. Fickas, "Goal-directed Requirements Acquisition", *Science of Computer Programming*, No. 20, 1993, pp. 3-50.
- [Fin89] A. Finkelstein, H. Fuks, "Multi-party Specification", *Proceedings 5th International Workshop on Software Specification and Design*, May 1989, pp. 185-195.
- [Hal93] R. Hall, "Validation of Rule-based Reactive Systems by Sound Scenario Generalization", submitted to the *Eighth Knowledge-Based Software Engineering Conference*, 1993.
- [Kel91] V. Kelly, U. Nonnenmann, "Reducing the Complexity of Formal Specification Acquisition", in *Automating Software Design*, M. Lowry, R. McCartney (Eds.), AAAI Press, 1991, pp. 41-64.
- [Kel93] V. Kelly, M. Jones, "KITSS: A Knowledge-Based Translation System for Test Scenarios", to appear in *AAAI*, 1993.
- [Rob93] W. Robinson, *Automated Negotiated Design Integration: Formal Representations and Algorithms for Collaborative Design*, Computer and Information Science Dept, University of Oregon, Eugene, OR 97403, CIS-TR-93-10, April 1993.

Appendix

Acquisition of Requirements for a Meeting Scheduler - Transcription of Acquisition Sessions

All subjects were members (graduate students, faculty or staff) of the Computer and Information Science Department at the University of Oregon. The acquisition sessions have been held in April 1993. For each of the acquisition sessions, the subject was presented with a hypothetical situation. This situation was that the CIS department was considering implementing a meeting scheduler system and making it available to everybody in the department. The subject would be a potential user of the meeting scheduler. The subject was asked, as a potential user, to explain what kind of meeting scheduler he/she would like to have at the Department. Some of the subjects took some time to think about what their requirements would be for a meeting scheduler before starting the acquisition session, some even took notes during that time. However, most of the subjects started describing their requirements right away. I tried to remain as passive as possible during the acquisition sessions, letting them talk without asking specific questions.

Subject #1

(Note: this subject took some time to think about the problem and wrote notes before starting the acquisition session.)

1 Anne: What kind of meeting scheduler you would be interested in the context of that study. That's what we are
2 interested in.

3 Subject1: I have taken a four step approach to answering that question and I'll outline the approach briefly before I
4 begin delving into the specifics of each step. The first step is to outline what I see as the goals of the system. You have
5 to outline why you are developing the system in the first place, and what is your general idea of what it's going to do,
6 and maybe why you need it. And then I have been heavily influenced by the domain of human-computer interactions
7 and more specifically a process that's been outlined in that domain for developing software. It's kind of like a life-cycle,
8 but it has a lot to do with, ah, acquiring the requirements of the software, how we can go about doing that. And that
9 involves three steps. The first step is to specify the functionality of the software, clearly. Not how it is we are going to
10 accomplish that functionality, but just what is the functionality of the software. The second is how we are going to
11 present that functionality. And the third point deals with the specifics of how users achieve the functionality specified
12 in the first part of this approach. So I figured that these four steps could help us to come up with some kind of document
13 that specifies the software, you know, in a way that could be implemented. And I am going to put in a few plugs during
14 certain steps for usability testing, because I think that usability testing is absolutely essential to any kind of
15 requirements acquisition, because a lot of times, you don't know what the requirements should be, until you actually
16 ask some real users. And by real users I don't mean computer scientists who are going to be designing this, but rather,
17 in this case, probably the secretaries would be good real users, although we probably would be users too, so we
18 probably could get by.

19 A: Yes, that was the idea, it was to have you be a user, a potential user too.

20 S1: So I am just going to stipulate somewhat arbitrarily the goals of the system, just from my perspective, since this is
21 pretty open-ended. I envision the scheduling system as a system that provides a centralized database of all department
22 members' schedules. And this database should serve as both a resource for other department members who are
23 interested in other people's schedules and as a way to communicate to other members. Perhaps if you are interested in
24 scheduling something you can drop them something similar to an e-mail message, only within the framework of this
25 scheduling system, to tell them, well, I see you have a spot open between ten and eleven, for example, how about, it's
26 looking sunny out, would you like to play some frisbee, or something. So that, I kind of see that as a general, a good
27 goal for the system, a centralized database, so that I will state that as my concise goal for the system. And also, I mean,
28 since schedules are by nature dynamic entities it should facilitate the easy changing of schedules so if your schedule
29 changed often or might change from day to day, there should be an easy way for you to update your new schedule
30 based on, you know, your current situation. So that would be probably a good goal. I am just stating that goal, on the

31 basis of my observations that schedules are by nature dynamic. So the basic functionality, I'll move to step two, as I
32 see it, would be the ability to add, remove, and update your schedule easily. As I say, there probably would be some
33 kind of template for specifying a schedule, and the template would make it easier for the system to store the
34 information in a fashion that could be, perhaps, easily retrieved or, so that, if you are interested in automating it at some
35 point, if it's stored in an orderly fashion, you know, with some kind of template, that would make that easier. But I am
36 jumping ahead of myself, as all computer scientists, always jump to implementation before we get... and that's a
37 mistake I think. All right, so the second piece of functionality, I need to be precise about, is to inquire about someone's
38 schedule so you should be able to call up someone's schedule, and perhaps there, some kind of system of security can
39 be built in, so that if a professor wanted to make his or her schedule private, maybe you could do that. Although I think
40 that's probably, I mean, if you don't want to share your schedule, you probably shouldn't use the system in the first
41 place. I think, the whole point is that it be a resource, so, ah. Inquiring about someone's schedule, ok. And then, as I
42 mentioned in my goal, I think it might be nice if you could send someone a schedule request, so you could say, well,
43 I need to meet with you, take a look at my schedule, and schedule yourself in, where there's a free spot, and then maybe
44 you could automatically update someone else's schedule based on that request. But see, I think you need some kind of
45 system of checks and balances there, so that people aren't scheduling themselves in randomly, you know, and by
46 surprise. I mean, there need to be some kind of, you need to grant that person permission to schedule yourself in, I
47 think, I don't know maybe not, but it's something to think about anyway. Ok. And then I think that there is some
48 element of an official schedule here. There are many departmental things, that would lend themselves to display in the
49 system, like the class schedules, the room schedules, and I'm calling that official information. It would be nice to be
50 able to have access to that, and those kinds of schedules change often too, and there could be someone in the office
51 responsible for updating the official schedules. And I think that this component of the system would lend itself past to
52 automation. There are some kind of graph-coloring algorithms that I am aware of, I don't know the details, but you can
53 do some kind of automated scheduling, especially in a domain that's as orderly as this. I mean, you have x number of
54 rooms, you have times, you know that these classes can meet, it's limited in such a way that I think you could do it
55 through a computer algorithm. And that might be nice. In fact, I'd be interested in hearing how it is done presently. I
56 know there are very sophisticated scheduling systems out there. The Major League of Baseball, for example, has a
57 system that schedules all the games for twenty-eight teams over a hundred sixty-two games each. There are lots of
58 constraints on the system. For humans to undertake something like that, would be unwieldy I would imagine. So if a
59 computer can do that, the computer could probably manage the ten or so, fifteen or so classes that we have. That's a
60 limited enough domain. As far as, I thought about scheduling, automated scheduling, of something other than classes.
61 Like, I don't know, you could maybe give it some kind of functional specification, some kind of specification of how
62 you could put some constraints on your schedule. Maybe you could cook up some kind of schedule based on that list
63 of constraints. I think that that kind of scheduling, at least in my mind right now, is too amorphous to specify precisely.
64 And I think you probably should talk to more users about ideas they have and I think that maybe some of those would
65 come out. There is some potential there. But at this point, I see that as bagging the question too much, so I'll hold off
66 on that. So that's the functional specification of the scheduling system. Now I'll move to the second part: presentation.
67 This is kind of hard to do without documents, more specifically a graphical editor. I like to cook up screen mock-ups
68 of what... how these will be presented, and more importantly to show those screen mock-ups to users, before you
69 actually start any kind of implementation, or cook them up in Hypercard or something. And just ask users, whether
70 that presentation makes sense based on the functionality that we are interested in. I assume some guidelines, I'm not
71 fond of guidelines, but I think, in the past, past systems have demonstrated that things like meaningful metaphors,
72 direct manipulation, any coherent conceptual model can contribute greatly to the ease of use of the software. So, I mean
73 there is no hard and fast algorithm for representing what you have, you know, in a way that is going to make sense to
74 the users, but that takes a lot of usability testing, and is a far hard work. But, I take that these are issues that are
75 important and should be considered before any implementation takes place. My personal favorite, I always put in a
76 plug in for the menu-bar style, conceptual models established by Apple. I think that's ingrained in a lot of people,
77 especially people in the department, I would imagine. So I guess, if I had to give some kind of specification of
78 presentation of the interface, right now I would say, you know, you have the menu bar with, perhaps a menu for
79 updating your own schedule, for browsing, you know. And, I think wherever possible there should be, you know, lots
80 of form-filling. You have a schedule with a kind of template schedule, and you can fill it in. And you know, using
81 direct manipulation techniques wherever possible would be desirable. Although one thing that we might have to
82 consider here, is that we are going to be dealing with many different kinds of computers probably, I mean X-terms are
83 probably the most frequent, but... then again I am jumping ahead of myself. So I'll refrain. All right, so let's move to
84 the third component of this specification, which is how users specify the functionality, in other words, what are the

85 meaningful action steps for each piece of functionality that I specified earlier. And see you get into that, and one thing
86 you could do is draw a, what are they called, augmented transition networks, which show the states, the possible states
87 of the software, and transitions among states based on mouse clicks, and menu selections. It's really tedious, and,
88 cumbersome, but I think it could really give you a better feel for, I mean it's pretty..., I think when it is done you can
89 pretty much take that and say "oh!, this is probably what... how we are going to have to..." I mean this is a good
90 document to work off, when we are implementing the system. So documents like that can put people in an excellent
91 position to implement the system as the users have originally specified it, as I see it. I tried this before in a user-
92 interface project and it worked reasonably well. So, at this point, we might be ready to consider implementation issues,
93 such as efficient data structures, what graphics platform we are going to use, how long are we going to need to develop
94 it, if that's of interest to this department. Although it doesn't seem to be... It looks pretty casual when it comes to time
95 frames. I mean, in the industry that would be very important, but here probably not. So, that's my approach if you have
96 any questions.

97 A: Well, I was trying to be really on the passive side, so that I would just let you talk. I sure would have questions, if
98 I was really interested in having your real requirements for a meeting scheduler, but as I was more interested in the
99 way you would describe it, that's just fine. So we can stop here.

Subject #2

(Note: this subject started describing the requirements right away.)

100 Subject2: The basic dichotomy, I think, in my planning, that would also have bearing on a meeting scheduler, namely,
101 that there are some hard occupancies, so to speak, that I cannot move the time, and that there are soft ones. So, hard
102 ones, hard timing, hard schedules, include my classes. If I have a class on Wednesday at ten thirty, there are thirty
103 people involved, and I cannot even think of trying to move it around, because of problems of thirty people's schedules.
104 The soft schedule is that of, say, my exercise. I like to exercise every lunch hour, but if I cannot, not a big deal, right?
105 I can live without it, with missing a day of exercise. So, this is major division, what I can change, what I cannot change.
106 So, I am getting a couple or three requests for blackout, or blackout times that we cannot meet. And that's tough, you
107 know, you get a weekly schedule, and from, say, Campus Planning Committee, you get "tell me when you cannot
108 meet". So, when I really cannot meet is during those hard schedule hours. But, for example, before every class, I have
109 to, I need at least thirty minutes, but preferably an hour for preparing, and even though I know what I'll be talking
110 about, I need preparation to load, to charge up for the lecture, so it's thirty minutes, but if I want to be very well
111 prepared, it takes an hour. So there is this soft margin before the class, for example. And then we have departmental
112 colloquia, every Thursday, but not every Thursday. So the question is how do I deal with those exceptions, that are not
113 predictable, because right now I know that there will be colloquium this Thursday, and probably no colloquium next
114 Thursday, but beyond that, I know that there is a potentiality of having a colloquium on Thursday. So that's, if I was
115 designing the best scheduler, a good scheduler should take under consideration the different degrees of definiteness of
116 events that happen. And also some scale of my willingness to change them, because even though I can skip a day of
117 exercise I'd do it rather unwillingly. So my perception of events that will happen, or are to be scheduled, and the events'
118 probability of happening, that's an important issue in the scheduler. And then, but of course, as with any such ranking,
119 I think it's difficult, of course, because different people have different scales. Even if you say, on a scale from zero to
120 ten tell me how much willing you are to change a particular timing of a particular event, on your past schedule. Because
121 what we think of is some background scheduling for eleven weeks of the term, right? And, on top of that, you put other
122 things. Then for some people five might be "ok I am willing to change a little", five might be already unacceptable, "I
123 put five but I am not going to change it", right?

124 A: Yeah, that's true. Well, we could replace numbers by values, you know like explicit values or "I won't change", or
125 "I might change", or... That might be a solution for that. Oh, I should be passive.

126 S2: No, actually that's right. I recall we had some other collective decision. So we made it explicit what zero, one, two
127 or whatever numeric value meant. Ok I think that's all at the moment.

128 A: Ok. How would you deal with a personal... like you would somehow need to have people's personal schedules
129 accessible to that meeting scheduler system. So that...

130 S2: So, that's what more or less Z and Y are doing, they are collecting all those faculty and GTF schedules. I am not
131 quite sure, I have all this problems with students who disregard completely the office hours. So lately I decided not to
132 hold office hours. If a student wants to meet with me, he or she sends me an e-mail inquiring for a time. I am around
133 most of the time, I don't like to be interrupted, but I even more than that hate to be sitting and waiting, and be stood
134 up. So, somehow people have to be made aware that if we go into the effort and expense of scheduling things, then
135 behavior should be appropriate to the schedule, and that's true not only for students, that I spoke again, but for faculty,
136 the meeting is at twelve thirty, then everybody should be there at twelve thirty promptly, so that we can go on, and not
137 slip for fifteen minutes, until everybody arrives. And if you are not going to arrive, let it be known that you'll be late.
138 But...

139 A: It should start, still.

140 S2: Right. And actually that's something about meetings, that applies both to the beginning and end of a meeting. I can
141 be at a meeting at two o'clock, but at two fifteen I have to be somewhere else. I cannot stay. So if the meeting doesn't
142 come to a conclusion. Well very much like yesterday's class, well I should have stopped at eleven twenty, because
143 people have a rather fixed schedule. And I'm trying, I am making an effort to do it, but yesterday I was kind of free-
144 flowing, you know what free-flowing is like, so it went over.

145 A: Yeah, it's not easy when you are close to the end, you think ok, because I've seen with my own classes, where I still
146 have like two sentences to say, and I think, ok, just one additional minute, and it takes me another five minutes.

147 S2: Because eleven twenty, eleven twenty-five, it's a lot of difference if you have to make another meeting, another
148 schedule.

149 A: Ok. I wasn't going to ask you any specific questions, or anything, I was just going to let you talk, so I guess that's...
150 If you don't think about anything else you would like to add, then that would be it.

151 S2: Well, let me think for a moment. Here again, different people have different ways, there are optimists, and
152 pessimists. Optimists always make a meeting too early and optimists, are always late to the meeting, because they
153 think, oh that meeting will end up, oh my class will end up at eleven twenty, so I will be able to be here at eleven twenty
154 five, like... And then it rains, and you have to put your rain jacket and rain pants on to get on a bike from one end of
155 campus to another. So that has to be made explicit as well, that people should plan exactly, not for the optimists to
156 make even for the worst case.

157 A: Yes, that's true.

158 S2: But that's more of the psychology of the meetings, than of the scheduling.

159 A: Yeah, you know, it's what you do once it has been scheduled.

160 S2: Ok.

Subject #3

(Note: this subject started describing the requirements right away.)

161 Subject3: Ok, first of all, for a meeting scheduler to be useful for me, it absolutely has to run under the Xwindow,
162 Xsystem, because that's what I use for most of my work during the day. In addition if it had a textual interface for times
163 when I need not have an Xterminal available to me, that would be a big plus. Other kinds of interface, MacIntosh is
164 completely unimportant to me, I would never use them. I want it to be reasonably user-friendly, you know, I am
165 obviously capable of operating command-line interfaces, and weird text and that sort of thing, but I don't think that I
166 would use such a tool very much, because it's just as easy for me to schedule my meetings with a text editor. Instead I
167 would like it to have a calendar sort of display with buttons and point-and-click kinds of interface. As far as what it

168 needs to do to be useful to me, first of all, it needs to give me as clear as possible a view of, not just my schedule, but
169 of the schedules of, you know, everybody, at least in terms of what times are available, for everybody in the
170 department. I want to be able to try and place meetings. I expect a meeting scheduler to help me place meetings, but I
171 want to be able to try and place meetings in times that look good. And, I mean, as much as other people in the
172 department are willing to, I want the scheduler to help me you know, as much as they will let me, I want them to tell
173 me what their conflicts are, so that I know what I am scheduling around, kind of... As far as algorithm software that
174 would go into such a scheduler, the one thing that I really want it to be able to do, is some kind of best-fit kind of thing.
175 The one part of meeting scheduler that is hard and that a piece of software does a lot better, is trying a lot of
176 possibilities, and I'd really like it to, you know, give me the five best times this meeting can be held, who would be left
177 out of the meeting if we hold it at these times and who would be included in the list. And, you know, show me some
178 graphic representation that shows how those meetings would fit at each of those five times, so that I can pick which
179 one of those was the best time to hold the meeting, you know, by finally overwriting it. It has to be completely
180 overwritable, it mustn't ever force me as a user to do anything that I wouldn't be willing to do, if you have to ignore it
181 part of the time, and use a piece of paper, then it is easier to use a piece of paper altogether, so it has to be completely
182 flexible. Ah. It would be nice, I guess as a user I am not sure, I mean, I have a pretty specific picture in mind. There
183 are some things I have to be able to customize, I have to be able, for instance, to make use of large fonts because my
184 vision is not all that good. And it would be very, very nice if I could customize it to do input the way I want it to,
185 because I'm accustomed to doing macros and things like that. It would be nice if it interacted with the e-mail system,
186 we use e-mail a lot for meeting scheduling, and just for office communications. It would be nice if there was, some
187 way that I could, I don't know, run an e-mail message through it, or have it generate e-mail, either of those things. It is
188 hard for me to say exactly how it would work. But either of those would be nice features, if some good way could be
189 thought of for them to work. Ah. Let's see anything else. As a user, as a sophisticated user it would be nice if it were
190 implemented in such a way that I could play with its insides. I would hate to have it be a sort of a single piece of C
191 code, that I couldn't read or understand. There might be times where I would like to actually build my own modified
192 meeting scheduler, or whatever. So I mean that kind of code reusability would actually be useful to me as a user of the
193 system. Those are probably the main requirements that I'd have. I'm trying to think if there's anything else that might
194 be extremely important. I guess, I mean, the other thing obviously, this isn't a requirement of the software per se, for
195 it to be very useful, other people besides me in the department really need to be willing to use it. I mean, as a
196 personalized scheduler, there again I mean, the text editor works. I mean for it to be a really big advantage it has to be
197 that I'm interacting not just with a piece of software, but with the department that uses the scheduler's schedules.
198 Although, to retract part of that, just have any piece of calendar interface would be a step in the right direction. I have
199 been lately struggling with the fact that I miss a lot of meetings and things just because I forgot about them, because
200 there really isn't a general purpose calendar software, so, well, I guess there would be an incremental advantage even
201 if it was just a personal scheduler. I'd much rather have the group scheduler. I can't think of too much else, that it really
202 needs to be, obviously it needs to be robust, and fairly intuitive to use, but that's true of any pieces of software. As far
203 as specific meeting scheduler things, if it met that set of constraints, I guess I would be pretty happy with it. I guess
204 actually one thing that I have thought about, it would be nice if it could present once it had most of the scheduling filled
205 in, at least for the stuff that was recurrent and that sort of thing, if it could put up some kind of, print out some kind of
206 bulletin board style presentation, that we could actually put up physically somewhere where people who don't use a
207 computer could find out when meetings are and that sort of things. Yes, something like that would be a feature that
208 doesn't seem to be very difficult to do, I don't know. It's not a primary requirement. As far as primary requirements I
209 think I am about done. I mentioned recurrent schedules, so that's something I should probably say something about. It
210 does want to be able to do recurrent scheduling, in some extent, although, again that is not a primary requirement for
211 me if I had to schedule every meeting individually, it would still be a very useful tool to me. If I had to actually to go
212 through the act of scheduling a meeting every week, I think it would still be all right.

213 A: Do you want to deal with also reservation of rooms for meetings?

214 S3: No really not, I think it's... you know I think, there is a lot of things that it could deal with, but I mean, as far as
215 what it has to deal with it can just get blocks of time when everybody could meet, that's enough for me. That's the... I
216 mean that's the... and I'm one of these people who likes the Unix, as a user likes, you know, the Unixy philosophy of
217 small separate tools, as much as possible, for small separate jobs. If there was some kind of room scheduler, I mean,
218 that as far as I am concerned, could be an independent piece of software. I think I'd like the meeting scheduler to
219 schedule meetings. OK. I think that's about it.

Subject #4

(Note: this subject started describing the requirements right away.)

- 220 Anne: You can describe what kind of meeting scheduler you would be happy to use in the department.
- 221 Subject4: What I see that it could work for benefit would be like... ah, not so much the colloquium because everyone,
222 pretty well, has it ingrained in their heads, that the colloquium is Thursday at 3:30, and you know, and the faculty
223 meetings are really consistent. But maybe something like this would be good for... ah, like scheduling meetings for the
224 women in computing, which everybody's schedule is really, really different you know, and so finding that good time
225 for everyone, I could see where a system like that would be beneficial. Here, for something of that purpose where
226 instead of pulling everyone's individual file, you know, looking at this, doing it manually this would take all that time
227 out from... But you know what I mean.
- 228 A: Yes, I see. So, it would save, it would do the same thing, but you wouldn't have to do it manually, by looking through
229 the schedules, you would have all the schedules already in the computer.
- 230 S4: Yes. What we have done in the past is everyone came, or we sent out a message to the women to come and fill out
231 this form in the office so we know when, what your schedule is. And then you know, then you line up all these forms
232 and you try to see where there is a vacant slot for meeting. Well a system like that would avoid all this unnecessary
233 paper work. I think the time concept is quickened greatly.
- 234 A: So that's good, yes that would be a good use of the system
- 235 S4: Yes. I don't deal much with scheduling things. So it is like... I am thinking, I can see that this system would be
236 beneficial, but maybe I am not always clear where, in the department. Ah... So I don't know if something like this, like
237 scheduling classes, but then that's different. Well another thing I guess is if one were to implement this, then there
238 would have to be people to input all the information, so like, ah... "Do you have to set up this criterion?", you know or
239 the database of what information you wanted and how... See this is all kind of a new area to me, it's like spreadsheets
240 or something, is that what you are referring to also?
- 241 A: Yeah, I guess that would be something similar to a spreadsheet, but I guess we would have a different way of
242 showing information on the screen which would look more like a personal calendar, with days in the week and times
243 automatically there, so you wouldn't have to type from eight thirty to nine, you would already have all the slots on the
244 screen, but empty. You would type whatever you want to schedule, like classes, in those empty slots. But...Yeah I
245 guess...But I shouldn't be talking.
- 246 S4: Yes you should. Ah! Because I need input from you and then I could maybe think of more things that I would like
247 to say.
- 248 A: I guess what we would need to have those faculty members and GTF's inputting their own information too, like it
249 would be less centralized, instead of writing their schedules on a piece of paper.
- 250 S4: Everybody would have access. That's handy. That would be a time saver if everyone just took a small segment of
251 their own time, to put on their own schedule, like five or ten minutes and then there would be...
- 252 A: And then, see, they could update also.
- 253 S4: You could even do that, yeah, with undergraduates students and graduate students, yeah. If you want to organize
254 a meeting again, this would even be good for the ACM meetings where it's primarily undergraduates, there are some
255 graduates, but that would be beneficial for that type of meeting too, when you are trying to gather students. I think it
256 is harder to gather students at times, than faculty. Faculty are more visible and can communicate. Students are kind of
257 scattered here and there, so that would be good. Maybe eliminate trying if they could access a file or access, you know

258 where things might be posted.

259 A: Well we could have it done by e-mail. I don't know how the system would be implemented, but, it could just send
260 their information by e-mail, and the meeting scheduler, and e-mail to that meeting scheduler, you know, which would
261 receive the message, and automatically update. I don't know or, you know, just run the system on there if there were
262 an application available, like Word or what would be available.

263 S4: So, are you going to implement the system or are you going to create one, like an experiment?

264 A: No, in fact, I am only using this as an example, to extract a description from people and not really, my goal is not
265 to implement the system. It's more to analyze the way people describe what they want the system to be doing. So, too
266 bad, you won't have such a system.

267 S4: Ha! No, I think it has lots of good ideas.... Yeah I am trying to think what else might be... what other information
268 we could extract or put in. I don't know, I mean this is all kind of new to me. I am not a computer science-minded
269 person. You know, I think about a different way of doing things.

Subject #5

(Note: this subject started describing the requirements right away.)

270 A: You can go ahead and describe what kind of system you would use.

271 S5: Well I was thinking that a user could define, well maybe if each user define their own calendar, they could define
272 aspects of it that could be public, so that if there was someone else who is curious of what their calendar was, they
273 might, or if there were somethings that they would rather... I don't know, it's hard to imagine what they would want to
274 keep private, maybe... Well I suppose you could have two different calendars, you could have one that you would use,
275 that could be read by other people who were curious of your calendar and maybe if you wanted you could specify
276 another calendar, just for your own use, that would be private. As far as,... like as if you were interested in making a
277 meeting, I think the important thing would be that you need to be able to specify who the other people you are interested
278 in meeting with. And then maybe you could look through their calendars individually, but then also the system would
279 have some way of being able to find the free spaces that were possible for everyone, as well as giving you some more
280 detailed information like if... I think if in each individual calendar you could specify certain priorities, like if certain
281 times I could be scheduled to play tennis or just go to a movie, or something, things that were more flexible, that you
282 were willing to change, that when someone was trying to make a group meeting, that maybe those times would also
283 show up. Or times were that was more possible that someone could change their plan. That would be kind of some
284 suggestions as far as, you know, what if there were no times at all that were totally free, you would at least know the
285 times where that's possible that some people could make some space. Let's see, ah. It might be nice to have some
286 automatic like mail facility that was included, so that you could maybe just, it is probably not so important, but if you
287 were trying to arrange things, maybe even a conferencing possibility, just to help people schedule things. If you need
288 to talk to a few different people, and you would not have to send mail one way to everyone, that would be nice.

289 A: So you would have... you mean that the system would be responsible for sending mail to participants?

290 S5: Yeah. Or have it like a... yeah that could be part of it. But especially the conferencing possibility where a bunch of
291 people could be talking at the same time. So that... Yeah, you would have their scheduling information and everyone
292 can sort of make some agreement, and everyone would be able to see what everyone else's schedule was, so...

293 A: So you wouldn't have the system suggest a meeting time, you would have people among themselves having their
294 own schedules accessible, everybody's schedule accessible to decide together.

295 S5: Well that would be more of a case where there wasn't an obvious time where people could meet, I think it would
296 be good that the system, I mean that would be suggested by the system, but if there were no... You know, it would give
297 a list of suggestions and priorities according to you know, if there was an absolutely free time among five people, then

298 that would be, maybe that would be a certain color or something like that if you could give different colors to different
299 blocks. And then, so I was just thinking if there were... So if it meant that the people who needed to meet had to talk
300 to each other to figure out what was the best time, because someone might have to change their schedule, or even to
301 agree upon the right time, or not so much of that, but it would be nice to have a way of immediately communicating
302 in conferencing. Actually one idea would be to have a way, like if you were trying to schedule a meeting, this could
303 sort of be part of... but also if you wanted to suggest a time for everyone to meet that would sort of show up in the
304 schedule, something that they could... hum, I don't know.

305 A: So, you would go ahead and update all people's schedules?

306 S5: Yeah, something like that.

307 A: That would show on the schedules.

308 S5: Yeah, that way you could like sort of reserve a little block on their schedule so that the next time that they were
309 looking to make plans, they could see, "ah, here's a possible time, someone wants to meet with me at this time", you
310 know, find some way of notifying them. I mean it could be through mail too, but it would make sense also to have it
311 appear on their schedule. I guess there would be a question of how you might associate, if you were making your own
312 schedule, you might associate some sort of explanation or some text with each of the blocks of time you have, like
313 describing what you're doing, or some reminders and things like that. So that would be nice when you are making your
314 own schedule, especially you could, that could be something too you could specify how public you want that to be, it
315 might be helpful for people who are making schedules to be able to see, I mean to find out what exactly you are doing
316 during certain times, to get an idea of how possibly...

317 A: If you think that that's all that you would have to say, that's fine, but if you still have things that you would like to
318 say, that's fine too. I don't want to have to keep you here if you don't..

319 S5: Let me think for just a second, about some of the other stuff, ah... I guess that's fine.

Subject #6

(Note: this subject started describing the requirements right away.)

320 S6: Ok. Well, since I recently have had to schedule some meetings that were difficult to schedule, I have thought of
321 some things, I have been thinking of it in those terms. So that... So, I guess the main thing is that it's hard to get every-
322 body's constraints so to find out everything and I have had then to end up sending out several sets of mail messages,
323 so that... So I think the main thing I would want is to avoid that problem so I could just set it up once and then be done
324 and I wouldn't have to keep sending out notes to everybody. So what happened was that somebody changed then I had
325 to send it out again, or... Basically I'd have to set up a tentative time and see if everybody could do that. And so that,
326 I have to get their responses back and send out something again. So it seems like it would be better to have something
327 where either it could get their constraints from their calendar automatically, or at least get their, maybe get their con-
328 straints back in an e-mail message or something that the program could understand and figure it out. So that's the
329 main thing. Another thing is, I think that it needs to be as successful as possible, both to me and to other people
330 because you can't like... If it runs on a MacIntosh or something like that, then it is not going to do much good, because
331 other people use other systems and so... I think that is an important thing and also I,... I mean it would be nice if it
332 didn't have to... If it worked for people that didn't have their schedules on-line, I'm sure it could do more if they had
333 their schedules on-line, but I think it needs some fall-back thing, so that if they don't, then it can send them a message,
334 you know, and ask them about a particular time or something, and then they could still answer. So I would think that
335 it should still do something, you know, if they don't have their schedule on-line. Let's see... Another good thing that I
336 have seen in the past is if it can't work with a calendar, if it has some way of dealing with recurrent things like some-
337 thing the first Tuesday of every month or whatever. That would be a good thing. And then it would be good if it could
338 interact with the calendar so it could put meetings in the calendar automatically and so if you want to schedule some-
339 thing else in your calendar, it would tell you if you have a meeting coming up or possible meetings, so...

340 A: So you mean even in the process of scheduling a meeting it would send the calendar potential meeting times?

341 S6: No, I meant some... If I am scheduling something else, like I want to take the dog for a walk. If it could say, well,
342 there might be a meeting Tuesday at two o'clock then I would say fine, I'll schedule the dog's walk for another time.
343 So it does seem like there is a lot of interaction between the calendars and scheduling the meetings even if it is not,
344 you know, even if I am not scheduling a meeting, then...

345 A: Ok. So, I don't know, do you want to consider ah... different kinds of meetings? With different scheduling, you
346 know, scheduling strategies depending on the kinds of meetings?

347 S6: Yeah. Some would have higher priorities, I suppose, priorities would be one way of doing it, maybe, or just a way
348 of saying how flexible it is. Some meetings are pretty flexible, and it doesn't matter that much, if you change them. So
349 you want to deal without that. Then some, depending on what kind of equipment you needed like... I guess when I
350 was scheduling these meetings it was a problem in the past couple of weeks. It was a problem because I wanted to get
351 the room too. As it turned out, the room was available most of the time I needed. But if it had been something that
352 wasn't available as much, then it could have been a big problem, because at times, because I need to schedule the
353 room enough in advance so that I would have a good chance of getting it. And if the meeting date was changing, I
354 wouldn't want to have to keep changing the room also. So I could see you know for rooms, or other equipment, that
355 you need to take that into account and it depends on how available that is. Ah... It might be nice if you had a way for
356 it to... well, I suppose you want to put in like groups or something for meetings, like you do for aliases for mail mes-
357 sages. So that if you say I want a meeting with this group, then it already knows whom you are talking about, so you
358 don't have to list a dozen people if it were that many people. The ones that I was setting up were more for two or four
359 people. So it was not a lot of people involved, but even so you don't want to list those, say, four, or five or six people
360 every time.

361 A: Is that all you can think about?... You know that's fine, if that's all you have in mind right now.

362 S6: Yeah. I mean... There's the question of confidentiality which I've heard discussed with respect to meeting sched-
363 ulers before. But, basically, as far as I'm concerned if there's, if I have an easy way to mark things on my calendar,
364 saying that I don't, that I want it confidential, then that's all I would need, I'm not that worried about details or people
365 figuring out what I am doing, or whatever. I guess I would just say well as long as there is an easy way of marking
366 things confidential, then that would be sufficient for that. There's the question of reminders too. I think it might prob-
367 ably be best for... It would probably be good to have it be automatically but I think you might also need to have... let
368 the receiver choose whether they want reminders or not because that could get pretty obnoxious if you didn't need
369 them. And I think that most people that need reminders know that they need them. So they probably would be willing,
370 they could probably say that they need it. It would probably be good if the reminder function had some way of know-
371 ing which dates are more likely to need reminders, like if it is two months in advance, then most people are going to
372 need reminders, unless all they calendars are on-line and they look at them all the time. So, that could make a differ-
373 ence. I think I already mentioned you know where the things might change, so I don't know if there's anything in par-
374 ticular you need to do about it other than... I mean, having some way of dealing with that.

375 A: So you mean if somebody has another constraint coming in, then maybe...

376 S6: Yeah, I think then it would need to change. I think that the main thing is that it would be easier to change if you
377 had the scheduler, but there might be some things you could do, in particular, like... I don't know like having an alter-
378 native day or be able to specify, you know, if this person can't be there, then it is not worth holding the meeting, but if
379 they can, or, you know, if this other person can't be there then we'll just have to go ahead and do it anyhow. So that
380 would probably be a good thing for it do... And then there's a lot of other associated calendar things, that would sort
381 of be related, like being able to see their schedule for the day, and for the week, and being able to see only certain
382 kinds of things that you have on the schedule, you know and that kind of stuff. And there might be some way I don't
383 know, if you are going to do anything to keep records of them, I guess if we put them into an account, it keeps
384 records, but you might want to have a... after it is actually held... Oh, that's probably all I have to say.

Subject #7

(Note: this subject started describing the requirements right away.)

385 A: Ok. So we can start. You can start telling me what you would expect from that meeting... automatic meeting sched-
386 uler system.

387 S7: I think the meeting scheduler is a really big part of a package that the user want or is going to see. I think that...
388 some kind of a daily planner where the user can bring up a calendar, bring up an entire calendar, and put major events
389 on that calendar, bring up days of the calendar, schedule events during certain times. That would be nice. And then in
390 a package like that, it's a personal organization system, it'd also be nice to have like a list of telephone numbers, and
391 maybe some kind of a datasheet, I don't know your wife's dress size, or something. I've used a Fragment Date Planner,
392 which I'm used to work with. I think it is a similar kind of system, it's really nice in the way that it's organized. And
393 also it would be nice to have like a list of things to do, and then a way to take those things to do, and cancel them or
394 check them off as being done, or forward them to a different day, and so on. And then as far as the meeting scheduler
395 goes, it would be kind of a background program that can pull data from everybody's planner and correlate that data
396 and find times that they are available. If this planner is to be used in a network, there should be some notion of mak-
397 ing certain portions of the data private, some portions public. You might want other people to be able to see your cal-
398 endar, but you might not want them to see everything on the calendar, just business-related things, or whatever. But as
399 far as the meeting scheduler portion goes, I suppose it would take a person and a number of user names or something,
400 and then the database system can bring up the records on these people, internally to try to correlate the meeting time
401 you might input, a preferred meeting time or let's say my preferred meeting time is two o'clock, plus or minus a cou-
402 ple of hours, before five o'clock, or something like that, some kind of relational specification of when you want to
403 meet, and tries to find a meeting time. It might also put tasks on your calendar, it might prioritize them, so that if the
404 meeting is important, it makes its priority over... So, that's about it.

Subject #8

(Note: this subject thought about the problem and wrote notes before starting the acquisition session.)

405 A: Ok. So you can start describing what you would expect from the system.

406 S8: Ok. The meeting scheduler... In general, what is especially important is that it is user-friendly, easy to use, what-
407 ever kind of functionality it provides. Another important point, a very general point, is the integration with other pro-
408 grams, so that it can for example, connect it to a database or an e-mail system, so that I can send e-mail directly to
409 people. And it should be flexible to use so that there is not only one way to use it, it should be more like a tool, which
410 might get used in the way I'd like to use it. Ok. A little bit more details of the functionality. In general, it should look
411 like a calendar, probably at the beginning. So that it looks familiar. I'd like to be able to use it as my own calendar so
412 that I can, that I have my own schedule there, so that I am not required to use it only when I want to schedule a meet-
413 ing with other people. So it should be, it should be combined. For meeting schedules, I guess there should be two dif-
414 ferent modes. The first mode is just a pure manual mode, where I'd like to know if some other person I have in mind
415 is available at a specific time, so that I can ask the meeting scheduler could I arrange a meeting at that date and time,
416 with this and this person, for example, without any scheduling actually.

417 A: No, you would give the meeting scheduler the list of participants and then the time you would like the meeting to
418 be?

419 S8: Yeah, and then say if it is possible or not and... So it's just a trial and error proceeding, because I mean if I actually
420 schedule a meeting, or let the scheduler schedule a meeting, then probably other people should be notified that there
421 is something going on and that they might be in your new meeting. At the beginning, maybe I just want to check if
422 there is time for a meeting or not, without getting involved the other people. But on the other hand, yeah, I'd like to
423 have a kind of more automatic mode, where I can specify the people who are involved, I'd like to involve in the meet-
424 ing, the time, the date, and maybe the location, and some kind of comment which the other people see, if they see
425 there is a meeting, so that I can communicate to the people, "this meeting is about that and that, I'd like you to come
426 too", that people know what's going on there. Yeah, and I'd like to get feedback, so who's able to join the meeting,

427 who's not able to join the meeting, and who would like to rearrange the meeting, so to see the current status of one
428 specific meeting. Ok. The presentation of the information is probably an important point, there should be different
429 modes to present the information. So if it is like a calendar, there should be a way to show my schedule for one day,
430 my schedule for one week, for one month, for one year, whatever. And I would like to search easily between, and
431 maybe see different modes, different views at the same time.

432 A: Like a week, and then a date...

433 S8: Yeah, and maybe fix this form, this kind of presentation. So whenever I start the calendar then I have the same
434 presentation of my information, so... A different kind of presenting the information is project. I know there are all
435 these activities belonging to a certain project, I'd like to see what is going on for a special project, so that is... ok this
436 Monday there is a meeting and Tuesday I have to do that, and that, so that I can really easily see the progress for one
437 specific project.

438 A: And see the different deadlines you have.

439 S8: Yeah, for example. yeah. So this might be not the functionality for a meeting scheduler, but more like for a project
440 management tool, but some kind of... Yeah, it is a little bit of this kind of management, that's probably.

441 A: Yeah, and you can have the meeting scheduler being used to know when in the context of this project you are
442 going to be meeting and it has to show the information some way or another so that...

443 S8: Yeah, I'd like also to see, to have the information organized by task. In a project there are a lot of people involved,
444 so you probably can say ok these people belong to that project, so... But a certain task involves maybe a lot of other
445 people I don't know in advance, so they might be, or maybe by keywords, not by task, so then... whenever I attach
446 keywords to some activity, so that I can easily see which activities are going on, when I get the keyword... At the
447 beginning I already mentioned that the integration with other programs is a big issue, because I don't want to quit the
448 program, and then start using phone, e-mail or whatever, to talk to people about the meeting and just arrange or
449 schedule it, but I'd like to... If I see, ok I try to arrange a meeting with some person, then I'd like to be able to mail this
450 person, send e-mail directly, so that there should be some access to a small database of e-mail addresses, phone
451 addresses, whatever. Ok. It's probably important that the software runs in different platforms, so MAc, Sun and all
452 these kinds of computers.

453 A: Yeah, you want to have it accessible to everybody.

454 S8: Yeah. And it should have the same look and feeling on all different computers, no differences, or almost. Another
455 point is [that] it must be easily updatable, for example, if I have on my own personal computer at home, part of my
456 schedule... I'd like to be able to combine these schedules which are handled by the meeting scheduler and by my tool
457 I am using at home, I mean if these are the same tools, that's great. Then maybe if I change something at home, but I
458 am not connected to the network, so there's no way to make this change properly, I'd like to, for example use a disk,
459 or whatever, when I'm here, just to say these are my new activities, my new schedule for the next week, update my
460 schedule, my database on this machine. That's a problem people usually have, if they have notebook portable com-
461 puter and desktop computer.

462 A: Yeah, so you keep the information consistent.

463 S8: Yeah, yeah consistency between the information on different machines. Yeah, let me see. If I don't want to
464 arrange a meeting, or schedule a meeting, but I am on the other side so, somebody wants me to join a meeting, or
465 something like that, it would be a good idea to have... to manage some kind of event list, so that I see ok, the meeting
466 scheduler system got the information that there's a new meeting and I'm invited, so that's one event. I can do what-
467 ever, I can reply ok, I can discard it, ignore it, or whatever. There might be another event that a meeting is not possi-
468 ble, so that the interaction between the meeting scheduler and me is the list. Maybe it is good to have a list of activity
469 events where I have to act in a certain way, so I am not required actually to do something, when the meeting scheduler
470 says me, ok, there's something going on, what do you want, but then I can think, and then come back to the program

471 after a while, but still see the list of all the things I have to do. So then I am not forced to do actually at the moment
472 when the scheduler detects something to act.

473 A: But, would it... isn't there a risk that information would be not relevant anymore, if you check it later?

474 S8: Yeah, that's right, but I guess I wouldn't like, I mean, I see a meeting scheduler like a tool, so that the meeting
475 scheduler doesn't tell me what to do, but what I could do, and if I haven't yet kind of a to-do list.

476 A: Oh, yeah, that would be updated then. If you check it later and it's not relevant anymore, maybe the meeting
477 scheduler system would just remove it from your list.

478 S8: Yeah, or tells me if this is urgent, don't do anything about it until tomorrow at five o'clock, and if I decide ok, if I
479 don't do anything that's my decision, yeah. Yeah I guess that's it, what I had in mind.