Scenario-Based Task Analysis

Kentaro Go¹ and John M. Carroll²
Yamanashi University and Virginia Tech

¹Dept. of Computer Science and Media Engineering, Yamanashi University, 4-3-11 Takeda, Kofu, 400-8511 Japan. Email: kgo@acm.org

²Dept. of Computer Science and Center for Human-Computer Interaction, Virginia Polytechnic Institute and State University, Blacksburg, VA 24060, U.S.A. Email: carroll@cs.vt.edu

This chapter examines the use of scenarios in analysing tasks in human-computer interaction. The scenario-based requirements process employs scenarios—specific stories and examples of past use of existing systems as well as future use of potential systems—as fundamental tools for capturing and articulating tasks and activities. This chapter defines basic terms for the scenario-based requirements process, which include scenarios, claims, scenario-based requirements analysis, task exploration and analysis using scenarios, claims analysis, and scenario exploration. In addition, it proposes a framework for concept and scenario evolution, in which concepts in the abstraction domain and scenarios in the detail domain iteratively evolve. This framework provides another facet of scenario-based requirements analysis. The chapter also illustrates an example of the scenario-based requirements process. The concept of Familyware—communication tools that provide status information about remotely located family members using peripheral displays and devices—is developed through an example process. This example illustrates where scenarios are obtained from and how they evolve. The chapter concludes with a discussion about the characteristics of the scenario-based approach and task analysis.

Keywords: claim, claims analysis, scenario, scenario-based design, scenario-based requirements analysis, tradeoff analysis

Introduction

Scenarios have been used as a powerful design tool. They are used throughout the design process and in various disciplines including human-computer interaction, requirements engineering, object-oriented design, and strategic planning (Go and Carroll 2002; Jarke, Bui, Carroll, 1998). In human-computer interaction, Carroll (1995) collected the early research and practices on scenario-based approach in design. More recently, several practitioners have documented efforts on scenario use in systems design (Beyer and Holtzblatt, 2002; Cooper, 1999; Johnson 2000; McGraw and Harbison, 1997).

Scenarios facilitate design activities to provide a lightweight way of creating and reusing usage situations (Carroll 2000). They are integrated and flexible, use-oriented design representations that are easily developed, shared, and manipulated. They are applicable to many system development activities. They are fundamental design artefacts in human-computer interaction and have several roles throughout the system lifecycle. For example, scenarios facilitate user-designer communication as a vehicle of knowledge. They will treat unforeseen activities by users as if they were tangible design artefacts.
This chapter consists of three parts. The first part describes basic terms used in scenario-based requirements analysis. It specifies scenarios and claims and then briefly explains Rosson and Carroll’s approach (2001). It also expresses a task exploration and analysis approach. The second part, which is the transition between the first part and the third part, illustrates the framework of concept and scenario evolution. It shows an alternative view of the scenario-based requirements analysis to the one described in the first part; to be precise, it models the process of concept evolution in the abstraction domain and scenarios in the detail domain. The final part demonstrates how scenarios can contribute to the activities of task analysis. Basically this chapter describes how we develop ideas for personal communication systems called Familyware (Go, Carroll, and Imamiya 2000). Familyware provides status information about remotely located family members using peripheral displays and devices. At the early design and analysis stage, we first develop a root concept of Familyware; then, we gather usage scenarios from interviews and brainstorming sessions. These activities result in an envisioning scenario, which is analysed later. At the end of this chapter, we discuss the contributions of scenarios to task analysis.

**Scenario-Based Requirements Analysis: Ideas and Concepts**

This section specifies scenarios and claims. It includes a short summary of Scenario-Based Requirements Analysis (SBRA) by Rosson and Carroll (2001). SBRA employs claims analysis as a key technique for analysis. Also, it explains scenario exploration, which is used a structured brainstorming session to produce new ideas.

**Scenario**

A scenario is a description that contains actors, the background information about them and assumptions about their environment, their goals or objectives, and sequences of actions and events. It may include actors’ obstacles, contingencies, and outcomes. In some applications, scenarios may omit one of the elements or express it simply or implicitly.

It is a shared story among various stakeholders in system design. For example, customers or project managers describe their visions in episodes. Users talk about problems they face as they happen. Designers record the rationale of a design in the form of an example and develop mock-ups to illustrate what users should do with the design. Technical writers explain the task of users in a manual and write it up as a story. These are examples of scenarios shared among stakeholders and distributed throughout the design cycle.

Scenarios are expressed in various media and forms. For example, they can be textual narratives, storyboards, video mock-ups, or scripted prototypes. In addition, they may be in informal, semi-formal, or formal notation. A typical example of an informal scenario is a story, a kind of scenario frequently used for envisioning user tasks in human-computer interaction.

The following is an example of a textual narrative scenario; it is an excerpt from Carroll (2000). It envisions ideas about an interface and interaction on a video information system.

*Looking for the fast-forward button*

Walter has been browsing some clips pertaining to the project manager’s views of the lexical network as they developed through the course of the project. One clip in particular seems to drag a bit, and he wonders how to fast forward through the rest of it—perhaps he can just stop the playout?
Claim

A designed artefact may contain various tradeoffs. Tradeoffs describe the pros and cons of an artefact. In contrast, claims are more specific in context than tradeoffs. A claim is a description of tradeoffs relating to specific usability concerns with a given artefact; in other words, it creates an instance of a tradeoff. It articulates the upsides and downsides of artefact usability.

Claims enumerate implicit causal relations in a scenario. They describe tradeoffs instantiated in a scenario context (and in that sense, they explain the scenario). The use of claims and scenarios in combination, therefore, helps designers discuss the consequences of design moves at various levels of analysis. Considering the “looking for the fast-forward button” scenario, for instance, the nature of video data has two sides: it is a very rich, intrinsically appealing medium, but difficult to search, and must be viewed linearly in real time. Articulating this important yet implicit nature together with the scenario enriches the use of scenarios in system design and analysis.

It is useful to provide a format for tradeoffs and claims in order to produce, document, analyse, and reuse them. Carroll and Rosson (1992) suggest the following practical form:

{Some design feature}
+ causes {desirable consequences}
- causes {undesirable consequences}

For example, the claims about video information related to the “looking for the fast-forward button” scenario can be described as:

Video Information claim: video information
+ is a very rich medium,
+ is an intrinsically appealing medium,
- is difficult to search,
- must be viewed linearly in real time.

This form of claims is useful for analysts because it naturally indicates what is missing in the analysis. For example, if there are several downsides listed in a claim about an artefact without any upsides, then analysts should consider its upsides.

The activity of investigating the claims of a scenario is called claims analysis. This is an analytic evaluation method involving the identification of scenario features that have significant positive or negative usability consequences. Articulating claims of technology in scenarios (i.e., systematically enumerating its potential tradeoffs) provides fair analysis of the pros and cons of its consequences because usability experts may have a tendency to produce and address the downsides of an artefact.

Scenario-based design

Scenario-Based Design (SBD) uses scenarios (and thereby claims) as a central representation throughout the entire system lifecycle. SBD has three key characteristics: (1) it is a lifecycle methodology; (2) it is strongly oriented toward inquiry (in Pott’s sense, it is intended for “a series of questions and answers designed to pinpoint where information needs come from and when”, Potts, Takahashi, and Anton 1994); and (3) it can be employed in a wide variety of ways.
SBD covers everything from requirements and visions through to summative evaluation; most task analysis methods are employed at a particular point in system development, and neither leverage nor are leveraged by any other representation or technique employed anywhere else in the system development lifecycle. This lack of integration with lifecycle development activities could be one reason that these other task analysis methods have had little impact (as Diaper (2002) admits in his review of Making Use) and are adopted only in organisations that employ formal waterfall methods (like military contractors or insurance companies).

SBD is strongly oriented to inquiry. Most task analysis methods take some sort of specification as a given and then further articulate it, and perhaps refactor it. However, this is mild stuff if one is interested in discovering insights into human activity, radically new ways of doing familiar things, or entirely new types of things to do. SBD includes and addresses these other, more creative, concerns. It may worry a person with very structured task analytic perspective that SBD helps to generate all sorts of novel activity concepts, because such a person may see that as being beyond the purview of task analysis. But it clearly is useful to imagine new tasks, and in creative and rapidly emerging areas of information technology, imagining new tasks may be much more critical to success than finely analysing out-of-date task concepts.

Traditional task analysis methods such as Hierarchical Task Analysis seek to achieve highly precise procedural specifications, so that a task analyst knows exactly what steps to perform, how to perform them, and what order to perform them in. This is highly desirable, and especially important if the task analyst is only marginally competent or not very creative. This is where structured methods really prove their value. However, the cost or downside of well-specified method scripts is that they lack flexibility. Thus, if circumstances are novel, or if new opportunities present themselves, the highly structured methods still just chug ahead just as they always do. SBD is highly flexible and accommodates different practitioner styles, different contexts of application, and idiosyncratic constraints and opportunities in a particular project. One can merely envision a scenario; one can employ the concepts from Making Use; one can methodically follow the more structured approach of SBRA; or one can even employ a Shepherd-like hierarchical task analysis of a systematic sample of scenarios. All of these approaches belong to the family of SBD methods, and each can be employed when it is most appropriate.

Scenario-based requirements analysis

Scenario-Based Requirements Analysis (SBRA) has been developed as the starting-point activity in SBD of human-computer interaction (Rosson and Carroll 2001). Its framework is illustrated in Figure 1.
Initiating SBRA, analysts prepare a root concept prior to going into the field. The root concept is a document describing the vision, rationale, assumptions, and stakeholders of the target system. It is derived from various sources. For example, the vision may come from open-ended discussions among various people related to the target project. Identifying those people—the stakeholders—is also part of the root concept. The rationale may come from discussions about the current technology and problems in the target domain. Finally, listing assumptions about the project and their impacts on it provides thoughts for analysts.

After preparing the root concept and questions about it, they conduct field studies. They use several tools and techniques of task observation and recording (Diaper 1989). They conduct qualitative research: observing the workplace, recording the work setting and activities, interviewing stakeholders, and analysing artefacts.

Then they summarise the collected data to identify and illustrate the stakeholders, activities, and tools or artefacts of the project. Also, they recognise the general themes or workplace themes of the project. During this process, analysts use task analysis techniques such as Hierarchical Task Analysis (Shepherd 1989) to decompose complex tasks into subtasks. In addition, this summarising step may employ a technique similar to the affinity diagram method of contextual design (Beyer and Holtzblatt 1998). Analysts use post-it notes to jot down ideas and then make them into a group and give it a title. More recently, practitioners implemented those techniques into design projects and reported their experiences.

Having identified the basic elements from the field data, analysts enter into iterative cycles of scenario description and claims analysis. First, they gather all the information about current practice to create problem scenarios. Problem scenarios contain the identified elements such as the project’s stakeholders, their activities, and tools or artefacts they use. They represent and illustrate the current practice of the project. Based on the problem scenarios, the analysts conduct a claims analysis. The iterative cycles of scenarios and claims can be supported with the scenario exploration technique.

**Scenario exploration**
Requirements analysis can be seen as an **inquiry process**: an iterative cycle of questions and answers, which aims to discover and explain hidden requirements (Potts, Takahashi, and Anton 1994). Scenario exploration is categorized into this view. It is a structured brainstorming session, in which a group of analysts conducts systematic inquiry into a usage situation. It may include actual users and customers. Also, various stakeholders in system analysis and design can contribute to the activity of scenario exploration. Usability specialists use the past experiences and knowledge of usability in addition to design heuristics and theories in human-computer interaction. Users can participate in this activity to provide their domain knowledge.

A scenario exploration session starts from a basic scenario illustrating part of the root concept of the target domain. The participants of the session put questions to the scenario. Typical questions may include who, what, when, where, why, and how questions (5W+1H questions) and what-if questions. For each question, they provide solutions as scenarios. The scenario and questions iteratively evolve as the analysis progresses.

Scenario exploration employs simple tools: piece of papers such as post-it notes and a wide-open area such as a wall, whiteboard, or table. There are two types of (possibly colour-coded) paper: scenario slips and question slips. On a scenario slip, analysts write a scenario; on a question slip, they write a question (a 5W+1H or what-if question) about the scenario. Analysts put them side-by-side or top-to-bottom (possibly with a link or line) in the wide-open area. The answers to the question form scenarios, so the analysts write each answer on a scenario slip and put it next to the question slip. This scenario in turn evokes new questions. As the analysts continue working like this, they obtain a large map of scenarios and questions in a relatively short time in a group session.

**Framework of concept and scenario evolution**

In the theory of design, one considers the tension between the abstract concept of design principles and the details of specific interaction techniques. The scenarios discussed in this chapter generally refer to the detail domain: they illustrate specific actors, contexts, and uses. As an example, the next section discusses the evolution of the Familyware concept. An envisioning scenario of Familyware illustrates that a girl is excited about her father’s message and expresses her feelings by shaking her teddy bear; the feelings inside her are expressed in a physical behaviour that serves as the trigger to signal her father. This episode does not need to be developed as it is, because it may illustrate one of the views of the Familyware concept. To develop the Familyware concept, therefore, discussion in the abstraction domain also is necessary. Figure 2 illustrates the framework of concept and scenario evolution.
Concepts at the abstraction level evolve along with scenarios at the detail level. Designers develop a root concept before they go into the field. They log real world episodes, which represent specific tasks and activities at the detail level. The episodes then contribute to strengthen the root concept. This step also provides the themes of the concept. They show helpful hints to envision problem scenarios, which are articulated at the detail level. Comparing the real world episodes and the problem scenarios allows users to articulate the root concept and add further issues, which have to be considered. The articulated concept is used to envision more detailed scenarios. Then, an analysis of the produced scenarios improves the concept that is used to produce them. This iterative cycle continues to articulate the concept of a target system. The framework of concept and scenario evolution supports and provides another facet of SBRA described in the previous section.

Example: Developing the Familyware concept and its use

This section demonstrates a scenario-based requirements process based on the framework of concept and scenario evolution. The example discussed here employs Familyware—a collection of communication tools for distant family members and people who have a close relationship (Go, Carroll and Imamiya 2000).

Root concept

The high-level goal of the Familyware project is summarised in Table 1. The fundamental idea of the project comes from supporting communication in a community. Unlike previous work like the Blacksburg Electronic Village project (Rosson and Carroll 1996), the Familyware project focuses on relatively small communities such as a family and friends. Familyware seeks to allow
those community members to exchange messages. The basic rationale behind it is that digital surroundings and peripheral displays with a communication function keep a feeling of connection among family members. Table 1 shows the parent-child relationship among family members as an example of a stakeholder group. (Go, Carroll, and Imamiya, 2000 discusses friends as another example of a stakeholder group.)

<table>
<thead>
<tr>
<th>Component</th>
<th>Contributions to the Root Concept</th>
</tr>
</thead>
<tbody>
<tr>
<td>High-level vision</td>
<td>Family members exchange messages</td>
</tr>
<tr>
<td>Basic rationale</td>
<td>Digital surroundings and peripheral displays with a communication function keep a feeling of connection</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Stakeholder group</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Family members</td>
<td></td>
</tr>
<tr>
<td>Parent</td>
<td>More effortless awareness</td>
</tr>
<tr>
<td>Child</td>
<td>Easy to communicate</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Starting assumptions</th>
<th>Use everyday artefact as display</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Internet is used for long distance communication</td>
</tr>
</tbody>
</table>

**First round interview**

The purpose of our interview was to obtain episodes of family communication in order to verify the assumptions of the root concept. We also gathered examples of opportunities of family communication to strengthen the root concept of Familyware. We conducted an informal interview with a college professor who has a daughter and a son. She explained her experience with her children in the following scenarios.

*Telescope Scenario:* Alison is a college professor and she lives with her two children. Her children did not want to go to kindergarten at first, so she stayed with them in their classroom on the first day of school. Next day, she sat in her car, watching the classroom activities through the window. Then, she told her children that she would watch them from her office (on campus) through a telescope—though that is physically (geographically) impossible. The children were comfortable going to kindergarten because they felt that their mother would still be with them. Now, the younger child sometimes says, “I did XYZ today. Did you see me?”

This episode articulates the opportunity of family communication. Alison’s younger child seeks to have a feeling of connection to her. He wants to share his experiences with her. Another episode she mentioned reveals a more complicated communication situation.

*Phone Call Scenario:* Alison is writing a research paper for a conference. The deadline is not far away, so she has often been staying late at her office. Kaz, Alison’s son, has been worrying about her: “Why doesn’t Mom come home?” He felt he had to do something. In the kitchen, he found his grandmother’s phone number on the wall. He called her and asked for his Mom’s office phone number. Then, Kaz called Alison’s office. She was surprised but also happy to hear from him. For the rest of the evening, Kaz called her office every ten minutes. Alison could understand his behaviour; nevertheless, she eventually became annoyed.
This episode raises the issue of availability for communication. Alison experienced a mixture of positive and negative feelings. Her primary work was to write the conference paper. Except for an emergency, she did not want to be disturbed by a direct communication device like a telephone. Using everyday artefacts to display information may take care of this issue. Table 2 summarises the themes and issues obtained from these scenarios.

<table>
<thead>
<tr>
<th>Theme</th>
<th>Issues Contributing to the Theme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shared feeling of connection</td>
<td>Family members want to keep in touch with each other to keep a feeling of connection and share experiences</td>
</tr>
<tr>
<td>Wider view of social activities using technology</td>
<td>Family members have their own primary work.</td>
</tr>
<tr>
<td>Communication among the small, local, private community</td>
<td>Do not want to be disturbed during primary work.</td>
</tr>
</tbody>
</table>

### Listing basic tasks and activities

Another activity we conducted in order to obtain scenarios was a participatory design meeting with potential users. From our experience, we understand that it is impossible to enumerate all the tasks and activities of a college professor. It is pointless to videotape all his or her tasks and activities because it is too expensive to analyse them and it may create privacy issues. Instead, we held an informal brainstorming session. In less than twenty minutes, we listed more than fifty tasks of a college professor. Some of them are given in Table 3.

<table>
<thead>
<tr>
<th>Prepare class material</th>
<th>Develop final exam</th>
<th>Order materials</th>
</tr>
</thead>
<tbody>
<tr>
<td>Participate in faculty meeting</td>
<td>Write grant proposal</td>
<td>Discuss with students</td>
</tr>
<tr>
<td>Grade midterm and final</td>
<td>Perform field work</td>
<td>Manage project</td>
</tr>
<tr>
<td>Go to lunch</td>
<td>Send FAX</td>
<td>Conduct experiment</td>
</tr>
<tr>
<td>Teach class</td>
<td>Make a phone call</td>
<td>Hold research seminar</td>
</tr>
<tr>
<td>Hold research meeting</td>
<td>Read a research paper</td>
<td>Plan business trip</td>
</tr>
<tr>
<td>Make a cup of tea</td>
<td>E-mail</td>
<td>Browse Web pages</td>
</tr>
<tr>
<td>Compose research paper</td>
<td>Review paper</td>
<td>Meet company people</td>
</tr>
</tbody>
</table>

Table 3 is not an exhaust list of the tasks and activities of a college professor. The purpose of creating this list is to derive day-in-the-life scenarios as a touchstone of requirements elicitation. We asked the participants to recall and imagine a typical day of work; then, each of them created a day-in-the-life scenario by picking tasks out of the task list. Because this scenario is a specific instance, tasks may require descriptions of assumptions and backgrounds, which are attached to the scenario. For example, when a college professor grades a midterm examination, this implies that the examination had been conducted. When she composes a research paper, we may assume that the paper deadline is soon.

The two leftmost columns of Table 4 demonstrate a college professor’s day-in-the-life scenario. Tasks are shown in chronological order: the first column represents the approximate time when the corresponding task starts. The tasks in parenthesis, “arrive at office” and “leave office,” are assumptions added by the analysts. Developing a day-in-the-life scenario like Table 4 reveals these assumed tasks by putting all the considered tasks in a wider context. In fact, working on a
day-in-the-life scenario helps the analysts to focus on linkages between individual tasks and assumed, forgotten, or overlooked tasks that occur before or after them.

After creating the day-in-the-life scenario, we visited the professor’s office to discover the context information for each task. During our visit, the participants added information about sites, artefacts, and participants to the original scenario. In addition, for each of the tasks, we asked ourselves what-if your children want to contact to you, as they did in Alison’s episode. Based on this we judged the professor’s availability during the tasks. Table 4 shows a day-in-the-life scenario with the context information. For each task, the Site column represents where the task is carried out; the Artefacts column shows artefacts used; the Participants column lists other people involved in the task; and the Availability column specifies the priority of the task—no. 1 does not accept interruption, no. 2 can accept interruption; and no. 3 accepts interruption. For example, at 10:30 the professor, who is the main actor in this scenario, starts teaching a class in a classroom. She makes use of an over-head projector and transparencies. She sometimes takes notes on a transparency with a marker while referring to her notebook. In the class, she teaches students: they are the main participants in teaching. When she teaches, she concentrates on the class, so she does not want to be disturbed by anyone from outside except in an emergency situation. In contrast, at 12:00 she joins her friends for lunch. At the café, she uses a glass, knife, fork, spoon, plates, and paper napkin. During lunch, she does not mind if other people interrupt her: she accepts interruptions. In short, the college professor’s tasks employ a PC as a primary artefact, and her availability depends on the task not on the site. The identified elements are used to envision a problem scenario for Familyware.

<table>
<thead>
<tr>
<th>Time</th>
<th>Task</th>
<th>Site</th>
<th>Artefacts</th>
<th>Participants</th>
<th>Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td>9:00</td>
<td>(Arrive at office)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9:00</td>
<td>Make a cup of tea</td>
<td>Office</td>
<td>Tea pot, cup, spoon</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>9:10</td>
<td>E-mail</td>
<td>Office</td>
<td>PC</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>9:50</td>
<td>Read a research paper</td>
<td>Office</td>
<td>Paper, notebook, maker, pen, PC</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>10:30</td>
<td>Teach class</td>
<td>Classroom</td>
<td>OHP, transparency, marker, notebook</td>
<td>Students</td>
<td>1</td>
</tr>
<tr>
<td>12:00</td>
<td>Go to lunch</td>
<td>Café</td>
<td>Glass, knife, fork, spoon, plate, paper napkin</td>
<td>Friends</td>
<td>3</td>
</tr>
<tr>
<td>13:00</td>
<td>Grade midterm</td>
<td>Office</td>
<td>Answer sheet, pen, notebook, PC</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>14:30</td>
<td>Compose research paper</td>
<td>Office</td>
<td>PC, notebook, pen, dictionary, book, whiteboard, maker</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>16:30</td>
<td>Participate in faculty meeting</td>
<td>Meeting room</td>
<td>Notebook, pen, whiteboard, maker</td>
<td>Faculty members</td>
<td>1</td>
</tr>
<tr>
<td>18:30</td>
<td>Discuss with students</td>
<td>Lab</td>
<td>Whiteboard, marker</td>
<td>Students</td>
<td>1</td>
</tr>
<tr>
<td>19:30</td>
<td>(Leave office)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The theme of mobility emerges from the analysis. Although most tasks in Table 4 are carried out in the professor’s office, she does conduct task in various other places such as a classroom, meeting room, and research laboratory. In fact, the assumed tasks, “arrive at office” and “leave
office,” involve movement. Furthermore, she might go on a business trip. In this situation, the travel time and distance might be longer. How to enable communicate during the moving process is a challenging theme of the root concept of Familyware. Table 5 shows an additional theme to those in Table 4. The identified themes are considered in order to create scenarios along with the root concept.

Table 5. An additional theme from the day-in-the-life scenario of a college professor

<table>
<thead>
<tr>
<th>Theme</th>
<th>Issues Contributing to the Theme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mobility</td>
<td>Family members do not remain in the same place.</td>
</tr>
</tbody>
</table>

Scenario development: Familyware scenario

Based on the activity so far, we developed the following problem scenario. The tasks and themes identified in the detail domain are put in context so that they become prominently illustrated for evaluation. In this scenario, two family members, Wendy and Sean, are the main actors. They are separated by a long distance and try to share their feelings via their everyday artefacts. The scenario assumes that the Internet is used as a background communication channel.

Wendy Scenario: Wendy, a five-year-old girl, picks up her favourite teddy bear and takes it to a corner of her room. In the corner there is a TV, and when she approaches it, the TV turns on automatically. An electronic greetings card appears on the TV screen; it is from her dad. The card plays back his voice, “I am going to bring a puppy home with me!”

Sean, Wendy’s dad, is working at a software company in College Park, Maryland. He has been gone all week from his home in Blacksburg, Virginia. Wendy loves him and is looking forward to the weekend.

Wendy is excited by the news: Dad is going to bring home a puppy! She holds her teddy bear tightly, then shakes it. These are Sean and Wendy’s special actions—if she thinks of him, she squeezes and shakes her teddy bear. Though she is not aware of it, the teddy bear contains an electronic device that has a wireless connection to the Internet. The device senses being shaken and sends a signal to Sean’s computer.

In his office, Sean is composing a project report on his computer. On the screen there is a small window displaying a photo of Wendy. He notices that a big smile has appeared on it, and understands that she is thinking of him. He also smiles and thinks about the coming weekend.

This problem scenario is used not only as a shared vision among analysts and designers but also as key material for analysis at the detail level. A claims analysis is conducted to examine the practical features that have key implications for use in the problem scenario. The features include an electronic greetings card on TV, the Internet as a background communication channel, a teddy bear as an interactive device, and a photoframe as a window on the PC screen.

Electronic greetings card on TV
+ can be implemented by a conventional e-mail system,
- but requires both the sender and receiver to have networked computers.

The Internet as a background communication channel
+ provides global access from anywhere in the world,
- but might be expensive to use or might require authentication actions from the user.

Teddy bear as an interactive device
+ is an everyday object for children,
allows a young child to express his/her feelings by physical manipulation,
- but may not provide feedback that a message has been sent.

*Photoframe as a window on the PC screen*
+ is an everyday object for office workers,
+ displays photos of his or her children,
+ could display different photos depending on the message received from the teddy bear
  via the Internet,
- but may not provide feedback that a message was received.

The claims add further details about the problem scenario, and the process of examining claims in the problem scenario poses questions for Familyware. For example, the claim related to the electronic greetings card on TV says that it can be implemented by a conventional e-mail system but requires both the sender and the receiver to have networked computers. This raises a general question about how to install the Familyware system at the sites. This question and others derived from the claims analysis process will be used in the next design step.

**From detail to abstraction**

The next step of our design activity obtains key abstraction-level design principles from the detail level. We held a design meeting on the Familyware concept to analyse the problem scenario. Table 6 shows the tasks and artefacts identified from the problem scenario. Most items, such as picking up a teddy bear, are directly obtained from the problem scenario, but the ones in italics are assumed tasks identified from the analysis. For example, the problem scenario does not mention how the father sends the electronic card; in addition, it does not define how to install the Familyware artefacts.

The analysis continues to derive the abstraction of the tasks. Table 7 lists pairs of the detail and abstraction of the tasks. The child may install a Familyware artefact herself, or she may have it installed at her site for her. She and her father make a promise: they define a special action. She triggers the Familyware artefact: she picks up her teddy bear and take it to a corner of the room. She notices a message from her father: she looks at an electronic greetings card on TV. After that, she sends a message to her father: she hugs the teddy bear and shakes it. In contrast, the parent may set up a Familyware artefact on his office computer. He and his daughter make a promise as described above. He may send a message using the artefact; in this case, he may send an electronic card to his daughter. He has a primary work: he works on a project report using a computer. He notices a message as a change in the photo displayed on his computer’s desktop.

| Table 6. Tasks and artefacts from the scenario analysis |
|---------------------------------|-----------------|-----------------|-----------------|-----------------|
| **Task** | **Artefact** | **Task** | **Artefact** |
|  |  |  |  |
| *Install Familyware artefact* |  |  |  |
| Define a special action | Teddy bear | Define a special action | Teddy bear |
| Pick up teddy bear and take it to a corner of the room | Teddy bear | Send electronic card |  |
| Look at electric card | TV monitor | Compose project report | PC |
| Hug teddy bear and shake it | Teddy bear | Glance at photo | PC |
Table 7. Detail vs. abstraction analysis of the problem scenario

<table>
<thead>
<tr>
<th>Child</th>
<th>Abstraction</th>
<th>Parent</th>
<th>Abstraction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Install Familyware artefact</td>
<td>Install Familyware artefact</td>
<td>Install Familyware</td>
<td>Install Familyware artefact</td>
</tr>
<tr>
<td>Define a special action</td>
<td>Make promises</td>
<td>Define a special action</td>
<td>Make promises</td>
</tr>
<tr>
<td>Pick up teddy bear and take it to a corner of the room</td>
<td>Trigger Familyware</td>
<td>Send electronic card</td>
<td>Send messages</td>
</tr>
<tr>
<td>Look at electric card</td>
<td>Notice messages</td>
<td>Compose project report</td>
<td>Primary work</td>
</tr>
<tr>
<td>Hug teddy bear and shake it</td>
<td>Send messages</td>
<td>Glance at photo</td>
<td>Notice messages</td>
</tr>
</tbody>
</table>

Two issues emerged during the analysis: the symmetry/asymmetry of Familyware artefacts and consciousness of sending a message. For example, there is asymmetry in the teddy bear and the photoframe on PC system. The child uses the teddy bear as an input device and TV monitor as a display, while the parent uses the PC as a display. A symmetrical interface is also possible for Familyware artefacts; a necklace-style peripheral display, for example, has symmetric features and uses. The necklace Familyware is intended to support communication between close friends (e.g., boyfriend and girlfriend). Each wears a necklace that can send to and receive from the partner a simple signal, such as the temperature of the mounted stone (Go, Carroll, and Imamiya 2000). It can be implemented by using wireless technologies such as radio transmission or a digital cellular system to transmit the signal.

The style of communication may depend on the degree of consciousness involved in sending the message. If the child is conscious of sending a message to her father when she holds and shakes her teddy bear, then the communication will succeed after he has noticed the message. If she is not conscious of sending messages to her father, then the communication is in fact one-way communication: only her father knows the meaning of the communication.

These issues contribute to articulate the concept of Familyware. They allow us to consider what-if questions when we create more scenarios: what-if Familyware artefacts have symmetry and what-if they do not involve conscious message sending? These questions lead to other groups of Familyware artefacts. For example, the necklace interface for a boyfriend and girlfriend is a design solution to the symmetry issue while a rattle-photoframe interface handles the unconscious actions of a baby (for further discussion, see Go, Carroll and Imamiya 2000). These identified issues are used to envision further scenarios for Familyware use, together with the root concept and the issues already recognised at the initial analysis.

**Envisioning further Familyware use**

The design process continues to generate new scenarios from the concept developed so far. The example here illustrates how to conduct scenario exploration. Initially, it assumes that the main actors are a company worker and his daughter as expressed in the problem scenario, but other actors are added if necessary. Though the company worker’s main task is to compose a project report using a PC, he notices that his Familyware artefact has changed status.
At the beginning of the scenario exploration, the analysts take notes about the basic tasks on scenario slips. In this example, there are five basic tasks: setting up Familyware, making promises among family members, triggering Familyware, sending messages, and receiving messages. Apart from the Familyware-related tasks, the family members have their primary work.

The analysts focus first on the message sending task. They gather any episodes, examples, and anecdotes. A straightforward way is to describe it from the problem scenario; thus, they write down how the teddy bear is being used on a scenario slip: Wendy holds her teddy bear and shakes it; then, a message is sent to her father. This simple scenario involves assumptions such as a teddy bear being an everyday artefact for a young girl. Challenging the assumption allows designers to ask the question: what if the actor is a boy? A boy might have a teddy bear too, but there are alternatives.

The analysts examine everyday objects for boys considering the root concept of Familyware. They observe a portable game machine as a boy’s everyday object; then, they compose a scenario: a boy has a portable game machine like a Nintendo GameBoy; when he uses it, he sends a message to his father in much the same way as in the teddy-bear case. The issue of being consciousness of sending a message derives another scenario at this point. The analysts move their focus to the situation in which a boy is not aware of sending a message. They set up a scenario in which a boy pedals his bicycle: the boy loves his bicycle; he usually plays with it outside his house; as he pedals it, information about its wheel rotation speed is sent to his parent as his message. Another question related to the gender issue in the Wendy scenario is what gender-free artefact is possible for kids. The analysts choose shoes as possible artefacts: a small kid has a favourite pair of shoes; she wears them almost everyday; at each step, the shoes send a signal to her parent by means of wireless mobile technology.

In contrast, discussion about the message noticing task tries to articulate an alternative use situation of a Familyware artefact in the Wendy scenario. The analysts set up a commuting situation: a father commutes by car everyday; he sometimes drives his car on a business trip; he notices that his car navigation system is displaying an e-mail message from his daughter. This radical change emerges if the analysts challenge an assumption of the Wendy scenario, which was that Wendy’s father stays in his office. If he is not in his office, he may be doing different tasks. Setting up the new situation generates the straightforward question: what items can provide messages without disturbing the driver. Possible channels include sound: music from the car radio or the sound of the car’s engine. Another channel is the illumination of dashboard meters. Part of the scenarios and questions is summarized below. The indentation represents the depth of scenarios and questions.

**Basic scenario:** Sending a message

**Scenario:** Wendy holds her teddy bear and shakes it; this causes a message to be sent to her father.

**Question:** What if the actor is a boy?

**Scenario:** A boy has a portable game machine like a Nintendo GameBoy; when he uses it, he sends a message to his father in much the same way as when he plays a game.
**Scenario:** A boy loves his bicycle; he usually plays with it outside his house; as he pedals his bicycle, information about its wheel rotation speed is sent to his parent as his message.

**Question:** What everyday artefacts are gender-free for kids?

**Scenario:** A small kid has a favourite pair of shoes; she wears them almost everyday; at each step, the shoes send a signal to her parent by means of wireless mobile technology.

**Basic scenario:** Noticing a message

**Scenario:** Father commutes by car everyday; he sometimes drives his car on a business trip; he notices that his car navigation system is displaying an e-mail message from his daughter.

**Question:** What items do not disturb a driver?

**Scenario:** He is listening to music while driving; it is jazz music, but it smoothly changes to rock music; he notices this change and understands that his son has sent him a message.

**Question:** What if he does not listen to music while driving?

**Scenario:** While driving, he notices that the sound of the car’s engine has become a bit higher; from this sign, he is aware of his son’s message.

**Scenario:** The dashboard meters are illuminated while he is driving; he notices that the illumination colour has changed from light blue to light green; he recognises that his son has sent him a message.

Though the scenario exploration seeks to create as many scenarios as possible so that the analysts can pinpoint hidden requirements behind the concept, an obvious question is where to stop. A stopping heuristic may be the point where scenarios produce a general usability rationale contributing to the concept of the target system. Another stopping heuristic is related to time—because a group of analysts and designers conducts a scenario exploration session as collaborative work, their time is expensive. The session should have a time limit: two hours seems realistic.

After this session, the analysts sketch potential artefacts in order to develop and concretise their further possibilities. In terms of the above scenario-question threads, they choose the shoes scenario for sending messages and the music-in-a-car scenario for noticing messages. They sketch simple interfaces and conduct a claims analysis.

*A pair of shoes, which sends signals*
+ is an everyday artefact for anybody,
+ can be used virtually anytime,
- but might be heavy for kids,
- but might cause a privacy issue.

*The music in a car that conveys a kid’s message*
+ can be used while driving,
These issues provide feedback to the concept at the abstraction level in order to articulate it. More iterative cycles of articulating the concept and detailing the scenarios might be needed to develop them further.

Figure 3 illustrates a summary of the example in this section. At the abstraction level, the root concept evolves in the sense that more issues and themes are added to articulate it; at the detail level, in contrast, several scenarios are derived to implement the concept with issues and themes and generate further issues and themes. The process starts with the development of the root concept. Then, in order to verify the root concept, analysts collects real world episodes such as the telescope scenario and the phone call scenario. The day-in-the-life scenario of a college professor is also envisioned to identify assumed tasks. These activities produce the themes of the root concept. The obtained theme and the root concept are used to create the Wendy scenario. This problem scenario is analysed to obtain further issues and themes. These are added to the root concept to articulate it. Again, the detailed scenarios that implement the concept with the themes are envisioned to identify further issues and themes placing them in a new context. The intertwining process of scenarios between the abstraction and the detail promotes the concept and scenario evolution.

**Conclusion**

Scenario-based design and traditional task analysis have shared properties: they both focus on users, concrete descriptions, and specific instances. Scenario-based design in human-computer interaction puts a strong emphasis on users. It articulates who the users are, what they do with a technology and how they do it, and under what circumstances they work. A scenario is a concrete description of work and activities, so it describes a specific instance and usage situation.
Similarly, the user-centred view is taken by task analysis, too. Unlike system analysis, task analysis puts the stress on users and their method of use. It obtains a concrete description of tasks; for example, Hierarchical Task Analysis (HTA) creates a task hierarchy and decomposed subtasks including plans for subtask execution. The final step of the decomposition may enumerate specific keystrokes such as pressing the 'Delete' key or clicking the 'Cancel' button. In this sense, task analysis also considers specific instances of tasks.

In contrast, scenario-based design and task analysis have different perspectives: scenario-based design is more oriented to inquiry, while task analysis is more oriented to obtaining a single, correct description. Scenarios articulate a view from specific instances. When they are used to describe the current use of a technology, they may be regular cases of use, or they may be rare ones. When they are used to describe the future use of a technology, that situation might or might not occur in the future. In this sense, the limited numbers of scenarios do not describe complete and exhaustive uses of a technology. Rather, scenarios focus on exemplifying a concept in the abstraction domain as a rich, concrete description in the detail domain so that designers and analysts, and even users, can share a vision, knowledge, and experience of the project. The shared scenarios provide a chance to makes question and answers about the situation described in the scenarios. The scenarios also allow those people to contribute to the concept—making an addition of their vision, knowledge, and experience because additional scenarios to the previous ones provide feedback to the concept. As the concept behind a scenario evolves, the scenario instantiating it evolves; thus, the scenario contributes to the evolution of the concept.

Task analysis is appropriate for producing a precise, correct description by analysing the current use of an existing technology. To design a training manual for a system, correct procedures of tasks performed on the system should be described. A step-by-step description of completing a task, produced by a task analysis method, offers the task information needed for the training manual. In addition, it can be used to estimate the time required to complete the task. These cases require a precise, correct description of the task. To examine knowledge transfer from a previous system to a new system, the procedures and knowledge for completing a task on the previous system should be articulated and analysed. The knowledge description can be used to evaluate how easily the users of the previous system learn the new system. This also requires a correct description. The description produced by task analysis methods reflects the view of the existing use of a system but may suggest future uses of it.

The scenario-based approach described in this chapter can contribute to the current view of task-analysis. Following the results of task analysis such as HTA, scenarios add various views to them; hence, designers and analysts explore new possibilities of use. Although this chapter concentrates on scenario elicitation with interviews, scenario development with brainstorming, and relatively heuristic analyses of them, the robust and detailed data obtained from task analysis of the current use situations may be a rich source of the scenario development process. The scenarios created through the process can be analysed using task analysis techniques even though the envisioned scenarios are not exhaustive, as mentioned. Because they are not exhaustive, the scenario development activities help designers and analysts to identify implicit tasks that may be assumed, forgotten, or ignored, taking into account the relationships among tasks. Overall, scenarios help to expand and create design ideas based on the result of task analysis.

References


