Designing the User Interface of a Handheld Device for Communication in a High-Risk Environment

Jesper Kjeldskov and Jan Stage

Department of Computer Science
Aalborg University, Denmark
{jesper,jans}@cs.auc.dk

Abstract. This paper discusses the process of designing the user interface of a handheld mobile communication device. The purpose of this device is to improve the coordination between parties that are conducting a complex work task in a high-risk environment by standardizing the communication between them. The discussion is illustrated with examples from the development of a handheld device interface designed to improve communication between actors involved in the work tasks on board a very large container ship when it arrives at or leaves a harbour. The improvement is achieved by shifting from the existing communication protocol based on radio-transmitted natural language to network-communication of selections from pre-defined menus. This new design was developed through a systematic combination of design methods, and the result overcomes key problems that are experienced with the existing technology.

1 Introduction

The emerging variety of handheld computing devices such as mobile phones, personal digital assistants, etc. is challenging our established practices for user interface design. On a mobile device, the means for interaction are very limited; the screen is very small and there is only a few buttons instead of a complete keyboard. Moreover, the fundamental characteristic is that such devices can be used while being mobile. This implies that designers cannot expect users to devote their full attention to the operation of the device. Apart from these limitations, however, today's handheld computers potentially offer users detached from a stationary workplace the power of information processing and communication known from desktop computers connected to the Internet. Thus a strong motivation exists for addressing the human-computer interaction issues inherent in mobile technology in order to exploit the potentials of mobile computing better.

The literature on user interface design for mobile devices is very rich on specific designs. Typically the rationale behind a design is explained and the design itself is described in detail. Yet the way in which designers developed the design and the underlying analysis of the domain of use is rarely described in a methodical manner that enables others to learn from the design process.

Our aim has been to overcome this lack of methodologies and address the process of designing user interfaces for mobile handheld devices. In order to accomplish that, we have combined specific object-oriented method fragments from the software engineering discipline with general guidelines from the human-computer interaction discipline. The idea of integrating object-oriented method fragments in human-computer interaction was based on the systematic way in which object-oriented methods deal with the context of use. These methods describe in a systematic manner what objects the prospective user is working with, and how this work can be supported by the system being designed. This systematic focus seems particularly valuable in a situation where the resources available for interaction are very scarce such as in the case of handheld computers.

The purpose of this paper is to present how we have designed a handheld mobile device interface using a combination of object-oriented method fragments and human-computer interaction guidelines. We have chosen to delimit the experiment to a specific application area: process control in high-risk environments. This area of application was chosen because 1) workers in such environments could often benefit from access to computer-based tools for process control and communication while being mobile and 2) the user interfaces of these tools need to be carefully designed to avoid (preferable eliminate) human errors leading to hazardous situations. Our example is the design of a handheld device interface intended to improve communication on board a very large container ship. Specifically, we have focused on the communication that is taking place when coordinating the cooperative work tasks involved when the ship arrives at or leaves a harbor. However, the basic ideas of our design would also apply to other instances of communicative coordinated work within the maritime context as well as other high-risk environments.

The following section 2 surveys related work on the design of user interfaces for mobile handheld devices and system design for high-risk environments. Section 3 presents and discusses the high-risk environment in which our field studies were conducted. This discussion emphasizes the problems of the existing communication protocol. Section 4 describes the analysis process that we conducted in that environment. The results of this analysis are documented in a formalized model of the task to be implemented in a computerized system. Section 5 presents the interface design of our experimental prototype. Section 6 compares the characteristics of this computerized system to the problems of the existing communication that were expressed in section 3. In section 7, it is concluded on how the user interface of the handheld communication device was designed, what problems we identified during the experiment and what avenues for further work we see.

2 Related Work

The design of user interfaces for mobile devices impose several challenges on interaction design. Screen real estate is very small due to limited physical size of the devices stressing the design of graphical interfaces [1] and forcing designers to explore the use of new means of output. Especially the use of sound is being carefully investigated [2,3,4]. Correspondingly, interaction with mobile device interfaces is limited due to handheld operation forcing designers to explore new means of input such as e.g. gestures [5], speech [6], environmental sensors [7] and context awareness [8,9]. Furthermore, mobile use contexts are very dynamic, forcing designers to explore more carefully the relation between their design and the physical surroundings, in which they are to be deployed [10].

Examining the literature on mobile HCI, it is thus clear that new constraints and approaches should be carefully considered in order to create useful applications for mobile devices. Motivating such effort within the domain of high-risk environments, recent research indicate that in situations where actors are concerned with computerized information and processes of critical importance remote from their current location, increased utility value can be gained from handheld computing. Examples count the use of remotely controlled service robots for aiding disabled or elderly people [11], distributed process control and error diagnosing in wastewater treatment plants [12] and the use of mobile multimedia communication channels for telemedicine and early diagnosing of patients in emergency ambulance services [13]. Similar to these cases, work activities in high-risk environments such as e.g. hospitals, nuclear power and air traffic control involves computerized information and automation systems and often count a number of distributed actors depending on access to these resources. However, the use of handheld computers for communication and cooperation in such environments has not been widely explored.

Safety-critical computer systems are usually defined as *"computer electronic[s] or electromechanical system[s] whose failure may cause injury or death to human beings"* [14]. Within the disciplines of software engineering and human-computer interaction, the research literature is rich on references on the design of such systems. From a software engineering perspective, the primary concern is the development of *reliable* software that does not malfunction – as in the case of e.g. the Ariane 5 accident [15] the London Ambulance Service breakdown [13] and in a number of accidents involving commercial aircrafts [16]. To avoid such situations, emphasis is, among others, put on the importance of detailed and valid requirement specifications and the use of safety analysis techniques [17]. Also the issue of evaluating software reliability is given a lot of attention [18,19]. From a human-computer interaction point of view, the problem is not malfunctioning software as such but human error caused by poorly designed user interfaces – as in e.g. the British Midland 1989 air crash accident, in which pilots erroneously shut down the only operational engine due to a simple mapping mismatch in the cockpit [16]. Designing human-computer interfaces that avoid such incorrect operation even in stressful situations represent a great challenge for the HCI community. For an introduction to human factors in safety-critical system design see e.g. [20].

Much of the discussion on human-computer interfaces in safety-critical system design originates from work inspired by the Three Miles Island nuclear power accident in 1979. Instead of forcing additional rules and regulations on the operators of complex computerized industrial installations, it was suggested that better designed user interfaces could support the avoidance of hazardous situations. During the last two decades, this discussion has extended into other areas of high-risk such as healthcare, aviation control, air traffic control and space mission control.

In order to avoid accidents in high-risk domains, understanding the state of the system operated or domain controlled is critical. [21]. Consequently, Rasmussen [22,23] suggest that computer interfaces should be designed to facilitate operators' reasoning about the domain of operation better and thus support human interaction contrary of total system automation as also discussed by Norman [24]. For this purpose, Lind [25,26,27] suggests a formalism for representing complex systems or environments as they were intentionally designed.

Acquiring the knowledge necessary to carry out these principles into design, however, demands a solid understanding of the use specific context [20]. Thus most of the designs discussed within the HCI literature are based on ethnographic studies of work activities. Having observed the importance of physical flight strips in air traffic control for example, Mackay [28,29] suggests the use of augmented reality for *relating* new software-based tools to the existing physical ones rather than *replacing* them. Observing radio communication overload among firefighters, Camp et al. [30] outlines a communication system that enables messages to be directed to relevant receivers only, rather than being broadcasted to everyone within radio contact.

Based on another case study in air traffic control, Fields et al. [31] suggests a method for comparing design options for safety-critical systems, helping designers to choose between the ideas that evolves from their empirical studies.

3 A High-Risk Environment

Operating a container vessel in a size equivalent of 3 ½ football fields when departing from or approaching a harbor is a high-risk task. If not carried out carefully, the operation may result in the vessel running aground or colliding with the quay or nearby ships. In either case, collisions involving a vessel of this size would cause serious material damage, potentially severe injuries on personnel and possible loss of human life.

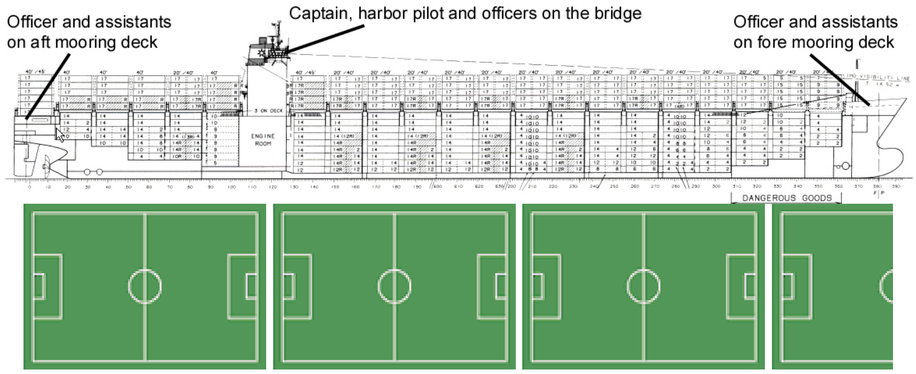

Fig. 1. Sally Maersk - one of the world's largest container vessels

Due to the size of container ships, the tasks of arrival and departure from a harbor have to be distributed among a number of actors. To prevent a hazardous situation, the work of these actors are carefully coordinated and carried out under the command of one person in charge of the whole operation.

Qualitative investigations into the overall operation of arriving at and departing from the harbor on a Maersk-Sealand container vessel have been carried out. Detailed documentation can be found in [32] and [33]. From this research, a number of limitations in present means for coordination have been identified. In the following, the task distribution and coordination of departing and arriving at a harbor are described. Subsequently, a list of limitations in present means for coordinating the task is presented.

3.1 Arriving and departing from the harbor

When the ship is in harbor, it is made fast to the quay in a fixed position by means of a number of mooring lines attached to bollards ashore. The specific pattern of mooring varies in accordance to weather conditions and the properties of the ship and the harbor. When the ship is ready for departure, the first step in leaving the quay is letting go the mooring lines. However, as the physical space of harbors is restricted and the means for maneuvering is limited in relation to the precision needed to avoid collisions, all lines holding the ship cannot simply be released simultaneously.

Fig. 2. The aft mooring of Sally Maersk in Felixstowe harbor

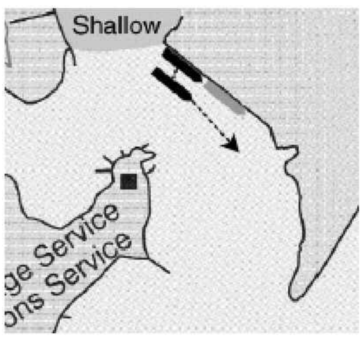

Fig. 3. Maneuvering the ship out of Felixstowe harbor

When a line is released from the bollard and pulled aboard, it will remain in the water for a period of time. Within this period of time, no means of propulsion is available as the line in the water may be sucked in and wrapped around the propeller or thrusters. Though only for a limited period of time, bringing the ship in a situation without means for maneuvering is typically not an option. Another reason for not just letting all lines go at the same time is the

dual function of the lines. While at quay the lines are means of fixation, when leaving the quay, the lines also becomes means of movement facilitating the ship to be pulled ahead or astern without the use of the propeller. Following these premises, the lines mooring the ship are released *sequentially* in accordance to the specific need for maneuvering within a given physical context.
When arriving at the quay, the sequence is more or less reversed. Instead of pulling the lines in, the lines are lowered down to the quay, fastened on the bollards and the ship is pulled into a fixed position. During approach an officer in addition typically reports distances to the target position continuously.

3.2 The distribution of the task

Carrying out the task of mooring and letting go the lines when arriving or departing from the harbor involves a number for physically distributed actors at designated locations on the ship and ashore (see figure 1).

- Bridge: the captain, chief officers and local harbor pilot
- Fore deck: first officer and three assistants
- Aft deck: second officer and two assistants
- Ashore: two teams of assistants

All coordination and decisions regarding navigation and maneuvering are made by the captain and the harbor pilot on the bridge and carried out by other actors upon request. On the bridge, the chief officers control the rudder, propeller and thrusters of the ship. Fore and aft, the first and second officers control the winches for rolling out or heaving in the lines via a central remote control. Assistants supervise the running of the winches. Ashore, the two teams of assistants lift the lines onto or off the bollards. While the actors on the bridge can see and hear each other, the actors on the deck cannot be heard or seen from the bridge. All communication between the captain and the officers on deck is thus conducted via VHF-based walkie-talkies. Subsequent communication between the officers and assistants fore and aft internally is carried out orally. Errors or misunderstandings in the communication may lead to a disparity between the actions believed to be taking place by the captain and the actions really taking place. As decisions made on a deceptive foundation may put the ship in a potentially dangerous situation, the work of the distributed actors is thus carefully coordinated.

3.3 Coordinating the work of the distributed actors

The work of the distributed actors is primarily communicatively coordinated. In order to prevent misunderstandings, the distributed actors refer unconditionally to the commands executed by the captain. Nothing is done, which has not been directly requested by the captain. Drawing on his practical experience with similar operations, the captain and the pilot typically plans the specific sequence of steps to be carried out prior to the operation of mooring or letting go the lines. The plan is then discussed with the officers involved and is revealed in a step-by-step fashion to the actors carrying it out during the operation. This strategy facilitates ad hoc changes by the captain and the harbor pilot adapting to the situation and prevents the teams of distributed actors getting “out of sync”.
At present, all commands are executed orally – either directly (on the bridge) or mediated through walkie-talkies (to personnel elsewhere on the ship). To verify that a command has been successfully received and understood, the receiver of a command is required to confirm it by repeating the command. If no confirmation is received, the command will be reissued within a given window of time.

In order to carry out the operation of arrival or departure in a safe manner, the captain needs an overview and full control over the propulsion, direction and mooring of the ship. While information about the rudder, propeller and thrusters are available via dedicated instruments on the bridge no information about mooring is facilitated. Maintaining an overview of the ship's mooring thus requires the captain to continuously keeping a mental model of the current state of affairs updated, based on the orders he has executed to the mooring crew and the confirmations he has received.

3.4 Limitations in present means for coordination

Coordination by means of oral communication has a series of limitations.

1. **Sound quality is often poor.** As walkie-talkies and VHF-radios often lack sound quality, incomprehensive messages, misperceptions and misunderstandings between the actors often occur. This leads to a need for repeating statements and meta-communicating.
2. **Spoken coordination lacks persistency.** Due to the ephemeral nature of spoken communication, messages are easily missed because they are only available during the limited period of time when they are "in the air". Afterwards, the information only exists in the memory of the actors taking part in the interaction and is not publicly available.
3. **Spoken coordination cannot be automated.** Spoken coordination involves actors remembering sometimes highly complex workflow and deciding for whom specific information may be relevant at which time. When the coordination is based on spoken communication, such workflows are hard to support and reducing the coordination workload is difficult.
4. **Spoken communication is time consuming.** Within a high-risk environment, time spent on communicating has to be minimized in order to maximize the time available for work tasks. As spoken communication is by nature sequential and can carry only a limited amount of information within a unit of time (compared e.g. digital networks), only the most essential information is typically communicated. Sometimes this is less than ideal.
5. **Spoken communication suffers from language barriers.** The communication on board the ship is usually conducted in a language different from the language being used by the local harbor pilot to communicate with other pilots, the pilot boat, tugboats, vessel traffic service etc. This results in the captain having limited immediate insight into the domain of the harbor pilot and visa versa and introduces a need for ongoing translations between the captain and the harbor pilot.
6. **Spoken coordination suffers from bottlenecks.** Regulating turn taking across several independent communication channels is not possible. Thus the radio messages between bridge and deck often disturb the communication between actors on the bridge and visa versa. The result is communication being cut up, and information being missed.
7. **Lack of information integration.** Spoken communication cannot be integrated with computer-based information sources. The captain can, of course, take the spoken information about distances, angles etc. to objects in the vessels immediate surroundings into consideration, but it cannot be made part of the computations regarding the ship's movements performed automatically on the bridge. As a result, the spoken information is usually not utilized to its full extent because it demands too many cognitive resources.

Supporting the coordination by means of a computerized information service, however, has promising potentials for overcoming some of these limitations. Using e.g. text-based computerized communication, poor sound quality from radio interference can be eliminated and the communication may be made persistent as in the case of e.g. chat applications. While text-based synchronous communication is also time consuming, facilitating the selection of predefined standard-phrases as seen on some SMS phones may reduce the time needed for

communicating a statement. Predefined standard-phrases may also facilitate optional translation between languages. In relation to bottlenecks, applying asynchronous properties to communication by capturing it in a computer system could facilitate parallel information being perceived in sequence rather than simultaneously. If the communication is captured by a computerized system, it furthermore potentially facilitates integration with other computer based data sources.
With these potentials in mind, we carried out a structured analysis and designed an information system meeting the described premises of the task.

4 Analysis

The problem and application domains were analyzed using the object-oriented method OOA&D which was developed within a software engineering context [34,35]. This method divides analysis into problem domain and application domain. The results of these analysis activities are described below.

4.1 Problem domain analysis

The problem domain of a computerized system is the objects and relations that the user is working with. OOA&D defines it as the part of the context that is administrated, monitored, or controlled by a system. In the case with the container ship this is what is administrated and monitored by the commanding officer of the ship. The classes of the problem domain are illustrated in Figure 4.

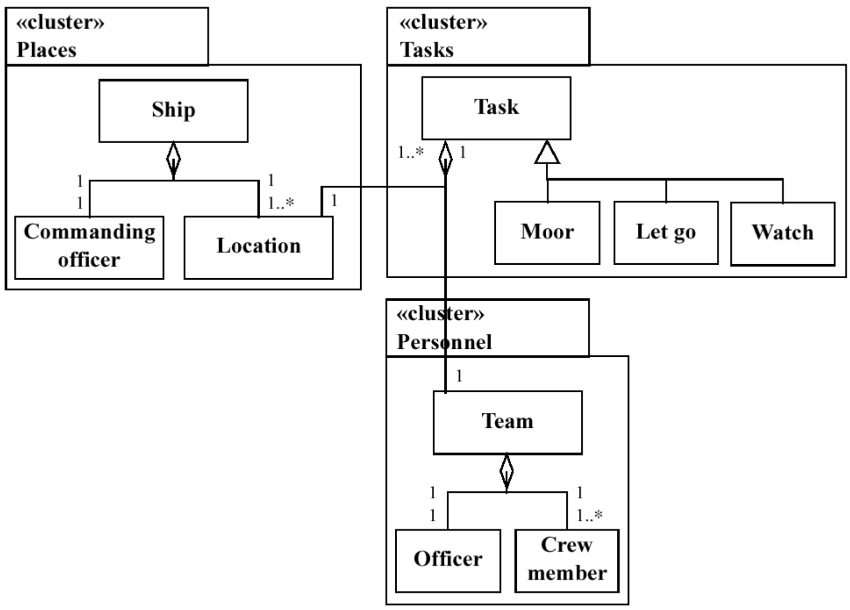

Fig. 4. Class diagram for the problem domain

The model has three clusters. The Places cluster models the physical aspects of the ship, and the most interesting class is Location, where a typical object is the bow. The Personnel cluster is used to model the people that solve tasks. The overall class is Team, and an object from this class aggregates an Officer that commands the team and a number of crewmembers. Finally, the cluster with tasks is what tie objects from the two other clusters together. Here there is a general class Task and a number of specializations that model the different task categories that should be supported by the system.

Each class has a statechart diagram that describes the dynamic behavior of an object from that class. Figure 5 shows an example, which is the diagram for the "Let go class". This diagram specifies the sequence that a team at one location on the ship goes through when the ship leaves a harbor. All events in the diagram are the commands, notifications, and confirmations that are communicated between the team officer and the commanding officer on the bridge.

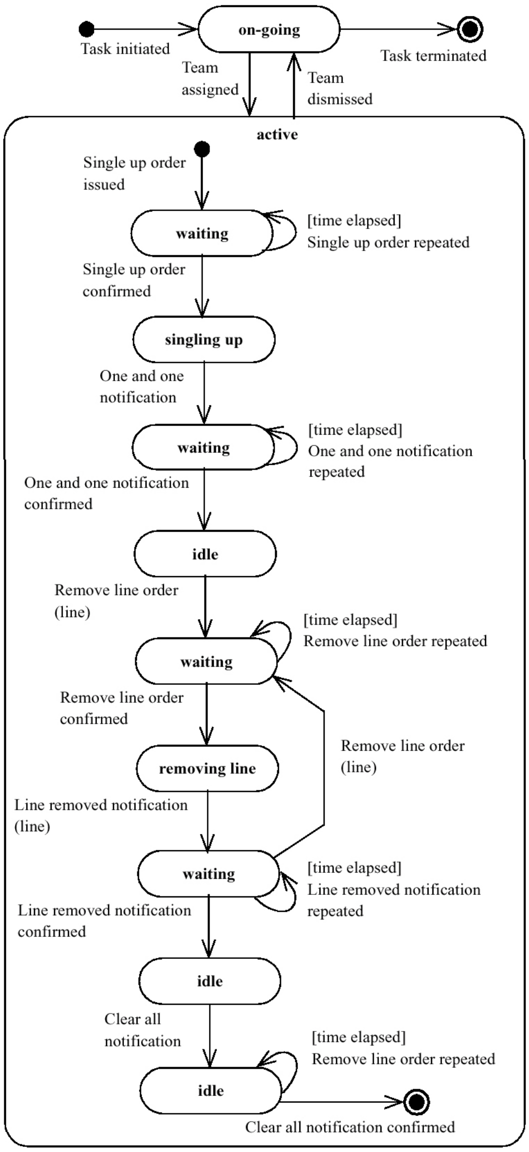

Fig. 5. Statechart diagram for the Let go class

4.2 Application domain analysis

The analysis of the application domain consisted of definition of actors and description of use cases that these actors participated in. The use cases were applied to describe the complete sequence of using the system for the commanding officer on the bridge and for the officers in each team. Based on the use cases a list of functions was developed. The functions are the building blocks that are used to describe how the user initiates processing on the computerized system.

4.3 Evaluation of analysis process

The analysis of the problem domain was conducted in a straightforward manner. There were no major methodical problems. The development of the class diagram took most of the time. The three overall clusters (Places, Tasks, and Crew where Crew was later renamed to Personnel) were not defined from the beginning, but it was quite clear that there were separate issues in the problem domain that might be modeled with clusters like these. There were physical locations where tasks were carried out. For example, in the case of arrival in a harbor, there would be two teams handling mooring lines at the bow and stern. These tasks were similar but carried out at two different locations. The team that carried out the tasks should also be modeled, and finally it was necessary to model the tasks themselves in order to be able to monitor progress.

The work on the Behavior activity quickly uncovered an important difference between the classes in the problem domain. The classes that were used to model the physical part of the ship were very static. Once the locations are defined, these objects do not usually change. They might change over a very long period of time, e.g. if the ship gets new equipment which is located and operated in a specific location, but generally these classes have no dynamic behavior; the objects are just there as a kind of hook where we can put teams and tasks. Thus we made no behavioral diagrams for these classes.

The behavior of the Team class was quickly defined. It was the kind of class where a lot of events can occur without any limitations on sequence. The Task class was described as the generalized pattern that applies to all tasks. After that, we went into describing the behavior of the individual tasks. This turned out to be quite difficult. The problem is that there is a typical sequence of behavior. But there may be many deviations from that. The other problem relates to the granularity of tasks. For example, when mooring lines are lifted off the bollard on the quay, they are winched up onto the ship. This can be seen as two tasks where one develops into the next or as one large task. This comes down to the question whether it is one or two classes.

The analysis of the application domain was very simple. The reason was that there are only two actors. There were a number of use-cases, but they were also quite simple. It seemed hard to get a lot out of this activity.

5 Experimental Prototype

An experimental prototype of the Handheld Maritime Communicator was designed and implemented. The present version of the prototype only supports communication concerning the task of letting go the lines when departing from the quay.

5.1 Applying analysis to interface design

Based on the object-oriented analysis discussed in section 4, we began to specify user interface requirements for the prototype. Assisted by the class diagram, we decided that each specializations of the general Task class (mooring, letting go and watching) should have a dedicated interface based on a general design for supporting coordinative communication. Each interface should then be tailored to the specific requirements of the task. As shifting between tasks during their execution was not identified as a significant function of the Task class, we decided that this function could be placed out of immediate focus e.g. in a menu.

Looking at the class diagram, we furthermore decided that different interfaces for the commanding officer and the officers on deck could perhaps be appropriate as the Commanding officer and Officer classes aggregates from fundamentally different classes (Ship and Task respectively). We were, however, not able to decide from the class diagram exactly how the interfaces should differ and which information and functionality should be available to which actors and which should be omitted.

Finally the class diagram helped us realize the role of the Location class as a mediator between the Ship and Task classes. Thus the location of a mobile device enables us to deduce the role of its user in an ongoing task or in overall relation to the ship. Subsequently a corresponding interface for that role or relation can be presented on the device. During the task of letting go, for example, a mobile device located on the aft deck should thus present the interface for the second officer, while a mobile device on the bridge should make available the interface for the captain/commanding officer. Supporting this functionality, we decided to include into the user interface a facility for specifying the user's location by means of simple menu selection. As we had not on the basis of our analysis identified a need for changing physical location *during* the execution of a task, we decided to place this functionality out of immediate focus e.g. on a task-startup screen or in a menu. In relation to tasks involving more frequent changes of actors' physical location, this design may not be appropriate. Alternatively, automatic positioning technology could be considered.

Looking at the statechart for the Let go class, we quickly realized that the overall task of letting go the lines follows a pattern of "subtasks" being carried out within a sequence of little variance. We thus decided to support partial automation of the let go task by deducing the most likely next steps of the sequence at any given point of the task. These steps should then be represented in a prioritized list of choices on the interface located on the bridge.

Furthermore, the statechart revealed that the carrying out of a subtask of the Let go class follows a general pattern with 5 different states of each command:

1. Not issued
2. Issued
3. Issued and confirmed
4. Completed
5. Completed and confirmed

Looking at these states from a temporal point of view, they can be grouped into 3 overall categories of belonging to the *future* (1), the *present* (2, 3, 4) or the *past* (5). From this insight, we correspondingly decided to divide the user interface graphically into three overall areas, each concerned with future, present and past commands/communication. From the OOA&D analysis, we were, however, not able to decide which (if any) of these categories would be of most importance to the commanding officer and the officers on deck respectively. Neither were we able to decide much on the specific design of each graphical area. Future commands should, of course, be selectable (e.g. a menu, list or graphical representation) while present and past commands should only be viewable. Also as the area concerning "present" communication is used for displaying three different states (2-4) of each command being carried out, this area should be designed to support a clear differentiation between these states. As unconfirmed commands are repeated after a given period of time, we also decided that timers or timestamps should be attached to all commands being issued.

Finally, the statechart made us realize that the desired graphical representation of the ship's mooring lines should be rich enough to capture both the state of mooring (past + present) as well as the current state of issued commands (present). We thus decided to use a combination of schematic drawings of the ship and its mooring lines along with textual labels. We also considered the use of animated lines and winches (but this was not implemented in the first prototype).

5.2 Additional interface design considerations

Though valuable in the implementation of the prototype interface, the general ideas originating from the OOA&D analysis above did not provide sufficient input to cover all aspects of the specific design. In addition we therefore also surveyed the design of related systems such as chat applications, SMS messaging, e-mail and newsgroups for inspiration.
One of the major issues, which were not immediately clear from the OOA&D analysis, was the fact that typically a number of commands or subtasks are carried out in parallel rather than in sequence. This results in rather fragmented communication. When shifting directly to a textual communication protocol, this fragmentation remains (as seen in e.g. chat applications) and has to be addressed through alternative design. Inspired by the handling of parallel threads of communication in newsgroups by relating new contributions thematically to prior ones, generating a simple three structure, we decided to represent all communication related to each issued command in a similar way.
Finally we did a lot of experimenting with specific solutions to the problems of e.g. representing the states of present communication in a simple and comprehensible manner. Also the use of colors and sound cues gave rise to a lot of experimenting. Due to the limited space available on the screen, designing the representation of the ship to include both textual and graphical information while keeping it very simple was also a challenge.

5.3 Hardware

The prototype was targeted at a Compaq Ipaq 3630 handheld computer with 32MB RAM and a color display of 240x320 pixels running Microsoft PocketPC. Apart from pen and touch screen, this device facilitated interaction by means of a five-way button located below the display suitable for one-handed interaction. The prototype-setup consisted of three Ipaqs connected through an IEEE 802.11b 11Mbit wireless network. One device was designed for the captain on the bridge while the other two were designed for the first and second officer located on the fore and aft deck respectively. The prototype was implemented using Microsoft Embedded Visual Tools and the PocketPC SDK.

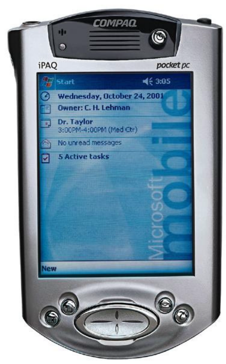

Fig. 6. Compaq Ipaq handheld computer

5.4 Architecture

The application running on the captain's device works as a central server containing all commands and a formalized representation of the communication pattern of the task. The applications running on the devices on deck logs on to the server and identifies their physical location. During operation, function calls and unique command identifiers are exchanged over the network via TCP/IP. All communication on the network is broadcasted to all devices but is processed and represented differently on each device in accordance to their physical location: bridge, fore or aft and the desired language. The language used on each device is defined in an external text file, which can be modified to match any desired language.

5.5 Interface design

The design consists of two similar but different interfaces for use on the bridge and on deck.

On the bridge, the interface is divided into four sections (see figure 7):

1. Visual representation of the ship and the mooring lines
2. List of completed communication threads
3. List of ongoing threads of communication
4. List of unexecuted (future) commands

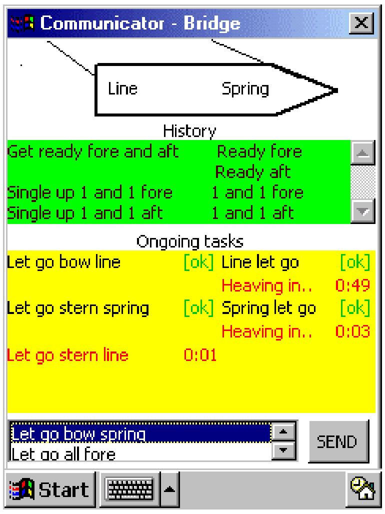

Fig. 7. The interface on the bridge

In the bottom of the interface, unexecuted commands representing tasks yet to be carried out are displayed on a list. The order of the list corresponds to the standard sequence of the overall operation. By default, the most likely next step of the sequence is highlighted. Commands only appear on the list when appropriate in relation to the pattern of the task. The user can browse this list by pressing the five-way key of the device up or down. A highlighted command is executed (send) by pressing the center of the five-way button. This causes the command to be removed from the list and the most likely next command to be highlighted. This is illustrated in figure 8.

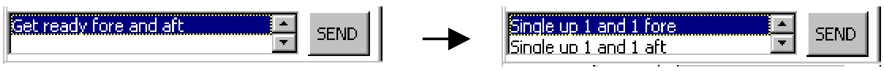

Fig. 8. Executed commands being removed from the command list while new commands appear

When a command is executed, it appears in red letters on the list of ongoing threads of communication representing uncompleted tasks. Next to it, a counter displays the time passed while waiting for confirmation (figure 9a). When a command is confirmed, it turns black and the timer is substituted by the text “[ok]” in green letters followed by a short statement describing the current activity (e.g. “Singling up...”) in red letters. A counter next to this displays the time passed since confirmation (figure 9b). When a task is completed, a short

statement in green letters (e.g. “1 and 1 fore”) substitutes the statement of activity and the captain is prompted for confirmation (figure 9c). When the completion of a task is confirmed, this is indicated by the text “[ok]” in green letters (figure 9d). The red color is thus used to indicate the parts of the operation that is being carried out at the moment, much like alarms requiring attention. The green color is used for indicating confirmations and completed tasks.
Completion of tasks, which automatically initiates a new task (e.g. heaving in lines when they are let go), results in an additional line of text in the corresponding thread, displaying the current activity.

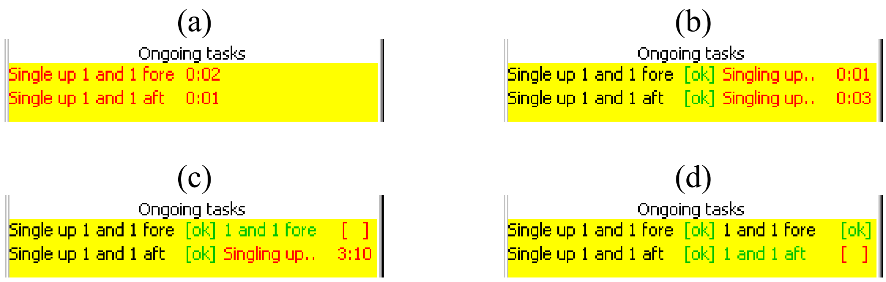

Fig. 9. Sequence of commands being executed, confirmed, completed and confirmed

When the captain confirms the completion of a task, the corresponding thread of communication is removed from the list of ongoing threads. Instead, the thread is added at the bottom of the list of completed communication threads. As this list represents a history of finished tasks for quick reference, the textual representation is simplified as much as possible by removing the “[ok]” representation of confirmations, the counters and by displaying all text in black letters.

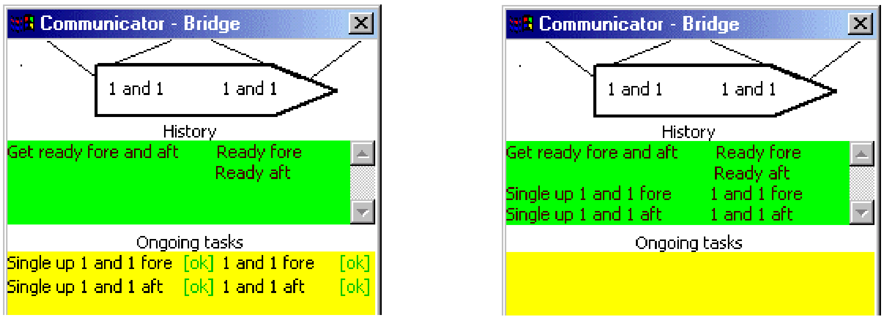

Fig. 10. Sequence of completed threads being transferred to the history list

When the list is full, it automatically scrolls the oldest commands and statements out of immediate sight (figure 10). The list can then be scrolled either by means of the touch screen or by selecting it using a dedicated function key and subsequently pressing the five-way key on the device up or down.

The visual representation in the top of the interface represents the present status of the mooring lines. It consists of a simple schematic drawing of the ship and the lines attached to the quay at present time. Additionally, the present status fore and aft is represented textually. A possible sequence is illustrated in figure 11.

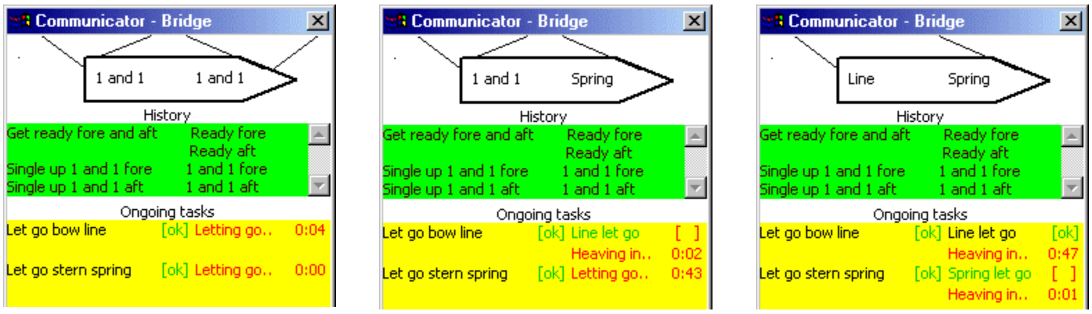

Fig. 11. Sequence of the visual representation reflecting the status of mooring

On deck, the interface is very similar to that on the bridge, thus providing the first and second officers with a view on the present status of the mooring and a list of all past and ongoing communication among the distributed actors (figure 12).

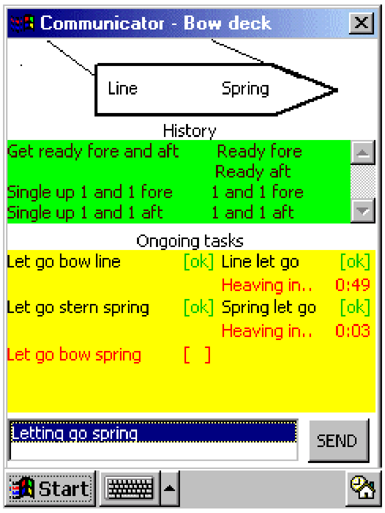

Fig. 12. The interface on the deck

In the list of ongoing tasks, however, the interface on deck requests commands from the bridge (like e.g. “Let go bow spring”) being confirmed and displays a counter showing the time past while awaiting confirmation of completed tasks from the bridge.
Correspondingly, the list of commands available on deck only contains commands appropriate from the specific location of the user (fore or aft) at the given state of the sequence (e.g. “Letting go spring...” for confirmation of the latter command or “Spring let go” for reporting the completion of the task”).

Generally, sound cues are used to direct attention towards the device. When a command appears on the devices on deck, a brief high-pitch two tone alert is played. When the command is confirmed, a brief low-pitch two tone alert is played on the bridge. Finally a short four-tone melody is played on the bridge whenever a task is completed (a line has been let go or is home).

6 Evaluation

Section 4 and 5 have described the analysis and design of a handheld device that is intended to replace existing radio communication devices such as VHF walkie-talkies. The new device embodies the idea of shifting from the existing radio-transmitted natural language protocol to network-based communication by selections from pre-defined menus. Below, we will evaluate the design by comparing it to the problems with the existing communication protocol that were emphasized section 3.

1. The network-based interchange of standardized phrases removes the potential for loss and distortion of messages that is experienced with the existing radio communication. This improvement limits the risk of misperceptions and misunderstandings and thereby increases safety.

2. Communication of standardized messages can be made persistent by keeping records of each command that has been interchanged. When communication follows a predefined pattern, this pattern can be represented visually, thereby supporting overview and reducing complexity. The visual representation can be extended with threads that illustrate series of parallel tracks or threads.

3. The formalization of commands and patterns of communication contribute to reduce the complexity of a visual representation and the administration of commands can be partly automated. Completed dialogs and interactions can be given less attention and focus can be turned solely towards on-going dialogs and interactions. Automation can be used to remove irrelevant options and suggesting appropriate commands.

4. From our present evaluation of the design it is not possible to decide whether the new device will reduce or increase the time spent on communication. The potential reductions originate from the fact that appropriate commands may typically be executed by means of only one or two clicks on a button. In addition, incoming commands are relatively short, they appear visually at a natural location of the interface in relation to preceding commands and they are accompanied by a sound cue. Potential increases in the time spent on communication originate from the extent to which the standardized messages fail to fulfill the need for communication.

5. Formalizing communication into a series of standardized commands that are interchanged through a computerized system facilitates seamless translation between languages and transparent interaction across the language barrier that is experienced between actors with different nationalities. The exchange of simple identifiers for predefined commands between the distributed computers enable the user to receive a simple translation of each message in whatever language desired.

6. From our present evaluation of the design it is not possible to decide whether the new device will reduce bottlenecks in the communication.

7. When commands and patterns of communication about a specific task are formalized, a formal representation of the current state of the task or domain can be created and maintained. Such representation can be used as foundation for a visual representation supporting overview and could optionally be combined with other computer-based data.

While the design of the prototype thus overcomes key problems experienced in present technology it does, however, also introduce at least two obvious limitations:

- Lack of communication flexibility
- Demand for visual attention

The present flexibility inherent in natural language communication is not supported equally well by the new device. Although communication in high-risk environments may by highly standardized, as shown in the analysis, a situation may arise, in which it is necessary to deviate from the regular procedure and communicate something out of the ordinary. For this purpose, the use of a natural language communication protocol facilitates seamless shifting to meta-communication or to a more conversation-like format. The purpose of the new handheld device is not to replace natural language communication completely, but rather to replace as much of it as possible but still have an option for complementing the standardized communication with natural language in unusual situations.

Unlike spoken communication, textual communication suffers from the demand for visual attention. This limitation is twofold. First, having to monitor a computer interface distracts visual focus from the task being carried out [4]. In a high-risk environment, this is not always an option. Second, visual representations lack the ability of attracting attention from a user not focused towards it. While the limitation of attracting focus may be supported by the use of non-speech audio cues [2,3,5], optionally supplementing a visual representation with synthetic speech [6,36] may remedy the limitation of demand for visual focus. Alternatively, mobile head-mounted displays could be used for locating a graphical interface within the visual field of view of the user at all times [37,38].

7 Conclusions

This paper has presented the design of a handheld device to support communication in the high-risk situation of performing docking maneuvers with a very large container ship. It has been argued that this design overcomes several of the problems that are presently experienced with the existing radio-based communication protocol.

The design has been developed by means of a combination of fragments from an object-oriented method and general guidelines on human-computer interaction. Thus the design experiment illustrates the potential gain of applying such a combination.

In order to gain more insight into the usability of the presented information system, the design should be made subject to usability evaluations. As it is obviously problematic to conduct real-world evaluations of early prototypes in high-risk environments, the use of lab-based evaluation methods must be explored. One approach could be the use of a ship simulator for the creation of a (safe) virtual environment. Another approach could be the use of simple low-fidelity mockup tools for representing the ship, the harbor and the mooring lines.

We are presently conducting a series of such evaluation. Papers concerning the usability of the presented design as well as comparing evaluation methods within the context of mobile device interfaces for high-risk environments are forthcoming.

References

1. Bergman, E. (ed.) (2000). *Information Appliances and Beyond: Interaction Design For* Consumer *Products,* London, Morgan Kaufmann.
2. Brewster, S. (2002). Overcoming the Lack of Screen Space on Mobile Devices. In Personal and Ubiquitous Computing vol. 6:188-205, London, Springer-Verlag.

3. Brewster, S. and Walker, V. (2000). A. Non-Visual Interfaces for Wearable Computers. In Proceedings of IEE workshop on Wearable Computing (IEE, London) IEE, 00/145.
4. Holland, S. and Morse, D. R. (2001). Audio GPS: spatial audio in a minimal attention interface. In Proceedings of the 3rd workshop on MobileHCI, IHM-HCI, Lille, France.
5. Pirhonen, A., Brewster, S. and Holguin, C. (2002). Gestural and Audio Metaphors as a Means of Control for Mobile Devices. ACM CHI Letters vol. 4(1): 291-298.
6. Sawhney, N. and Schmandt, C.: Nomadic Radio: Speech and Audio Interaction for Contextual Messaging in Nomadic Environments. In ACM Transaction on Computer-Human Interaction, vol. 7(3): 353-383 (2000)
7. Hinckley, K., Pierce, J., Sinclair, M., Horvitz, E. (2000). Sensing Techniques for Mobile Interaction, ACM CHI Letters vol. 2(2): 91-100.
8. Cheverst, K. et al. (2001). Investigating Context-aware Information Push vs. Information Pull to tourists. In Proceedings of MobileHCI'01 workshop on HCI with Mobile Devices, IHM-HCI, Lille, France
9. Kjeldskov, J. (2002). "Just-in-place" Information for Mobile Device Interfaces. To appear in Proceedings of MobileHCI'02, Pisa, Italy .
10. Andersen, P. B. and Nowack, P. (2003). Modeling Moving Computers. To appear in Qvortrup, L. and Andersen, P. B. (eds.) *Virtual Applications: Applications With Virtual Inhabited 3D Worlds,* Springer-Verlag, London
11. Hüttenrauch, H. and Norman, M. (2001). PocketCERO – mobile interfaces for service robots. In Proceedings of MobileHCI'01 workshop on HCI with Mobile Devices, IHM-HCI, Lille, France.
12. Nielsen, C. and Søndergaard, A. (2000). Designing for mobility - an integration approach supporting multiple technologies. In Proceedings of NordiCHI 2000 (CD-ROM), 23-25 October 2000, Royal Institute of Technology, Stockholm, Sweden.
13. van den Anker, F. W. G. and Lichtveld, R. A. (2000). Early Evaluation of New Technologies: The Case of Mobile Multimedia Communications For Emergency Medicine. In Vincent C. and de Mal B (eds.) *Safety in Medicine.* Qxford: Elsewier Science.
14. Palanque, P. Paterno, F. and Wright, P. (1998). Designing User Interfaces for Safety-critical Systems. In proceedings of the conference on CHI 98 summary. Los Angeles, USA, ACM Press.
15. Ladkin, P. B. (1998). The Ariane 5 Accident: A programming Problem. University of Bielefeld, Article RVS-J-98-0. Available at http://www.rvs.uni-bielefeld.de/publications/
16. Ladkin, P. B. (1998). Computer-Related Incidents with Commercial Aircrafts: A compendium of Resources. Reports, Discussion and Commentary. University of Bielefeld. Available at http://www.rvs.uni-bielefeld.de/publications/Incidents/
17. Leveson, N. G. (1995). *Safeware: System Safety and Computers.* Addison-Wesley Longman Publ. Co., Inc., Reading, MA.
18. Leveson, N. G. (1990). Evaluation of Software Safety. In Proceedings of the 12th International Conference on Software Engineering. IEEE Computer Society Press, 223-224.
19. Parnas, D. L., van Schouwen, J. and Kwan S. P. (1990). Evaluation of Safety-Critical Software. In Communications of the ACM vol. 33(6) pp 636-648.
20. Redmil, F. and Rajan, J. (1996). *Human Factors in Safety-Critical Systems.* Butterworth-Heinemann.
21. Andersen, P. B. (2002). The Visible Computer. To appear.
22. Rasmussen, J. (1983). Skills, Rules and Knowledge: Signals, Signs and Symbols and Other Distinctions in Human Performance Models. IEEE Transactions on Systems, man and Cybernetics vol. 13(3).
23. Rasmussen, J. (1986). *Information Processing and Human-Machine Interaction.* New York, North-Holland.
24. Norman, D. (1990). The 'Problem' With Automation : Inappropriate Feedback And Interaction Not Over-automation. In Broadbent D.E. et al. (eds.) *Human Factors In Hazardous Situations,* 137-145, Oxford, Clarendon Press.
25. Lind, M. (1990). Representing Goals and Functions of Complex Systems. Technical report, 90-D-38, Technical University of Denmark.
26. Lind, M. (1994). Modeling Goals and Functions in Complex Industrial Plants, Applied Artificial Intelligence vol. 8(2): 259-283.
27. Lind, M. (1999). Plant Modeling for Human Supervisory Control. Transactions of the Institution of Measurement and Control vol. 21(4/5): 171-180.
28. Mackay, W. E. and Fayard, A. (1998). Designing Interactive Paper: Lessons from Three Augmented Reality Projects. In Behringer et al. (eds.) *Augmented Reality: placing artificial objects in real scenes: Proceedings of IWAR'98.* Natick, A K Peters, pp 81-90.
29. Mackay, W. E. (1999). Is Paper Safer? The role of Paper Flight Strips in Air Traffic Control. ACM Transactions on Computer-Human Interaction vol. 6(4): 311-340.
30. Camp, P. J., Hudson J. M. Keldorph, R. B. Lewis, S. and Mynatt, E. D. (2000). Supporting Communication and Collaboration Practices in Safety-Critical Situations. In Proceedings of ACM CHI 2000 (short paper) pp 248 - 250. The Hague, Netherlands.
31. Fields, R. Paterno, F., Santoro C. and Tahmassebi, S. (1999). Comparing Design and Options for Allocating Communication Media in Cooperative Safety-Critical Contexts: A Method and a Case Study. In ACM Transanctions on Computer-Human Interaction, 6(4), Special Issue on Interface Issues and Designs for Safety-critical Interactive Systems.

32. Nielsen, M. (2000). Letting go the lines: Departure from the Felixstowe harbor. Center for Human-Machine Interaction. Research report CHMI-4-2000.
33. Andersen, P. B. and May, M. (2001). Tearing up Interfaces. In Liu K. et al (eds.) *Information, organization and technology. Studies in organizational semiotics.* Boston, Kluwer.
34. Mathiassen, L., Munk-Madsen, A., Nielsen, P. A., and Stage, J. (2000). *Object-Oriented Analysis and Design.* Aalborg, Marko.
35. Mathiassen L., Munk-Madsen, A. Nielsen, P. A., and Stage, J. (1995). Modeling Events in Object-Oriented Analysis. In Patel, D. et al. (Eds.), *Object-Oriented Information Systems*. Pp. 88-104. Springer-Verlag, Berlin.
36. Lai, J., Cheng, K. Green, P. and Tsimhoni, O. (2001). On the Road and on the Web? Comprehension of Synthetic and Human Speech While Driving. ACM CHI Letters vol. 3(1): 206-212.
37. Feiner S., MacIntyre B, Höllerer T. (1999). Wearing It Out: First Steps Toward Mobile Augmented Reality Systems. In Onto, Y. and Tamura, H. (eds.) *Mixed reality – Merging Real and Virtual Worlds,* Berlin, Springer-Verlag.
38. Kjeldskov, J. (2003). Lessons From Being There: Interface Design for Mobile Augmented Reality. To appear in Qvortrup, L. and Andersen P. B. (eds.) *Virtual Applications: Applications With Virtual Inhabited 3D Worlds,* Berlin, Springer-Verlag.