

An Underwater Safety-Critical Mobile Communication System

by

Jennifer Wong

B. Sc., University of Victoria, 2007

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Computer Science

© Jennifer Wong, 2009

University of Victoria

All rights reserved. This thesis may not be reproduced in whole or in part, by
photocopy or other means, without the permission of the author.

An Underwater Safety-Critical Mobile Communication System

by

Jennifer Wong

B. Sc., University of Victoria, 2007

Supervisory Committee

Dr. Jon Muzio, Supervisor

(Department of Computer Science)

Dr. Melanie Tory, Departmental Member

(Department of Computer Science)

Dr. Yvonne Coady, Departmental Member

(Department of Computer Science)

Supervisory Committee

Dr. Jon Muzio, Supervisor
(Department of Computer Science)

Dr. Melanie Tory, Departmental Member
(Department of Computer Science)

Dr. Yvonne Coady, Departmental Member
(Department of Computer Science)

ABSTRACT

Recreational scuba diving is a highly social activity where divers are encouraged to work in groups of two or more people. Though collaborative, divers are unable to freely and naturally communicate. Additionally, the distortion of sensory information (e.g. distances and sounds cannot be judged as accurately underwater) affects the ability to keep track of critical information which impairs their ability to engage in this underwater world. We have studied and designed a fault tolerant system, including the software, the device, and the network, to foster underwater communication. We studied the technology required, the software design for both single user and multiple users, as well as, the network design in order to support such a system. In the thesis, we have set up and analyzed the result of three user studies and a simulation to investigate the viability of the proposed design.

Contents

Supervisory Committee	ii
Abstract	iii
Table of Contents	iv
List of Tables	x
List of Figures	xi
Acknowledgements	xiv
1 The Land of Atlantis	1
2 The Universe of Fault Tolerance, HCI, and CSCW	4
2.1 Fault Tolerance	5
2.1.1 Redundancy	5
2.1.2 Fault Detection	8
2.1.3 Fault Containment	8
2.1.4 Reconfiguration	8
2.2 Human Computer Interaction	9
2.2.1 Design Process	9
2.2.2 Is Usability Really Common Sense?	10
2.3 Computer Supported Cooperative Work	12

2.3.1	Time Space Taxonomy	13
2.3.2	Awareness	13
2.4	Conclusion	15
3	Communication Technology	17
3.1	Transmission Methods	18
3.1.1	Acoustic	18
3.1.2	Radio	19
3.1.3	Radar and Sonar	19
3.2	Related Work	20
3.2.1	High-Speed Optical Transmission	20
3.2.2	EvoLogics - Hydroacoustic modems with S2C Intelligent Underwater Telemetry	20
3.2.3	Acoustic Modem	21
3.2.4	Underwater Robotics Communications	22
3.2.5	Float Buoy Ranging System	25
3.3	Communication for Dive Computers	26
3.4	Conclusion	26
4	Network	28
4.1	Proactive Routing	28
4.2	Reactive Routing	29
4.3	Hybrid Routing	29
4.4	Related Work	29
4.4.1	Mobile Ad Hoc Wireless Networks	30
4.4.2	Sensor Networks for Health and Safety Monitoring	30
4.4.3	Intentional Naming System	31

4.4.4	Seamless Networking	32
4.5	Transmitters	33
4.6	The Design of the Dive Computer Communication Network	39
5	Human Computer Interaction	42
5.1	Case Scenarios	42
5.1.1	Direction	43
5.1.2	Bottom Time and Depth	43
5.1.3	Tank Pressure	43
5.2	Variables	44
5.3	Current Approach	44
5.3.1	Dive Table	45
5.3.2	Dive Planning Software	45
5.3.3	Dive Computer	45
5.3.4	Communication Device	48
5.4	The Missing Information	49
5.4.1	Derived Data	50
5.4.2	The Missing Functionality	50
5.5	Operational Modes	51
5.5.1	Interaction Zen	51
5.5.2	Display Modes	51
5.6	Colour	52
5.7	Graphical Information Representation	53
5.7.1	Depth and Time	53
5.7.2	Direction	54
5.7.3	Tank Pressure	55
5.8	Design Choices	55

5.8.1	Graphics	56
5.8.2	Screen Navigation Buttons	56
5.9	Display of Choice	57
5.9.1	Dive Mode	58
5.9.2	Pre-Dive Mode	59
5.9.3	Surface Mode	61
5.10	Physical Device	62
5.11	Warnings	63
5.12	Conclusion	64
6	Computer Supported Collaborative Work	65
6.1	Research Goals	67
6.1.1	Underwater Collaboration	67
6.1.2	Communication	68
6.1.3	Increase Safety	68
6.1.4	Increase Awareness	69
6.2	Related Work	69
6.2.1	Mobile Devices Supporting Collaborative Work	69
6.2.2	Awareness	70
6.2.3	Instant Messaging	71
6.2.4	Fire Fighters Communication System	72
6.2.5	Mobile Collaboration System	72
6.3	Limitations and Constraints	73
6.4	Design	74
6.5	Conclusion	76
7	Implementation	77

7.1	Paper Prototype	77
7.2	Software Mockup	78
7.3	Simulation	79
7.4	Conclusion	82
8	User Studies and Simulation	83
8.1	User Study 1: Usability	83
8.1.1	Study Participants	83
8.1.2	Study Methodology	84
8.1.3	Questions and Reasoning	84
8.1.4	Results	85
8.2	User Studies 2: CSCW (Pilot)	86
8.2.1	Methods	86
8.2.2	Execution	88
8.2.3	Results	89
8.3	User Studies 3: CSCW	90
8.3.1	Methods	90
8.3.2	Execution	91
8.3.3	Results	92
8.4	Simulation	92
9	Analysis	102
10	Conclusion	106
A	Simulation Information	108
B	Software Flow	110

C Paper Prototype	111
D HCI Software User Study Evaluation Questionnaire	115
E CSCW Paper Prototype Evaluation Survey	117
F CSCW Software Prototype Evaluation Survey	119
G NS2 Implementation Details	121
Bibliography	122

List of Tables

Table 3.1 LinkQuest UWM-100 specification [1].	22
Table 4.1 An example of pair-and-a spare.	40
Table 8.1 A summary of result for user study 3.	95
Table 8.2 Average packet drop percentage.	97
Table 8.3 Median packet drop percentage.	98
Table 8.4 Average transmission time in seconds.	100
Table 8.5 Median transmission time in seconds.	101

List of Figures

Figure 2.1	Levels of abstraction.	4
Figure 2.2	An example of TMR. All three modules perform the same task. When there is a discrepancy between the outputs, the output with most matches is used. [2]	6
Figure 2.3	An example of N-modular standby sparing. While one module is provided the actual output, all other $n-1$ modules act as spares. [2]	7
Figure 2.4	An example of time redundancy. [2]	8
Figure 2.5	The design process circle for HCI. [3]	10
Figure 2.6	PCPAL 3-in-1 remote control, mouse, and presenter [4].	11
Figure 2.7	Kensington wireless presenter 33374 [5].	12
Figure 2.8	Groupware time space matrix [6].	13
Figure 2.9	Screenshots of TouchGraph.	16
Figure 3.1	An example of ubiquitous system that can be taken underwater when secured in the underwater housing [7].	17
Figure 3.2	The S2C-180 hydroacoustic modem [8].	21
Figure 3.3	The S2C-280 hydroacoustic modem [8].	21
Figure 3.4	The S2C-280 hydroacoustic modem [1].	22
Figure 3.5	Results of Nagothu’s experiment 2 [9].	24
Figure 3.6	The float buoy ranging system proposed by Kurano <i>et al.</i> [10].	26

Figure 4.1	Triangulation with one transmitter.	34
Figure 4.2	Triangulation with two transmitters.	35
Figure 4.3	Triangulation with three transmitter.	36
Figure 4.4	Pair-and-a-spare design [11].	40
Figure 4.5	An overview of the communication network design.	41
Figure 5.1	The console design is the most common design of dive computers [12].	46
Figure 5.2	A dive computer in the form of a wristwatch [13].	46
Figure 5.3	Dive computer integrated with the mask [14].	47
Figure 5.4	An example of what the LCD looks like from inside the mask [14].	48
Figure 5.5	A newly developed underwater communication device [15]. . .	48
Figure 5.6	Graphical representation of depth and time.	53
Figure 5.7	The compass.	54
Figure 5.8	Vertical direction indicating, (a) the display is facing upward. (b) the display is facing downward (c) the display is facing in the direction of the arrow at 45° from face-up (d) the display is facing the direction of the arrow perpendicular to the horizon (e) the display is facing in the direction of the arrow at 45° from face-down.	55
Figure 5.9	Device orientation according to graphics in Figure 5.8.	55
Figure 5.10	Graphical representation of tank pressure.	56
Figure 5.11	Two examples of original design, (a) and (b), with the “Back” button placed on the right, and the final placement (c).	57
Figure 5.12	Main screen of dive mode.	58
Figure 5.13	Detailed view for depth and time.	58
Figure 5.14	Detailed view for tank pressure.	59

Figure 5.15	Detail view for direction.	59
Figure 5.16	Diver has to enter the planned maximum depth.	60
Figure 5.17	Diver is instructed to set their intended direction.	60
Figure 5.18	Dive computer picks up information regarding airflow rate and tank pressure with the sensors that is part of the regulator. . .	61
Figure 5.19	Test result screens.	61
Figure 5.20	Pairing dive computers.	62
Figure 5.21	Main screen for surface mode.	62
Figure 5.22	There are seven valid dive records currently existing on the dive computer.	63
Figure 5.23	An example of a complete record.	63
Figure 5.24	The dive computer.	64
Figure 6.1	The collaborative development process.	71
Figure 6.2	A screenshot of the group stat screen.	75
Figure 7.1	Examples of typical dive pattern.	79
Figure 8.1	A modified version of the PADI dive log.	86
Figure 8.2	Average and median packet drop rate from 5 nodes to 105 nodes.	96
Figure 8.3	Average and median packet transmission rate from 5 nodes to 105 nodes.	99
Figure 9.1	The suggested message alert screen.	104

ACKNOWLEDGEMENTS

This chapter is finally complete. The baby who sprint into this world some time ago is finally going to venture into the next chapter of adventures. Each page was filled with colors and each line was filled with memories. This would not have been possible without the support of many people.

I wish to thank all of the faculty who helped educate this mind. In particular: Dr. Jon Muzio, my supervisor, for his patience, support, guidance, mentorship, and thinking-outside-the-box; his humor and wisdom has helped me stay sane and grow in multiple dimensions. Dr. Melanie Tory for being advantageous and willing to dive into crazy ideas. Dr. Yvonne Coady for her mentorship and for always pushing me a little further than my limits. It has led me to strive to make the impossible possible. Dr. Ulrike Stege for always having an open heart and ears. Dr. Margaret-Anne Storey for her tips and career guidance, which has opened up a new playing field. Without any one of them, this thesis would not be shaped the way it is.

I would also like to thank Dr. LillAnne Jackson, one of many who urged me to apply for graduate school. Without her, this thesis would not exist today. Dr. Alex Thomo for the computation power; without which, my analysis would have never been finished.

Thanks and love goes to my family, to whom I dedicate this thesis to for their immeasurable support. Mom and dad, who relit my inner fire with their talks and reminders that fun is as important as work. My little sister and best friend, Jessica, whom has read my thesis almost as many times as Jon and myself. She is also the one who would stay up with me all night doing crazy things, like playing 21 consecutive sets of singles at VRC, trying to beat a certain scene in a Wii game, and camping in the living room until we finally outgrew the tent.

Finally, many thanks to all my friends who have always been there with me through thick and thin. Special mention to: Adrian Schroeter, who repeatedly reassured me that there is an end to this thesis, for his endless amount of love and support and the Easter penguin hunt; Maryam Daneshi the network guru; Katherine Gunion, for making all-nighters in the ECS fun; Tom Lai who insisted that I do “field tests” often; CBers, with whom we find 20 people laughing to tears every time!

“In everyone’s life, at some time, our inner fire goes out. It is then burst into flame by an encounter with another human being.” - Albert Schweitzer

Chapter 1

The Land of Atlantis

While humans explore new lands and areas above water, we take oral communication for granted, but when we take exploration underwater, oral communication immediately becomes a privilege.

Recreational scuba diving is one of many different types (e.g. commercial diving, military diving) of scuba diving. As the word “recreational” suggests, it is done for fun and leisure. As much enjoyment as this brings, it also comes with safety risks; it is recommended that divers should dive in groups of two or more, thus making this activity highly collaborative.

In the current setting, recreational divers rely on hand signals to communicate. There are two obvious problems to this method: (1) it requires the communicatee to look at the communicator; (2) the meaning of hand signals must be understood by both parties. Problem 1 is inevitable since divers are constantly swimming forward being absorbed by the ever-changing sceneries. The difficulty of this problem increases proportionally with the group size.

Water not only takes away the privilege of easy intake of oxygen and verbal expression, it turns the world upside down and takes orientation to a whole new level.

In an area of air particles, we always know the up and down directions due to the gravitational force. However, in a space full of water particles, whose density is heavier, we can no longer be certain of our orientation, especially when diving in the dark, at a deeper depth, or in a wreck/cave; we are only really certain when we hit the water surface or the surface of the earth. Additionally, because water has a much higher density than gas, technologies used for completing the same transmission task in the two mediums can be very different.

Due to the aforementioned issues, this underwater mobile collaborative setting is calling for alternatives. The objective of this thesis is to design a network and a device to be used in the underwater environment, such that it allows inter-device communication via direct connections (up to 100 meters spherical range) as well as out-of-range communication via surface units. Each device maintains up-to-date information about other participants in the group. In addition, the device should be reliable and be able to seamlessly join and disconnect from the network. The device should also fail in a graceful manner, such that the system can discontinue without disruption to other systems or any compromises to safety. Furthermore, the device should work as a standalone machine, as well as a collaborative tool.

In Chapter 2, a general overview of the three main topics of this thesis – Fault Tolerance, Human Computer Interaction (HCI), and Computer Support Cooperative Work (CSCW) – is given. Following is a discussion on current underwater communication technology in Chapter 3.

In Chapters 4, 5, and 6, we touch upon the network design, HCI, and CSCW respectively. In the network design, we have looked at various current technologies and protocols that could be used to support our system. For HCI, we focus on the usability of the system. We have proposed a dive computer design that has never been seen in practice before. Furthermore, in the CSCW chapter, we focus on the

collaborative aspect of the dive computer design.

In Chapter 7, we look at the implementation of the paper prototype, software mockup, and the network simulation which we used in our user studies and simulation. The execution process and results are then presented in Chapter 8. In Chapter 9, we presents an analysis of our results and finish with a conclusion in Chapter 10.

Chapter 2

The Universe of Fault Tolerance, HCI, and CSCW

When designing critical life dependent systems, it is important to maximize fault tolerance and usability at each level of the system: hardware, software, and network (Figure 2.1). While fault tolerance can ensure the robustness of the system, looking at usability that the system needs can help ensure effectiveness, efficiency, and satisfaction to users.

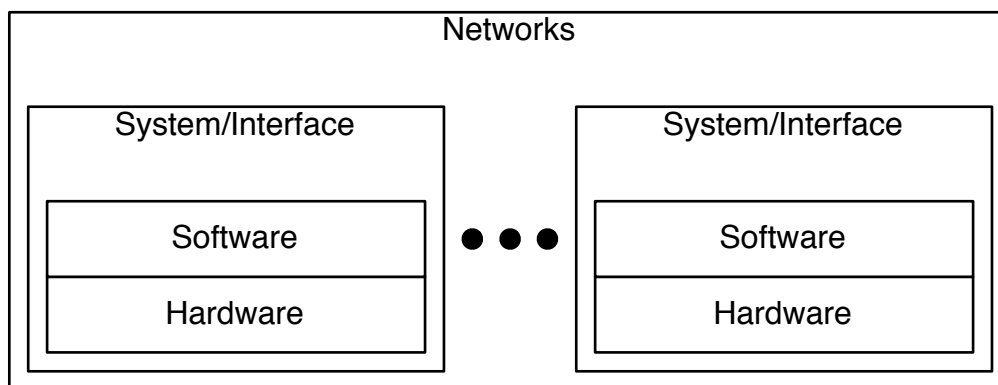


Figure 2.1: Levels of abstraction.

2.1 Fault Tolerance

A fault tolerant system is composed of two parts: fault prevention and fault tolerance. Through careful design methodology, design rules, design reviews, and quality control, we can implement fault prevention techniques. Similarly, fault tolerance can be attained through redundancy, fault detection, fault containment, and reconfiguration.

In the scheme of fault tolerance, the following measurements such as, testability, maintainability, safety, availability, and reliability are used [2]. The testability of a system is high when it can determine if an individual component is faulty or fault free. Maintainability refers to the time required to reinstate the system and the ability to maintain operation while delaying the problematic issue until the next scheduled check up [2]. Safety of a system is evaluated based on two factors: correctness of performance and ability to discontinue service without comprising the safety of other systems [2]. To have a system with high safety, the system should fail safely (e.g. discontinue service without compromises) and be robust. The terms availability and reliability refer to the correct performance of a system, where the former looks at a particular instant and the latter looks at an interval [2]. If we let an instant in time be t , then availability would only look at the system's correctness at t and the reliability during $[t_0, t]$.

2.1.1 Redundancy

One of the most well-known fault tolerant approaches is redundancy. There are four types of redundancy: hardware, software, information, and time. In general, redundancy is accomplished through the addition of information and resources beyond what is needed for the normal system operation.

Hardware redundancy exists in a passive form and an active form. In the passive form, the system attempts to hide, or mask, the faults that have occurred. This is usually achieved by using techniques such as Triple Modular Redundancy (TMR) (Figure 2.2) and N-Modular Redundancy (NMR). Triplicated TMR and Multiple Stages Triplicated TMR exist as variations of the TMR model. In the dynamic form, the system uses sparing. There are three sparing approaches, hot standby sparing, cold standby sparing, and pair and a spare. In hot standby sparing, all spare modules are powered and running so that they can take over failed modules at any point in time. On the contrary, all spare modules in the cold standby mode are powered off until they are needed. With these designs on the two extreme ends of the spectrum, surely the hybrid version of the two approaches would emerge – the pair and a spare. In this approach, there are at least two modules powered on and operating. The results of the two operating modules would then be compared. If the two results disagree, it automatically goes to a third module for an additional answer to compare with. In the TMR model, faults are masked by the two matching outputs; in all sparing models, an error detector is on one module to determine whether it is faulty or not. If a fault is detected, then an alternate module is used (Figure 2.3).

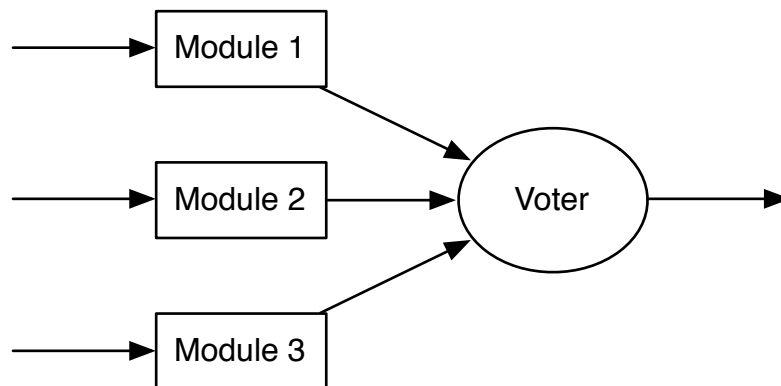


Figure 2.2: An example of TMR. All three modules perform the same task. When there is a discrepancy between the outputs, the output with most matches is used. [2]

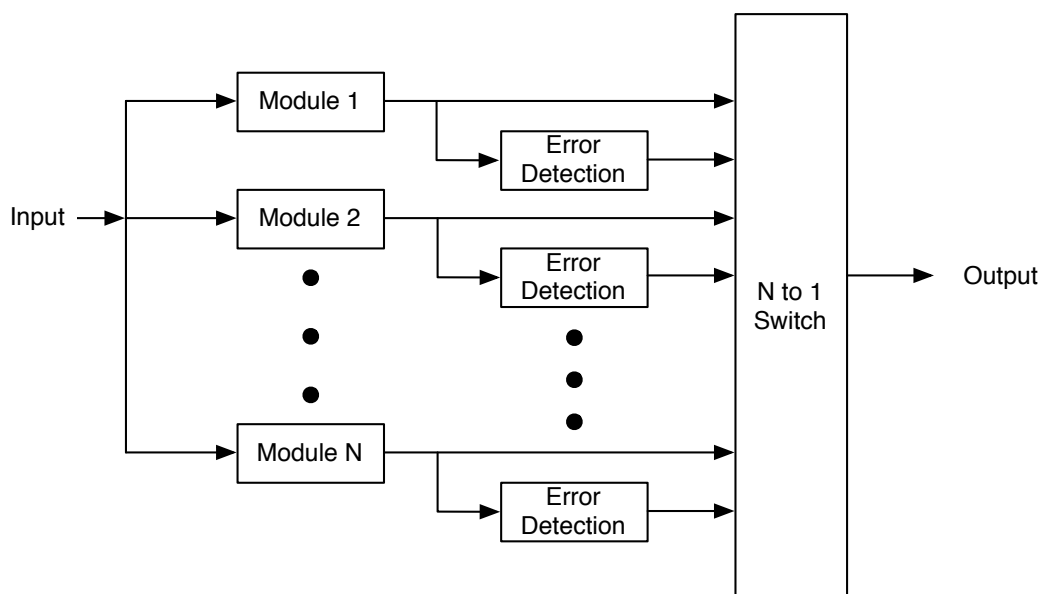


Figure 2.3: An example of N-modular standby sparing. While one module is provided the actual output, all other $n-1$ modules act as spares. [2]

Software redundancy is a less common term and type of redundancy used when thinking about fault tolerance. However, software redundancy appears in many forms. Two examples of software redundancy are: (1) writing two versions of the same software, and (2) using extra lines of code to catch the same exception. Alternatively, the same program can be written in different languages and approaches. Then one can execute both programs and compare the results.

Similar to hardware redundancy, information redundancy also uses extra hardware. In information redundancy, additional information is added to data in order to allow detection, location, masking, and correction of faults within the system [11]. Techniques such as separable codes and maximum likelihood decoding can be used.

In contrast to hardware and information redundancy, time redundancy attempts to achieve redundancy at the expense of more time instead of hardware. The idea of time redundancy is to repeatedly execute the same computation and compare all results (Figure 2.4).

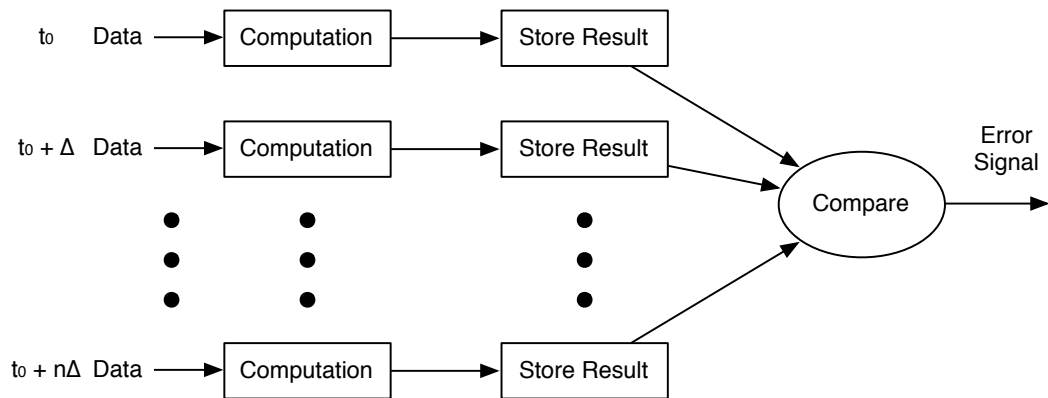


Figure 2.4: An example of time redundancy. [2]

2.1.2 Fault Detection

The recognition of a fault having occurred is called fault detection. The measure of the system's ability to detect faults is referred to as the level of fault detection. Such detection is usually used at the circuit level.

2.1.3 Fault Containment

Fault containment attempts to hold the effects of a fault to a particular area. Its coverage is measured by the system's ability to extend effects only to local surroundings.

2.1.4 Reconfiguration

Reconfiguration refers to the process of eliminating and/or masking a fault such that it reconfigures the system to restore the normal state of a system. During this process, the system detects, locates, and recovers from the fault.

2.2 Human Computer Interaction

Regardless of the robustness of the system design, without well designed Human Computer Interaction (HCI), the system is essentially useless. The importance of HCI is shown through the Three Mile Island Nuclear Power Plant Disaster example in “User Interface Design and Evaluation” [16]. Details of this event can be found on the United States Nuclear Regulatory Commission website [17].

When designing interfaces and ways of interaction between the user and the computer, we need to ensure that it is intuitive and engaging to use. Furthermore, we should also consider whether or not users can easily complete their goals. In general, we should take the choice of output devices, icons, colour, shape, size, and grouping into consideration when designing. Thus, usability is a priority for safety critical systems.

Other than the physical interaction, we must also examine the portability of the system. Variables such as size, weight, robustness, and battery life also need to be looked at.

2.2.1 Design Process

In the normal design process, we usually go through four stages: “study”, “design”, “build”, and “evaluate” (Figure 2.5). However, in 2007, Microsoft research introduced an “additional” stage called “understand” into the current process [3], for their own HCI design.

In both models, we start with learning the current processes and practices. The “understand” stage focuses on values, attempting to distinguish the values of stakeholders whom we are designing for. At the outset of this stage, we are able to specify requirements and stakeholders we wish to serve. During the “study” stage, we seek

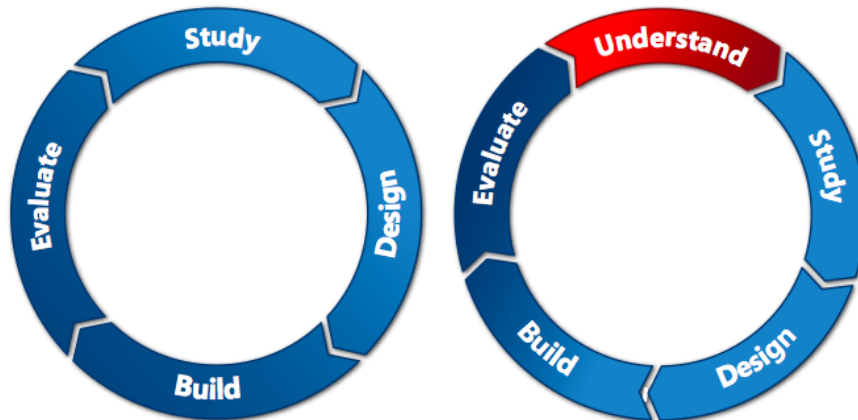


Figure 2.5: The design process circle for HCI. [3]

to develop deeper cognitive understanding of important factors that change values of interest. Instead of looking at interaction around a particular technology, it considers details of tasks and asks questions such as how the interaction can help people achieve their goals. Meanwhile, in the “design” stage, the primary goal is to incorporate creativity with design goals. In this process, we need to consider the culture and setting in which the device is deployed. Moreover, not only do we wish to concentrate on user experience, we must also consider the underlying infrastructure (like networking and sensing) that may open up a new kind of engagement and collaboration. While there are modifications to the stages mentioned above, the “build” stage remains unchanged. Building the designed tool can range from low-cost methods such as a paper prototype to high-cost methods such as partially working physical systems. In the last stage, we evaluate the design through evaluation methods.

2.2.2 Is Usability Really Common Sense?

Through HCI, we are able to understand the user’s experience when interacting with the device. However, by providing a user friendly environment, the device does not guarantee itself to be useful and fulfill the initial purpose. While most believe that

usability is only an application of common sense, is it really? In this section we are going to look at an example of good and bad HCI from the perspective of usability.

Usability Example

In this example, we are looking at the design of two presentation remote controls. While the PCPAL (Figure 2.6) presenter exemplifies a comparatively poorer design, the Kensington model (Figure 2.7) demonstrates what a good design can include.



Figure 2.6: PCPAL 3-in-1 remote control, mouse, and presenter [4].

The PCPAL is rectangular shaped with small buttons evenly placed in a grid, while Kensington only places buttons on one side of the remote. Because of this slight difference, users are able to tell whether or not they have picked up the Kensington remote the right-side-up based solely on feeling, while this is not the case with PCPAL. This can also cause a problem for the PCPAL user when they attempt to use the arrow keys while holding the remote upside-down.

The size of buttons for the PCPAL remote control can also cause problems because of the various sizes of fingers an individual can have. Due to the proximity of the buttons, individuals with larger fingers can easily press the wrong button.



Figure 2.7: Kensington wireless presenter 33374 [5].

These two examples demonstrated the tradeoff between usability and functionality. While the main function of both devices is to serve as a presentation tool, the PCPAL model also tries to serve as a mouse and a remote control at the same time. Because the PCPAL designer wished to incorporate multiple functionalities, the level of usability goes down. On one hand, multifunction devices can gain convenience; conversely, the resulting device may end up with low usability for all its intended functions.

2.3 Computer Supported Cooperative Work

CSCW is often referred to as Computer Supported Cooperative Work. It is used as the “general and neutral designation of multiple persons working together to produce a product or service” [18]. However, in most cases, such work is completed in collaboration. Thus, the alternate term Computer Supported Collaborative Work emerged and the two are used interchangeably.

2.3.1 Time Space Taxonomy

According to the CSCW time space taxonomy, there are four categories: face-to-face interaction, asynchronous interaction, synchronous distributed interaction, and asynchronous distributed interaction (Figure 2.8) [6]. In face-to-face interaction, individuals physically must stay in the same room. An example of asynchronous interaction is a message written on a whiteboard, since individuals can be at the same location but communicating at different times. Synchronous distributed interaction can be in the form of a group chat where multiple individuals at different geographic locations can view the same message (e.g. IRC channels). Last but not least, asynchronous distributed interaction, with email being a prime example of such interaction since individuals can communicate across time and space.

	Same Time	Different Time
Same Place	face-to-face Interaction	asynchronous Interaction
Different Places	synchronous distributed Interaction	asynchronous distributed Interaction

Figure 2.8: Groupware time space matrix [6].

2.3.2 Awareness

Awareness has been shown to be one of the main focuses at CSCW conferences in recent years. The key goal is to design and look at “interfaces that help people stay aware of information without being overwhelmed or distracted” [19]. Awareness can be further divided into eight categories:

Presence Awareness refers to when an individual knows who is currently “here”.

An example would be a person’s online chat list where various statuses are shown [20].

Identity Awareness includes enough information for an individual to identify another person on the system. In the case of the online chat list, screen names provide identity awareness by providing a link between the virtual account and the individual in real life.

Location Awareness provides information with regards to the current geographic location of the user and/or others [20].

Information Awareness alerts individuals if there is an update in the information that is marked important. An example is an email notifier that beeps when new mail arrives.

Social Awareness contains information as to whom is connected to whom. Facebook applications like TouchGraph can generate such information (Figure 2.9) for everyone that the individual is connected to.

Activity Awareness refers to individuals, in a group, who know what the others are doing. Functionality such as the automatic display of the song that you are currently listening to on the chat list is an example for activity awareness.

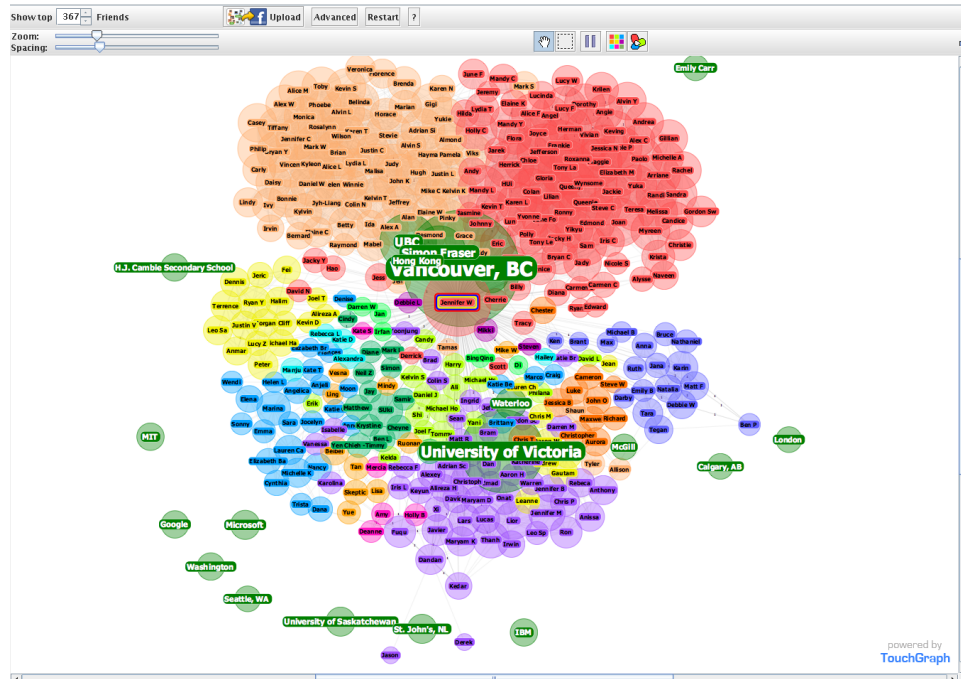
Workspace Awareness refers to individuals who work in the same visual workspace. These individuals are also aware of others who are working concurrently in the same workspace [21].

Situational Awareness refers to the individual’s perception of the environment. It may include the individuals’ understanding of how the information and actions can impact the end goals and objectives [22].

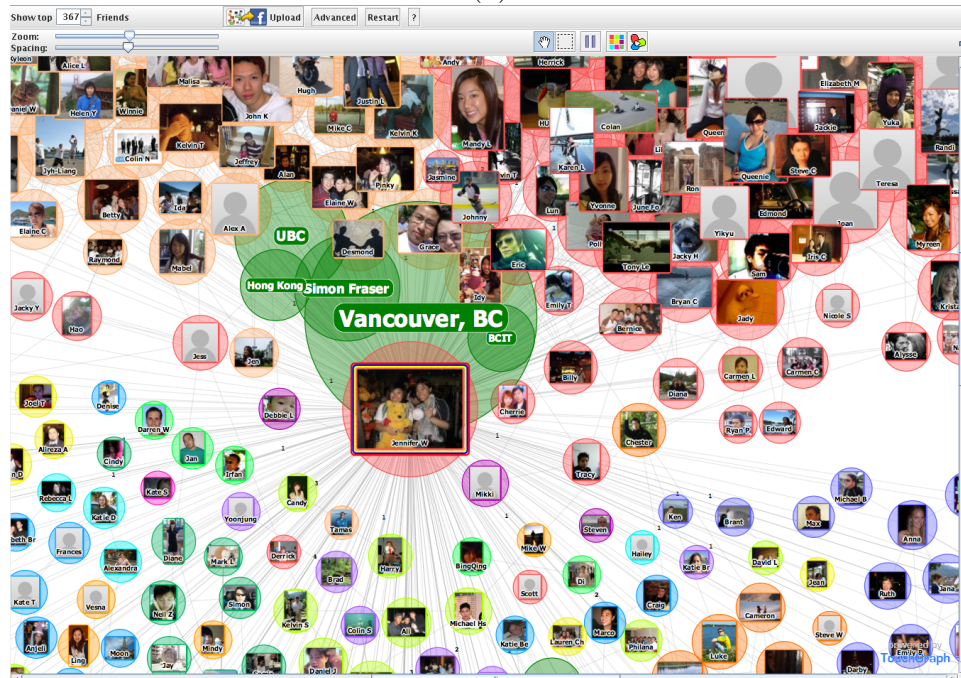
As a side-effect, awareness also increases the chance of disruption to the tasks at hand [23]. Thus, a careful balance between the two is needed when designing collaborative software.

2.4 Conclusion

Because safety and ease of use are the number one priorities when designing critical life dependent system, we must look into all aspects of the system including: fault tolerance, human computer interaction, and computer supported cooperative work. In this chapter, we have looked at the general overview of what needs to be considered; in the following chapters we examine specific items in each of these areas that apply to dive computers.



(a)



(b)

Figure 2.9: Screenshots of TouchGraph.

Chapter 3

Communication Technology



Figure 3.1: An example of ubiquitous system that can be taken underwater when secured in the underwater housing [7].

“One of the major technological achievements of modern history has been the design and implementation of means of communication and transmission of information... The social and cultural implications of this development are huge and far-reaching” [24]. Since providing an alternative means of communication between scuba divers is one of our main goals, we must determine the most suitable tool and platform to fulfill this goal. Thus, we need to carefully study the various available

technologies, along with their pros and cons, in order to help us make a sound decision.

3.1 Transmission Methods

There are numerous ways to implement underwater communication. These include: acoustic propagation, radio waves, radar, and sonar. As we can see in the following sections, challenges with bandwidth and latency always exist regardless of the technology used; this is due to the conductivity level of water. In general, the attenuation of waves depends on temperature, salinity, and frequency [25].

3.1.1 Acoustic

A standard underwater acoustic network is constructed by forming bi-directional links between all devices. The use of underwater acoustics comes with problems like limited bandwidth, large signal propagation time, and a low transmission power level [26]. Devices may be incapable of transmitting and receiving at the same time [9].

Acoustic noise is defined as unwanted sound or meaningless data that is transferred between acoustic systems. Noise can degrade the quality of the received signal and may make it unintelligible. Because noise can be created from various waveforms, Luton [24] described and categorized noise into four categories: ambient noise, self-noise, reverberation, and acoustic interference.

Ambient noise includes any random waveforms created outside of the acoustic system. It can be triggered by nature or an artificial source. Self-noise is a type of noise where the system picked up noises created by itself and/or any supporting platform within the system. An example of self-noise is electrical interference. The third category, reverberation, affects sonar systems where its own echoes are louder than the expected target echoes. Lastly, there is acoustic interference which is generated

by other acoustic systems working nearby.

3.1.2 Radio

The absorption of electromagnetic waves of given frequency over a given distance can be calculated by using the salt level, temperature, depth, and radio frequency [25, 27]. In some situations, depending on the application and choice of radio wavelength, radio transmission is accomplished in the following ways: water-to-air, air-to-air, and air-to-water [27].

3.1.3 Radar and Sonar

Radar is a short form coined from “radio detection and ranging”; it uses electromagnetic waves to identify variables such as range and direction. Underwater radar systems emit radio waves in the microwaves frequency range. However, the absorption rate of the microwaves by the water is too great for this type of frequency to be in any practical use [28].

Sound navigation ranging, or more commonly known as sonar, can be seen in two forms: active and passive [29]. In its active form, it emits pulses of sound waves and waits for the signal to be reflected off the closest object. Highly similar to the technique use by whales, dolphins, and bats to locate prey; we can use the knowledge of speed of sound and the time sound wave used to travel to the target and back, we can calculate the distance between the emitter and receiver. In the passive form, instead of emitting, sonar waits and listens for sounds generated by others.

3.2 Related Work

As underwater research has recently become one of the “hot” topics, we must look at such literature along with commercially used products in order to determine the technology which is best to use.

3.2.1 High-Speed Optical Transmission

The Japan Marine Science and Technology Center developed a remote operated underwater vehicle for deep-diving called Kaiko [30]. There are six cameras mounted on Kaiko with five being part of the main operation while the sixth acts as a spare; there are also other sensors and equipment attached to the vehicle along with a 250-meter secondary cable.

Kaiko is able to transmit real-time data at the speed of 840 megabytes per second. A launcher unit is located between the vehicle and the supporting ship unit. Its main function is to relay and translate data between the vehicle and the ship unit because of the different optical fibers used in the primary and secondary cables.

3.2.2 EvoLogics - Hydroacoustic modems with S2C Intelligent Underwater Telemetry

The S2C technology is known as the Sweep-Spread-Carrier, which attempts to provide optimum underwater data transmission [8]. It also incorporates fault tolerant techniques such as built-in error correction codes. At the final stage of development, as stated in an article (September 2008), the technology had moved from S2C-180 (Figure 3.2) to S2C-280 (Figure 3.3). The S2C-180 has a depth rating of 100 meters, telemetry distance up to 2000m, and bitrate up to 33 kb per second; while the S2C-280 has a depth rating of 6000 meters, telemetry distance to 4000 meters, and bitrate

up to 20 kb per second.



Figure 3.2: The S2C-180 hydroacoustic modem [8].



Figure 3.3: The S2C-280 hydroacoustic modem [8].

3.2.3 Acoustic Modem

The LinkQuest company has developed numerous types of acoustic modem. The acoustic modem that we are going to focus on in this section is the UWM-100 (see Figure 3.4) [1], which is intended to be used for shallow water data transmission. This acoustic modem has an acoustic link that can transfer up to 17.8 kilobits per second and operates at 26.77 to 44.62 kHz. In Table 3.1, we have listed some UWM-100 specifications.



Figure 3.4: The S2C-280 hydroacoustic modem [1].

Table 3.1: LinkQuest UWM-100 specification [1].

Acoustic link	17.8 kilobits per second
Bit error rate	less than 10^{-9}
Operating Frequency	26.77 to 44.62 kHz
Operating temperature	-5 to 45°C
Weight in water	2.3 kg

3.2.4 Underwater Robotics Communications

Autonomous underwater vehicles (AUVs) are robots that could potentially bring humankind a step closer to a solution when faced with marine difficulties such as a search and rescue in deep sea. Currently, researchers are looking to minimize the size of an AUV and its related operational costs, as well as, research in the area of improving underwater communication between two AUVs.

Underwater communication poses problems as temperature differences in the water causes many difficulties with transfer rate and wave diffraction. Underwater com-

munication can take three forms: acoustic propagation, fiber-optic communication and radio modems. Acoustic propagation is not a very effective form of communication as described in section 3.1.1. Due to the many inconveniences of acoustic propagation, it is generally ruled out for use in an AUV. Fiber-optic communication is also not a realistic choice for an AUV due to its cost, upkeep and fragile fiber-optic cables. The impracticality of the first two forms of communication has led researchers to use radio modems, particularly Zigbee modules, in AUVs. Zigbee modules provide many benefits as they do not require much power, reduce data size which allows for a simpler, and therefore less expensive, network, and its networks can cover large areas using routers. Particularly, one of the Zigbees networks is called a mesh network; a mesh network is reliable as it has a self healing capacity which reroutes a message through the network when a node fails.

To further analyze the suitability of Zigbee modules in underwater communication, two experiments were run by Nagothu *et al.* [9]. The first experiment consisted of placing the base and remote of a Zigbee module next to each other near the water and extending the antennas at different depths to record the amount of information sent from remote to base and base to remote. The hit rate, or the number of times information was received from remote to base divided by the total amount of times information was sent, was 100%; in other words, information was received every time. The second experiment had the modules shielded with aluminum foil to test whether the hit rate remains the same. However, after being covered with aluminum foil, the signal strength from the base to the remote decreased as depth and distance increased (Figure 3.5) and thus the hit rate is expected to fall.

In “Communications for underwater Robotics Research Platform”, Nagothu *et al.* presented two different types of underwater communication using Zigbee modules: the brute force approach and their proposed approach. In the brute force approach,

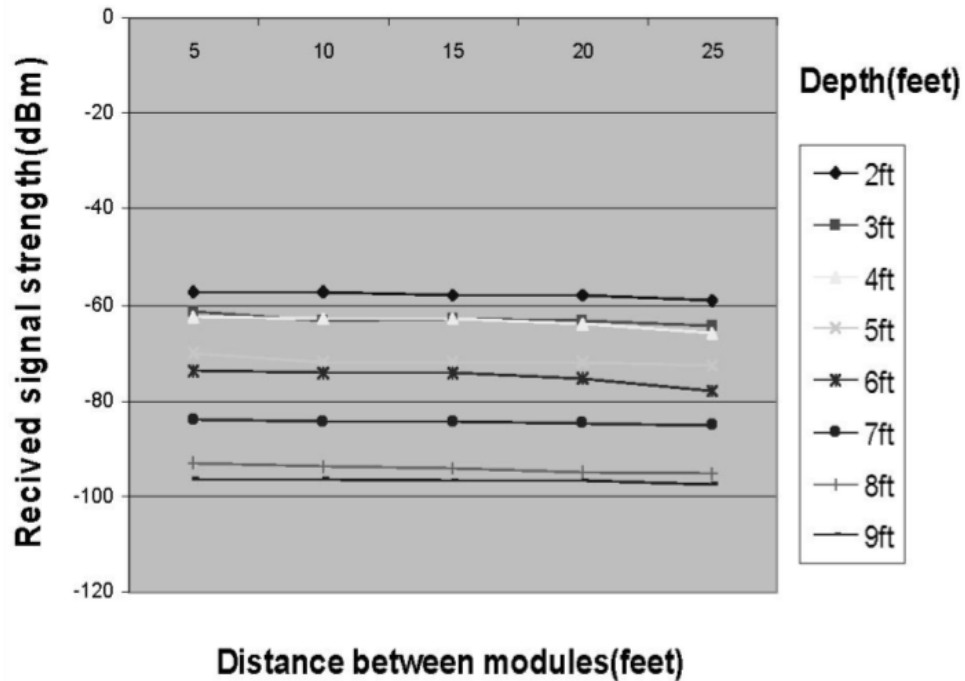


Figure 3.5: Results of Nagothu's experiment 2 [9].

one node is named the master node and it controls the amount of information that is circulating between all the other nodes in other AUVs. When transmitting a packet of information to another AUV, the packet of information has an identification number that contains the name of the destination node. When a node receives information, it verifies the identification number. If the identification number matches, it stores the received information in its memory; if the identification number shows that the packet of information is not meant for the node, the node passes it onto the next node. If two packets of information are received at once, the node ignores the second packet. The downfall of the brute force approach is related to the way that nodes receive information. Not only is the time needed and power consumption increasing with every node that the information has to pass through before reaching its destination; this process would also take a lot of memory, battery power and bandwidth which makes it impractical. In the proposed approach, the positions of all the nodes are

known and when transmitting information, the position of the node is included with it. The process of transporting the information is the same as the process in the brute force method but in the proposed approach, the master node can be switched to another node if there is a system failure.

Some other experiments to test the underwater acoustic communication system of an AUV include one completed by Marine Systems Engineering Laboratory [31]. Two EAVE III AUVs equipped with sensors for things like acoustic altitude and depth, pressure depth, water temperature, were used in the experiment. To find the position of the other AUV, a navigation algorithm was created which needs the following three pieces of information: location coordinates (x,y,z) of the transponder on the AUV, the depth and heading of the AUV that is trying to find the other AUV and the turnaround time delay for the transponders on each AUV. By sending out three transmit pulses from one AUV to another and calculating the time it takes for the other transponder to return the three pulses, one AUV is able to find the position of the other.

3.2.5 Float Buoy Ranging System

Kurano *et al.* designed an experiment with a ranging system in attempt to determine the underwater running locus' position [10]. Figure 3.6 shows the entire system setup when deployed.

The ranging system is made up of a AUV-mounted pinger, three surveying float buoys that receive the pinger signal, and a receiving station on the test ship. Kurano *et al.* carried out their test twice. The initial test was only between the float buoys and the ship mount pinger, in attempt to track the floating buoys. The second test, they have tracked the running locus through both the floating buoys and the information from the AUV itself. In their evaluation, it was shown that there could be an error of

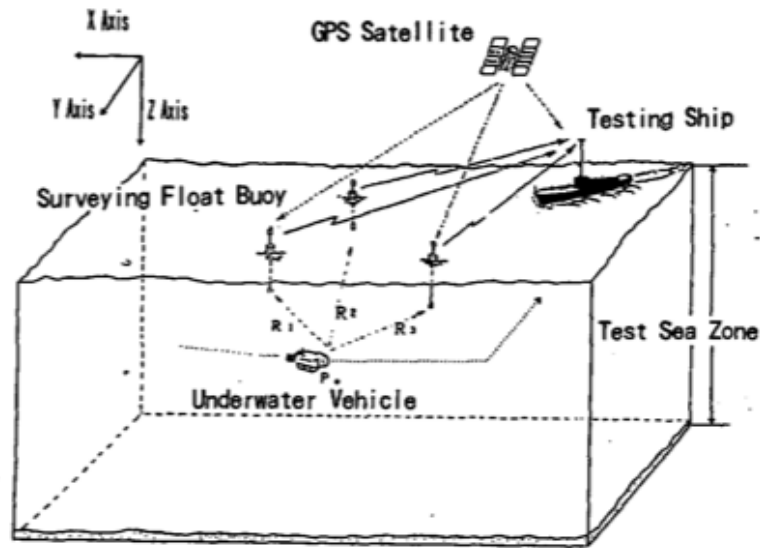


Figure 3.6: The float buoy ranging system proposed by Kurano *et al.* [10].

± 15 metres in measured values.

3.3 Communication for Dive Computers

In most literature that we have seen, the two major data transmission tools used for underwater robotics are acoustic modems and the Zigbee module. However, it appears that more research is based on acoustic modem; in other words the chance of having an improved underwater communication for acoustic modem in the future is higher than using the Zigbee module. For this reason, we have chosen to use the acoustic modem in our design.

3.4 Conclusion

In this chapter, we started by presenting possible forms of underwater data transmission. Then we looked at the current state of research and technology and in the end chose a technology for our design.

After selecting the appropriate tools that can meet our requirements, in the next chapter, we look at how the underwater network can be built using the tool chosen in Section 3.3.

Chapter 4

Network

To make communication between dive computers possible, we need to interconnect the set of dive computers for “gathering, processing and distributing information” [32]. In other words, a wireless sensor network is in order. In this chapter, we look at ways to attempt and build both a mesh network and an extended mesh network. This includes an analysis of current routing approaches.

We first start by looking at types of routing protocols, namely the proactive routing, reactive routing, and hybrid routing approach. Next, we look at existing research in underwater network. This is followed by a proof on how triangulation can occur using spheres and finishes with a discussion on our network design for dive computers.

4.1 Proactive Routing

Proactive routing is mostly used in applications like the Global Positioning System. In the proactive approach, all nodes are constantly exchanging routing messages and maintaining sufficient and fresh network topological information. By periodically sending routing tables around the network, all nodes maintain up-to-date information such as lists of nodes and routes.

The disadvantage of using this approach is that it requires a consistent transfer of data which grows proportionally with the number of nodes within the network. The total number of links can be derived from the total number of nodes by using the formula $\frac{n(n-1)}{2}$. In addition, if there is a failure in the network, it will take a relatively longer time to respond.

4.2 Reactive Routing

Reactive routing is triggered by communication demand at sources. The node that needs to send a message will first flood the network requesting routes to try and find a path between itself and its desired destination.

The disadvantage of this method is the high delay in packet delivery because of the time required for path finding.

4.3 Hybrid Routing

Hybrid routing is a combination of proactive and reactive routing. At the start of the network, partial routes would be determined using the proactive approach. For any nodes that are not part of the predetermined routes, the reactive approach will be used when a message needs to be sent.

4.4 Related Work

In this chapter we are focusing on mobile wireless communication networks. Thus, we are going to present related work that focuses on mobile sensor networks and ubiquitous system computing.

4.4.1 Mobile Ad Hoc Wireless Networks

Similar to crisis systems designed in “Crisis Management using Mobile ad-hoc Wireless Networks” [33], most land-base (above water) crisis systems rely on the Global Positioning System for communication. These systems are used for reporting emergencies such as accidents, natural disasters, and acts of terrorism. Details of the incident (e.g. the location and current situation) can be reported directly into the system by both authorities or the individual in distress. The biggest challenges in these environments are the breakdown of communication and trying to find a way “to store and retrieve information in ad hoc, distributed, and loosely connected networks” [33]. As the authors have suggested, although distribution and duplication of information on every node is impossible, information on the disconnected node should still be made available. For these reasons, the authors suggested to give partial, redundant information to neighbouring nodes.

In addition to the proposed information fault tolerance, we also need to look at efficiency in mobile ad hoc wireless networks. Su *et al.* developed a tool such that it uses the late-binding technique for data transmission. In other words, the device “adapt[s] to its mobile environment by delaying network connectivity interface and protocol selection until the moment of data transmission” [34]. By using this approach, it allows concurrent transmission across multiple protocols and applications.

4.4.2 Sensor Networks for Health and Safety Monitoring

There is a great potential for ubiquitous computing and embedded wireless systems to improve health and safety processes. When researching in such fields, a great deal of field study through interviews and observations is required. Kortuem *et al.* identified “three beneficial uses of ubiquitous technologies: 1) improving the quality of recorded health and safety data; 2) providing timely, personal attention to work-

ers and operatives about health and safety risks; 3) improving the understanding of company-wide health and safety risks.” [35] While designing such ubiquitous embedded systems, there are usually two types of architecture that one can follow. One is an architecture based on sensor network concepts and the other is an architecture that is built around the idea of smart everyday objects. Smart everyday objects are objects that link everyday items with technology. “A smart object can perceive its environment through sensors and communicates wirelessly with other objects in its vicinity. Given these capabilities, smart objects can collaboratively determine the situational context of nearby users and adapt application behavior accordingly.” [36]

The wireless sensor networks approach is usually constructed by a group of self-forming and self-healing wireless networks of low-power embedded sensor nodes. In this approach, a number of sensor network nodes can be scattered around a work site or attached to work related objects and people streaming sensor data. It is ideal to attach these sensor nodes on people because of the direct and accurate measurements; however, this approach can be obtrusive to subjects.

The second approach utilizes the key component of ubiquitous computing – the use of smart everyday objects. These artefacts are part of our everyday lives and are integrated with technology that can provide us with sensing computation, and communication capabilities.

The major differences between these two approaches come from the relationship between the ubiquitous embedded device and the people rather than the technology itself.

4.4.3 Intentional Naming System

Mobile nodes and services create dynamic environments and cause rapid fluctuations in performance. Because the routing between such nodes is dynamic, Adjie-Winoto *et*

al. [37] identified four design goals for a naming system that enables dynamic resource discovery. These goals are: expressiveness, responsiveness, robustness, and easy configuration. In the Intentional Naming System (INS) where the authors proposed to utilize name specifiers, clients use name specifiers as part of the header of a message to identify the message's final destination. Systems will also periodically broadcast their intentional names along with the description of the services they provide.

The main activity in the proposed system is mapping name specifiers to the corresponding network locations. At the event when a message arrives at the system, it would find the name specifier in a name tree which then returns a record that contains the actual IP address of the destination. In this approach, there are two possible bottlenecks: (1) name tree lookups and (2) name update processes. These bottlenecks are solved by delegating work to inactive resolver nodes.

4.4.4 Seamless Networking

In today's world, a wide range of radio technologies based communication have been developed, from short-range platforms like Bluetooth to long-range platforms like cellular radios. However, there are situations where two people side-by-side cannot share resources and data due to low network connectivity. Su *et al.* [34] presented Huggle, an architecture that separates application logic and transport bindings. A key to the suggested approach is that applications should not be concerned with or aware of the data transportation mechanism. It uses the data-centric architecture to internally manage data handling and data propagation tasks.

Another example of seamless networking was shown in the idea of collaborative download for multi-homed wireless devices [38]. Because Wireless Local Area Network (WLAN) offers a much higher speed than Wireless Wide Area Network (WWAN), it is obvious that WLAN will be the transportation interface of choice when available.

The authors attempt to design a protocol that can support seamless collaborative download. In their protocol design section, they drew attention to three key components for designing such protocol: (1) a protocol for mobile devices to form groups, (2) a scheme to distribute work amongst the group, and (3) a mechanism for low-level data transport and connection management to fetch data from servers.

As part of the group formation protocol, an initiator (any mobile device) must identify the set of collaborators (other mobile devices that wish to collaboratively download) that can work correctly while individual devices move in and out of range. Each device will also periodically broadcast a message, notifying others that they are still “alive”. Similar to MapReduce [39], work is partitioned into chunks. These chunks are then put into a work queue where free devices can dequeue chunks to work on. This approach allows dynamic adjustment to work distribution and consequently offers a potential performance gain.

4.5 Transmitters

In “The study of the float buoy ranging system for the underwater vehicle”, Kurano *et al.* [10] presented an analysis, using matrices, to find the coordinates of one underwater ranging locus (see Section 3.2.5 for details). However, it is unclear as to why three buoys were used in their design. Thus, in this section, using basic geometry, systems of equation, and the quadratic formula, we determine the minimal number of buoys needed in order to determine the position of dive computers based on triangulation theory.

Because signals are broadcast in every dimension of the three dimensional space, we start by defining the equation for spheres.

$$(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 = r_1^2 \quad (4.1)$$

$$(x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 = r_2^2 \quad (4.2)$$

$$(x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2 = r_3^2 \quad (4.3)$$

By using one transmitter, we can only obtain the radius between the transmit and the dive computer (Figure 4.1). Thus, we start by attempting to use only two transmitters to calculate the dive computer position through calculating the intersection (Figure 4.2).

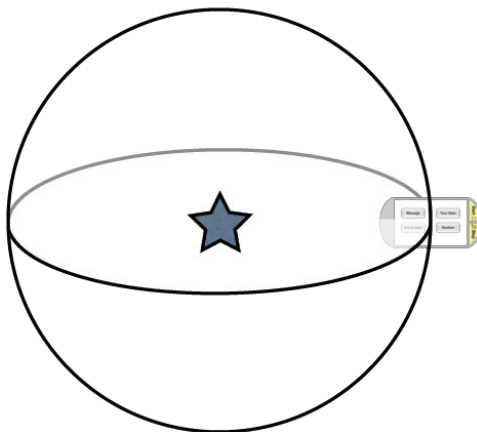


Figure 4.1: Triangulation with one transmitter.

First, we assume the centre for sphere 1 is $(0,0,0)$. Next, we can assume the centre for sphere 2 is $(x_2,0,0)$ because the link between the two transmitters is a straight line, thus allowing us to orientate the two spheres in such a way that they lie on the same two dimensional plane.

Based on our theory above, equations 4.1 and 4.2 can be rewritten in the following forms:

$$x^2 + y^2 + z^2 = r_1^2 \quad (4.4)$$

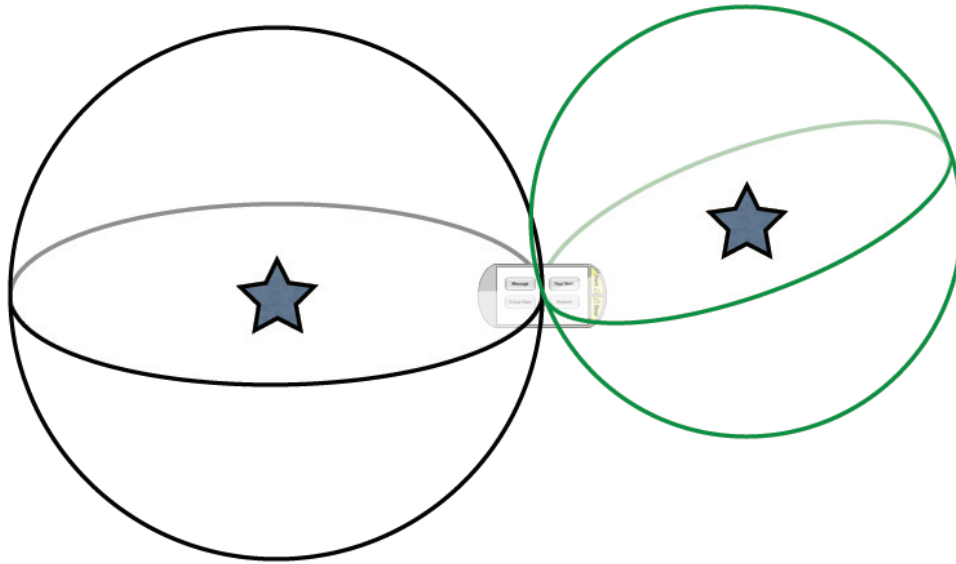


Figure 4.2: Triangulation with two transmitters.

$$\begin{aligned}
 (x - x_2)^2 + y^2 + z^2 &= r_2^2 \\
 x^2 - 2x_2x + x_2^2 + y^2 + z^2 &= r_2^2
 \end{aligned} \tag{4.5}$$

Next, we can find the intersecting x-coordinate by combining equations 4.4 and 4.5 using systems of equations.

$$\begin{aligned}
 (x^2 + y^2 + z^2) - (x^2 - 2xx_2 + x_2^2 + y^2 + z^2) &= r_1^2 - r_2^2 \\
 2xx_2 - x_2^2 &= r_1^2 - r_2^2 \\
 2xx_2 &= r_1^2 - r_2^2 + x_2^2 \\
 x &= \frac{r_1^2 - r_2^2 + x_2^2}{2x_2}
 \end{aligned} \tag{4.6}$$

To find the intersecting y, z-coordinates, we can substitute x in equation 4.4 with equation 4.6 to form 4.7.

$$\left(\frac{r_1^2 - r_2^2 + x_2^2}{2x_2}\right)^2 + y^2 + z^2 = r_1^2 \quad (4.7)$$

$$y^2 + z^2 = r_1^2 - \left(\frac{r_1^2 - r_2^2 + x_2^2}{2x_2}\right)^2 \quad (4.8)$$

From equation 4.8, it proves that only an equation of the y and z-coordinates can be found but not the exact coordinates. Since we wish to find the exact location, we must use more than two transmitters. Thus, we use three transmitters (Figure 4.3).

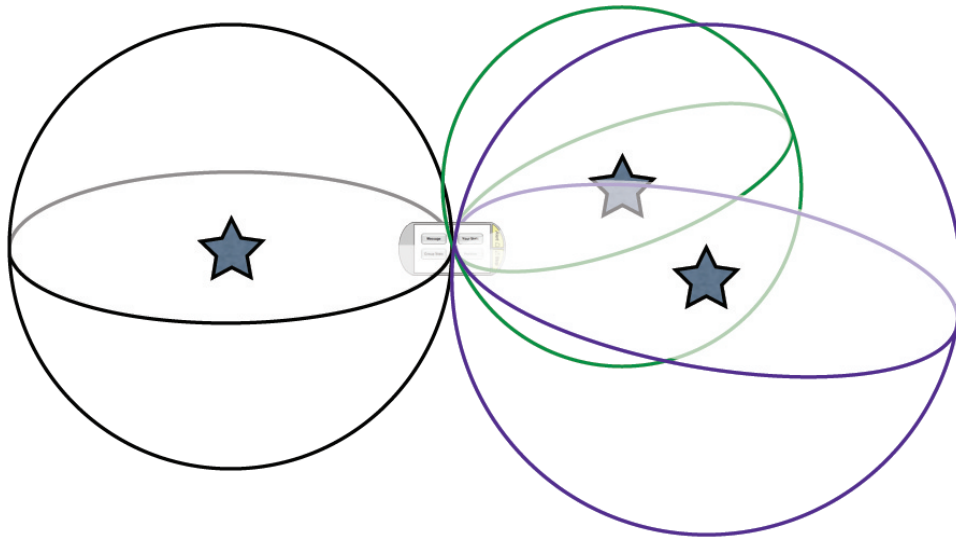


Figure 4.3: Triangulation with three transmitter.

Again, we can assume the coordinates for sphere 1, 2, and 3 to be $(0,0,0)$, $(x_2,0,0)$, and (x_3, y_3, z_3) , respectively. None of the coordinates for sphere 3 can be set to zero because we cannot force any of its axis to be the same as sphere 1 or 2. Consequently, all coordinates must be marked as variables. In order to find the location of the dive computer, we must find the coordinates of where the three spheres intersect.

While equation 4.4 and 4.5 can be reused, we can rewrite equation 4.3 as follow.

$$\begin{aligned}
x^2 - 2xx_3 + x_3^2 + y^2 - 2yy_3 + y_3^2 + z^2 - 2zz_3 + z_3^2 &= r_3^2 \\
r_1^2 - 2xx_3 + x_3^2 - 2yy_3 + y_3^2 - 2zz_3 + z_3^2 &= r_3^2 \\
2xx_3 - x_3^2 + 2yy_3 - y_3^2 + 2zz_3 - z_3^2 &= r_1^2 - r_3^2 \\
r_1^2 - r_3^2 + x_3^2 + y_3^2 + z_3^2 - 2xx_3 &= 2yy_3 + 2zz_3 \\
\frac{1}{2} \left[r_1^2 - r_3^2 + x_3^2 + y_3^2 + z_3^2 - 2x_3 \left(\frac{r_1^2 - r_2^2 + x_2^2}{2x_2} \right) \right] &= yy_3 + zz_3 \\
\frac{x_2(r_1^2 - r_3^2 + x_3^2 + y_3^2 + z_3^2) - x_3(r_1^2 - r_2^2 + x_2^2)}{2x_2} &= yy_3 + zz_3 \quad (4.9)
\end{aligned}$$

As we can see from equation 4.9, the left side of the equation is made up of known variables. Thus, to avoid confusion, we can substitute the left side with a single variable n and continue to solve for y .

$$\begin{aligned}
n &= yy_3 + zz_3 \\
y &= \frac{n - zz_3}{y_3} \quad (4.10)
\end{aligned}$$

Now that we have solved x and y of the moving dive computer in equation 4.6 and 4.10, we can substitute those back into equation 4.1 and solve for z . Again, to make the equation more readable, we substituted the right side of equation 4.6 with a single variable m . In this case $m = \frac{r_1^2 - r_2^2 + x_2^2}{2x_2}$.

$$\begin{aligned}
m^2 + \left(\frac{n - z_3 z}{y_3}\right)^2 + z^2 &= r_1^2 \\
\frac{n^2 - 2nz_3 z + z_3^2 z^2}{y_3^2} + z^2 &= r_1^2 - m^2 \\
n^2 - 2nz_3 z + z_3^2 z^2 + z^2 &= y_3^2(r_1^2 - m^2) \\
-2nz_3 z + z_3^2 z^2 + z^2 &= y_3^2(r_1^2 - m^2) - n^2 \\
(z_3^2 + 1)z^2 - (2nz_3)z - [y_3^2(r_1^2 - m^2) - n^2] &= 0
\end{aligned} \tag{4.11}$$

From equation 4.11, we can see that z can be solved simply by applying the quadratic formula $\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$.

$$\begin{aligned}
z &= \frac{2nz_3 \pm \sqrt{(2nz_3)^2 + 4(z_3^2 + 1)[y_3^2(r_1^2 - m^2) - n^2]}}{2(z_3^2 + 1)} \\
&= \frac{2nz_3 \pm \sqrt{4n^2 z_3^2 + 4(z_3^2 + 1)(r_1^2 y_3^2 - m^2 y_3^2 - n^2)}}{2(z_3^2 + 1)} \\
&= \frac{2nz_3 \pm 2\sqrt{n^2 z_3^2 + (z_3^2 + 1)(r_1^2 y_3^2 - m^2 y_3^2 - n^2)}}{2(z_3^2 + 1)} \\
&= \frac{nz_3 \pm \sqrt{n^2 z_3^2 + (z_3^2 + 1)(r_1^2 y_3^2 - m^2 y_3^2 - n^2)}}{z_3^2 + 1}
\end{aligned} \tag{4.12}$$

Equations 4.6, 4.10, and 4.12 show that the running coordinate of a dive computer can be found when three additional transmitters are used on the surface. Although the precise coverage by these transmitters depends on the exact placement of transmitters, the required coverage area is small enough for this approach to work effectively. Divers are restricted to stay within 15 meters from their flag unless otherwise specify by local laws. Assuming the three transmitters are deployed 3.5 meters apart from each other, forming an equilateral triangle, we can comfortably cover an approximate area of 42.60 meters in diameter (21.30 meters in radius) at the depth of 40 meters.

Despite the complexity of the equations, no further analysis is undertaken because we are only demonstrating the feasibility of location finding.

4.6 The Design of the Dive Computer Communication Network

In our design, we use a hybrid routing approach that is slightly altered from the one described in Section 4.3. In our approach, the initialization is done above water using the proactive routing approach. Once the dive computer is underwater, it starts to use the reactive routing approach to avoid generating heavy traffic.

There are three types of equipment required within our network: a receiver mounted on a boat, transmitters mounted on float buoys, and dive computers. The boat mount unit is optional and is useful for mapping the exact location of the devices through the device's relative position to the buoys and the exact location of the buoys by utilizing GPS. Additionally, we can incorporate Su *et al.* late binding approach and have multiple types of transmitter to allow the devices to choose the most efficient approach to use in real time.

From Section 4.5, the minimal number of transmitters that we need to use is three. Furthermore, these transmitters need to be separated to achieve an optimal triangulation; we can achieve such separation by putting these transmitters on separate buoys. In addition, because these buoys are essential in triangulation, we should apply fault tolerant techniques. In this case, we chose to use the pair-and-a-spare approach [2] with three transmitters on each buoy (Figure 4.4). In this approach, on each buoy, there are two transmitters active while the third is on cold standby. Following the possible senerio of how pair-and-a-spare works on each buoy in Table 4.1, Figure 4.5 shows a possible scenario when our designed networked is deployed.

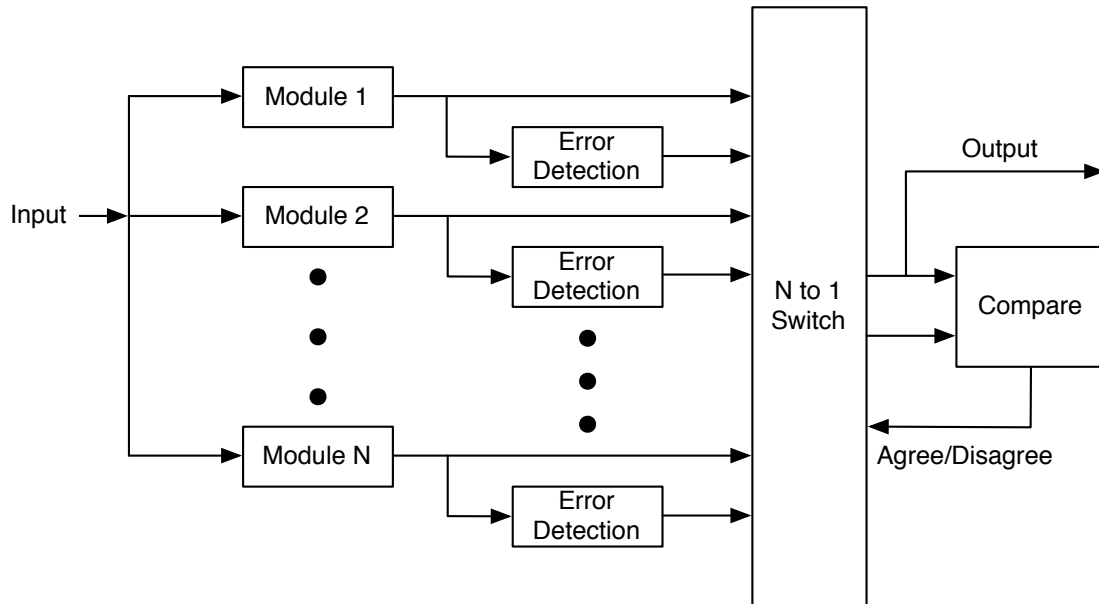


Figure 4.4: Pair-and-a-spare design [11].

Table 4.1: An example of pair-and-a spare.

Time	Transmitter A	Transmitter B	Transmitter C
t_0	On (in-use)	On	Off
t_1	On (in-use but unstable)	On	Off
t_2	Off (failed)	On (in-use)	On (powered on)

Alternatively, based on our finding in Section 4.5, if one of the buoys becomes out of reach, we can use one of the underwater dive computers as a reference point to calculate the position. To further extend this idea, the positioning functionality can continue to work without buoys, when there are four or more dive computers within the network.

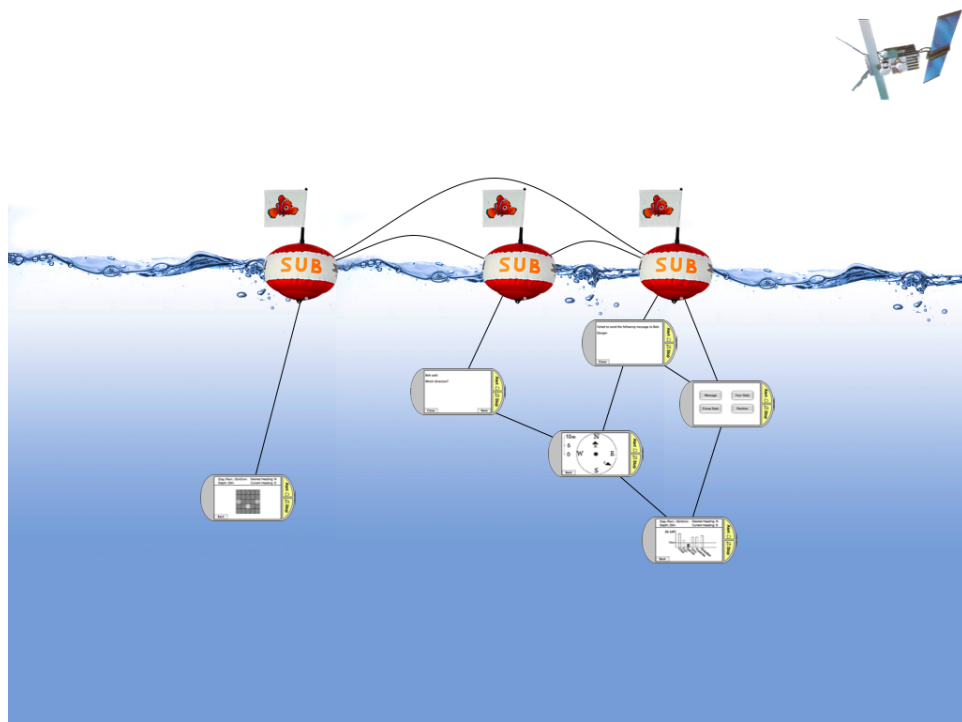


Figure 4.5: An overview of the communication network design.

Chapter 5

Human Computer Interaction

Without gills, fins, and tube feet, human explorers need to use tools like regulators, snorkels, rubber fins, buoyancy control devices, and masks to help them stay underwater “comfortably”. Although this equipment can help us stay underwater, we must keep track of the direction, temperature, and air supply because there is a limit to the time and depth that one can spend underwater.

With all the various variables, it is obvious that we need a device that displays all the information. Because robustness of a system does not guarantee usability. In this chapter, we look at HCI in attempt to design a user interface that is easy and intuitive to use, and our focus is on single user interaction.

5.1 Case Scenarios

Planning is essential in scuba diving. Before each dive, scuba divers must predetermine and plan their route, underwater time, and depth. This plan acts as a general guideline for each diver to follow while underwater, ensuring his or her own safety for the duration of the dive. Failure to follow plans can lead to comprising safety and life.

Following the design process as described in Section 2.2.1, we must first look at the current process and practices.

5.1.1 Direction

Because dive routes are pre-planned by divers, the compass is a mandatory piece of equipment ordered by diving organizations. The compass will allow the diver to follow their pre-planned route by informing them of the direction in which they are heading. While it is common for divers to end their dive at the location where they started and have multiple waypoints where a directional change is done, a function that marks exact coordinates of the diver's start point is needed.

5.1.2 Bottom Time and Depth

One of the most serious hazards that a diver has to be aware of is nitrogen narcosis. Nitrogen narcosis occurs due to too many nitrogen molecules in the bloodstream. It affects a diver's composure and judgment which leads to panic and boorish behaviour/judgement. To help divers minimize their risk of contracting nitrogen narcosis, a threshold for nitrogen saturation is included in the maximum dive time and depth calculations during the pre-dive planning. Due to nitrogen levels building and accumulating during dives, it is critical to note the duration and depth of the dive as well as how frequently dives occur.

5.1.3 Tank Pressure

Since humans do not possess the ability to breathe underwater, a breathing apparatus and a pressured air tank is needed before underwater exploration can occur. However, failing to keep track of air usage during a dive is a common mistake that leads to an unexpected end of a dive. This mistake can be fatal if a diver runs out of air before

he/she can reach the surface. Therefore, it is vital for divers to continuously monitor the pressure gauge connected to the scuba tank in order to keep track of the air they have left. For every 10 metres in depth, the rate of pressure increases by 100 kPa. Hence, more air is used at deeper depths. For this reason, this habit is essential.

5.2 Variables

From Section 5.1, it is obvious that we must at least include the follow parameters in our design: direction, dive time, depth, scuba tanks pressure. Other optional parameters on some of the existing devices include: temperature, surface interval, current date and time. The values of the mandatory parameters can be taken from existing sensors that are built into the current dive gear. Moreover, with the help of computational power and additional sensors , we can calculate and provide instant updates on additional information such as maximum depth, maximum time, decompression stop requirements, and excessive ascent rate.

5.3 Current Approach

There are two major approaches to dive planning: (1) using the dive table and (2) using dive planning software. The advantage of a dive table over dive planning software is that electronics can fail at any time, before, during, or after a dive. Moreover, in the cases where a desktop or laptop is required to run the software, it is less flexible and convenient for divers.

In both approaches, divers rely on dive computers to help keep track of data during their dives.

5.3.1 Dive Table

The dive table was originally developed in 1907; it had been the main tool used for dive planning. This table provides the theoretical nitrogen consumption by a diver during a dive. Because there is a maximum limit on ones nitrogen intake, the dive table allows a diver to see how long, and at what depth, they can stay underwater during each dive.

5.3.2 Dive Planning Software

While some dive planning software and dive computers are sold separately, there are dive computers that incorporate planning software into their devices. In either case, dive software mainly focuses on the pre-dive planning and automates the dive table look up process.

5.3.3 Dive Computer

After the planning stage, divers then execute their plan. At this stage, many would agree that one of the most important pieces of equipment is the dive computer. Current dive computers come in a wide range of price, size, and functionalities. Below, the three most common types of dive computers are introduced.

The Console Design

In the standard console design, a compass, air pressure gauge and a depth gauge are displayed; however, due to each functionality having its own display, the size of the device is increased which can lead to some inconvenience during a dive.

In this particular design, in addition to the aforementioned gauges, it also shows the time and temperatre. A common problem that occurs in console designs is the



Figure 5.1: The console design is the most common design of dive computers [12].

retrieval of cognitive data. Data can be found by searching through the three displays until locations are learned but because the location of each display varies with different models, divers generally have to spend more time familiarizing themselves with the device before a dive.

The Wristwatch Design



Figure 5.2: A dive computer in the form of a wristwatch [13].

This type of wristwatch dive computer generally displays dive time and air pressure. However, since there are many different types of wristwatches, some models

might show more sophisticated data. The advantage of a wristwatch design compared to the console design is that the attaching hose between the diver and the air tank is eliminated thus allowing more freedom in movement. A disadvantage of the wristwatch design, compared to the console design, is the small display area that the wristwatch offers. It limits the amount of viewable data that the diver can see at one time on the display.

The Mask Design



Figure 5.3: Dive computer integrated with the mask [14].

In the mask design, the dive computer has a small LCD screen at the bottom right corner of the mask (Figure 5.3). As the mask is a mandatory piece of equipment, this simplifies the design into a more convenient form than the wristwatch and console styles of devices. There is no extra equipment to take care of during a dive. Although the data displayed on the screen (Figure 5.4) contains the same information as other devices, dive time, air pressure, and depth, our device also suffers from the same problem as wristwatch style devices – the limited display area restricts the amount of data shown.



Figure 5.4: An example of what the LCD looks like from inside the mask [14].



Figure 5.5: A newly developed underwater communication device [15].

5.3.4 Communication Device

The computer shown in Figure 5.5 was presented at the DEMA 2008 show on October 22-25 2008 [40]. This device was advertised to have the capability of sending messages up to 500m in range and the capability of locating the starting point when an additional unit is purchased. As this device is still under development, limited technical information can be found about the device.

5.4 The Missing Information

Diving organizations define mandatory equipment based on information required to be collected by divers during a dive (as described in Section 5.1). However, the information collected is minimal.

One of the important pieces of data, which is not being included in any of the current dive equipment and accessories: the airflow rate from the scuba tanks valve. This piece of data can mark the difference between life and death.

When traveling, divers usually rent their equipment at their destination. Since each dive shop handles their equipment differently, there can be oversights. For example, this true story happened in the summer of 2006.

Because rubber valve caps can be easily misplaced, a dive shop that provided the scuba tanks put tapes over the opening of the valves to identify full air tanks. As one of their participants was carefully completing the pre-dive checks, they overlooked the tape and attached the regulator to the valve without removing the tape.

Before entering the water, the individual inhaled from the regulator and felt that not enough air was coming out of the regulator. So they got another participant to double-check the tank pressure (to check if the tank is full) and triple-check with other divers; because no one found any problems, the individual went into the water.

After 45 minutes of underwater exploration, the individual suddenly started to feel dizziness and lost all their senses, including the sense of direction. Luckily, a fellow diver was close by and the two went for an emergency ascend.

As the individual unequipped themselves after getting back onto the speedboat, they found the tape. It was still stuck tightly on the mouth of the valve, only with a little hole where the regulator poked through. It was only luck that another person was close by. This incident could have been prevented, if the airflow rate through the scuba tanks valve was shown on a display.

5.4.1 Derived Data

Current dive computers do not show useful information like maximum depth and time and consequently do not allow dynamic dive changes. As mentioned in Section 5.1.2, nitrogen narcosis occurs after there is a certain density of nitrogen molecules in the blood. It can be extremely useful to a diver to monitor their dive depth and time against the maximum “safe” depth and time limit, regulated by the dive table, in order to avoid nitrogen narcosis.

5.4.2 The Missing Functionality

Unless special electronic communication devices are used, divers cannot communicate orally underwater. Because oral communication is rendered ineffective while underwater, when a diver wants to communicate with another diver some distance away, the individual must create sound by knocking two metal objects (usually any metal objects and the diver’s scuba tank). However, this method of communication has its flaws: there may not always be other metal objects around for the diver to use and if the diver is disoriented, the person may not be able to hit the tank.

Since water is approximately 800 times denser than air, sound waves diffract differently in water than in air. In fact, sound can travel faster and further in water. By having an emergency button that can start a sound emitting function on the dive computer, we can make use of this fact. This is also more advantageous to divers as reaching and pressing a button that is within an arms length will be much easier than using a solid object to hit a scuba tank.

5.5 Operational Modes

In this section, we look at how the user interacts with the device as well as the various display modes.

5.5.1 Interaction Zen

The prime reason that HCI existed was to provide a better platform for human-data interaction. The essence of the spirit in many Zen principles, concerns with aesthetics, mindfulness, connectedness [41] can be analogically applied to HCI. Like Zen, we apply aesthetics with balance and harmony to our design, focusing on user interaction, while maintaining usefulness.

Since using the same button for multiple functions may cause confusion, the dive computer uses a touch screen for most functions. The touch screen allows for more flexibility and ease when accomplishing a task. There is more interactive information present per screen and a task can be done by a single press rather than many presses of a button.

5.5.2 Display Modes

We have taken our overall design and categorized three types of the display modes based on the standard dive process: surface mode, pre-dive mode, and dive mode.

Pre-Dive Mode

During the pre-dive mode, information like direction and depth can be entered into the device and safety checks like scuba tank airflow rates are also conducted. The pre-dive mode automates the pre-dive planning and allows for divers to spontaneously change their dive plan while underwater.

Dive Mode

During the dive, the device is used in dive mode. Critical information, as described in Section 5.2, information on direction, dive time, depth, and scuba tank pressure along with additional information like maximum depth, maximum time, and vertical direction are shown in our design.

Surface Mode

The surface mode contains a summary page that lists the diver's last 10 dives for the diver to view. Because divers are required to record their dives in the dive log, In this mode, users can upload their previous dive records from the computer onto the device through an accompanying program. When the record is successfully uploaded, the record will be deleted from the computer. Users can also delete records through on-screen instructions.

5.6 Colour

In water, spectrum colours, also known as the colour combination that creates white light (such as red, orange and yellow), behave differently than they would in air [42, 43]; as the depth of water increases, spectrum colours start to become filtered out. For this reason, spectrum colours cannot be used in underwater technology. However, fluorescent colours (colors that emit bright lights) are not filtered out by the water. Thus, neon fluorescent colours are used to display information on the physical dive computer prototype.

5.7 Graphical Information Representation

As we can see from Section 5.3.3, all current dive computers are text based. Since we wish to make the information more easily processed by the brain, we are using an innovative approach – a graphic based dive computer. Depth and time, direction, and tank pressure, are used as organizational categories for the displayed information.

5.7.1 Depth and Time

In the traditional dive computers, it is difficult or even impossible for divers to alter their dive plan while underwater. We want to design a dive computer that offers this flexibility. Because depth and time are often viewed together when making important decisions (e.g. increasing the dive time or depth of the dive) are made, depth and time are grouped as one category. In Figure 5.6, there are three different ways of representing depth and time graphically.

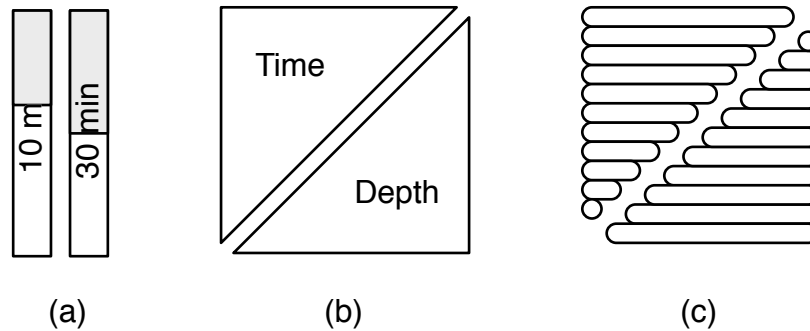


Figure 5.6: Graphical representation of depth and time.

In all three cases, the graphs are unfilled at the start of a dive. As the dive progresses, they will be filled appropriately.

5.7.2 Direction

While living above water, only a two-dimensional compass is needed because gravity allows us to know which way is up or down. However that is not the case when one is underwater. We need to have a compass to show horizontal direction and a graphic to indicate vertical direction.

The compass follows the standard design and markings are made every 45° . Like a normal compass the arrow always points towards the north. When the arrow aligns with the straight line, which is set by the user in the pre-dive mode, it indicates that the diver is swimming in a straight line along their intended heading.

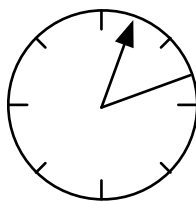


Figure 5.7: The compass.

To differentiate the vertical direction from the compass, a square is used to represent vertical direction indicator. The design of the vertical direction indicator is divided into two parts. When the dive computer display is facing upward (or above horizon) a circle is displayed in the centre of the graphic; similarly, when the dive computer display is facing downward (or below horizon) a square is displayed in the centre of the graphic. The implementation of vertical direction is done through an accelerometer. The length and direction of the arrow shows the amount and direction of tilt respectively.

While Figure 5.8 shows the graphic being displayed on screen for the vertical direction, Figure 5.9 shows the respective physical orientation of the device.

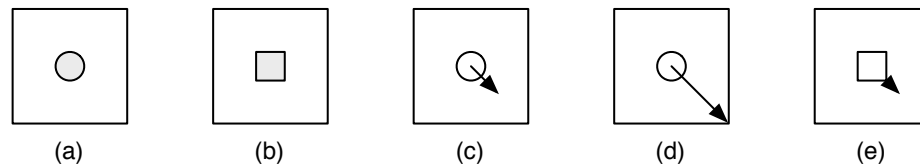


Figure 5.8: Vertical direction indicating, (a) the display is facing upward. (b) the display is facing downward (c) the display is facing in the direction of the arrow at 45° from face-up (d) the display is facing the direction of the arrow perpendicular to the horizon (e) the display is facing in the direction of the arrow at 45° from face-down.

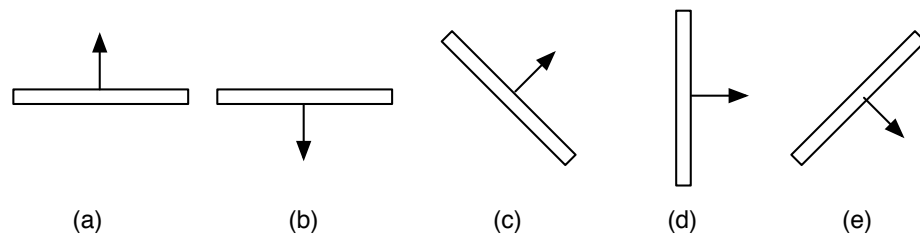


Figure 5.9: Device orientation according to graphics in Figure 5.8.

5.7.3 Tank Pressure

The design for the tank pressure is inspired by the battery indicator on a cellular phone. The graphic of the air left in the tank decreases as the dive goes on and to make things extra clear, the amount of air can also be shown as a percentage in the graphic. At the beginning of the dive, the dive computer will assume the initial tank pressure is the maximum tank pressure. As the dive goes on and tank pressure decreases, the graphic starts to become unfilled according to the amount of air left in the scuba tank.

5.8 Design Choices

We have noted the importance of choosing the right design. This includes not only the visual representation, but also the size of each component. In this section, we look

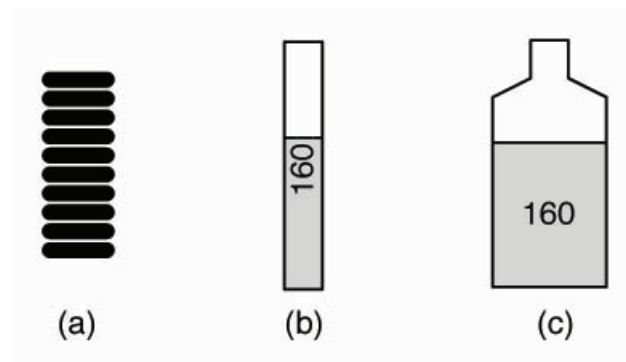


Figure 5.10: Graphical representation of tank pressure.

at both the design of individual components and the overall design of the display.

5.8.1 Graphics

Various combinations of the graphics introduced in Section 5.7, were experimented with. Presenting direction graphics in the largest size possible is necessary because:

1. Directional graphics are more accurately interpreted when the graphics are larger.
2. There are no numerical values available for direction graphics.

In order to yield maximal space for both direction graphics, we need to minimize the space taken up by the other two categories. For this reason, Figure 5.6 (a) was used in the final design.

Because the rectangular shape was chosen to represent depth and time, to make an obvious distinction with tank pressure, Figure 5.10 (a) was used in the final design.

5.8.2 Screen Navigation Buttons

The original design (Figure 5.11) had the buttons laid out along the bottom of the screen, starting from the right. The button on the bottom right corner of the screen,

the back button, was initially placed there for two reasons: (1) the back button is a button that is in most screens; (2) the fixed position of the button can make it easier for users to intuitively complete tasks because they have learned that the button is always going to be in that spot. However, after evaluation of the software prototype, it was shown that placing the back button in the bottom right is not the most natural position. In current computer applications and web browsers, the back button is located on the left side of the application. Following this trend, the back button was then moved to the bottom left of the screen.

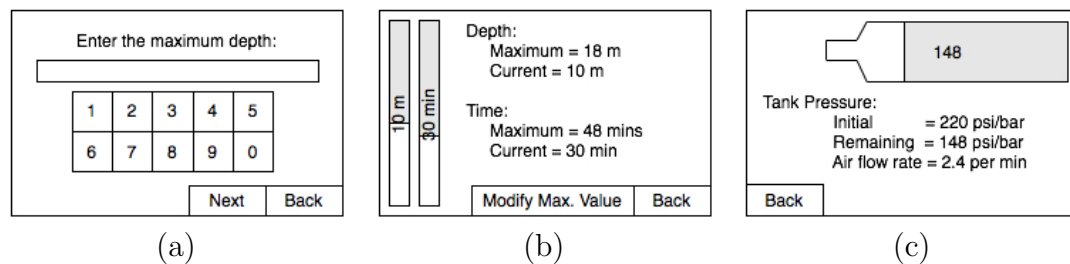


Figure 5.11: Two examples of original design, (a) and (b), with the “Back” button placed on the right, and the final placement (c).

5.9 Display of Choice

Since we are designing mobile devices, there is a trade off between the selection of graphical representations and screen size (which affects the amount of space available for information display). For example, a graphical representation may be easily and immediately understood, but can take up much room on the screen. Due to the space constraint for information display, those graphical representations may not be used which leads to a trade off between graphical representation and screen size. After much experimenting with various placements of graphics, we have come to the final design detailed in the following subsections.

5.9.1 Dive Mode

Figure 5.12 is a one-to-one representation of the dive mode's main screen. Once a diver clicks on any of the graphics, the screen goes into a screen filled with details about a selected group, either depth and time, tank pressure or direction. Figure 5.13, Figure 5.14, and Figure 5.15 show the detailed view of these three groups. To get back to the main screen from the depth and time, tank pressure or direction screen, the user can press the "Back" button on the bottom left of each screen.

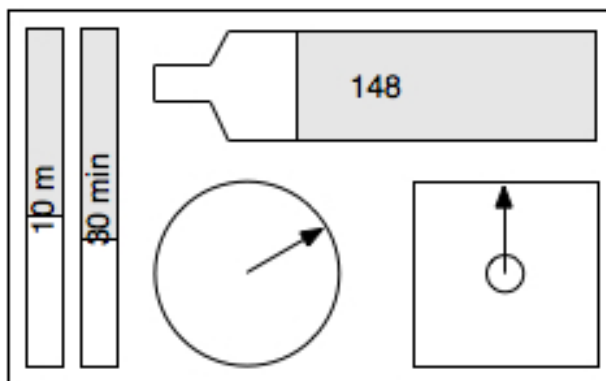


Figure 5.12: Main screen of dive mode.

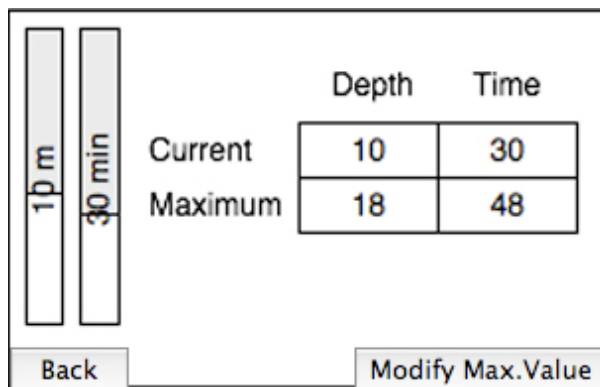


Figure 5.13: Detailed view for depth and time.

In dive mode, once the depth reaches 0 metre (i.e. the user has reached the surface), the dive computer automatically asks the user if it is the end of the dive.

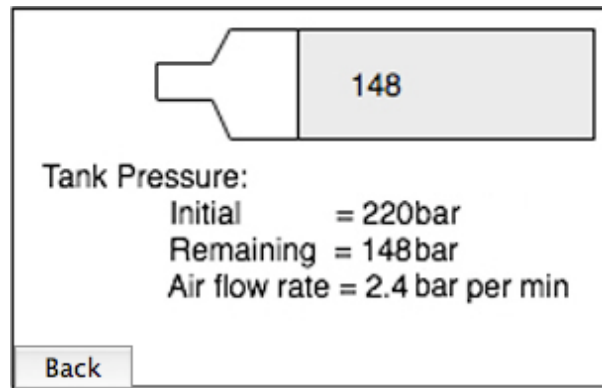


Figure 5.14: Detailed view for tank pressure.

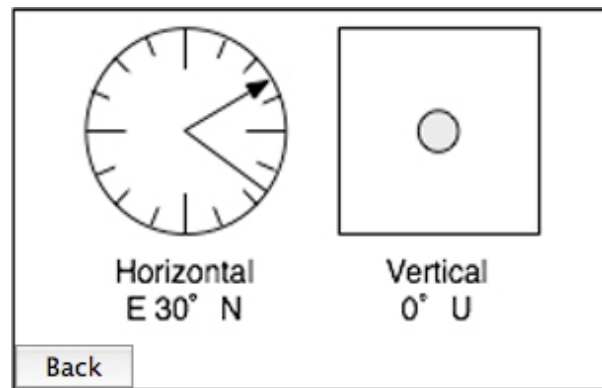


Figure 5.15: Detail view for direction.

If the user confirms that the dive has ended, the dive computer terminates the dive mode and switches to surface mode.

5.9.2 Pre-Dive Mode

The pre-dive mode starts once the “Go for a Dive” button in Figure 5.21 is pressed.

The pre-dive mode starts by asking the user to enter the maximum depth of the dive (Figure 5.16). If the maximum depth of the dive is changed at any time, the dive computer automatically recalculates the maximum dive time allowed.

After entering the maximum depth, the dive computer asks the user about the direction that they plan on swimming in (Figure 5.17). Then, the dive computer

Enter the maximum depth (meters):

- 10 +

Current Pressure Group: N/A
Maximum Time: 145 mins

Back Next

Figure 5.16: Diver has to enter the planned maximum depth.

checks for the regulator's airflow rate and scuba test by asking the diver to attach the regulator to the tank (Figure 5.18).

Please hold the device in front.
Then turn and face the direction
you intend to venture.

Back Set Direction

Figure 5.17: Diver is instructed to set their intended direction.

The computer show the results of all tests once they are completed. If no problems exist, then the computer displays a tests successful message and the "Next" button appears; if a problem is found, then a test fail message and the "Back" button appears (Figure 5.19).

Next, the computer asks the user about whether or not he/she wants to pair up the dive computer with another device. (Figure 5.20).

Once pairing is completed, the dive mode initiates. Alternatively, the pre-dive mode can be skipped and dive mode would be initiated once the diver starts to

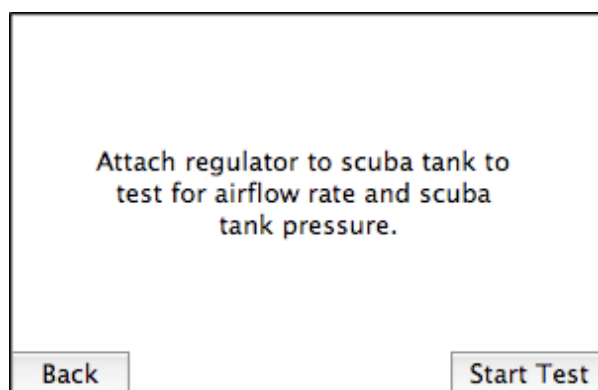


Figure 5.18: Dive computer picks up information regarding airflow rate and tank pressure with the sensors that is part of the regulator.

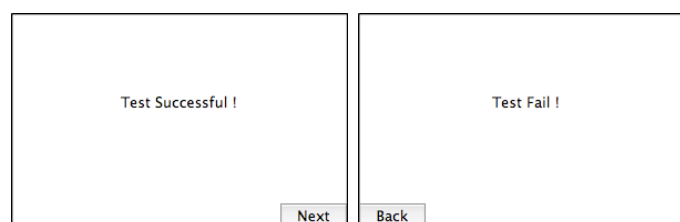


Figure 5.19: Test result screens.

descend.

5.9.3 Surface Mode

Figure 5.21 is surface mode's main screen. The graph below the two buttons on the top of the screen shows the different combinations of maximum time and depth that is currently possible for the diver. The "Dive Records" button can let the diver view past dive records while the "Go for a Dive" button will switch the dive computer into the pre-dive mode (Figure 5.16).

The "Dive Records" button takes the user from the main screen to the main records screen. There are ten buttons, each representing a particular record, that make up the main record screen. If a record already exists, a numeric number will be used as the button's label but if no record exists, then the word "Empty" is used as

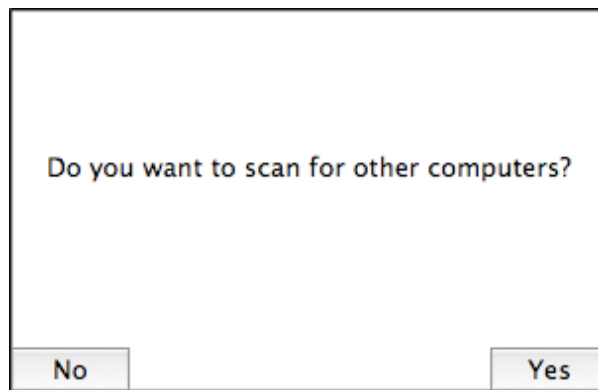


Figure 5.20: Pairing dive computers.

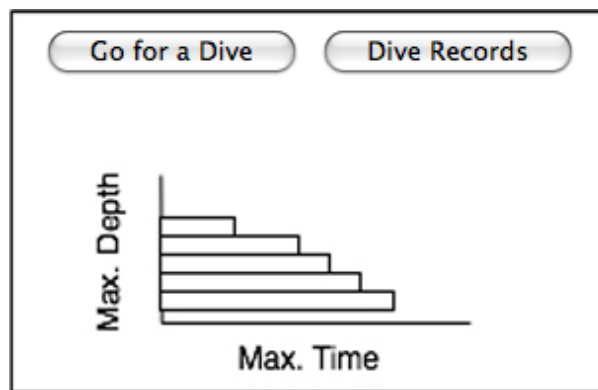


Figure 5.21: Main screen for surface mode.

the label instead.

When the user presses a button that has a numeric number, a detailed version of that particular record will be displayed on the screen (Figure 5.23).

5.10 Physical Device

So far, there has been an emphasis placed upon the displayed information but in order to create a well-rounded computer design, attention has to be paid to full interaction between the user and the device. Figure 5.24 represents the appearance of the physical dive computer. The dive computer is a capsule shape with round ends to reduce water resistance by cutting into the water.

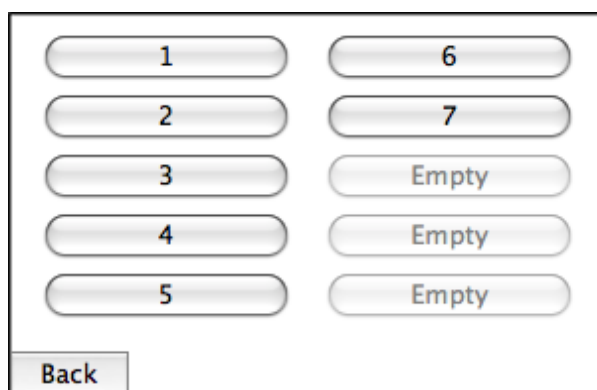


Figure 5.22: There are seven valid dive records currently existing on the dive computer.

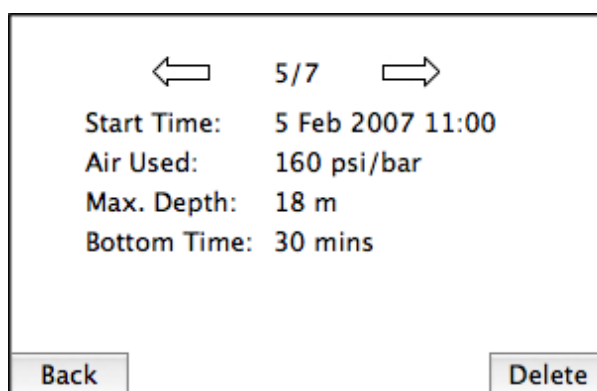


Figure 5.23: An example of a complete record.

To attract attention, the diver can use a start and stop button to control an audible signal emitted by the dive computer.

5.11 Warnings

The dive computer uses audio, visual, and motion warnings which can all be terminated by pressing the stop button located on the device. The audio warning can be activated by pressing the “Alert” button; once activated, the audio warning uses beeps emitted at a regular interval whereas the motion warning manifests itself by vibration. In the collaborative setting, the diver’s computer emit sound and light;



Figure 5.24: The dive computer.

while all network computers get a vibration signal. In addition to pressing the “Alert” button, warnings are also issued when a dive computer is about to be out of range.

5.12 Conclusion

In this chapter, we studied the dive process and have designed a graphic based dive computer based on our understanding and knowledge of this process. We have also shown the process that lead to the detailed final design.

Chapter 6

Computer Supported Collaborative Work

Recreational scuba diving is a socializing event where scuba divers plan and dive in groups. However, current dive computers in the market (see Section 5.3.3) are strictly designed to be used by single users. In this section, we incorporate collaborative aspects into the dive computer design as described in Chapter 5.

While scuba divers are underwater, they frequently stop to allow the dive leader to check their status. However, having to make frequent stops during a dive takes away joy and time. In addition, the dive leader has to constantly look back to see if the entire group is still together. Without collaborative support, this incredible experience can turn into a dreadful task.

Scuba diving is a social activity, in which divers are encouraged to dive in a group of two or more people. Group diving makes the adventure more enjoyable; it allows divers to share their experiences before, during, and after the dive. There can be many instances where someone has an interesting find, yet they cannot share it because they are not able to attract the others' attention. Moreover, group diving

serves as a safety precaution for both the pre-dive and dive stage. Before entering the water, divers must pair up to check if all equipment is attached and working properly. While underwater, each person provides emergency assistance to everyone else in the group. This is especially true for the group leader; in addition to leading the dive, they are also responsible for the safety and whereabouts of group members.

Without expensive equipment, divers are only able to communicate underwater by using hand signals. This may not be the most effective form of communication because visibility varies with environment changes. In addition, current dive computers only support single user use; they only monitor time, depth, and air supply for a single user. It is not interactive for divers as they are not able to communicate or check the status of other divers within the group.

Due to the lack of support for collaborative work in current dive computers, our goal is to design a dive computer to support collaboration between recreational scuba divers. Our design is based on work completed in Chapter 5. Our development is based on the following goals: (i) to support underwater collaboration, (ii) to provide a better means of communication, (iii) to increase safety precautions, and (iv) to increase awareness.

By supporting underwater collaborative work on a dive computer, we are allowing divers to interact with each other through an additional channel. By the use of instant messaging, we have opened up conversations between divers without the requirement that they be face-to-face. In applications such as crisis systems mentioned in “Crisis Management using Mobile ad-hoc Wireless Networks” [33], it is proven that we can increase safety precautions with the help of CSCW. We have attempted to increase awareness with Papadopoulos’ suggestions (see Section 6.2.5) made in “Improving Awareness in Mobile CSCW” [20] in mind.

In this chapter, we design a paper prototype and a software mockup of a collaborative dive computer system that is integrated with a messaging system, multiple divers' status view, localization of divers' position, and a safety alert enhancement.

6.1 Research Goals

According to the CSCW system taxonomy, our system could be classified as “same time, same place” [6]. However, our system differs from the traditional “same time, same place” CSCW systems which deal with face-to-face communication. Since the diving environment is not the same as a meeting room, the dive computer is expected to include special features that are not addressed in most of the current synchronous co-located CSCW systems.

6.1.1 Underwater Collaboration

Group diving is a social activity. Divers are encouraged to dive in groups mainly for fault tolerance purposes. Divers in the same group can help and look out for each other. To support collaborative work underwater, the first issue we want to address is communication. By exchanging information, group members can gain knowledge such as: who needs help and possible danger(s) at a nearby location. Furthermore, by providing the capability to crosscheck each other's status, it increases the chance of spotting emergencies such as “low on air”. Lastly, if an emergency arises, there should be a way to send an alert quickly to attract other dive buddies' attention. Generally speaking, diving is a high-risk activity; thus, underwater collaboration should be focused on improving the overall safety and reliability.

6.1.2 Communication

In a dive group, reliable communication can guarantee an increase in diving safety and provide a better way for divers to share information underwater.

Without additional expensive equipment, divers can only exchange information underwater through the use of hand signals. A dive computer with a messaging system can provide better support in environments that have unpredictable changes.

Though hand signals are a more efficient way to communicate when dive buddies are nearby, the visibility underwater may vary depending on the weather and locations. Thus, by solely relying on hand signals, reliable communication cannot be guaranteed – this presents a serious problem if an emergency should emerge. One of the most important features that the messaging system should include is the ability to send a message with a minimal number of steps.

6.1.3 Increase Safety

Since diving is a high-risk activity, safety is one of the most important issues that we want to address. First, the new dive computer design must provide an effective and efficient way of displaying group status such that divers do not make frequent stops and review all statuses. If someone notices that their buddy's air supply is low, then they can send out a message to alert that particular person. For dive leaders, the dive computer can help ensure that the activity is under control, and emergencies are handled quickly. Moreover, when a diver detects danger, he or she should be able to broadcast that information as soon as possible so that the appropriate measures can be taken.

6.1.4 Increase Awareness

According to the definition of awareness, awareness is the knowledge about the state of the environment and the update with the environment changes [21]. In an underwater environment where changes are always unpredictable, we want to be aware of our surroundings. Gutwin and Greenberg [21] identified three elements, within the context of awareness, that must be answered when designing collaborative systems: “who”, “what”, and “where”.

In our situation, “who” should pose knowledge such as “who is nearby” and “who sent the message”. “What” should include answers to “is anyone in danger?”, “are there interesting findings?”, and “is anyone low on air?”. The “where” component should include the location of the message sender/person in distress, as well as, the planned direction that the group should be heading towards.

Furthermore, the dive computer should increase presence awareness, identity awareness, action awareness, and situation awareness. In fact, more importantly, safety alert should attract the attention of other group members immediately so that quick countermeasures can be carried out in an emergency.

6.2 Related Work

6.2.1 Mobile Devices Supporting Collaborative Work

Groupware supports group members by creating a shared virtual environment to share artefacts or self-representations. Since group members can be distributed geographically in different places, mobile devices are demanded to support such needs. Currently, portable small devices are used in collaborative scenarios, especially in disaster notification management. However, these devices have various constraints in supporting group work, due to limited screen size, input facilities and communi-

cation intermittence. One important paradigm in any mobile device application is context-aware computing [44] in which the user of contextual information is described associated with location, time, nearby individuals and devices, and user activities. Since our application also includes social context, the communication model needs to support both multicasting and unicasting within a group. In addition to delivering data to multiple receivers to allow members to be aware of group activities, private conversations are also available from between members of the group. Peer-to-Peer architecture can also support such group communication in CSCW system [45].

To design for collaborative mobile applications, we need to follow the typical development process [46] (see Figure 6.1). First, is focusing on collaborative tasks. In our application, the dive computer needs to provide communication and group members' location. Second, is the supporting technology for groupware. Our application needs to support wireless underwater communication (see Chapters 3 and 4). Then these connections become the interaction and computational support for the system. In the social context, groupware should support different roles in the group i.e. the dive leader needs to check others' status to ensure the group's safety. Last but not least, the physical scenario where the groupware is applied. By providing reliable communication we are increasing safety precautions and presence awareness, which leads us to believe that networked dive computers can improve underwater collaboration.

6.2.2 Awareness

There are several different types of awareness [46]: social awareness, activity awareness, workspace awareness [21], and situation awareness. Maintaining awareness of a team member's development is associated with recognizing changes over time.

One can “monitor” the activities of their colleagues through the state change of one's status and to determine whether activities need to be adjusted. In like manner,

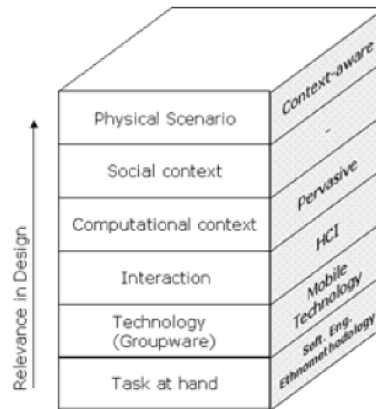


Figure 6.1: The collaborative development process.
[46]

their own activities should also be visible to their colleagues. As described by Gutwin and Greenberg in the basic characteristics of awareness [21], awareness is usually a secondary goal. In this chapter, the overall goal is to carry out the task of executing the dive as planned successfully and ensuring the safety of all members in dive group by maintaining awareness. Our design is helpful for improving presence awareness, activity awareness, situation awareness, and identity awareness.

6.2.3 Instant Messaging

Empirical research by Avrahami and Hudson [47] compares the instant message communication between work and social relationships. They find that people in a work relationship exchange longer messaging sessions on average, whereas those in social relationship spend less focus and attention to individual sessions. Strictly speaking, our application scenario is not in a “work” oriented situation. However, as the application of our device is in a safety-critical environment, we must treat the situation more seriously than a purely social environment.

Based on the constraints on hardware and group nature, short preset messages are suitable for our purpose. Through the use of the messaging system, we are providing an alternate but more familiar means of communication for recreational scuba divers.

6.2.4 Fire Fighters Communication System

The method of communication between fire fighters and the control centre mentioned in [48] is similar to our project with the exception that our device requires communication between two or more group members while the forest fire fighter and control centre is one to one communication. In the paper [48], mobile devices that provide maps for firefighters and the control centre to locate a target, such as a house on fire, are described. These networked mobile devices allow for quick variations in plans which is very similar to our requirements as our devices need to allow for quick variations too. In an environment where humans are slower than the organisms in their surroundings, there needs to be an easy way to catch everyone's attention if there is a sudden find; the finding(s) could easily disappear if a diver is too slow to look. It is much easier to look at one screen to stay updated on statistics and vitals rather than monitoring ten different people; similarly, it is easier to broadcast one message to everyone rather than trying to catch everyone's attention separately – the finding could disappear by that time.

6.2.5 Mobile Collaboration System

By examining handheld CSCW systems, a device that allows for collaboration by using networking technology [49], we can examine mobile CSCW systems. Ideally, mobile CSCW systems should respond to its local environment and be able to accomplish specific tasks. Key aspects needed to support awareness in mobile CSCW systems, as mentioned in Papadopoulos [20], include: “presence and activity aware-

ness of other group members”, “minimization of user disruption”, “access to awareness information and shared data regardless of location”. These three aspects act as the guidelines for the design of our device. The positioning system in our device provides presence awareness with individual divers possessing the option to view their surrounding group mates in a radar view and situation awareness with alerts sent to divers who are out of the range of the diving group.

In “Crisis Management using Mobile ad-hoc Wireless Networks” [33] the design of a crisis system is laid out but it is very different from our system. Contrary to our system, the crisis system uses GPS for communication and location of emergencies. It is an efficient way to deal with emergencies and minimizes the amount of time needed for help to arrive; however, generic GPS systems cannot be integrated into our device as GPS systems do not work underwater.

6.3 Limitations and Constraints

Since our application is deployed underwater, there are many constraints for common mobile devices and techniques. Conventional GPS and radio does not work well underwater because of the nature of conductivity of the medium and traditional antennae [50]. Most current underwater applications utilize acoustic communication. However, acoustic communication has a low data transfer rate and is easily interfered with in shallow water. Wireless Fibre Systems enable high-speed data transmission, but have limitations on vertical transmission range with the maximum depth of 10 metres. As Shelley [50] pointed out, specialized underwater radio can eventually become a practical tool for improving communication between above water units and divers. By November 2005, such technology has reached into the market as a commercial product [51]. Our only question that remains is the possibility of scaling

down the physical size of the technology to be used between two dive computers. Details on communication technology can be found in Chapter 3.

6.4 Design

The user interface is designed based on the four research goals stated in Section 6.1 with a special focus on increasing awareness. The goal is to make the design the simplest it can be with a minimal number of clicks needed to accomplish a task and a limited number of screens needed when searching for information.

When designing the messaging interface, one of the first decisions that had to be made concerned whether or not a touch screen keyboard or a preset messages system should be used. In the end, a system of preset messages was chosen because of its efficiency; we believe that users would be able to complete a task more quickly, thus communicating more efficiently, since preset messages would make sending the message a faster process and the diver would have less screens to go through. The only possible disadvantage with using a preset messages is that spontaneous messages cannot be sent; however, the probability of needing to use a spontaneous message during a dive is low. To offset this disadvantage, there is the option of allowing users to enter their own custom preset messages before their dive.

The number of screens is also minimized during the “Group Stats” view (see Figure 6.2). A bar graph was chosen to represent the amount of air left for everyone because the amount of air is a major statistic that every diver is concerned about. It was decided that the rest of the information, such as depth, is similar between all of the divers in a group so it was elected that only one version of all the information, except for amount of air left, would be displayed. By only displaying the individual diver’s information along with the amount of air every diver in the group has left

together on one screen, the number of screens a diver would have to flip through to view important information has decreased. The decreased number of screens allows for a quick comparison of everyone's data while providing an efficient way to view everyone's statuses so the dive leader can oversee the overall status of the dive group and every diver in the group can look after each other.

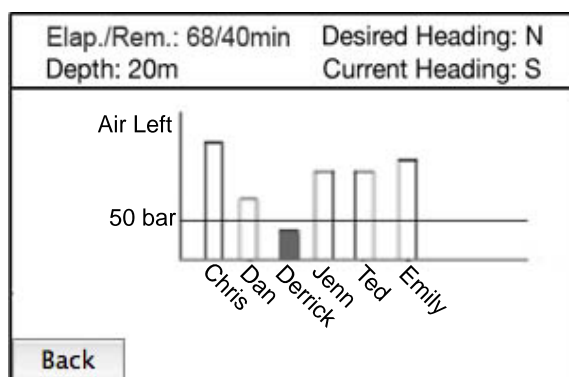


Figure 6.2: A screenshot of the group stat screen.

By minimizing the number of screens a diver needs to go through in order to access pertinent and relevant information, the amount of time a person needs to spend using the device is minimized thus allowing them to spend more time on their dive.

Another issue that affects the design of our device is the deployment environment. Because our users will mostly be underwater, and possibly wearing gloves, while using this device, the number and sizes of the buttons used will be affected. It was decided that the buttons used have to be large enough to press with gloved fingers and that the buttons would not have different colours as colours are filtered as depth of the water increases. In order to attract the attention of all divers in the group, when a distress call is issued by pressing the alert button, all the devices in the group will blink and vibrate to attract attention.

It was decided that only the dive leader could add others to the dive group in order to reduce the chance of omitting a diver. The process of pairing two devices

is envisioned to be similar to the way MSN messenger group chats occur, except the requested device has to accept the pair request before the two devices can be paired.

It was also decided that there would be no notification for the sender of a message if the message was successfully delivered; the sender would only be notified if their message failed to be sent to another party. We did not want to have a positive confirmation alert because it might cause a flood of messages to be sent to the sender if the sender were to message the whole group simultaneously. See Appendix B for the screen interaction sequence and Appendix C for detailed screen design.

6.5 Conclusion

Diving is a social activity, but there is a lack of collaborative support underwater with current dive computers. Current models only provide monitoring of a single user's status. In this chapter, we have designed and developed a collaborative dive computer integrated with a messaging system, the ability to view multiple divers' status and the divers' position, and an enhanced safety alert system.

Chapter 7

Implementation

Three types of implementation were used in the user studies and the simulation: paper prototype, software mockup, and network simulation. While paper prototypes and software mockups were used for three user studies to ensure usability of the dive computer design and functionalities, network simulation was used to check the viability of the network protocol used for the dive computer network.

7.1 Paper Prototype

All paper prototypes created in both the design stage and the user studies stage were created in one-to-one ratio to the marketable version of the device (i.e. everything on the paper prototype was the same size as how it would be realistically). This is because some studies results can be skewed if this ratio is not used. As mentioned in Section 2.2, regardless of the robustness of a design, a design would be considered useless unless it is useable (or has high usability values). Thus, size and proportion of the device and individual items on screen are all part of our interests. For this reason, we used Omnigraffle to create vector graphics prototypes so that images can be reused without blurring if results shows that a change in size if required. Throughout

Chapter 5 we have been showing examples of various paper prototypes that were used; a list of all the paper prototypes can be found in Appendix C.

7.2 Software Mockup

Similar to the paper prototype, because we do not wish for our results to be skewed, we have created a software mockup with one-to-one ratio to the actual design.

During this process we experimented with two different tools: Google Android and RealBasic.

Google Android is a free development platform for mobile platforms. As of February 2009, Android SDK has been updated to version 1.1 release 1. However, the latest version of Android available at the time of investigation in April 2008 was version m5 rc15. Thus, what is reported in this thesis is based on version m5 rc15. A number of graphic intensive applications has shown that Android can support features which we wish to implement. However, Android applications are largely dependent upon the Android API, the initial ramp up time before actual development is great. A more detailed analysis of Google Android can be found in “Communicating Like Nemo: ‘Scale-ability’ from a Fish-Eye View” [52].

RealBasic is a commercially available product which provides an intuitive IDE and uses VisualBasic-similar syntax. RealBasic also provides drag-and-drop interface for user interface development. Thus, making interface programming comparatively easier than some other languages. In addition, it provides the ability to compile the application into Windows, Macintosh, and Linux programs with the same code.

The software flow can be found in Appendix B.

7.3 Simulation

NS2 version 2.33 was used for implementing our network simulation. NS2 is an open source discrete event simulator and is widely used for simulation using various routing and multicast protocols, as well as, ad-hoc networks. It “has proved useful in studying the dynamic nature of communication networks” [32].

In our simulation we simulated the environment as well as the behavior as to how the nodes typically move (Figure 7.1).

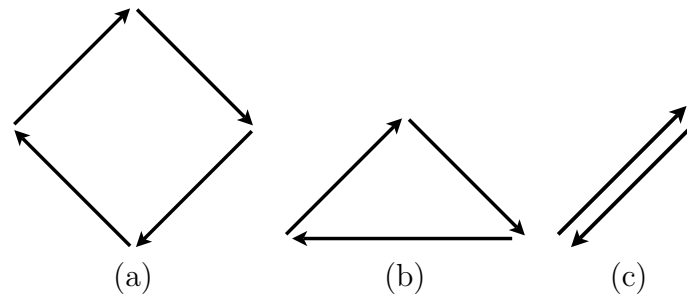


Figure 7.1: Examples of typical dive pattern.

We implemented the simulation with the following protocols: Ad Hoc On-Demand Distance Vector(AODV), Constant Bit Rate (CBR), and Transmission Control Protocol (TCP). AODV is commonly used for routing algorithms for multihop ad hoc networks. This routing protocol is source driven; the source periodically broadcasts routing requests and the destination node then responds through a single path [53]. The CBR is a type of traffic which offers to have a constant transmission rate. TCP is a transfer protocol which deals with passing data between two end nodes. TCP offers a reliable delivery such that it ensures the data is delivered when possible. In a general picture, AODV would first attempt to find the shortest path to the destination. After the route is found, CBR passes data at a constant rate to TCP which then transfer the data from the source node to the destination node. Alternate protocols are presented in Appendix A, while implementation details are presented in

Appendix G.

In order to mimic the real-world model, we have made assumptions based on real-world restrictions. We have given our program values that are higher than the possibility of an actual occurrence to accommodate for a margin of error. These assumptions are listed below:

Topography. In the Open Water Diver Manual provide by PADI (one of the largest international scuba diving certification program) it states that “[l]ocal laws regulate how close you have to stay to your flag, and how far boaters and skiers must stay away. For areas where no laws stipulate these distances, the rule of thumb is for you to stay within 15 metres/50 feet of your flag and for boats to stay at least 30 to 60 metres/100 to 200 feet away.” [54] Thus, we have set our plane to be 120 metres by 120 meters.

Sensing and Receiving Range. The sensing range was set as 400 meters and receiving range as 350 meters based on the values given by the specification of the LinkQuest UWM 1000 underwater acoustic modem [1].

Transfer Rate. Similar to the sensing and receiving range, we have set the transfer rate to be 17.8 kilobit per second based on the specification of the LinkQuest UWM 1000 underwater acoustic modem [1].

Node Movement. A common pattern was implemented based on typical dive paths in which scuba divers use patterns as shown in Figure 7.1.

Broadcast. The term broadcast in this case refers to emergency priority broadcast. As the name suggests, since it is an emergency, we assumed that the occurrence of this is very low and broadcasts once over a 30-minute interval.

Normal Messaging Frequency. Normal Messaging Frequency is defined as any non-priority message. In reality, scuba divers are unlikely to communicate with each other very often. However, since we wish to demonstrate the feasibility of how well the design of our network works, we have set the shortest time between messages sent from the same diver to be 90 seconds. Thus, the maximum volume of message traffic can be calculated as follows:

$$\text{Number of Messages} = \frac{\text{Dive Time}}{\text{Message Frequency}} \times \text{Number of Divers}$$

Faults Since there is no data for transmission failure rate and failure rate varies depending on the technology used, instead of using a probability of error rate, we accommodate for interferences mentioned in Section 3.1 by including 10 faults in all our simulations.

Swim Speed. We have set the speed for all nodes to be a random variable. The minimum swim speed for a diver node is set to an average adult swim speed (90 seconds for 100 metres), while the maximum speed is set to match the olympic record (53.12 seconds for 100 metres). Since failure rate decreases with slower speeds, this range is a conservative estimate.

In our implementation, we placed all nodes on a two dimensional topography, since points within the 3D sphere can be covered in a 2D plane. As a property of a sphere, the distance between any point of the surface and the center is exactly r (the radius). Therefore, any point within the sphere must be less than r . For this reason, we are able to effectively simulate a 3D environment on a 2D plane.

We have also randomized the start position, the direction in which the nodes move, and the speed in which they move for both buoys and divers with the following restrictions:

1. All buoys are within 25 metres of each other.
2. All divers are within 10 metres of each other.

7.4 Conclusion

In this chapter, we have shown the various tools that were implemented in preparation for the user studies and the network simulation. In the following chapter, we present details on how the user studies and simulation were structured and carried out, as well as, the outcome and the results.

Chapter 8

User Studies and Simulation

This chapter is divided into four sections: three user studies and one simulation. While the first user study focuses on the usability of the device for single users, the second user study focuses on the design of collaborative aspects. The third user study focuses on collaborative features.

8.1 User Study 1: Usability

This user study was the first of the three studies. It was conducted to accomplish two things: (1) to determine whether a graphical representation of information or plain text is more useful to users; (2) to compare the effectiveness of digital and analog devices. The dive computer which we used for this study predates the collaborative version.

8.1.1 Study Participants

All study participants were fully certified PADI open water divers. This requirement was set in place because the design uses calculations/units based on information given

by PADI; if the participant is not PADI certified, then their calculations/units will differ and their safety may be compromised.

A sample group of five certified PADI open water divers between the ages of 22 to 35 were chosen and surveys were emailed to them. This sample group consisted of three advanced level divers and two novice divers with the male-female ratio being four to one.

During the time that this report was written, there were only two responses to the prototype evaluation request. Both responses were from novice male divers whose ages are in the range of 20 to 29 and both use the console style dive computer.

8.1.2 Study Methodology

The user evaluation was conducted by sending a copy of the software prototype, with an introduction to the software included, a questionnaire and a consent form to the sample group. A questionnaire was chosen to be the format of evaluation because of the geographical distance between the participants in the sample group; it was not geographically feasible to arrange for interview user evaluation sessions.

The software introduction that was included with the software prototype that was sent was written in order to: (1) give the evaluators a brief overview of how the device was intended to be used; (2) enable the evaluators to understand how the device works without using simulations.

8.1.3 Questions and Reasoning

The questionnaire that was sent to the sample group consisted of six main questions. It started with a few questions about the general background of the evaluator in order to ensure that there was a diverse sample group. The background information can also show us possible trends in preferences with a particular age group or divers with

a certain amount of diving experience.

The first two main questions were used to evaluate the display of information divers preferred, graphical or plain text, and whether or not it was effective. If the user answered "console style design" to question 1, then questions 1 and 2 could also be used to analyze the effectiveness of digital and analog representation.

Questions 3 and 4 were mainly designed to gain user feedback on the quality of graphics used in the design. Questions 5 and 6 were written in case any other analysis on an aspect of the device was missed; these two questions provide the user a chance to comment on the device.

8.1.4 Results

The general reception of the developed dive computer design was positive. Users indicated that they liked the way that critical information was displayed and preferred graphical representation of information over plain text and analog form of representation. Having information in a graphical form is much more efficient. It allows divers to view approximate data values at a glance whereas plain text and analog forms of representation would require the diver to take time to process the information.

Users felt that the dive computer design should become more flexible (i.e. a user should be able to spontaneously change the dive route). Users also felt that "Dive Buddy" should include the dive information of more than one person, in the case of a diving buddy or a diving group, and the idea of using PADI graphical dive logs instead of text was also suggested. Currently, "Dive Buddy" uses text for records but using PADI graphical dive logs would increase the amount of information that a diver could gain from a glance. Figure 8.1 is a screenshot of how records will be shown graphically in a style similar to the PADI dive log form.

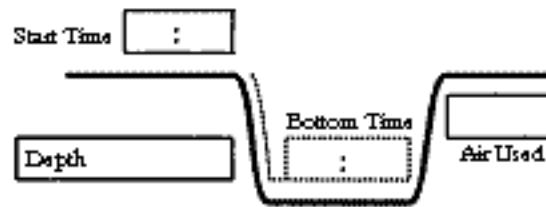


Figure 8.1: A modified version of the PADI dive log.

Other requests included:

1. Having the inflation level for the the buoyancy control jacket shown in dive mode. This would especially help novice underwater photographers balance and hover in mid-water for their photographs.
2. Allowing divers to see the data of other divers who are on the same dive trip (i.e. allowing the dive computer to receive data from more than one sender). This would minimize the dangers of having to regularly stop for data comparison since divers would be able to compare data while swimming.

A conclusion cannot be drawn about response differences between analog and digital device users at the time of writing, due to the insufficient number of responses.

8.2 User Studies 2: CSCW (Pilot)

The focus of this user study was to evaluate the design of collaborative functions for our designed dive computer. The detail result of this section are reported in “Collaborative Dive Computer” [55].

8.2.1 Methods

A paper prototype of the interface was designed to allow users to evaluate the collaborative dive computer design without a heavy development cost. The paper prototype

contained a real sized representation of the user interface for each screen and each screen was printed and numbered separately. Included with the paper prototype was a navigation tree that had all the paths a user could take during their interaction with the device ensuring a more realistic interaction with the paper prototype.

Based on the functions that our device can help users accomplish, we designed three commonly encountered scenarios that a user could encounter:

1. The setup and pairing of other devices when diving with a group (performed above water)
2. Viewing group status information while diving (performed underwater)
3. Sending a message to other group members during the dive (performed underwater)

These three scenarios were chosen in accordance with our research goal and because these scenarios require users to interact with all of the collaborative features in our device as well as moving through most of the screens. This allows for the user to have to use the whole device by themselves and comment on difficulties they might encounter if they were to use the device underwater.

In scenario 1, feedback about the communication system used above water was desired. In scenario 2, users were asked to evaluate the effectiveness and efficiency of viewing group status in supporting underwater collaboration. Users were also asked to comment on whether or not they thought that interface design could give a quick response if an emergency should arise, if the display of information given by the bar graph was effective and if the information provided by the device (e.g. direction) was necessary and useful. In scenario 3, we looked at the effectiveness of the messaging system and whether or not the localization of divers' positions could improve communication and awareness in underwater collaborative work.

8.2.2 Execution

The evaluation of the paper prototype was completed by five volunteer users with the intent to solve some of the design problems in the user interface. Three of the five users had five or more years of HCI experience while the other two users were regular Computer Science students. Certified scuba divers were unable to be recruited for the paper prototype evaluation due to the narrow target space. After the results from the evaluation of the paper prototype are incorporated into the software prototype, more experienced divers can be involved with the evaluation process.

Out of the five evaluations given, three of them were conducted in a research lab with two evaluators present to record feedback given. The other two evaluations were conducted at a location chosen by the user with only one evaluator to run the evaluation and record feedback.

All of the evaluations began with the signing of a consent form. Users were then given copies of the paper prototype; they were allowed time to familiarize themselves with the printouts and ask any questions for clarification. As none of the users were divers, they had questions about the meaning of some of the information and statistics that were displayed on the printouts. These questions do not cause a concern as it is assumed that real users of the device would be certified divers who would know what the information and statistics meant. After familiarizing themselves with the paper prototype, users were given the three scenarios previously mentioned and were left to guide themselves through the scenario without any external help from the evaluator(s). Users were asked to “think aloud” while working through the scenario; the “thinking aloud” process affected the speed at which the users performed each scenario but since our evaluation sessions were not timed, the reduced speed does not affect our results. By using the “think aloud” strategy, evaluators were able to record assumptions, comments and confusions that the users experienced; the information

obtained helped identify parts of the design that needed improvement and revision.

The last part of the evaluation sessions, after the completion of the three scenarios, consisted of the users filling in a short questionnaire about their overall experience with the paper prototype. The questionnaire consisted of questions about using aspects of the system design that we wanted more feedback on, such as whether an on screen keyboard or preset messages should be used. Users were also asked to comment on areas that they were not able to test, such as sound and vibration.

8.2.3 Results

From the completed evaluation sessions, several interface design problems were pinpointed. The most obvious problem, as remarked upon by three out of five users, was the inconsistent positioning of the “Back” and “Menu” buttons on the screen. Another problem that was also identified by three out of five users was the lack of units and labels on some of the statistical information displayed on the group stats and initial setup screens.

The general response towards using preset messages were positive; all users preferred the idea of using preset messages rather than an on screen keyboard. Out of the five users, three thought that preset messages were more effective than hand gestures while one user replied with “I am not a diver. I don’t know”. The last user suggested using a combination of messages rather than just preset ones.

A more serious issue that was identified involved confirmation screens during the process of pairing with other devices and sending a message; there were missing confirmation screens during these two processes. Without the ability to talk underwater, confirmation screens are essential for divers since they provide feedback and increase user awareness.

During the prototype evaluation, one user provided many useful suggestions on improving diver awareness when receiving messages from other users. It was suggested to include the option of allowing the receiver to view the sender's relative location; in situations where the message is something like "Follow me!", this option is essential as without the location of the sender, the message would be useless.

Our overall results suggest that users performing basic use scenarios encounter minimal difficulty.

By allowing users to evaluate the design, inconsistency and interface design problems were identified thus saving time by eliminating issues to be resolved later. Feedback from the evaluation also allowed for criticism on areas where the collaborative aspect lacked. These issues were rectified before the software mock up and implementation phases began.

8.3 User Studies 3: CSCW

The goal of this user study is to evaluate the collaborative functionalities from a scuba divers' perspective, focusing on user satisfaction and usefulness of features. We have incorporated comments from previous studies in our software implementation.

8.3.1 Methods

A software mockup, with navigation and minimum networking capability which models the real software, was implemented to allow users to evaluate the collaborative functions. based on user satisfaction and usefulness of features.

In addition to the three scenarios in the pilot study (Section 8.2), we added a fourth common scenario – finding the location of a group member. The four scenarios are as follows:

1. Pairing with other devices when diving with a group (performed above water)
2. Viewing group status information while diving (performed underwater)
3. Sending a message to other group members during the dive (performed underwater)
4. Finding the current location of a group member (performed underwater)

For the newly added scenario (Scenario 4), users were asked to evaluate the effectiveness and usefulness of the relative position function. We looked at whether or not localization of divers' position could improve location awareness.

8.3.2 Execution

This user study was completed by six PADI certified scuba divers, mostly Dive Masters and Dive Instructors. We used the same approach as the pilot study where users were presented with the software prototype and were given time to click around. Then, users were given the four tasks one-by-one. During this process, users were asked to “think aloud”. At the end of the study, the users were interviewed and asked to complete a questionnaire. The interview consisted of three questions (listed below) and general comments; a sample questionnaire can be found in Appendix F.

1. How do you attract your dive buddy's attention?
2. If you are leading a dive, how often do you check up on the status of others?
3. Other than airflow, can you think of other group status information that you might want to know?

8.3.3 Results

Overall, the results were positive and as expected. No users reported to have any difficulties in performing any given tasks. All users think that messaging would be helpful in addition to hand gesture, especially as “our local conditions (cold water diving) often involve bulky gloves making it hard to make clear signals and they needed a good way to send message when buddy is not looking directly at you”. However, two out of six users prefer to have a “message sent” confirmation. While only five out of six users thought that knowing the approximate position of group member is useful, especially at low visibility situations, the sixth user answered with “to effectively supervise, you would want to be able to see them at all times.” A summary can be found in Table 8.1. Even though most of these questions can be answered with yes’ and no’s, most users elaborated on their answers. For instance, three users commented that there should be an option to switch between units displayed.

During the interview, users commented on the general design. One user commented on the usefulness of allowing users to set the direction at the pre-dive stage because it can help with situations like attempting to swim straight when close to curvy coast lines. In general, almost all users commented that our design can be highly beneficial for search and rescue missions, open water dive instruction, and locations where visibility of low. Another interesting suggestion that arose from the interview is to include a 2-player game of tic-tac-toe on the computer (to be played during decompression stops).

8.4 Simulation

For our simulation, we have set the minimum number of nodes as five. This is because there are three buoys deployed at the surface and we need at least two

divers to “communicate”. We have run our simulation 45 times for each number of nodes. Additionally, we have ran the simulation for up to 105 nodes in attempt to find a network breakage point as well as to show scalability of our design. In all instances, the simulation time was set as 3600 seconds to reflect a typical underwater dive duration. The message length was deemed to be a random number between 1 second to 2.5 seconds, with an uniform distribution for the probability of generating a number within the range. The message length was chosen to be between 1 to 2.5 seconds because the message list can generate different combinations of messages that all have varying lengths. Since, in general, a node sends a message every 90 seconds on average (Poisson arrivals) and the average message length is 1.75 seconds, the traffic intensity 0.006349 per node.

In Figure 8.2, we have shown the average and median package drop rate (detailed data points are shown in Table 8.2 and Table 8.3 respectively). The drop rate has included the errors caused by noise (Section 7.3), collision and a long waiting period for packet delivery. As mentioned in Section 7.3, the number of faults due to noise that occurred per simulation was set to 10 each time. Therefore, the packet drop error rate due to noise during each simulation depends on the total number of messages sent (as previously discussed in Section 7.3); the error rate due to noise can be given by Equation 8.1. The packet drop error rate due to collisions can be determined by referring to the average package drop percentage given in Table 8.2, multiplying by 100, and subtracting the error rate of packet drop due to noise (Equation 8.2). If a non-priority packet is dropped, then there will be no amendment measures. Conversely, if it is a priority packet that is dropped, the packet will immediately retry to resend itself (Section 7.3).

$$Error\ Rate\ (Noise) = \frac{10}{Total\ Number\ of\ Messages\ Sent} \quad (8.1)$$

$$\text{Error Rate (Collision)} = \text{Value from Table 8.2} \times 100 - \text{Error Rate Due (Noise)} \quad (8.2)$$

Each point on the graph indicates the average percentage of the packet drops that occurred in the 45 runs for each set of nodes. As we can see, the percentage is slowly increasing. This is caused by the increased traffic in the network. However, the percentage is still relatively small as we scale up in the number of nodes.

Figure 8.3 shows the average and median time for packets to be transmitted from one node to another. In this figure we can see that the transmission time increases in a linear fashion. It starts from below 0.01 seconds and slowly approaches 0.01 seconds when the number of nodes increase to 10 (i.e. seven divers). We can see that the median transmission time stays below 0.015 seconds as we scale up to 55 nodes. All in all, if we take the median as the worse case for transmission time required, all data can be transmitted in under 0.0308 seconds, which is faster than we expected. Table 8.4 is a detailed chart of the average transmission times and Table 8.5 is a detailed chart of the median transmission times.

Table 8.1: A summary of result for user study 3.

Question	Total Response	Yes	No	Maybe/ Depends
Did you encounter any difficulties performing any of the tasks described in the use cases?	6	0	6	0
Do you think it is effective to have preset messages in addition to hand gesture?	6	6	0	0
Do you find the preset messages a good option that will save time or would you use an on-screen keyboard to type custom messages if the device supported it?	6	6	0	0
Can you check other divers status while swimming without having to pausing for too long?	6	4	1	1
Is it useful to know the approximate position of your buddies? Why?	6	5	1	0
Would you find it distracting or useful to have multiple alerts like sound, vibration, text messages, and blinking lights?	6	0 (distracting)	5 (not distracting)	1
If you have seen a conventional dive computer do you think this is a more effective way to check status?	6	4	0	2

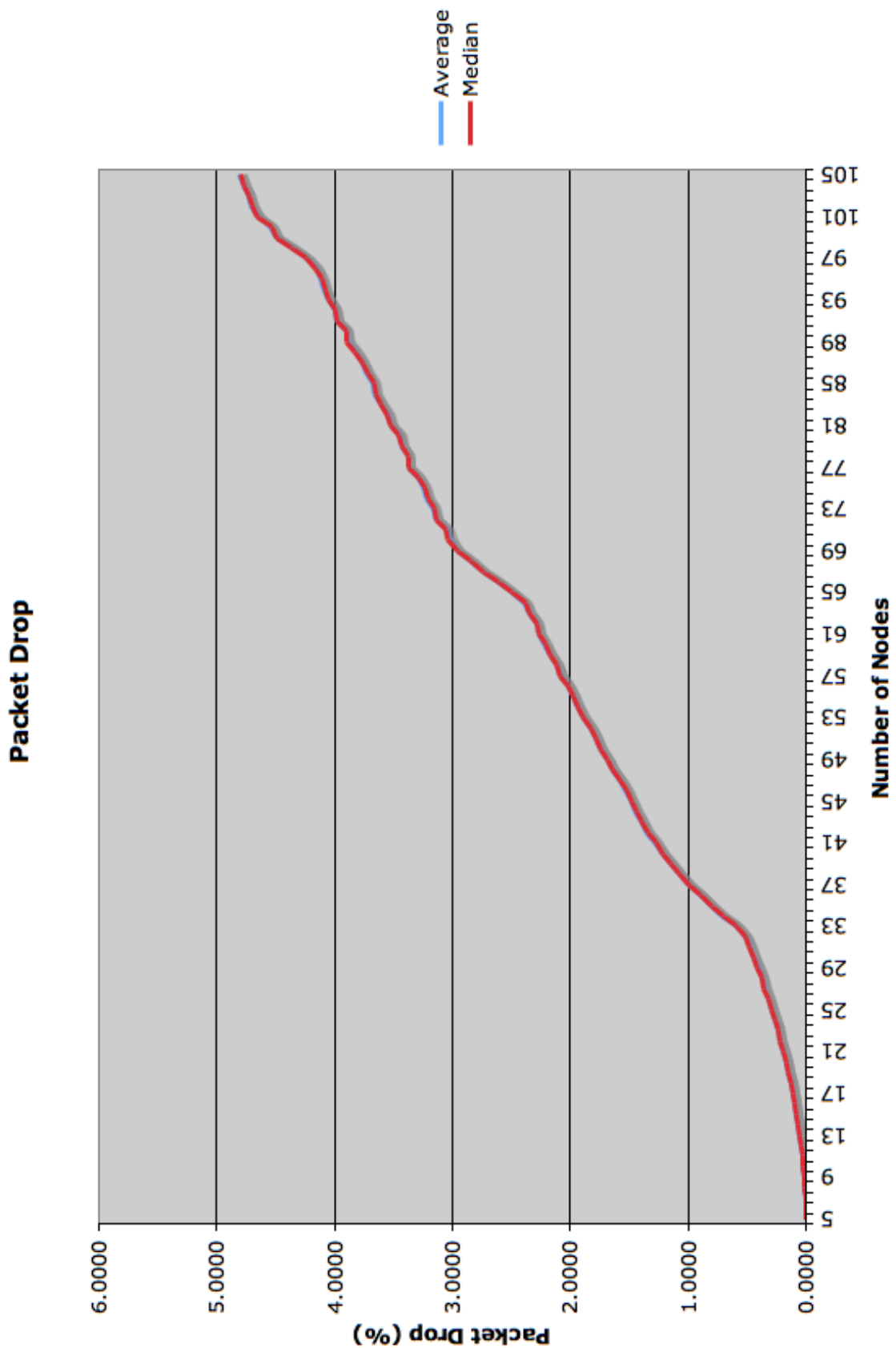


Figure 8.2: Average and median packet drop rate from 5 nodes to 105 nodes.

Table 8.2: Average packet drop percentage.

Nodes	Drop Rate (%)	Nodes	Drop Rate (%)	Nodes	Drop Rate (%)	Nodes	Drop Rate (%)	Nodes	Drop Rate (%)
5	0.0019	31	0.4724	57	2.0772	83	3.6145		
6	0.0035	32	0.5047	58	2.1154	84	3.6597		
7	0.0062	33	0.5909	59	2.1718	85	3.682		
8	0.0114	34	0.7122	60	2.2248	86	3.7389		
9	0.0162	35	0.8047	61	2.2741	87	3.7648		
10	0.0251	36	0.8963	62	2.2848	88	3.8104		
11	0.0338	37	0.9928	63	2.353	89	3.8801		
12	0.0418	38	1.0776	64	2.3915	90	3.893		
13	0.0538	39	1.1506	65	2.4955	91	3.982		
14	0.0648	40	1.2195	66	2.6105	92	3.9995		
15	0.0823	41	1.2844	67	2.7366	93	4.0466		
16	0.0943	42	1.3517	68	2.8438	94	4.0875		
17	0.1145	43	1.3943	69	2.9544	95	4.1143		
18	0.1258	44	1.4491	70	3.0174	96	4.1478		
19	0.1488	45	1.4804	71	3.0447	97	4.2454		
20	0.1694	46	1.5306	72	3.1341	98	4.354		
21	0.1924	47	1.5808	73	3.1713	99	4.4871		
22	0.2156	48	1.6397	74	3.2267	100	4.5446		
23	0.2339	49	1.692	75	3.2477	101	4.6444		
24	0.2643	50	1.7456	76	3.2869	102	4.7162		
25	0.2909	51	1.783	77	3.3703	103	4.7268		
26	0.317	52	1.8262	78	3.3756	104	4.7662		
27	0.3514	53	1.8886	79	3.4319	105	4.8123		
28	0.375	54	1.9315	80	3.4447				
29	0.4074	55	1.9832	81	3.5267				
30	0.4386	56	2.0185	82	3.5369				

Table 8.3: Median packet drop percentage.

Nodes	Drop Rate (%)	Nodes	Drop Rate (%)	Nodes	Drop Rate (%)	Nodes	Drop Rate (%)
5	0.0018	31	0.4772	57	2.0831	83	3.615
6	0.003	32	0.5109	58	2.1202	84	3.6433
7	0.0064	33	0.5888	59	2.1709	85	3.6689
8	0.0109	34	0.7095	60	2.2158	86	3.7213
9	0.015	35	0.8035	61	2.2734	87	3.7638
10	0.0254	36	0.8939	62	2.288	88	3.8258
11	0.0335	37	0.9927	63	2.3502	89	3.8954
12	0.0421	38	1.0764	64	2.3791	90	3.9007
13	0.0541	39	1.1537	65	2.5038	91	3.9825
14	0.0634	40	1.2128	66	2.6085	92	3.9973
15	0.081	41	1.2724	67	2.7419	93	4.0542
16	0.0924	42	1.3419	68	2.8541	94	4.0791
17	0.1116	43	1.4021	69	2.9571	95	4.1083
18	0.1256	44	1.4426	70	3.0394	96	4.1625
19	0.1483	45	1.4847	71	3.0619	97	4.2479
20	0.1698	46	1.5212	72	3.1352	98	4.3643
21	0.1917	47	1.5721	73	3.1456	99	4.4921
22	0.2169	48	1.6401	74	3.2076	100	4.5302
23	0.2337	49	1.6852	75	3.2406	101	4.6626
24	0.2632	50	1.7401	76	3.2836	102	4.7034
25	0.2868	51	1.7852	77	3.3701	103	4.7315
26	0.3174	52	1.8311	78	3.369	104	4.765
27	0.3526	53	1.8918	79	3.4336	105	4.7933
28	0.3756	54	1.9344	80	3.4503		
29	0.4096	55	1.98	81	3.5267		
30	0.4377	56	2.0222	82	3.5585		

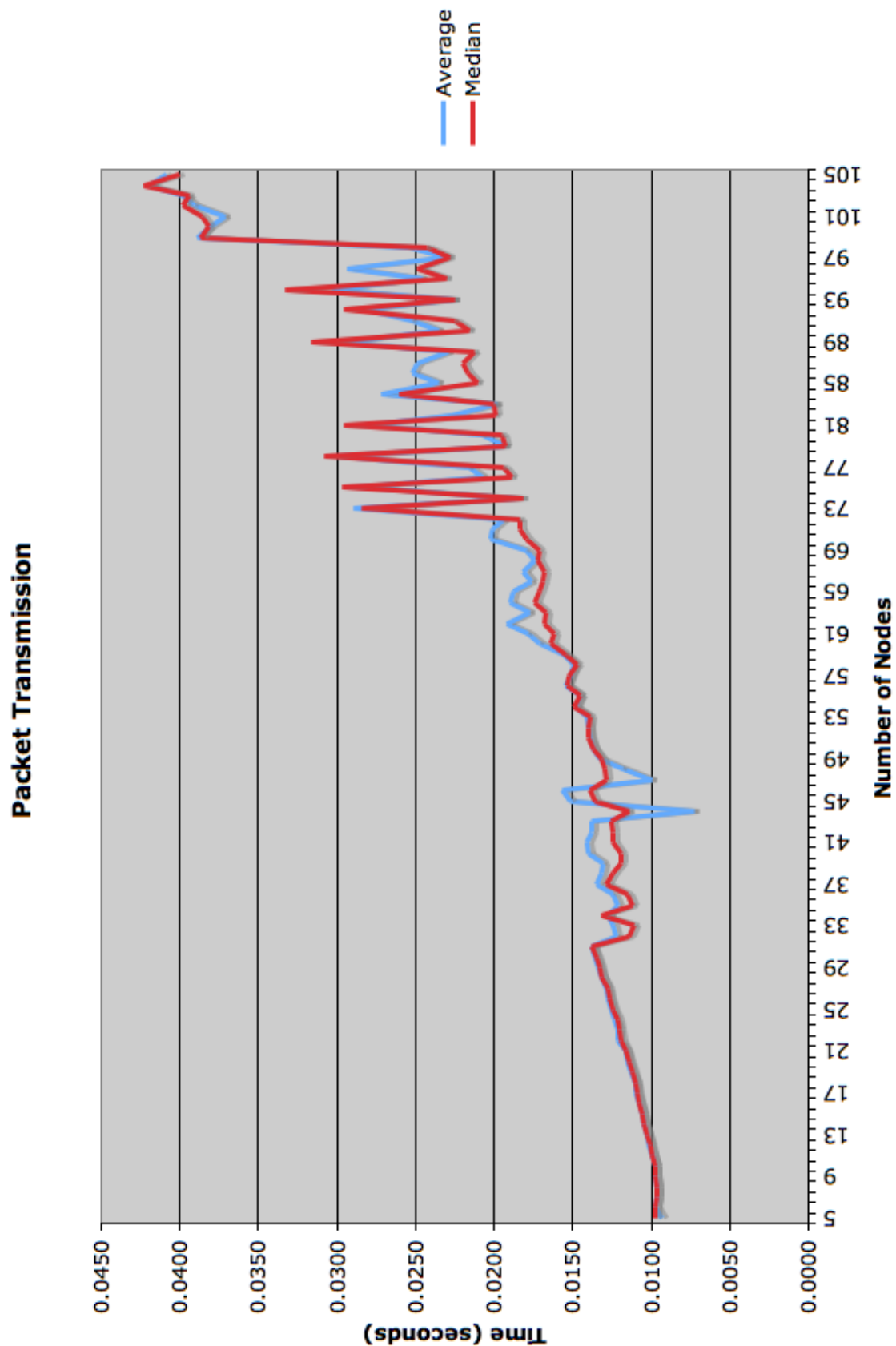


Figure 8.3: Average and median packet transmission rate from 5 nodes to 105 nodes.

Table 8.4: Average transmission time in seconds.

Nodes	Time (seconds)	Nodes	Time (seconds)	Nodes	Time (seconds)	Nodes	Time (seconds)
5	0.0093	31	0.0138	57	0.0152	83	0.0198
6	0.0098	32	0.0123	58	0.015	84	0.0272
7	0.0097	33	0.0124	59	0.0155	85	0.0236
8	0.0097	34	0.0128	60	0.0171	86	0.0252
9	0.0097	35	0.0122	61	0.0178	87	0.0248
10	0.0098	36	0.0124	62	0.0192	88	0.0229
11	0.0099	37	0.0135	63	0.0177	89	0.0295
12	0.0101	38	0.0131	64	0.0189	90	0.0235
13	0.0103	39	0.0131	65	0.0187	91	0.0252
14	0.0105	40	0.014	66	0.0176	92	0.0283
15	0.0106	41	0.0141	67	0.0181	93	0.0225
16	0.0108	42	0.0138	68	0.0174	94	0.0302
17	0.011	43	0.0138	69	0.0179	95	0.0246
18	0.011	44	0.0073	70	0.0202	96	0.0293
19	0.0113	45	0.0151	71	0.0201	97	0.023
20	0.0115	46	0.0156	72	0.0194	98	0.0254
21	0.0116	47	0.01	73	0.0289	99	0.0389
22	0.0121	48	0.0119	74	0.0184	100	0.0381
23	0.0121	49	0.0133	75	0.0287	101	0.0371
24	0.0124	50	0.0137	76	0.0207	102	0.0391
25	0.0125	51	0.0139	77	0.0216	103	0.0398
26	0.0128	52	0.014	78	0.03	104	0.042
27	0.0129	53	0.0141	79	0.0192	105	0.0409
28	0.0132	54	0.0148	80	0.0207		
29	0.0133	55	0.0145	81	0.0281		
30	0.0136	56	0.0155	82	0.0227		

Table 8.5: Median transmission time in seconds.

Nodes	Time (seconds)	Nodes	Time (seconds)	Nodes	Time (seconds)	Nodes	Time (seconds)
5	0.0097	31	0.0138	57	0.0153	83	0.02
6	0.0097	32	0.0114	58	0.0147	84	0.026
7	0.0097	33	0.0111	59	0.0154	85	0.021
8	0.0097	34	0.0131	60	0.0164	86	0.0216
9	0.0097	35	0.0112	61	0.0162	87	0.022
10	0.0098	36	0.0115	62	0.0168	88	0.0213
11	0.0099	37	0.0128	63	0.0167	89	0.0316
12	0.0101	38	0.0125	64	0.0174	90	0.0215
13	0.0102	39	0.0119	65	0.0172	91	0.0225
14	0.0104	40	0.0119	66	0.0169	92	0.0296
15	0.0105	41	0.0124	67	0.0168	93	0.0225
16	0.0107	42	0.0124	68	0.0172	94	0.0333
17	0.0109	43	0.0126	69	0.0171	95	0.023
18	0.011	44	0.0114	70	0.0179	96	0.0249
19	0.0112	45	0.0136	71	0.0183	97	0.0228
20	0.0114	46	0.0139	72	0.0183	98	0.0243
21	0.0116	47	0.0129	73	0.0284	99	0.0387
22	0.0119	48	0.0129	74	0.0182	100	0.0382
23	0.012	49	0.0131	75	0.0296	101	0.0385
24	0.0122	50	0.0137	76	0.0189	102	0.0398
25	0.0124	51	0.014	77	0.0194	103	0.0394
26	0.0127	52	0.014	78	0.0308	104	0.0423
27	0.0128	53	0.0139	79	0.0193	105	0.04
28	0.0132	54	0.0149	80	0.0195		
29	0.0133	55	0.0145	81	0.0296		
30	0.0135	56	0.0153	82	0.0199		

Chapter 9

Analysis

To gain feedback about the device, three user studies and one simulation were conducted.

The first user study that was conducted focused on the usability aspect of the device. Through feedback from the users, it was shown that the graphics size are proportional and they appeal to the audience. The graphics are simple and self explanatory while the things that need a larger space for clarity, like analog information, were given maximal space. Overall, it was felt that the dive computer design is easy to understand.

There were two main concerns that were raised from the discussion:

1. The clarity of the compass – the amount of checker marks on it make it hard to distinguish exact information.
2. The complexity of the vertical direction representation that is currently used.

As an additional support to the safety precaution, we can include an emergency light such that it emits light along with sound when the emergency button is pressed. By having both features, we can increase the chance of divers locating the diver in distress faster and easier.

The second user study focused on the design of the device, specifically the collaborative aspect. Subjects were asked about three topics, awareness, communication and safety precaution. Through analyzing the evaluation results from user studies, all subjects showed a positive feedback towards our design.

In the area of awareness, the feedback towards the positioning system providing presence awareness and identity awareness, which is a dominant feature in the collaboration system, was all positive. The positioning system helps install an extra sense of security in divers and provides a communication method to deal with emergencies, should one arise. The knowledge of the whereabouts of other group members enables divers to feel more safe and secure and if an emergency occurs, divers can immediately find out the location of the emergency and whether or not help is being provided. The system can also be used by groups of all sizes since initials, rather than full names, are used to identify people thus allowing larger dive groups.

A preference towards a vibration alert over an on-screen blinking light alert emerged amongst the feedback. However, using a vibration alert for every incoming item brings up a safety issue. If every alert caused a vibration, then divers would become desensitized to incoming alerts thus compromising their safety; a safety alert could potentially be ignored by a desensitized diver.

Results from interviews on the subject underwater communication suggested that communication can be improved if the messaging system is integrated with position information. Feedback also pointed out that there are three essential components needed in the messaging system: content, identity and position; four out of five testers also mentioned the possibility of using preset messages underwater. To improve the design's flexibility, the possibility of using default messages combined with customized messages set using accompanying software above water was considered.

The use of initials on the position screen helps the receiver immediately identify the sender thus processing the message faster; in a underwater situation where emergencies can occur in a heartbeat, processing messages faster will help increase the communication and safety of all divers. Initially, the usefulness of combining position information with a message was not realized until a tester suggested the integration of the two (see Figure 9.1). The necessity of this function is evident as some messages, such as “Follow me!”, would have insignificant meaning without the position information of the sender.

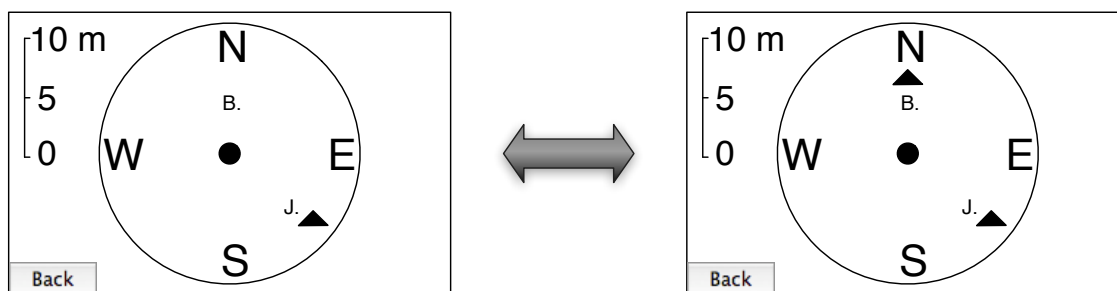


Figure 9.1: The suggested message alert screen.

The design’s modified safety precautions cannot be evaluated because the subjects interviewed were all non-divers. Since the design also involves lights, sounds and vibrations, all things which a paper prototype cannot do, evaluation of this portion of the design would not be undertaken in this study.

The third user study also focused on the collaborative features of the device but this time, focus was put onto the practicality of the device. Through the positive feedback from the user in this study, we confirmed our conclusion drawn from the second user study which is that our design can provide awareness and the warning functions can increase the overall safety, while satisfying our criteria for user satisfaction.

Through the analysis the results obtained from the simulation, it is clear that an improved version of the current technology can foster communication in our setting. The major issue is the size and weight of the current acoustic modems.

Chapter 10

Conclusion

In this thesis, we have focused on: (1) looked at the need for better dive computers; (2) demonstrated how current technology can be integrated into a better dive computer; (3) given some initial ideas on how dive computers can be improved in the areas of usability, information display, and wireless networking.

Our initial goal was to provide an alternative means of computer supported underwater interaction that would improve underwater collaborative work, safety and communication. Hence we have focused on areas that can drastically affect these three things. It is important to observe what current technology can be integrated into our system as using existing technology to build a better dive computer would bring it one step closer to reality as well as keeping the cost to a relative minimum. If we were to invent a new type of technology to be used in the dive computer, its distance from becoming a real object would be further as the time and cost needed to build new technology would be higher than using existing technology. Observing downfalls of current dive computers and making suggestions to improve a device is an essential part of the evolution of technology. It is only through the process of noting and trying to improve a device that better devices are created. Our initial suggestions

and ideas may be improved in the future which would lead to further increases in the quality of computer supported underwater interactions.

In the current world, we have moved from lollipop looking trees in the Nintendo 64 to realistic looking trees on the Wii and PS3. With all this advanced technology, there are no excuses to not provide an elegant visual solution because we are in a water-filled setting. Our user studies have brought us a step further in providing this visual solution such that, we have move from text-only to something that is easier to be comprehended by our brain. Additionally the simulation has shown the viability of such mobile communication.

We have gained considerable insight into what is required to build a fault tolerant underwater mobile communication system, as well as demonstrating the overall feasibility of utility of our design. However, there are still many questions to be answered. The following four directions are the first that need to be pursued to take our design to the next stage:

- It would be important to look at alternative methods of visually presenting information when we scale up our system.
- It would be interesting to extract the ideas presented in this thesis, such as network design, and apply them to other situations with high noise interference.
- Building and testing a physical prototype
- Investigation of the newly developed Underwater Global Positioning System [56].
- Psychology studies should be done to decide on the most effective way to attract divers' attention.

Our expectation is that further development of the overall design whose feasibility we have demonstrated in this thesis can lead to a patentable device.

Appendix A

Simulation Information

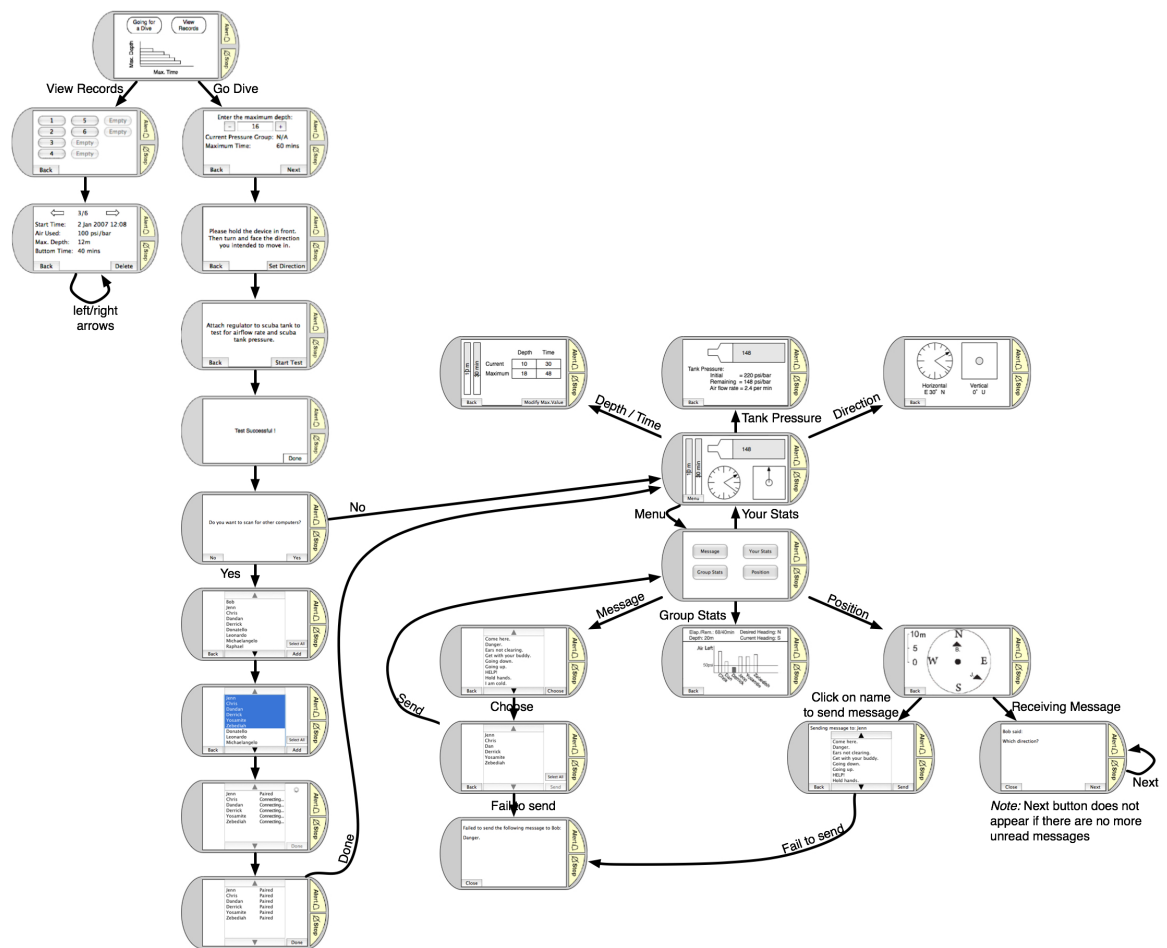
There are multiple options for routing protocols and transfer protocols in the NS2 simulator. In this appendix, we present a list of alternate protocols that are available in NS2.

Other than the AODV routing protocol which we chose for our design, DumbAgent, Destination-Sequenced Distance-Vector Routing (DSDV), and Dynamic Source Routing Protocol (DSR) can also be used. DumbAgent is used when one wishes to disable ad-hoc routing. DSDV routing protocol is commonly used for mobile ad hoc networks. It is a table-driven protocol and every node distributes information throughout the network at irregular intervals. Similar to the DSDV, DSR is also used for mobile ad hoc networks [57]. Similar to AODV, DSR operates in a reactive manner (see Section 4.2). This protocol is designed to handle a maximum of 200 high mobility nodes [57]. Because our design is an ad hoc network that is purely reactive when underwater, DumbAgent and DSDV are not a suitable candidates. While, DSR is very promising, its maximum limit for the number of mobile nodes supported does not suit our scalable design.

The alternate choice to the TCP transfer protocol is UDP. The major difference between UDP and TCP is the resend after a failed delivery. As mentioned in Section 7.3, TCP ensures data delivery whenever possible; whereas, UDP does not reattempt the delivery even if the delivery failed. Due to the safety-critical nature of our system, the use of UDP is unacceptable.

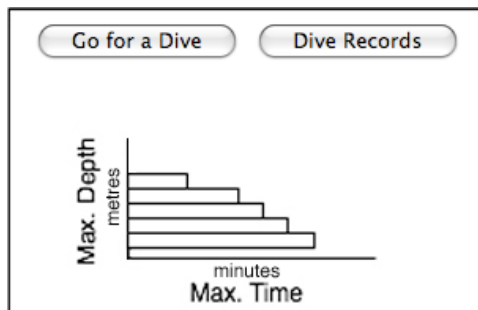
Appendix B

Software Flow



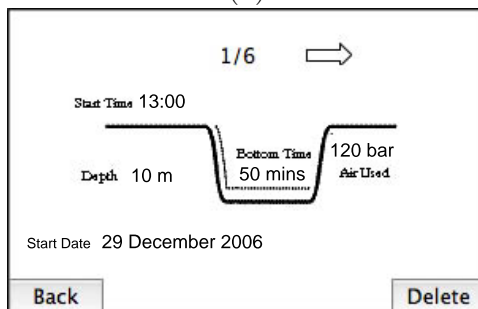
Appendix C

Paper Prototype



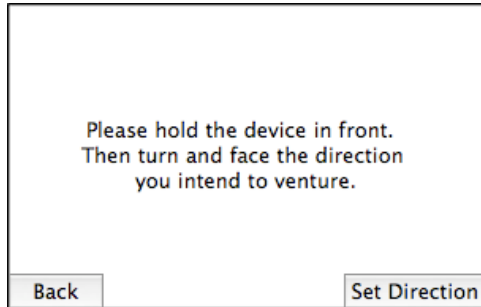
(a)

(b)

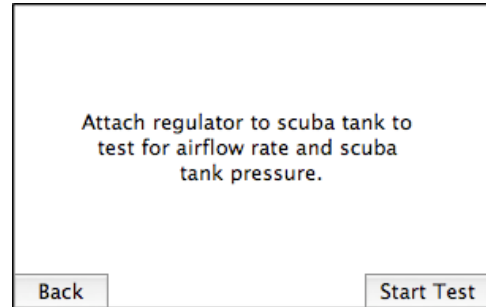


(c)

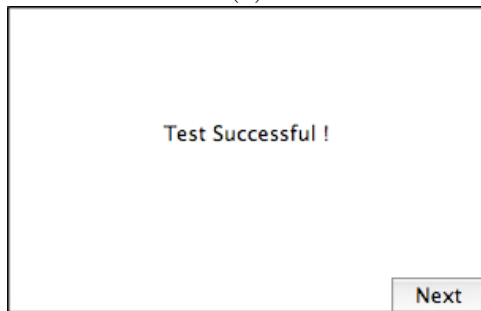
(d)



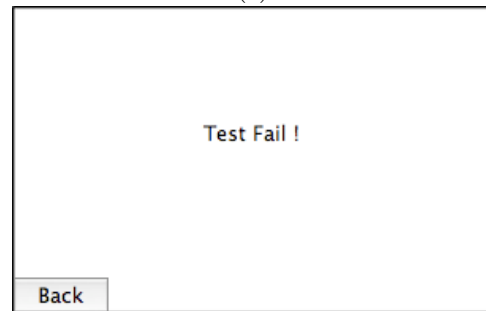
(e)



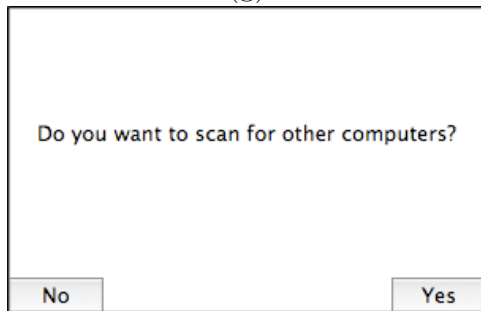
(f)



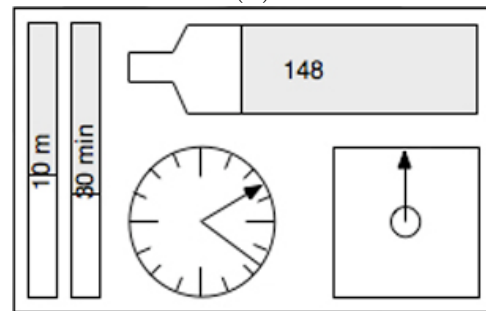
(g)



(h)



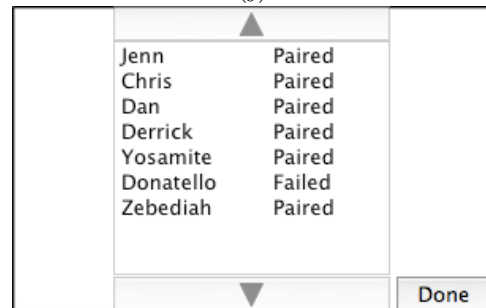
(i)



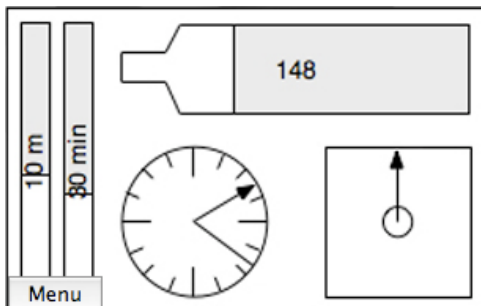
(j)



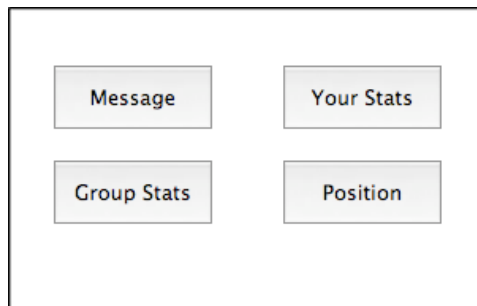
(k)



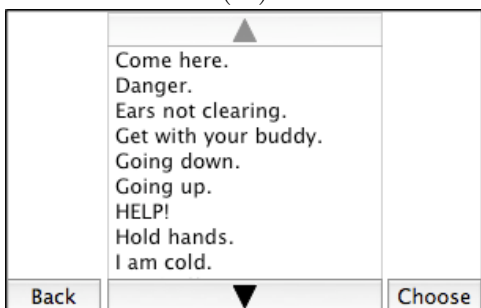
(l)



(m)



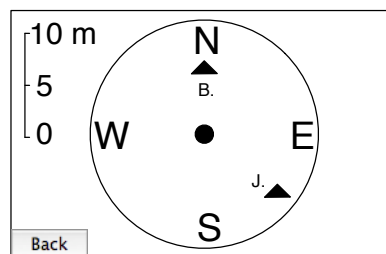
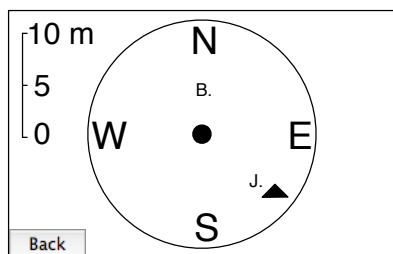
(n)



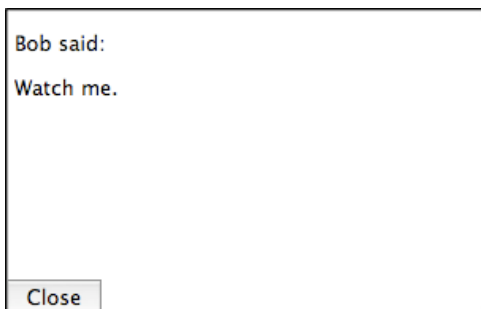
(o)



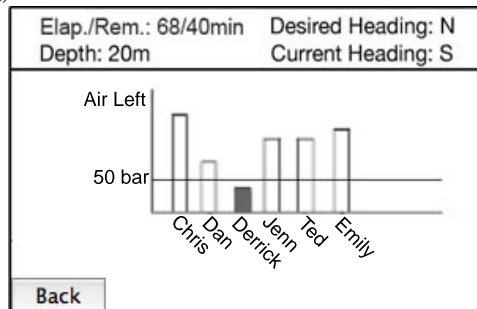
(p)



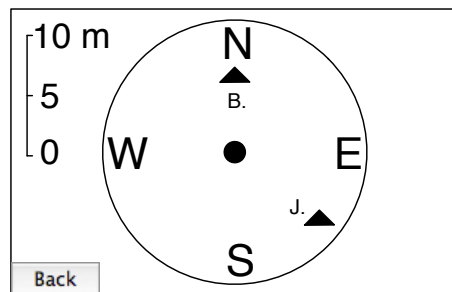
(q)



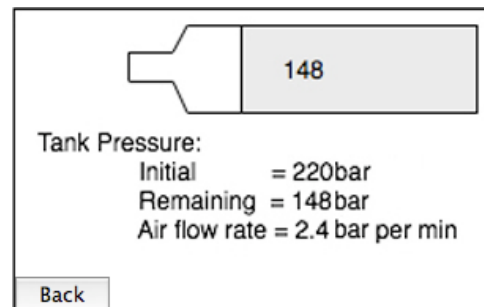
(r)



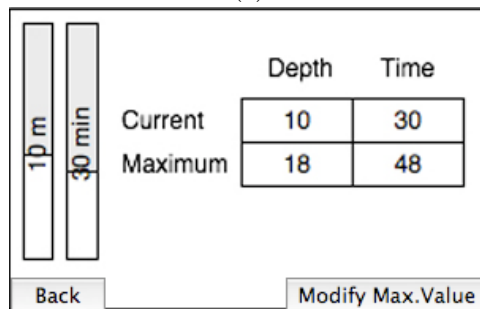
(s)



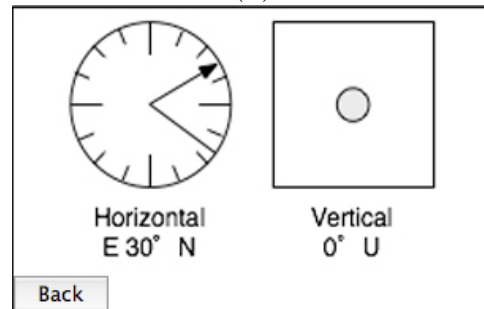
(t)



(u)



(v)



(w)

Appendix D

HCI Software User Study Evaluation Questionnaire

Please circle the appropriate categories.

Age: <20 20-29 30-39 40+

Gender: Male Female Others

Certification: Open Water Advanced Open Water

Others(please specify): _____

1. Do you currently use a dive computer? Why or why not?
2. Compare the current devices that you are using (i.e. the regular console, other dive computers) with “Dive Buddy”. What are the advantages and disadvantages of each?
3. How do you feel about the overall organization and layout of Dive Buddy (e.g. are the order of actions in a logical order)?
4. Is each of the graphics in Dive Buddy easy to read (i.e. the functionality of the graphic is understood just by observation)? If not, how can they be improved?

5. Are there missing features that you think should be included?

6. Other Comments:

Appendix E

CSCW Paper Prototype

Evaluation Survey

1. Did you encounter any difficulties performing any of the tasks described in the use cases?
2. Do you think it is effective to have preset messages rather than using hand gesture?
3. Do you find the preset messages a good option that will save time or would you use an on-screen keyboard to type custom messages if the device supported it?
4. Can you check other divers status while swimming without having to pausing for too long?
5. Is it useful to know the approximate position of your buddies? Why?
6. Would you find it distracting or useful to have multiple alerts like sound, vibration, text messages, and blinking lights?

7. If you have seen a conventional dive computer do you think this is a more effective way to check status?

Appendix F

CSCW Software Prototype

Evaluation Survey

Please circle the appropriate categories.

Age: <20 20-29 30-39 40+

Gender: Male Female Others

Certification: Open Water Advanced Open Water

Others(please specify): _____

1. Did you encounter any difficulties performing any of the tasks described in the use cases?
2. Do you think it is effective to have preset messages in addition to hand gesture?
3. Do you find the preset messages a good option that will save time or would you use an on-screen keyboard to type custom messages if the device supported it?
4. Can you check other divers status while swimming without having to pausing for too long?
5. Is it useful to know the approximate position of your buddies? Why?

6. Would you find it distracting or useful to have multiple alerts like sound, vibration, text messages, and blinking lights?
7. If you have seen a conventional dive computer do you think this is a more effective way to check status?

Appendix G

NS2 Implementation Details

Wireless Interface (unstate variables are left as default):

```

CSThresh_ : 5.5735e-11
RXThresh_ : 9.5081e-11
bandwidth_ : 2.1728515625e3
freq_      : (26.77e+3 + 44.62e+3) ÷ 2

```

Node Configuration:

```

Channel Type           : Channel/WirelessChannel
Propagation Model      : Propagation/TwoRayGround
Network Interface Type : Phy/WirelessPhy
MAC Type               : Mac/Simple
Interface Queue Type   : Queue/DropTail/PriQueue
Link Layer Type        : LL
Antenna Model          : Antenna/OmniAntenna
Max Packet in Interface Queue : 50
Error Model             : MultistateErrorProc

```

Bibliography

- [1] “Linkquest underwater acoustic modem models.” <http://www.linkquest.com/html/models1.htm>, April 2006.
- [2] B. W. Johnson, *The Design and Analysis of Fault Tolerant Digital Systems*. Addison-Wesley, 1989.
- [3] “Being human: Human-computer interaction in the year 2020.” <http://research.microsoft.com/hci2020/>, 2007.
- [4] “Pcpal 3-in-1 slim pc/laptop rf remote control, mouse, laser presenter.” <http://www.vavolo.com/productdetails.asp,mode,1,ProductID,3586,,.htm>, September 2008.
- [5] “Kensington - k33374us - wireless presenter with laser pointer.” <http://us.kensington.com/html/11190.html>, September 2008.
- [6] C. A. Ellis, S. J. Gibbs, and G. Rein, “Groupware: some issues and experiences,” *Commun. ACM*, vol. 34, pp. 39–58, January 1991.
- [7] “H2o audio reviews - ipod - waterproof headphones - mp3 players — reviews by technotalks.com.” <http://www.technotalks.com/reviews/h2o-audio-unveils-iphone-friendly-surge-amphibx-waterproof-headphones-and-armbands/>, March 2008.

- [8] “Evologics - hydroacoustic modems with sc2 intelligent underwater telemetry - naval technology.” <http://www.naval-technology.com/contractors/navigation/evollogics/>, September 2008.
- [9] K. Nagothu, M. Joordens, and M. Jamshidi, “Communications for underwater robotics research platforms,” in *Systems Conference, 2008 2nd Annual IEEE*, pp. 1–6, 2008.
- [10] S. Kurano, T. Ishiwata, and N. Konishi, “The study of the float buoy ranging system for the underwater vehicle,” in *Underwater Technology, 2000. UT 00. Proceedings of the 2000 International Symposium on*, pp. 161–166, 2000.
- [11] D. K. Pradhan, *Fault-Tolerant Computing: Theory and Techniques*. Englewood Cliffs New Jersey: Prentice Hall, 1986.
- [12] “Diveaeris.com - computers - atmos 2.” http://www.diveaeris.com/p-computers_atmos2.html, December 2006.
- [13] “Diveaeris.com - computers - epic hoseless dive computer watch.” http://www.diveaeris.com/p-computers_epic.html, December 2006.
- [14] “Diveaeris.com - computers - compumask hud.” http://www.diveaeris.com/p-computers_compumask.html, December 2006.
- [15] “Underwater digital interface.” <http://www.utc.co.il>, June 2008.
- [16] D. Stone, C. Jarrett, M. Woodroffe, and S. Minocha, *User Interface Design and Evaluation (The Morgan Kaufmann Series in Interactive Technologies)*. Morgan Kaufmann, March 2005.

- [17] “Nrc: Fact sheet on the three mile island accident.” <http://www.nrc.gov/reading-rm/doc-collections/fact-sheets/3mile-isle.html>, March 2009.
- [18] L. J. Bannon and K. Schmidt, “Cscw: Four characters in search of a context,” in *Proceedings of the First European Conference on Computer Supported Cooperative Work (EC-CSCW '89)*, (Gatwick, London, UK), pp. 358–372, Computer Sciences House, Slough, UK, 1989.
- [19] J. J. Cadiz, G. Venolia, G. Jancke, and A. Gupta, “Designing and deploying an information awareness interface,” in *CSCW '02: Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work*, (New York, NY, USA), pp. 314–323, ACM Press, 2002.
- [20] C. Papadopoulos, “Improving awareness in mobile csw,” *IEEE Transactions on Mobile Computing*, vol. 5, no. 10, pp. 1331–1346, 2006.
- [21] C. Gutwin and S. Greenberg, “The importance of awareness for team cognition in distributed collaboration,” in *Team Cognition: Understanding the Factors That Drive Process and Performance*, pp. 177–201, American Psychological Association.
- [22] C. Gutwin and S. Greenberg, “A descriptive framework of workspace awareness for real-time groupware,” *Computer Supported Cooperative Work (CSCW)*, vol. 11, no. 3 - 4, pp. 411–446, 2002.
- [23] S. E. Hudson and I. Smith, “Techniques for addressing fundamental privacy and disruption tradeoffs in awareness support systems,” in *CSCW '96: Proceedings of the 1996 ACM Conference on Computer Supported Cooperative Work*, (New York, NY, USA), pp. 248–257, ACM, 1996.

- [24] X. Lurton, *Underwater Acoustics: An Introduction*. Springer, 1 ed., December 2002.
- [25] L. Butler, “Underwater radio communication.” <http://www.qsl.net/vk5br/UwaterComms.htm>, November 2008.
- [26] E. M. Sozer, M. Stojanovic, and J. G. Proakis, “Underwater acoustic networks,” *IEEE Journal of Oceanic Engineering*, vol. 25, no. 1, pp. 72–83, 2000.
- [27] “Under water radio waves.” <http://www.qsl.net/vk5br/UwaterComms.htm>, June 2008.
- [28] “Why isn’t radar used underwater instead of sonar if radio waves are faster than sound waves?.” <http://www.physlink.com/Education/AskExperts/ae456.cfm>, April 2008.
- [29] M. Brain and C. Freudenrich, “How submarines work.” <http://science.howstuffworks.com/submarine4.htm>, April 2008.
- [30] T. Aoki and Y. Asao, “High-speed optical transmission,” January 1999.
- [31] S. G. Chappell, J. C. Jalbert, P. Pietryka, and J. Duchesney, “Acoustic communication between two autonomous underwater vehicles,” in *Autonomous Underwater Vehicle Technology, 1994. AUV ’94., Proceedings of the 1994 Symposium on*, pp. 462–49, 1994.
- [32] T. Issariyakul and E. Hossain, *Introduction to Network Simulator NS2*. Springer, 2009.
- [33] B. Tatomir and L. J. M. Rothkrantz, “Crisis management using mobile ad-hoc wireless networks,” *Second International Conference on Information Systems for Crisis Response and Management ISCRAM 2005*, 2005.

- [34] J. Su, J. Scott, P. Hui, J. Crowcroft, E. de Lara, C. Diot, A. Goel, M. Lim, and E. Upton, “Haggle: Seamless networking for mobile applications,” in *UbiComp 2007: Proceeding of Ubiquitous Computing 2007*, pp. 391–408, 2007.
- [35] G. Kortuem, D. Alford, L. Ball, J. Busby, N. Davies, C. Efstratiou, J. Finney, M. White, and K. Kinder, “Sensor networks or smart artifacts? an exploration of organizational issues of an industrial health and safety monitoring system,” in *UbiComp 2007: Proceedings of Ubiquitous Computing 2007*, pp. 465–482, 2007.
- [36] F. Siegemund, “A context-aware communication platform for smart objects,” in *2nd International Conference on Pervasive Computing*, Springer-Verlag Berlin Heidelberg, 2004.
- [37] W. Adjie-Winoto, E. Schwartz, H. Balakrishnan, and J. Lilley, “The design and implementation of an intentional naming system,” in *SOSP '99: Proceedings of the Seventeenth ACM Symposium on Operating Systems Principles*, vol. 33, (New York, NY, USA), pp. 186–201, ACM Press, December 1999.
- [38] G. Ananthanarayanan, V. N. Padmanabhan, C. A. Thekkath, and L. Ravindranath, “Collaborative downloading for multi-homed wireless devices,” in *Mobile Computing Systems and Applications, 2007. HotMobile 2007. Eighth IEEE Workshop on*, pp. 79–84, 2007.
- [39] J. Dean and S. Ghemawat, “Mapreduce: simplified data processing on large clusters,” *Commun. ACM*, vol. 51, pp. 107–113, January 2008.
- [40] “Dema show 2008.” <http://www.demashow.com/DEMA-Show-2008.html>.
- [41] G. Reynolds, *Presentation Zen: Simple Ideas on Presentation Design and Delivery (Voices That Matter)*. New Riders Press, January 2008.

- [42] “Color underwater.” <http://www.underwaterphotography.com/Underwater-Video/Lighting/Color-Underwater.asp>, February 2007.
- [43] G. D. Campbell, “Color underwater.” <http://www.deep-six.com/page77.htm>, February 2007.
- [44] G. Chen and D. Kotz, “A survey of context-aware mobile computing research,” Tech. Rep. TR2000-381, Dept. of Computer Science, Dartmouth College, November 2000.
- [45] C. Hua, Q. Jianfei, and H. Qingming, “A p2p architecture for supporting group communication in csw systems,” in *Computer Supported Cooperative Work in Design, 2006. CSCWD '06. 10th International Conference on*, pp. 1–5, 2006.
- [46] R. Alarcon, L. A. Guerrero, S. Ochoa, and J. A. Pino, “Context in collaborative mobile scenarios,” *Fifth International and Interdisciplinary Conference on Modelling and Using Context (CONTEXT'05)*, vol. 133, July 2005.
- [47] D. Avrahami and S. E. Hudson, “Communication characteristics of instant messaging: effects and predictions of interpersonal relationships,” in *CSCW '06: Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*, (New York, NY, USA), pp. 505–514, ACM, 2006.
- [48] J. M. Carroll, M. B. Rosson, G. Convertino, and C. H. Ganoe, “Awareness and teamwork in computer-supported collaborations,” *Interacting with Computers*, vol. 18, pp. 21–46, January 2006.
- [49] H.-W. Gellersen, “Guest editorial: Handheld csw: Personal technologies for collaboration,” *Personal and Ubiquitous Computing*, vol. 3, pp. 1–7, March 1999.

- [50] T. Shelley, “Radio waves transmit information underwater.” <http://www.eurekamagazine.co.uk/article/3325/Radio-waves-transmit-information-underwater.aspx>, 2005.
- [51] T. Shelley, “Underwater wireless product reaches launch.” <http://www.eurekamagazine.co.uk/article/7663/Underwater-wireless-product-reaches-launch.aspx>, 2006.
- [52] J. Wong, Y. O. Yazr, D. Minifie, C. Gibbs, J. Muzio, and Y. Coady, “Communicating like nemo: scale-ability from a fish-eye view,” in *Google Scalability Conference 2008*, June 2008.
- [53] C.-C. Wang, “Ns2 training course - wireless network.” <http://163.17.21.42/NS2/content/0709m/Wireless%20Network.ppt>, July 2008.
- [54] K. Shreeves, *Open Water Diver Manual*. Rancho Santa Margarita: International PADI Inc., 1999-2004.
- [55] J. Wong, D. Huang, C. Todorov, and D. Plotsky, “Collaborative dive computer.” C SC 578A Project Paper, Available at <https://secure.csc.uvic.ca/twiki/bin/view/Courses/FinalReport>, November 2007.
- [56] E. Gamroth, “Design, implementation, and testing of an underwater global positioning system,” Master’s thesis, (In Progress).
- [57] D. A. Maltz, “Dynamic source routing protocol.” <http://www.cs.cmu.edu/~dmaltz/dsr.html>, July 2003.