Rapid Prototyping of Computer Systems: Experiences and Lessons

Daniel P. Siewiorek, Asim Smailagic, Daniel Salber*
Carnegie Mellon University
*IBM T.J. Watson Center
.

Abstract--Carnegie Mellon University has developed a User-Centered Interdisciplinary Concurrent System Design Methodology (UICSM) that takes teams of electrical engineers, mechanical engineers, computer scientists, industrial designers, and human computer interaction students that work with an end-user to generate a complete prototype system during a four-month long course. The methodology is web-supported and defines intermediary design products that document the evolution of the design. These products are posted on the web so that even remote designers and end-users can participate in the design activities. The design methodology proceeds through three phases: conceptual design, detailed design, and implementation. End-users critique the design at each phase. In addition, simulated and real application tasks provide further focus for design evaluation. The methodology has been used by the class, in designing over a dozen wearable computers, with diverse applications ranging from inspection and maintenance of heavy transportation vehicles to augmented reality in manufacturing and plant operations. The methodology includes monitoring and evaluation of the design process. This methodology is illustrated through a description of developing pervasive computing applications in collaboration with IBM during the Spring 2000 course.

1. Introduction

The Rapid Prototyping of Computer Systems (RPCS) class is a unique cross-disciplinary course, drawing students from computer science, electrical engineering, mechanical engineering, human computer interaction, and industrial design. This is a project-oriented course, which deals with all four aspects of project development: the application, the artifact, the computer-aided design environment, and the physical prototyping facilities.

The students experience a rapid design cycle [1] including:

- A complete design and manufacturing product cycle
- Application of the state-of-the-art technologies in an innovative design.
- Team effort and responsibility
- Critical and creative application of engineering fundamentals
- Interaction with widely distributed suppliers and designers
- Translation of user requirements into successful artifacts, tested by users

The Spring 2000 Rapid Prototyping of Computer Systems class was sponsored by the Pervasive Computing group at IBM T.J. Watson Research Center. Preparation started during the Fall 1999, when a half a dozen IBM researchers had weekly teleconferences with CMU to identify and organize over three dozen pervasive computing functions to provide a starting framework for the class. The pervasive computing scenarios were organized into a Distraction Matrix that formed an initial point of departure for the class. The Distraction Matrix categorizes activities by the amount of attention they require. In addition, handheld platforms were acquired and an experimental IBM Research software infrastructure was imported to CMU to serve as the basis for the software architecture. With this background, the thirty students identified two services that, based on personal experience, would facilitate the design process in project oriented classes. The first is the ability to instantaneously locate team members on campus. Students have a large number of meetings held at various times and various places. Students often are confused about where the next group meeting is to be held. The ability to observe the team's members geographical location on campus enables them to determine where the meeting is and when a member who is late to the meeting might arrive. The second service is based upon the observation that once the meeting starts, when a member arrives late the meeting stops to brief the late member. As soon as the briefing brings everyone up to the same level of knowledge, another team member might arrive necessitating yet another briefing. The service provides a shared meeting space that remote members can participate in by actively drawing or reviewing the drawings that have been created since the beginning of the meeting. The prototype developed by the RPCS class uses Hewlett Packard Jornada handheld computers communicating via Lucent Wavelan cards on Wireless Andrew.

The students developed two prototypes: Portable Help Desk (PHD) and Idealink. PHD allows a user to determine the location of other users on campus as well as information about them. It also provides other services such as notifying the user of the closest available printer or where food might be available. Idealink provides a shared meeting space and white board. Idealink allows the user to draw in a variety of colors and line. As the user draws, each line is transmitted by wireless communication to the Idealink server and synchronized with all other devices in the same communications channel.

2. Approach and Design Methodology

A User-Centered Interdisciplinary Concurrent System Design Methodology (UICSM) [2], based upon user-centered design and rapid prototyping, has been applied to the design of over a dozen computer system prototypes. Table 1 depicts the Project Matrix used in the course. The design methodology proceeds through three phases: conceptual design, detailed design, and implementation. The numbers represent concurrent activities in the various disciplines. At different times, different disciplines initiate activities with other disciplines contributing to the results. For example, the HCI group performs a field evaluation and produces the problem scenario which is reviewed and refined by the other groups (Step 1 in Table 1). Based on user interviews, and observation of their operations, baseline scenarios are created for the current procedures. Visionary scenarios identify opportunities for technology injection. User feedback on scenarios and storyboards become input to the conceptual design phase. Designers alternate between the abstract and the concrete; preliminary sketches are evaluated, new ideas emerge, and more precise drawings are generated. This iterative process continues with soft mock-ups, appearance sketches, computer and machine shop prototypes, until finally the product is fabricated. As a result of UICSM, we have achieved a four month design cycle for each new generation of wearable computers. The cycle time of the new products is ideally suited to the academic semester. Student designers initially visit the user site for a walkthrough of the intended application. A second visit after a month of design, ending the conceptual phase, elicits responses to story boards of the use of the artifact and the information content on the computer screen. After the second month a software mock-up of the system running on a previous generation wearable computer is evaluated in the end-user's application, representing the results of the detail design phase. During the third month, implementation takes place and a prototype of the system receives a further user critique. The final system is delivered after the fourth month for field trial evaluation.

3. Course Structure

The design methodology described in this paper is web-supported and defines intermediary design products that document the evolution of the design. These products are posted on the web so that even remote designers and end-users can participate in the design activities. Cross functional teams insure consistency between disciplines. Group leaders form a Project Management Team responsible for execution of the methodology. Each phase culminates in web products, a written report, and an oral presentation produced by the entire group. These activities are represented at the bottom of Table 1.

3.1 Conceptualization

<u>Problem Definition</u>. The goal of this sub phase is to define the problem, which is being solved, perform requirements analysis, and evaluate user needs. A variety of brainstorming techniques is employed to develop a product design definition including attributes such as functionality, cost, performance, technology acquisition, and fabrication techniques. The visionary scenario for the Handy Andy class is presented below.

Larry is a senior English major and he has a group project in his integrated product design class. The group decided to hold their first meeting in Porter Hall A18C at 4pm, but it is 4:15 and no one from his group has shown up yet. Larry pushes the button on the top of his hand held device to activate Waldo, his audio information retrieval system. "Launch my Portable Help Desk, please" requests Larry. Waldo launches the Portable Help Desk, which will help to locate his group members. "Where are the people in my group?" he asks. Waldo tells the Portable Help Desk to pinpoint the location of all the members in his group. The Portable Help Desk looks up the location of every person in his group, and sends that information back to Waldo. Waldo responds, "Larry, your group members are in Hamburg Hall, second floor. You might be in the wrong place." Since Larry is already 15 minutes late, he wants to find out what he has missed. "Waldo," he says, "launch my Idealink." Waldo launches Idealink on Larry's hand held device, and he enters his group's whiteboard discussion. On his way to the meeting, Larry reviews what has happened in the past fifteen minutes, and sends a quick text message to through Idealink to let them know where he is. Larry arrives at his

Table 1: Product Matrix

Project Matrix			
	Product Development Phases		
Discipline	Conceptual Design	Detailed Design	Implementation
Hardware (HW)	1 Review field data/refine HCI Problem scenario 2 Select/refine HCI Target technologies 3 Review/refinel HCI Visionary scenario 4 HW Product/feature matrices 4 HW Feasibility studies 5 Input to PM Design Decision tracking form (HW selection criteria and choices) 6 HW architecture	9 Input to HCI Design scenario 10 Add to Task / Issue Tracking form 10 Add Resolutions to Task/Issue Tracking form 11 Status list HW tasks and issues 12 Provide input to schedule 13 Resolve issues 13 Perform unit HW implementation 14 HW design phase summary 14 User evaluation and feedback plan	17 Updates to status list of HW tasks and issues 19 Input to HCI Demo Script 20 Input to PM Integration Tree 21 Integrate HW components 22 Dry run of demo and testing
Software (SW)	1 Review field data/refine HCI Problem scenario 2 Select/refine HCI Target technologies 3 Review/refine HCI Visionary scenario 4 SW Product/feature matrices 4 SW Feasibility studies 5 Input to PM Design Decision tracking form (SW selection criteria and choices) 6 SW architecture	9 Input to HCI Design scenario 10 Add to Task / Issue Tracking form 10 Add resolutions to Task/Issue Tracking form 11 Status list SW design tasks/issues 12 Input to schedule 13 Resolve issues 13 Perform unit SW implementation 14 SW design phase summary 14 User evaluation and feedback plan	18 Updates to status list of SW tasks and issues 19 Input to HCI Demo Script 20 Input to PM Integration Tree 21 Integrate SW components 22 Dry run of demo and testing 23 Archive and document source and object
Mechanical / Industrial (MEI)	1 Review field data/refine HCI Problem scenario 2 Select/refine HCI Target technologies 3 Review/refine HCI Visionary scenario 4 MEI Product/feature matrices 4 MEI Feasibility studies 5 Input to PM Design	9 Input to HCI Design scenario 10 Add to Task / Issue Tracking form 10 Add resolutions to Task/Issue Tracking form 11 Status list MEI design tasks/issues 12 Input to schedule 13 Resolve issues	18 Updates to status list of MEI tasks and issues 19 Input to HCI Demo Script 20 Input to PM Integration Tree 21 Integrate MEI components 22 Dry run of demo and testing

Human Computer Interface (HCI)	1 Field evaluation reports and data 1 Problem scenario 2 Target technologies 3 Visionary scenario 4 HCI Feasibility studies **5 Review product/feature matrices** **5 Review feasibility studies** 5 Refined solution scenario 6 Initial user interface concepts	9 Input to HCI Design scenario 10 Add to Task / Issue Tracking form 10 Add resolutions to Task/Issue Tracking form **11 Status list of UI design tasks/issues** **12 Input to schedule** **13 Resolve issues** **13 Perform unit UI implementation** **14 Design phase summary** **14 User evaluation and feedback plan** **15 Coordinate user evaluation and prepare feedback report**	**18 Updates to status list of HW tasks and issues** **19 Produce Demo Script** **20 Input to PM Integration Tree** **21 Integrate user interfaces** **22 Dry run of demo and testing** **23 Archive and document source and object**
Cross- Functional Groups	1 Review field data/refine HCI Problem scenario 2 Review/refine HCI Target technologies 3 Review/refine HCI Visionary scenario 4 Product/feature matrices 4 Feasibility studies **5 Input to PM Design Decision tracking form (selection criteria and choices)** **6 High level design (2/9)**	9 Subsystem interface specifications 10 Add to Task / Issue Tracking form 10 Add resolutions to Task/Issue Tracking form **11 Status list subsystem tasks/issues** **12 Input to schedule** **13 Resolve issues** **13 Perform unit subsystem implementation** **14 Design phase summary** **14 User evaluation and feedback plan**	**18 Updates to status list of subsystem tasks and issues** **19 Input to HCI Demo Script** **20 Input to PM Integration Tree** **21 Integrate subsystem** **22 Dry run of demo and testing**
Project Management (PM) **Group Leaders**	**Bi-weekly update of form data** **Develop work breakdown/schedule** **Phase 1 Task dependency graph**	**Bi-weekly update of form data** **Update work breakdown/schedule** **Phase 2 Task dependency graph**	**Bi-weekly update of form data** **Update work breakdown/schedule** **Phase 3 Task dependency graph** **20 Produce integration tree.**
Team Products Link to group membership and meeting pages	2 Requirement Table 6 Requirement - Feature table 7 Product design specification **8 Presentation slides (2/18)** **8 Conceptual Design**	15 Product design specification **16 Presentation slides** **16 Detailed Design Phase Report**	21 Product design specification **22 Presentation Slides** **23 Final Report**

group's meeting in Hamburg Hall and shares an Oreo Smoothie with Tom.

Technology Survey. The final shape of a system is often determined by what technology is currently available. A survey of available technology, with special emphasis on input and output devices, further refines the Product Design Specification. Lessons learned from prior generations of mobile computers are discussed. New components such as spread spectrum radio and VGA displays are acquired and interfaced to existing systems to determine the feasibility and complexity of the new technology. Videotapes of current practice as well as discussions with end users generate interactive scenarios. Individual disciplines are responsible for generating technology specific product/feature matrices (Step 4 in Table 1) for the target technologies identified (Step 2) to support the visionary scenario (Step 3). Table 2 is an example product feature matrix of handheld computers, identifying features that were studied and evaluated in Step 4.

System Architecture Specification. Given the constraints of available technology and the user's computational environment, the architecture for the system is developed. Topics such as local versus distributed processing, position sensing, computer/human interface, and information updating must be addressed by the selected architecture. Planning also includes interdependencies between the technologies, people, and resources available in the course. For example, in the Handy Andy Architecture, five layers were used, Devices, Device Proxies, User Proxies, Services and Database. (Figure 1) All services and user proxies are granted access to the database based on the privileges of the user authenticated to them. This prevents common data such as user name, address and contact information to be duplicated across systems. Updating this information can be done with a single application. The Conceptual Design Phase concludes with discipline specific architecture definition (Step 5).

Subsystem Specification. The system functionality is partitioned and assigned to hardware and/or software components. Refinement of the Product Design Specification includes attributes of the subsystems, such as performance, interface, and evaluation criteria.

3.2 Detailed Design

Each subsystem is designed and implemented. Design is supported by contemporary computer-aided design tools. Mini workshops as necessary on relevant tools provide students a basic introduction to the state of computer-aided design. Human computer interaction studies are designed in conjunction with mechanical/electronic/software mock-ups to provide data for design decisions. IBM researchers provided workshops on Pervasive System Architecture [3]. They visited classes bi-weekly and worked alongside the students at critical points in the project.

3.3 Implementation

The detailed hardware/software designs are implemented using both on-campus and off-campus facilities. On-campus physical prototyping facilities are used when appropriate. The state-of-the-art in rapid prototyping is presented.

System Integration. The various hardware and software subsystems must be individually tested and then integrated into a working system. System integration and testing plans are formulated commencing with the system architecture specification phase. The system is evaluated through controlled user experiments.

The final prototype developed in the Spring 2000 Class was called the Portable Help Desk, or PHD, which provides quick information retrieval. This tool allows a mobile user to build maps of their immediate area, including static and dynamic resources and the location of their colleagues, contact information and resources availability. While tracking a colleague, their contact information is displayed. Printer queues, restaurant hours and stock of carbonated beverages and food in connected vending machines can be displayed. The PHD application is a spatially aware system. Figure 2 illustrates the visual user interface for

Company	*Product*	*Processor*	*OS*	*Handheld Andrew*	*Price*
HP	Jornada 680	SH-3	Win CE	Yes	$800
Compaq	Itsy	StrongARM SA-1100	Linux	Yes	$1000
IBM	Workpad z50	MIPS	Win CE	Yes	$1000
Palm	Palm VII	DragonBall	Palm OS	No	$500

Table 2: Product Feature Matrix for Handheld Computers

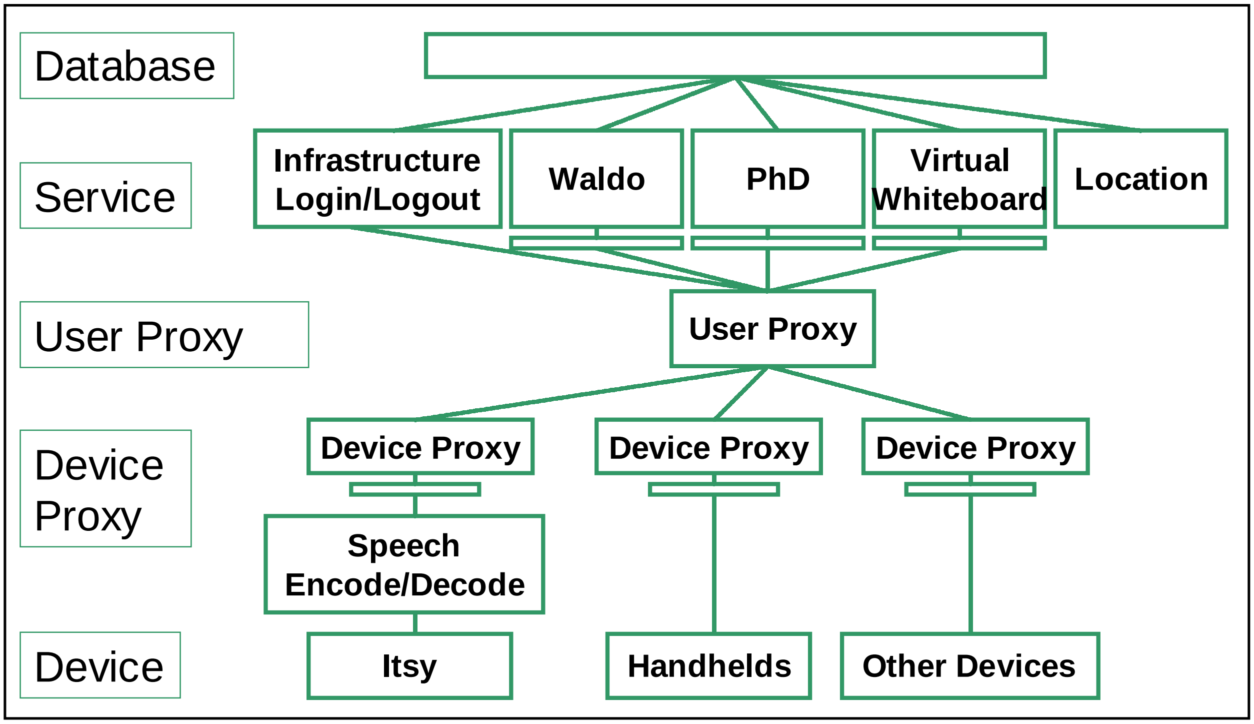

Figure 1: Handy Andy Architecture

the PHD application. People and resources are selected in the left pane, the results of the queries are presented in the middle pane while locations of people and resources are displayed in the right pane.

Methodology Evaluation. As a final phase, the methodology followed in the course is quantitatively and qualitatively evaluated and modifications suggested.

During the class, the students recorded the time spent on every aspect of the project. The class organized into the following groups, Infrastructure, Portable Help Desk (PHD), Audio Centric Help Desk (Waldo), Information Exchange (Idealink), Database and Location Service (Stalker). Idealink provides a shared whiteboard to support group meetings and collaborative efforts. Waldo supports an audio interface for retrieving information about other users and resources. Figure 3 is a chart of time spent on individual activities over the entire course of the project. The graph shows a result validating the student's hypothesis, that the time spent in meetings exceeds all other project activities. For example, 65 more hours were spent in meetings than coding activates. Figure 4 depicts the total amount of time spent per phase, indicating that the Integration Phase was the longest phase.

4. Evaluation

Table 3 compares three example computer systems that have been designed and built by the RPCS class. VuMan 3 is one example of a hardware intensive design, which included a custom printed circuit board. The Handy Andy

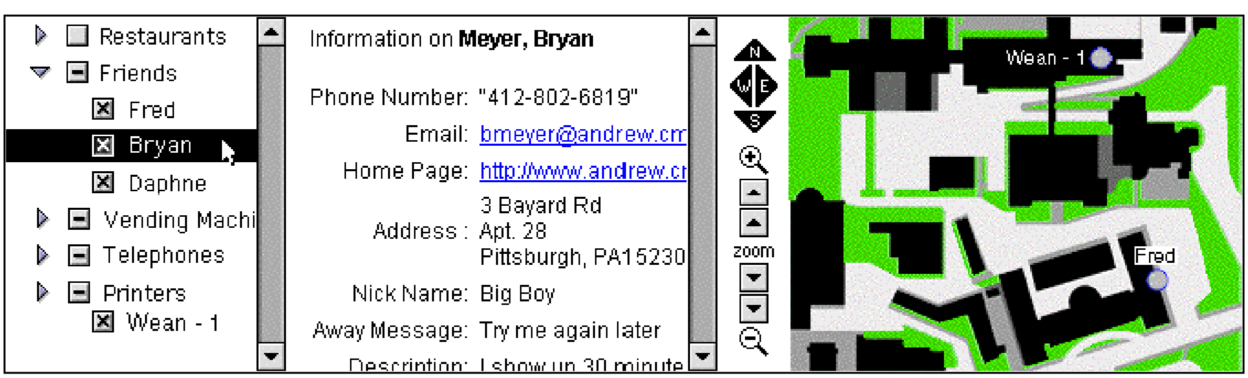

Figure 2: Portable Help Desk Screen Shot

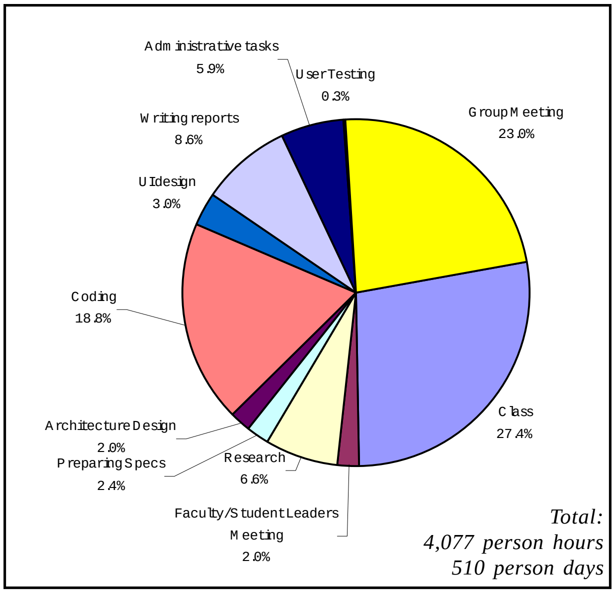

Figure 3: Time by Task for all Three Phases

pervasive computing system represents a software intensive design. MoCCA (Mobile Computing Communication Architecture) was a design by composition, including a number of off-the-shelf components. VuMan 3 and Handy Andy required significantly more person hours than MoCCA, since more intensive work was required for the hardware or software designs. As Handy Andy included a collection of system applications, the implementation phase required more time for integration than the first two phases. In the other two wearable computer designs, VuMan 3 and MoCCA, the detailed design phase consumed most of the time, as that was required by the complexity of the whole system design.

5. Conclusions

In this paper, we have described a User-Centered Interdisciplinary Concurrent Design Methodology, as applied to design and implementation of over a dozen novel generations of wearable computers at CMU. The methodology is web - supported, defines three phases of a design and implementation cycle, and documentation of the design evolution. Timelogs were collected by group and by activity and measured results presented. The time spent in the phases of a project was compared for three example systems built by the class.

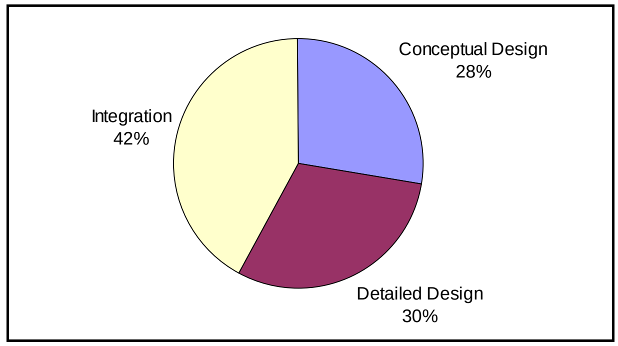

Figure 4: Total Time per Phase

Another group in the Spring 2001 class is studying how to improve the efficiency of time spent in meetings, using the Idealink and PHD applications.

6. Acknowledgment

We would like to acknowledge the support provided by IBM Research, Hewlett Packard, Lucent, the National Science Foundation, and the Defense Advanced Research Projects Agency.

7. References

[1] Siewiorek, D.P., Smailagic, A., Lee, J.C. Tabatabai, A.R.A., “An Interdisciplinary Concurrent Design Methodology as Applied to the Navigator Wearable Computer System”, Journal of Computer and Software Engineering, Vol.2, No.3, pp.259-292, 1994.

[2] Smailagic, A., Siewiorek, D.P., “User-Centered Interdisciplinary Design of Wearable Computers”, ACM Mobile Computing and Communications Review, Vol.3, No.3, pp43-52, 1999.

[3] Weber, S. Jennings, J., Beck, J., IBM Pervasive Architecture, IBM T.J. Watson Document, 2000.

Artifact	Conceptual Design (%)	Detailed Design (%)	Implementation (%)	Total Effort (Person Days)
VuMan 3	19	48	33	690
MoCCA	25	39	36	305
Handy Andy	28	38	34	510

Table 3: Person-Effort per Phase for Three Example Systems