EVALUATING USABILITY-SUPPORTING ARCHITECTURE PATTERNS:

REACTIONS FROM USABILITY PROFESSIONALS

Edgardo Luzcando

Submitted to the faculty of the School of Informatics
in partial fulfillment of the requirements
for the degree of
Master of Science in Human-Computer Interaction,
Indiana University

December 2008

Accepted by the Faculty of Indiana University,

in partial fulfillment of the requirements for the degree of Master of Science
in Human-Computer Interaction

Master's Thesis
Committee

Davide Bolchini, Ph.D., Chair

Anthony Faiola, Ph.D.

Mark Pfaff, Ph.D.

Dedicated to my wife and third party.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to thank Dr. James Fletcher who put me in contact with the HCI program at IUPUI and Dr. Antony Faiola. Several conversations with Tony resulted in me leaving a comfortable job in the East Coast to pursue my degree in Indianapolis. During the program my life changed a great deal, and I am grateful to several people that gave me their support. I would like to thank my colleagues Rick Kim and Alan Hoyt from the Midwest ISO, who facilitated a flexible work schedule while I pursued my masters. Also, my friends were there for me during difficult times and I would like to recognize their support and encouragement. In particular, a small band of HCI students that started the program with me and were my sounding board on countless occasions, and in particular to Erin Horner who gave me a reality check about using statistics. Learning the ropes of academic research was a new experience, and I received crucial guidance from Dr. Karl MacDorman at a key point of my research that helped steer me in the right direction. Similarly, a very special thank you goes to Dr. Len Bass from the SEI whose ideas were instrumental during the development of my research. In addition, right before the finish line I had the great opportunity to work with Dr. Davide Bolchini, whose enthusiasm and encouragement was a positive influence to help me complete my thesis. Also, thanks to Dr. Mark Pfaff who encouraged me to further examine and learn from my data. Finally, I would not have been able to carry out my research without the voluntary participation of colleagues and peers. A special thanks to the Indianapolis UPA and the Swiss UPA for their support. Lastly, a special thanks to my parents Jim and Conny who always give me their support.

ABSTRACT

Edgardo Luzcando

EVALUATING USABILITY-SUPPORTING ARCHITECTURE PATTERNS:

REACTIONS FROM USABILITY PROFESSIONALS

Usability professionals and software engineers typically approach software design differently; driven by a similar goal to create usable software products yet advocating distinct design methodologies. This distinction often creates a communication gap that hinders effective usability design discussions. A potential way to bridge this gap is to leverage Usability-Supporting Architecture Patterns. Recent studies suggest that they enable software engineers to include usability considerations in the architecture of software systems. A better understanding of what the usability community thinks about these patterns can lead to their improvement as well as increased adoption by software engineers, hence more effectively integrating usability into software design. The purpose of this study was to evaluate how usability professionals react to these patterns via an online pretest-posttest control group design experiment, where participants answered questions about the patterns based on prior software design experience. Participants verified there is a communication gap with software engineers, and that Usability-Supporting Architecture Patterns are perceived as useful to account for usability in software architectures. Results suggest that participants recognize the patterns' usability benefits as important and that the presentation of these usability benefits could be improved by using language more familiar to usability professionals.

CHAPTER ONE: INTRODUCTION AND BACKGROUND

Introduction to Subject

Software popularity is usually determined by the success of two attributes: software capability and software usability. Regardless of the technical capabilities a software product can offer, it is the user's reaction to what is happening and what is possible at the Graphical User Interface (GUI) that typically determines the success of a software product. Software usability is as essential to the success of a software product as the traditionally more important software capability, and how to integrate these attributes into the user's everyday life determines success (Beyer & Holtzblatt, 1998). Therefore, although software capability and software usability can be managed separately during software design, it is vital that they be managed jointly to create successful software; their synergy is likely to result in software qualities that far outweigh their individual consideration.

Capability-focused and usability-focused software design methodologies are grounded in two different schools of thought that most accept today as formal disciplines: Software Engineering (SE) and Human-Computer Interaction (HCI). Over time both disciplines have evolved independently to create differing methodologies based on their own theoretical foundation (Preece, Rogers, & Sharp, 2002); nevertheless there remains an interconnection between SE and HCI given their shared objective of designing useful software for end-users (Seffah, Gulliksen, & Desmarais, 2005). However, this implicit relationship does not guarantee effective communication and cooperation between software engineers and usability professionals during software design.

In addition to methodology differences, there are intrinsic challenges that multidisciplinary teams must overcome because each domain expert brings their own ideas and experience when it is time to collaborate in software design (Preece et al., 2002). Seffah et al. (2005) explained that there are several reasons for the collaboration challenges between usability professionals and software engineers, and further elaborate that this is a difficult problem to overcome in software design. Therefore, a different school of thought is attempting to merge SE and HCI concepts to create new approaches in software design that facilitate communication between software engineers and usability professionals (Karat, 1991; Seffah et al., 2005).

One of the new approaches is to leverage Usability-Supporting Architecture Patterns (USAPs) in software design to communicate usability concerns between software engineers and usability professionals. USAPs were developed in cooperation between the Carnegie Mellon University HCI Institute and the Software Engineering Institute (SEI) (John, Bass, Sanchez-Segura, & Adams, 2004). Although this approach has not gained wide acceptance, it may hold the key to an effective way of designing software where usability professionals and software engineers effectively communicate usability concerns, ultimately benefiting the most important stakeholder: the end-user.

Importance of Subject

Software Development and its related services is a multi-billion dollar industry, which often brings wealth to nations, or individuals (Osterweil, 2007). However, even though billions of dollars and significant time are devoted to software development projects, many of them fail, resulting in wasted time and effort (*Making IT Better:*

Expanding Information Technology Research to Meet Society's Needs, 2000). Although it is not clear how many of these projects fail because of usability issues, it is likely that usability was a factor in some of the failed projects since user related problems have been shown to represent up to 80% of maintenance costs of software products (Seffah et al., 2005). Therefore, even if projects do not fail and manage to launch a product, they often receive negative feedback about their usability. As a result, the SE-dominant Software Development industry has seen HCI methodologies gain popularity in software design to better account for usability, hence shifting the focus from traditional software-centric (capability-focused) to newer user-centric (usability-focused) software design (Beyer & Holtzblatt, 1998).

Beyer & Holtzblatt (1998) also explain that while leveraging HCI principles or including the participation of usability professionals in software design often results in improved usability, it does not guarantee the success of software projects. For example, Seife (2000) exposes the dangers of allowing a user to incorrectly enter the number zero in a user interface and how it resulted in an unexpected system failure of the USS Yorktown. Commander John Singley of the Atlantic Fleet Surface Force explained how the warship was “dead in the water” after an engineer entered the number zero in an interface which caused the engine controlling software to attempt a division by zero, subsequently causing the system to fail ("Sunk by Windows NT," 1998). Interestingly, Singley said that human factors were considered in the design of the system and one of the reasons why a windows-based interface (considered friendlier for users) was selected instead of a UNIX-based interface.

Creating a friendlier user interface is generally left to usability professionals, yet empirical evidence that support that "effects" and "acceptance" of proposed HCI-based techniques is largely missing (Seffah et al., 2005). Consequently, certain software engineers attempt to leverage HCI principles without including usability professionals or simply do not consider usability a relevant concern in software design. Preece et al. (2002) report examples of communication difficulties between people of different backgrounds such as a computer scientist and a psychologist, indicating that their understanding when working in multidisciplinary teams is different. She explains that confusion, misunderstanding, and communication breakdowns can often surface in such teams. However, she does emphasize the value of usability best practices in interaction design. She provides the example of WetPC (and underwater computer for undersea divers) which, after several failed prototypes, experienced success only after its keyboard was redesigned by a usability expert to include only five keys, noting that usability problems and not the engineering challenges were the main difficulty.

Despite the existing challenges of leveraging usability best practices, it is generally recognized that software design should account for usability in better ways than it has in the past. However, the collaboration of software engineers with usability professionals is not optimal. Snyder (2003) points out existing communication challenges stating that software engineers have a myriad of high-tech jargon that is difficult for others to understand, further recommending that users (and usability professionals) be shielded from this high-tech jargon. However, how software design activities should evolve to better communicate usability concerns in software design is precisely the very challenge that needs to be addressed given the differences between software engineers and usability professionals.

Traditionally, software engineers think of usability as a late software design activity in which usability professionals decorate software capability. In contrast, usability professionals think their early involvement in software design activities is critical to the usability of software capability. There is, at the very core, a distinct difference between software engineers and usability professionals that impacts how they work. Therefore, instead of selecting a singular SE-based or HCI-based approach, the application of a blended HCI-SE approach such as USAPs in software design might be an effective way to create synergy between the two groups, potentially resulting in successful software products and ultimately a better user experience.

CHAPTER TWO: LITERATURE REVIEW

Software Engineering and HCI in Software Design

Throughout the evolution of software design, only recently have users come to play a more vital role (Proctor & Zandt, 1994). When software design started in the mid 1900s it was primarily focused on achieving specific software capability (capability-focused software design) whereas software usability (usability-focused software design) was not a primary concern (Beyer & Holtzblatt, 1998). Since then, technology has significantly advanced, creating new opportunities for software applications in end-user products (Preece et al., 2002). As a consequence, software-based products began to increase their presence in our everyday lives (Nielsen, 2000). This progress has not slowed and mainstream users continue to see increased use of software in everyday products, making software usability a primary concern alongside software capability.

Following the rapid increase of software incorporation into end-user products, software design methodologies began to recognize the importance of usability in software-based products, and evolved accordingly to better include usability during design (Beyer & Holtzblatt, 1998). The incorporation of additional usability concerns atop traditional capability concerns created the need to develop software that is more flexible. However, this added flexibility often makes software increasingly complex as a result (Brinck, Gergle, & Wood, 2002; Curtis, Krasner, & Iscoe, 1988). Alas, while increased software design maturity and flexibility continue to advance opportunities to leverage software in ways never before possible, user acceptance of the improved software capability continues to be strongly dependent upon software usability; thus, making the user experience a significant factor in user adoption.

As is typical with the adoption of new technology, user adoption of new software that offers new capability is often slow. It is common for users to inherently resist change until they are comfortable with a specific technology, regardless of the benefits that such technology advances can provide. This behavior was eloquently described by James Kinsley; former executive of America Online (AOL) during a speech at the School of Business at George Washington University in 1999, where he pointed out that the biggest challenge for broadband adoption for AOL at that time was user acceptance, not the implementation of broadband technology. He elaborated on his vision stating that in years to come most users would have broadband, but that adoption would be slow and take many years because users would be unwilling to easily give up their trusty[1] dial-up connections.

However, as users feel more comfortable with technology and its applications, they often become more involved in software design. Furthermore, as users become more technology savvy, the opportunities created by advances in technology and software design are eventually developing into user demands (*Making IT Better: Expanding Information Technology Research to Meet Society's Needs*, 2000). Today it is more common to see users seeking specific software capability and directly requesting specific functionality for products, which strongly encourages software designers from both the SE and HCI disciplines to provide them.

This paradigm shift where the end-user is now at the helm directly requesting

[1] The word trusty was used with irony to imply that users felt dial-up (analog technology) was more reliable than broadband (digital technology) when in reality broadband technology offers better reliability.

both software capability and usability from software designers increased the importance of software design methodologies with a user-centric focus (Beyer & Holtzblatt, 1998). This need for user-centric methodologies had been previously noted by Norman (1988), who explained that usability professionals are not often called upon when designing computer systems therefore resulting in systems which are difficult to use for the average user. Additionally, Mayhew (1999) had also emphasized a need for cross-functional teams during software design in order to achieve an optimal implementation that blends capability with usability. As a result of the increased user focus, the new methodologies that emerged to address usability focused more on the user than on software capability or the invisible software qualities which are classically at the foundation of good software design. Therefore, while the newer user-centric software design methodologies better accounted for user participation through an iterative software design approach, it created new challenges to more traditional software-centric design methodologies.

For example, software design that focuses solely on the user tends to ignore the software qualities that address the foundational principles of software design or the invisible software qualities, which user interface designers often take for granted. Software designers that utilize user-centric approaches which ignore under the cover implementation tend to be successful in low complexity projects, where existing technology can cover up flaws in the software architecture, particularly when it comes to performance, scalability, and reliability. Edwards (2008) warns that we have been successful at "covering up ill-suited infrastructure features with interface veneer, but there are limits to how far this can take us." He argues that infrastructure and interaction features need to be jointly designed, and not performed ad-hoc.

As the complexity and demands of a software system increase, the application of user-centric software design methodologies in a vacuum, without accounting for the invisible software qualities might result in functional, yet unusable software due to technical limitations. Brink et al. (2002) explain that using usability methodologies alone for small web site design is not a problem. However, when building complex web sites, properly leveraging software engineering techniques can greatly increase usability. Complex software systems tend to have more components than low complexity systems that often exist outside of what would be considered directly linked to the GUI, and are therefore not easily part of user-centric design scope. For this reason user-centric software design can result in failed attempts to create usable software because it is driven only by what users want from an interface perspective and not restricted by what is possible given existing technology constraints.

Conversely, software design that focuses solely on software fundamental qualities without accounting for its usability can result in a poor user experience, and potentially an unusable product. Brink et al. (2002) point out that software engineering methods "do not include clear linkages from usability requirements" and can result in systems built according to specification, but useless. This is the more traditional state of software design observed today where software engineers focus primarily on software capability using capability-focused software design, which usually regards software usability as an afterthought.

Focusing independently on usability or capability can result in costly software modifications during later stages of software design (*Making IT Better: Expanding Information Technology Research to Meet Society's Needs*, 2000). This is often true regardless of whether usability is bolted on toward the end of the software design process or if capability is architected at the end without well established technology constraints. , Edwards (2008) advocates the importance of designing usability and capability simultaneously and points out one of the main reasons why this is difficult in practice. Reflecting on a paper by Greenberg and Marwood (1994) , Edwards describes the interconnection between capability (as infrastructure) and usability stating that one of the reasons for the disconnect between them is the communication gap between software engineers and usability professionals. He says that a "disconnect exists because neither community sufficiently understands the other's domain, nor do they have good mechanisms for passing requirements up and down the stack to each other." Therefore, striking a balance between methodologies that concentrate either on the user experience or software invisible qualities is often a difficult endeavor that, if achieved, may not be easy to repeat or reproduce in subsequent projects.

While SE methodologies are driven by the field of Computer Science, HCI methodologies are driven by a combination of fields concerned mostly with cognition and human behavior; thus, the methodologies are fundamentally different. However, both SE and HCI methodologies are commonly used in software design and even though it is difficult to conclude which are better, SE-based methodologies currently dominate the software industry. These differences in methodologies often result in different mental models for software engineers and usability professionals when it comes to software design. Based on Norman's classic three aspects of a mental model (1988) one could convey the current communication challenge between usability professional and software engineers with a modified version of the model, shown in Figure 1. This modified version of Norman's model shows the communication gap between usability professionals and software engineers based on their different design models.

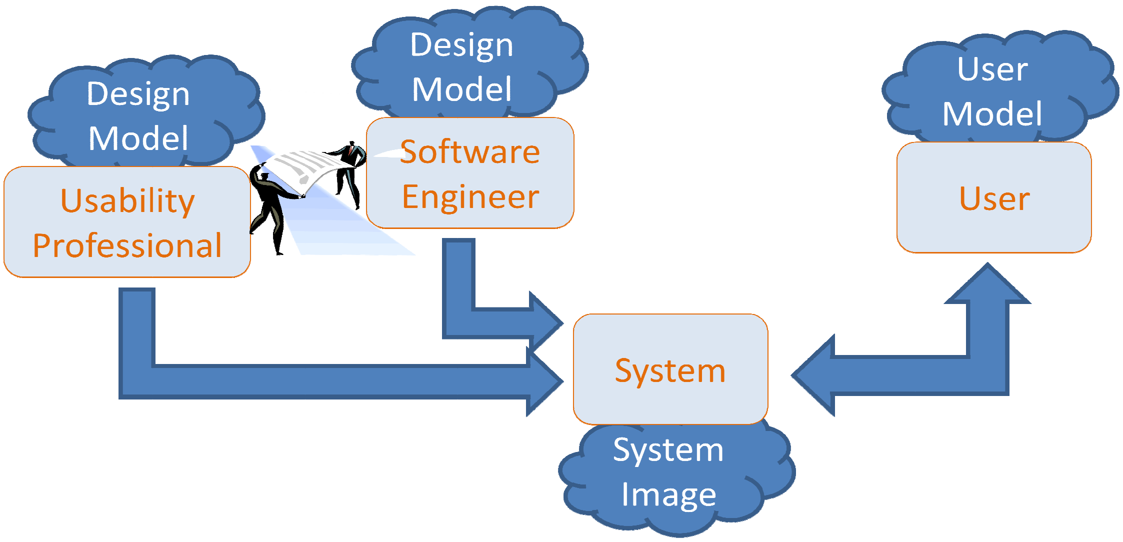


Figure 1 - Mental Model Gap between Usability Professionals and Software Engineers

Instead of discriminating between SE or HCI models, some believe that merging the SE and HCI disciplines to create blended methodologies that infuse HCI into SE might be more appropriate in software design (Karat, 2005). USAPs are an example of infusing HCI principles into SE with the vision that usability be recognized formally in the SE field and considered in conjunction with the software architecture of a system.

Usability-Supporting Architecture Patterns

The development of USAPs began with work from Len Bass and Bonnie E. John when they established that software architecture-level decisions "can have far-reaching effects on the qualities of a computer system" (Bass & John, 2000, p. 171). They suggested that there is a link between architecture styles and the eventual manifestation of such system qualities; hence they began defining a collection of Attribute-Based

Architecture Styles (ABAS) that pertained to usability quality attributes, and their initial work eventually progressed into what are now called USAPs. Similarly to how traditional software architecture patterns help achieve desired qualities, USAPs attempt to do the same for usability.

USAPs were first tested with software engineers when Golden conducted an experiment to evaluate how one USAP (Cancelling a Command) could assist software engineers in the redesign of an existing architecture to include command cancellation capabilities (Golden, John, & Bass, 2005). Golden's experiment suggests that utilizing USAPs in software architecture redesign promotes the creation of software that better supports usability concerns, hence adding value from a Software Engineering perspective. However, the experiment only included software engineers and not usability professionals.

The relevance of USAPs to usability professionals was first assessed as part of the MERBoard[2] National Aeronautics Space Administration (NASA) project by leveraging USAPs in its software design to create a wall-sized interactive system (Adams, Bass, & John, 2005). The effectiveness of USAPs in a software project was additionally evaluated by Bonnie John (2005) using evidence maps, by asking direct questions to determine whether or not USAPs were relevant and influenced project decisions. John's evidence map work suggests that USAPs were useful for usability of relevant features in the project, and the MERBoard field study suggests that USAPs are relevant to the software architecture of a system as well as its usability concerns. Additionally, the MERBoard field study found that the application of USAPs facilitated the engagement of

[2] Project name for a wall-sized system to allow Mars Rover science teams to collaborate

usability professionals, who were otherwise silent during the first conventional architecture presentation from software engineers.

In contrast to Golden's experiment, the MERBoard field study included usability professionals and recorded that the application of USAPs allowed usability professionals to get involved in the early stages of software design and collaborate directly with the software engineers. However, while the MERBoard field study validated a desired goal from USAP application in software design, it is still not clear how usability professionals from a larger usability community would react to USAPs. Based on these recent studies and suggestions from the USAPs' creators (L.B. Bass, personal communication, January 31, 2007), an opportunity exists to further explore how usability professionals react to USAPs in order to ascertain the value of USAPs to the usability community. The intention of this study is to see how usability professionals react to USAPs and potentially make recommendations of how to improve them.

Research Questions and Hypotheses

Research questions:

1. Do usability professionals think there is a communication gap between usability professionals and software engineers during software design?
2. What do usability professionals think about using USAPs to bridge this gap?
    - 2.1. How relevant are USAPs to usability professionals?
    - 2.2. To what degree do USAPs address usability concerns considered important by usability professionals?

Hypotheses

H.1 - Usability professionals can perceive Usability-Supporting Architecture Patterns as relevant in their everyday work.

H.2 - Usability professionals consider the usability benefits of Usability-Supporting Architecture Patterns important for their everyday work.

H.3 - If Usability-Supporting Architecture Patterns are communicated in more natural HCI terminology to usability professionals, they can better appreciate the value of Usability-Supporting Architecture Patterns in their everyday work.

CHAPTER THREE: METHODOLOGY

Participants

This study surveyed a convenience sample of usability professionals from the Indianapolis Usability Professionals Association (UPA) and the Swiss UPA. The sample included approximately 80 participants that have academic training in HCI, HCI professional experience, or both. The study did not differentiate between HCI professionals and HCI students, but it was expected that most participants would have some degree of professional experience in HCI or related fields given their involvement with the UPA.

Research Design

The study is based on a mixed-methods research design to attempt to analyze an area where little research has been conducted, leveraging both the strength of quantitative research and parametric statistics as well as qualitative research and its flexibility of design and exploration through non-parametric statistics. The study followed a Concurrent Triangulation Strategy (Creswell, 2003) where quantitative and qualitative research data are collected during a single data collection phase and the quantitative data given higher priority during the analysis.

The quantitative portion of the experiment used a Pretest-Posttest Control Group Design (Campbell & Stanley, 1963), which employs a Pretest-Posttest Control Group experiment technique. This is a classic between-subjects design where participants are randomly assigned to any of two groups during the data collection phase. Participants in the experiment group receive a treatment in the form of specific training materials and information for software design based on USAPs; while participants in the control group do not receive the treatment. A questionnaire format was used for the pretest as well as the posttest, including both quantitative and qualitative questions. Demographic information was solicited after the questionnaire, in addition to the opportunity to provide additional comments. The research design is visually represented in Figure 2.

R O_1	X	O_2	+	D	+	C
R O_3		O_4	+	D	+	C

<u>Research Artifact Description</u>
R= Random assignment
O_1, O_3= Observation point: pretest questions
X = Exposure to USAPs materials
O_2, O_4= Observation point: posttest questions
D= Demographic questions
C=Additional Comments

Figure 2 - Research Design

The research design allows primarily for the comparison of O_2 and O_4 where posttest answers from O_2 can be compared to posttest answers from O_2 to see if the treatment has an effect on participants. This follows a between-subjects design, where participants in the control and experiment group are asked the same set of questions, but only the experiment group is exposed to the treatment. How the survey design maps to the research design is visually represented in Table 1.

Table 1 - Survey Design Mapping to Research Design

Research Artifact	**Survey Section**	**Survey Question**
O_1, O_3	Usability and Software Design	Q1, Q2, Q3, Q4, Q5, Q6
X	Usability-Supporting Architecture Patterns - A Software Project Scenario - What are USAPs - A USAP Example: Canceling a Command	N/A

O_2, O_4	Reactions to Usability-Supporting Architecture Patterns	Q7, Q8, Q9, Q10, Q11, Q12
D	Education Experience	Q13, Q14 Q15, Q16
C	Additional Comments	Q17

Prior to the survey, initial conversations with usability professional peers and a pilot survey suggested that the wording of certain USAP usability benefits was unclear, therefore we wanted to find out if a newly worded version for these (unclear) USAP usability benefits would elicit a different response from participants. Therefore, in addition to the classic design comparison of O_2 and O_4, the design includes an additional approach to traditional survey methodology, where one question by itself allows for analysis of the control group[3] following a within-subjects design. Question seven (Q7) asked participants to rate the importance of USAP usability benefits in two forms: the original wording according to USAP materials, and a newly worded second version of the USAP usability benefit. This way, the importance rating of the original wording can be compared to the importance rating of the new wording, in order to determine if the new wording –which attempted to use language more familiar to usability professionals– changes the rating of importance.

The subcomponents of question seven (Q7) are visually represented in Table 2, which outlines the original wording versus the new wording. Only two of the original USAP usability benefits were targeted for analysis since the other seven already had wording that did not arise any comments from peers and the pilot study; thus it was assumed these seven USAP usability benefits were sufficiently understood. They were

[3] The same analysis could be performed for the experiment group, but this was not the intention of the study. However, the data for the experiment group is provided in the Results section.

also reworded however, to avoid simply asking participants what would appear as the same question twice. Therefore, the new wording applied to these non-targeted usability benefits served the purpose of consistency in the survey question.

Table 2 - Design of USAP Usability Benefits Comparisons

Set	Usability Benefit Original Wording	Usability Benefit New Wording	Intended Purpose
1	1.a - Accelerates error-free portion	1.b - Increases efficiency	Study reaction
2	2.a - Reduces impact of slips	2.b - Reduce the impact of errors	Study reaction
3	3.a - Supports problem solving	3.b - Provides user help	Survey consistency
4	4.a - Facilitates learning	4.b - Helps learnability	Survey consistency
5	5.a - Prevents mistakes	5.b - Avoids mistakes	Survey consistency
6	6.a - Increases confidence and support	6.b - Assists memorability	Survey consistency
7	7.a - Accommodates mistakes	7.b - Tolerates mistakes	Survey consistency
8	8.a - Tolerates system errors	8.b - Masks system errors	Survey consistency
9	9.a - Prevents system errors	9.b - Prevents software lockup	Survey consistency

All pretest and posttest questions were designed to collect quantitative data; however four questions additionally collected qualitative data[4]. This qualitative data can be categorized as volunteered or requested as seen in Table 3. Table 3 also shows how all the survey questions map to either quantitative or qualitative data[5], while the Instrument Development section further explains how each question was constructed to fit its intended purpose.

Table 3 - Quantitative and Qualitative Data Mapping to Survey Questions

Question	Quantitative	Qualitative	Qualitative Type
Q1	Y	N	-
Q2	Y	N	-
Q3	Y	Y	Volunteered
Q4	Y	N	-
Q5	Y	N	-
Q6	Y	Y	Requested
Q7	Y	Y	Volunteered
Q8	Y	N	-
Q9	Y	N	-
Q10	Y	N	-

[4] Q17 is also treated as a qualitative data question
[5] Q13 through Q16 are not included because they pertain to demographic information

Q11	Y	N	-
Q12	Y	N	-
Q17	N	Y	Requested

Instrument Development

The survey questions were developed with guidance from the existing literature. In particular the work of Don Dillman (Dillman, 2000) which provides guidance in the creation of internet surveys, and the work from Shuman and Presser (Schuman & Presser, 1981) that extensively analyzes questions and answers in attitude surveys. Using the literature recommendations, the survey used both close-ended and open-ended questions. Table 4 shows these question types, and their mapping to survey question numbers. The two-choice alternative questions are yes or no questions while the agreement and importance questions use Likert scales.

Table 4 - Closed-Ended and Open-Ended Question Design

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q17
Closed-ended													
Two-choice alternative		x	x			x			x			x	
Agree Disagree scale	x			x	x			x		x	x		
Important Not Important scale							x						
Open-ended													
Textbox			x			x							x

Based on the work of Schuman and Presser, the developed survey utilized open-ended as well as close-ended questions to better deal with attitude-question wording challenges, which is whether or not to limit respondents with the investigator's frame of reference of the study or to use open questions that allow respondents to answer freely. In addition, the survey questions were constructed in consideration of issues such as the "don't know problem", "agree-disagree issues", and "tone of wording" in order to address known issues such as when users might not know the answer to closed questions, might feel agreeing-response bias, or can be influenced by the tone of wording.

While Schuman and Presser also elaborated on other types of issues during their research such as "middle position" or "balanced versus unbalanced questions", this study focused on those issues that were more salient during the pilot study and conversations with peers. On that note, we agreed that including the option *don't know* when using a rating of importance was consequential to not force an answer from participants. This principle was similarly applied to agreement scales, where participants were given the *no opinion* choice. While giving participants these filtering choices could reduce sample size, it was considered sufficiently relevant that we included them in our questions.

In terms of "agree-disagree issues", we used two strategies to ameliorate its effect. To deal with the specific issue of participants acquiescing with the survey, open-ended questions where introduced whenever possible to substantiate their agreement or disagreement to a closed-ended question. Also, some questions were asked more than once in similar form at different points of survey to observe if participants would provide a similar answer. Lastly, in terms of "tone of wording," questions were worded based on the Schuman and Presser premise that "the more blatant the attempt to influence a respondent, the less likely it will succeed."

In the Tailored Design Method, the survey guidelines suggested by Dillman also provided good instruction when developing the survey questions. According to Hillman, by using previously employed survey questions, one could avoid potential issues of utilizing previously untested surveys. Untested instruments could fail to measure the desired response because they could produce multiple interpretations for a question. Based on this premise and additional materials from the Tailored Design Method we developed internal guidelines for the development of the survey: Pretest Guidelines, Treatment Guidelines, Posttest Guidelines, and Conclusion Guidelines.

Pretest Guidelines:

1. The first question should be easy and help respondents feel comfortable and less intimidated by using a simple yet engaging close-ended question.
2. No demographic information will be asked up-front to avoid discomfort.
3. Use some open-ended questions to collect unbiased feedback and help participants engage in the survey.
4. Use the pretest to gage positions regarding usability and experience as a usability professional.
5. Use the pretest to determine if participants had previous knowledge of USAPs.

Treatment Guidelines:

1. Give a brief overview of the challenges in software engineering and usability
2. Explain that there are attempts to ameliorate this problem and explain how USAPs envision the resolution to this challenge.
3. Give an introduction to USAPs details with a brief example.
4. Explain some of the findings to date with USAPs experiments and field work.

Posttest Guidelines:

1. Ask for rating of importance for the original USAP usability benefits
2. Ask for rating of importance for the reworded USAP usability benefits
3. Ask the relevance of USAPs in everyday work
4. Ask about the understanding of a USAPs
5. Ask about the potential use of USAPs

Conclusion Guidelines:

1. Ask demographic information
2. Ask for any additional comments
3. Thank participants
4. Prize explanation
5. Get contact information (for prize distribution)

Survey Technical Design

The developed instrument was implemented in the form of a customized online survey made available to participants via the internet. The conceptual design and technical specifications of the online survey leveraged available university infrastructure resources as depicted in Figure 3.

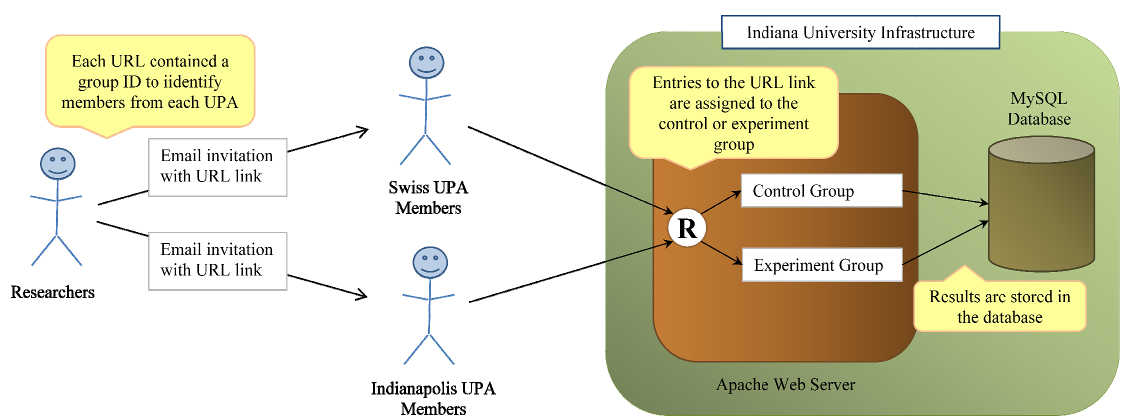


Figure 3 - Survey Technical Design

The survey was developed with a combination of HTML and PHP that ran in and Indiana University Apache web server, and all data was stored in an Indiana University MySQL database. For additional technical details refer to Appendix J. All of these components together would function similarly to currently available online survey tools such as Survey Monkey[6] and QuestionPro[7]; however the researchers developed a complete solution from scratch to achieve three goals: (a) store all the data and original survey artifacts in protected university infrastructure that is only accessible to the researchers, and not in the property of any other party external to the university; (b) randomize the entry of participants upon entering the survey URL to go to either the control group or the experiment group; and (c) make the developed survey artifacts available to other students that would like to conduct similar surveys with university resources.

While existing internet survey tools would facilitate the creation of the survey, collection of the data and presentation of the results, they did not meet the requirements to achieve the three stated goals. The flexibility of a custom solution allowed us to send an email invitation with a unique group ID to differentiate participants from the Indianapolis UPA from the Swiss UPA without the need to know the emails from the participants[8]. It also allowed the introduction of a toggling function that assigned

---

[6] http://www.surveymonkey.com/
[7] http://questionpro.com/
[8] Survey Monkey and QuestionPro allow survey administrators to track each user based on an automatically generated URL for each user, but this required the email address of each participant.

incoming participants to either the control group or the experiment group[9]. Furthermore, we envisioned that the custom-built survey would convey the academic intent of the research to the participants when they entered a web URL hosted with an Indiana University internet address.

Procedure

The custom-developed online survey's mechanics can be described in Table 5.

Table 5 - Survey Mechanics

Step	Description
1	Participants enter the study URL provided in the invitation email and are presented details about the study and its intent according to the Indiana University Institutional Review Board (IRB) guidelines.
2	Each participant is randomly assigned to either the experiment or control group when they choose to continue forward with the study.
3	From this point forward the web tool will use server-based session technology to track each unique user as they continue, recording all their answers in a back-end database.
4	All participants in the control group and experiment group receive a pretest.
5	Participants in the experiment group receive the treatment immediately after receiving the pretest, while participants in the control group skip the pretest and go directly to the posttest.
6	After all participants complete the posttest, they proceed to answer demographic information and provide additional comments.
7	After they complete the study participants are notified that their participation is finished and are given conclusion instructions, including the choice to accept a gift for their participation.

After a the introductory information required by the IRB, the survey introduction begins with a few paragraphs describing a brief history about the desire to improve usability in software products and how the study attempts to collect information about participants' experiences in this area. Participants are then given pretest questions to record their existing knowledge and experience. Following the pretest, the treatment introduces USAPs (to the experiment group only) and explains how leveraging USAPs

---

[9] In Survey Monkey and QuestionPro we were only able to find functionality that allowed to change the course of the survey (e.g. go to the control group or experiment group) based on the answer of a previous question, not on a randomizer of any kind.

could facilitate the communication between usability professionals and software engineers. The treatment provides a software design scenario describing the communication challenges regarding usability in software design and presents a USAP example. Subsequently, participants are given a posttest question that asked them to rate the importance of USAP usability benefits from an HCI perspective using a Likert scale. Additional posttest questions explore further perceptions about USAPs and software design, asking participants to state their opinions about USAPs and their potential applications in practice.

Data Analysis

The intention of our survey questions was to yield some form of classification, were a question randomly assigned to participants is the experimental factor, and hence treated as the independent variable. The responses to the questions would provide the categories of a dependent variable. An approach which follows the method of analysis used by Schuman and Presser during their research, where the resulting bivariate table is tested for statistical significance using the likelihood-ratio chi square, with probabilities evaluated as two-tailed (where a probability of less than .10 is regarded as borderline, $p < .05$ as significant, and $p < 0.01$ as highly significant).

The study however, did not yield a sample of usability professionals large enough to derive parametric statistics of statistical significance for the intended analysis. Further analysis of the data uncovered one unexpected finding where a *t* test could be used. Therefore, discrete statistics are primarily employed to describe and summarize the results of the survey.

CHAPTER FOUR: RESULTS

From the convenience sample of 80 usability professionals, a total of 67 participants began the survey. From these, 49 completed the pretest and 45 completed the posttest; which means that there were 18 participants who started the survey, but quit before completing the pretest and 22 that quit before completing the posttest. Of the 45 participants that completed the pretest and posttest only the results of 35 participants were deemed as complete data sets and usable for analysis; these 35 complete data sets are therefore summarized in this section. The 12 omitted datasets where dropped due to unforeseen technical problems with the survey, which are further elaborated upon in the Conclusion as limitations of the study. The survey results for the 35 complete data sets are presented in four subsections: Demographic Information, Pretest Results, Posttest Results for USAP Usability Benefits Ratings, and Posttest Results for Additional Questions. There were 17 participants in the experiment group and 18 in the control group.

Demographic Information

Of the 35 participants, 1 did not provide any demographic information. Of the remaining 34 participants, 20 had a masters, doctorate or post-graduate degree (59%), 12 had a bachelors degree (35%), and 2 did not have any degree (6%). Two participants listed dual degrees in HCI and Psychology and Fine Arts and HCI. All the reported degrees are grouped in Table 6. Of the 34 participants, 25 have six or more years of experience (73%) as seen in Table 7. Additionally, the reported number of usability conferences they attend a year is listed in Table 8. Of the 35 participants, 15 were from the Swiss UPA (region 1), and 20 from the Indianapolis UPA (region 2).

Table 6 - Reported Education Background

Degree Category	Number of participants with degree
Art or Fine Arts	7
Communications	6
Computer Science	6
HCI	6
Engineering	2
MBA	2
English	1
Human Factors Engineering	1
Industrial Technologies	1
Math	1
Physics	1
Psychology	1
Writing	1

Table 7 - Reported Years of Experience

Years of Experience Category	Number of Participants
1-2 years	5
3-5 years	4
6-10 years	12
11+ years	13

Table 8 - Reported Number of Conferences Attended Per Year

Number of Conferences	Number of Participants
0	14
1	19
3	1

Pretest Results

This section summarizes the six pretest questions and their results. When asked to what extent they agreed that usability is an important aspect of software design, all 35 participants agreed (100%), and when asked if they had worked in close contact with software engineers, 28 of 35 participants agreed (80%), as shown in Table 9. When asked to what extent they agreed that USAPs would assist usability professionals identify usability concerns that impact the architecture of a software system, 23 of 35 participants agreed (66%), as seen in Table 10.

Table 9 - Reported Answers for Pretest Agree-Disagree questions (N*o Opinion* Filter)

		Strongly agree	Somewhat agree	Neither agree nor disagree	Somewhat disagree	Strongly disagree	No Opinion
Q1	"Usability is a very important aspect of software design."	33	2	0	0	0	0
Q5	In your experience, did you work in very close contact with software engineers throughout the software development process?	15	13	4	2	0	1

Table 10 - Reported Answers for Pretest Agree-Disagree Questions (*Don't Know* Filter)

		Strongly agree	Somewhat agree	Neither agree nor disagree	Somewhat disagree	Strongly disagree	Don't Know
Q4	"Usability-Supporting Architecture Patterns assist usability professionals identify usability concerns that impact the architecture of a software system."	6	17	4	1	0	7

When asked if they found it challenging to apply usability principles in software design projects, 30 of 35 participants answered *yes* (86%), as shown in Table 11. When asked if there is a communication gap between usability professionals and software engineers, 33 of 35 participants answered *yes* (94%), also shown in Table 11. Additionally, participants volunteered comments about the existence of a communication gap between usability professionals and software engineers, as shown in Table 12, which summarize the identified issues and potential reasons for such a gap. When participants were asked if they were familiar with any methodologies that would improve communication between usability professionals and software engineers, 21 of 35 answered *yes* (60%) as seen in Table 11. In addition, those participants who answered *yes* where asked to list the known methodologies to substantiate their quantitative answer, and their responses are summarized in Table 13.

Table 11 - Reported Answers for Pretest Yes or No questions

		Yes	No
Q2	In your experience, have you ever found it challenging to apply usability principles in a software design project?	30	5
Q3	Do you think there is a communication gap between usability professionals and software engineers when they work together in software design?	33	2
Q6	Are you familiar with any methodologies or techniques to improve communication between usability professionals and software engineers during software projects? If so, which ones?	21	14

Table 12 - Identified Reasons for the Communication Gap between Groups[10]

Answer	Identified Issue	Number of Participants
Yes	Knowledge: software engineers only know software development and usability professionals only know usability. They don't know each other's disciplines.	5
	Core focus in project: software engineers focus on getting all system parts to work, and usability professionals only focus on system parts that impact the user interface.	7
	Mutual understanding: Both groups struggle to understand each other's needs.	4
	Awareness: software engineers have not been exposed to usability and usability professionals have not been exposed to software engineering.	2
	Process: The software design process is may or may not include usability.	1
	Availability of usability people: Not all project benefit from the participation of usability professionals.	2
	Stated there is gap, but did not elaborate on the reason.	2
No	No gap	1

Table 13 - Reported Methods to Improve Communication[11]

Listed Methods	Number of Participants
MILE+	2
Open communications (e.g. meetings, workshops)	10
AWARE	1
HCI-driven methodologies	1
Using prototypes and mockups	3
Software development methodologies	6

[10] Included five additional responses outside the 35 clean data sets because these were answers from the pretest where participants from the control group and experiment group had seen the same questions in the survey.
[11] Included three additional responses outside the 35 clean data sets because these were answers from the pretest where participants from the control group and experiment group had seen the same questions in the survey.

Conceptual Comics	1

Table 12 shows that of the 24 participants that provided qualitative data for Q3, 23 reported that there is a gap (96%). Of these 23 participants, 5 reported that knowledge or experience is the reason for the communication gap (22%), 7 that core focus in a project is the reason (30%), and 4 that mutual understanding is the reason (17%). Table 13 shows that of the participants that provided qualitative data for Q6, none mentioned USAPs. Instead, 10 participants listed common ways to communicate in project such as meetings, workshops and discussions. Three participants listed some form of prototyping technique, and 5 participants listed design techniques to improve general communication, though not specifically to improve the communication of usability concerns between usability professionals and software engineers. Six participants listed known software development techniques.

In order to better understand the control group and experiment group responses for the pretest quantitative questions, the results are additionally presented using control and experiment group categories as shown in Table 14, Table 15, and Table 16.

Table 14 - Reported Answers for Pretest Agree-Disagree Questions by Group (No Opinion Filter)

		Control						Experiment					
		Strongly agree	Somewhat agree	Neither agree nor disagree	Somewhat disagree	Strongly disagree	No Opinion	Strongly agree	Somewhat agree	Neither agree nor disagree	Somewhat disagree	Strongly disagree	No Opinion
Q1	"Usability is a very important aspect of software design."	18	0	0	0	0	0	15	2	0	0	0	0
Q5	In your experience, did you work in very close contact with software engineers throughout the software development process?	9	5	2	1	0	1	6	8	2	1	0	0

Table 15 - Reported Answers for Pretest Agree-Disagree Questions by Group (*Don't Know* Filter)

		Control						Experiment					
		Strongly agree	Somewhat agree	Neither agree nor disagree	Somewhat disagree	Strongly disagree	Don't Know	Strongly agree	Somewhat agree	Neither agree nor disagree	Somewhat disagree	Strongly disagree	Don't Know
Q4	"Usability-Supporting Architecture Patterns assist usability professionals identify usability concerns that impact the architecture of a software system."	4	11	1	0	0	2	2	6	3	1	0	5

Table 16 - Reported Answers for Pretest *Yes-No* Questions by Group

		Control		Experiment	
		Yes	No	Yes	No
Q2	In your experience, have you ever found it challenging to apply usability principles in a software design project?	14	4	16	1
Q3	Do you think there is a communication gap between usability professionals and software engineers when they work together in software design?	17	1	16	1
Q6	Are you familiar with any methodologies or techniques to improve communication between usability professionals and software engineers during software projects? If so, which ones?	9	9	12	5

Posttest Results for USAP Usability Benefits Ratings

This section presents the results of question seven (Q7), which asked participants to rate the importance of USAP usability benefits in two forms. First participants were asked to rate the importance for original wording of USAP usability benefits, and then participants were asked to rate the importance for the newly worded USAP usability benefits. These results, shown in Table 17[12], are suitable for a between-subjects comparison between the control group and experiment group, and a within-subjects comparison within the control group. The ratings used are: Very Important =1, Important=2, Somewhat Important=3, Not Important =4 and Don't Know=5.

An independent groups *t* test was used to test the difference in the mean response or rated importance of the target USAP usability benefits *Accelerates error-free portion* and *Reduces impact of slips* shown in Table 17. Respondents from region 2 ($M = 1.76$) showed a lower mean response than those from region 1 ($M = 2.29$), $t(30) = 2.09$, $p < .05$, $r = .36$ as shown in Figure 4.

[12] One answer was not given by participants in ratings 1.a, 2.a, and 4.a of Q7; therefore the total count for those ratings is 34 and not 35.

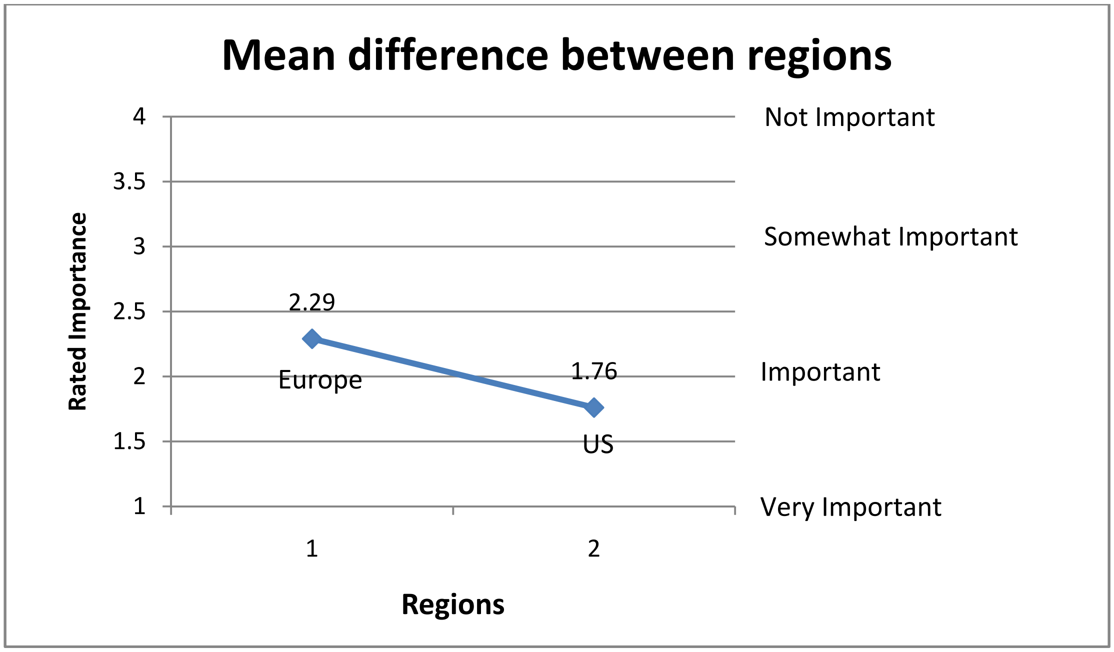


Figure 4 - Usability Benefit Importance Rating Difference between Regions

When comparing between-subjects, in the control group, 12 out of 17 participants reported the *Accelerates error-free portion* USAP benefit as important (71%), whether as 9 out of 17 participants rated it as important in the experiment group (53%); this is a reduction of 18% when comparing the control group to the experiment group. Similarly, the *Reduces impact of slips* USAP benefit had 13 out of 17 participants rate it as important (76%), compared to 9 out of 17 in the experiment group (53%), for a reduction of 23%. The *Supports problem solving* USAP benefit had 15 out of 18 participants rate it as important (83%) in the control group compared to 11 out of 17 participants (65%) in the experiment group, for a reduction of 18%. The *Facilitates learning* USAP benefit had 16 out of 18 participants rate it as important (89%), compared to 11 out of 16 in the experiment group (69%), for a reduction of 20%.

The other between-subjects comparisons of the importance rating for USAP benefits experienced less variation when contrasting the experiment group with the control group. The *Increases confidence and support* USAP benefit had 14 out of 18 participants rate it as important in the control group (78%), compared to 14 out of 17 in the experiment group (82%). The *Prevents mistakes* USAP benefit had 14 out of 18 participants rate it as important in the control group (78%), compared to 13 out of 17 participants in the experiment group (76%). The *Accommodates mistakes* USAP benefit had 12 out of 18 participants rate it as important (67%), compared to 14 out of 17 participants in the experiment group (82%). The *Tolerates system errors* USAP benefit had 10 out of 18 participants rate it as important in the control group (55%), compared to 11 out of 17 participants in the experiment group (65%). The *Prevents system errors* USAP benefit had 13 out of 18 participants rate it as important in the control group (72%), compared to 12 out of 17 participants in the experiment group (71%).

When comparing within-subjects of the control group, the newly worded USAP benefits targeted in the study exhibit a directional increase when compared to the original USAP wording. For the first set (1.a versus 1.b in Table 17) the original USAP benefit *Accelerates error-free portion* had 12 of 17 participants rate it as important (71%), compared to 16 of 18 participants for the newly worded USAP benefit *Increases Efficiency* (89%); this represents a increase of 18%. For the second set (2.a versus 2.b in Table 17) the original USAP benefit *Reduces impact of slips* had 13 of 17 participants rate it as important (76%), compared to 17 of 18 participants for the newly worded USAP benefit *Reduce the impact of errors* (94%); representing an increase of 18% .

Table 17 - Reported Posttest Ratings for USAP Usability Benefits

			Control					Experiment				
Set	Wording	USAP usability benefit	Very Important	Important	Somewhat Important	Not Important	Don't Know	Very Important	Important	Somewhat Important	Not Important	Don't Know
1.a	Original	Accelerates error-free portion	4	8	4	0	1	5	4	1	1	6
1.b	Introduced	Increases efficiency	13	3	2	0	0	7	5	4	0	1
2.a	Original	Reduces impact of slips	4	9	4	0	0	3	6	3	1	4
2.b	Introduced	Reduce the impact of errors	7	10	1	0	0	8	7	1	0	1
3.a	Original	Supports problem solving	7	8	3	0	0	5	6	5	0	1
3.b	Introduced	Provides user help	6	7	5	0	0	5	7	1	3	1
4.a	Original	Facilitates learning	4	12	1	1	0	2	9	3	1	1
4.b	Introduced	Helps learnability	8	9	1	0	0	6	6	3	1	1
5.a	Original	Prevents mistakes	8	6	3	1	0	7	6	2	1	1
5.b	Introduced	Avoids mistakes	7	8	2	1	0	4	8	4	0	1
6.a	Original	Increases confidence and support	5	9	4	0	0	11	3	1	1	1
6.b	Introduced	Assists memorability	7	9	2	0	0	5	8	2	1	1
7.a	Original	Accommodates mistakes	5	7	4	1	1	8	6	2	0	1
7.b	Introduced	Tolerates mistakes	6	7	4	0	1	8	7	1	0	1
8.a	Original	Tolerates system errors	6	4	6	1	1	6	5	4	0	2
8.b	Introduced	Masks system errors	2	7	5	2	2	4	7	3	2	1
9.a	Original	Prevents system errors	8	5	4	1	0	6	6	2	1	2
9.b	Introduced	Prevents software lockup	10	2	6	0	0	7	3	4	2	1

The total weighted average of the original USAP usability benefits importance rating across the control and experiment group are displayed in Table 18 and Figure 5. In Table 18 the weighted average difference between the control group and experiment group for the targeted USAP usability benefits exhibited the largest decrease: 3.25 for *Accelerates error-free portion*, and 3.5 for *Reduces impact of slips*. Of the non-targeted USAP usability benefits, only one experienced a similar decrease: 3.25 for *Facilitates learning*. Figure 5 shows that the two targeted USAP usability benefits fall in the *Somewhat Important* category and the other seven in the *Important* category.

Table 18 - Weighted Average Comparison for USAP Usability Ratings

#	USAP usability benefit	Control					Experiment					Total
		Very Important (4)	Important (3)	Somewhat Important (2)	Not Important (1)	Weight Average	Very Important (4)	Important (3)	Somewhat Important (2)	Not Important (1)	Weight Average	Weight Average
1	Accelerates error-free portion	16	24	8	0	12.00	20	12	2	1	8.75	10.38
2	Reduces impact of slips	16	27	8	0	12.75	12	18	6	1	9.25	11.00
3	Supports problem solving	28	24	6	0	14.50	20	18	10	0	12.00	13.25
4	Facilitates learning	16	36	2	1	13.75	8	27	6	1	10.50	12.13
5	Prevents mistakes	32	18	6	1	14.25	28	18	4	1	12.75	13.50
6	Increases confidence and support	20	27	8	0	13.75	44	9	2	1	14.00	13.88
7	Accommodates mistakes	20	21	8	1	12.50	32	18	4	0	13.50	13.00
8	Tolerates system errors	24	12	12	1	12.25	24	15	8	0	11.75	12.00
9	Prevents system errors	32	15	8	1	14.00	24	18	4	1	11.75	12.88

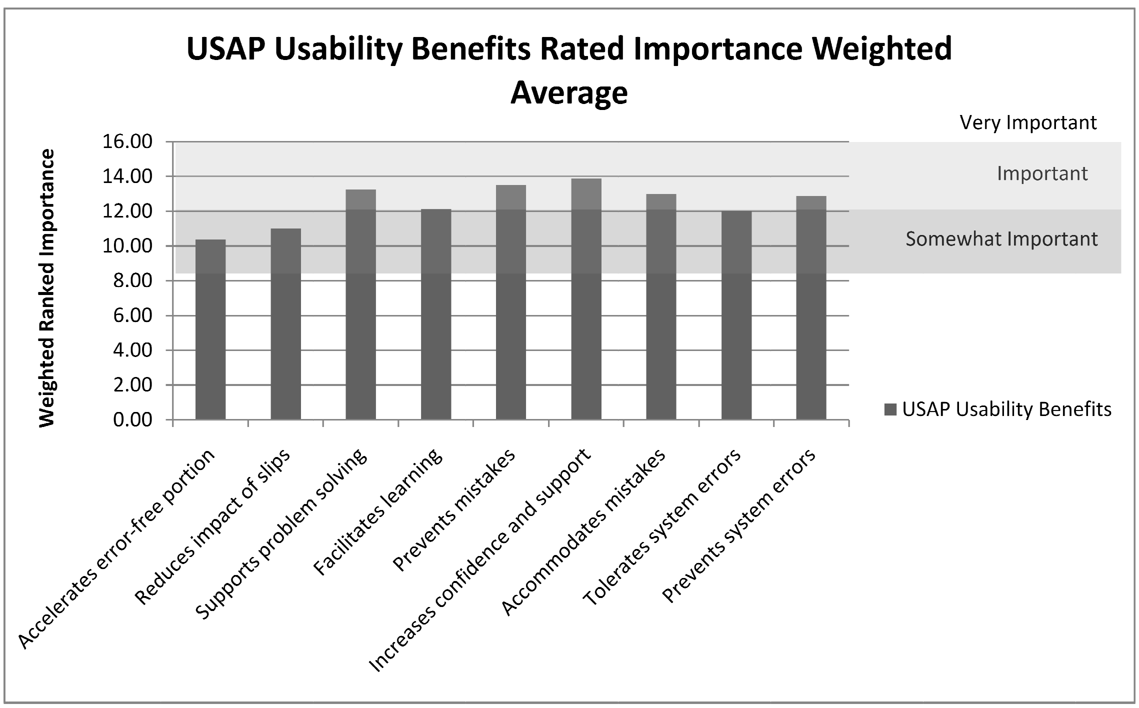


Figure 5 - Weighted Averages for Importance Rating of USAP Usability Benefits

The results for the two targeted USAP usability benefits in Table 17 are also presented another way in Table 19, which shows the weighted results for the targeted USAP benefits, for the between-subjects comparisons and the within-subjects comparison. These weighted results can also be visualized in Figure 6 for the USAP benefit *Accelerates error-free portion* (set 1), and Figure 7 for the USAP benefit *Reduces impact of slips* (set 2). In Figure 6, the *Accelerates error-free portion* benefit shows a decrease in weight from 48 to 35 for the between-subjects comparison (27%), and an increase in weight from 48 to 65 for the within-subjects comparison (26%). In Figure 7, the *Reduces impact of slips* benefit shows a decrease in weight from 51 to 37 for the between-subjects comparison (28%), and an increase in weight from 51 to 60 for the within-subjects comparison (15%).

Table 19 - Weighted Posttest Ratings for Targeted USAP Usability Benefits

			Control						Experiment					
Set	Wording	USAP usability benefit	Very Important (4)	Important (3)	Somewhat Important (2)	Not Important (1)	Don't Know (0)	Total Weight	Very Important (4)	Important (3)	Somewhat Important (2)	Not Important (1)	Don't Know (0)	Total Weight
1.a	Original	Accelerates error-free portion (control)	16	24	8	0	0	48	20	12	2	1	0	35
1.b	Introduced	Increases efficiency (control)	52	9	4	0	0	65	28	15	8	0	0	51
2.a	Original	Reduces impact of slips (control)	16	27	8	0	0	51	12	18	6	1	0	37
2.b	Introduced	Reduce the impact of errors (control)	28	30	2	0	0	60	32	21	2	0	0	55

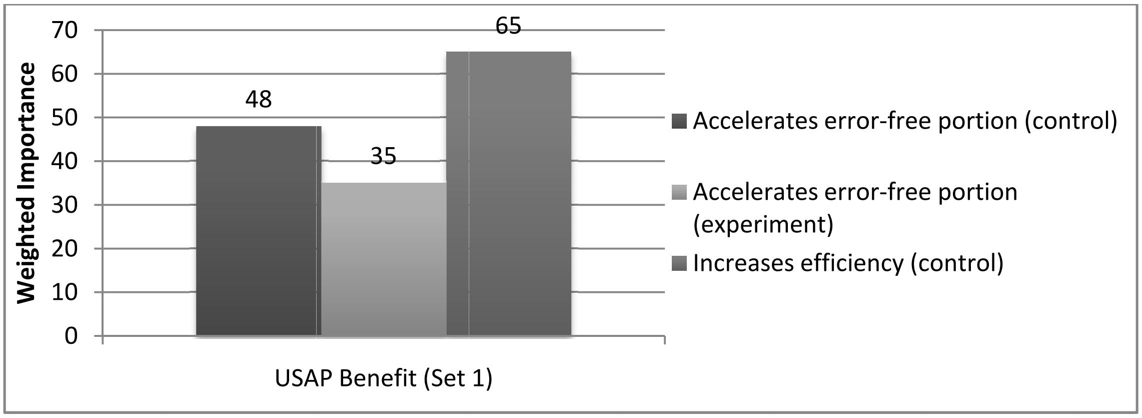


Figure 6 - ***Accelerates Error-Free Portion*** **USAP Benefit Weight Comparisons**

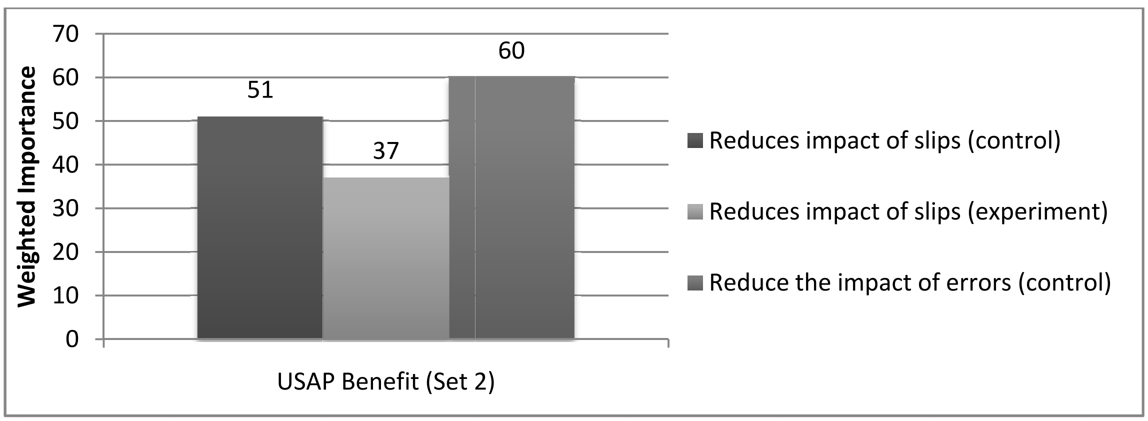


Figure 7 - ***Reduces the Impact of Slips*** **USAP Benefit Weight Comparisons**

The rating of USAP usability benefits also collected qualitative data by asking participants to provide any comments if any of the USAP usability benefits were not clear to them. Of the 14 rated USAP usability benefits, only 7 elicited responses, as summarized in Table 20. While few participants provided qualitative feedback for USAP usability benefits in Q7, the two targeted USAP usability benefits had the most feedback with eight explicit comments stating difficulty in understanding the meaning of the usability benefit.

Table 20 - USAP Usability Benefits Qualitative Responses

USAP usability benefit	Unclear Meaning	Not Frequent	Never Seen This	Context Clarification
a. Accelerates error-free portion (original)	5			
b. Reduces impact of slips (original)	2	1		
c. Supports problem solving (original)			1	
e. Prevents mistakes (original)	1			
p. Tolerates mistakes (new)				1
q. Masks system errors (new)				1
r. Prevents software lockup (new)				1

Posttest Results for Additional Questions

The posttest results of the other questions are summarized in Table 21, Table 22, and Table 23, respectively. When asked if they found that leveraging USAPs would be useful for their software design activities, 24 of 35 agreed (68%) as seen in Table 21. However, there is a difference between the control group and the experiment group. Of the 24 that agreed, 15 were from the control group (62%) and 9 were from the experiment group (38%). Additionally, the experiment group experienced an increase from 0 to 6 participants in the selection of no opinion when compared to the control group, or 37% increase for the experiment group.

Table 21 - Reported Answers for Posttest Agree-Disagree Questions

		Control						Experiment					
		Strongly agree	Somewhat agree	Neither agree nor disagree	Somewhat disagree	Strongly disagree	No Opinion	Strongly agree	Somewhat agree	Neither agree nor disagree	Somewhat disagree	Strongly disagree	No Opinion
Q8	On the basis of what you have learned in this survey about USAPs, do you think it is useful to leverage their use in software design activities?	7	8	3	0	0	0	4	5	1	0	0	6
Q10	"The work relationship with software engineers is difficult."	2	8	5	3	0	0	1	7	3	0	4	0
Q11	"There is a gap between software engineers and usability professionals when they collaborate in software design."	3	11	2	2	0	0	6	7	2	2	0	0

Table 22 - Reported Answers about the Likelihood of Further Investigating USAPs

		Control		Experiment	
		Yes	No	Yes	No
Q9	After you complete the survey, how likely are you to investigate USAPs further and learn more about them?	15	3	10	6

Table 23 - Reported Answers about Who Should Lead Software Design

		Control		Experiment	
		Usability Professionals	Software Engineers	Usability Professionals	Software Engineers
Q12	Who in your opinion should lead software design activities - usability professionals or software engineers?	13	5	7	9

When participants were asked how likely it would be for them to go and learn more about USAPs after completing the survey, 25 of 35 agreed (71%). When asked if there is a communication gap between usability professionals and software engineers, 29 of 35 participants agreed (83%).

CHAPTER FIVE: DISCUSSION

Software Design and Perception of USAPs

H.1 predicted that usability professionals expect to get benefits of Usability-Supporting Architecture Patterns in their everyday work. During the pretest 66% of participants agreed that USAPs could enable usability professionals identify usability concerns that impact the architecture of a software system. However, it is unclear why the majority agreed because no participants reported to have a priori knowledge of USAPs. When participants were asked if they were familiar with any methodologies to improve communication between usability professionals and software engineers, 60% said yes; however when asked to elaborate on which methodologies they knew about there was no mention of USAPs.

One possible explanation for this result could be that the term “usability-supporting” along with “architecture-patterns” could lead to an implicit belief that USAPs are beneficial. When participants were asked in the posttest if they found USAPs useful for software design activities based on what they had learned in the survey, 68% agreed. However, of the 68% participants that agreed, there is a directional difference between the control group and experiment group. In the control group, 62% agreed compared to only 38% in the experiment group. This decrease in agreement in the experiment group occurred alongside a 37% increase in the selection of the *no opinion* choice. While the experiment group had 6 participants make this selection, the control group had none. In addition, the decrease in agreement could mean that participants that received the experiment and were presented additional USAPs materials took the survey more seriously and considered their choices more carefully, selecting *no opinion* when they were not sure of the answer.

Another factor that could explain the decrease in agreement is the participants' experience in the field, in that more experienced participants are often more comfortable selecting a position of *no opinion*. Of the six participants that selected *no opinion*, five had 11 or more years of professional experience, and one between 6 and 10. However, the results also show that there were seven participants that also had 11 or more years of experience and selected an answer in the scale instead of *no opinion*. Therefore, the participant's experience does not appear to be a determining factor for the decrease in agreement.

The selection of the *no opinion* choice could also be an effect of receiving the treatment. It is possible that after participants received the treatment and were exposed to the USAP scenario, they did not understand its purpose or were perhaps confused by the presentation of the materials (see Appendix E). For example, it could be that the USAP scenario of cancelling a command did not easily apply to their experience, and therefore did not add clarity about the usefulness of USAPs. Conversely, it is possible that participants that did not receive the treatment and did not see the USAP materials were able to imagine (or construct) their own idea of what USAPs are, which in their view might be more effective than the actual USAPs.

Reactions to USAPs

H.2 predicted that usability professionals can perceive the importance in using Usability-Supporting Architecture Patterns for their everyday work. During the pretest, 100% of the participants acknowledged that usability is an important aspect of software design, and 86% of participants acknowledged they have previously found it challenging to apply usability principles in a software design project. This suggests that participants understood the importance of usability in software design and the challenges of applying usability principles therein. Hence, the fact that 71% of participants responded that they would likely investigate USAPs further and learn more about them is a potential indication of their usefulness.

This finding is strengthened by results from the pretest that show 94% of participants in agreement that there is a communication gap between usability professionals and software engineers. The existence of a communication gap was alternatively asked in another question of different form that shows 83% of participants in agreement that there is such a gap. However, it is possible that the perceived importance of USAPs is a result of recognizing that any technique to improve usability is innately important to usability professionals.

All of the pretest results were as expected based on our prior experience and from the available literature, which often points out the differences and challenges of usability professionals and software engineers working together. An interesting observation however, is that while 68% participants agreed that leveraging USAP's would be useful in software activities, 80% have worked in close contact with software engineers. This suggests that those who have experience in working with software engineers and dealing with usability concerns in software design are more likely to perceive the importance of USAPs.

Evaluation of USAP Usability Benefits

H.3 predicted that if Usability-Supporting Architecture Patterns are communicated in more natural HCI terminology to usability professionals, they can better appreciate the value of Usability-Supporting Architecture Patterns in their everyday work. We predicted that when participants received the treatment they would rate USAP usability benefits as more important since they had (in the treatment) been exposed to a positive introduction of USAP usability benefits and potential use in software design. However, this was not the case for the USAP usability benefits targeted in the study. The rated importance of *Accelerates error-free portion* was 18% less in the experiment group when compared to the control group, and the rated importance of *Reduces impacts of slips* was 23% less in the experiment group when compared to the control group. This reduction in rating of importance suggests that participants in the experiment group did not consider the targeted USAP usability benefits to be as important as the control group did, and were more inclined to choose the *not important* or *don't know* categories.

While the other seven USAP usability benefits were not targeted in the study, two of them had similar results to the ones targeted in the study. The rated importance of *Supports problem-solving* was 18% less in the experiment group when compared to the control group, and the rated importance of *Facilitates Learning* was 20% less in the experiment group when compared to the control group. The remaining five USAP usability benefit ratings did not exhibit this change, which suggests a similar response to the treatment (when comparing between-subjects) for the ratings: *Accelerates error-free portion, Reduces impacts of slips, Supports problem-solving,* and *Facilitates Learning*.

While these four USAP usability benefits exhibited a comparable response, there was a unique difference between the ones targeted in the study and those that were not. The ones targeted in the study also exhibited a directional difference when their ratings for the original USAP wording and the new USAP wording are compared within the control group, as we had expected; the ones not targeted in the study did not experience that same different in rating. Furthermore, none of the remaining five USAP usability benefits (not targeted in the study) exhibited this directional difference in rating of importance.

For the targeted USAP usability benefits, the decrease in rating of importance for the within-subject comparison in the control group is 18% for the original USAP usability benefit *Accelerates error-free portion*, when compared to its newly worded counterpart *Increases efficiency*. Similarly, the original USAP usability benefit *Reduces impacts of slips* exhibited an 18% reduction in rating of importance when compared to its newly worded counterpart *Reduce the impact of errors*. This suggests that the decrease in rating of importance for the between-subject comparisons of the control group and experiment group for the targeted USAP usability benefits could be related to the increase in rating of importance for the within-subject comparisons of the control group for the same USAP usability benefits. This relationship suggests that the targeted USAP usability benefits reworded with more familiar (HCI) terminology are better understood by usability professionals.

While these findings hint at a potential correlation, it should be noted that they could also be an effect of order-bias, since the nine USAP usability benefits rated were not randomized for different participants, always presenting the targeted USAP usability benefits first. This means that participants might think more about the first USAP usability benefits they rated, and pay less attention as they moved down the list of USAP usability benefits. However, this type of order-bias could on the other hand strengthen the results of the survey if participants indeed provided more thoughtful answers based on added scrutiny of the first ratings. Additional studies would need to be conducted to ascertain if there is a significant correlation between the difference found for the between-subjects results and the within-subject results for the targeted USAP usability benefits.

An unexpected yet interesting result of the experiment was that participants in region 1 (Europe) responded differently than those in region 2 (US) when rating the importance of the target USAP usability benefits *Accelerates error-free portion* and *Reduces impact of slips*. Since a rating of 1 represents the highest importance, the lower mean from US participants in contrast to a higher mean from European participants is indicative that US usability professionals consider the target USAP usability benefits more important than European usability professionals. This is a potential indication that USAPs are more difficult to understand for European usability professionals than for US usability professionals. Another interpretation is that USAP usability benefits use language more familiar to usability professionals in the US.

When responding to the open-ended question of whether or not there is a communication gap between usability professionals and software engineers, all participants except one reported that there is gap, albeit in different forms. These results, though not conclusive, help support the notion that a communication gap exists between usability professionals and software engineers. Furthermore, this helps confirm that the sample population for the survey had a certain level of awareness of such a gap, validating the quantitative results which obtained the same affirmation from 94% of participants in one question from and 83% in its alternate form.

The participants' qualitative responses to whether or not they knew any existing methodologies that aimed at improving the communication gap between usability professionals and software engineers similarly strengthened other findings in the survey since they did not report prior knowledge of USAPs when beginning the survey. A potential risk of the survey, which would weaken any findings, was that participants had been previously exposed to USAP materials and would bring any historical influence, either positive or negative, into the sample population.

From the nine USAP usability benefits shown in Table 18 and Figure 5, seven fall in the *Important* category and two in the *Somewhat Important* category. These findings suggest that USAP usability benefits are perceived as relevant to software design in the context of usability by usability professionals surveyed in this study.

CHAPTER SIX: CONCLUSION

Participants confirmed that there is a communication gap between usability professionals and software engineers when collaborating in software design projects. This result was expected and served simply to validate the framework of the study. Similarly, participants confirmed that there are challenges in the communication between usability professionals and software engineers during software design projects.

The overall reactions to USAPs are positive. However, although the initial reactions from participants to the potential benefit of USAPs is positive, it is not clear on what basis given that they did not have a priori knowledge of USAP materials. The reactions from participants after being introduced to USAP materials are also positive, though to a lesser degree when compared to the initial responses. This study could not determine the cause for this difference.

The results of the rating of importance for USAP usability benefits were not as expected. The presentation of USAP materials in a positive light within the context of software design to the experiment group did not increase importance rating of USAP usability benefits, it decreased it. In contrast, when directly asked if USAPs can be perceived as having potential benefits, participants in the experiment group reported a positive increase when compared to the control group. Additionally, the rating of importance for the USAP usability benefits *Accelerates error-free portion* and *Reduces impact of slips* suggest that the current wording is not adequate for usability professionals and can be further improved. The suggested wording *Increases efficiency* and *Reduce the impact of errors* directionally improve their acceptance.

The difference in rating of importance for the USAP usability benefits *Accelerates error-free portion* and *Reduces impact of slips* suggests that usability professionals in the US can more easily understand them than their counterparts in Europe. While this was not the primary purpose of the study, this particular finding suggests that different wording of USAP usability benefits might be better suited for European usability professionals.

Limitations

Due to the limitations of time and budget of this study, a convenience sample was selected with participants from the Usability Professional Association (UPA). Two different UPA groups, in separate geographical locations, were used to ameliorate this weakness. The number of participants was also limited by budget and time constraints, but it would be possible to repeat the study with a larger sample from other UPA groups or similar organizations. It can be argued however, that a convenience sample is acceptable because the study targets such a specific population. In order to increase participation an Amazon $10 gift card was offered as an incentive to all participants that completed the survey.

Two additional considerations about the participants are that they implicitly bring a specific bias toward usability to the experiment or that culture could have been a factor that impacted results. However, one could argue that this bias toward usability is actually desired since the study targets usability professionals. Furthermore, the differences in culture between the European and US participants could actually strengthen the results by balancing results with the inclusion of background from more than one geographical area.

While we attempted to reuse previously used survey question constructs to ensure the instrument had been pretested, the introduction of new language and slight differences could potentially result in questions that do not serve their intended purpose and mislead the participants, evoking incorrect answers. We used recommended techniques to create online surveys and formulate questions that deal with the measurement of attitude, however these recommendations came from questions in other disciplines and their transcription was merely our best attempt.

Since the qualitative data was collected using an online survey there was no personal interaction with the users in order to capture reactions and feedback typical of qualitative studies such as ethnographic observation, however using ethnography would have limited this study to a small sample given the time and budget constraints. It was considered prudent to attempt to reach a higher number of usability professionals and loose part of the qualitative responses that could have been captured with more personal interaction. Additionally, it was the intention of the survey to hide (as much as possible) from the user the fact that they were participating in a pretest-posttest experiment by making the survey seem like one continuous questionnaire.

The online survey had certain technical limitations that we had not foreseen. While the use of more popular online survey tools was available, we believed that making every question optional and keeping all technology within the university granted the survey a certain amount of confidence to participants regarding the anonymity of participation and the protection and disclosure of any information. As a consequence, we were bound to use shared university infrastructure resources, which imposed certain technical limitations by default. The session tracking mechanism required that the use of session cookies be enabled, and it was not for certain participants, which created difficulties when tracking those participants during the survey. In addition, the session expiration time was 24 minutes, so participants that began the survey and had a period of inactivity greater than 24 minutes posed a problem. In most of these cases, the pretest questions were captured correctly, but the posttest questions (and the rest of the survey) was captured but could not be mapped to the corresponding pretest answers, other than by speculation. In the interested of using clean data, these data sets were discarded. Six data sets were lost due to the fact that users did not have cookie sessions enabled in their browsers, and 6 data sets due to the fact that participants took an unusually long time to answer the survey and although the survey recorded the results from these participants there was no definitive way to map the pretest to the posttest answers for these participants. Unfortunately, these issues did not surface during the pilot survey.

The resulting 35 clean data sets is a small sample size, which imposes certain limitations regarding the use of statistics and potential generalization of the results. It was the intention of the research to capture enough participants to be able to use parametric statistics with a certain level of assurance that applying them to non-continuous data (e.g. ratings) would result in satisfactory findings and conclusions. As it is, most of the statistics leverages are discrete, augmented only by the qualitative responses or demographic information provided by participants.

Some of the open-ended questions were truncated due to a technical error in the implementation of the web survey. Any answer that exceeded 255 characters was truncated because only the initial 255 characters were being stored in the database. Unfortunately, this condition did not arise during the pilot study, and while we noticed this behavior during the data collection phase and could have fixed it, we did not want to change anything during this period, to avoid introducing new issues. Fortunately, in the cases were this condition occurred, the essence of the information was preserved. Only in a few answers was the meaning truly lost, and hence the opportunity to report it. Nevertheless, we do regret the loss of qualitative information.

Future Research

Our first suggestion for future research would be to conduct an improved version of this study with modifications to question seven. It is possible that while the question wanted to probe the clarity of the wording for the USAPs usability benefits, asking about importance was misleading, instead of asking about the clarity of the wording directly. An improved version would consist of two different questions, one that would ask about the importance, and another that would ask about the clarity of wording.

Another suggestion would be to create controlled experiment similar to Golden's experiment with software engineers where usability professionals are introduced to USAPs and asked to use them in a specific scenario and provide comments. This could be a 2-3 hour lab session with a software design scenario where one could evaluate general reactions to USAPs.

In order to generalize the findings of this research a larger sample should be utilized, but funding could be an issue. The mere fact that the target population of the research is very specific and that most candidates are practicing professionals, makes reaching them somewhat difficult. Perhaps the right setting or more enticing incentives can interested a larger sample and generate larger data sets and more conclusive results.

However, it might be implicitly difficult to determine what would be a large enough sample since the size of the target population itself is not easy to determine. Also, additional research is needed to test USAPs in practice on real-life projects, in order to obtain empirical evidence to evaluate USAPs further. We propose a setting where two different teams can set out to design a software product, one utilizing USAPs and another not using them. This would help determine whether the application of USAPs in software design has an impact on the final product, however obtaining funding to develop the same product twice might prove difficult.

Summary

Usability professionals and software engineers come from different backgrounds and have intrinsic differences when thinking of usability in software design. This difference is often the cause of a communication gap that impedes effective communication between the two groups, often affecting the usability of software products. Leveraging USAPs is a suggested way to improve communication between the groups, ultimately improving usability for the end-user. Previous studies have shown that USAPs enable software engineers to better account for usability, but there is little information about how USAPs are perceived in the usability community.

The purpose of this study was to conduct an experiment that would capture the perception from usability professionals when exposed to USAP materials in different ways, while at the same time confirming the existence of a communication gap between usability professionals and software engineers. The study found that usability professionals agree there is a communication gap between them and software engineers.

In addition, results suggest that participants perceive USAPs as useful to account for usability in software design, and that they would be willing to learn more about them. Also, results suggest that the USAPs usability benefits are perceived as important in software design; none of these findings are conclusive since the sample population of 35 participants was not large enough to derive statistically significant results. However, there was an unexpected finding that they data could support indicating a different reaction to the importance of USAP usability benefits between US and European participants.

REFERENCES

Adams, R., Bass, L., & John, E. B. (2005). Experience with using general usability scenarios on the software architecture of a collaborative system. In A. Seffah, J. Gulliksen & M. C. Desmarais (Eds.), *Human-Centered Software Engineering: Integrating Usability in the Software Development Lifecycle* (pp. 391). Dordrecht, The Netherlands: Springer.

Bass, L., & John, B. E. (2000). *Achieving Usability Through Software Architectural Styles.* Paper presented at the CHI, The Haghe, Netherlands.

Beyer, H., & Holtzblatt, K. (1998). *Contextual Inquiry: Defining Customer-Centered Systems*. San Diego: Academic Press.

Brinck, T., Gergle, D., & Wood, S. D. (2002). *Usability for the Web: Designing web sites that work*. San Francisco: Morgan Kaufmann.

Campbell, D. T., & Stanley, J. C. (1963). *Experimental and Quasi-Experimental Designs for Research*. Boston: Houghton Mifflin Company.

Creswell, J. W. (2003). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*. Thousand Oaks: Sage Publications.

Curtis, B., Krasner, H., & Iscoe, N. (1988). A field study of the software design process for large systems. *Commun. ACM, 31*(11), 1268-1287.

Dillman, D. A. (2000). *Mail and Internet Surveys: The Tailored Design Method*. New York: John Wiley & Sons.

Edwards, W. K. (2008). Infrastructure and Its Effect on the Interface. In T. Erickson & D. W. McDonald (Eds.), *HCI Remixed: Reflections on Works That Have Influenced the HCI Community* (pp. 119-122). Cambridge: MIT Press.

Golden, E., John, B. E., & Bass, L. (2005, May). *The value of a usability-supporting architectural pattern in software architecture design: a controlled experiment.* Paper presented at the 27th International Conference on Software Engineering ICSE, St. Louis, Missouri.

Greenberg, S., & Marwood, D. (1994). *Real-Time Groupware as a Distributed System: Concurrency Control and Its Effect on the Interface.* Paper presented at the ACM Conference on Computer Supported Cooperative Work, Chapel Hill, NC.

John, B. E. (2005). Evidence-Based Practice in Human-Computer Interaction and Evidence Maps. *ACM SIGSOFT Software Engineering Notes, 30*(4), 1-5.

John, B. E., Bass, L., Sanchez-Segura, M. I., & Adams, R. (2004, July 11-13). *Bringing Usability Concerns to the Design of Software Architecture.* Paper presented at the 9th IFIP Working Conference on Engineering for Human-Computer Interaction and 11th International Workshop on Design, Specification and Verification of Interactive Systems, Hamburg, Germany.

Karat, J. (1991). *Taking Software Design Seriously*. San Diego: Academic Press.

Karat, J. (2005). Bridging HCI, Usability and Software Engineering. In A. Seffah, J. Gulliksen & M. C. Desmarais (Eds.), *Human-Centered Software Engineering: Integrating Usability in the Software Development Lifecycle* (pp. xvii-ixx). Dordrecht, The Netherlands: Springer.

Making IT Better: Expanding Information Technology Research to Meet Society's Needs. (2000). Washington DC: National Academy Press.

Mayhew, D. J. (1999). *The Usability Engineering Lifecycle: A Practicioner's Handbook for User Interface Design*.

Nielsen, J. (2000). *Designing Web Usability: The Practice of Simplicity*. Indianapolis, IN: New Riders.

Norman, D. A. (1988). *The Design of Everyday Things*. New York: Basic Books.

Osterweil, L. J. (2007). *A Future for Software Engineering?* Paper presented at the International Conference of Software Engineering, Minneapolis, MN.

Preece, J., Rogers, Y., & Sharp, H. (2002). *Interaction Design: Beyond Human-Computer Interaction*. New York: John Wiley & Sons.

Proctor, R. W., & Zandt, T. V. (1994). *Human Factors in Simple and Complex Systems*. Needham Heights, MA: Allyn and Bacon.

Schuman, H., & Presser, S. (1981). *Questions and Answers in Attitude Surveys*. New York: Academic Press.

Seffah, A., Gulliksen, J., & Desmarais, M. C. (2005). Integrating Usability in the Development Process. In A. Seffah, J. Gulliksen & M. C. Desmarais (Eds.), *Human-Centered Software Engineering: Integrating Usability in the Software Development Lifecycle* (pp. 3-14). Dordrecht, The Netherlands: Springer.

Seife, C. (2000). *Zero: The Biography of a Dangerous Idea*. New York: Penguin Group.

Snyder, C. (2003). *Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces*. San Francisco: Morgan Kaufmann.

Sunk by Windows NT. (1998). Retrieved November 25, 2008, from

http://www.wired.com/science/discoveries/news/1998/07/13987

APPENDICES

Appendix A: Survey - Invitation Email

Dear usability professional,

You have been carefully selected to participate in this study because of your experience in the area of usability and software design. As a usability specialist we value your input and would like to gage your opinions about usability in the context of software design via an online survey. As a small token of our appreciation we would like to give you a $10 Amazon email gift card for completing the survey.

This research is being conducted by Dr. Davide Bolchini and Edgardo Luzcando, School of Informatics, Indiana University Purdue University Indianapolis.

Link to survey:
https://www.iupui.edu/~cccres/upse/index.php?main=entryCode (codes will be activated in the future)

Thank for your participation!

Appendix B: Survey - Introduction

A Survey Study of Software Design and Usability
Thank you for participating in this study!
This is a study of software design and usability to support graduate student research in Human-Computer Interaction (HCI). This research is being conducted by Edgardo Luzcando and Davide Bolchini, School of Informatics, Indiana University Purdue University Indianapolis (IUPUI). This study will ask you to answer questions related to usability and software design to gauge your opinions as a usability specialist.

-- APPROVAL: This exempt study has been approved (EX0805-17B) by the Institutional Review Board (IRB) for human subjects research at IUPUI. If you agree to participate, you will be one of approximately 100 subjects, and you will complete a web survey of no more than 20 questions, which will take no more than 10-15 minutes to complete.

-- RISKS: There are no risks associated with participating in the study outside of potential stress for having to complete online survey questions.

-- BENEFITS: You will experience the satisfaction of contributing to the advancement of knowledge in the area of usability research related to software design.

-- CONFIDENTIALITY: Subjects will respond to the online survey anonymously. No personal data will be collected or used for research purposes, however participants may optionally provide contact information after concluding the survey to receive a small token of appreciation (via email) for participating in the survey. No personal information will be distributed or shared with anyone outside this research study, unless required by law.

-- COSTS: There are no costs to you to participate in the study.

-- VOLUNTARY: Your participation is voluntary and may choose not to take part or may leave the study at any time. Leaving the study will not result in any penalty, and your decision whether or not to participate in this study will not affect your current or future relations with IUPUI.

-- CONTACTS: For questions about the study, contact the researchers: Edgardo Luzcando (eluzcand@iupui.edu) or Davide Bolchini (dbolchin@iupui.edu). Also, for questions about your rights as a research participant or to discuss problems, complaints or concerns about the research study, or to obtain information or offer input, contact the researchers.

SUBJECT'S CONSENT: Filling out this online survey is an acknowledgment that you understand the nature of the study and have given your permission to participate.

Appendix C: Survey - Instructions

You have been selected to participate in this study given your expertise in the usability and software design. If you received this survey in error, please notify Edgardo Luzcando (eluzcand@iupui.edu).

As a token of our appreciation for completing the survey, we would like to send you a $10 Amazon email gift card. To distribute the card, we will need your name and email to confirm your valid participation in the survey as usability professional. This personal information is not part of the survey and will not be used for any other purposes.

1. Please answer all the questions on each page and then hit "Continue" to proceed to the next page.

2. Do not use the BACK button!

3. There is no option to save and continue after you begin, so we ask that you complete it in one run. However, there is no time limit if you leave your browser window open and choose to take a small break after you begin.

These instructions will be available as a link at the bottom of the survey, which will launch a pop-up window.

Completing the questions should take between 10-15 minutes of your time. Thank you!

Appendix D: Survey - Pretest

Usability and Software Design

1. To what extent do you agree or disagree with this statement: "Usability is a very important aspect of software design."

- ☐ Strongly agree
- ☐ Somewhat agree
- ☐ Neither agree nor disagree
- ☐ Somewhat disagree
- ☐ Strongly disagree
- ☐ No opinion

---

2. In your experience, have you ever found it challenging to apply usability principles in a software design project?

- ☐ No
- ☐ Yes

---

3. Do you think there is a communication gap between usability professionals and software engineers when they work together in software design?

- ☒ No
- ☐ Yes

Comments (optional)

---

4. To what extent do you agree or disagree with this statement: "Usability-Supporting Architecture Patterns assist usability professionals identify usability concerns that impact the architecture of a software system."

- ☐ Strongly agree
- ☐ Somewhat agree
- ☐ Neither agree nor disagree
- ☐ Somewhat disagree

- Strongly disagree
- Don't know

---

5. Over the last 25 years it has been recognized that to design usable software, a blend of skills is required from several disciplines such as Computer Science, Software Engineering, and Psychology.

In your experience, did you work in very close contact with software engineers throughout the software development process?

- Strongly agree
- Somewhat agree
- Neither agree nor disagree
- Somewhat disagree
- Strongly disagree
- No opinion

---

6. HCI attempts to bridge the gap between usability professionals and software engineers with different levels of success.

Are you familiar with any methodologies or techniques to improve communication between usability professionals and software engineers during software projects? If so, which ones?

- No
- Yes

(If Yes, which ones?)

Appendix E: Survey - Treatment

Usability-Supporting Architecture Patterns (USAPs)

Imagine the following software project scenario:
Your team is developing a web-based collaboration application for a manufacturing plant. The developers created the software architecture and programmed the application according to the user requirements, and passed it to the usability professionals for evaluation. The usability test results are in, and the development team is ready to hear the recommendations from the usability professionals. As a major usability problem is presented, one of the developers says, "Oh, no, we can't change THAT! The way the software has been designed from the beginning does not allow changing this feature; it will take months, we have to rebuild the entire software architecture if you want to do that."

The requested modification to improve usability reaches too far in the architecture of the system to allow economically viable and timely changes to be made, even if the recommendation is right. As a result a less usable product is shipped to the customer in order to meet a timeline.

What are USAPs?
Usability-Supporting Architecture Patterns are a list of "solutions" to address the problem of how to take into account usability requirements in developing software architectures. They help translate an end-user usability requirement to the software properties useful to developers to program the system. Some of these patterns include: the ability to undo a running command, predicting task duration, and canceling a running command. Each USAP has three parts: a description, its responsibilities, and a sample solution. As they are defined, USAPs have the potential to serve as powerful design tools for software architects to incorporate usability requirements in their development process as early as possible.

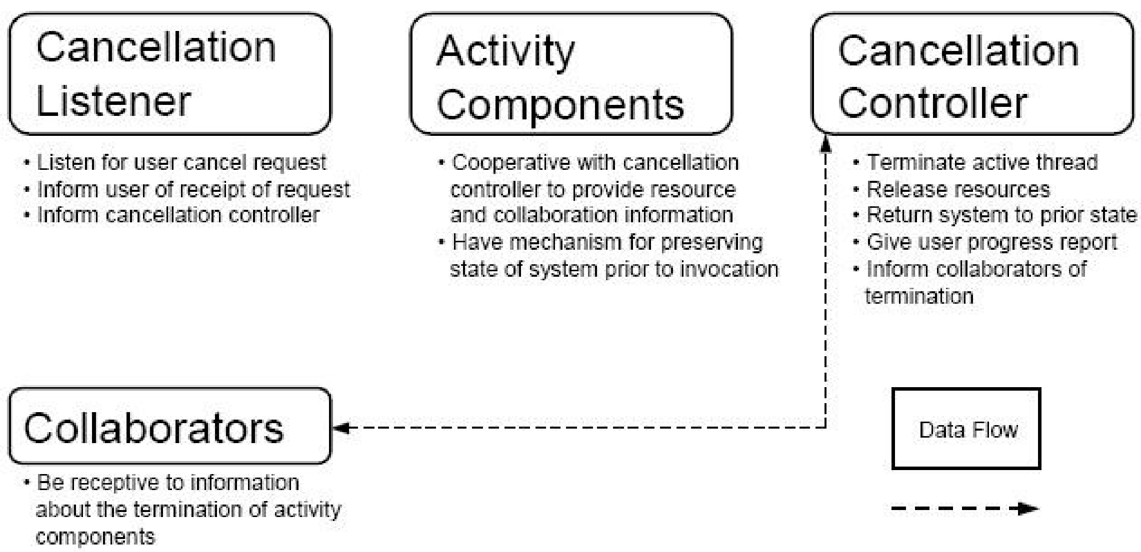
USAP Example - Canceling a Command

<u>USAP Description</u>
A user invokes an operation, and no longer wants the operation to be performed. The user now wants to stop the operation rather than wait for it to complete. It does not matter why the user launched the operation. The mouse could have slipped. The user could have mistaken one command for another. The user could have decided to invoke another operation. For these reasons (and many more), systems should allow users to cancel operations.

<u>USAP Responsibilities</u>
There are 19 responsibilities and only four are listed for brevity:

A button, menu item, keyboard shortcut and/or other means must be provided, by which the user may cancel the active command
The system must always listen for the cancel command or changes in the system environment.
The system must always gather information (state, resource usage, actions, etc.) that allow for recovery of the state of the system prior to the execution of the current command.
If the system is capable of rolling back all changes to the state prior the execution of the command, the system state must be restored to its state prior to execution of the command.

<u>USAP Sample Solution</u>
The sample solution is a low-level software diagram that delineates responsibilities for software modules:



Appendix F: Survey - Posttest

Using Usability-Supporting Architecture Patterns

7.The application of Usability-Supporting Architecture Patterns to software design helps software engineers better account for usability concerns. The following tables contain a list of usability-related benefits software should experience after the use of Usability-Supporting Architecture Patterns in a project. According to your experience, please rate each of the following usability benefits with regard to how important they are to the usability of software products, and provide optional comments if the benefit description is unclear.

Please rate each of the following usability benefits with regard to how important they are to the usability of software products.

Benefits from a software perspective	Very Important (VI)	Important (I)	Somewhat Important (SI)	Not Important (NI)	Don't Know (DK)	Optional Comments
Accelerates error-free portion	VI ☐	I ☐	SI ☐	NI ☐	DK ☐	
Reduces impact of slips	VI ☐	I ☐	SI ☐	NI ☐	DK ☐	
Supports problem-solving	VI ☐	I ☐	SI ☐	NI ☐	DK ☐	
Facilitates learning	VI ☐	I ☐	SI ☐	NI ☐	DK ☐	
Prevents mistakes	VI ☐	I ☐	SI ☐	NI ☐	DK ☐	
Increases confidence and support	VI ☐	I ☐	SI ☐	NI ☐	DK ☐	
Accommodates mistakes	VI ☐	I ☐	SI ☐	NI ☐	DK ☐	
Tolerates system errors	VI ☐	I ☐	SI ☐	NI ☐	DK ☐	
Prevents system errors	VI ☐	I ☐	SI ☐	NI ☐	DK ☐	

Please rate each of the following usability benefits with regard to how important they are to the usability of software products.

Benefits from a usability perspective	Very Important (VI)	Important (I)	Somewhat Important (SI)	Not Important (NI)	Don't Know (DK)	Optional Comments
Increases efficiency	VI	I	SI	NI	DK	
Reduce the impact of errors	VI	I	SI	NI	DK	
Provides user help	VI	I	SI	NI	DK	
Helps learnability	VI	I	SI	NI	DK	
Avoids mistakes	VI	I	SI	NI	DK	
Assists memorability	VI	I	SI	NI	DK	
Tolerates mistakes	VI	I	SI	NI	DK	
Masks system errors	VI	I	SI	NI	DK	
Prevents software lockup	VI	I	SI	NI	DK	

---

8. On the basis of what you have learned in this survey about USAPs, do you think it is useful to leverage their use in software design activities?

- Strongly agree
- Somewhat agree
- Neither agree nor disagree
- Somewhat disagree
- Strongly disagree
- No opinion

9.After you complete the survey, how likely are you to investigate USAPs further and learn more about them?

- [ ] Likely
- [ ] Not Likely

---

10. In your experience, to what extent do you agree or disagree with this statement: “the work relationship with software engineers is difficult”

- [ ] Strongly Agree
- [ ] Somewhat agree
- [ ] Neither agree nor disagree
- [ ] Somewhat disagree
- [ ] Strongly disagree
- [ ] No opinion

---

11. In your experience in software design to what extent do you agree or disagree with this statement: “There is a gap between software engineers and usability professionals when they collaborate in software design."

- [ ] Strongly agree
- [ ] Somewhat agree
- [ ] Neither agree nor disagree
- [ ] Somewhat disagree
- [ ] Strongly disagree
- [ ] No opinion

---

12. Who in your experience should lead software design activities - usability professionals or software engineers?

- [ ] Usability professionals
- [ ] Software engineers

Appendix G: Survey - Demographic Information

Education

13. Do you have a degree in any of the following?

- [ ] No degree
- [ ] Anthropology
- [ ] Computer Science
- [ ] Engineering
- [ ] Fine Arts
- [ ] Human Computer Interaction
- [ ] Psychology
- [ ] Other? ____

---

14. Did you get a Master, Doctorate, or post-graduate Degree?

YES ○ or NO ○

Experience

15. How many years of professional experience do you have?

○ None
○ 1-2
○ 3-5
○ 6-10
○ 11 or more

---

16. How many usability conferences do you attend a year?

○ None
○ 1-2
○ 3-5
○ 6-10
○ 11 or more

Appendix H: Survey - Additional Comments

Additional Comments (Optional)

17. Is there any additional information that you would like us to know?

Comments:

Bottom of Form

Appendix I: Survey - End

Thank you for completing the survey! As a token of our appreciation we would like to send you a $10 Amazon gift card. To do this we need a way to send it to you via email. Please fill out the form below with a valid email address. If you prefer not to disclose your personal information here, please feel free to send an email to Edgardo Luzcando (eluzcand@iupui.edu) and give us your survey ID: (e.g. 980516151602) so we can confirm you completed the survey and were a valid recipient of the survey.

All your information will be kept confidential and will not be shared with anyone else. It will only be used to send you the Amazon gift card.

Please enter your email address and name

Email ______

Name ______

For verification purposes that you were a valid recipient of this survey please tell us the following:

How did you hear about this study?

- ☐ UPA Officer
- ☐ HCI University Faculty
- ☐ Other ______

Who sent you the invitation to participate in the survey?

Name: ______

Appendix J: Survey Technical Design Details

The survey was conducted using HTTPS for encrypted communications between every participants and the university server. This would further assist in protecting the participant's privacy over typical HTTP connections. While any user could access the URL, only entry in the URL with a valid code (prescribed by the researchers) would allow the users to continue and generate valid data that would be used for analysis.

The GUI of the survey was constructed with HTML Frames, which allowed for a bottom frame to be available at all times and offer navigation and the opportunity to view the initial instructions at any time during the survey via a link to a pop-up window. The survey leveraged PHP version 5.2.4 powered by Zend Engine version 2.2.0 running on a Linux lux2 2.6.9-78.0.5.ELsmp #1 SMP system. The database MySQL 5 provided the client API version 5.0.45. All PHP files used a header as depicted in Figure 3.

```
#!/usr/local/bin/php
<?php
# ========================================================================#
# Graduate Student Research - Usability-Supporting Architecture Patterns
# Edgardo Luzcando - eluzcand@iupui.edu - www.iupui.edu
# Indianapolis, Indiana, USA
# ---------------------------------------------------
#  Version 1.0
#  Creator: Edgardo A. Luzcando
#  Copyright: 2008
# ======================================================================
# This is a study of USAPs applied to software design
# ======================================================================

```

Figure 8 - Survey PHP File Header Example

The entire framework might be useful in future research conducted at the university, and can be used for such purposed. All of the artifacts created for this survey are available upon request to students conducting research at the university that which to develop their own surveys internally. Figure 4 contains a list of the existing artifacts available upon request and Figure 5 contains a list of the database tables used.

additionalComments.php	5 KB
demo.php	8 KB
end.php	7 KB
finisher.php	5 KB
index.php	9 KB
instructions.php	5 KB
popupInstructions.php	5 KB
posttest.php	16 KB
pretest.php	4 KB
qual.php	7 KB
storeDemo.php	1 KB
storePosttest.php	9 KB
storePretest.php	2 KB
storeQual.php	1 KB
storeTreatment.php	1 KB
survey.php	7 KB
test.php	1 KB
treatment.php	4 KB
upseDebug.php	3 KB
upseFrameSet.php	3 KB
upseGuide.php	6 KB
upseInit.php	7 KB
upseRouter.php	3 KB

Figure 9 - Survey PHP File List

Table	Action						Records	Size
upseClosedQs	Browse	Select	Insert	Properties	Drop	Empty	1,358	81.4 KB
upseComments	Browse	Select	Insert	Properties	Drop	Empty	46	6.9 KB
upseDemo	Browse	Select	Insert	Properties	Drop	Empty	45	5.0 KB
upseEnds	Browse	Select	Insert	Properties	Drop	Empty	46	4.5 KB
upseInit	Browse	Select	Insert	Properties	Drop	Empty	59	4.8 KB
upseOpenQs	Browse	Select	Insert	Properties	Drop	Empty	100	14.4 KB
upseOptComments	Browse	Select	Insert	Properties	Drop	Empty	828	54.5 KB
upsePersonal	Browse	Select	Insert	Properties	Drop	Empty	43	6.2 KB
upsePopInstru	Browse	Select	Insert	Properties	Drop	Empty	8	2.2 KB
upseStarts	Browse	Select	Insert	Properties	Drop	Empty	69	6.4 KB

Figure 10 - Survey MySQL Database Tables

Appendix K: Open-ended Questions Data

Table 12 and Table 13 show the open-ended data for questions Q3 and Q6 in coded format for the participants. Table 24 and Table 25 show the actual answers[13] used to code the results. The data presented here includes all participant responses, including the ones outside of the 35 clean datasets. Five such responses are in Table 24, and three in Table 25.

Table 24 - Reported Open-Ended Comments about the Communication Gap?

Usability professionals close the gap between end-users and software engineers (2)
Yes, between communication professionals and software engineers
The difference is in the point of view: software engineers focus on insfrastructure and usability professionals on making more usable interfaces
Sometimes software engineers have no appropriate knowledge of usability
Yes (most of the time), depending on a software engineer's background and expertise in usability matters
Yes…about platform issues
Yes…in terms of required specification documents
The biggest problem is not a knowledge problem, but a trust and respect problem (regarding each group's commitment to develop usable products)
Yes, from not fully understanding how both groups can work together to develop a product
There will always be a gap between the groups, and it is the job of usability professionals to bridge this gap and create overall usability awareness
Software engineers welcome suggestions for usability improvements, but do not consider usability their core competency
Software engineers do not understand that end-users are not like them, and that the product must take into account the end-users' level of expertise at using software
Yes…sometimes usability professionals do not understand the limitations of software/hardware, and software engineers do not understand the need for testing usability principles
Yes, but it depends on the organization and development process. The benefits of "selling" better usability can be limited by time or technical reasons.
Yes, but it is getting better. Some software engineers do not have access to usability professionals and they go off on their own insticts, but both groups are open to usability research and improvements in design practices
Software engineers think differently when designing, and not through the eyes of intendend users. Withough collaboration between both groups, usability will not be optimal
Usability professionals are rarely available for software engineers to collaborate with in projects

[13] A few minor edits were done for readability to correct grammar and spelling

Usability professionals demand that “simple” changes (in their eyes) be turned around quickly by sotware engineers without understanding the complexities of the requested changes, and often is it also difficult to understand what they need and how to translate that to software
It is more an awareness problem than a gap. Sofware engineers typically want to create usable software, but are more concerned with software features and deadlines
Engineers fail to understand usability principles, and usability professionals are often brough in late in the product lifecycle. Earlier entry would help bridge the gap and improve communication
Sometimes a common understanding is missing
Software engineers are generally uneducated in the ways of usability. However, once they understand the benefits of developing usable products, a large majority of them embrace it

Table 25 - Reported Open-Ended Answers for Known Gap-Bridging Methodologies

Mile + (2)
Workshops, user experience posters
Notation standards or communication protocols
Have used AWARE, but have found through experience that it is effective to use a simple representation of application behavior
I am familiar with "cross-discipline" usability methods but not specifically aimed at improving communication between the 2 groups
Engineers have to attend usability cycles and usability professionals have to understand platforms and system architecture to be able to make recommendations for a better usability
I mainly use mock-ups or examples
The usual suspects: Meetings, workshops, participation in user testing/ research sessions...
-- early communication between the groups -- engineers involved in usability observations -- walk-troughs with engineers who request usability input
Communication and awareness. This cannot be stressed enough. Never assume an engineer knows usability
Scrum methods can be effective due to co-location and understanding the goals of co-workers
Contextual inquiry
Not sure how to answer this question. I use a lot of techniques to communicate with software engineers - however, they are not specific 'methodologies'. Most are just techniques to educate teams about the role and value of usability and user-center
Agile?
Building of cross-functional teams: having a Usability Analyst sit "outside" of the product development team creates the gap you want to bridge. Including the Usability Analyst as a core member of a dev team (which also includes UI Design, Interaction Design, etc.)
Project planning with meetings before, during, and post software engineering
An iterative approach to research is very informative and can gradually bring in feedback - they may otherwise be overwhelming to both usability professionals and engineers. Consumer feedback is always valuable to show engineers
Domain Drive Design, Behavior Driven Development, Modeling Color. While these techniques are not specifically designed for usability professionals and/or communication they do promote good communication between developers, analysts, customers, etc.
Conceptual Comics, Prototypes, Wireframes

Maybe the Rational Unified Process counts, I'm not sure. Use cases can be usability tools, but they're still really hard to understand.
I think modeling tools could help.
Prototyping paper and otherwise.
Scenario based design and cooperative design are really good in letting people work together and express design issues.

VITA

Edgardo A. Luzcando F.

edgardo@luzcando.com - (317) 753-6298

Education

Master of Science in Human-Computer Interaction, Expected December 2008
School of Informatics, Indiana University Purdue University at Indianapolis (IUPUI)
Thesis: Evaluating Usability-Supporting Architectural Patterns: Reactions from Usability Professionals
Advisor: Davide Bolchini

Bachelor of Science in Electrical Engineering May 1998
Marquette University, USA.

Research Interests

User interface requirements and software engineering
Usability of user interfaces
Cross-cultural web design

Professional Experiences

Midwest ISO (Sep 05 - present) – Applications Architect
Subcontractor (Jun 05 – Sep 05) – HCI Consultant
Midwest ISO (Jul 03 – Sep 05) – System Analyst
Verizon Communications (Jul 98 - Jul 03) - Software Engineer
Marquette University (Jan 97 - May 98) - Teaching Assistant / Lab Assistant
Eaton Corporation (Jun 95 - Dec 96) – IT Consultant (Co-op)
Marquette University (Jan 95 - May 96) – Restaurant Manager
US Army (Jun 94 - Aug 94) – Office Clerk (Internship)

Personal

Phi Mu Epsilon
Eta Kappa Nu
SHPE President, Marquette University chapter, 1997-1998
IUPUI Graduate Student Organization VP and representative of the School of Informatics, 2003-2005
Chair of the Architecture Working Group for ISO/RTOs of North America, 2009