

Claims Maps: Treasure Maps for Scenario-Based Design

Shahtab Wahid, D. Scott McCrickard
Center for Human-Computer Interaction and Department of Computer Science
Virginia Polytechnic Institute and State University (Virginia Tech)
Blacksburg, VA 24061-0106
{swahid, mccricks}@cs.vt.edu

Abstract: Design representations express the state of a design with respect to a followed design procedure. Scenario-Based Design (SBD) is an HCI usability engineering process that leverages claims, knowledge units encapsulating tradeoffs associated with design features. HCI students need to express the concepts and rationale in their interface design related to SBD, but the lack of a design representation and tool inhibits expressing key details. We present claims maps as a representation to show the connections established between claims as a result of the SBD process. An evaluation in a introductory course shows students can identify claim types, associate system goals to features, gain insight into potential evaluation plans, and do find greater utility in using claims maps.

Introduction

The discipline of Human-Computer Interaction (HCI) has yielded various usability engineering processes for the development of interfaces. These processes emphasize different aspects of a design at various points in time. Students learning these design techniques need to be aware of how usability engineering affects the state of their design. HCI education demands students need to express the concepts and rationale behind their work. However, in many cases, such explanations can be a daunting task if students do not have a representation, or diagram, of their design. Simple prototypes of interfaces do not always make the steps taken toward creating the system explicit. Design representations portray the current state of a design by taking procedural steps into account. There is a need for a visual representation of the fundamental usability engineering work undertaken by students, ensuring they can express and justify the necessary steps. Additionally, this representation must be flexible, but also tailored to the specific concepts students must be exposed to. Scenario-Based Design (SBD) is one such usability engineering process that focuses student design efforts on using claims, knowledge units constantly weighing the consequences of design features. Unfortunately a design representation and a supporting tool that articulates the use of claims in SBD does not exist. This can lead students to an inability to articulate distinctions between key design steps, fundamental system goals, and areas of focus for evaluation and redesign.

This paper proposes a new design representation called a claims map. A claims map visually defines the key connections that are established between claims as a result of following SBD. We first give an overview of SBD and claims and then provide an explanation of the representation. We continue with an evaluation which compares our representation to the current method of portraying SBD work. The observed phenomena is chronicled and discussed. Finally, we conclude by discussing ways to use claims maps and what will be needed for tool support.

Scenario-Based Design

SBD (Rosson & Carroll 2002) is a usability engineering process in which scenarios, narratives describing a particular task, are used as a base for creating interactive designs. SBD uses various types of scenarios to guide the design process. An analytic evaluation process within SBD, called *claims analysis*, identifies scenario features that have usability consequences and stores this information in a structure called a *claim*. Claims are natural language records of the positive and negative effects of design features (see Figure 1) (Rosson & Carroll, 2002). They address a variety of situational and interface aspects that affect the compatibility of the design and user models, such as user satisfaction and feeling of reward, color and object layout, and strength of affordances. Inherently objective, claims provide designers with a pure view into what makes an artifact live and breathe, grounded in theories and observations of user experiences.

Organizing information items using a collage metaphor

- + Allows users to informally post information without any regard to organization
- + Allows users to gain an understanding of an item's age/applicability with respect to the number of items that may be covering it
- + Lack of information categorization accommodates a wide range of different types of information to be placed
- BUT the lack of organization can hinder efforts to find a particular information item
- BUT overlapping items may force users to move items in order to fully reveal themselves

Figure 1: A claim about the collage metaphor showing the positive and negative consequences (Greenberg & Rounding 2001).

SBD advocates four distinct phases on design: requirements analysis, activity, information, and interaction design. In the requirements analysis phase, practitioners are asked to consider the current practices of potential users through problem claims. Once completed, focus is placed on identifying claims for the activities that must be provided in the new interface. Claims depicting the methods for conveying the information and interactions needed to support the chosen activities follow, creating a vital link between goals and features. Together, the different claims describe the conceptual design of an interface. The clear distinction between the four phases permits new student practitioners to grasp crucial experiences which are internalized by more experienced designers. The simple, yet effective concept of a claim makes them ideal knowledge units to put into use within educational settings. Students exposed to claims develop their skills by explicitly considering impacts of their decisions. Through a constant process of gauging effects, aspects of features often ignored are placed into the forefront, aiding the focus of user evaluations.

As a result of this usability engineering technique, we have observed students being confronted with a large amount of claims, leading to several potential problems. Identifying the most important claims to reflect the goals of a system becomes a challenge. Often students can not distinguish the four different types of claims because the natural language used can seem to fit multiple design phases. Even more important is being able to identify which information and interaction claims are chosen to support particular activities. Finally, identifying the correct set of claims to target for specific evaluations is largely left to a designer's own intuition.

Related Work

Various forms of visual representations have been used by different communities. Each form provides a unique perspective into a design, yielding what is believed to be the most imperative information. We wish to have a representation that will depict requirements, goals, tasks, tradeoffs, distinctions between information and interaction techniques, and the connections between them.

The software engineering community has embraced the Unified Modeling Language (UML) as a standard way of providing multiple perspectives into software designs (Purchase et al. 2001). The incorporation of UML diagrams into university courses is discussed on many occasions (Moritz & Blank 2005)(LeBlanc 2000)(Burton & Bruhn 2004)(Brewer & Lorenz 2003). Perhaps the most common UML diagram is the class diagram, a representation of the elements of classes and the relationships that exist between classes. They provide valuable insight into software architecture, but can not show goals, tasks, or design tradeoffs. Use cases are components that describe system interactions by incorporating scenarios. A use case diagram describes the overall behavior of a system and how it interacts with external actors. The incorporation of a scenario within a use case does leverage the notion of interaction, however, use cases are not enough to portray the design rationale and possible side-effects of actions. For such reasons, we can not rely upon these diagrams, but wish to take advantage of the notion of specific relationship types to show important connections.

Concept maps are graphical representations introduced by Novak (1984) for organizing knowledge and representing the relationships between them. They present great utility for teachers as they are known to be an effective way for students to learn new concepts (Wu et al. 2004). The process of making concept maps allows students to identify the key concepts and integrate them through linking words or relationships. They can be presented in a hierarchical structure starting with broader concepts that trickle down to smaller ideas (Novak & Canas 2006). This structure is

beneficial showing the connections between goals and information and interaction techniques. Without strict rules for how they should be created, concept maps offer a flexible format for expressing knowledge. The relationships, however, are left to the individuals to create the concept maps, potentially leading to inconsistencies.

One of the most common diagrams in HCI is the Hierarchical Task Analysis (HTA). Through the process of task analysis, individual tasks and subtasks are organized into a hierarchy (Rosson & Carroll 2002). The HTA representation starts with a general node depicting a task. Each successive lower level contains nodes which summarize subtasks that need to be accomplished. The advantage of HTAs are their ability to show connections between differing granularity levels. Their strength lies in depicting interactions, but they are weak in expressing information delivery techniques. Students creating such diagrams learn to decompose tasks and gain a sense of how complex a particular task might be. We require both components to depict the intricacies of tasks within systems.

Claims Maps

The plethora of UML diagrams forces us to recognize the need for a persistent design representation—one that will express the most important factors needed by student practitioners. Unfortunately UML diagrams are not sufficiently acceptable for usability engineering processes like SBD. Concept maps offer flexibility, but the lack of specific relationship types hinders efforts to create a consistent representation. Although HTAs are used by usability engineers, certain pertinent information is left out. Inspired by the need for representation and the weaknesses of available ones, we propose *claims maps*—a new visual representation technique for SBD using claims. Claims maps seek to show the claims in each phase of SBD and the connections between them, providing valuable insight needed to understand the current state of an interface. Utilizing the connections allows for identification of claim types, association of information and interaction claims to activity claims, and realization of the claims to target for evaluations and redesign. Although fully functional tool support does not exist yet, we plan to create tools that will aid students in the creation and maintenance of claims maps.

Claim Relationships

As we previously mentioned, designs created as a result of SBD yields several claims from each design phase. Together, the claims as represent the conceptual design of an interface. It is important to create a “whole” by understanding how each claim plays a role in the overall system and connects to other claims. Claims map utilize the concept of *claim relationships* (Wahid et al. 2004) to describe the intricacies of such connections. We present the 10 relationships and explain their value.

- Postulation/Predication: A postulation relationship exists when connecting a problem claim to an activity claim and an activity claim to information and interaction claims. The predication relationship exists in the opposite direction.
- Execution/Evaluation: Interaction claims describe user execution while information claims show opportunities for the user to evaluate the state of a system. The execution and evaluation relationships depict the connection among and between interaction and information claims, portraying user task flow.
- Generalization/Specification: This relationship connects claims written with different scopes in mind to leverage different granularity levels. A broader claim leads to a specific claim through a specification relationship. A generalization is established in the opposite direction.
- Fusion/Diffusion: Two or more claims may be combined together through the fusion relationship. Diffusion portrays the breaking up of a claim into different claims. These relationships are important because they can express design decisions that yield new interface artifacts through a combination of claims.
- Translation: Often designers are faced with a choice between multiple possibilities. The bidirectional translation relationship illustrates alternative features that complement each other within an interface.

- Mitigation: The tradeoffs within a claim are the essence of claims. The negative tradeoffs are adverse consequences designers must strive to eliminate or minimize. The unidirectional mitigation relationship creates a connection to claims that are used to specifically target downsides.

Claims Map Example

To thoroughly explain the notion of a claims map we provide an example and discuss the types of insights that can be acquired. We choose to describe the Notification Collage (NC) in our claims map. The NC is a collaborative system which allows a community of users to share information through public and private displays (Greenberg & Rounding 2001). The information can be in various forms such as notes, video feeds, slideshows, webpages, and graphs (see Figure 1).

A complete claims map consists of problem, activity, information, and interaction claims (see Figure 2). The claims map starts with problem claims at the top, activity claims situated in the middle, and information and interaction claims toward the bottom. For the sake of simplicity, this claims map uses orange for problem claims, blue for activity claims, and yellow for information and interaction. The upsides and downsides are normally included with a claims map as supplementary information.

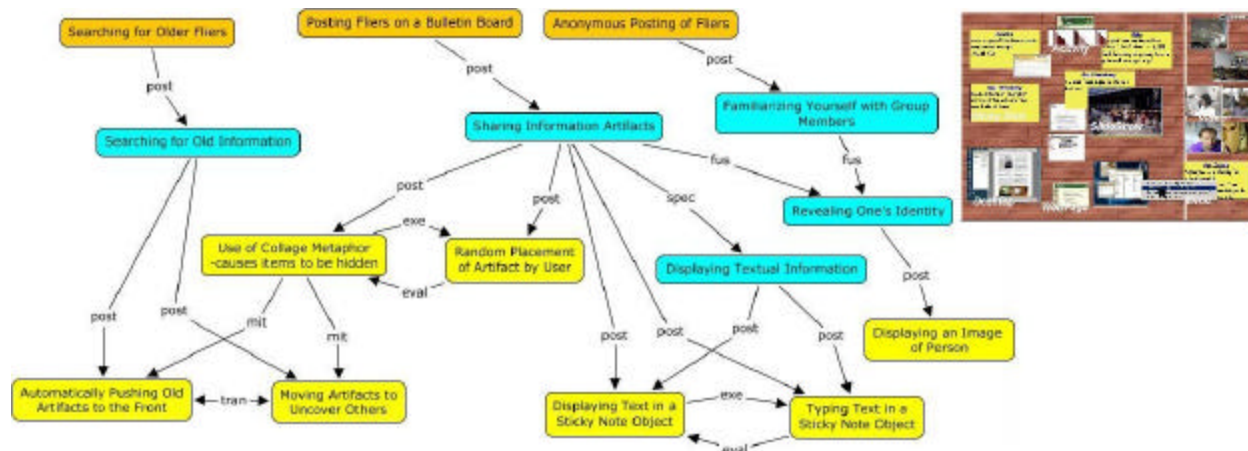


Figure 2: A claims map for the Notification Collage system shows the driving problems, main activities, and supporting information and interaction techniques. The relationships illustrate connections between goals and system features and establish task flow.

The first piece of information that can be gathered is the association of the requirements analysis to the activity phase. From the claims map one can infer that the current method for sharing information uses a bulletin board. A bulletin board allows a community to post fliers anonymously. Since fliers get covered up, individuals will have to uncover fliers to find older fliers. These three characteristics, represented as problem claims, are directly tied by various reasons to three general activities supported by the NC through a postulation relationship. For example, the activity claim about sharing information artifacts is a postulation of posting fliers problem claim because the NC wants to retain the information transfer capabilities of bulletin boards. On the other hand the activity claim about familiarizing oneself with group members is a direct reaction to a bulletin board incapability to do so. Such reasoning is valuable to justifying key characteristics of any new system. By looking at such a representation, students learn the importance of the requirements analysis process since they benefit from articulating motivating problems that lead to the design of the system. The postulation relationships are critical to establishing such insights and identifying the distinction between the claims.

The activity claims represent the goals of the NC. The claim about sharing information is specified into another claim about sharing textual information, demonstrating different granularities in goals for the system. The claim is also fused with the claim about familiarizing oneself with others to yield the notion of revealing one's identity. This is another activity that will be supported as a result of wanting to share information and foster a tighter community.

Activity claims are the characteristics that must be supported by the information and interaction claims. It is important to note the same activities could be supported in other ways through different information and interaction claims. Therefore, students should learn to express the exact claims used to support activity claims. The second layer of postulations forms the connection between system goals and design features. For example, the activity regarding the sharing of textual information is achieved by an information claim that leverages the notion of a sticky note. Placing text within this artifact allows others to quickly interpret the text as a note. Similarly, typing text into the note allows users to manipulate the artifact. The cyclical relationship between the information and interaction claims is also expressed by the execution and evaluation relationships, identifying a possible user task flow within the interface and making them critical to distinguishing information from interaction.

The mitigation relationship also makes explicit the adverse consequence of the collage metaphor. The collage metaphor allows for artifacts to be placed over other artifacts without any regards to organization. As a result, some items can get hidden by others. The two mitigating claims, designed to support searching activities, are also chosen because they negate the effect of the collage metaphor's downside. While every relationship in the claims map provides a form a rationale for why a claim exists, the mitigation relationships makes this notion more apparent. Students engaging in the process of interpreting relationships are exposed to the justifications of designers. One designer solution is to allow users to move artifacts around to uncover others. The other possibility is to wait for the system to automatically force a covered item to the surface. The first solution is an interaction claim and the second is an information claim, however, an execution and evaluation relationship does not exist because they are not part of the same task flow. Instead, the translation relationship symbolizes the existence of alternative methods.

The degree to which the upsides and downsides have an impact must also be evaluated to fully appreciate the knowledge encapsulated within claims. In SBD, students learn to refer to the tradeoffs to understand what aspects must be tested. In situations where there are many claims, students must also know which claims to target for specific evaluations. The claims map can express which claims to target when a portion of an interface must be evaluated. For example, if a student wishes to evaluate the textual information sharing capabilities of the NC, the student must recognize that the activity claim and the two information and interaction claims must be considered all together to when preparing an evaluation plan. If, as a result of an evaluation, one of the claims must be dropped from the system, the same inferences may be used to consider which other claims must also be eliminated.

Evaluation

Our goal was to evaluate claims maps to determine how effective they are for students following the SBD methodology. We concentrated on investigating four points of interests in our evaluation. First, we wanted to gauge the ability of claims maps to let students identify different types of claims. Second, we needed to know if students can identify which information and interaction claims belong to activity claims. Third, we aimed to investigate whether claims maps would aid students in determining which claims are affected during evaluation and redesign. Finally, we set out to determine how well a claims map could describe an overall system.

Experimental Design

Students in two introductory undergraduate HCI courses were chosen to participate in our claims map study. All of the students were exposed to SBD and claims from the beginning of the semester and completed a previous assignment on claims authoring. In total 43 students were divided into 20 groups of 2 or 3 students and asked to complete an in-class activity. The activity was designed to compare claims maps to the current method of listing all the claims for a system. We chose two distinct systems, SideShow and the NC, as systems to represent and created the required material for each. SideShow is a notification system designed to deliver dynamic information such as the online status of buddies, weather conditions, the number of bugs, and the number of e-mails (Cadiz et al. 2002). SideShow was represented using a typical list of claims which included complete upsides and downsides. Although the claims for SideShow were ordered by design phase (activity, information, and interaction) the students were not told of this ordering. Due to the time constraints of the class, the problem claims were left out to allow for students to complete the activity in a timely manner. The NC was represented by a claims map similar to Figure 2 and was accompanied by a complete list of claims showing the upsides and downsides of each claim in the map. The claims map did not include the problem claims and the color coding used in Figure 2. Explanations of each relationship type and examples were provided. Screenshots of both systems were made available to all the students.

Two conditions were created under our experimental design. In the first condition (SS1-NC2), half of the groups were first exposed to the SideShow system. Once they finished their task, they repeated the same task for the NC. Groups in the other condition (NC1-SS2) were first given the NC documents and then the SideShow material for the same task. The task asked the groups to analyze the representations and answer a series of questions. To test our first hypothesis the class identified all of the activity, information, and interaction claims. For our second hypothesis participants were instructed to list all the corresponding information and interaction claims for each activity claim they identified. Our third hypothesis led us to asking the class which additional claims they would target if the two claims were to be evaluated or dropped for the system. Three questions were asked for our last hypothesis. Groups provided a rating based on a 7-point Likert scale to three questions regarding how well the particular system's goals were reflected, how well the features were expressed, and how well the representation allows the connections between goals and features.

Results and Discussion

Our first target was to analyze how accurately students could label claims as activity, information, and interaction claims. Accuracy was determined by taking the total number of correctly identified claims and dividing it by the total number of claims given as answers. Significant differences were calculated using a t-Test. In the SS1-NC2 condition we did not observe a significant improvement in accuracy once students were given the claims map and in the NC1-SS2 condition a significant decrease in accuracy was not detected when the list of claims was given. We did find an interesting result when comparing the representation the students were first exposed to. Students working on SideShow in condition SS1-NC2 had a lower accuracy (Mean = 0.66, SD = 0.16) for activity claims than the students working on the NC in condition NC1-SS2 (Mean = 0.92, SD = 0.12). A significant difference was found between these results ($p = 0.001$). The same could not be said when analyzing the information and interaction claims in the same manner. Although significant improvements were not found within conditions, this result is a step toward confirming our first hypothesis.

A possible explanation for this is the students in the SS1-NC2 condition relied on interpreting the natural language of claims to determine the type. They then carried on this approach for the claims map, relying less on the relationship types. Conversely, the NC1-SS2 students relied both on natural language and the relationships to determine the claim type. The students acquired simple skills to detecting the differences between activity claims and the other claims. Therefore, when faced with the claims list, a significant decrease in activity claim accuracy did not occur for SideShow (Mean = 0.90, SD = 0.12). This analysis points us to two important phenomena. First, an inherent understanding of connections between claims provides a better understanding of the claim itself. Second, these results may mean there is additional value in color coding claims in maps to ease identification.

To investigate our second hypothesis we analyzed the results of the second set of questions. For every identified activity claim, students were asked to list the corresponding information and interaction claims. Only the answers for the correctly identified activity claims were analyzed. The analysis yielded appealing results, allowing us to accept our second hypothesis (see Table 1). In the SS1-NC2 condition a significant improvement was found when comparing the accuracy for the map ($p = 0.03$). A significant degradation was observed in the NC1-SS2 condition ($p = 0.01$). We find the placement of activity claims toward the top of the map and the explicit relationships allowed the students to easily identify the corresponding claims. The structure of the map afforded a more concrete mental model of the system that may even be an easier way for students to think about the design. Although not concrete evidence, we did observe one group attempt to create a claims map out of the SideShow claims list.

SS1-NC2	
SS	NC
Mean=0.35 SD=0.14	Mean=0.58 SD=0.27
NC1-SS2	
NC	SS
Mean=0.64 SD=0.21	Mean=0.39 SD=0.16

Table 1: Results of linking activity to information and interaction. Significant results are colored.

SS1-NC2	
SS	NC
Mean=0.79 SD=0.24	Mean=0.81 SD=0.29
NC1-SS2	
NC	SS
Mean=1.00 SD=0.00	Mean=1.00 SD=0.00

Table 2: Results of identifying additions claims to evaluate. Significant results are colored.

For our third hypothesis we asked what other claims should be targeted in the event a claim we chose were to be evaluated or dropped. Most of the students in both conditions were able to correctly answer the questions. Although no significant differences were found between systems within conditions, we did detect a significant difference when comparing the systems students worked on first. The students who worked on the NC claims map first did significantly better than the groups that worked on the SideShow list first ($p = 0.01$) when answering one question about evaluating a claim (see Table 2). Significant differences for the questions regarding dropped claims were not found. Consequently we can not fully accept nor reject our third hypothesis. It is our belief that the claims map offers more insight into which claims should be evaluated, but perhaps the relationships themselves are not the only factor to consider when choosing which claims to target. Students also need to apply their own judgment in this process.

Our last goal was to get an overall impression of the descriptive capabilities of claims maps. Participants were asked to provide ratings based on a 7-point Likert scale. The average rating of all the questions in both conditions consistently favored the claims map (see Table 3), however only some comparisons yielded significant differences. When asked to rate how well the goals of a system were represented the NC1-SS2 condition yielded a significant difference ($p = 0.01$). Student comments indicated the claims map was better at showing the goals because they were at the top of the map to express high level concepts, showing the structure of claims maps proves to be an aid to HCI education. Comments for the list of claims for SideShow indicated some difficulty in associating the goals to each other, causing the drop in the rating. Others mentioned the lack of claim tradeoffs in maps as a weakness. Although this was not anticipated, it is not surprising. It is harder for a claims map to show all the claim tradeoffs when the number of claims increases.

System Goals		System Features		Goals to Features Connection	
SS1-NC2		SS1-NC2		SS1-NC2	
SS	NC	SS	NC	SS	NC
Mean=4.75 SD=0.70	Mean=5.2 SD=1.31	Mean=3.65 SD=1.30	Mean=4.3 SD=1.33	Mean=2.65 SD=1.99	Mean=5.6 SD=0.96
NC1-SS2		NC1-SS2		NC1-SS2	
NC	SS	NC	SS	NC	SS
Mean=4.8 SD=0.63	Mean=3.2 SD=1.68	Mean=4.2 SD=1.03	Mean=3.8 SD=1.03	Mean=5.0 SD=1.56	Mean=3.1 SD=2.02

Table 3: Results from the questions regarding the representations methods favor claims maps. Significant differences are colored.

The second question regarding the representation of system features did not express any significant differences. For the claims map, participants noted the upsides and downsides of the information and interaction claims were not shown. Other comments were about some of the feature claims not being specific enough, but these concerns were expressed for both representations. This may be because of student expectations for information about features beyond claims even though such information is not necessarily an important concern of SBD.

The last question regarding the connections between goals and features had significant differences within both conditions ($p = 0.0007$ for SS1-NC2 and $p = 0.01$ for NC1-SS2). Comments overwhelmingly favored the claims map because it used the relationships to explicitly describe the connections. These connections were not easily found when looking at the list of claims. This explains the higher accuracy rates for the NC when identifying information and interaction claims for activity claims. The relationships clearly emphasize to students that in SBD they need to identify information and interaction claims that specifically support activity claims rather than randomly creating features for a design. The answers to these last three questions allow us to conclude that students do find greater utility in claims maps than lists.

Conclusions and Future Work

Inspired by the need for a design representation for SBD, we proposed claims maps. Claims maps are true reflections of work done under the SBD process with respect to claims. They are designed as tools for students to use to ensure they better understand the intricacies of SBD and design in HCI. We evaluated the claims map and a

traditional list of claims. Results indicate they can help claim type identification, be a beneficial aid to identifying connections between system goals and features, and provide insight into potential evaluation areas. Additionally, students find greater advantages in using claims maps.

We consequently advocate the use of such a representation for courses that follow the SBD methodology. Our results show there is great potential for claims maps as a manifestation of SBD work. Knowing this allows us to consider the claims map as a probable representation of evolving interfaces for student projects. As systems are created, maps symbolize the state of an interface at each stage. Furthermore, a series of maps can express the changes a project has been through, providing valuable insight into which changes were made and the reasons for why they were made, leading up to the eventual creation of a case study. To support such a vision, we are working with students from an undergraduate research group and the introductory HCI class where they are instructed to create claims maps for each iteration of their projects. Tool support will be needed to ease the creation of claims map. Students using such a tool will be guided during the creation of claims maps. Additionally, digital claims maps can allow students to leverage each others claims by extracting them from other projects and reusing them in new and innovative ways, enhancing their knowledge sharing capabilities.

References

- Brewer, J. & Lorenz, L. (2003). Using UML and agile development methodologies to teach object-oriented analysis & design tools and techniques. *Information Technology Curriculum, Lafayette, Indiana*, 54-57.
- Burton, P. J. & Bruhn, R. E. (2004). Using UML to facilitate the teaching of object-oriented systems analysis and design. *Journal of Computing in Colleges*, 19 (3), 278-290.
- Cadiz, J. I., Venolia, G., Jancke, G., & Gupta, A. (2002) Designing and Deploying an Information Awareness Interface. *Computer Supported Cooperative Work*, 314-323
- Greenberg, S. & Rounding, M. (2001) The Notification Collage: Posting Information to Public and Personal Displays. *SIGCHI conference on Human factors in computing systems*, 524-521
- LeBlanc, C. (2000). UML for undergraduate software engineering. *Journal of Computing in Small Colleges*, 15, (5), 8-18.
- Moritz, S. H. & Blank, G. D. (2005). A design-first curriculum for teaching Java in a CS1 course. *SIGCSE Bulletin*. 37, (2), 89-93.
- Novak, J. D. & Canas A. J. (2006). The Theory Underlying Concept Maps and How to Construct Them. Technical Report IHMC CmapTools 2006-01, Florida Institute for Human and Machine Cognition
- Novak, J. D. & Gowin, D. R. (1984) *Learning How to Learn*. Cambridge Press, New York.
- Purchase, H. C., Colpoys, L., McGill, M., Carrington, D., & Britton, C. (2001). UML class diagram syntax: an empirical study of comprehension. *Australian Symposium on Information Visualisation*, Darlinghurst, Australia, 113-120.
- Rosson, M. B. & Carroll, J. M. (2002). *Usability Engineering: Scenario-Based Development of Human-Computer Interaction*. Morgan Kaufmann.
- Wahid, S., Allgood, C. F., Chewar, C. M. & McCrickard, D. S. (2004). Entering the Heart of Design: Relationships for Tracing Claim Evolution. *Software Engineering and Knowledge Engineering*, Banff, Alberta, 167-172.
- Wu, C., Lee, G. C., & Lai, H. (2004). Using Concept Maps to Aid Analysis of Concept Presentation in High School Computer Textbooks. *Education and Information Technologies*, 9 (2), 185-197.