Asia Pacific Journal of Information Systems
Vol. 25, No. 1, March 2015 http://dx.doi.org/10.14329/apjis.2015.25.1.025

# The Use of Ontology in Knowledge Intensive Tasks: Ontology Driven Retrieval of Use Cases*

Jongwoo Kim**, Jordi Conesa***, Balasubramaniam Ramesh****

Use cases are commonly used to represent customer requirements during systems development. In a large software development environment, finding relevant use cases from a library of past or related projects is a complex, error-prone and expensive task. This study proposes an ontological methodology to support use case retrieval in an interactive manner. The architecture of a prototype system that implements this methodology is presented. To evaluate whether the proposed approach can provide satisfactory results to users, this study develops a research model and hypotheses based on interaction theory. These hypotheses are empirically tested using a laboratory experiment which controls information filtering and perceived interaction. Our study suggests that a system which interacts with a user intelligently reduces cognitive load and increases self-efficacy and satisfaction.

Keywords : Ontology, Use Case, Perceived Interaction, Cognitive Load, Self-Efficacy

* This research was partly funded by the Spanish Government through the project TIN2013-45303-P "ICT-FLAG" (Enhancing ICT education through Formative assessment, Learning Analytics and Gamification).
** Corresponding Author, Assistant Professor, University of Massachusetts Boston, USA
*** Assistant Professor, Universitat Oberta de Catalunya, Spain
**** Board of Advisors Professor, Georgia State University, USA

# Ⅰ. Introduction

Requirements engineering (RE), which is concerned with acquiring and analyzing customer requirements, is a critical activity in software development. Use cases are gaining popularity in RE to represent customer requirements due to their simplicity and the use of natural language that facilitates the interaction between analysts and customers. In complex and large-scale systems development, the quantity of use cases that are needed to fully specify customer requirements can grow tremendously. Therefore, the ability to manage the development and reuse of use cases will significantly enhance project success and user satisfaction. Specifically, appropriate techniques for the development, reuse, and modification of use cases can tremendously enhance productivity and user satisfaction.

The use of natural language in use cases, while arguably the primary reason for their popularity, also poses interesting challenges. Use cases expressed in natural language inherit some of the major problems with natural language based specifications; i.e. they are more likely to be inherently imprecise, ambiguous, incomplete and inconsistent. Therefore, approaches that accurately capture the meaning of use cases will significantly improve the ability to manage them. Our study uses an ontological approach to improving the retrieval of use cases thereby increasing user satisfaction. This approach can enhance the ability to retrieve use cases that are relevant to a current project from a repository developed in past and similar projects.

It is well established in the literature on software reuse that significant benefits in software development productivity can be gained by reusing artifacts developed early in the lifecycle rather than late in the lifecycle. We suggest that the reuse of use cases provides such an opportunity for significant savings. Therefore, we focus on improving the ability to retrieve use cases from a repository, which is a critical step in facilitating their reuse. Current Computer Aided Software Engineering (CASE) tools provide only keyword based search capability to retrieve use cases. In contrast, our research proposes an approach which draws from concepts used in the development of a Semantic Web.

The Semantic Web [Berners-Lee *et al*., 2001] refers to the Internet of the future. The goal of the Semantic Web is to provide information with a well-defined meaning for machine-to-machine as well as machine-to-human communication. The Semantic Web approach uses ontologies to achieve this goal. The term of ontologies is borrowed from philosophy, where ontology refers to a systematic account of existence [Gruber, 1993]. By adding a well-defined meaning with ontological information, the Semantic Web promises to provide better and meaningful search for information to both machines and humans.

The goal of our research is to apply the Semantic Web approach to improve the ability to retrieve relevant use cases in response to user queries. The creation and management of use cases expressed in natural language is often a difficult task in system development. We posit that ontological information will help overcome problems caused by the use of natural language in their specification and provide satisfactory search results to users. Prior study has focused on the improvement of task performance with limited research on how user satisfaction can be enhanced [e.g., Qin *et al.*, 2010; Vulić *et al.*, 2013]. Our study attempts to fill this research gap. Thus, the primary research question addressed in this research is "*How does the use of ontologies improve satisfaction in complex knowledge intensive organizational tasks such as use case retrieval IS development?*"

To overcome the problems inherent in complex tasks such as the retrieval of use cases specified in natural language, prior research in RE has proposed several approaches. Sutton [2000] proposes mutual learning that results from frequent and close interaction through prototyping and rapid application development. Park *et al.* [2000] argue that syntactic parsing may be used to analyze requirements more accurately. To improve requirement verification and validation, some researchers suggest the use of restricted formal expressions for requirements [Fuchs and Schwertel, 2003; Marcia and Pulman, 1995]. Also, recently researchers have used an ontological approach in requirement engineering to improve requirements analysis [Kaiya and Saeki, 2005]. They found that the ontological approach was useful in detecting incompleteness and inconsistency in requirements specification, measuring the quality of a specification and predicting requirements changes. While Kaiya and Saeki [2005] are concerned with the analysis of informal specifications, our approach uses an ontology to improve queries used to retrieve use cases, and is novel in the area of requirements engineering.

This paper is structured as follows. The next section discusses related research. The proposed ontological approach is introduced in section 3. Section 4 discusses the development of queries based on an ontology. The architecture for a system that implements the proposed approach is presented in section 5. The design of an experiment that evaluates the effectiveness of the proposed approach to increase user satisfaction is introduced in section 6. The research model used in the experimental study is discussed in section 7. The results of experiment are presented in sections 8 and 9. Conclusions and future work are discussed in section 10.

## Ⅱ. Related Research

Ontology is defined as the explicit specification of a conceptualization [Gruber, 1993], which is an abstract and simplified view of the world we want to represent. Obviously, due to the wide variations in "the world we want to represent," there exists a wide range of different kinds of ontologies [Guarino, 1998; Lassila and McGuinness, 2001; Poli, 2002]. A general ontology contains general information rather than specifics relevant to a particular context. General ontologies tend to contain information, such as time or space, that is independent of any domain, as well as general information dependent on a particular domain. Whereas general information is useful in common sense reasoning, domain information provides semantics that will be helpful in understanding that domain.

The creation of use cases is often the first step in the acquisition of requirements from users. Its role as an effective communication vehicle to capture requirements from users is a reason for its increasing popularity. Designers develop system design artifacts like state transition diagrams and class diagrams on the basis of use cases. Thus, use cases often represent a critical starting point in the development lifecycle. When stakeholders need to examine the relationships between the actual implementation and system requirements, they may rely on use cases that document requirements. Thus, from a RE perspective, a use case is a key artifact that is created and used throughout the processes of systems development.

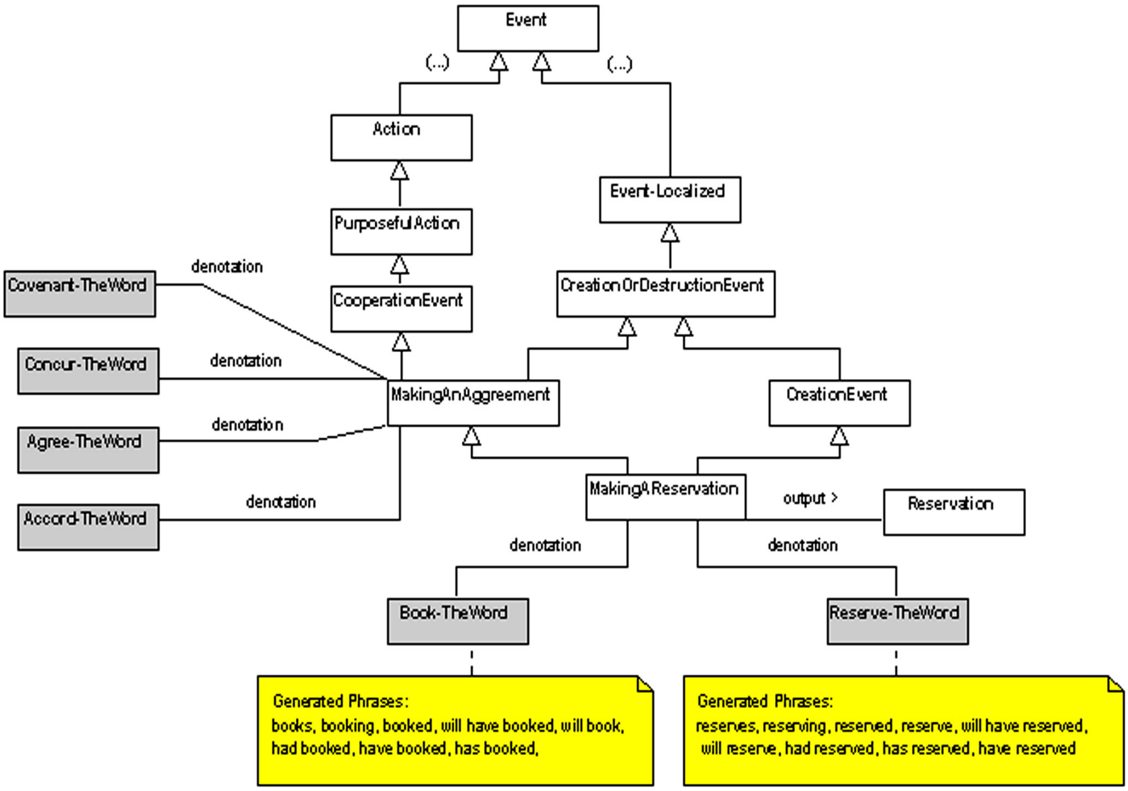
A common task in RE is the search and exploration of requirements (which may be documented as use cases) that were created in earlier phases of a project or in other similar projects. These activities are supported only to a limited extent by current RE and CASE tools. These tools typically provide keyword-based search capabilities similar to those used in web search engines like Yahoo or Google. Typically, such searches are not very helpful because of the ambiguities inherent in natural language. To overcome this problem, ontology based searching has been suggested by prior research. For example, Storey *et al*. [2008] proposed a context-aware query processing methodology called CONQUER which uses lexicons and ontologies to improve web query results. Researchers argue that real progress can be made in Software and Systems engineering when this approach is integrated with popular notations such as UML. Saeki [2004] introduces an ontology-based technique to support software requirements elicitation and to compose software from reusable architectures, frameworks, components and software packages. Their work focuses on semantic processing of requirements and reusable artifacts. Kaiya and Saeki [2005] use an ontology that consists of a thesaurus and a set of inference rules to detect incompleteness and inconsistency, measure the quality of a specification, and predict requirements changes. Although prior research has tried to address issues related to resolving ambiguities in requirements by applying linguistic techniques to use case analysis, limited attention has been paid to the use of domain knowledge in these activities [Fantechi *et al*., 2002; Fantechi *et al*., 2003]. Our work addresses this research gap by using a domain ontology to support the retrieval and reuse of use cases.

## Ⅲ. The Use of an Ontological Approach to Use Cases Query

Our approach uses a combination of linguistic, semantic and extensional knowledge to improve the queries of use cases. We use ResearchCyc ontology in this research because it appears to be the only ontology that contains linguistic, semantic and extensional knowledge. ReserchCyc, which is a complete version of the Cyc knowledgebase (Cyc) for the scientific community, contains more than 2 million assertions (facts and rules) describing more than 250,000 terms and including nearly 15,000 predicates [Matuszek *et al*., 2006]. The quantity and quality of the information it contains about actions are superior to those of other ontologies. For example, ResearchCyc contains a taxonomy of more than 6,000 actions. Since use cases often specify actions that are supported by a system, ResearchCyc is an excellent candidate for supporting the creation and use of use cases. Linguistic information (such as synonyms) is used to deal with some ambiguities of the natural language. Semantic information is used to develop intelligent queries, as can be seen in the following example: Suppose a designer is interested in use case diagrams that describe the rental of a GPS in a system for making reservations for a rental car. If the repository of specifications does not have a specific use case for renting a GPS, a use case that explains how to make a car reservation is likely to be of interest to the designer. Current ontological approaches that deal with requirements use a thesaurus to support the query process. Since a thesaurus does not contain semantic or extensional information, advanced inferences cannot be made with such approaches. On the other hand, semantic information present in general ontologies may provide a more powerful ability in the retrieval of use cases.

## 3.1 ACTION: The Actions Ontology

We create Actions Ontology (ACTION) from the ResearchCyc ontology by selecting all the sub types of a concept that represents events. An event concept that represents actions is of interest in our research. In ResearchCyc, an event is defined as the dynamic situation in which the state of the world changes. We create ACTION ontology by including concepts that are related to events or any of its subtypes through relationship types. The rest of concepts contained in the ResearchCyc ontology have been deleted using a pruning algorithm [Conesa and Olivé, 2006]. The taxonomy of Events defined in the ontology contains more than six

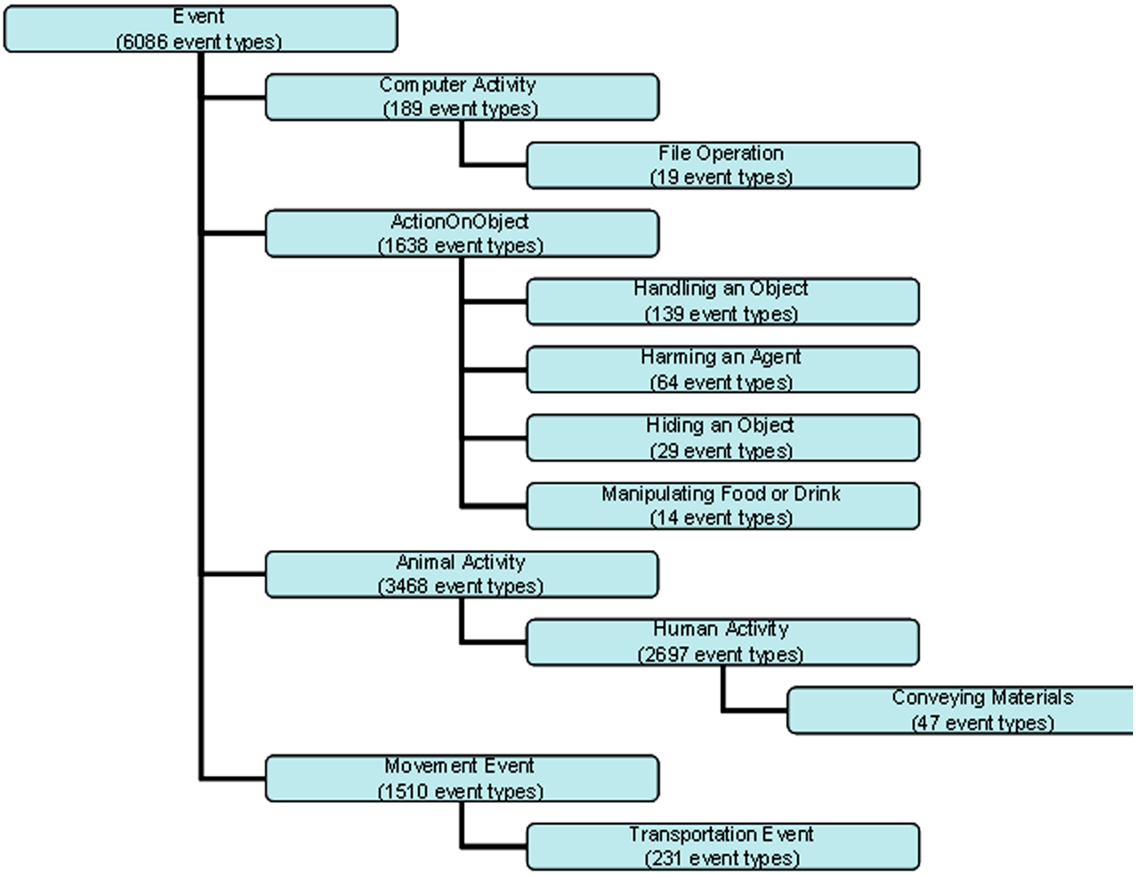



<Figure 1> Fragment of ACTION Ontology (The Gray Classes Denote Lexical Information)

thousand different kinds of events that cover most of the actions that are commonly supported by information systems. In addition, Aaction also contains lexical information. In particular, it contains words that denote each of the actions. <Figure 1> shows a fragment of the action ontology. Concepts in grey represent the linguistic information related to the events in the ontology. For example, Making A Reservation event is related to the following words: book and reserve. The ontology also contains the different ways (conjugations) in which those words may be used in a text or a query. For example, the word book may be written as "books, booking, booked, will have booked…". ACTION covers information on many domains (e.g. rental processes, terrorist actions, piracy actions etc.). <Figure 2> shows some of the most important topics ACTION covers and the number of subtypes associated with each topic.

### 3.2 The Domain Ontology

Even though ACTION can be used to retrieve use cases, sometimes it may not provide sufficient information to generate a satisfactory answer to a users' query. For example, suppose the designer searches a use case repository with




<Figure 2> The Main Event Types of ACTIONS and the Number of Event Types

a query: How do I rent a GPS? As mentioned earlier, the system can infer that the use case "Make a Reservation" is relevant for renting a GPS. To perform this inference, the system needs to know that in the context of this query, the GPS is a part of a car. Since ResearchCyc does not contain the information that a GPS may be part of a Car, the use of a domain ontology focused on automobiles is necessary. Thus, the use of a domain ontology (or ontologies) relevant to the domains that the use cases deal with will be very useful in improving the inferences that are necessary for effective RE.

### 3.3 WordNet

WordNet is being maintained as a semantic lexicon for the English language. It groups English words into sets of synonyms with short, general definitions. In addition, it records the various semantic relations between these synonym sets. The purpose of WordNet is twofold. One is to produce a combination of dictionary and thesaurus that is more intuitively usable. The other is to support automatic text analysis and artificial intelligence applications. In our study, WordNet can be used to provide lexical information, which may not be available in ACTION and domain ontology.

## Ⅳ. Ontology Driven Requirements Query

The proposed methodology uses semantic and linguistic knowledge to identify the use cases that fit the query of the user. Therefore, any web query methodology that deals with linguistic and semantic knowledge can be adapted for this study. The adaptation only requires replacing the web query engine with a use case query engine. In the following sections, we describe a query methodology and demonstrate the improvements that can be achieved with the use of ontologies. The methodology is similar in spirit to the approach used by a web query methodology presented in [Conesa *et al*., 2006]. The methodology is distinctive in that the interaction between user and the system is integral process in refining original queries.

The proposed query methodology is composed of four phases:

### 4.1 Query Parsing Phase

This phase receives a query expressed in natural language by the user. The nouns, noun phrases and verbs are identified from the initial query using a POS Tagger [Mason Accessed on Jan 7, 2005]. The output of this phase is a set of query terms $w_1, \cdots, w_n$ that are used as the initial query. POS tags of each word are used to identify the most suitable terms in both the domain and the ACTION ontology. For example, when a user makes a query: 'Identify when a resume matches a job offer', a POS tagger captures a word 'match' as a verb. This information is used to discard noun concepts (e.g., competitions, a match-a slender piece of wood to burn things) related to match.

### 4.2 Concepts Identification Phase

The input of this phase is a set of words related to the use cases that are of interest to the user. The purpose of this phase is to find the concepts of ACTION and domain ontologies that represent the words of the query, or are closely related to them.

We say that a concept C represents a query term $w_i$ if:

1. $w_i$ is part of the noun of the concept, or
2. There is any linguistic relationship between the word $w_i$ and the concept $C$ in the ontology (the relationship types of ACTION that define linguistic information are *termStrings*, *denotation* and *genStringAssertion*), or
3. a WordNet concept which is synonym of $w_i$ satisfy any of the two previous conditions.

The output of this phase is a set of the concepts that represent the candidate query words of the input $\{C_1, ..., C_m\}$.

### 4.3 Interaction Phase

The interface module receives the output of the concepts identification phase and presents it the user. The purpose of this phase is to allow the user to select relevant concepts in order to refine his/her query. The interface module sends the refined query to the inference module.

### 4.4 Query Inferences Phase

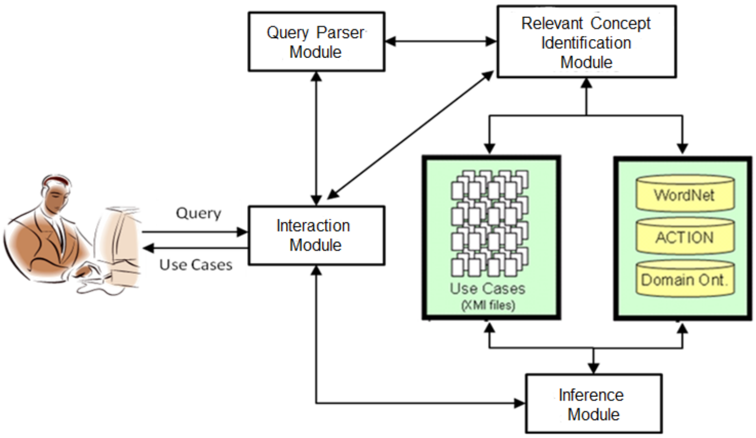
The input of this phase is a set of concepts provided by the interaction phase. The purpose of this phase is to select all the use cases relevant to the input concepts.

Suppose, in the context of the rental car information system, the user wants to find use cases that deal with the reservation of the 'child seats' and uses the query term "children car seat rental." The system looks up the concept in the ontology library. It identifies that 'reservation', 'automobile', and 'seat' are the relevant concepts. The user selects the concepts that may help refine the original query. Then, the system searches a library of use case and sends the search results to the interface module which presents the information to the user.

## Ⅴ. Prototype

### 5.1 Architecture

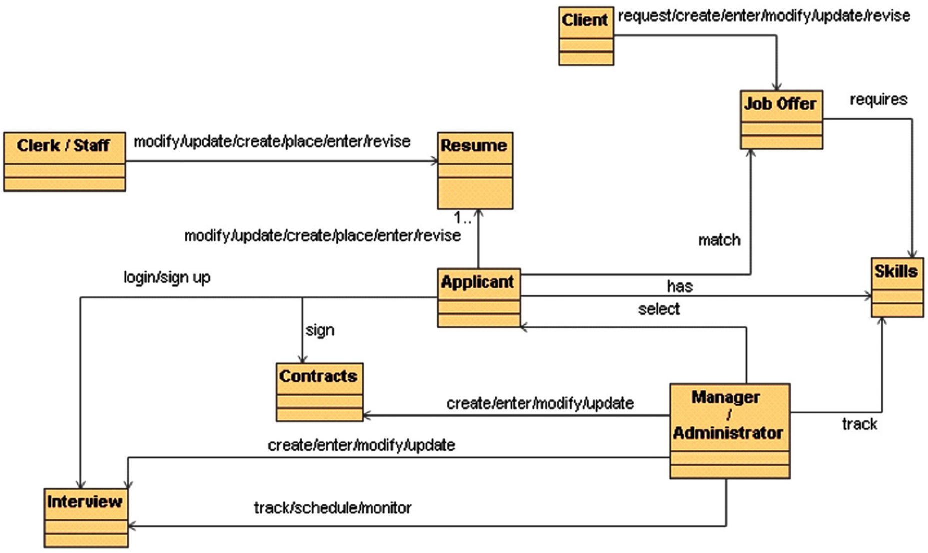
<Figure 3> shows the architecture of our prototype system for ontology-driven use case retrieval. It consists of: a) Query Parser Module, b) Relevant Concepts Identification Module, c) Interaction Module, and d) Inference Module. The Interaction Module enables the interaction between users and the system. Users can enter and revise their queries to search use cases. The Query Parser Module captures the user's query and parses it to return the part-of-speech for each term. A POS tagger marks up words in users' queries. The Relevant Concepts Identification Module interacts with ACTION, domain ontology (if necessary) and WordNet. For each query, it obtains related concepts from ACTION and domain ontologies. To obtain this information, linguistic relationships such as synonyms and antonyms can be used. The output of this module is sent to the Interaction Module. Users can refine their queries based on the results via the Interaction Module. The Inference Module receives the selected concepts and finds relevant use cases. Finally, the Interaction Module will present selected use cases to the user.




<Figure 3> System Architecture

## 5.2 Domain Ontology

To illustrate the proposed approach, a domain ontology in the domain of job search was created. <Figure 4> shows a simplified conceptual representation of the ontology. The domain ontology identifies relationships among key terms and synonyms. These relationships are used to identify terms that are semantically related terms used in a user query. For example, several terms including resume, contracts, and job offer are directly related to applicant. Multiple terms in a class or a property indicate that the terms have synonyms. For example, track, schedule, and monitor can be considered synonyms. When a user's key term has related terms and synonyms, these are displayed to the user (see <Figure 5>).

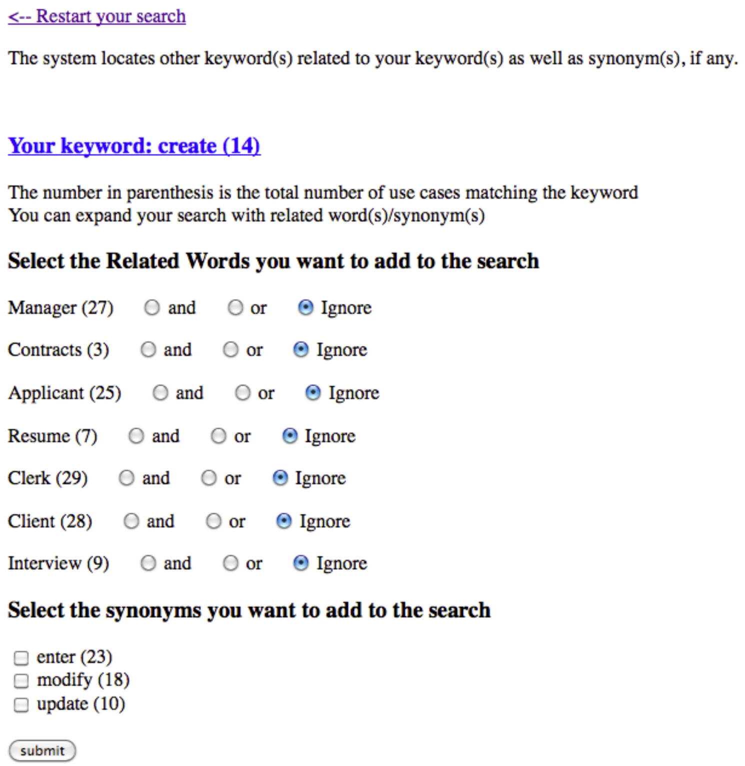



<Figure 4> Conceptual Representation of Domain Ontology

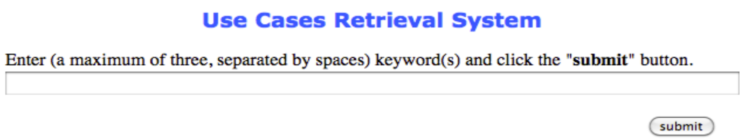
### 5.3 Implementation

The prototype is implemented as a web application using JSP (Java Server Pages). This development environment was chosen because it would make the system portable and easily accessible through the World Wide Web. The web application interacted with the MySql database of Use Cases and the domain ontology in OWL web ontology language [W3C, 2004].

On the client side, web pages were used to gather information from a user, such as an initial query keyword and relevant keywords. On the server side, several java servlet modules are used to parse multi keywords query and identify both the elements of the ontology and the synonyms related to the initial query. For the interaction with the domain ontology, the prototype uses the OWL API (http://owlapi.sourceforge.net/index.html), which is an open source Java tool that is used to read the domain ontology in OWL. The interaction with the ontology is shown at <Figure 5> and <Figure 6>.




<Figure 5> Ontology Supported Interactive Prototype System



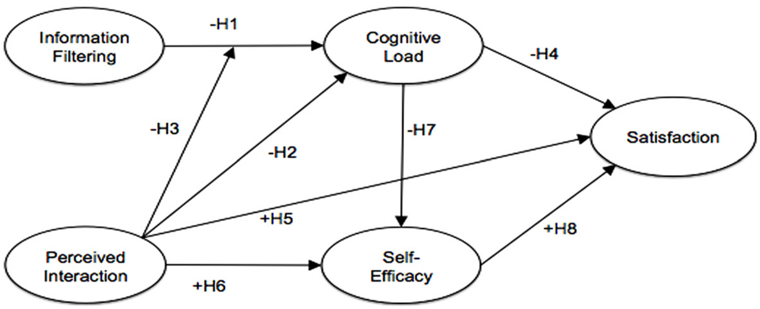

<Figure 6> User Interface of the Ontology Supported Interactive Prototype System

# Ⅵ. Empirical Evaluation of The Effectivness of The Proposed Approach

The effectiveness of the proposed approach is evaluated using an experimental study based on cognitive load and self-efficacy theory. In evaluating our prototype system, we focused on answering our research question: "How does the use of ontologies improve satisfaction in complex knowledge intensive organizational tasks such as use case retrieval IS development?" We developed and empirically tested a theoretical research model by following a design science method. Hevner *et al*. [2004] notes that a design science research can be rigorous and relevant by developing and selecting appropriate means to evaluate an artifact. Our prototype is a component of a human-machine problem-solving system as it helps users improve a retrieval of use cases by making the prototype interactive. Thus, developing and empirically testing a theoretical model based on behavioral theories is a relevant and important part of our study [Hevner *et al*., 2004]. In this experiment, information filtering and perceived interaction are used as exogenous constructs whereas cognitive load, self-efficacy, and satisfaction are identified as endogenous constructs. A research model for this study is shown in <Figure 7>.

## 6.1 Cognitive Load

Cognitive load refers to the load on the working memory during problem solving, thinking and reasoning. All information necessary for performing the task must be processed before meaningful learning and problem solving can continue. Psychology researchers classified cognitive load into three categories: intrinsic, extraneous, and germane cognitive load [Paas *et al*., 2003; Sweller, 1988; Sweller and Chandler, 1994]. Intrinsic load is the inherent level of difficulty related to instructional materials. Extraneous load refers to the load generated by the manner in which information is presented to learners. Germane load is the load imposed by the learning activities (e.g. processing, construction, and automation of mental schema) required of learners. Information filtering may reduce intrinsic cognitive load because the level of intrinsic cognitive load depends on the nature of the material to be learned and the amount of information processing needed. In other words, the higher the number of elements that must be processed simultaneously by a user of an ontology, higher the level of intrinsic cognitive load [van Merriëboer and Sweller, 2005].




<Figure 7> Research Model

Information filtering refers to the process of determining which part of the information has the best chance of matching the user's information needs [Belkin and Croft, 1992]. Malone *et al*. [1987] note that information filtering is a method for the delivery of relevant information. Information filtering and information retrieval are similar in that users collect information as a result of information processing. However they are different as information filtering involves the process of selecting relevant information from a large set of possibilities and removing irrelevant information, while information retrieval involves the process of finding information [Foltz and Dumais ,1992]. Malone *et al.* [1987] suggest that three different approaches to information filtering can be used for an automated information filtering system. These approaches include cognitive filtering, social filtering, and economic filtering. Cognitive filtering characterizes the content of data and the information needs of users for the intelligent matching of data to users. Simple keyword based matching is a form of cognitive filtering. Social filtering works by focusing on the interrelationships of individuals in a community. It emphasizes the characteristics of data senders as well as the topic. Economic filtering focuses on the cost-benefit assessments and pricing mechanisms of filtering. In our evaluation, we subscribe to cognitive filtering because we use keywords chosen by users.

Due to the limited capacity that humans possess to store current information in memory, people tend to generate and respond to simpler information when they are overloaded with large amounts of information [Jones *et al*., 2004]. The problem is even more pronounced when unnecessary information is included in a large volume of information presented to the user. Relevant information can be retrieved after irrelevant information is filtered out. When users are provided the ability to receive only information relevant to a given context and discard irrelevant information, the cognitive load involved in information processing may be reduced. The processing of unfiltered information increases the cognitive load faced by users. Therefore, in order to reduce the cognitive load, it is important to reduce the amount of information presented to the user.

Information filtering can reduce cognitive load by selecting relevant parts from a larger set of information and presenting it in a prioritized order [Malone *et al*., 1987]. Billinghurst *et al*. [1999] note that information systems can be used to filter out unnecessary information thereby reducing cognitive load. Ontology can play an important role in information filtering in tasks such as information retrieval. Users may perceive a high cognitive load while identifying relevant search terms for use in information retrieval tasks. Often, users need to identify several keywords and select among them for effective information retrieval. This keyword generation and selection processes involve high cognitive load.

Ontology has been recognized as an effective tool in improving information retrieval [Klischewski, 2006]. The use of ontology helps overcome the limitations of keyword-based search by providing the ability to represent class hierarchies and relationships. The additional representational power provided by ontology can significantly improve the identification of keywords by expanding the queries with related, relevant terms.

Prior literature on cognitive load and information filtering suggests that Information filtering will reduce cognitive load. Thus, we hypothesize

*Hypothesis 1: Information filtering will negatively affect cognitive load.*

## 6.2 Perceived Interaction

Interaction is an action that occurs as two or more objects have an effect upon one another. In human-computer interaction, the interaction between users and computers occurs at the user interface which may be implemented in software or hardware. Interaction in information retrieval systems occurs when users communicate with computers by specifying queries, receiving results, and revising queries (say, by experimenting with different keywords). Prior research establishes that poorly designed human-computer interfaces can lead to unexpected problems such as misinterpretation of information. Therefore, much of the research in this area has focused on the design of better human-computer interfaces. However, recent studies note that while research that addresses procedural or functional aspects of interaction is important, more research is needed to address perceptional aspects of interaction. McMillan and Hwang [2002] identify three dimensions of perceived interaction: conversation, delay, and engaging. Our study subscribes to this new perspective on interaction. Perceived interaction rather than interaction as a feature of the system is the focus of this study.

Perceived interaction can affect cognitive load perceived by uses in information retrieval tasks. The high cognitive load associated with the generation and refinement of keywords in information retrieval can be reduced by the ability to perform these tasks interactively. In the absence of interaction, users do not receive any feedback on the appropriateness of the keywords they use for information retrieval. When the users can interactively perform these tasks, they may be able to successively refine their keywords and successfully complete their information retrieval tasks. Thus,

*Hypothesis 2: Perceived interaction will negatively affect cognitive load.*

The quality of the perceived interaction in information retrieval tasks can be significantly improved with the use of ontology. With the use of relevant knowledge provided by an ontology, users may quickly identify and refine keywords used, thereby reducing the cognitive load involved in these tasks. For example, by interacting with information systems, users may learn that car and automobile are synonymous. They can retrieve more relevant information by using both terms instead of a single term. While the development of algorithms and performance measures to improve information retrieval have received much attention in prior research, the use of ontology to improve the quality of results in information retrieval tasks hasn't been examined adequately. Recognizing this gap, Storey *et al*. [2008] propose CONQUER, a methodology for context-aware query development. This research establishes the use of interaction and ontology in facilitating information retrieval. Through the interaction between the user and an ontology-supported retrieval system, users can improve their search queries. In similar spirit, Vallet *et al*. [2005] adapt a vector-based ranking model, which takes advantage of an ontology to help users interactively improve their queries. These studies suggest that interaction moderates the relationship between information filtering and cognitive load.

These studies suggest that higher levels of perceived interaction accelerate the reduction of cognitive load when coupled with higher levels of information filtering. Thus,

*Hypothesis 3: Perceived interaction moderates the effects of information filtering on cognitive load.*

## 6.3 Satisfaction

Satisfaction is defined as 'a judgment that a service provided a pleasurable level of consumption-related fulfillment' [Oliver, 1996]. Thus, satisfaction is the user's sense that consumption provides outcomes against a standard of pleasure versus displeasure. Anderson and Sullivan [1993] postulate that satisfaction can be "broadly characterized as a post-purchase evaluation of product quality given pre- purchase expectations." Past research suggests that satisfaction is influenced by perceived performance of a product or a service [Cronin and Taylor, 1994]. Therefore, perceived quality and satisfaction need to be separated because these are different at a conceptual level [Kettinger and Lee, 1994]. In this research, satisfaction is considered stemming from transaction experiences [Parasuraman *et al*., 1994].

Prior studies on information retrieval suggest that a trade-off relationship between cognitive load and user satisfaction exists [Branting, 2001; McSherry, 2004]. A study on consumer behavior finds that cognitive load decreases the tendency to choose the better quality option thereby reducing user satisfaction [Drolet and Frances Luce, 2004]. Back and Oppenheim [2001] assume that low cognitive load from a user-friendly interface of an information retrieval system can result in high user satisfaction.

Based on prior literature on cognitive load and information satisfaction, this study predicts that cognitive load will reduce satisfaction. Thus,

*Hypothesis 4: Cognitive load will negatively affect satisfaction.*

Prior research suggests that user interaction with the system has an impact on a participant's satisfaction [Lamport, 1993]. Driver [2002] argues that interaction stimulated by online discussions may effectively enhance students' class experience and increase their satisfaction. Wells *et al*. [1999] note that information technology can support one-to-one customer interaction thereby increasing customer satisfaction.

Cognitive feedback theory provides a theoretical explanation of the relationship between perceived interaction and satisfaction. Sengupta and Te'eni [1993] define cognitive feedback as information about the decision maker's decision strategy and the extent to which the strategy is applied accurately. While outcome feedback describes the accuracy of a decision, cognitive feedback provides decision makers with insight into their decision processes [Balzer *et al*., 1989]. Inter ACTIONS between users and the system with the support of ontological knowledge can provide cognitive feedback, which would be increase user satisfaction. Thus, *Hypothesis 5: Perceived interaction will positively affect satisfaction.*

## 6.4 Self-Efficacy

Self-efficacy refers to one's belief in one's capability to perform a specific task [Bandura and Adams, 1977]. Social cognitive theory posits that neither inner forces nor external stimuli drive people to exhibit a certain behavior. The theory explains that human behavior, cognitive and personal factors, and environmental events all operate interactively with one another. Self-efficacy, defined as a person's judgment of his/her capabilities to perform a given task, is a key regulatory mechanism in this relationship. Bandura [1982] postulates that self-efficacy helps determine what ACTIONS to take, how much effort to invest, how long to persevere and what strategies to use in challenging situations. Prior studies support this proposition in a variety of settings such as technology acceptance [Agarwal *et al*., 2000; Venkatesh, 2000], computer skill acquisition [Mitchell *et al*., 1994], and complex decision making [Wood and Bandura, 1989].

Cognitive feedback theory suggests that feedback supported by information systems increases decision quality and confidence in individual decision making [Hogarth, 1996]. When the user provides general and ambiguous terms during information retrieval system, the system may generate irrelevant results. In contrast, with an interactive system, user can refine their queries based on the feedback provided by intermediate results. Thus, perceived interaction increases the user's confidence in their capability or their self-efficacy. Thus,

*Hypothesis 6: Perceived interaction will positively affect self-efficacy.*

When people encounter difficult tasks, which require a significant effort and time to complete, they might lose their belief in their abilities to cope with those tasks. Prior research on e-learning environments identifies several variables as motivators of students. These include perceived importance, usefulness, and the value of engaging in a task [Pintrich and Schrauben, 1992]. When learners perceive the effort as a waste of energy or as unnecessary, they are not motivated to exert sufficient mental effort. Another important variable affecting a person's motivation to take challenging tasks is his/her preconceptions about the effort required to accomplish a task. Keller and Suzuki [2004] note that self-efficacy is an important component of motivation. Therefore, preconceptions on the effort required to complete a task affect not only motivation but also some characteristics of the learner such as self-efficacy.

Perceived cognitive load (e.g. perceived difficulty and complexity) about a task can affect people's self-efficacy. When people perceive a given task as very difficult and complex, they assume that the task requires a lot of effort and time. Then, people question whether they have the ability to invest such effort and time required for the task.

Problem solving requires cognitive effort and places a certain amount of load on working memory processes [Wood *et al*., 2000]. For example, people who are good at puzzles or mathematics are likely to have high self-efficacy. They can handle tasks which require high cognitive load. This suggests that people with high self-efficacy can handle tasks requiring objectively high cognitive load. When people subjectively perceive low cognitive load from tasks, they believe they have the ability to successfully accomplish the tasks. Thus,

*Hypothesis 7: Cognitive load will negatively affect self-efficacy.*

Computer self-efficacy (CSE) is a two-level construct which operates at the general computing level (general CSE) and at the specific application level (application specific CSE). General CSE refers to an individual judgment of efficacy across multiple computer domains whereas application specific CSE refers to an individual perception of efficacy in using a specific application or system within the domain of general computing. Prior research on technology acceptance focuses on the effects of general CSE on users' attitude to a system [Venkatesh and Davis, 1996]. Agarwal *et al.* [2000] proposed a model which differentiated a general CSE and an application specific CSE. They empirically established that an application specific CSE has statistically more significant effect on users' attitude toward a system use and adoption. Our study subscribes to this view.

User satisfaction is considered as one of the most important measures of information systems success [Delone and McLean, 1992]. In particular, users with high-level of a general CSE perceive high satisfaction on the use of an information system. Prior research suggests that self-efficacy has a positive relationship with user satisfaction [Henry and Stone, 1994]. Based on prior research on self-efficacy and satisfaction, we hypothesize

*Hypothesis 8: Self-efficacy is positively associated with satisfaction.*

# Ⅶ. Research Method And Design

## 7.1 Treatment, Task, and Prototypes

A laboratory experiment was conducted to test the causal relationships between the constructs in the research model. The experiment involves a two-factor, four-cell design with two exogenous variables: information filtering and interaction. Both information filtering and interaction were manipulated at two levels. However, perceived interaction was measured by multiple items developed by McMillan and Hwang [2002]. All subjects in four cells were asked to find two relevant use cases in a given context. After the study instruments were developed, pilot tests were conducted to refine the treatments and validate the measures. Students taking courses in information systems at two large southeastern and northeastern universities in U.S. served as subjects. A total of 121 subjects participated this experiment of which 99 passed manipulation checks. Their mean experience with system analysis was 16 months. Sixty four percent of the subjects were male, and thirty six percent were female.

Subjects were randomly assigned to one of the four cells on the basis of the last digit of their birthday. The experimental task involved the retrieval of two use cases from a library of use cases.

Four prototypes were developed to provide two levels each of information filtering and interaction as shown at <Table 1>. The *first* prototype provided no interaction and no informa-

<Table 1> Group Design and Four Prototypes

| | | Information Filtering (n = number of participants) | |
|---|---|---|---|
| | | No | Yes |
| Interaction | No | Prototype I (n = 26) | Prototype III (n = 18) |
| | Yes | Prototype II (n = 23) | Prototype IV (n = 32) |

tion filtering. Subjects used keywords to retrieve use cases relevant to their given task. To minimize the interaction between subjects and the system, subjects were allowed to use the retrieval system only one time. In addition, twenty seconds delay was added before the system delivers the result. The *second* prototype provided high levels of interaction and no information filtering. In this prototype, the subjects could return to home page and enter additional keywords. The *third* prototype supported no interaction, but provided information filtering with the use of an ontology. When a user enters a keyword, the retrieval system identified other related keywords and showed those keywords and matching use cases. In addition, twenty seconds delay was added before the system delivers the result. Finally the *fourth* prototype supported high levels of both interaction and information filtering. When users enter a keyword, the system suggests additional terms and synonyms. Subjects can choose to include these terms with the use of conjunctions or disjunctions (see Appendix A for screenshots of the four prototypes).

After finishing the task of finding relevant two use cases, subjects were asked to complete a questionnaire. The questions consisted of items that measured cognitive load, satisfaction, perceived interaction, and self-efficacy with the system.

## 7.2 Measures

Information filtering was measured as a dichotomous variable (0 or 1). Perceived interaction was measured by multi-item measures along three dimensions of perceived interaction (i.e. conversation, delay, and engagement) [McMillan and Hwang 2002].

Multi-item measures for cognitive load were used for this study [Paas, 1992; Sweller and Chandler, 1994]. Cognitive load can be measured as a subjective variable. This study assumes that people are able to introspect their cognitive processes and report the amount of perceived cognitive load. Prior research has demonstrated that people are quite capable of assessing their perceived mental burden involved in performing a task [Gopher and Braune, 1984].

Multi-item measures for satisfaction were used for this study. Wixom and Todd [2005] propose measurements for information satisfaction and system satisfaction. This study uses measurement items for system satisfactions. Finally self-efficacy was measured by three items adapted from prior research [Johnson and Marakas, 2000; Yi and Hwang, 2003].

All measurement scales were validated through a pilot test. Items for perceived interaction were anchored on a seven-point likert scale ranging from “not at all descriptive” (1) to “very descriptive” (7). Items for cognitive load were anchored on a seven-point likert scale ranging from "very little" (1) to "very much" (7). Items for self-efficacy were anchored on a eleven-point likert scale ranging from "completely disagree" (0) to "completely agree"(10). Items for satisfaction were anchored on a seven-point likert scale ranging from "strong disagree" (1) to "strongly agree" (7). Based on the results of the pilot study, minor modifications were made to the survey design. The final survey included 22 items representing the four constructs identified in <Figure 7>. Appendix B shows the measures used in the study.

# Ⅷ. Results

## 8.1 Manipulation Checks

Manipulation checks were employed to ensure that the subjects received the intended treatments for information filtering and interaction. Subjects were asked to report whether they used system that filtered information in completing their task. Subjects who received interaction treatment were asked whether they perceived the system to be interactive. A total of 99 cases that passed the manipulation checks were retained for subsequent analysis.

## 8.2 Partial Least Squares Analysis

Partial Least Squares (PLS)-a second-generation structural equation modeling technique-was used to evaluate the adequacy of the measurement model and then to test the hypothesized structural model [Chin, 1998b]. Three considerations motivated the choice of PLS.

First, PLS' ability to handle items with different scales is superior to multiple regression and traditional path-analytic techniques. PLS is considered robust at handling data with different scale types. This study uses measurement items with different scales Information filtering was included as dichotomous in the model. This categorical variable was coded as 0 or 1 in PLS analysis whereas other variables were measured differently as mentioned at the section 7.2.

Second, PLS analysis is considered suitable for testing a theoretical model in its early stages. Since this study is an initial attempt at empirical examination of the impact of information filtering and interaction on satisfaction through cognitive load and self-efficacy in the context of ontology supported information filtering, the use of PLS analysis is appropriate.

Third, PLS is appropriate for testing a model which has both reflective and formative constructs [Chin, 1998a; Rai *et al*., 2002; Ringle *et al*., 2012]. A formative construct consists of a composite of multiple measures. Perceived interaction in our model is treated as a formative construct.

SmartPLS version 2.0.M3 was used for the analysis, and the bootstrap resampling method (with 500 resamples) was used to determine the significance of the paths within the structural model.

## 8.3 Measurement Model Assessment

Our research model has both reflective and formative constructs. This affected the manner in which convergent validity was assessed as we made no assumption that formative indicators will covary. Therefore, traditional methods for assessing construct reliability cannot be applied to formative constructs. Multicollinearity was examined for a formative construct (interaction). A variance inflation factor (VIF) value of interaction was calculated. Prior research recommends that VIF values for formative measures should be less than 10. Since all VIF values (shown in <Table 2>) are less than 10, there is minimal risk of multicollinearity and all items for perceived interaction were retained to preserve content validity.

We assessed whether the scales exhibit sufficient convergent and discriminant validity. Standardized loadings were examined to test convergent validity of constructs used in this research. Standardized loadings should be higher than 0.707 to meet the condition that the shared variance between each measurement item and its latent construct exceed the error variance. As seen in <Table 3>, loadings of items for all constructs were higher than 0.857. Therefore all items were retained in the analysis.

To test the internal consistency, Cronbach's alpha, composite reliability, and average variance extracted (AVE) of cognitive load were examined. Cronbach's alpha and composite reliability values are higher than 0.7, which is a recommended minimum value for reliability [Bearden *et al*., 1993; Yi and Davis, 2003]. As another measure of construct validity, AVE measures the amount of variance that a latent construct captures from its indicators relative to the amount of variance from measurement error [Fornell and Larcker, 1981]. According to Chin [1998], AVE of higher than 0.5 means that 50 percent or more variance of the indicators is accounted for and acceptable for analysis. AVE for all constructs in this study is higher than 0.769. Thus, convergent validity is established according to the evaluation of Cronbach's alpha, composite reliability, and AVE.

<Table 2> Variance Inflation Factor for Formative Construct

| Construct | Items | Variance Inflation Factor (VIF) |
|---|---|---|
| Perceived Interaction | CONV1 | 2.40 |
| | CONV2 | 2.89 |
| | CONV3 | 3.48 |
| | CONV4 | 1.64 |
| | DELAY1 | 4.10 |
| | DELAY2 | 1.94 |
| | DELAY3 | 4.50 |
| | ENG1 | 3.78 |
| | ENG2 | 2.63 |
| | ENG3 | 2.28 |
| | ENG4 | 2.14 |
| | ENG5 | 1.64 |
| | ENG6 | 2.83 |
| | ENG7 | 2.83 |

<Table 3> Item Loadings and Construct Measurement Properties

| Construct | Item | Standardized Loading | Cronbach's Alpha | Composite Reliability | Average Variance Extracted |
|---|---|---|---|---|---|
| Cognitive Load | CL1 | 0.893 | 0.909 | 0.943 | 0.847 |
| | CL2 | 0.949 | | | |
| | CL3 | 0.917 | | | |
| Self-efficacy | SE1 | 0.871 | 0.850 | 0.909 | 0.769 |
| | SE2 | 0.902 | | | |
| | SE3 | 0.857 | | | |
| Satisfaction | SA1 | 0.981 | 0.961 | 0.981 | 0.962 |
| | SA2 | 0.980 | | | |

To test discriminant validity, we conducted two tests. First, we compared AVE for each construct with the shared variance between all possible pairs of constructs [Fornell and Larcker, 1981]. AVE for each construct is higher than the squared correlation between the construct pairs. This means that more variance is shared between the latent construct and its block of indicators than with another construct representing a different block of indicators. Therefore discriminant validity is established as shown in <Table 4>.

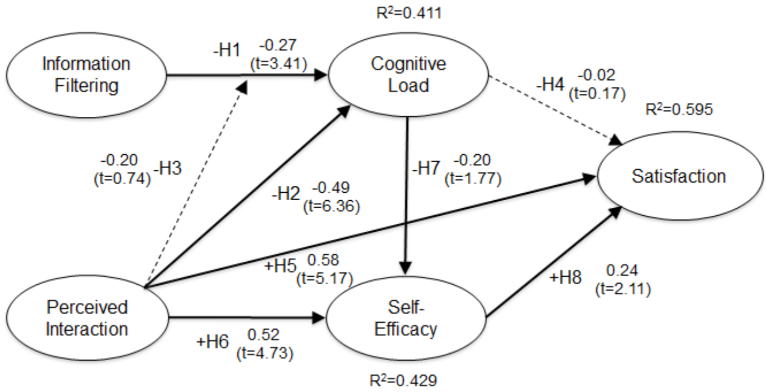
Second, we calculated each indicator's loading on its own construct and its cross-loading on all other constructs were calculated [Chin, 1998b]. Three sub-constructs of interaction (i.e. conversation, no delay, and engagement) were added. The loadings for the intended indicators for each construct are higher than the cross- loadings for indicators from other constructs as shown in <Table 5>. Two items for interaction (i.e. CONV4 and ENG5) were dropped from further analysis because they did not load high. Each indicator has a higher loading with its intended construct than a cross-loading with any other construct.

<Table 4> AVE versus Squares of Correlations between Constructs

| Construct | Average Variance Extracted (AVE) | CL | SE | SAT |
|---|---|---|---|---|
| CL | 0.847 | - | | |
| SE | 0.769 | 0.253 | - | |
| SAT | 0.962 | 0.232 | 0.386 | - |

<Table 5> Items to Own Construct Correlation versus Correlations with Other Constructs

| Construct | | Item | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|---|
| Cognitive Load | | CL1 | **0.893** | -0.440 | -0.259 | -0.501 | -0.432 | -0.420 |
| | | CL2 | **0.950** | -0.334 | -0.279 | -0.516 | -0.474 | -0.427 |
| | | CL3 | **0.917** | -0.436 | -0.341 | -0.594 | -0.482 | -0.481 |
| Perceived Interaction | Conversation | CONV1 | -0.351 | **0.860** | 0.433 | 0.556 | 0.526 | 0.484 |
| | | CONV2 | -0.338 | **0.894** | 0.381 | 0.509 | 0.448 | 0.415 |
| | | CONV3 | -0.445 | **0.881** | 0.501 | 0.675 | 0.405 | 0.532 |
| | | CONV4 | -0.234 | **0.442** | 0.023 | 0.209 | 0.076 | 0.069 |
| | No-Delay | DELAY1 | -0.313 | 0.408 | **0.915** | 0.644 | 0.455 | 0.486 |
| | | DELAY2 | -0.211 | 0.352 | **0.798** | 0.525 | 0.413 | 0.286 |
| | | DELAY3 | -0.313 | 0.496 | **0.929** | 0.648 | 0.504 | 0.457 |
| | Engaging | ENG1 | -0.524 | 0.604 | 0.677 | **0.814** | 0.508 | 0.651 |
| | | ENG2 | -0.481 | 0.605 | 0.471 | **0.772** | 0.524 | 0.695 |
| | | ENG3 | -0.429 | 0.396 | 0.347 | **0.701** | 0.367 | 0.421 |
| | | ENG4 | -0.337 | 0.321 | 0.451 | **0.727** | 0.321 | 0.413 |
| | | ENG5 | -0.347 | 0.482 | 0.289 | **0.505** | 0.299 | 0.244 |
| | | ENG6 | -0.354 | 0.509 | 0.699 | **0.742** | 0.507 | 0.590 |
| | | ENG7 | -0.463 | 0.346 | 0.437 | **0.744** | 0.409 | 0.439 |
| Self-Efficacy | | SE1 | -0.462 | 0.353 | 0.553 | 0.583 | **0.872** | 0.606 |
| | | SE2 | -0.478 | 0.462 | 0.482 | 0.529 | **0.902** | 0.545 |
| | | SE3 | -0.373 | 0.524 | 0.305 | 0.433 | **0.856** | 0.471 |
| Satisfaction | | SA1 | -0.491 | 0.522 | 0.447 | 0.699 | 0.615 | **0.981** |
| | | SA2 | -0.454 | 0.507 | 0.478 | 0.690 | 0.604 | **0.981** |




<Figure 8> Results of PLS Analysis

## 8.4 Structural Model

The structural model was assessed by examining path coefficients, their significance level, and the $R^2$ values. Perceived Interaction was treated as $1^{st}$ order construct in the model[1]. Path coefficients indicate the strengths of the relationships between two constructs <Figure 8>. $R^2$ values show the amount of variance explained by the independent constructs [Barclay *et al.*, 1995]. The final dependent construct, Satisfaction, has an $R^2$ value of 0.595, which indicates that the research model accounts for 59.5% of the variance in the dependent variable when Satisfaction was measured by accuracy. It is also instructive to examine the $R^2$ values for the intermediate variable in the structural model. The $R^2$ values for "Cognitive Load" and "Self-Efficacy" were 0.411 and 0.429 respectively. $R^2$ of both Cognitive Load and Self-Efficacy were high enough to make the interpretation of the path coefficients meaningful.

Path coefficients in the structural model were computed with the entire sample. Bootstrapping with 500 resamples was computed to obtain the t-values corresponding to each path, as shown in <Figure 8>. The acceptable t-values for one-tailed tests are 1.64 and 2.33 at the significance levels of 0.05 and 0.01. Information Filtering had a negative impact on Cognitive Load ($\beta$ = -0.26, $p < 0.01$), and therefore H1 was supported. Interaction had a negative impact on Cognitive Load ($\beta$ = -0.49, $p < 0.01$), supporting H2. Interaction effect of Information Filtering and Interaction on Cognitive Load had a negative association ($\beta$ = -0.20, not significant), and H3 was not supported. Though, Cognitive Load had a negative impact on Satisfaction, it is not significant and therefore H4 was not supported ($\beta$ = -0.02, not significant). Interaction had a positive impact on Satisfaction ($\beta$ = 0.58, $p < 0.01$), supporting H5. Interaction also had a positive impact on Self-Efficacy, ($\beta$ = 0.52, $p < 0.01$) supporting H6. Cognitive Load had a negative impact on Self-Efficacy ($\beta$ = -0.20, $p < 0.05$), and H7 was supported. Finally Self-efficacy had a positive impact on Satisfaction ($\beta$ = 0.24, $p < 0.05$) and therefore, H8 was supported.

1) We analyzed the model either Perceived Interaction as 1st order construct and 2nd order construct. The results of both analyses were similar.

### 8.5 Tests for Common Method Bias

We conducted two types of statistical analyses to assess the threat of common methods bias: Harman's one factor test and latent variable test [Podsakoff *et al.*, 2003]. First, in Harman's one factor test, the emergence of a single factor that accounts for a large proportion of the variance in factor analyses suggests a common methods bias. However, no such factor emerged. We loaded all items used to measure both independent and dependent variables into a single exploratory factor analysis. The analysis produced six factors with eigenvalues higher than 1. Taken together, these factors explained 80.7% of the variance in the data, with the first extracted factor accounting for 47.1% of the variance. Given that more than one factor was extracted from the analysis and the first factor accounted for less than 50% of the variance, common method bias is unlikely to be a significant issue.

Second, in a latent variable approach, we added a first-order factor with all of the measures in the theoretical model as indicators [Podsakoff *et al.*, 2003]. A common method factor was therefore added in the research model [Liang *et al.*, 2007]. The results presented in Appendix C demonstrate that the average substantively explained variance of the indicators is 0.713, whereas the average method-based variance is 0.054. The ratio of substantive variance to method variance is 13.2:1. Given the small magnitude and insignificance of method variance, common method bias is unlikely to be a serious concern in this study.

### 8.6 Limitations

Laboratory experimentation provides a highly controlled environment for hypothesis testing while it has several methodological limitations. First, users had a limited time to use the prototype. The prototypes with interactive and information filtering features provided features which might be unfamiliar to some users. Users may have needed more time to get familiar with interactive and information filtering capabilities that were used in the study. Second, the experiment of this study is limited to the retrieval of use cases and does not examine its use on other contexts [Happel and Seedorf, 2006]. Third, our testing of interactive systems is limited. In particular, the prototype II system has a limited interaction by providing the option of going back to the search page. Finally, the use of student subjects may limit the generalizability of the results. However many prior studies on software engineering demonstrate that student subjects provided valid results [Runeson, 2003; Singer and Vinson, 2002].

## Ⅸ. Discussion and Implications

This study empirically confirms that information filtering and perceived interaction have significant impact on cognitive load and self-efficacy. Perceived interaction and self-efficacy are found to have significant impact on satisfaction. This study confirms the relationship between information filtering and cognitive load that was established in a study on ontology pruning (presented in Chapter 3). Six of the eight hypotheses were supported.

The interaction effect between information filtering and perceived interaction on cognitive load was not significant although it had a negative association with cognitive load. We posit that the interactive use of a well-organized ontology can reduce cognitive load thereby increasing user's satisfaction (i.e. hypotheses 3 and 4). However these hypotheses were not supported. Although the negative directions among constructs (i.e. interaction effect, cognitive load, and satisfaction) were found, they were not statistically significant. The statistical power of both hypotheses 3 and 4 was less than 0.3, which was weak to capture the hypothesized relationships. A stronger treatment or increased sample size may help address this issue.

Both perceived interaction and self-efficacy had significant impact on satisfaction. Perceived interaction had a positive impact on satisfaction. Perceived interaction also has an indirect impact on satisfaction via self-efficacy. Thus, Self-efficacy partially mediated the relationship between perceived interaction and satisfaction. The main effect between perceived interaction and satisfaction was statistically significant. The calculated effect size (($R^2$ with mediator-$R^2$ with mediator)/(1-$R^2$ with mediator) = (0.595-0.569)/(1-0.595)) was weak (0.064) [Cohen, 1988]. However the mediation effect of self-efficacy between perceived interaction and satisfaction was significant according to Sobel test (see Appendix D). This result suggests that an interactive system can directly increase user's satisfaction with the system. Indirect increase of user's satisfaction via self-efficacy is not strong. However we should interpret this weak intervening effect of self-efficacy with caution. Even though the intervening effect is weak, it is statistically significant. Therefore, system developers should develop interactive information retrieval systems that increase users' self-efficacy in order to increase user satisfaction. For example, providing positive feedback in a prompt and engaging way can increase users' belief about their capabilities to produce results and enhance user satisfaction.

Cognitive load has a negative impact on satisfaction. When users perceive high-levels of cognitive load, their satisfaction decreases. However, this relationship was not statistically significant. Prior research notes that users stop perusing a long list of retrieved items. Instead, 1) simply discard a large number of results, 2) restart a query in order to reduce the size of the retrieved result, and 3) examine only the top five or six results and select among them. Satisfaction can be maximized when cognitive load is minimized. When users receive a short list of items which are relevant, they will be satisfied with the system. Prior research on e-commerce systems suggests that customers abandon a site when presented with a lengthy list of items for perusal [Nielsen, 2004]. Therefore, the design of information systems which reduce cognitive load will lead to high user satisfaction. In our study, if the results provided by the system had been more compact and rank-ordered based on their relevance, it might have reduced the cognitive load and thereby increased satisfaction significantly.

The mediation effect of self-efficacy between cognitive load and satisfaction was minimal and not significant. The path coefficient values of H7 (-0.2) and H8 (0.24) were significant although the path coefficient value of H4 (-0.02) was not significant. The main effect between cognitive load and satisfaction was not significant. The mediation effect of self-efficacy was also weak and not significant. The calculated effect size is weak (0.084) [Cohen, 1988]. The result of Sobel test shows that the mediation effect of self-efficacy was not significant (Appendix D).

This result suggests that a large variation of satisfaction can be explained by a direct/mediation effect of self-efficacy. Self-efficacy directly affects satisfaction and works as a mediation variable between perceived interaction and satisfaction. Therefore, the developers of information retrieval systems should pay particular attention to factors such as interaction and cognitive load that affect perceived self-efficacy.

## X. Conclusions and Future Work

This research contributes to the literature on systems development in several ways. To facilitate the retrieval of use cases from a library and assess how user satisfaction is affected, this study develops a methodology and prototypes and empirically evaluates them with a theoretically grounded model. Our study suggests that a system which interacts with a user intelligently reduces cognitive load and increases self-efficacy and satisfaction. The model draws on interaction theory, which provides explanation on how interaction between the user and the system with ontological knowledge can increase self-efficacy and satisfaction.

The proposed approach is implemented in a prototype. The interactive query system allows users to retrieve relevant use cases accurately, thereby enhancing the reuse of use cases in large and complex system development projects. An existing requirement management tools like Requisite Pro [Rational, 2005] which provides only a keyword based use case retrieval feature can benefit from incorporating this approach.

Topics for future research include the extension of our approach by using semantic information that will help infer the relationships among use cases. Also, our interactive approach using ontology can be extended to other critical tasks such as system maintenance and testing [Happel and Seedorf, 2006}. Further validation of the prototype is needed to assess its impact on task performance (e.g., the accuracy of results).

## 〈References〉

[1] Agarwal, R., Sambamurthy, V., and Stair, R., "Research Report: The Evolving Relationship Between General and Specific Computer Self-Efficacy-An Empirical Assessment," *Information Systems Research* Vol. 11, No. 4, 2000, p. 418.

[2] Anderson, E. and Sullivan, M., "The antecedents and consequences of customer satisfaction for firms," *Marketing science*, Vol. 12, No. 2, 1993, pp. 125-143.

[3] Back, J. and Oppenheim, C., "A model of cognitive load for IR: implications for user relevance feedback interaction," *Information Research*, Vol. 6, No. 2, 2001, pp. 6-2.

[4] Balzer, W.K., Doherty, M.E., and O'Connor, R.J.R., "Effects of cognitive feedback on performance," *Psychological bulletin*, Vol. 106, No. 3, 1989, pp. 410-433.

[5] Bandura, A., "Self-efficacy mechanism in human agency," *American Psychologist*, Vol. 37, No. 2, 1982, pp. 122-147.

[6] Bandura, A. and Adams, N., "Analysis of self-efficacy theory of behavioral change," *Cognitive Therapy and Research*, Vol. 1, No.

4, 1977, pp. 287-310.

[7] Barclay, D., Higgins, C., and Thompson, R. "The partial least square (PLS) approach to casual modeling: Personal computer adoption and use as an illustration," *Technology*, Vol. 2, No. 2, 1995, pp. 285-309.

[8] Bearden, W., Netemeyer, R., and Mobley, M., *Handbook of marketing scales* Sage Publ, 1993.

[9] Belkin, N.J. and Croft, W.B., "Information filtering and information retrieval: two sides of the same coin?," *Communications of the ACM*, Vol. 35, No. 12, 1992, pp. 29-38.

[10] Berners-Lee, T., Hendler, J., and Lassila, O., "The Semantic Web," *Scientific American*, Vol. 5, 2001, pp. 1-19.

[11] Billinghurst, M. and Starner, T., "Wearable devices: new ways to manage information," *COMPUTER*, Vol. 32, No. 1, 1999, pp. 57-64.

[12] Branting, L., "Acquiring customer preferences from return-set selections," *Case-Based Reasoning Research and Development*), 2001, pp. 59-73.

[13] Chin, W.W., "Issues and opinion on structural equation modeling," *Management Information Systems Quarterly*, Vol. 22, No. 1, 1998a, pp. 7-16.

[14] Chin, W.W., "The partial least squares approach to structural equation modeling," in: *Modern Methods for Business Research,* G.A.Marcoulides (ed.), Lawrence Erlbaum Associates, Mahwah, 1998b, pp. 295-336.

[15] Conesa, J. and Olivé, A., "A Method for Pruning Ontologies in the Development of Conceptual Schemas of Information Systems," *Journal on Data Semantics*, Vol. 5, 2006, pp. 64-90.

[16] Conesa, J., Storey, V.C., and Sugumaran, V., "Using Semantic Knowledge to Improve Web Query Processing," in: *Proceedings of the 11th International Conference on Applications of Natural Language to Information Systems*, Springer, Klagenfurt, Austria, 2006, pp. 106-117.

[17] Cronin, J.J.J. and Taylor, S.A., "Measuring Service Quality: A Reexamination and Extension," *Journal of Marketing*, Vol. 56, 1994, pp. 55-68.

[18] Cyc, "Cyc Ontology," (http://www. cyc.com).

[19] Delone, W.H. and McLean, E.R., "Information Systems Success: The Quest for the Dependent Variable," *Information Systems Research*, Vol. 3, No. 1, 1992, pp. 60-95.

[20] Driver, M., "Exploring student perceptions of group interaction and class satisfaction in the web-enhanced classroom," *The Internet and Higher Education*, Vol. 5, No. 1, 2002, pp. 35-45.

[21] Drolet, A. and Frances Luce, M., "The rationalizing effects of cognitive load on emotion-based trade-off avoidance," *Journal of Consumer Research*, Vol. 31, No. 1, 2004, pp. 63-77.

[22] Fantechi, A., Gnesi, S., Lami, G., and In, A.M., "Application of Linguistic Techniques for Use Case Analysis," *IEEE Joint International Conference on Requirements Engineering*, 2002, pp. 157-164.

[23] Fantechi, A., Gnesi, S., Lami, G., and Maccari, A., "Applications of linguistic techniques for use case analysis," *Requirements Engineering*, Vol. 8, No. 3, 2003, pp. 161-170.

[24] Foltz, P.W. and Dumais, S.T., "Personalized information delivery: an analysis of information filtering methods," *Communications of the ACM*, Vol. 35, No. 12, 1992, pp. 51-60.

[25] Fornell, C. and Larcker, D., "Evaluating struc-

tural equation models with unobservable variables and measurement error," *Journal of marketing research*), 1981, pp. 39-50.

[26] Fuchs, N.E. and Schwertel, U., "Reasoning in Attempto Controlled English," Proceedings of the International Workshop on Principles and Practice of Semantic Web Reasoning, 2003, pp. 174-188.

[27] Gopher, D. and Braune, R., "On the psychophysics of workload- Why bother with subjective measures?," *Human Factors*, Vol. 26, 1984, pp. 519-532.

[28] Gruber, T.R., "Toward principles for the design of ontologies used for knowledge sharing," in: *Formal Ontology in Conceptual Analysis and Knowledge Representation,* N. Guarino and R. Poli (eds.), Kluwer Academic Publishers, Deventer, The Netherlands, 1993.

[29] Guarino, N., "Formal Ontology and Information Systems," in: *Proceedings of the FOIS'98*, IOS Press, 1998, pp. 3-15.

[30] Happel, H. and Seedorf, S., "Applications of ontologies in software engineering," Citeseer, 2006, pp. 5-9.

[31] Henry, J. and Stone, R., "A structural equation model of end-user satisfaction with a computer-based medical information system," *Information Resources Management Journal*, Vol. 7, 1994, pp. 21-21.

[32] Hevner, A.R., March, S.T., Park, J., and Ram, S., "Design Science in Information Systems Research," *MIS Quarterly*, Vol. 28, No. 1, 2004, pp. 75-105.

[33] Hogarth, R.M., *Judgment and Choice* John Wiley and Sons, 1996.

[34] Johnson, R. and Marakas, G., "Research report: the role of behavioral modeling in computer skills acquisition: toward refinement of the model," *Information Systems Research*, Vol. 11, No. 4, 2000, pp. 402-417.

[35] Jones, Q., Ravid, G., and Rafaeli, S., "Information overload and the message dynamics of online interaction spaces: A theoretical model and empirical exploration," *Information Systems Research*, Vol. 15, No. 2, 2004, pp. 194-210.

[36] Kaiya, H. and Saeki, M., "Ontology Based Requirements Analysis: Lightweight Semantic Processing Approach," Fifth International Conference on Quality Software (QSIC'05), 2005, pp. 223-230.

[37] Keller, J. and Suzuki, K., "Learner motivation and e-learning design: A multinationally validated process," *Learning, Media and Technology*, Vol. 29, No. 3, 2004, pp. 229-239.

[38] Kettinger, W.J. and Lee, C.C., "Perceived service quality and user satisfaction with the information services function," *Decision Sciences*, Vol. 25, No. 5-6, 1994, pp. 737-766.

[39] Klischewski, R., "Ontologies for e-document management in public administration," *Business Process Management Journal*, Vol. 12, No. 1, 2006, pp. 34-47.

[40] Lamport, M.A., "Student-Faculty Informal Interaction and the Effect on College Student Outcomes: A Review of the Literature," *Adolescence*, Vol. 28, No. 112, 1993.

[41] Lassila, O. and McGuinness, D., "The Role of Frame-Based Representation on the Semantic Web," KSL-01-02, Stanford University, Stanford, California, p. 9.

[42] Malone, T.W., Grant, K.R., Turbak, F.A., Brobst, S.A., and Cohen, M.D., "Intelligent information-sharing systems," *Communications of the ACM*, Vol. 30, No. 5, 1987, pp.

390-402.

[43] Marcia, B. and Pulman, S., "A method for controlling the production of specifications in natural language," *The Computer Journal*, Vol. 38, 1995, pp. 310-318.

[44] Mason, O., "QTag-a Portable POS Tagger," Accessed on Jan 7, 2005 (http://www.english.bham.ac.uk/staff/omason/software/qtag.html).

[45] Matuszek, C., Cabral, J., Witbrock, M., and DeOliveira, J., "An Introduction to the Syntax and Content of Cyc," 2006 AAAI Spring Symposium on Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering, 2006.

[46] McMillan, S. and Hwang, J., "Measures of perceived interactivity: An exploration of the role of direction of communication, user control, and time in shaping perceptions of interactivity," *Journal of Advertising*, Vol. 31, No. 3, 2002, pp. 29-42.

[47] McSherry, D., "Balancing user satisfaction and cognitive load in coverage-optimised retrieval," *Knowledge-Based Systems*, Vol. 17, No. 2-4, 2004, pp. 113-119.

[48] Mitchell, T., Hopper, H., Daniels, D., George-Falvy, J., and James, L., "Predicting self-efficacy and performance during skill acquisition," *Journal of Applied Psychology*, Vol. 79, No. 4, 1994, pp. 506-517.

[49] Nielsen, J., *Designing web usability* Pearson Education, 2004.

[50] Oliver, R., "Varieties of value in the consumption satisfaction response," *Advances in Consumer Research*, Vol. 23, 1996, pp. 143-147.

[51] Paas, F., "Training strategies for attaining transfer of problem-solving skill in statistics: a cognitive-load approach," *Journal of educational psychology*, Vol. 84, No. 4, 1992, pp. 429-434.

[52] Paas, F., Tuovinen, J., Tabbers, H., and Van Gerven, P., "Cognitive load measurement as a means to advance cognitive load theory," *Educational Psychologist*, Vol. 38, No. 1, 2003, pp. 63-71.

[53] Parasuraman, A., Zeithaml, V., and Berry, L.L., "Reassessment of Expectations as a Comparison Standard in Measuring Service Quality: Implications for Further Research," *Journal of Marketing*, Vol. 58, 1994, pp. 111-124.

[54] Park, S., Kim, H., Ko, Y., and Seo, J., "Implementation of an efficient requirments-analysis supporting system using similarity measure techniques," *Information and Software Technology*, Vol. 42, No. 6, 2000, pp. 429-438.

[55] Pintrich, P. and Schrauben, B., "Studentsí motivational beliefs and their cognitive engagement in classroom academic tasks," *Student perceptions in the classroom*) 1992, pp. 149-183.

[56] Podsakoff, P., MacKenzie, S., Lee, J., and Podsakoff, N., "Common method biases in behavioral research: A critical review of the literature and recommended remedies," *Journal of Applied Psychology*, Vol. 88, No. 5, 2003, pp. 879-903.

[57] Poli, R., "Descriptive, formal and formalized ontologies," *International Journal of Human-Computer Studies*, Vol. 56, No. 6, 2002, pp. 639-664.

[58] Qin, T., Liu, T.-Y., and Li, H., "A general approximation framework for direct optimization of information retrieval measures," *Information retrieval*, Vol. 13, No. 4, 2010, pp. 375-397.

[59] Rai, A., Lang, S.S., and Welker, R.B., "Assessing the Validity of IS Success Models: An Empirical Test and Theoretical Analysis," *Information Systems Research*, Vol. 13, No. 1, 2002, pp. 50-69.

[60] Rational, I.B.M., "Rational RequisitePro," 2005 (http://www3.software.ibm.com/ibmdl/pub/software/rational/web/datasheets/version6/reqpro.pdf).

[61] Ringle, C., Sarstedt, M., and Straub, D., "A critical look at the use of PLS-SEM in MIS quarterly," *MIS Quarterly (MISQ)*, Vol. 36, No. 1, 2012, pp. iii-xiv.

[62] Runeson, P., "Using students as experiment subjectsñan analysis on graduate and freshmen student data," Citeseer, 2003.

[63] Saeki, M., "Ontology-Based Software Development Techniques," in: *ECRIM News*, 2004, pp. 14-15.

[64] Sengupta, K. and Te'eni, D., "Cognitive Feedback in GDSS: Improving Control and Convergence," *MIS Quarterly*, Vol. 17, No. 1, 1993, pp. 87-113.

[65] Singer, J. and Vinson, N., "Ethical issues in empirical studies of software engineering," *IEEE Transactions on Software Engineering*) 2002, pp. 1171-1180.

[66] Storey, V.C., Burton-Jones, A., Sugumaran, V., and Purao, S., "CONQUER: A Methodology for Context-Aware Query Processing on the World Wide Web," *Information Systems Research*, Vol. 19, No. 1, 2008, pp. 3-25.

[67] Sutton, D.C., "Linguistic Problems with Requirements and Knowledge Elicitation," *Requirements Engineering*, Vol. 5, No. 2, 2000, pp. 114-124.

[68] Sweller, J., "Cognitive load during problem solving: Effects on learning," *Cognitive Science*, Vol. 12, 1988, pp. 257-285.

[69] Sweller, J. and Chandler, P., "Why Some Material Is Difficult to Learn," *Cognition and Instruction*, Vol. 12, No. 3, 1994, pp. 185-233.

[70] van Merriëboer, J.J.G. and Sweller, J., "Cognitive Load Theory and Complex Learning: Recent Developments and Future Directions," *Educational Psychology Review*, Vol. 17, No. 2, 2005, pp. 147-177.

[71] Venkatesh, V., "Determinants of Perceived Ease of Use: Integrating Control, Intrinsic Motivation, and Emotion into the Technology Acceptance Model," *Information Systems Research*, Vol. 11, No. 4, 2000, pp. 342-365.

[72] Venkatesh, V. and Davis, F., "A Model of the Antecedents of Perceived Ease of Use: Development and Test," *Decision Sciences*, Vol. 27, No. 3, 1996, pp. 451-481.

[73] Vulić, I., De Smet, W., and Moens, M.-F., "Cross-language information retrieval models based on latent topic models trained with document-aligned comparable corpora," *Information retrieval*, Vol. 16, No. 3, 2013, pp. 331-368.

[74] W3C, "OWL Web Ontology Language," 2004 (http://www.w3.org/TR/owl-features/).

[75] Wells, J.D., Fuerst, W.L., and Choobineh, J., "Managing information technology (IT) for one-to-one customer interaction," *Information and Management*, Vol. 35, No. 1, 1999, pp. 53-62.

[76] Wixom, B. and Todd, P., "A theoretical integration of user satisfaction and technology acceptance," *Information Systems Research*, Vol. 16, No. 1, 2005, pp. 85-102.

[77] Wood, R., Atkins, P., and Tabernero, C., "Self-efficacy and strategy on complex tasks,"

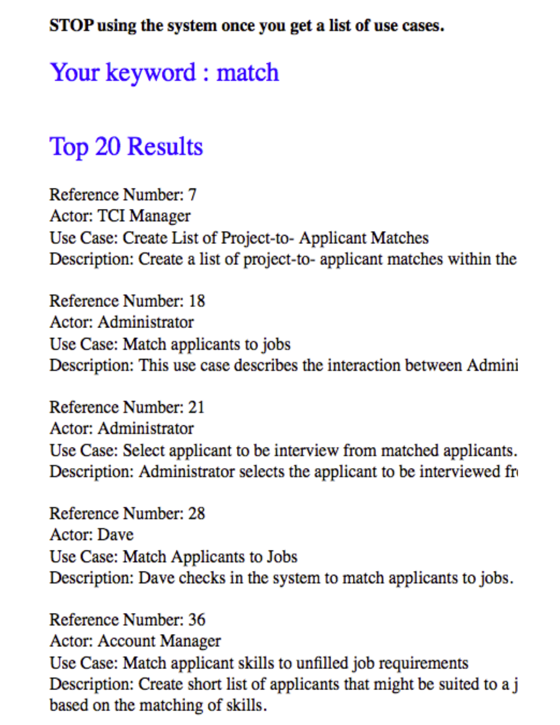
*Applied Psychology*, Vol. 49, No. 3, 2000, pp. 430-446.

[78] Wood, R. and Bandura, A., "Impact of conceptions of ability on self-regulatory mechanisms and complex decision making," *Journal of personality and social psychology*, Vol. 56, No. 3, 1989, pp. 407-415.

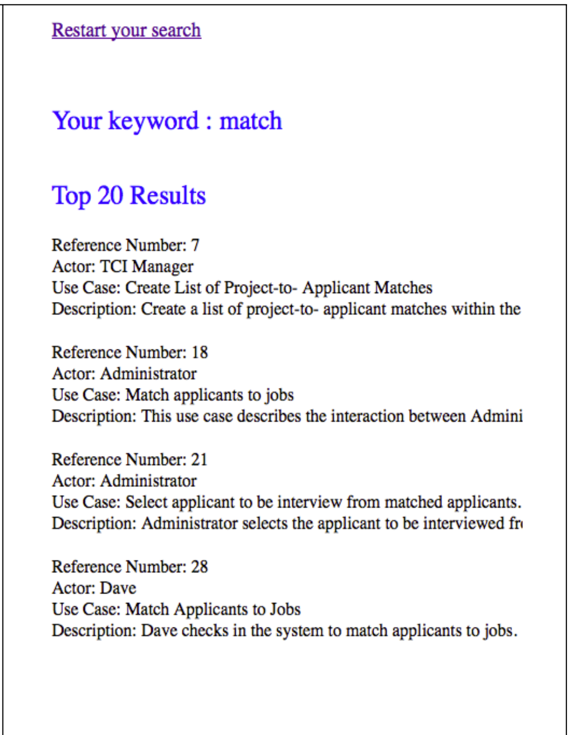
[79] Yi, M. and Davis, F., "Developing and validating an observational learning model of computer software training and skill acquisition," *Information Systems Research*, Vol. 14, No. 2, 2003, pp. 146-169.

[80] Yi, M. and Hwang, Y., "Predicting the use of web-based information systems: self-efficacy, enjoyment, learning goal orientation, and the technology acceptance model," *International Journal of Human-Computer Studies*, Vol. 59, No. 4, 2003, pp. 431-449.

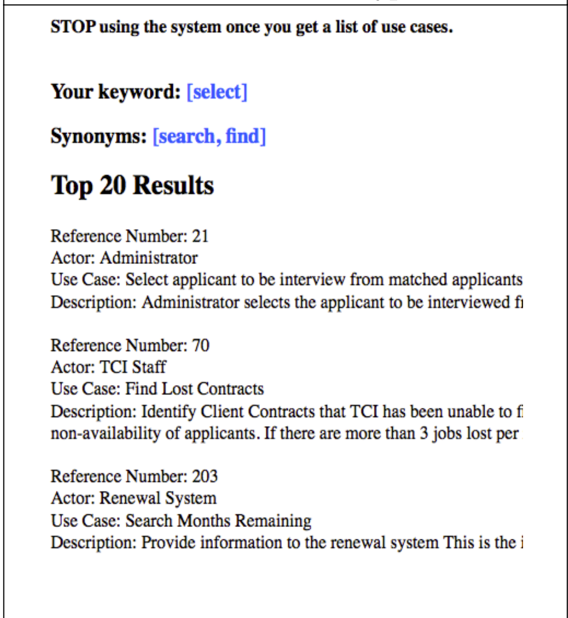
# 〈Appendix A〉 Screenshots of Four Prototypes



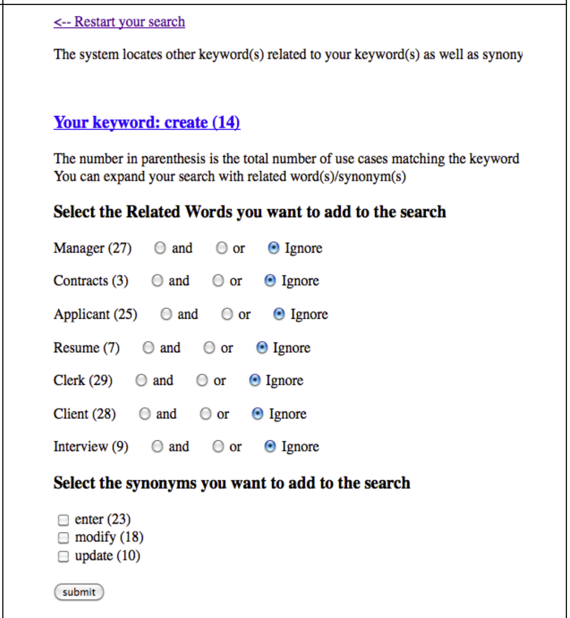

Screenshot of Prototype I




Screenshot of Prototype II




Screenshot of Prototype III




Screenshot of Prototype IV

# 〈Appendix B〉 Survey Questionnaire

## Instruction

You are a systems analyst at TCI, a company that develops web based systems. You are working on a project for the development of a web based system that helps users find suitable dating partners. Your assignment is to develop use cases for the critical functionalities that should be supported by the system. You recognize that reviewing use cases from past projects that have included functionalities similar to your current project will be very helpful in your assignment. Your manager has provided you access to an online library of use cases from past projects completed at TCI. You can use a web based retrieval system that allows you to retrieve use cases from this library.

Your task in this experiment is to use this online retrieval system to find two of the most relevant use cases from the library. Each use case in the library is identified by a reference number. At the conclusion of your search, you need to report the reference numbers of the two most relevant use cases.

Specifically, your task involves the following steps:

1) Access the Use Cases Retrieval System (by clicking on the link at the end of these instructions).
2) Enter keyword (s) and click the Submit button to find use cases that match the keyword(s). Do NOT use the system more than two times.
3) Identify the two of the most relevant use cases from the retrieved results.
4) Report the reference numbers of the two most relevant use cases in the text boxes provided below.

Online Use Cases Retrieval System (Depending on the treatment, one of the four prototypes is provided)

**1. Report the reference numbers of the two most relevant Use Cases in the text boxes provided below.**

_____

_____

**2. Did you use the use cases retrieval system?**

Yes No

**3. Is the retrieval system interactive?**

Yes No

**4. Please take the survey below**

**Cognitive load (Three items, Seven likert scale (1 = very little; 7 = very much))**

- How much mental effort was required to perform the entire task (identifying keywords, using the system and selecting two use cases)?
- How difficult was it for you to perform the entire task (identifying keywords, using the system and selecting two use cases)?
- How burdensome was the task of identifying keywords, using the system and selecting two use cases?

**Perceived Interaction (Fourteen items, Seven likert scale (1 = not at all descriptive; 7 = very descriptive)**

Conversation (4 items)

- The retrieval system helps me INCREMENTALLY refine my search by adding more keywords.
- The retrieval system provides the ability to add ADDITIONAL keywords after displaying the results.
- The retrieval system is interactive.
- The retrieval system DOES NOT provide the ability to refine my search by adding more keywords.

No-Delay (3 items)

- The retrieval system provides fast response.
- The retrieval system responds slowly.
- The retrieval system operates at high speed.

Engaging (7 items)

- The retrieval system keeps my attention focused on the task.
- It is easy to select relevant use cases from the results provided by the system.
- The interaction with the retrieval system is unmanageable.
- The retrieval system DOES NOT allow me to keep my focus on the task.
- The retrieval system interacts with me passively rather than actively guiding me in my task.
- The retrieval system provides immediate answers to my search request.
- The retrieval system DOES NOT provide relevant use cases.

**Self-efficacy (Three items, Eleven likert scale (0 = completely disagree; 10 = completely agree))**
- I believe I have the ability to retrieve the most relevant use cases from the system.
- I believe I have the ability to INTERACTIVELY use the system by refining my search to find the most relevant use cases.
- I believe I have the ability to locate the most relevant use cases with ADDITIONAL relevant keywords provided by the system.

**Satisfaction (Two items, Seven likert scale (1 = strongly disagree; 7 = strongly agree))**
- All things considered, I am very satisfied with the retrieval system.
- Overall, my interaction with the retrieval system is very satisfying.

**Please provide the following background information.**
Age
Gender (Male/Female)
Past educational experience with systems analysis (in month)
Past professional experience with systems analysis (in month)
Past system development experience (in month)

My level of proficiency in using information retrieval system (such as google) is
(seven likert scale, 1 = very low; 7 = very high)

# 〈Appendix C〉 Common Method Bias Analysis

| Construct | Indicator | Substantive Factor Loading (R1) | $R1^2$ | Method Factor Loading (R2) | $R2^2$ |
|---|---|---|---|---|---|
| Ontology | Ontology | 1 | 1.000 | 0 | 0.000 |
| Perceived Interaction | CONV1 | 0.443 | 0.196 | 0.255 | 0.065 |
| | CONV2 | $0.587^{*}$ | 0.345 | 0.065 | 0.004 |
| | CONV3 | $0.849^{**}$ | 0.721 | -0.079 | 0.006 |
| | DELAY1 | $1.109^{**}$ | 1.230 | $-0.355^{*}$ | 0.126 |
| | DELAY2 | $1.065^{**}$ | 1.134 | $-0.468^{*}$ | 0.219 |
| | DELAY3 | $1.156^{**}$ | 1.336 | $-0.378^{*}$ | 0.143 |
| | ENG1 | $0.603^{**}$ | 0.364 | 0.25 | 0.063 |
| | ENG2 | 0.225 | 0.051 | $0.557^{*}$ | 0.310 |
| | ENG3 | 0.392 | 0.154 | 0.203 | 0.041 |
| | ENG4 | $0.858^{**}$ | 0.736 | -0.25 | 0.063 |
| | ENG6 | $0.819^{**}$ | 0.671 | -0.043 | 0.002 |
| | ENG7 | 0.454 | 0.206 | 0.198 | 0.039 |
| Cognitive Load | CL1 | $0.901^{**}$ | 0.812 | 0.011 | 0.000 |
| | CL2 | $1.007^{**}$ | 1.014 | 0.079 | 0.006 |
| | CL3 | $0.852^{**}$ | 0.726 | -0.09 | 0.008 |
| Self-Efficacy | SE1 | $0.754^{**}$ | 0.569 | 0.136 | 0.018 |
| | SE2 | $0.888^{**}$ | 0.789 | 0.019 | 0.000 |
| | SE3 | $0.995^{**}$ | 0.990 | -0.152 | 0.023 |
| Satisfaction | SA1 | $0.969^{**}$ | 0.939 | 0.015 | 0.000 |
| | SA2 | $0.993^{**}$ | 0.986 | -0.015 | 0.000 |
| Average | | **0.806** | **0.713** | **-0.002** | **0.054** |

$^{*}p < .05$, $^{**}p < .01$.

# 〈Appendix D〉 Test of Mediation Effect of Self-Efficacy

Perceived Interaction → **Self-Efficacy** → Satisfaction

| | Test Stat | Std. Error | p-value |
|---|---|---|---|
| Sobel | 2.051 | 0.061 | 0.040 |
| Aroian | 2.016 | 0.062 | 0.044 |
| Goodman | 2.088 | 0.060 | 0.037 |

Cognitive Load → **Self-Efficacy** → Satisfaction

| | Test Stat | Std. Error | p-value |
|---|---|---|---|
| Sobel | -1.251 | 0.033 | 0.147 |
| Aroian | -1.374 | 0.035 | 0.169 |
| Goodman | -1.542 | 0.031 | 0.123 |

## ◆ About the Authors ◆

Jongwoo Kim

Jong woo Kim is an Assistant Professor of Management Sciences and Information Systems (MSIS) at the University of Massachusetts Boston. He received his Ph.D. degree in CIS from the Georgia State University. His work appears in several journals including Information and Management, IEEE Transactions on Engineering Management, Information Technology and People, IEEE Transactions on Professional Communication, and several conferences. His research interests include conceptual modeling, IT in inter-organizational networks, and IS security.

Jordi Conesa

Jordi Conesa is an assistant professor of Information Systems at Universitat Oberta de Catalunya. In 2008 he received his Ph.D. in Software Engineering from the Technical University of Catalonia, after a research stay of six months in the Georgia State University of Atlanta (US). During the year 2011 he had enjoyed a one year postdoc stay in the Internet Interdisciplinary Institute as a Resident Researcher. His research interest concerns the areas of Conceptual Modeling, Ontologies, Semantic Web, Knowledge-Based Systems and e-Learning. His long-term goal is to develop methodologies and tools to use ontologies effectively in several application domains, such as conceptual modeling, software engineering and e-learning. He has authored more than twenty research papers, has participated in several research projects, including EU-FP7, CICYT and AVANZA funded projects, and has contributed in the organization of some international conferences, such as the ER 2008.

Balasubramaniam Ramesh

Balasubramaniam Ramesh is Board of Advisors Professor of Computer Information Systems at Georgia State University. His research interests include requirements engineering and traceability, agile software development, decision support systems and knowledge management. His work appears in several leading journals including MIS Quarterly, ISR, JMIS, JAIS, EJIS, ISJ, IEEE Transactions on Software Engineering, ACM Transactions on MIS, DSS, Communications of the ACM, IEEE Computer, IEEE Intelligent Systems, IEEE Software among others.

Submitted : August 13, 2014
1st revision : December 04, 2014
Accepted : December 06, 2014