2014

Towards Accessible, Usable Knowledge Frameworks in Engineering

Jeffrey Mcpherson
University of Massachusetts Amherst

TOWARDS ACCESSIBLE, USABLE KNOWLEDGE FRAMEWORKS IN ENGINEERING

A Thesis Presented

by

JEFFREY D. MCPHERSON

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

February 2014

Department of Mechanical and Industrial Engineering

TOWARDS ACCESSIBLE, USABLE KNOWLEDGE FRAMEWORKS IN ENGINEERING

A Thesis Presented

by

JEFFREY D. MCPHERSON

Approved as to style and content by:

Sundar Krishnamurty, Co-Chair

Ian Grosse, Co-Chair

Jack Wileden, Member

Donald Fisher, Department Head
Department of Mechanical and Industrial
Engineering

ACKNOWLEDGEMENTS

I would like to thank my advisors Dr. Sundar Krishnamurty, Dr. Ian Grosse, and Dr. Jack Wileden. Without their guidance and support since my Junior year as an undergraduate researcher at the University of Massachusetts Amherst, I would not be pursuing my master's degree.

I have received ample guidance from my fellow researchers in the Center for e-Design. Dr. Tony McCaffrey has provided expertise in the field of experimental design and administration that was instrumental in the completion of this thesis. Douglas Eddy has provided guidance as an expert in the field of engineering knowledge management and ontology. I would like to acknowledge Jay Breindel for his guidance in getting me started on my first project in graduate school, and John Altidor for his mentorship during my time as an undergraduate researcher. Edward Roy and Ryan Szeto have provided programming expertise to aid in the customization and configuration of various semantic wiki platforms investigated in this thesis. I would also like to thank my fellow lab mates Vivek Premkumar, Michael Berthaume, and Leo Xuzhang Lin for their support over the past few years.

I would also like to thank my family and friends for their tremendous support over my past 6 years of school at the University of Massachusetts Amherst. Without this support, I would not have been able to complete my goals.

This research was made possible by funding from the NSF to the Center for e-Design Award Number 0838747.

ABSTRACT

TOWARDS ACCESSIBLE, USABLE KNOWLEDGE FRAMEWORKS IN ENGINEERING

FEBRUARY 2014

JEFFREY D. MCPHERSON, B.S., UNIVERSITY OF MASSACHUSETTS AMHERST

M.S.M.E., UNIVERISTY OF MASSACHUSETTS AMHERST

Directed by: Professor Sundar Krishnamurty and Professor Ian Grosse

A substantial amount of research has been done in the field of engineering knowledge management, where countless ontologies have been developed for various applications within the engineering community. However, despite the success shown in these research efforts, the techniques have not been adopted by industry. This research aims to uncover the reasons for the slow adoption of engineering knowledge frameworks, namely ontologies, in industry.

There are two projects covered in this thesis. The first project is the development of a cross-domain ontology for the Biomesh Project, which spans the fields of mechanical engineering, biology, and anthropology. The biology community is known for its embrace of ontologies and has made their use quite popular with the creation of the Gene Ontology. This ontology spawned the establishment of the Open Biological and Biomedical Ontologies (OBO) Foundry, a consortium which approves and curates ontologies in the biology field. No such consortium exists in the field of engineering. This project demonstrates the usefulness of curated reference ontologies. Ontological knowledge bases in four different domains were imported and integrated together to connect previously disparate information. A case study with data from the Biomesh

Project demonstrates cross-domain queries and inferences that were not possible before the creation of this ontology.

In the second part of this thesis we investigate the usability of current ontology tools. *Protégé*, the most popular ontology editing tool, is compared to *OntoWiki*, a semantic wiki. This comparison is done using proven techniques from the field of Human-computer interaction to uncover usability problems and point out areas where each system excels. A field of 16 subjects completed a set of tasks in each system and gave feedback based on their experience. It is shown that while *OntoWiki* offers users a satisfying interface, it lacks in some areas that can be easily improved. *Protégé* provides users with adequate functionality, but it is not intended for a novice user.

TABLE OF CONTENTS

Page

CHAPTER

APPENDICES

LIST OF TABLES

Table Page

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Knowledge capture and reuse are vastly important in many disciplines due to the massive rise in data and information being gathered in everyday practice [1]. In the engineering design community, semantic frameworks seek to systematically categorize and reveal relationships in this information to form knowledge. Significant work has been done to develop semantic knowledge frameworks in the form of ontologies by the engineering design community, specifically by the researchers at the University of Massachusetts and the National Science Foundation (NSF) Center for e-Design [2-8]. Despite advancements in the development of semantic frameworks, there has been a lack of adoption by the engineering industry where these frameworks are meant to be used. In this thesis, we investigate two reasons why ontologies have not been adopted by the engineering industry: lack of proper ontology standards and curation, and lack of ontology usability.

It is widely known that the biology community experiences much larger buy-in to their ontological knowledge management approach compared to the engineering community. They have demonstrated that proper ontology creation and curation can lead to massive success [9, 10]. In Chapter 3 of this thesis, we demonstrate the usefulness of curated reference frameworks by interlinking ontologies from the engineering and biology fields in a cross-domain knowledge management case study.

It has been identified in previous work that the usability of engineering knowledge frameworks is a significant deterrent to their adoption [11]. It was observed that the engineering design research community relies heavily on the popular ontology editing tool *Protégé* in case studies to demonstrate the usefulness of ontologies. However, semantic wikis offer an alternative ontology experience that promises to be more familiar and user-friendly [12-19]. Chapter 4 of this thesis outlines a usability study of *Protégé* and *OntoWiki*, which is an open-source semantic wiki. This study aims to do two things: first, identify which system is superior to the other in terms of usability; second, suggest ways in which the superior system can be customized and improved to make the user experience even better.

1.1. Objective

The goal of this thesis is to explore ways in which to make engineering knowledge frameworks more accessible and usable. By demonstrating the success of curated and shared knowledge frameworks in other disciplines, namely biology, and by uncovering the usability problems with *Protégé* and current semantic wikis, we hope to suggest ways to encourage greater ontology usage in engineering.

1.2. Research Challenges

The first research challenge addressed in this work was to create an interlinked, cross-domain ontology for Biomesh (www.biomesh.org). This project involved a deep understanding of the current ontological culture in the biology community. It also required the use of widely accepted high-level ontologies that had been adopted by the Open Biological and Biomedical Ontologies (OBO) Foundry [10]. It is shown that a powerful knowledge base, capable of interdisciplinary queries and inferences, can be built by linking cross-domain reference ontologies.

The second research challenge is to use proven techniques and best practices from the human-computer interaction (HCI) field to explore the usability of the ontology editing tool *Protégé* and the open-source semantic wiki *OntoWiki*. In order to properly execute this research challenge, an extensive review of HCI usability testing methods was required. A study involving 16 participants was administered where subjects completed a set of tasks in both *Protégé* and *OntoWiki*. Quantitative data was collected in the form of user feedback and subjects were given the chance to answer open response questions. Recommendations are made in the conclusions of Chapter 4 concerning which system is superior in terms of usability.

1.3. Approach

The challenges outlined in the previous section will be overcome in two separate stages, as outlined below:

1. Develop a cross-domain, interlinked ontology which satisfies the needs of the Biomesh project and demonstrate its usefulness.
2. Perform a usability analysis of *Protégé* and *OntoWiki* using proven techniques from the field of HCI and present results.

1.4 Organization of Proposal

This introduction section provided a high level overview of the intent and scope of this work. Next, Chapter 2 provides a detailed literature review of the subjects covered in this thesis. Chapter 3 discusses the integration of biological and engineering ontologies through the creation of the Biomesh Ontology. Next, a usability study comparing *Protégé* and *OntoWiki* is conducted and discussed in Chapter 4. Chapter 5 provides a research summary and Chapter 6 discusses future work on this topic.

1.5 Definitions

In this thesis, we will refer to data, information, and knowledge as they are defined in [20]: Data are syntactic entities, information is interpreted data, and knowledge is learned from information. For example, data could be seen as an individual number. A collection of individual numbers in the form of a table would form information. An understanding of the collection of tables would be considered knowledge.

The term “curated” is used frequently in this thesis. A curated ontology is one which is looked after by a consortium of expert individuals who have a vested interest in the content represented in the ontology.

CHAPTER 2

LITERATURE REVIEW

2.1. Ontology and the Semantic Web

According to Barry Smith [21], philosophical ontology has been a subject of study since the time of Aristotle around 300 BC. The history of the word "ontology" dates back to the 1600s and was defined in English as "an Account of being in the Abstract". In today's terms, philosophical ontology "has sought the definitive and exhaustive classification of entities in all spheres of being" [21].

In the 1980s, the information science and artificial intelligence (AI) fields took the term "ontology" and slightly skewed its meaning to cover their contributions in knowledge engineering [21]. Later, in 1993, Thomas Gruber published a paper that said: "A specification of a representational vocabulary for a shared domain of discourse – definitions of classes, relations, functions, and other objects – is called an ontology" [22]. This definition is the one which the knowledge engineering field has maintained for the past 20 years and is the one that will be used in this thesis. During this time, a slew of publications across many fields (e.g. information science, biology/bioinformatics, engineering, etc.) have sought to use ontologies to define strict knowledge models.

In 2001, Tim Berners-Lee outlined his vision of the semantic web [23]. This was a web that would use strict markup definitions and data structures to turn information into meaningful content. This structured "web of knowledge" would allow computers to perform automatic operations that are not currently possible on today's jumbled World Wide Web. Berners-Lee specified that ontologies were to be used to define knowledge in the semantic web. More recently and possibly counterproductively, there has been a push for Linked Open Data on the web. Linked Open Data provides "a set of best practices for publishing and connecting structured data on the Web" [24]. Some believe that this effort is counterproductive because the standard of annotating and structuring this data before it is on the web is not being adhered to by all participants [25].

Although the semantic web has made no official "debut", popular web applications and tools like Facebook and Google are moving to more semantically enabled search systems. Facebook has implemented a "graph search" feature which has the ability exploit the connected nature of objects and terms which are stored in this service [26].

While these more expressive search engines and web features are being developed, software companies like Microsoft are working on creating more intuitive user interfaces to be used for completing complicated tasks. The software giant recently introduced Microsoft Power BI Q&A, which interprets natural language inputs from users to perform complicated data analysis tasks [27]. The "interface first" design approach used to create this program is one that is slowly being adopted by the semantic web community as evidenced in the recent emergence of semantic wikis for ontology interaction.

2.2. Semantic Wikis

Semantic wikis are very similar to normal wikis, in that they are editable and community based. They differ in the fact that they rely on an underlying knowledge model to store and relate the information that resides in them. This knowledge model, in the applications described in this thesis, is an OWL [28] based ontology. Semantic wiki alternatives are numerous, including *DataWiki* [12] (formerly *SMW+* [13]), *OntoWiki* [14, 15], *Knoodl* [16], *Kiwi* [17], *SweetWiki* [18], and *AceWiki* [19, 29]. Here, we discuss the advantages and disadvantages of each wiki system.

Knoodl [16] is a web-based application that allows users to create "Communities" where various things can be uploaded and linked. Each Community can consist of Members, Wikis, Ontologies, and Queries, among others. Members can be defined so that only select individuals can make changes to a community page. Wikis can be created that pertain to the content of the Community page, but the Wikis cannot be linked to the underlying ontology. Ontologies can be uploaded and viewed using a content tree. SPARQL [30] queries can be created using a primitive tool.

SweetWiki [18] was a short-lived project which lasted from about 2006-2009. This web-based semantic wiki was focused on social contribution to semantic content. SweetWiki claimed to be a more usable semantic wiki because it implemented web standards in its format, macros, semantic annotation, and ontology manipulation. This wiki implemented a "What You See Is What You Get" (WYSIWYG) text editor which made it easy for novice users to make changes to the wiki pages.

AceWiki [19, 29] is a project that aims to meld controlled English (specifically Attempto Controlled English [31], or ACE), with a semantic wiki to improve usability and expressivity. In short, ACE is a controlled subset of English for writing requirements which can be processed by computers. AceWiki claims to display ontologies in a manner that is friendly to novices and very closely resembles natural language. Adding knowledge to AceWiki is meant to be simple and is performed by typing natural sentences stating facts. The usability of AceWiki was evaluated in [19] and a normal distribution was observed around "medium" usability.

Although they are still in a new, developmental stage, form-based wikis, such as *OntoWiki [14]* (successor to *pOWL*), seem to have reached the necessary point for implementation in a case study to investigate their potential. The most appropriate, currently available form-based wiki option was determined to be *OntoWiki*. *OntoWiki* is an open-source wiki developed by researchers as part of the Agile Knowledge Engineering and Semantic Web (AKSW) [32] project headed by the University of Leipzig. *OntoWiki* has a significant user support community and additionally offers multiple extensions to the basic semantic wiki. This wiki is highly customizable; however, it was discovered that it takes a high level of programming expertise to make even small customizations. *OntoWiki* has very robust ontology import capabilities as it is capable of importing large ontologies.

DataWiki [12] is a commercially supported text-based semantic media wiki run by DIQA [33]. Formerly known as *SMW+*, this wiki resurged after being unsupported for nearly a year after the collapse of its owner Ontoprise. It was not tested in this thesis because no working version was available when testing was administered. *DataWiki* is built on top of the MediaWiki engine which powers Wikipedia. This is certainly the most popular semantic wiki as displayed by the number of participants in recent conferences known as SMWCon [34]. *DataWiki* was heavily investigated and has many appealing features, including a large suite of extensions, faceted search capabilities, easy ontology navigation, user access control, graphical query building, and a WYSIWYG text editor.

2.3. <u>Information and Knowledge Management in Engineering Design</u>

Here we present a background of knowledge management in the engineering design community. We discuss expert systems which were first used in the 1980s for engineering decision making. Next, we discuss some of the standards that have been published to try to deal with the increase in engineering information caused by the advent of Computer-aided Engineering (CAE) and Computer-aided Design (CAD). We then present some of the progress on Product Lifecycle Management (PLM) tools that were created to manage engineering information. Finally, we discuss some of the semantic approaches to knowledge management in engineering design.

2.3.1. Expert Systems

Expert systems, like MYCIN [35], were first developed in the early 1970s to aid in complex decision making processes for physicians. MYCIN was able to make recommendations on antimicrobial therapy selection based on a set of criteria for patients who had bacterial infections. Those in the engineering design community saw that they could adapt this type of decision aid so that it could be used in their field. Research in Carnegie Mellon University's Engineering Design Research Center has yielded several publications on the subject of expert systems for various fields within engineering design (civil engineering, mechanical engineering etc.). Maher et al. describe the applicability of OPS5, SRL, and PROLOG programming languages to a knowledge-based engineering system in [36]. Here, they outline an engineering problem and attempt to solve it using a system based in each language. Later, Rychener in 1988 [37] covered the contributions of many authors to the field of expert systems in engineering design. After the publication of this book, research on expert systems slowed as new areas were investigated.

2.3.2. Engineering Information Standards

The advent of CAE and CAD systems has contributed to a significant increase in engineering information and data. In order to cope with the large rise in engineering information, standards were issued. The most widely used and studied standard is ISO 13033 [38], or the STEP (STandard for Product Data Exchange) standard. Development of STEP began in 1984 and the first parts were published in 1994. This standard is meant to describe a wide range of product-related data over the entire product life-cycle. Recent work has demonstrated that STEP is now capable of capturing finite details, such as 2-dimensional procedural drawings that are used to build 3-dimensional features in CAD systems [39].

2.3.3. Product Lifecycle Management Tools

Commercial product lifecycle management (PLM) tools have become nearly essential in the engineering industry in the past decade. Popular tools include PTC's Windchill [40] and Siemens' Teamcenter [41]. These tools are very widely used and afford large companies the ability to manage projects involving many engineers, software tools, parts, and manufacturing processes. PLM systems allow information to be passed transparently between all participants in a project. Access control can be used to restrict user rights of lower level project members, while managers are able to see all parts of a project. These systems are very robust and feature rich, but they lack a semantic layer that would allow the information stored in them to be linked together and turned into knowledge. Next, we discuss we discuss semantically rich approaches to engineering knowledge management.

2.3.4. Semantic Approaches to Engineering Knowledge Management

In the previous subsections, we have shown the ways in which the engineering community has sought to manage *information*. Here, we look at *knowledge* management tools that employ semantic technology to connect *information* in order to form *knowledge*, as called for by Sainter et al. in 2000 [42].

Despite the many advantages of commercial PLM systems, researchers are still attempting to bridge the gap between these information systems and semantic knowledge management systems. Sudarsan et al. outline a framework which can support all product information, can ease interoperability between CAD/CAE systems, and can capture product evolution [43].

ANSYS developed its Engineering Knowledge Manager (EKM) in an attempt to address shortcomings in metadata tagging of engineering analysis models [44]. While this tool does add value to analysis models, it is unable to store some valuable model knowledge. In a recent thesis, Breindel demonstrated the value that semantic frameworks can add to ANSYS EKM [45].

Research at the University of Massachusetts Amherst and the National Science Foundation Center for e-Design has resulted in a full suite of engineering ontologies [2, 3, 6-8]. This network of ontologies comprises the e-Design Framework and provides a modular approach to define "engineering analysis, optimization, design decision making, etc." [46]. Figure 1 shows the ontologies that are currently part of the framework within the domains depicted along the outside. Ideally, the e-Design Framework would utilize the fully developed ontologies created at UMass with fully developed, web-based ontologies such as units, materials and products. Unfortunately, not all of these ontologies exist in a fully developed state. The e-Design Framework connects all of the available ontologies to allow for local classes, properties, and instances to create an integrated knowledge base suitable for capturing and reusing knowledge.

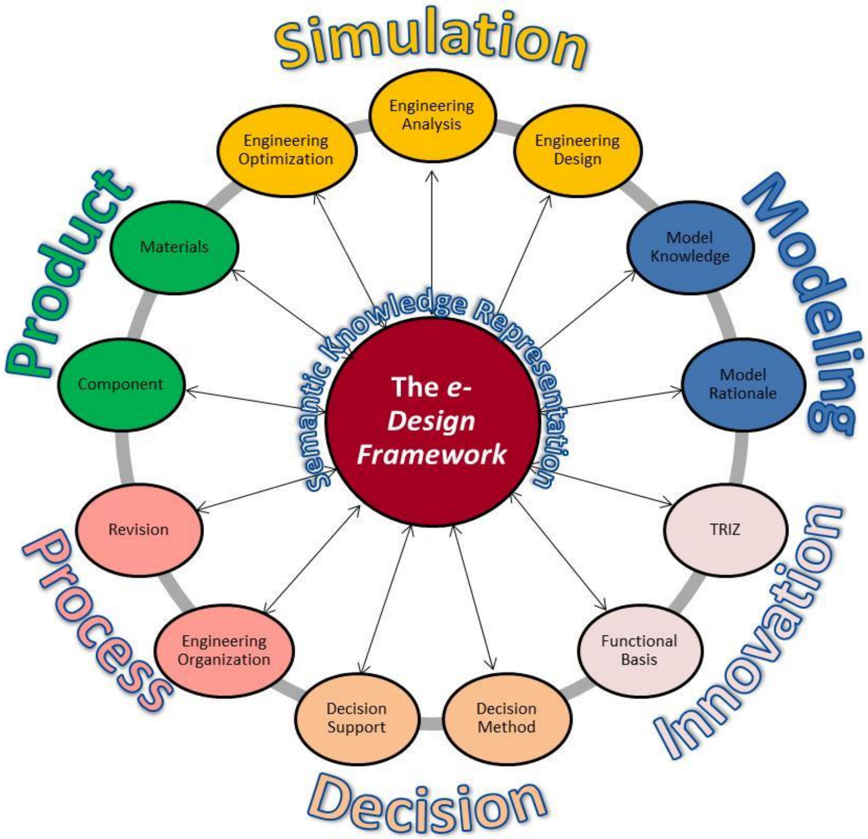


Figure 1: e-Design Framework of ontologies

Increased efforts by others to utilize semantic technologies in engineering have also advanced the cause. Semantic techniques were implemented by McMahon et al. in their Waypoint system [47]. This system is intended as an engineering document search and retrieval tool. Through automatic semantic classification of documents, the system enables search refinements in order to focus queries toward relevant and specific results. A case study implemented the system into the Airbus UK network. A large number of electronic documents were classified with taxonomies that were preexistent in the Airbus UK organization. Although the successfulness of the Waypoint system is still being determined, the semantic relationships as a 'back-end' and a continual feedback system as a ‘front-end’ user interface allows this system to have large potential to meet their specific goals. However, the system has some limitations. It relies on company documents initially being in a structured electronic form, as opposed to paper copies of documents. Further, even with the preexisting organization taxonomies, the system is not able to make inferences about the back-end ontology as many emerging semantic technologies allow.

To overcome the seeming lack of preexisting organization taxonomies and classifications, researchers at Purdue University have placed considerable interest in the acquisition of engineering information and the development of an ontology-based design document analysis and retrieval tool (ODART) [48-50]. This research addresses the need for a facilitated and more automated way to index and locate design documentation. This system mainly focused on the information retrieval and classification portion of the overall knowledge acquisition and reuse process, leaving other aspects of the process unaddressed. The system does not include the ability to allow users to interact with documents in a central location nor the ability to employ security restrictions on sensitive material. These types of system limitations have led to the growing use of Semantic Web technologies.

Ameri et al. [46] used SKOS [47] to create a lightweight ontology to classify and define terms in the manufacturing process. This lightweight ontology was then used to create an OWL-based model which contained properties. This formal model contained axiomatic restrictions, making it possible to reason about the “castability” of a part. This application of a semantic tool to an industry need gives insight into the future of semantic technologies in engineering.

Verhagen et al. [51] used Ardens Knowledge Maker (AKM) in a case study to prove the usefulness of ontological structure in the aerospace industry. They discovered limitations in this web-based KM tool, including its inability to use inference mechanisms and lack of support for automatic ontology import.

Patil et al. have presented work on the exchange of product data using ontologies in [52]. Here, they give an example where different disciplines which work on a product within its lifecycle (product design and manufacturing) have different meanings for the same words. This is an area where meaning would normally be lost, but the use of an ontological approach allows communication between disciplines without such loss of meaning.

CHAPTER 3

INTEGRATING BIOLOGICAL AND ENGINEERING ONTOLOGIES

3.1. Project Summary

This chapter outlines the research performed in the creation of the Biomesh Ontology. This work was published in the proceedings and presented at the ASME 2013 International Design Engineering Technical Conferences and Computer and Information in Engineering Conference (IDETC/CIE) [62].

Methods for acquiring data in engineering and other fields have resulted in the possession of extremely large amounts of information and have sparked the “big data” revolution [1]. However, much work remains to be done to turn that information into knowledge. This is especially apparent in the engineering community, where standardization of knowledge is rare. Although there has been a recent call to develop more ontologies for KM in the engineering community [63], this is only part of the solution. As Professor Barry Smith, arguably the founder of the biomedical ontology community, stated in a recent talk [64], “…the semantics, the content, the meanings of the terms, are still unconstrained because there’s no governance. The governance is applied to the syntax, to the logic of OWL [Web Ontology Language], but not to the content of OWL.” There must be some communal discussion to move towards standardized KM. There must be full buy-in and common understanding of the problem from the community in order for sharing and reuse to work [65]. To this end, the engineering community can look at advances in other fields to better address the KM challenges that it is facing.

One of these advances comes from the biomedical community, where significant steps have been taken towards standardizing their KM techniques. Particularly, organizations like the Open Biomedical Ontologies (OBO) Foundry [10] (http://www.obofoundry.org/) have assembled a team of ontology and domain experts to work towards a standard method of classifying information in several biological domains. Although there are many ontologies published in this domain, very few are accepted as part of the OBO Foundry. The ontologies listed as OBO Foundry ontologies must pass through a stringent review process before being accepted, which is paramount to their success. They are also accepted as domain reference ontologies such that their classes do not overlap. This means that they can be used together in a referential manner without introducing consistency problems. The most famous and widely used ontology residing here is the Gene Ontology (GO) [9].

In this project, we present the formal process used to create an ontology for a cross-domain application, namely for the interlinking of biological materials and finite-element (FE) models which were used as part of the Biomesh project (www.biomesh.org). Biomesh draws information from engineering, species taxonomy, anatomy, and materials domains. All of these domains overlap in various ways, but this overlap is not exposed in the current database. The goal of this paper is expose this overlap and to demonstrate the benefit that engineering can gain from the advances that the biology community has made in the field of KM by interlinking ontologies from engineering and biological fields. Although these ontologies will be interlinked via properties, they will not contain any overlapping classes that will lead to conflicting facts. In doing this, we will be able to satisfy the requirements of an ontology for the Biomesh project. These requirements include 1) facilitate free sharing of the FE models developed by the Biomesh project, 2) provide easy access to the biological materials database, and 3) link knowledge between materials and models that was previously disconnected. This chapter serves as a demonstration of how one can seamlessly interlink well defined ontologies from different domains and allow for automated reasoning to infer new relationships across domains.

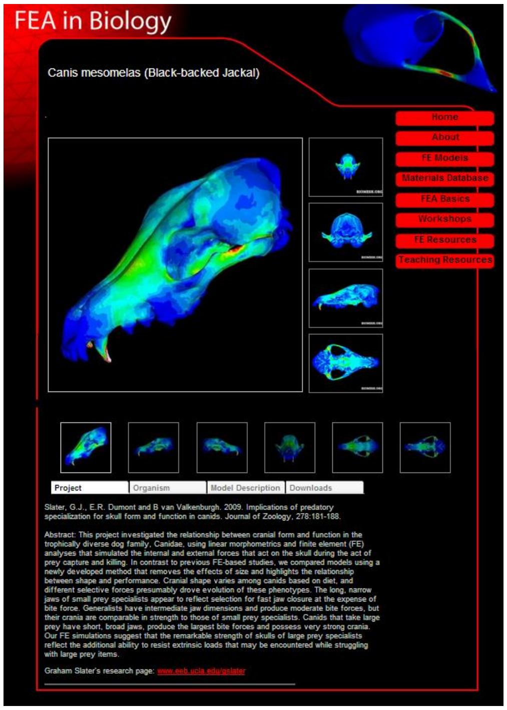
3.2. Background

3.2.1. Biological Ontologies: Lessons Learned

KM is hugely popular and important in the biology community. With so much information being collected, it is necessary to store it intelligently so that it can be properly reused. Therefore, there has been major investment in the development and sharing of ontologies for standardized KM. Once a common vocabulary is used to describe things in a domain, it is possible to perform computational analysis because of the standardization [66]. The largest community ontology project is the Gene Ontology (GO) Consortium [9]. Due to the scope of this project and the number of contributors, it was paramount to create a standardized annotation system. The success of the GO led to the creation of the Open Biomedical Ontologies (OBO) consortium [10]. The OBO requires that all of its ontologies be open, orthogonal, instantiated in a well-specified syntax, and designed to share a common space of identifiers [10]. The principle of orthogonality, which requires that each term is only defined in one ontology [67], allows ontologies to be interoperable so that they can be *imported seamlessly* [65]. The OBO Foundry website (http://www.obofoundry.org/) now boasts more than 100 ontologies contributed by the community. However, only six of these ontologies have passed the review process of the OBO. These six ontologies are critiqued and therefore curated by the community. The National Center for Biomedical Ontology (NCBO) hosts the BioPortal (http://bioportal.bioontology.org/ontologies) which boasts over 300 ontologies in the biology domain. NCBO is also credited with creating the ontology development tool *Protégé* in collaboration with Stanford University. The biology community is also active in sharing biosimulation models. Repositories like CellML [68] (http://models.cellml.org/cellml) hold over 500 standardized, shareable biosimulation models which have been taken from peer reviewed literature. There has also been extensive work done in the area of integrating biological system models with ontologies [69, 70]. This work demonstrates the direction that the engineering community should take to establish standardized knowledge sharing.

3.2.2. Biomesh

Comparative biologists are using finite element analysis to investigate form, function and evolution in organisms as diverse as plants, insects, jellyfish, bats and dinosaurs. The Biomesh project (NSF 0743460 "Biomesh: A Digital Resource Collection at the Biology-Engineering Interface") has played a leading role in this movement by providing unique digital resources coupled with training, expert consultation and outreach. Some of the goals of Biomesh are; 1) to create the cyber infrastructure required to archive existing finite element data and modeling knowledge, and support the continued development of finite element models of biological systems, 2) provide a free, open, community-based platform for sharing finite element data, knowledge, and tools among biologists and engineers, 3) establish and develop community standards and develop and an integrated set of ontologies for finite element models and material properties of biological systems that support archiving of models and sharing and extraction of meaningful metadata and knowledge and 4) provide educational and outreach resources. As a digital repository Biomesh (biomesh.org) hosts numerous downloadable FEA models of biological systems, a searchable database of material property values of biological tissues, educational FE resources, and freely-available software tools developed by the project to support finite element modeling of biological systems. An example of a typical project page describing a finite element model is shown in Figure 2 below.



Canis mesomelas (Black-backed Jackal)

Slater, G.J., E.R. Dumont and B van Valkenburgh. 2009. Implications of predatory specialization for skull form and function in canids. Journal of Zoology, 278:181-188.

Abstract: This project investigated the relationship between cranial form and function in the trophically diverse dog family, Canidae, using linear morphometrics and finite element (FE) analyses that simulated the internal and external forces that act on the skull during the act of prey capture and killing. In contrast to previous FE-based studies, we compared models using a newly developed method that removes the effects of size and highlights the relationship between shape and performance. Cranial shape varies among canids based on diet, and different selective forces presumably drove evolution of these phenotypes. The long, narrow jaws of small prey specialists appear to reflect selection for fast jaw closure at the expense of bite force. Generalists have intermediate jaw dimensions and produce moderate bite forces, but their crania are comparable in strength to those of small prey specialists. Canids that take large prey have short, broad jaws, produce the largest bite forces and possess very strong crania. Our FE simulations suggest that the remarkable strength of skulls of large prey specialists reflect the additional ability to resist extrinsic loads that may be encountered while struggling with large prey items.

Graham Slater's research page: www.eeb.ucla.edu/gslater

Figure 2: Example of FE model project page at Biomesh.org

3.2.3. Existing Ontologies Leveraged/Imported

The Biomesh Ontology was created by interlinking reference ontologies from the various domains that were covered. Here, we discuss the reference ontologies that were utilized.

This work has benefited from the efforts of the NCBO BioPortal and the OBO since it borrows several reference ontologies from the biology domain. It was necessary to store information about the anatomy of each model and material test. The most popular ontology for anatomy, the Foundational Model of Anatomy (FMA) [71], is beyond the scope of this chapter and only deals with the anatomy of humans. It was necessary to find an ontology that could describe the anatomy of all plants and animals in a minimal fashion. Therefore, the choice was made to use the Minimal Anatomical Terminology (MAT) ontology, which is acceptable for use with any organism [72]. Although this terminology does not sport the restrictive properties that a robust ontology does, its 461 classes were found to be adequate for providing an annotative description of the anatomy of the materials and models for this project.

Species classification information also needed to be captured for each model and material test. Therefore, the National Center for Biotechnology Information (NCBI) organismal classification taxonomy was investigated [73]. The 847,760 classes of this taxonomy provide the structure for complete biological classification of every organism following a Linnaean taxonomic system. However, it is similar to MAT in that it does not provide restrictive properties on classes and therefore does not classify as a rigid ontology.

It was also necessary to store information about each finite-element model, such as the element type used, loading types, material treatment and properties, material test results, constraints, and modeling assumptions. This was done by leveraging another pre-existing ontology called the Engineering Analysis Models (EAM) ontology which was created as part of the e-Design Framework at the University of Massachusetts Amherst [7]. This ontology contains 19 classes and 16 properties which allow deep description of the finite-element models in the Biomesh database. The e-Design Framework also provides adequate classification of materials and material property data. The NASA Units Ontology was also imported so that a semantic information model could be provided for the units used in each model, such as units used for material property values [74]. The NASA Units Ontology includes 12 classes, 8 properties, and about 100 individual units, therefore making it a very rigid and robust ontology.

3.2.4. Scope and Limitations

This chapter shows an application of interlinking ontologies from disparate domains within engineering and biology. It outlines the formal process used for this interlinking which can be followed for other applications. It is meant to explore the benefits of well defined, curated ontologies that exist in the biology domain but are rarer in engineering. Due to the large size of the full Biomesh database, the ontology created for the case study in this chapter does not include all classes and instances that would be needed to cover the full database. Instead, it focuses on showing the usefulness of the integration to reveal knowledge that was formerly lost in the disparate databases.

3.3. Interlinking Cross-Domain Ontologies

3.3.1. Objective

As stated previously, we are presenting the formal process used to create an ontology for a cross-domain application, namely for the interlinking of biological materials and finite-element (FE) models which were created as part of the Biomesh project (www.biomesh.org). There is an opportunity to improve KM in engineering by leveraging on the success of KM in biology. The link between biology and engineering will be created by using ontologies from multiple domains to create a knowledge repository for the models and materials.

In order to demonstrate the usefulness of reference ontologies and sharing repositories, a specific use case is addressed from the Biomesh project. Biomesh currently possesses about 30 finite element models and over 800 material properties relating to various anatomical structures of various species in its ever-growing database. The models and materials currently reside in separate databases, although they may share certain information. For example, a particular model may use material properties that are located in the material database. Previously, this link was hidden and could only be deduced by reading the textual description of the model. The following demonstrates the methods used to expose this link.

3.3.2. Formal Procedure

First, the domains covered by the project were identified. Reputable repositories were searched for existing ontologies that covered the domains in appropriate levels of granularity. Once the set of ontologies was identified, they were imported into *Protégé* 3.4.8. In the case of this chapter, some of the ontologies were imported using BioPortal and others using the “Import Ontology” feature of *Protégé*. This allowed for namespace management which mitigated naming conflicts. Once all ontologies were successfully imported into *Protégé*, consistency checks allowed for identification of class conflicts. If consistency checks were passed, the hybrid ontology could be manipulated to fulfill the needs of the application. Properties which would link the previously disparate domains together were added. Finally, the hybrid ontology could be instantiated with the information from the original sources.

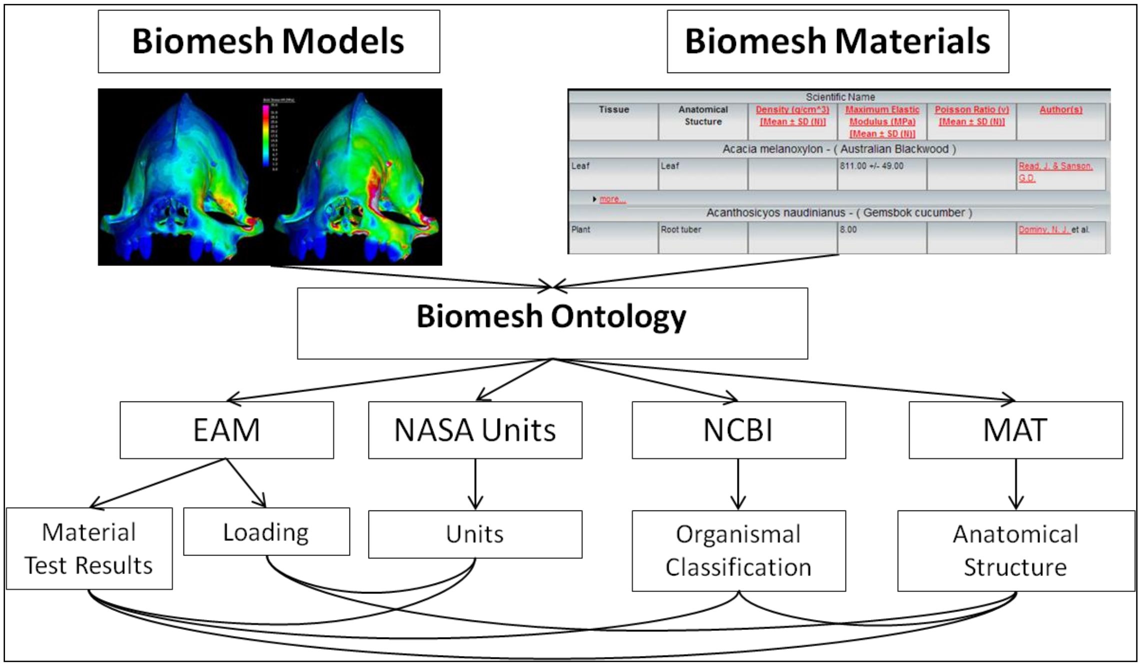
3.3.3. Application-Specific Procedure

In this case, there was data, information, and knowledge related to FE models, anatomy, species classification, and materials. The EAM ontology and parts of the e-Design Framework were imported to store FE model and material property knowledge.

MAT was imported to store anatomical knowledge. NCBI was imported to store species classification knowledge. The NASA Units Ontology was imported to store knowledge about units for material properties. A more extensive justification for the ontologies imported for this project can be found in Section 3.2.2. The ontologies from these disparate domains were imported into one hybrid, interlinked ontology, called the Biomesh Ontology. Figure 3 shows the previously disparate model and material databases that currently exist for the Biomesh project which are now integrated into the Biomesh Ontology by the linking of concepts between imported ontologies.

Protégé was used to create the hybrid ontology. This particular ontology editor is linked to the BioPortal, which is a repository of ontologies maintained by the NCBO. It is possible to import specific classes from the BioPortal repository by simply selecting them in a graphical window. This allowed the direct importation of the NCBI organismal classification ontology, as well as the MAT ontology. Due to the size of the NCBI organismal classification ontology (over 800,000 classes), only small subsections of it were imported that would demonstrate its usefulness. Similarly, only classes relating to animal and plant anatomy were imported from the MAT ontology.

Since the BioPortal tool does not allow importation of many properties (likely because of its flexibility with class importation and the possible conflicts this could cause), properties needed to be added to the NCBI and MAT ontologies to extend them for our particular application. An opportunity for improvement was seen in the NCBI ontology, where restrictive Boolean properties could be added to species classification levels to enforce the principles of biological classification. For example, a Boolean property ***hasHair*** (note: all properties will be *italicized* and **bold**) with a restricted value of “True” was added to the class **Mammalia** (note: all class names will be in **Bold**), since all mammals have hair. This way, an organism without hair, which would carry the value “False” for the ***hasHair*** property, could not be mistakenly classified as a mammal.



Scientific Name					
Tissue	Anatomical Stucture	Density (g/cm^3) [Mean ± SD (N)]	Maximum Elastic Modulus (MPa) [Mean ± SD (N)]	Poisson Ratio (v) [Mean ± SD (N)]	Author(s)
Acacia melanoxylon - (Australian Blackwood)					
Leaf	Leaf		811.00 +/- 49.00		Read, J. & Sanson, G.D.
▸ more...					
Acanthosicyos naudinianus - (Gemsbok cucumber)					
Plant	Root tuber		8.00		Dominy, N. J. et al.

Figure 3: Merging of Biomesh model and material databases into the Biomesh Ontology

Once the ontologies from BioPortal were successfully imported, it was necessary to import the NASA Units ontology. This was done as a full import, including all classes, properties, and instances. This ontology is very robust and no editing was needed.

The EAM ontology from the e-Design Framework was imported next. This was also done as a full import of classes and properties. This ontology contains classes and properties relating to materials as well as engineering analysis models.

The Biomesh Ontology now contained classes and properties from four different reference ontologies: NCBI, MAT, NASA Units, and EAM. Classes were added at the top level to host People and Publications related to the other instances that would reside in the ontology. Once all ontologies were successfully imported and the classes were finalized, it was necessary to add properties to create links between concepts that needed to be related. As an example, the multiple domains of the property ***hasMaterial*** tie a material to the organismal classification (NCBI), anatomy (MAT), and model (EAM) ontologies.

Consistency checks using the Pellet Reasoner [75] in *Protégé* 3.4.8 revealed no conflicting facts in the Biomesh Ontology. Now, the Biomesh Ontology was ready to be instantiated with the knowledge from the case study. This is demonstrated in the following section.

3.4. Case Study: *Macaca Fascicularis* in the Biomesh Ontology

The Biomesh project currently stores information about biological FE models and materials in separate databases. There are instances where an FE model exists in the database and uses material properties that are stored in a separate database. As an example this occurs with the crab-eating macaque (*Macaca fascicularis*). This particular monkey species is native to Southeast Asia and is widely studied in laboratories due to its close physiology to humans [76].

First, all major instances had to be created in the ontology. *Macaca fascicularis* had to be classified in the NCBI organismal classification ontology, so the instance “NCBI_Macaca_fascicularis_1” (note: all instances will be in “quotes”) was created to reside in the species level class Macaca_fascicularis**. Next, the instance “MAT_Macaca_fascicularis_cranium” was created in the** Cranium **class of the MAT ontology. This process continued until the ontology was fully populated with this example.**

Table 1 below shows the main instances that were created along with the reference ontologies in which they reside.

Once instantiation of the ontology was complete, datatype and object properties had to be given values. Datatype properties contain values such as strings, integers, and Booleans. Object properties related instances to one another. This is where the connection was created between instances which had been unrelated in the previous database. The species *Macaca fascicularis* was connected to its anatomy (the cranium in this case), its FE model, the material properties that were used in that model, and the publication reference using the properties ***hasAnatomy***, ***hasModel***, ***hasMaterialTest***, and ***hasPaper***, respectively. This is displayed in Figure 4 below, where classes are shown with yellow circles, and instances with purple diamonds. The connecting lines between classes display the subclass relationship, and the connecting lines between instances represent object property relationships. Note the similarity between Figure 3 and Figure 4, where Figure 3 represents the theoretical goal of the project and Figure 4 represents the results.

Table 1: Main instances in the Biomesh Ontology

Instance	**Class**	**Reference Ontology**	**Description**
“NCBI_Macaca_fascicularis_1”	**Macaca_fascicularis**	NCBI	Species instance of macaque classified in NCBI

“MAT_Macaca_fascicularis_cranium”	**Cranium**	MAT	Anatomical instance of macaque cranium
“EAM_Static_Tests_Macaca_fascicularis_cranium”	**Static_Tests**	EAM	Material property instance for the macaque cranium
“EAM_Finite_Element_Models_ Macaca_fascicularis”	**Finite_Element_Models**	EAM	Model instance of macaque cranium
“pascal”	**ComplexUnit**	NASA Units	Units of measure of elastic modulus in macaque cranium
“Modeling_Elastic_Properties _in_Finite-Element_Analysis”	**Paper**	None	Reference publication where model came from
“Dechow_P”	**People**	None	Author of the publication for model
“Elastic_Properties_of_External_ Cortical_Bone_in_the_Craniofacial_ Skeleton_of_the_Rhesus_Monkey”	**Paper**	None	Reference publication for material properties
“Strait_D”	**People**	None	Author of publication for material properties

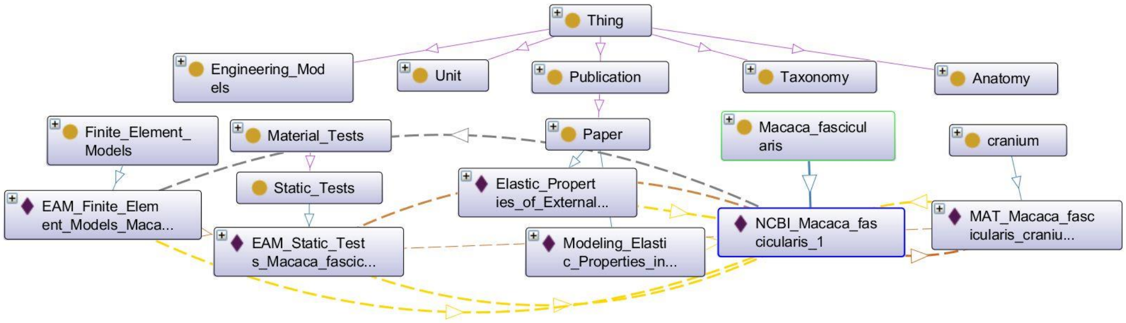

Thing
Engineering_Models
Unit
Publication
Taxonomy
Anatomy
Finite_Element_Models
Material_Tests
Paper
Macaca_fascicularis
cranium
Static_Tests
Elastic_Properties_of_External...
EAM_Finite_Element_Models_Maca...
EAM_Static_Tests_Macaca_fascic...
Modeling_Elastic_Properties_in...
NCBI_Macaca_fascicularis_1
MAT_Macaca_fascicularis_craniu...

Figure 4: Visualization of Biomesh Ontology in *Protégé*’s OntoGraph

Now that the databases are connected in the Biomesh Ontology, queries and inferences can be performed that span across the previously disparate biological materials and FE models. For example, a powerful example of an inference was performed using a SWRL rule [77]. SWRL rules can be used to infer relationships between individuals using properties and classes. In our example, we wanted to populate a class with all individuals in the ontology which fell into a very specific category. A SWRL rule was written which placed all FE models of mammalian crania which assumed simple isotropic material properties in a class called **MammilianCraniumIsotropicFEModels**. The code for the SWRL rule is as follows:

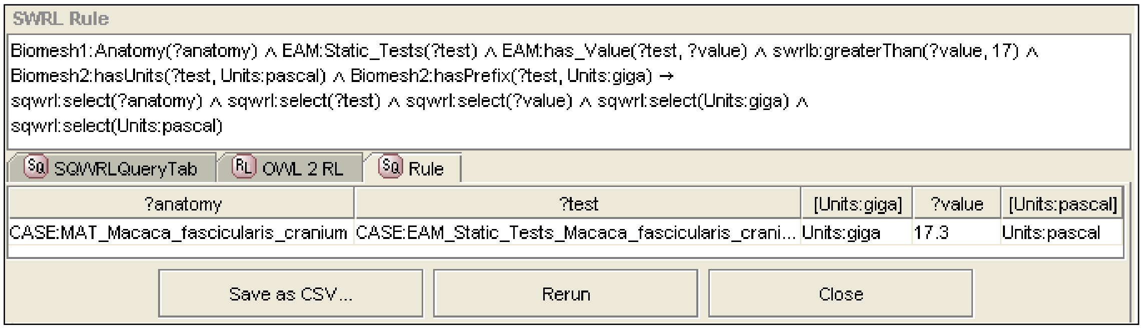
```
EAM:Finite_Element_Models(?model)                                   ∧
Biomesh1:Mammalia(?organism)      ∧      cranium(?anatomy)          ∧
Biomesh1:hasOrganism(?model,             ?organism)                 ∧
Biomesh2:hasAnatomy(?model,              ?anatomy)                  ∧
EAM:has_Assumptions(?model, CASE:Simple_Isotropy) →
CASE:MammilianCraniumIsotropicFEModels(?model)
```

When this rule is applied to the ontology in our case study, one model becomes an inferred instance in the class **MammilianCraniumIsotropicFEModels**. This is a powerful inference because it spans the gap between the previously disparate databases. An FE model which pertains to a certain species and anatomical structure with a particular material property assumption is revealed immediately to the user. This is knowledge that *could not have been uncovered automatically* by the old database. Thus, we are able to use *machine reasoning* to reveal *new knowledge*. This is especially powerful in massive knowledge bases which have many thousands of instances. Automated reasoning is invaluable when it comes to developing scalable knowledge bases.

Additionally, queries can be performed in the SQWRL Rule [78] tab in *Protégé*. These queries can reveal knowledge stored in the Biomesh Ontology. An example was investigated where a researcher was looking for an anatomical structure that had a Young's modulus (E) greater than 17 Gigapascals (GPa). This static material test result is located in the **Material_Tests** class of the ontology. By typing this query in as shown in Figure 5, the researcher can see that one instance in the ontology has E>17 GPa, and that it is the cranium of *Macaca fascicularis*. This query strongly relies on the meaning contained in the knowledge base, as it ensures that we are seeing results that not only have the correct value but the correct units. This shows the value gained from the importation of the NASA Units Ontology.

Simpler queries can also be conducted using *Protégé*'s built in query builder. For example, if a researcher is looking at cranial morphology in crab-eating macaques, they may want to check if any FE models or material properties exist that could be reused. This researcher may also have some specific preferences as to which models they want to see, e.g. only models which assume simple isotropy. Using the Biomesh Ontology, a simple query can reveal this information. Figure 6 shows this example, where the first line of the query suggests that we are looking for an FE model which is related to the organism *Macaca fascicularis*. The second line of the query restricts the anatomy of the models, since we only want to see FE models of the cranium of this species. Since we are also interested in materials properties, the third line of the query looks for material test results that are related to the species *Macaca fascicularis*.

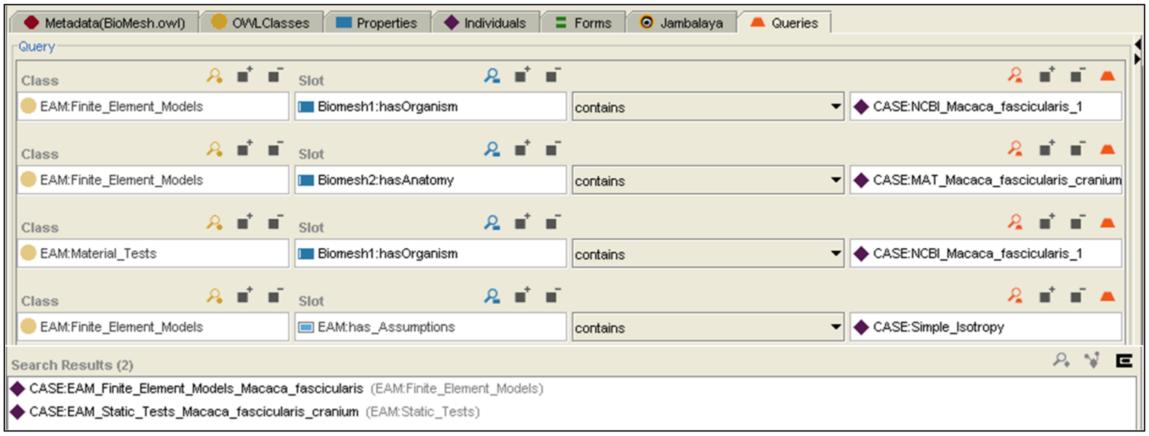
The Biomesh Ontology is currently being shared on the University of Massachusetts Center for e-Design website at (http://edesign.ecs.umass.edu/ontology-downloads/).



```
Biomesh1:Anatomy(?anatomy) ∧ EAM:Static_Tests(?test) ∧ EAM:has_Value(?test, ?value) ∧ swrlb:greaterThan(?value, 17) ∧
Biomesh2:hasUnits(?test, Units:pascal) ∧ Biomesh2:hasPrefix(?test, Units:giga) →
sqwrl:select(?anatomy) ∧ sqwrl:select(?test) ∧ sqwrl:select(?value) ∧ sqwrl:select(Units:giga) ∧
sqwrl:select(Units:pascal)
```

?anatomy	?test	[Units:giga]	?value	[Units:pascal]
CASE:MAT_Macaca_fascicularis_cranium	CASE:EAM_Static_Tests_Macaca_fascicularis_crani...	Units:giga	17.3	Units:pascal

Figure 5: SQWRL query and results from the Biomesh Ontology in *Protégé*



Class	Slot		
EAM:Finite_Element_Models	Biomesh1:hasOrganism	contains	CASE:NCBI_Macaca_fascicularis_1
EAM:Finite_Element_Models	Biomesh2:hasAnatomy	contains	CASE:MAT_Macaca_fascicularis_cranium
EAM:Material_Tests	Biomesh1:hasOrganism	contains	CASE:NCBI_Macaca_fascicularis_1
EAM:Finite_Element_Models	EAM:has_Assumptions	contains	CASE:Simple_Isotropy

Search Results (2)

- CASE:EAM_Finite_Element_Models_Macaca_fascicularis (EAM:Finite_Element_Models)
- CASE:EAM_Static_Tests_Macaca_fascicularis_cranium (EAM:Static_Tests)

Figure 6: Query and results from the Biomesh Ontology in *Protégé*

3.5. Discussion of Results

The goal of this project was to demonstrate the benefit that engineering could gain from the advances that the biology community has made in the field of KM by interlinking ontologies from engineering and biological fields. We have satisfied the requirements set forth by the Biomesh project, which were to 1.) facilitate free sharing of the FE models developed by the Biomesh project, 2.) provide access to the biological materials database, and 3.) link knowledge between materials and models that was previously disconnected. Requirement 1 was satisfied by using an ontology as a means of organization of the models. This is an open access medium which allows quick search and retrieval of the desired model. Requirement 2 was satisfied in the same way. Requirement 3 was realized when the multiple ontologies were linked together so that previously disparate information could be shared between models and materials, and the power of this linkage was demonstrated in our case study.

The existence of the NCBI organismal classification and MAT ontologies greatly reduced the amount of groundwork needed for this project. The ontology builder did not need to be an expert in either of these fields because the ontologies used have already been published and accepted by their domain experts. If this is paralleled in engineering, it would be possible to reduce workload in the KM field significantly. For example, a published and curated Materials Ontology could be used in many engineering fields where information about material properties, types and tests would be stored. The existence of this reference ontology would mean that a researcher would not have to "reinvent the wheel" and could reuse material property data, information, and knowledge. This has already been seen with the NASA Units ontology. Since this ontology had already been created, we did not have to develop our own way to store information about units for this project.

As stated previously, the Biomesh Ontology is currently being shared online. However, it does not represent the entire suite of biological models and materials properties that the Biomesh project has collected. In the future, when the Biomesh Ontology is fully instantiated, a semantic wiki [79] such as *OntoWiki* [14, 15] may be used for online browsing of the ontology. The models could be shared here for investigation and download.

The NCBI and MAT ontologies were both very light on properties when they were imported. However, the addition of properties and class restrictions that was done in this project increased the rigidity and decidability of these ontologies. The EAM and NASA Units ontologies already had many properties and no additions were needed. The decidability of the Biomesh Ontology was demonstrated through the query which bridged across previously disparate databases (Figure 3).

Although the interlinking of ontologies from different domains was fairly seamless in this application, there could be many challenges that could arise. These include naming conflicts, overlapping classes and properties, and variable levels of descriptiveness. The problems were mitigated in this application because of the use of robust reference ontologies, but certainly could pose large problems in areas where a strong ontology building community does not exist.

While this project does demonstrate a step in the right direction in terms of the reuse of reference ontologies, it does not claim to have solved the problem of disjointed KM in the engineering community. To solve this problem, we can build on the lessons learned by consortiums like the OBO and successful ontologies like the GO. This can be seen as an *opportunity* for a standards organization like the National Institute of Standards and Technology (NIST) or the Defense Advanced Research Projects Agency (DARPA). With curated ontologies backed by federal organizations, the full power of the semantic web envisioned by Berners-Lee [23] can be realized.

CHAPTER 4

ONTOLOGY USABILITY STUDY

4.1. <u>Project Summary</u>

This research involves the evaluation of the current state of the usability of ontologies. It has been previously stated that the reason that ontologies are slow to be adopted by the engineering industry as knowledge management tools is that they lack usability. Despite the technical advantages that they offer, the lack of usability is enough to discourage their use. Here, we aimed to figure out why ontologies are not user friendly. We analyzed the most common ontology interface, which is an ontology editing tool, called *Protégé* [80, 81]. We also analyzed an original configuration of the semantic wiki *OntoWiki* that has been selected for this project. Note that *OntoWiki* was studied here as a representative semantic wiki. It is known by the authors that using a different semantic wiki would likely yield altered usability results. The availability of *OntoWiki* at the time that this testing was performed led to its selection over other semantic wikis. The objective of this study was to reveal usability problems within *Protégé* and *OntoWiki* and to possibly reveal areas where one performs better than the other.

4.2. <u>Background</u>

4.2.1. Human-computer Interaction Evaluation Methods

Human-computer interaction (HCI) is a long studied field which has led to massive advances in computing technology. It is defined, quite reciprocally, as the study of the interaction between humans and computers. This has been an important field since computers became widely used in the mid-1970s. Figure 7, from Ali et al. [82], shows the advances in user interfaces (UI's) over time. These advances can be largely credited to the needs identified by the study of HCI.

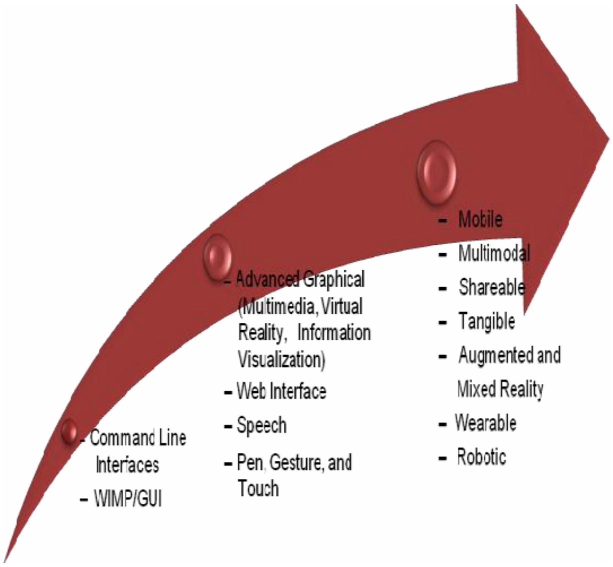

- Command Line Interfaces
- WIMP/GUI
- Advanced Graphical (Multimedia, Virtual Reality, Information Visualization)
- Web Interface
- Speech
- Pen, Gesture, and Touch
- Mobile
- Multimodal
- Shareable
- Tangible
- Augmented and Mixed Reality
- Wearable
- Robotic

Figure 7: Advances in user interface technology over the past 50+ years [82]

Just as important as the study of HCI in general are the evaluation methods of user interfaces. It is not satisfactory to have ad hoc approaches for evaluating interfaces. Instead, step-by-step guided approaches, such as the ones discussed here, will yield quantitative and qualitative results.

In one of the first papers on HCI, Bailey and Pearson [83] developed a definition for computer user satisfaction. This definition was based on the work of psychologists, specifically Wanous and Lawler [84], who suggested the following formula for satisfaction measure:

$$S_i = \sum_{j=1}^{n} R_{ij} W_{ij}$$

where
S_i= The satisfaction of individual i.
n= The total number of factors, where a factor is an evaluation criterion.
R_{ij}= The reaction to the factor j by individual i.
W_{ij}= The importance of factor j to individual i.

Bailey and Pearson developed a questionnaire which was able to help identify usability problems that had been formerly overlooked by programmers. They noticed that when the questionnaire was administered to employees who used software at a company, there was apprehension to be honest because of fear of termination. Therefore, they encouraged anonymity when administering the questionnaire.

Root and Draper [85] tested the usefulness of questionnaires in the human-computer interaction field. They gained insight into the types of questions that would gather the most meaningful information for a questionnaire to evaluate a user interface. They found that a checklist is very useful, and that open-ended questions would allow the tester to input a problem that had not been identified by the administrator. Chin et al. [86] attempted to develop a questionnaire, called the Questionnaire for User Interface Satisfaction (QUIS), which would measure a user's rating of a human-computer interface. They noted that command line systems (e.g. MS-DOS) were given poor ratings, while menu driven applications (e.g. WordPerfect) performed better.

Nielsen and Molich [87] found that a group of heuristic evaluators of a user interface were much more effective at identifying problems than a single evaluator. Using a group of 37 computer science students, they surveyed them and asked them to identify usability problems with a system that was known to have issues. They found that a single evaluator was only able to identify “between 20% and 51% of the usability problems” [87]. By collecting results from between 3 and 5 evaluators, the number of usability problems identified went up substantially. This suggests that when evaluating the usability of a system, a team of investigators should be assembled. It is also mentioned that the investigators should work individually so as to not influence each other’s responses. Using one expert on usability can allow problems to slip through. In a similar paper, Molich and Nielsen [88] looked at the capabilities of designers and programmers at identifying usability problems. 77 designers and programmers from industry and academia tested a system that was known to have 30 usability issues according to the authors’ 9 principles of good human-computer dialogue, which are listed below [88]:

1. Simple and Natural Dialogue
2. Speak the User’s Language
3. Minimize the User’s Memory Load
4. Be Consistent
5. Provide Feedback
6. Provide Clearly Marked Exits
7. Provide Shortcuts
8. Provide Good Error Messages
9. Error Prevention

When the testers struggled to identify the issues independently, without guidance from a method, Molich and Nielsen concluded that designers and programmers are not sufficiently aware of usability issues. They also replicated their findings that “…the more people that look at the interface, the more problems are detected.”[88]

Jeffries et al. [89] discuss the advantages of four different methods of usability evaluation: heuristic evaluation (UI experts look for known issues), software guidelines (similar to heuristic methods, where known issues are identified), cognitive walkthrough (tasks are attempted with the interface), and usability testing (UT) (the interface is studied under real-world conditions). They found that heuristic evaluation identified the highest number of problems. However, they identified that the weakness of heuristic evaluation is the need for a UI specialist. If UI specialists are not available, it is still effective to use guidelines and cognitive walkthroughs [89].

Karat et al. [90] compared the results of empirical testing and cognitive walkthrough methods. They found that empirical methods did a good job identifying more problems than cognitive walkthroughs. In addition, they said that team walkthroughs were more effective than individual ones. In general, Karat et al. suggested that it is most effective to do more than one usability test.

Nielsen and Landauer [91] attempted to create a mathematical model to describe the amount of evaluators needed to achieve desired levels of usability. Using a Poisson process model, it was found that 16 evaluations would produce optimum results based on cost/benefit. Using this method, the number of evaluators needed to catch most usability issues for a project could be calculated before any testing was done.

Lewis analyzed several usability questionnaire types in [92]. Here, he declared that two different types of usability questionnaires, the Post-Study System Usability Questionnaire (PSSUQ) and the Computer System Usability Questionnaire (CSUQ), both are psychometrically sound measures of usability. The PSSUQ and CSUQ consist of very similar questions which measure overall system satisfaction. The only difference between the two questionnaires is in their administration. The PSSUQ is given in a controlled environment and the CSUQ is given as a field test, allowing users to take the test in any environment. The questions for the PSSUQ are listed below. All responses are given on a Likert-type scale [93] as shown in Question 1.

1. Overall, I am satisfied with how easy it is to use this system.
   Strongly Agree 1 2 3 4 5 6 7 **Strongly Disagree**
2. It was simple to use this system.
3. I could effectively complete the tasks and scenarios using this system.
4. I was able to complete the tasks and scenarios quickly using this system.
5. I was able to efficiently complete the tasks scenarios using this system.
6. I felt comfortable using this system.
7. It was easy to learn to use this system.
8. I believe I could become productive quickly using this system.
9. The system gave error messages that clearly told me how to fix problems.
10. Whenever I made a mistake using the system, I could recover easily and quickly.
11. The information (such as on-line help, on-screen messages and other documentation) provided with this system was clear.
12. It was easy to find the information I needed.
13. The information provided for the system was easy to understand.
14. The information was effective in helping me complete the tasks and scenarios.
15. The organization of the information on the system screens was clear.
16. The interface of this system was pleasant.
17. I liked using the interface of this system.
18. This system has all the functions and capabilities I expect it to have.
19. Overall, I am satisfied with this system.

This questionnaire allows the collection of user reactions after using a system to complete a set of tasks. As stated in [92], the data collected using these types of questions is accurate provided an adequate number of users are surveyed. Tullis and Stetson [94] compared several questionnaire types including QUIS and CSUQ when comparing web sites. They found that both methods yield quality results as long as the sample size was around 12-14 participants.

In a very recent work, Nielsen demonstrated the use of a simple evaluation method in [95]. Here, Nielsen tested the usability of a program called GRAVSOFT which is used for gravity field modeling. He described in great detail the methods used to collect data from the users in order to make improvements to the software. A survey questionnaire was administered to the subjects after they complete a set of tasks in GRAVSOFT. This questionnaire used ratings scales from 1-6 with descriptive words on either end of the scale. For example, subjects were asked to answer: “Overall, the software was terrible or wonderful”. The negative word, terrible in this case, was on the bottom end of the scale and was scored with a 1. The positive word, wonderful in this case, was on the top and was scored with a 6. Therefore a higher score meant a more positive response. This same method of collecting feedback from the user is used this chapter as described below.

Here, we discussed four different interface evaluation methods: heuristic evaluation, software guidelines, cognitive walkthrough, and user testing (questionnaire). Of these four general methods, we have chosen to use user testing.

4.2.2. Example Ontology

This study involved the testing of usability of two different programs, *Protégé* and *OntoWiki* , which are designed to edit and display ontologies. In order to test ontology usability, a test ontology needed to be chosen. This ontology would preferably be simple, but would contain a vast amount of information which would make finding specific items challenging.

The ontology used in this project, which is called the Raytheon Document Support (RDS) Ontology, was developed by previous researchers in the University of Massachusetts Amherst Center for e-Design [11]. It was made to store knowledge captured in memorandum documents currently stored in file cabinets by an engineering directorate at Raytheon. Previous work was done to digitize these memorandums and then extract important information from them such as author, recipient, subject, and date. All of this information was then imported in the RDS Ontology. “Dummy data” was used for this project so that no sensitive information from Raytheon was lost. The ontology contained 5,000 memorandums written by 1,000 different authors covering 8 projects. The main function that concerns the users of this ontology is the efficient retrieval of information. This means that the use case will be based on a retrieval task where *Protégé* and *OntoWiki* are evaluated as to their performance in this area.

4.3. User Testing

In order to evaluate the usability of ontologies in both *Protégé* and *OntoWiki*, a user testing questionnaire was designed and administered to subjects at the University of Massachusetts Amherst. Subjects were asked to complete various tasks in each system and then asked to rate their experience. Information such as the time to complete each task and errors encountered were recorded by the proctor of the test. The following subsections give a detailed overview of the testing performed in this chapter.

4.3.1. Subject Population

We calculated the number of subjects needed to notice a large effect and achieve a power of 0.80 using the standard alpha cutoff of 0.05. The power calculation resulted in requiring 15 subjects, but 16 were chosen so that the same number of subjects would initially be exposed to *Protégé* as *OntoWiki* to eliminate the effects of ordering on the data. Approximately 25% of the subjects were between 25-44 years old and the remaining 75% were between 18-24 years old. 41% of the subjects had a background in engineering, 6% had a background in biology, computer science, mathematics, or psychology, and 35% had an unspecified background. Subjects did not need any prior background knowledge about ontologies to complete the tasks for this experiment. No information was gathered about subject gender because it was deemed unnecessary. Subjects were assumed to be competent because they were enrolled as students at the University of Massachusetts Amherst.

4.2.2. Testing Environment

All subjects performed testing in the same controlled environment located in a cubicle complex at the University of Massachusetts Amherst. No subjects were disturbed during testing by outside influences because this was a closed environment. All subjects used the same desktop computer with a Windows 7 operating system. Qualtrics [96] was used to create the task list and survey, both of which can be found in Appendix A. The proctor of the test sat near the subjects to take note of their navigation through each task, as well as to record the amount of time it took the subjects to complete each task. Subjects were not allowed to ask the proctor for assistance on any of the questions.

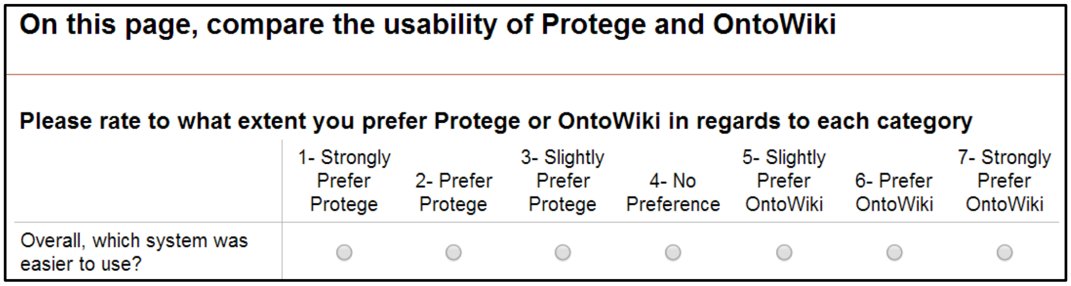
4.3.3. Survey – Tasks and Feedback

Subjects performing this testing completed five tasks in each of two different software applications. All of the tasks involved finding memorandums in the ontology based on different criteria. For example, Task 1 prompted subjects to find any memorandum written by a certain author e.g. John Smith. The tasks became increasingly more difficult, with Task 5 asking subjects to find a memorandum written by a certain author *and* on a certain date, e.g. John Smith on November 4, 1978. A full listing of the tasks can be found in Appendix A.

After subjects completed all of the tasks, they were prompted to provide feedback about each software application they had used. The questions for the survey were customized for this application but were based strongly upon the past work of other researchers in this area. Specifically, the recent work of Nielsen [95] provided a basis for the feedback questions.

There were three sections of user feedback. First there was a comparison of the two systems. Then, subjects rated *Protégé* and *OntoWiki* separately. These ratings were split into two pages, one for each system. The pages were presented to the subjects in random order so that no ordering effects would be observed. Questions in these sections were split into four categories: Overall usability, learning, system capabilities, and open response. The full feedback survey can be found in Appendix A.

Usability of *Protégé* and *OntoWiki* was compared on a relative scale. A preview of this section of the survey can be seen in Figure 8. Here participants compared the usability of the two systems by showing their preference toward one or the other. This was done on a seven-point scale, giving subjects the option to show no preference toward either system. Subjects were also given a chance to provide written open response feedback about their preference toward one system or the other.



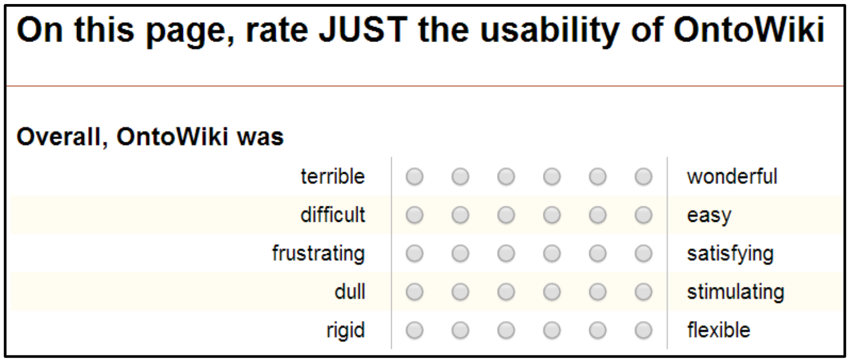
On this page, compare the usability of Protege and OntoWiki

Please rate to what extent you prefer Protege or OntoWiki in regards to each category

	1- Strongly Prefer Protege	2- Prefer Protege	3- Slightly Prefer Protege	4- No Preference	5- Slightly Prefer OntoWiki	6- Prefer OntoWiki	7- Strongly Prefer OntoWiki
Overall, which system was easier to use?	○	○	○	○	○	○	○

Figure 8: Example question from the first section of user feedback

When *Protégé* and *OntoWiki* were evaluated separately, the questions were formatted in a different manner. Figure 9 shows a preview of the “overall usability” section of the *OntoWiki* evaluation page of the survey. The same question format was used to ask subjects to describe their usability experience in terms of learning and system capabilities. Notice that a six-point scale was used to force subjects to pick a positive or negative word to describe their experience. The same strategy was employed by Nielsen in [95].



On this page, rate JUST the usability of OntoWiki

Overall, OntoWiki was

terrible	○	○	○	○	○	○	wonderful
difficult	○	○	○	○	○	○	easy
frustrating	○	○	○	○	○	○	satisfying
dull	○	○	○	○	○	○	stimulating
rigid	○	○	○	○	○	○	flexible

Figure 9: The "overall usability" section of the *OntoWiki* evaluation page

4.4. Results

Subjects rated *Protégé* and *OntoWiki* separately with questions formatted as shown in Figure 9. Responses were given on a 1-6 scale, with 1 being the most negative answer and 6 being the most positive answer. The responses from each subject can be averaged to provide a mean score for each question. A mean score of 3.5 indicates that subjects showed no preference in either direction. A score on either side of this mean shows a tendency in the positive or negative direction. In order to figure out if this tendency is statistically significant, a t-test had to be performed around the constant of 3.5. Those questions with mean response scores lower than 3.5 and with p-values of less than 0.05 were considered to be statistically significant negative responses. Likewise with responses that have mean values greater than 3.5, p-values of less than 0.05 suggest a statistically significant positive response.

Protégé had a mean response score of below 3.5 on 11 out of the 14 questions. It was found that *Protégé* received statistically significant negative scores on 3 out of these 11 questions, most notably "Is *Protégé* designed for all levels of users?" This question had a mean response score of 2.44 and a p-value of 0.0005. The results of this question are displayed in a histogram in Figure 10 below. *Protégé* scored positively (average of above 3.5) on 3 out of the 14 questions, two of which were statistically significant. One question concerned system speed, and the other concerned system reliability. The results of the question involving system reliability in *Protégé* can be seen displayed in a histogram in Figure 11.

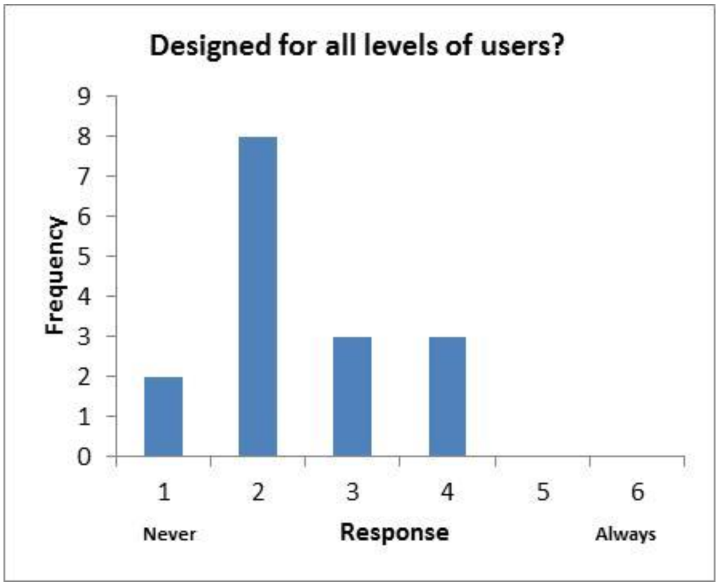


Figure 10: Histogram showing responses to survey question about Protégé: "Designed for all levels of users?"

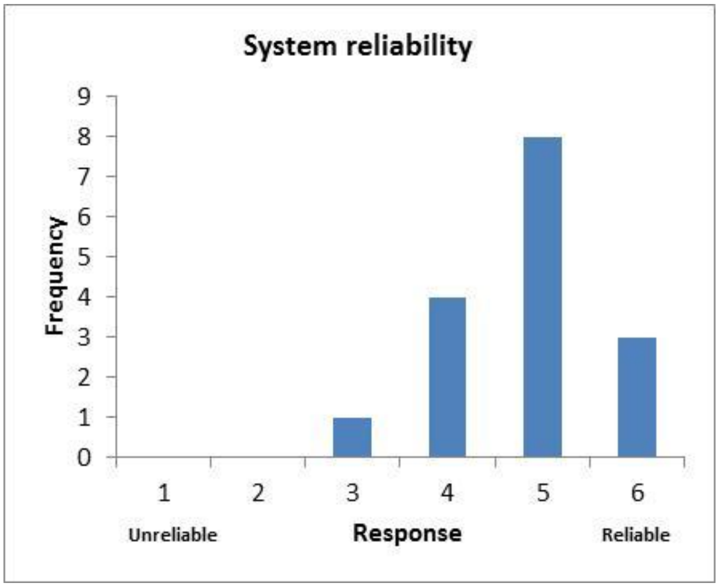


Figure 11: Histogram showing responses to survey question about *Protégé* system reliability

OntoWiki was evaluated in the same way as *Protégé* which was described in the previous paragraph. *OntoWiki* had a mean score below 3.5 on 2 out of 14 questions, neither of which was statistically significant. *OntoWiki* scored positively on the other 12 questions, 5 of which were statistically significant. The question "Remembering names and commands" had the highest p-value for significance over 3.5 and the responses can be seen in the histogram in Figure 12.

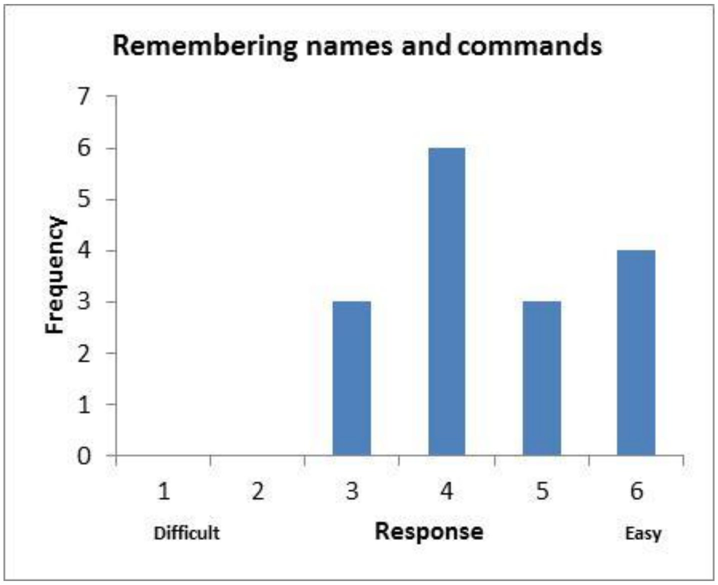


Figure 12: Histogram showing responses to survey question for *OntoWiki*: “Remembering names and commands”

Protégé and *OntoWiki* were also compared on a relative scale as shown previously in Figure 8. A score of 4 in this case showed no preference toward one system or the other. Scores lower than 4 showed preference towards *Protégé* and greater than 4 towards *OntoWiki*. The mean score for every question in this comparison was greater than 4 indicating a preference towards *OntoWiki*. A t-test was used to compare each score to the constant of 4 to test if it was statistically significant. Those questions with a p-value of less than 0.05 were deemed to have scores statistically significantly greater than 4.

Statistical significance of preference towards *OntoWiki* was seen on 2 out of the 12 questions. The first of these questions was “It was simple to use this system”. The second was “It was easy to learn this system”. Histograms of the responses to these questions can be seen in Figure 13 and Figure 14.

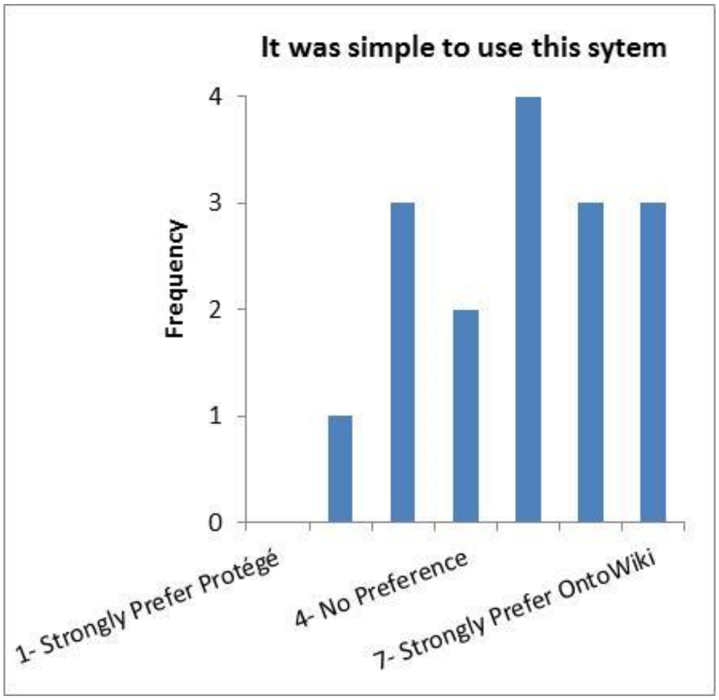


Figure 13: Histogram showing responses to the question "It was simple to use this system"

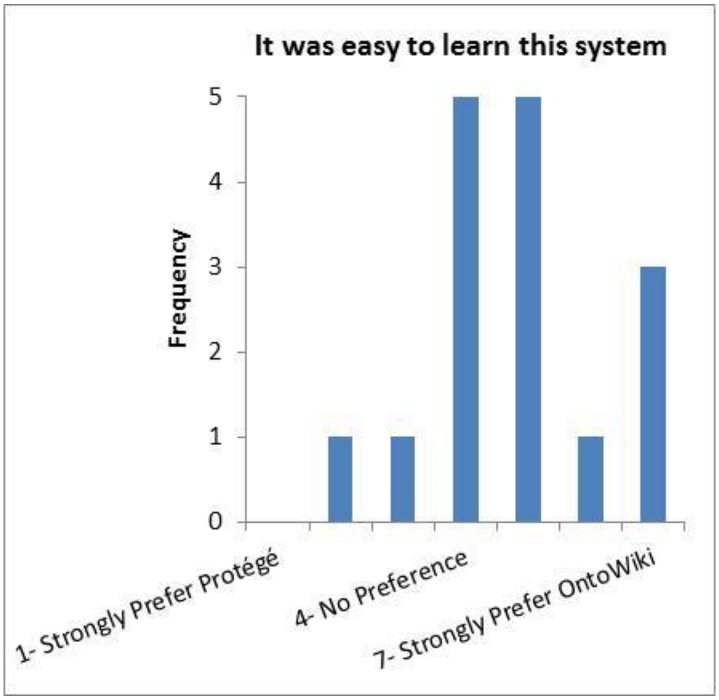


Figure 14: Histogram showing responses to the question "It was easy to learn this system"

Subjects were given the opportunity to point out three problematic and three positive aspects of *Protégé* and *OntoWiki* in the open response section of the survey. We read and coded these responses into 8 categories: Navigation, Speed, Organization, Search, Design, General Use, Learning, and Capabilities. Examples of responses coded into each of these categories can be found in Appendix B. Once coding was complete, the number of responses that fell into each category was counted. The coding results of the positive comments can be seen in Table 2 and negative comments in Table 3.

Table 2: Open response coded positive comments; green highlights were most frequently observed

Positive Comments		
Categories	OntoWiki	Protégé
Navigation	3	1
Speed	1	3
Organization	1	2
Search	6	3
Design	6	7
General Use	5	0
Learning	7	1
Capabilities	1	5

Table 3: Open response coded negative comments; red highlights were most frequently observed

Negative Comments		
Categories	OntoWiki	Protégé
Navigation	2	2
Speed	9	1
Organization	0	3
Search	6	10
Design	5	4
General Use	0	9
Learning	4	1
Capabilities	4	3

4.5. Discussion

The experiment described in the previous section yielded interesting results which are described in this section. As stated in the introduction, the objective of the study performed in this research was to reveal usability problems in both *Protégé* and *OntoWiki* as well as to reveal areas where one system excels over the other. Users preferred *OntoWiki* in the comparative study, and gave it better marks in the individual studies. They also provided more positive and less negative feedback on the open response questions. Here, we discuss the feedback on each system.

Protégé recorded its most positive open response comments in the category of Design. This is not reflected in the quantitative data presented previously. Upon review of the questions in that section, none of them query about design of the system in particular. This could point out a flaw in the survey and suggest that this question should have been asked. The most negative comments on *Protégé* were in the category of Search. Users complained that searching in *Protégé* was not flexible and was hard to learn. *Protégé* received its lowest scores on the question: "Designed for all levels of users?" This is an expected response since *Protégé* was built as an ontology editing tool where experts were the anticipated users. It is not specialized for ontology browsing.

OntoWiki recorded the most positive open response comments in the category of Learning. This was echoed loudly in the quantitative data, where *OntoWiki* scored above a 3.5 on all five questions concerning learning, three of which were statistically significant. *OntoWiki* recorded the most negative comments in the category of Speed. Out of the 16 subjects tested, 9 of them reported that *OntoWiki* was too slow. There was a quantitative question about speed where users reported an average score of 3.9, suggesting that speed was adequate. However, this was considerably lower than *Protégé*’s score of 4.7.

It is important to note that *OntoWiki* was used as a demonstrative semantic wiki. There are many other alternative wikis that exist as discussed in Section 2.2. The results of this testing suggest that semantic wikis are a step in the right direction in terms of ontology usability. However, there are still many areas that can be improved. The subjects who completed this test complained most about the speed of *OntoWiki*. Preliminary testing has been done in the new semantic wiki *DataWiki*, and this system is considerably faster than *OntoWiki*.

CHAPTER 5

RESEARCH SUMMARY

In this thesis, we have investigated ways in which ontologies could be made more appealing to the engineering industry. First, a comprehensive review of the use of ontologies in the engineering design community was presented. Here we discussed the science of ontology, semantic wikis, and the history of knowledge management systems in engineering design.

In Chapter 3 we demonstrated the successful reuse of published, curated reference ontologies from different domains. The simple methods of reference ontology importation showed the benefits of leveraging existing KM systems. The case study presented demonstrated the instantiation and query capabilities of a decidable ontology. While a great amount of progress had been made by the Biomesh project in the collection and sharing of biological models and materials, vast improvements have been made by this project. If the engineering community recognizes the benefits that arise when KM techniques are shared in a communal manner, our domain can move to a better and more interoperable place.

Chapter 4 explained research that was performed to attempt to uncover usability problems in current ontology tools. Protégé, the most popular ontology editing tool, and *OntoWiki*, a semantic wiki, were studied in this project. A study was developed based on an extensive review of past usability studies in the field of HCI. 16 subjects were selected to perform a set of similar tasks in both *Protégé* and *OntoWiki*. They were then asked to rate their experience in terms of usability. It was found that subjects preferred *OntoWiki* over *Protégé* in all categories, but only a small set of the question responses showed statistical significance. When *Protégé* and *OntoWiki* were rated separately, subjects reported that *OntoWiki* was far easier to learn and master. This was echoed in the section of open response questions, where many subjects reported that *OntoWiki*'s learnability was pleasing. It is important to note that the results of this study do not suggest that *OntoWiki* is the perfect solution to the ontology usability problem. Several problems with *OntoWiki* were identified, including system speed.

CHAPTER 6

FUTURE WORK

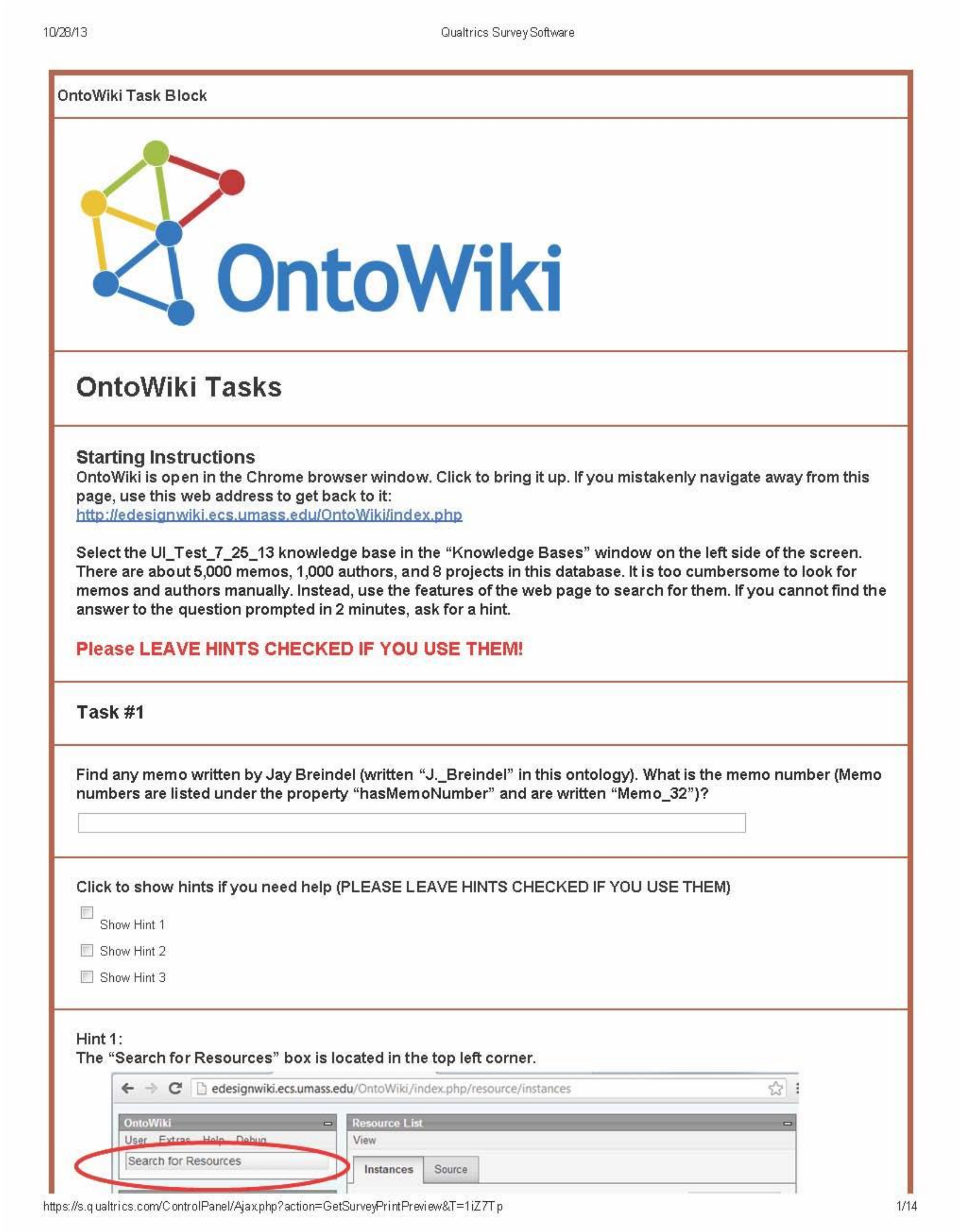
There were certain limitations imposed on the work covered in this thesis in order to make the projects reasonably sized. The project presented in Chapter 3, Interlinking Biological and Engineering Ontologies, only investigated the importation and use of a small portion of the Biomesh database. Had the entire database been imported, more new knowledge could have been formed because the ontology reasoner would have been able to perform inferences over a larger set of information. In the future, it is the hope that this ontology will be used as the main database for the Biomesh Project. This largely hinges on the usability of the ontology so this is strongly linked to the work done in Chapter 4. If customizations are made to a semantic wiki making it easy for operators to populate and search for information, *OntoWiki* or *DataWiki* could be the true front-end for the Biomesh Project. It is also our hope that this could be done for the e-Design Framework of ontologies.

Limitations also affected the work done in Chapter 4 of this thesis. Only one HCI evaluation method was used because resources were not present to hire experts to carry out any of the other methods. Also, subjects had to be selected from a specific population at the University of Massachusetts Amherst in order to adhere to Institutional Review Board regulations. If the testing was repeated in the future, it would be useful use different methods and to bring in subjects from different backgrounds. Additionally, *DataWiki* appears to be the most promising semantic wiki available at this point in time. Performing an individual analysis of this semantic wiki could bring to light some of its usability issues and make it a truly adequate front-end for the Biomesh Project and the e-Design Framework.

It is our hope that the demonstration in Chapter 3 of the usefulness of curated reference ontologies leads to further cultural embrace in the engineering community. As stated previously, this can be seen as an *opportunity* for a standards organization like the National Institute of Standards and Technology (NIST) or the Defense Advanced Research Projects Agency (DARPA). With curated ontologies backed by federal organizations, the full power of the semantic web envisioned by Berners-Lee [23] can be realized.

APPENDIX A

TASKS AND QUESTIONNAIRE



10/28/13 Qualtrics Survey Software

OntoWiki Task Block

OntoWiki Tasks

Starting Instructions
OntoWiki is open in the Chrome browser window. Click to bring it up. If you mistakenly navigate away from this page, use this web address to get back to it: http://edesignwiki.ecs.umass.edu/OntoWiki/index.php

Select the UI_Test_7_25_13 knowledge base in the "Knowledge Bases" window on the left side of the screen. There are about 5,000 memos, 1,000 authors, and 8 projects in this database. It is too cumbersome to look for memos and authors manually. Instead, use the features of the web page to search for them. If you cannot find the answer to the question prompted in 2 minutes, ask for a hint.

Please LEAVE HINTS CHECKED IF YOU USE THEM!

Task #1

Find any memo written by Jay Breindel (written "J._Breindel" in this ontology). What is the memo number (Memo numbers are listed under the property "hasMemoNumber" and are written "Memo_32")?

Click to show hints if you need help (PLEASE LEAVE HINTS CHECKED IF YOU USE THEM)

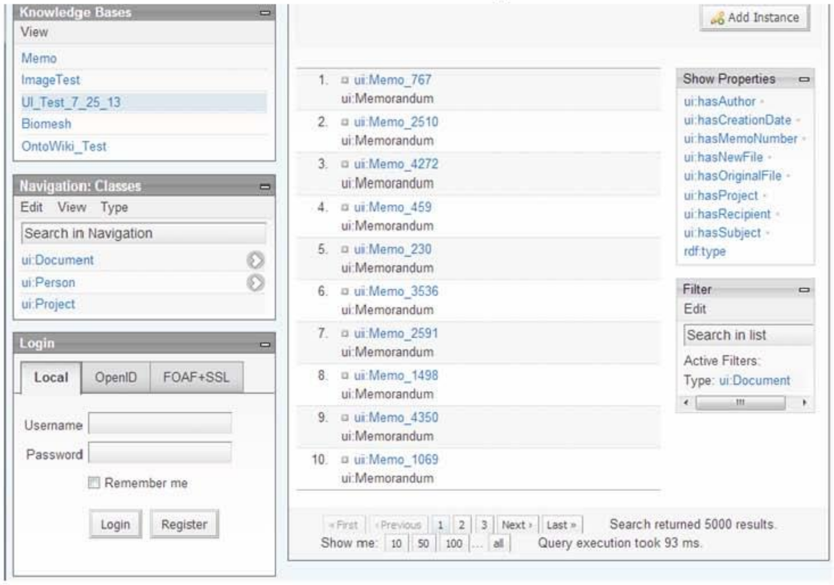
☐ Show Hint 1

☐ Show Hint 2

☐ Show Hint 3

Hint 1:
The "Search for Resources" box is located in the top left corner.

https://s.qualtrics.com/ControlPanel/Ajax.php?action=GetSurveyPrintPreview&T=1iZ7Tp 1/14

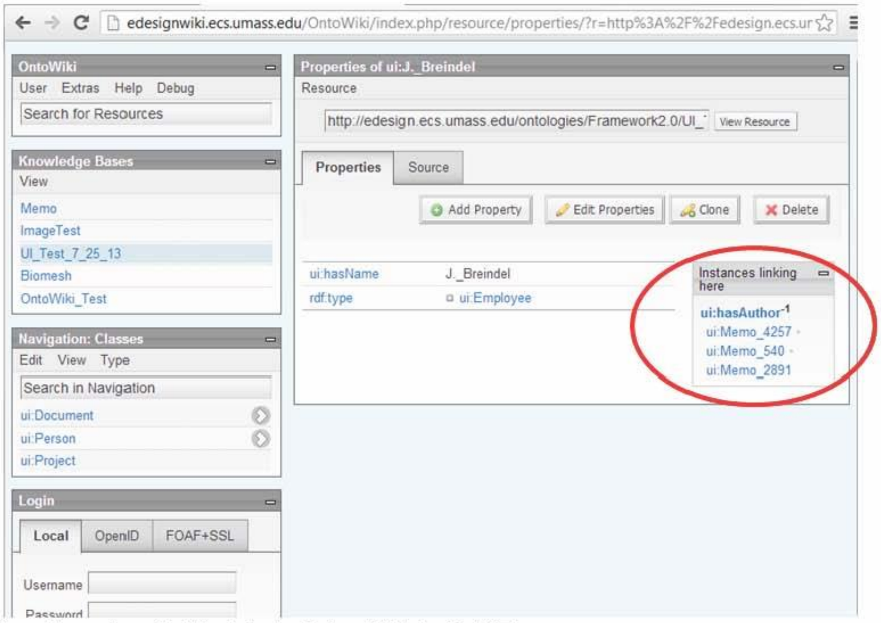


1. ui:Memo_767 ui:Memorandum
2. ui:Memo_2510 ui:Memorandum
3. ui:Memo_4272 ui:Memorandum
4. ui:Memo_459 ui:Memorandum
5. ui:Memo_230 ui:Memorandum
6. ui:Memo_3536 ui:Memorandum
7. ui:Memo_2591 ui:Memorandum
8. ui:Memo_1498 ui:Memorandum
9. ui:Memo_4350 ui:Memorandum
10. ui:Memo_1069 ui:Memorandum

Search returned 5000 results.
Query execution took 93 ms.

Hint 2:
Memo's that Jay wrote link to his page.

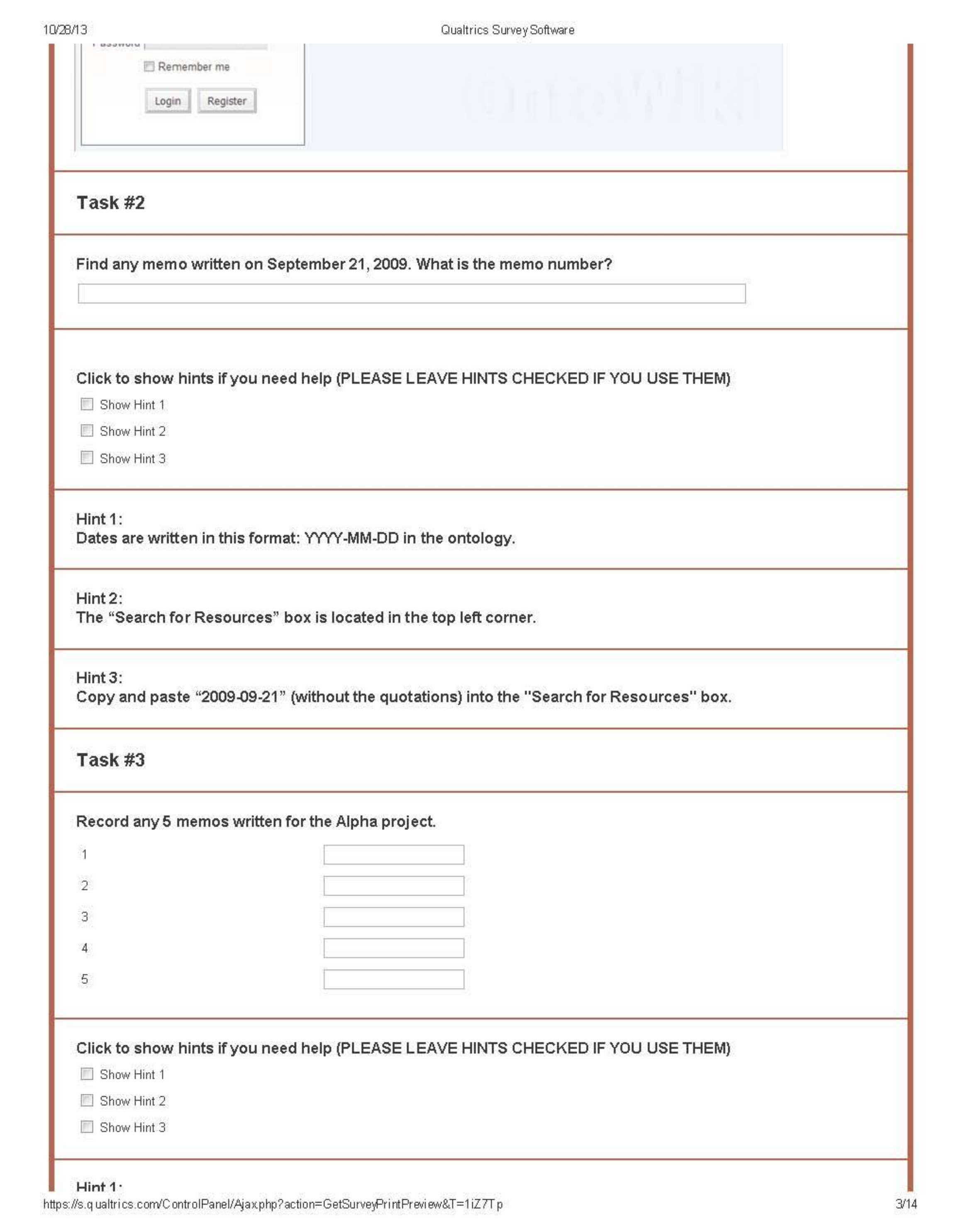
Hint 3:
The "Instances linking here" box on the right of Jay's page shows all memos to which Jay is related.



http://edesign.ecs.umass.edu/ontologies/Framework2.0/UI_

ui:hasName	J._Breindel
rdf:type	ui:Employee

ui:hasAuthor^{-1}
ui:Memo_4257
ui:Memo_540
ui:Memo_2891



Task #2

Find any memo written on September 21, 2009. What is the memo number?

Click to show hints if you need help (PLEASE LEAVE HINTS CHECKED IF YOU USE THEM)

□ Show Hint 1

□ Show Hint 2

□ Show Hint 3

Hint 1:
Dates are written in this format: YYYY-MM-DD in the ontology.

Hint 2:
The "Search for Resources" box is located in the top left corner.

Hint 3:
Copy and paste "2009-09-21" (without the quotations) into the "Search for Resources" box.

Task #3

Record any 5 memos written for the Alpha project.

1

2

3

4

5

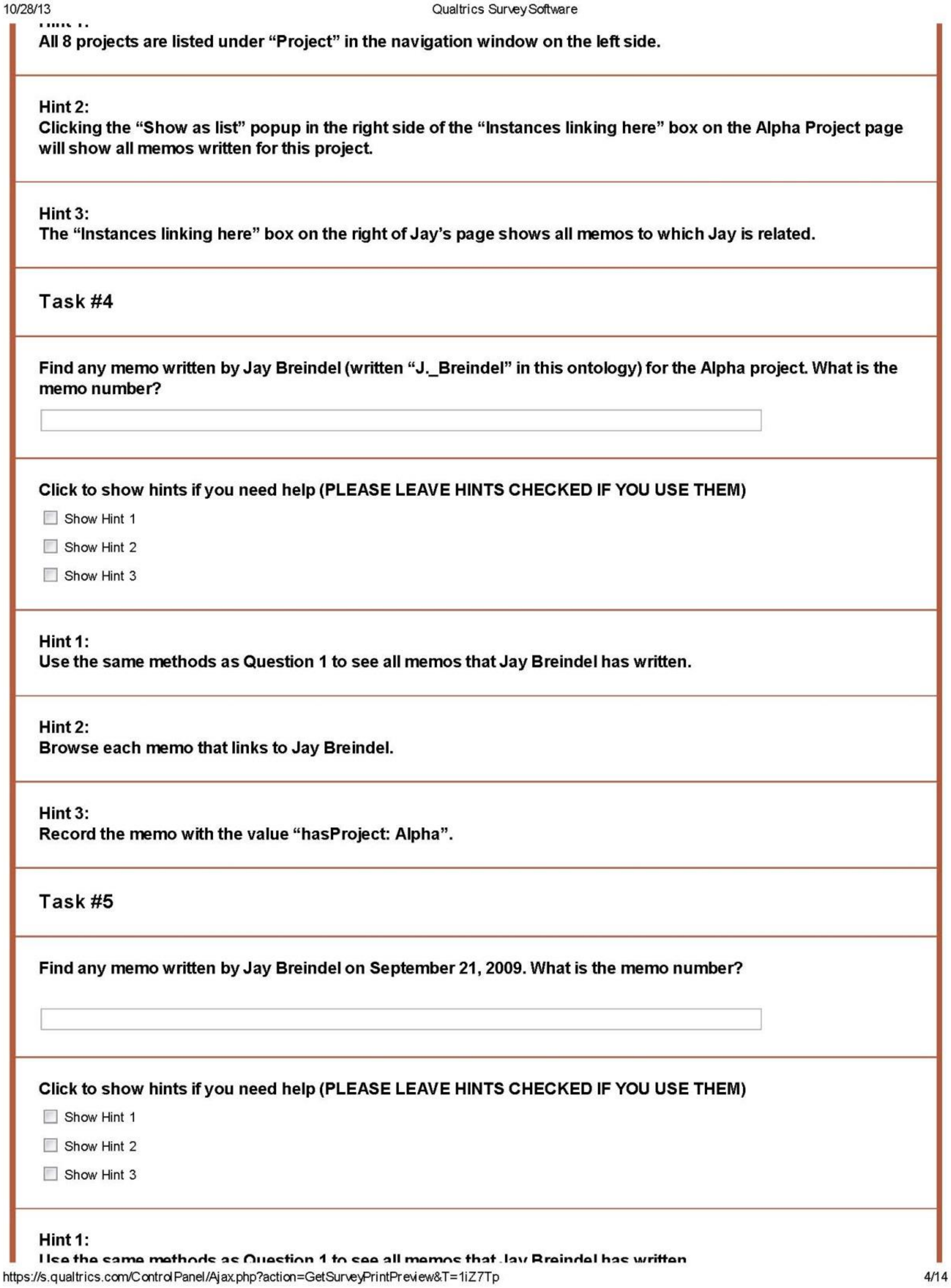
Click to show hints if you need help (PLEASE LEAVE HINTS CHECKED IF YOU USE THEM)

□ Show Hint 1

□ Show Hint 2

□ Show Hint 3

Hint 1:



10/28/13 Qualtrics Survey Software

Hint 1:
All 8 projects are listed under “Project” in the navigation window on the left side.

Hint 2:
Clicking the “Show as list” popup in the right side of the “Instances linking here” box on the Alpha Project page will show all memos written for this project.

Hint 3:
The “Instances linking here” box on the right of Jay’s page shows all memos to which Jay is related.

Task #4

Find any memo written by Jay Breindel (written “J._Breindel” in this ontology) for the Alpha project. What is the memo number?

Click to show hints if you need help (PLEASE LEAVE HINTS CHECKED IF YOU USE THEM)

□ Show Hint 1

□ Show Hint 2

□ Show Hint 3

Hint 1:
Use the same methods as Question 1 to see all memos that Jay Breindel has written.

Hint 2:
Browse each memo that links to Jay Breindel.

Hint 3:
Record the memo with the value “hasProject: Alpha”.

Task #5

Find any memo written by Jay Breindel on September 21, 2009. What is the memo number?

Click to show hints if you need help (PLEASE LEAVE HINTS CHECKED IF YOU USE THEM)

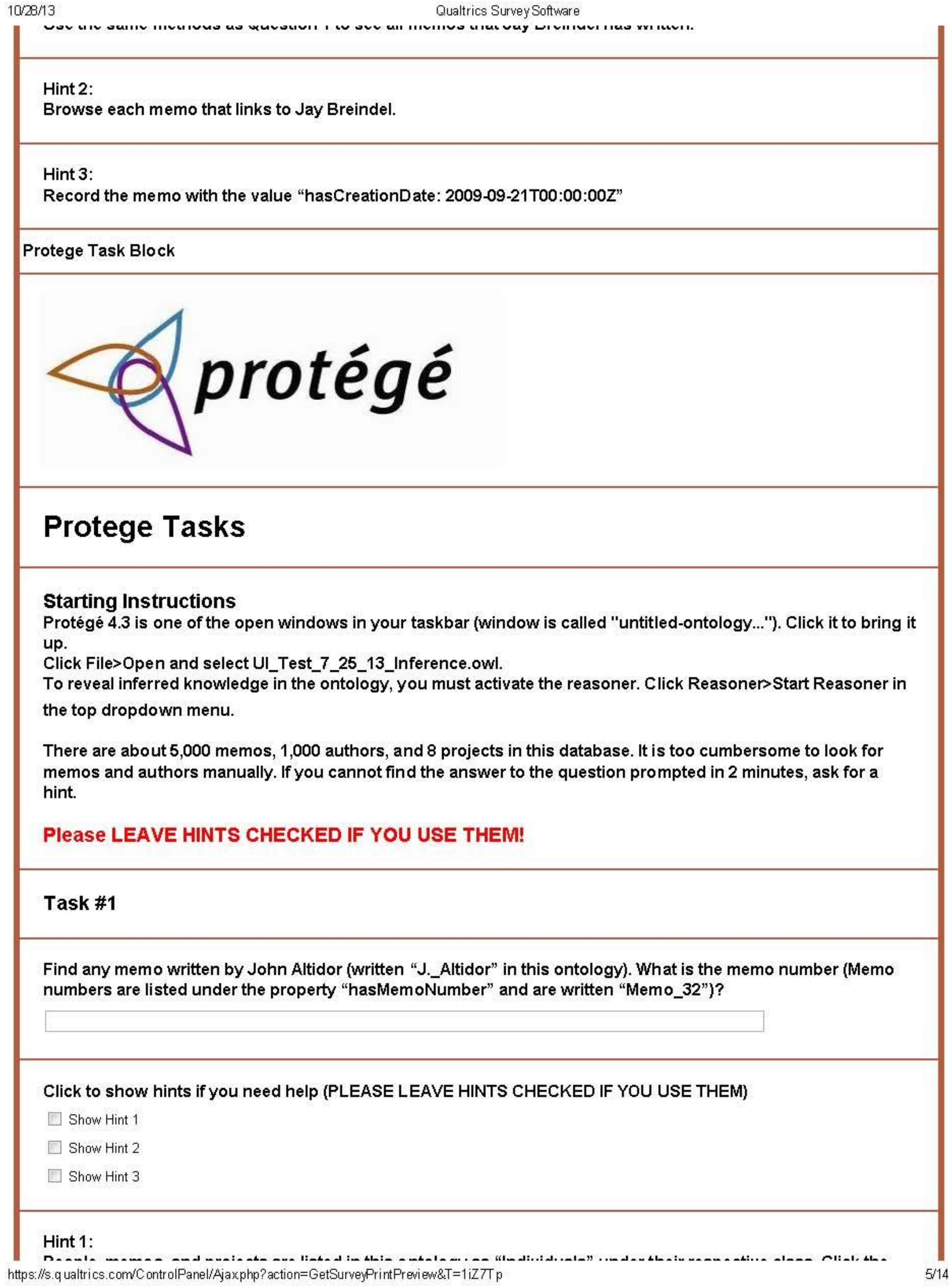
□ Show Hint 1

□ Show Hint 2

□ Show Hint 3

Hint 1:
Use the same methods as Question 1 to see all memos that Jay Breindel has written.

https://s.qualtrics.com/ControlPanel/Ajax.php?action=GetSurveyPrintPreview&T=1iZ7Tp 4/14



Use the same methods as question 1 to see all memos that Jay Breindel has written.

Hint 2:
Browse each memo that links to Jay Breindel.

Hint 3:
Record the memo with the value “hasCreationDate: 2009-09-21T00:00:00Z”

Protege Task Block

Protege Tasks

Starting Instructions

Protégé 4.3 is one of the open windows in your taskbar (window is called "untitled-ontology..."). Click it to bring it up.
Click File>Open and select UI_Test_7_25_13_Inference.owl.
To reveal inferred knowledge in the ontology, you must activate the reasoner. Click Reasoner>Start Reasoner in the top dropdown menu.

There are about 5,000 memos, 1,000 authors, and 8 projects in this database. It is too cumbersome to look for memos and authors manually. If you cannot find the answer to the question prompted in 2 minutes, ask for a hint.

Please LEAVE HINTS CHECKED IF YOU USE THEM!

Task #1

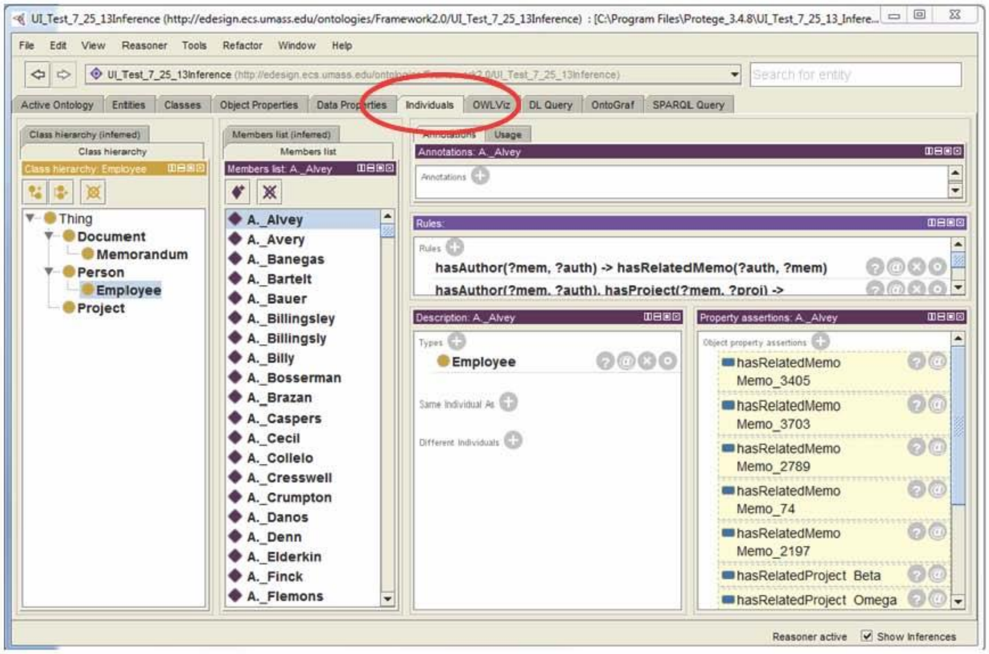
Find any memo written by John Altidor (written “J._Altidor” in this ontology). What is the memo number (Memo numbers are listed under the property “hasMemoNumber” and are written “Memo_32”)?

Click to show hints if you need help (PLEASE LEAVE HINTS CHECKED IF YOU USE THEM)

- ☐ Show Hint 1
- ☐ Show Hint 2
- ☐ Show Hint 3

Hint 1:
People, memos, and projects are listed in this ontology as “Individuals” under their respective class. Click the

People, memos, and projects are listed in this ontology as "Individuals" under their respective class. Click the "Individuals" tab to browse all individuals.



UI_Test_7_25_13Inference (http://edesign.ecs.umass.edu/ontologies/Framework2.0/UI_Test_7_25_13Inference) : [C:\Program Files\Protege_3.4.8\UI_Test_7_25_13_Infere...

UI_Test_7_25_13Inference (http://edesign.ecs.umass.edu/ontologies/Framework2.0/UI_Test_7_25_13Inference)

Class hierarchy: Employee

- Thing
  - Document
    - Memorandum
  - Person
    - Employee
  - Project

Members list: A._Alvey

A._Alvey
A._Avery
A._Banegas
A._Bartelt
A._Bauer
A._Billingsley
A._Billingsly
A._Billy
A._Bosserman
A._Brazan
A._Caspers
A._Cecil
A._Collelo
A._Cresswell
A._Crumpton
A._Danos
A._Denn
A._Elderkin
A._Finck
A._Flemons

Annotations: A._Alvey

Rules:

```
hasAuthor(?mem, ?auth) -> hasRelatedMemo(?auth, ?mem)
hasAuthor(?mem, ?auth), hasProject(?mem, ?proj) ->
```

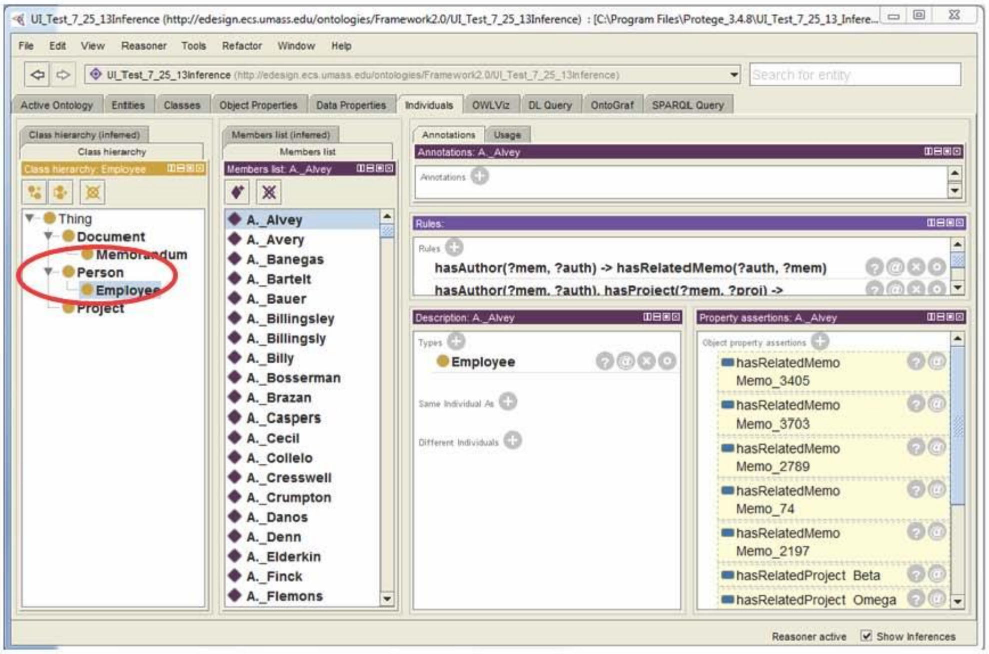
Description: A._Alvey

Types: Employee

Property assertions: A._Alvey

hasRelatedMemo Memo_3405
hasRelatedMemo Memo_3703
hasRelatedMemo Memo_2789
hasRelatedMemo Memo_74
hasRelatedMemo Memo_2197
hasRelatedProject Beta
hasRelatedProject Omega

Hint 2:
John Altidor is an Employee, so click the Person>Employee class. People are listed alphabetically.


UI_Test_7_25_13Inference (http://edesign.ecs.umass.edu/ontologies/Framework2.0/UI_Test_7_25_13Inference) : [C:\Program Files\Protege_3.4.8\UI_Test_7_25_13_Infere...
File Edit View Reasoner Tools Refactor Window Help
UI_Test_7_25_13Inference (http://edesign.ecs.umass.edu/ontologies/Framework2.0/UI_Test_7_25_13Inference)
Search for entity
Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query OntoGraf SPARQL Query
Class hierarchy (inferred)
Class hierarchy
Class hierarchy: Employee
Thing
Document
Memorandum
Person
Employee
Project
Members list (inferred)
Members list
Members list: A._Alvey
A._Alvey
A._Avery
A._Banegas
A._Bartelt
A._Bauer
A._Billingsley
A._Billingsly
A._Billy
A._Bosserman
A._Brazan
A._Caspers
A._Cecil
A._Collelo
A._Cresswell
A._Crumpton
A._Danos
A._Denn
A._Elderkin
A._Finck
A._Flemons
Annotations Usage
Annotations: A._Alvey
Annotations
Rules:
Rules
Description: A._Alvey
Types
Employee
Same Individual As
Different Individuals
Property assertions: A._Alvey
Object property assertions
Reasoner active
Show Inferences

hasAuthor(?mem, ?auth) -> hasRelatedMemo(?auth, ?mem)

hasAuthor(?mem, ?auth), hasProject(?mem, ?proj) ->

hasRelatedMemo Memo_3405

hasRelatedMemo Memo_3703

hasRelatedMemo Memo_2789

hasRelatedMemo Memo_74

hasRelatedMemo Memo_2197

hasRelatedProject Beta

hasRelatedProject Omega

Hint 3:
Memos that John Altidor wrote appear in the “Property assertions” box and populate the “hasRelatedMemo” property.

Task #2

Find any memo written on February 20, 1972. What is the memo number?

Click to show hints if you need help (PLEASE LEAVE HINTS CHECKED IF YOU USE THEM)

☐ Show Hint 1

☐ Show Hint 2

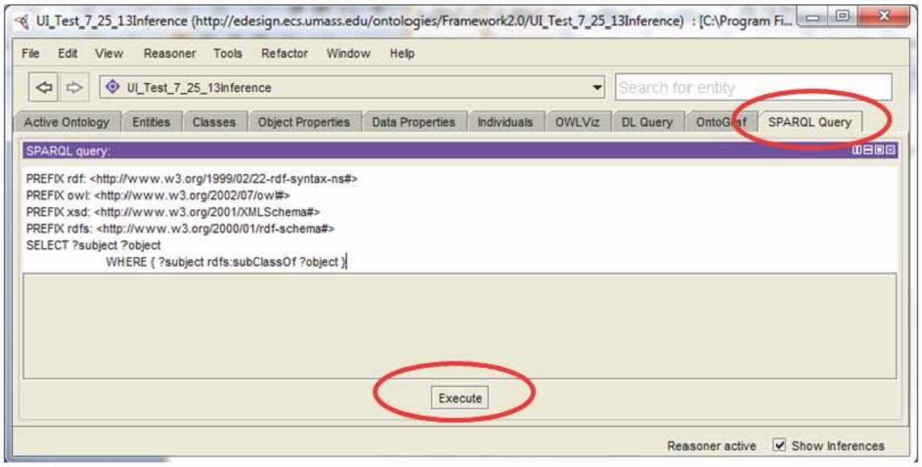
☐ Show Hint 3

Hint 1:
This must be done using a SPARQL query. Open the SPARQL Query tab as shown in the figure.
Use this text, pasted in the “SPARQL query” box, to start your query:

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

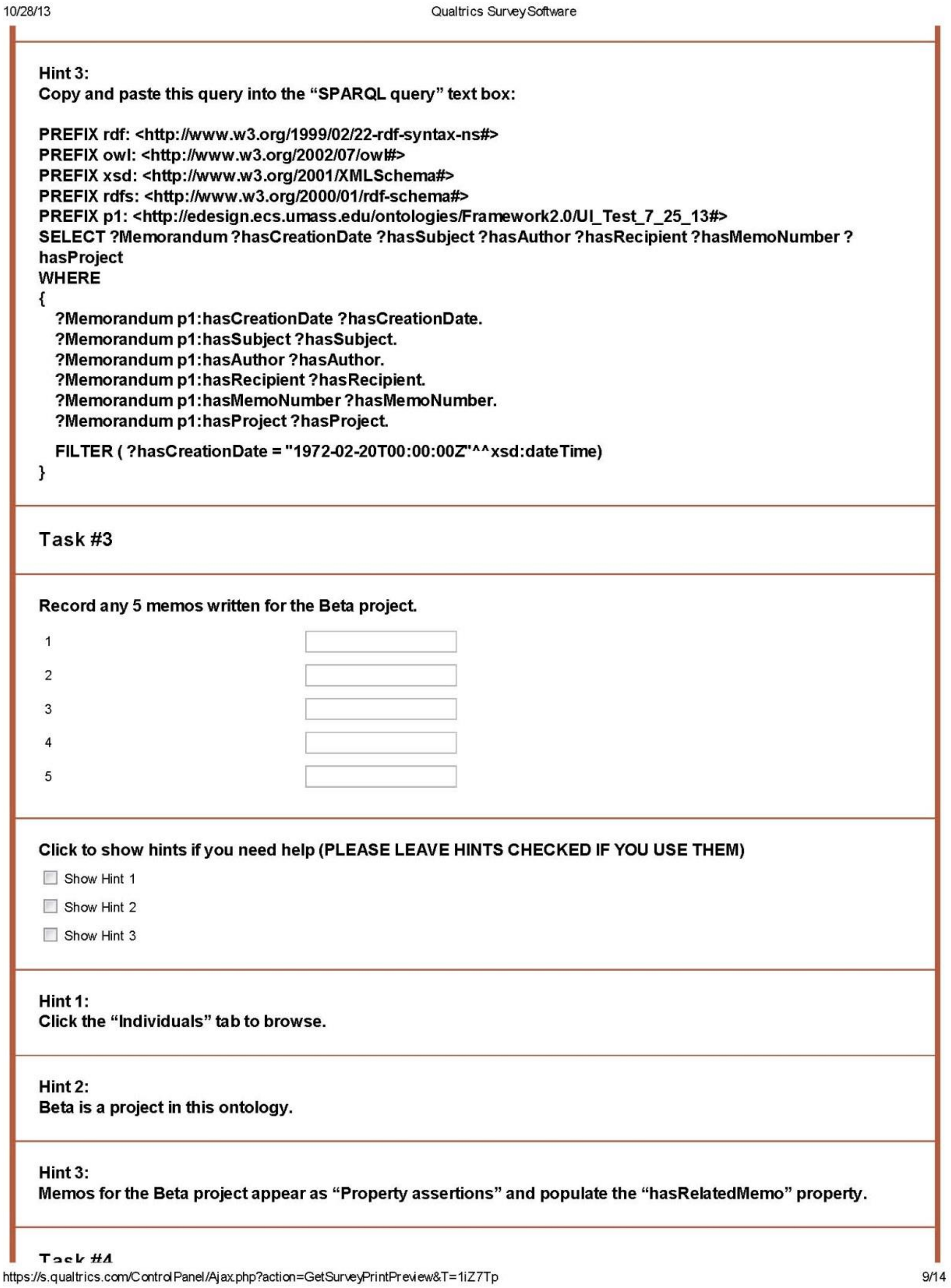
```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX p1: <http://edesign.ecs.umass.edu/ontologies/Framework2.0/UI_Test_7_25_13#>
PREFIX p2: <http://edesign.ecs.umass.edu/ontologies/Framework2.0/UI_Test_7_25_13Inference#>
SELECT ?Memorandum ?hasCreationDate ?hasSubject ?hasAuthor ?hasRecipient ?hasMemoNumber ?
hasProject
WHERE
{
  ?Memorandum p1:hasCreationDate ?hasCreationDate.
  ?Memorandum p1:hasSubject ?hasSubject.
  ?Memorandum p1:hasAuthor ?hasAuthor.
  ?Memorandum p1:hasRecipient ?hasRecipient.
  ?Memorandum p1:hasMemoNumber ?hasMemoNumber.
  ?Memorandum p1:hasProject ?hasProject.
  FILTER ( ?hasCreationDate = "2008-04-16T00:00:00Z"^^xsd:dateTime)
}
```

Click “Execute” at the bottom of the screen and this query will return Memo_3487, which is the only memo written on this date.



```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?subject ?object
        WHERE { ?subject rdfs:subClassOf ?object }
```

Hint 2:
Change the date to match the one in the question, like the example below:
FILTER (?hasCreationDate = "1972-02-20T00:00:00Z"^^xsd:dateTime)



Hint 3:
Copy and paste this query into the “SPARQL query” text box:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX p1: <http://edesign.ecs.umass.edu/ontologies/Framework2.0/UI_Test_7_25_13#>
SELECT ?Memorandum ?hasCreationDate ?hasSubject ?hasAuthor ?hasRecipient ?hasMemoNumber ?
hasProject
WHERE
{
   ?Memorandum p1:hasCreationDate ?hasCreationDate.
   ?Memorandum p1:hasSubject ?hasSubject.
   ?Memorandum p1:hasAuthor ?hasAuthor.
   ?Memorandum p1:hasRecipient ?hasRecipient.
   ?Memorandum p1:hasMemoNumber ?hasMemoNumber.
   ?Memorandum p1:hasProject ?hasProject.

   FILTER ( ?hasCreationDate = "1972-02-20T00:00:00Z"^^xsd:dateTime)
}
```

Task #3

Record any 5 memos written for the Beta project.

1 ____

2 ____

3 ____

4 ____

5 ____

Click to show hints if you need help (PLEASE LEAVE HINTS CHECKED IF YOU USE THEM)

□ Show Hint 1

□ Show Hint 2

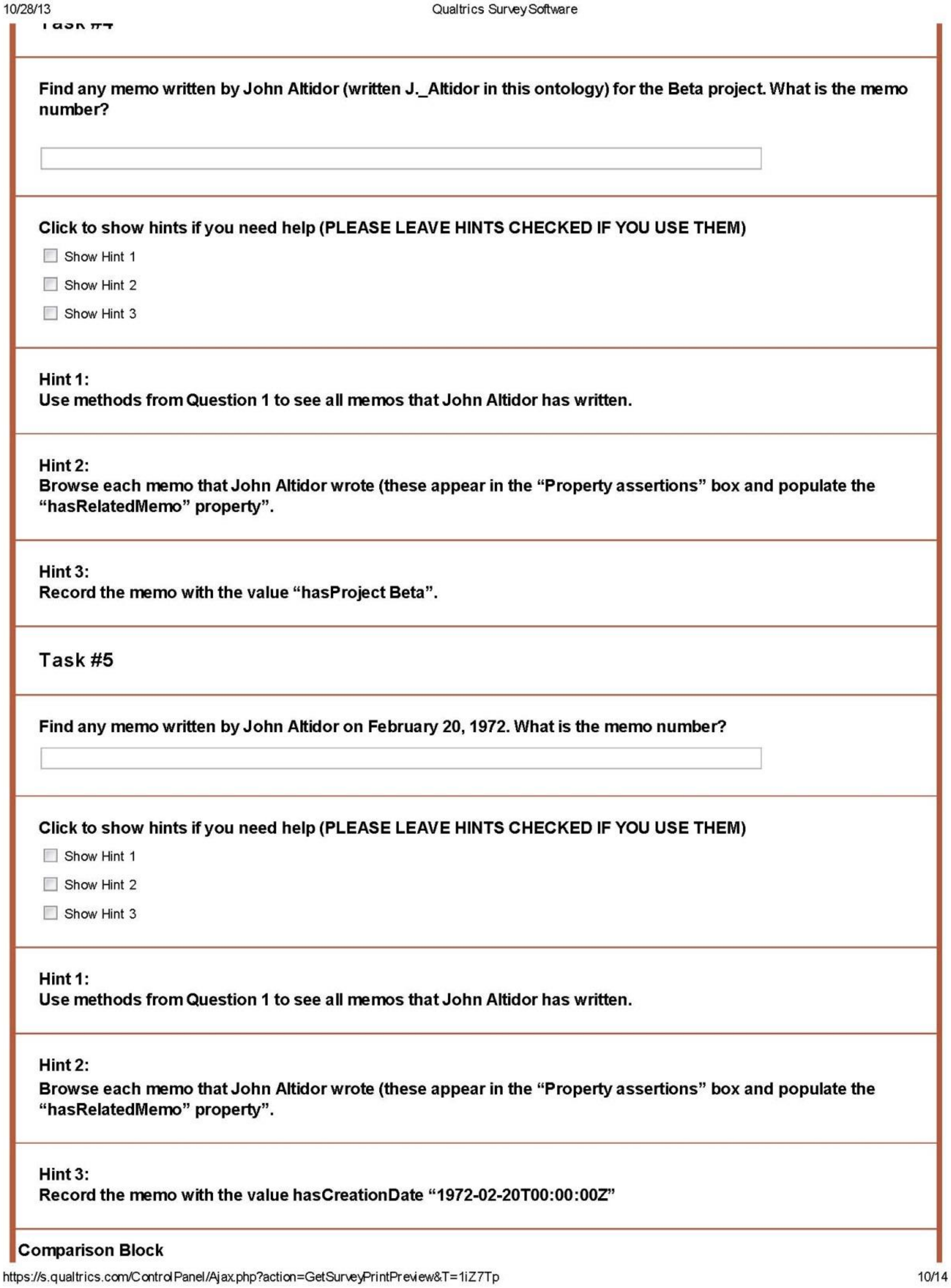
□ Show Hint 3

Hint 1:
Click the “Individuals” tab to browse.

Hint 2:
Beta is a project in this ontology.

Hint 3:
Memos for the Beta project appear as “Property assertions” and populate the “hasRelatedMemo” property.

Task #4



Task #4

Find any memo written by John Altidor (written J._Altidor in this ontology) for the Beta project. What is the memo number?

Click to show hints if you need help (PLEASE LEAVE HINTS CHECKED IF YOU USE THEM)

- □ Show Hint 1
- □ Show Hint 2
- □ Show Hint 3

Hint 1:
Use methods from Question 1 to see all memos that John Altidor has written.

Hint 2:
Browse each memo that John Altidor wrote (these appear in the “Property assertions” box and populate the “hasRelatedMemo” property”.

Hint 3:
Record the memo with the value “hasProject Beta”.

Task #5

Find any memo written by John Altidor on February 20, 1972. What is the memo number?

Click to show hints if you need help (PLEASE LEAVE HINTS CHECKED IF YOU USE THEM)

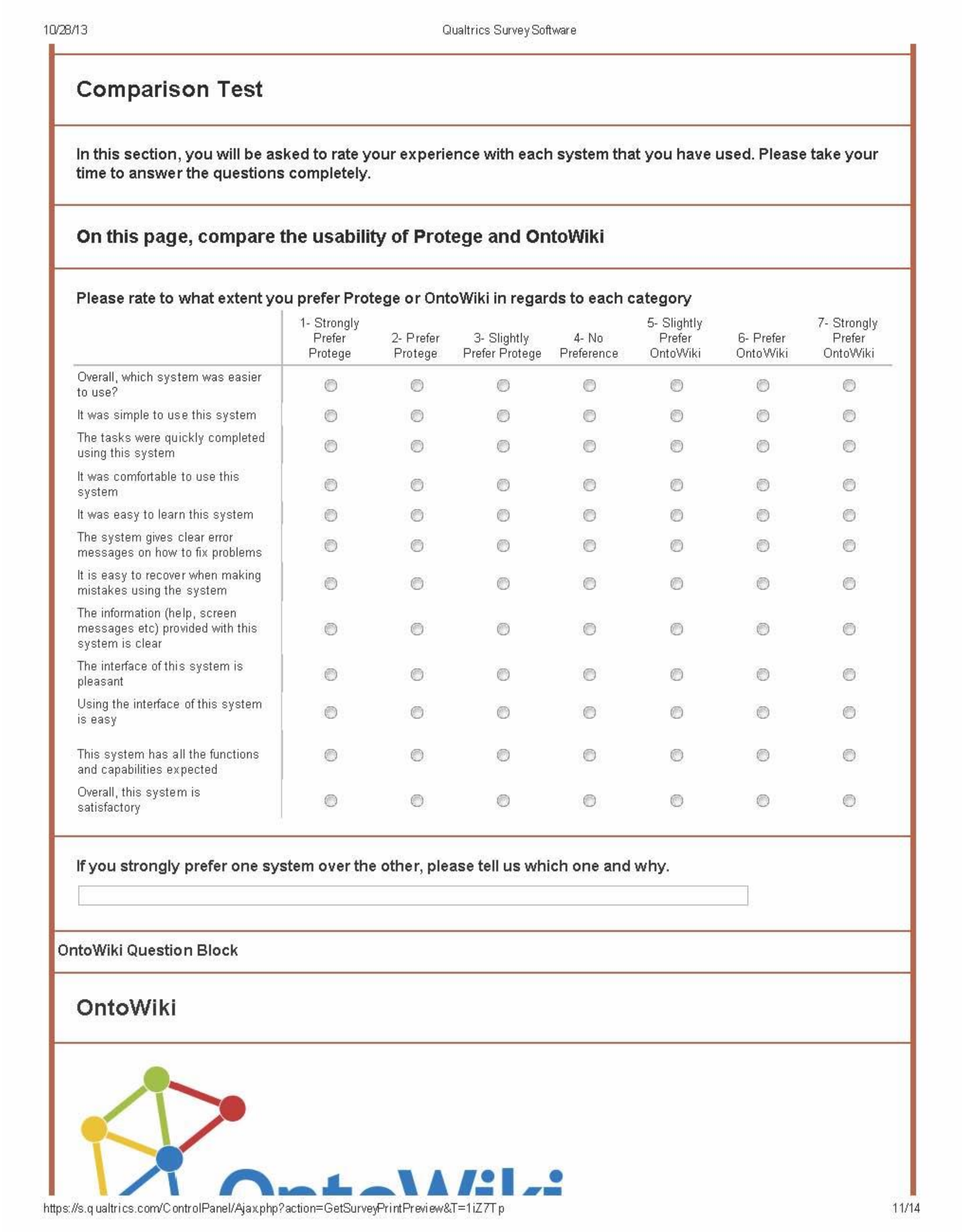
- □ Show Hint 1
- □ Show Hint 2
- □ Show Hint 3

Hint 1:
Use methods from Question 1 to see all memos that John Altidor has written.

Hint 2:
Browse each memo that John Altidor wrote (these appear in the “Property assertions” box and populate the “hasRelatedMemo” property”.

Hint 3:
Record the memo with the value hasCreationDate “1972-02-20T00:00:00Z”

Comparison Block



Comparison Test

In this section, you will be asked to rate your experience with each system that you have used. Please take your time to answer the questions completely.

On this page, compare the usability of Protege and OntoWiki

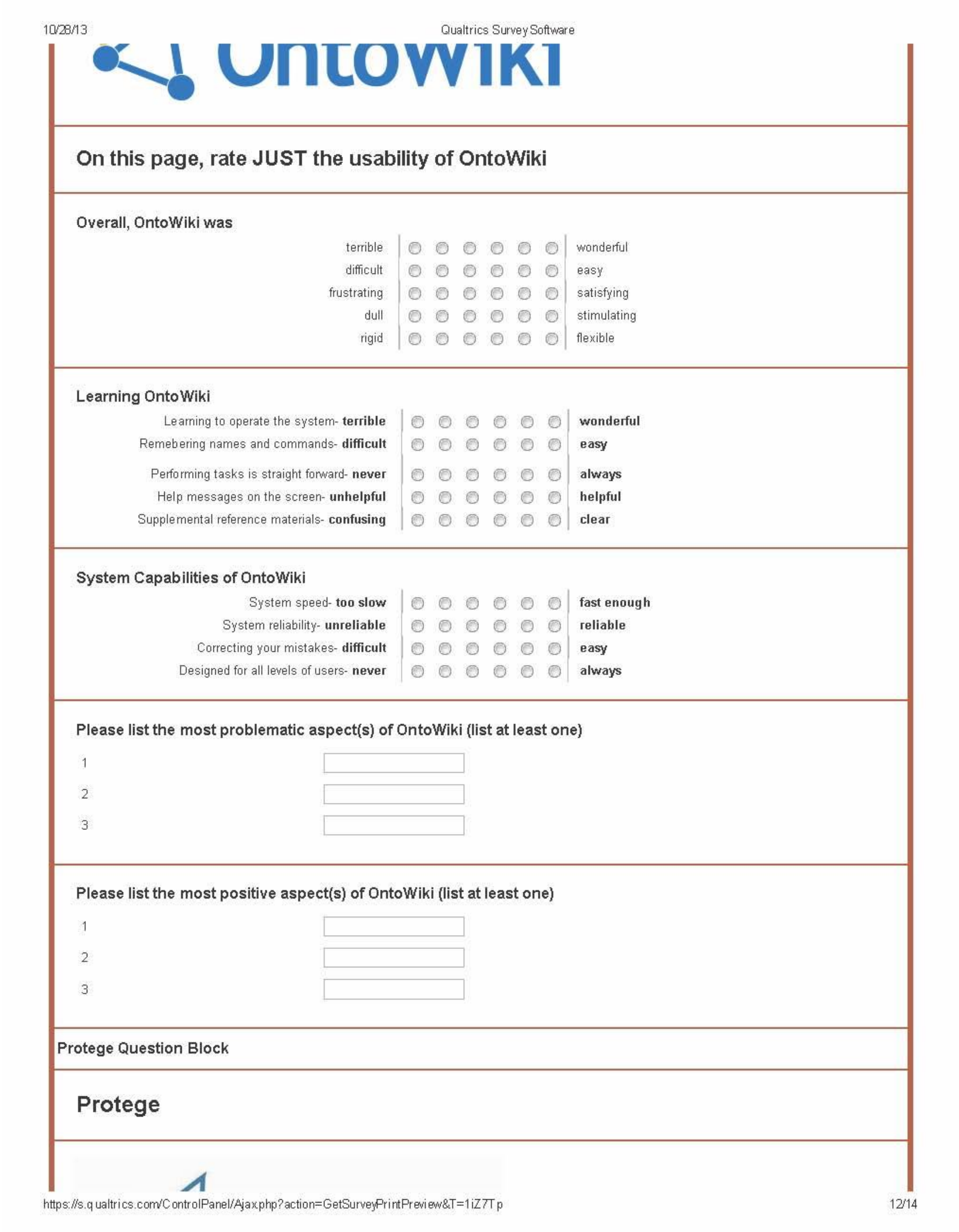
Please rate to what extent you prefer Protege or OntoWiki in regards to each category

	1- Strongly Prefer Protege	2- Prefer Protege	3- Slightly Prefer Protege	4- No Preference	5- Slightly Prefer OntoWiki	6- Prefer OntoWiki	7- Strongly Prefer OntoWiki
Overall, which system was easier to use?	○	○	○	○	○	○	○
It was simple to use this system	○	○	○	○	○	○	○
The tasks were quickly completed using this system	○	○	○	○	○	○	○
It was comfortable to use this system	○	○	○	○	○	○	○
It was easy to learn this system	○	○	○	○	○	○	○
The system gives clear error messages on how to fix problems	○	○	○	○	○	○	○
It is easy to recover when making mistakes using the system	○	○	○	○	○	○	○
The information (help, screen messages etc) provided with this system is clear	○	○	○	○	○	○	○
The interface of this system is pleasant	○	○	○	○	○	○	○
Using the interface of this system is easy	○	○	○	○	○	○	○
This system has all the functions and capabilities expected	○	○	○	○	○	○	○
Overall, this system is satisfactory	○	○	○	○	○	○	○

If you strongly prefer one system over the other, please tell us which one and why.

OntoWiki Question Block

OntoWiki



10/28/13 Qualtrics Survey Software

On this page, rate JUST the usability of OntoWiki

Overall, OntoWiki was

terrible	○	○	○	○	○	○	wonderful
difficult	○	○	○	○	○	○	easy
frustrating	○	○	○	○	○	○	satisfying
dull	○	○	○	○	○	○	stimulating
rigid	○	○	○	○	○	○	flexible

Learning OntoWiki

Learning to operate the system- **terrible**	○	○	○	○	○	○	**wonderful**
Remebering names and commands- **difficult**	○	○	○	○	○	○	**easy**
Performing tasks is straight forward- **never**	○	○	○	○	○	○	**always**
Help messages on the screen- **unhelpful**	○	○	○	○	○	○	**helpful**
Supplemental reference materials- **confusing**	○	○	○	○	○	○	**clear**

System Capabilities of OntoWiki

System speed- **too slow**	○	○	○	○	○	○	**fast enough**
System reliability- **unreliable**	○	○	○	○	○	○	**reliable**
Correcting your mistakes- **difficult**	○	○	○	○	○	○	**easy**
Designed for all levels of users- **never**	○	○	○	○	○	○	**always**

Please list the most problematic aspect(s) of OntoWiki (list at least one)

1 ______

2 ______

3 ______

Please list the most positive aspect(s) of OntoWiki (list at least one)

1 ______

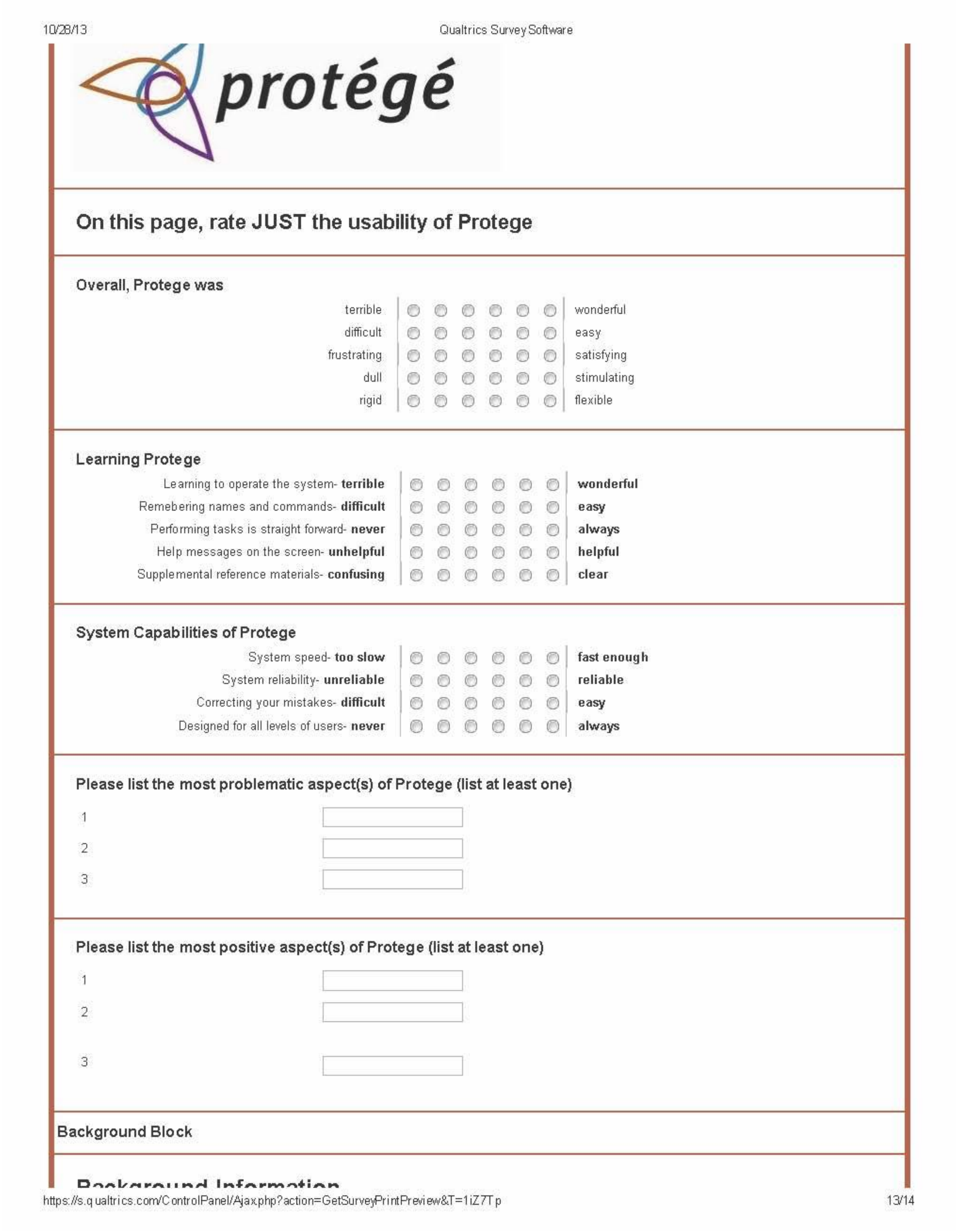
2 ______

3 ______

Protege Question Block

Protege

https://s.qualtrics.com/ControlPanel/Ajax.php?action=GetSurveyPrintPreview&T=1iZ7Tp 12/14



10/28/13 Qualtrics Survey Software

protégé

On this page, rate JUST the usability of Protege

Overall, Protege was

terrible	○	○	○	○	○	○	wonderful
difficult	○	○	○	○	○	○	easy
frustrating	○	○	○	○	○	○	satisfying
dull	○	○	○	○	○	○	stimulating
rigid	○	○	○	○	○	○	flexible

Learning Protege

Learning to operate the system- **terrible**	○	○	○	○	○	○	**wonderful**
Remebering names and commands- **difficult**	○	○	○	○	○	○	**easy**
Performing tasks is straight forward- **never**	○	○	○	○	○	○	**always**
Help messages on the screen- **unhelpful**	○	○	○	○	○	○	**helpful**
Supplemental reference materials- **confusing**	○	○	○	○	○	○	**clear**

System Capabilities of Protege

System speed- **too slow**	○	○	○	○	○	○	**fast enough**
System reliability- **unreliable**	○	○	○	○	○	○	**reliable**
Correcting your mistakes- **difficult**	○	○	○	○	○	○	**easy**
Designed for all levels of users- **never**	○	○	○	○	○	○	**always**

Please list the most problematic aspect(s) of Protege (list at least one)

1 ▭

2 ▭

3 ▭

Please list the most positive aspect(s) of Protege (list at least one)

1 ▭

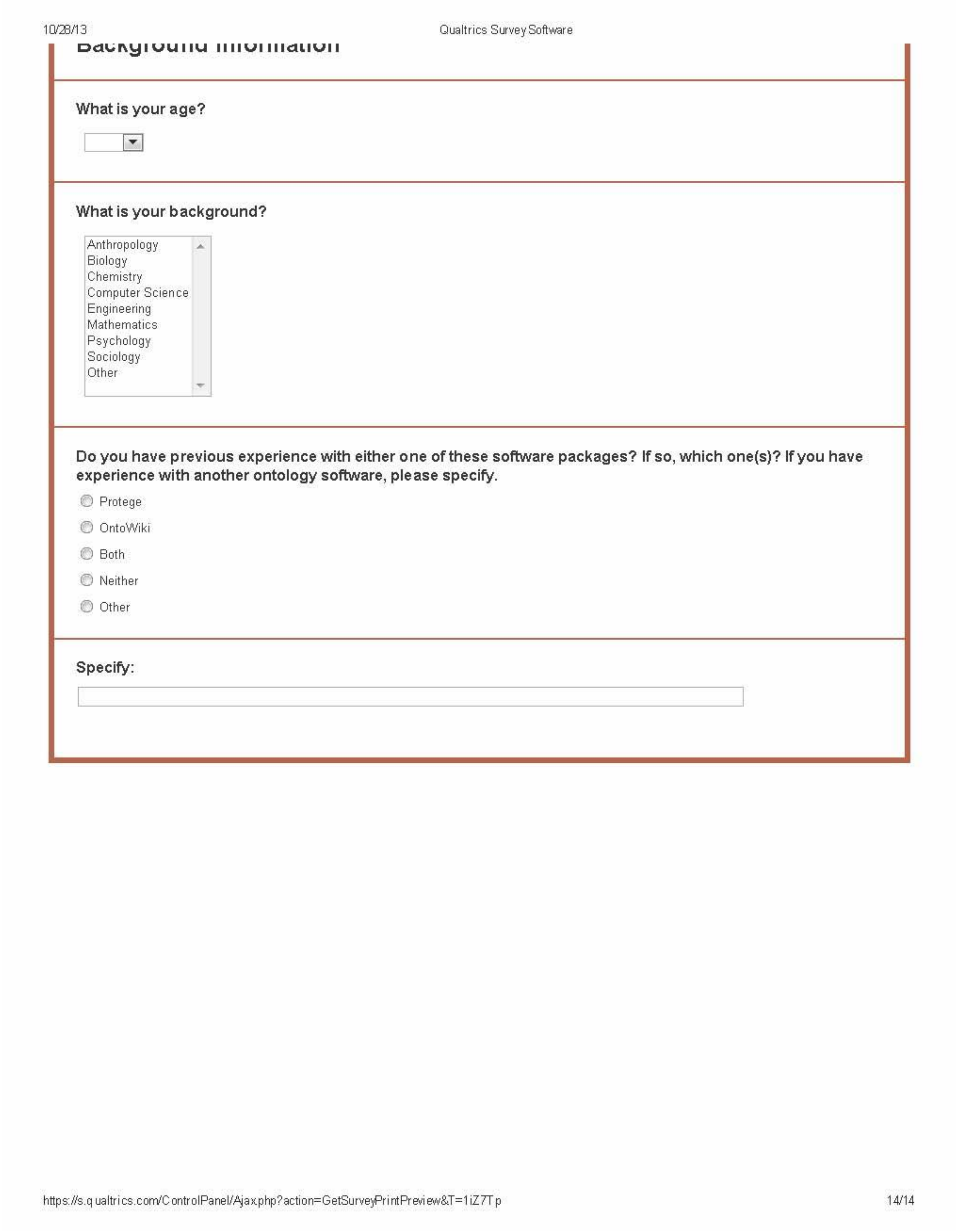
2 ▭

3 ▭

Background Block

Background Information

https://s.qualtrics.com/ControlPanel/Ajax.php?action=GetSurveyPrintPreview&T=1iZ7Tp 13/14



10/28/13 Qualtrics Survey Software

Background Information

What is your age?

▾

What is your background?

Anthropology
Biology
Chemistry
Computer Science
Engineering
Mathematics
Psychology
Sociology
Other

Do you have previous experience with either one of these software packages? If so, which one(s)? If you have experience with another ontology software, please specify.

○ Protege

○ OntoWiki

○ Both

○ Neither

○ Other

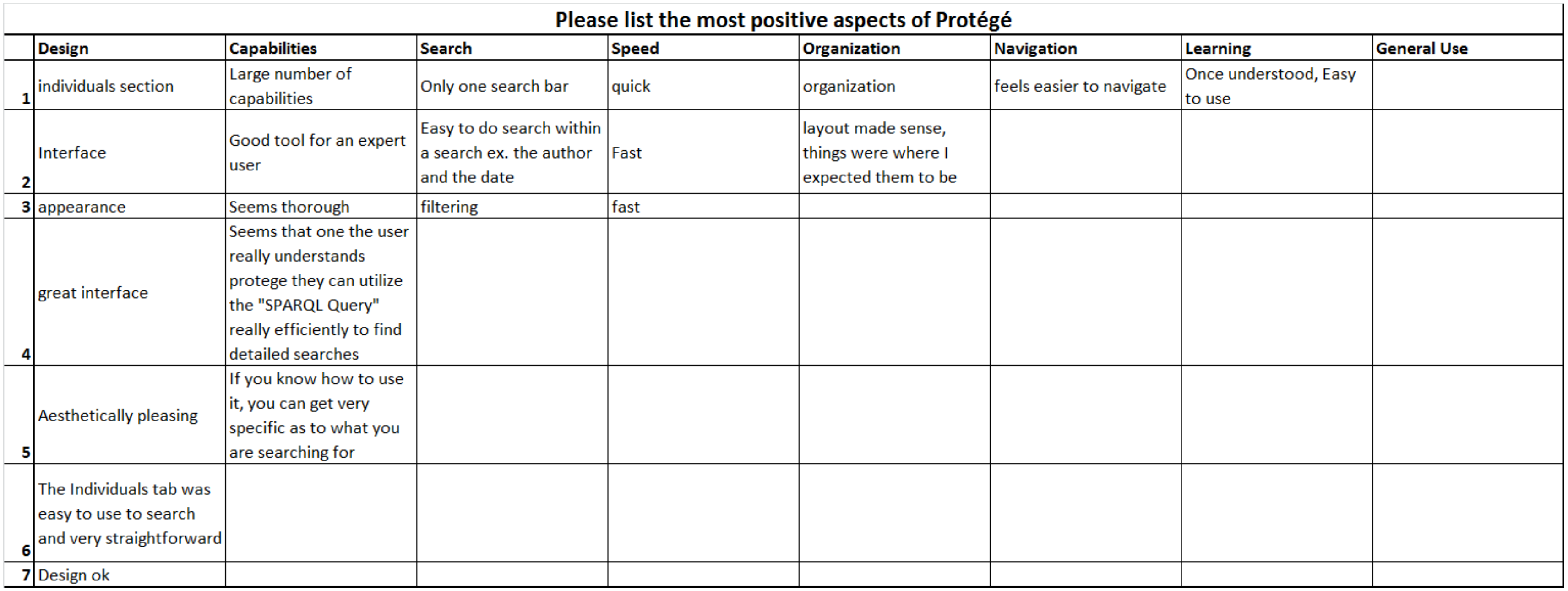
Specify:

https://s.qualtrics.com/ControlPanel/Ajax.php?action=GetSurveyPrintPreview&T=1iZ7Tp 14/14

APPENDIX B

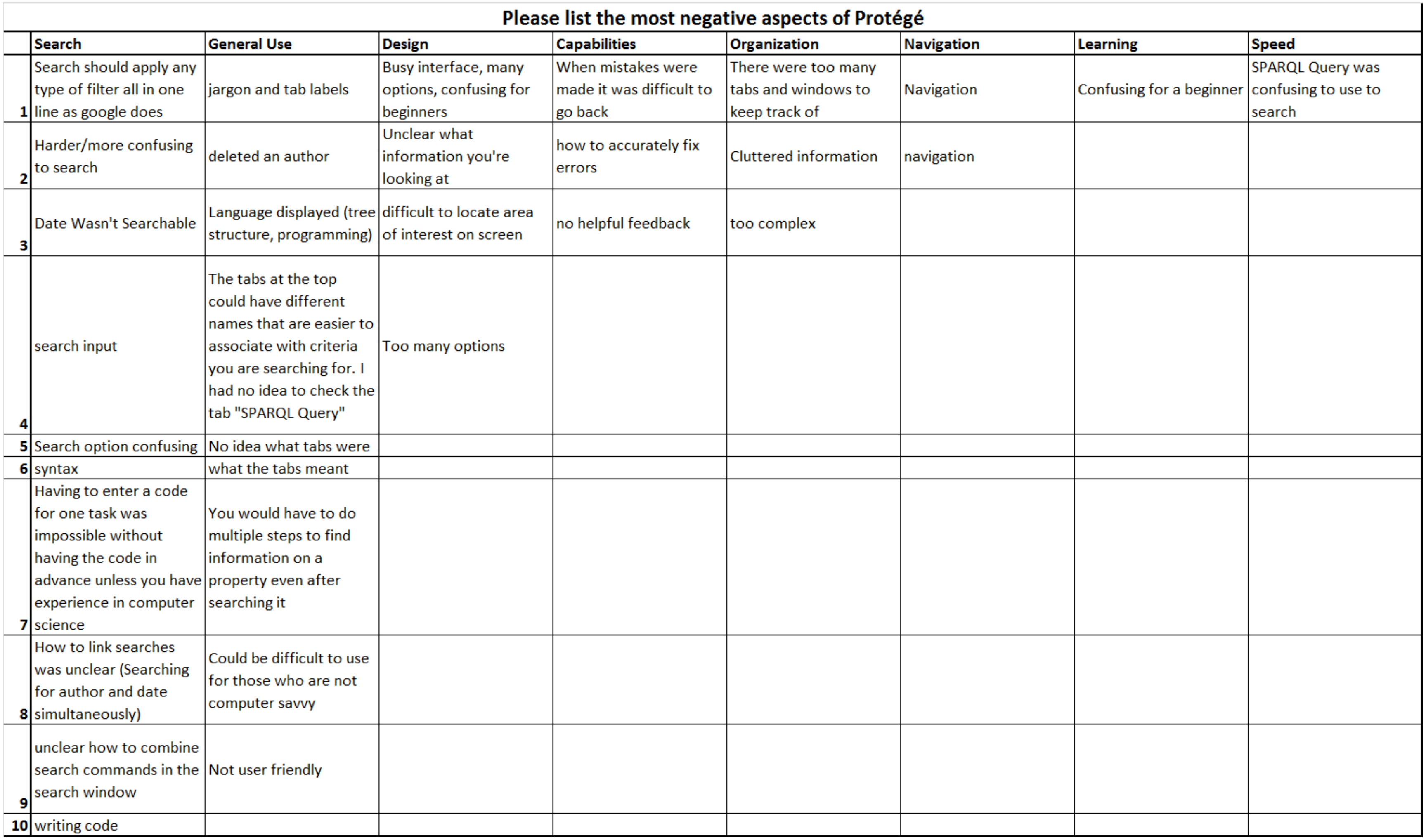
CODED OPEN RESPONSE DATA

Here, we present summary tables of subjects' answers to the open response questions found on the survey.



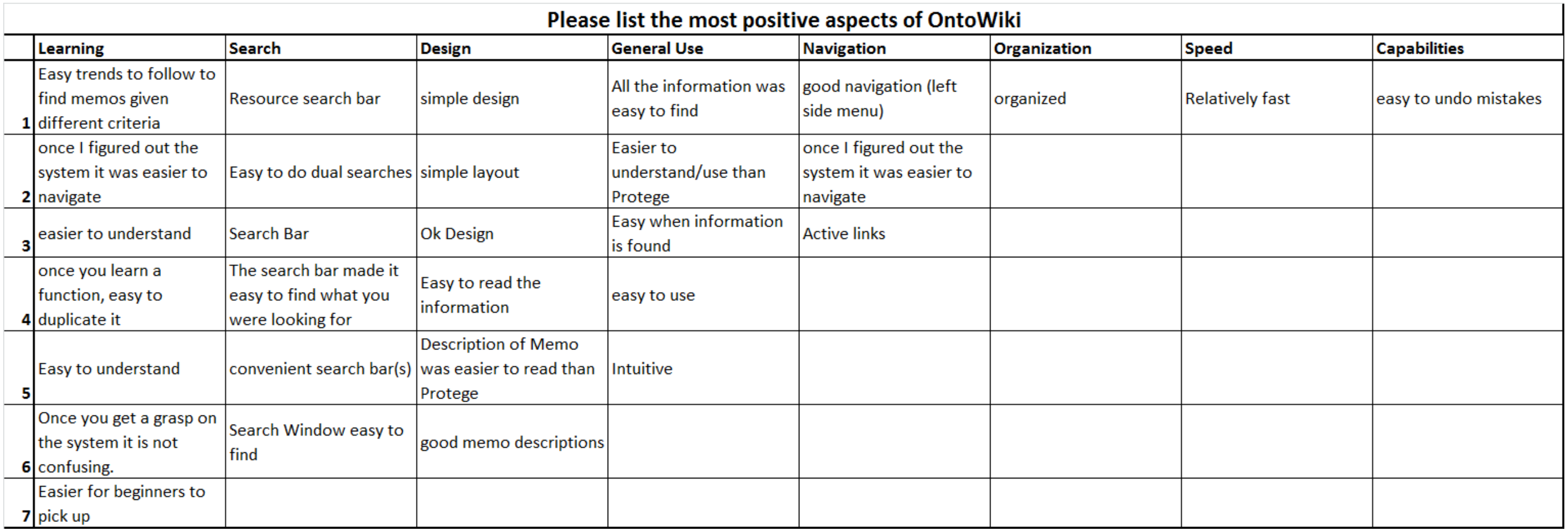
Please list the most positive aspects of Protégé

	Design	Capabilities	Search	Speed	Organization	Navigation	Learning	General Use
1	individuals section	Large number of capabilities	Only one search bar	quick	organization	feels easier to navigate	Once understood, Easy to use	
2	Interface	Good tool for an expert user	Easy to do search within a search ex. the author and the date	Fast	layout made sense, things were where I expected them to be			
3	appearance	Seems thorough	filtering	fast				
4	great interface	Seems that one the user really understands protege they can utilize the "SPARQL Query" really efficiently to find detailed searches						
5	Aesthetically pleasing	If you know how to use it, you can get very specific as to what you are searching for						
6	The Individuals tab was easy to use to search and very straightforward							
7	Design ok							



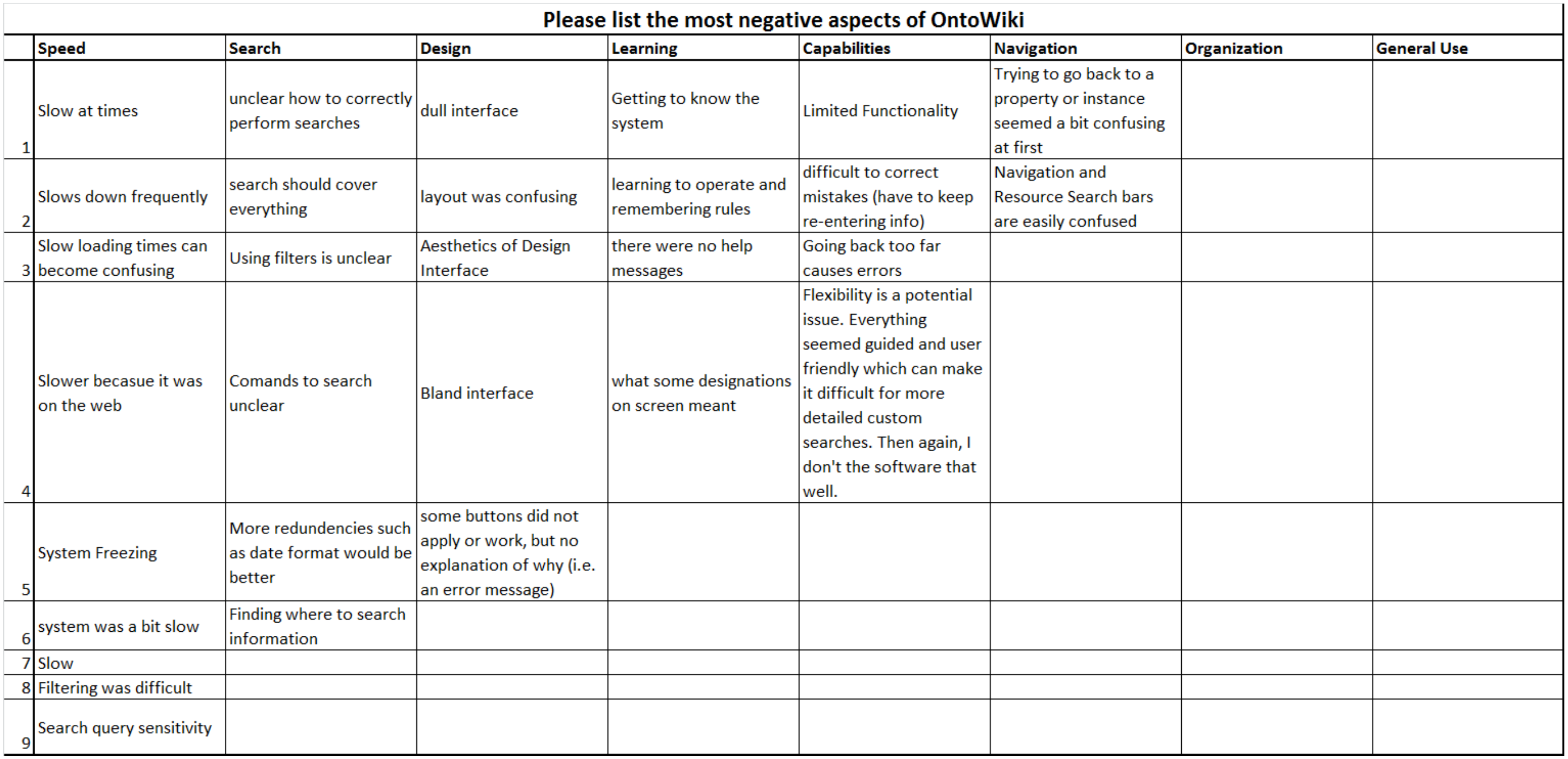
Please list the most negative aspects of Protégé

	Search	General Use	Design	Capabilities	Organization	Navigation	Learning	Speed
1	Search should apply any type of filter all in one line as google does	jargon and tab labels	Busy interface, many options, confusing for beginners	When mistakes were made it was difficult to go back	There were too many tabs and windows to keep track of	Navigation	Confusing for a beginner	SPARQL Query was confusing to use to search
2	Harder/more confusing to search	deleted an author	Unclear what information you're looking at	how to accurately fix errors	Cluttered information	navigation		
3	Date Wasn't Searchable	Language displayed (tree structure, programming)	difficult to locate area of interest on screen	no helpful feedback	too complex			
4	search input	The tabs at the top could have different names that are easier to associate with criteria you are searching for. I had no idea to check the tab "SPARQL Query"	Too many options					
5	Search option confusing	No idea what tabs were						
6	syntax	what the tabs meant						
7	Having to enter a code for one task was impossible without having the code in advance unless you have experience in computer science	You would have to do multiple steps to find information on a property even after searching it						
8	How to link searches was unclear (Searching for author and date simultaneously)	Could be difficult to use for those who are not computer savvy						
9	unclear how to combine search commands in the search window	Not user friendly						
10	writing code							



Please list the most positive aspects of OntoWiki

	Learning	Search	Design	General Use	Navigation	Organization	Speed	Capabilities
1	Easy trends to follow to find memos given different criteria	Resource search bar	simple design	All the information was easy to find	good navigation (left side menu)	organized	Relatively fast	easy to undo mistakes
2	once I figured out the system it was easier to navigate	Easy to do dual searches	simple layout	Easier to understand/use than Protege	once I figured out the system it was easier to navigate			
3	easier to understand	Search Bar	Ok Design	Easy when information is found	Active links			
4	once you learn a function, easy to duplicate it	The search bar made it easy to find what you were looking for	Easy to read the information	easy to use				
5	Easy to understand	convenient search bar(s)	Description of Memo was easier to read than Protege	Intuitive				
6	Once you get a grasp on the system it is not confusing.	Search Window easy to find	good memo descriptions					
7	Easier for beginners to pick up							



Please list the most negative aspects of OntoWiki

	Speed	Search	Design	Learning	Capabilities	Navigation	Organization	General Use
1	Slow at times	unclear how to correctly perform searches	dull interface	Getting to know the system	Limited Functionality	Trying to go back to a property or instance seemed a bit confusing at first		
2	Slows down frequently	search should cover everything	layout was confusing	learning to operate and remembering rules	difficult to correct mistakes (have to keep re-entering info)	Navigation and Resource Search bars are easily confused		
3	Slow loading times can become confusing	Using filters is unclear	Aesthetics of Design Interface	there were no help messages	Going back too far causes errors			
4	Slower becasue it was on the web	Comands to search unclear	Bland interface	what some designations on screen meant	Flexibility is a potential issue. Everything seemed guided and user friendly which can make it difficult for more detailed custom searches. Then again, I don't the software that well.			
5	System Freezing	More redundencies such as date format would be better	some buttons did not apply or work, but no explanation of why (i.e. an error message)					
6	system was a bit slow	Finding where to search information						
7	Slow							
8	Filtering was difficult							
9	Search query sensitivity							

BIBLIOGRAPHY

[1] McAfee, A., and Brynjolfsson, E., 2012, "Big Data: The Management Revolution," Harvard Business Review**, 1**.

[2] Rockwell, J., Grosse, I., Krishnamurty, S., 2009, "A Decision Support Ontology for collaborative decision making in engineering design," Collaborative Technologies and Systems, 2009. CTS'09. International Symposium on, IEEE, pp. 1-9.

[3] Rockwell, J., 2009, "A Semantic Framework for Reusing Decision Making Knowledge in Engineering Design," **Paper 329**.

[4] Rockwell, J. A., Witherell, P., Fernandes, R., 2008, "A web-based environment for documentation and sharing of engineering design knowledge," ASME, .

[5] Eddy, D., Krishnamurty, S., Grosse, I., 2011, "Support of Product Innovation With a Modular Framework for Knowledge Management: A Case Study," ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, ASME, Washington, DC, .

[6] Fernandes, R., Grosse, I., Krishnamurty, S., 2007, "Design and innovative methodologies in a semantic framework," ASME, .

[7] Grosse, I. R., Milton-Benoit, J. M., and Wileden, J. C., 2005, "Ontologies for Supporting Engineering Analysis Models," Artificial Intelligence for Engineering Design, Analysis and Manufacturing**, 19**, (01), pp. 1.

[8] Witherell, P., Krishnamurty, S., and Grosse, I. R., 2007, "Ontologies for Supporting Engineering Design Optimization," Journal of Computing and Information Science in Engineering**, 7**, pp. 141.

[9] Ashburner, M., Ball, C. A., Blake, J. A., 2000, "Gene Ontology: Tool for the Unification of Biology," Nature Genetics**, 25**, (1), pp. 25.

[10] Smith, B., Ashburner, M., Rosse, C., 2007, "The OBO Foundry: Coordinated Evolution of Ontologies to Support Biomedical Data Integration," Nature Biotechnology**, 25**, (11), pp. 1251-1255.

[11] Breindel, J. T., Grosse, I. R., Krishnamurty, S., 2011, "Towards industrial implementation of emerging semantic technologies," ASME, .

[12] DIQA, "DataWiki- The semantic wiki for your company," 2013, <http://diqa-pm.com/en/DataWiki>, Visited: 2/25/2013.

[13] Ontoprise, "SMW+: Semantic Enterprise Wiki Homepage," 2010, <http://wiki.ontoprise.de/smwforum/index.php/Main_Page>.

[14] Auer, S., Dietzold, S., and Riechert, T., 2006, "OntoWiki–A Tool for Social, Semantic Collaboration," The Semantic Web-ISWC 2006, pp. 736-749.

[15] Dietzold,S., "Ontowiki Home Page," 2012, <http://aksw.org/Projects/OntoWiki.html>, Visited: 3/2/2013.

[16] Revelytix,I., "Knoodl Home Page," 2012, <http://knoodl.com/ui/home.html>, Visited: 3/2/2013.

[17] KiWi, "KiWi Project Home Page," 2012, <http://www.kiwi-project.eu/>, Visited: 3/2/2013.

[18] Buffa, M., Gandon, F., Ereteo, G., 2008, "SweetWiki: A Semantic Wiki," Web Semantics: Science, Services and Agents on the World Wide Web**, 6**, (1), pp. 84-97.

[19] Kuhn, T., 2008, "Acewiki: A Natural and Expressive Semantic Wiki," arXiv Preprint arXiv:0807.4618.

[20] Aamodt, A., and Nygård, M., 1995, "Different Roles and Mutual Dependencies of Data, Information, and Knowledge—an AI Perspective on their Integration," Data & Knowledge Engineering**, 16**, (3), pp. 191-222.

[21] Smith, B., and Welty, C., 2001, "Ontology: Towards a new synthesis," Formal Ontology in Information Systems, ACM Press, USA, pp. iii-x, pp. 3-9.

[22] Gruber, T. R., 1993, "A Translation Approach to Portable Ontology Specifications," Knowledge Acquisition**, 5**, (2), pp. 199-220.

[23] Berners-Lee, T., Hendler, J., and Lassila, O., 2001, "The Semantic Web," Scientific American**, 284**, (5), pp. 34.

[24] Auer, S., and Lehmann, J., 2010, "Creating Knowledge Out of Interlinked Data," Semantic Web**, 1**, (1), pp. 97-104.

[25] Hitzler, P., and van Harmelen, F., 2010, "A Reasonable Semantic Web," Semantic Web**, 1**, (1), pp. 39-44.

[26] Van Grove,J., "Facebook takes on Google with Graph Search," 2013, <http://news.cnet.com/8301-1023_3-57564077-93/facebook-takes-on-google-with-graph-search/>, Visited: 10/17/2013.

[27] "Power BI Q&A Preview," <http://office.microsoft.com/en-001/office365-sharepoint-online-enterprise-help/power-bi-q-a-preview-HA104114866.aspx>, Visited: 10/17/2013.

[28] McGuinness, D. L., and Van Harmelen, F., 2004, "OWL Web Ontology Language Overview," W3C Recommendation**, 10**, (2004-03), pp. 10.

[29] Kuhn, T., 2009, "How Controlled English can Improve Semantic Wikis," arXiv Preprint arXiv:0907.1245.

[30] Prud'Hommeaux, E., and Seaborne, A., 2008, "SPARQL Query Language for RDF," W3C Recommendation**, 15**.

[31] Fuchs, N. E., and Schwitter, R., 1996, "Attempto Controlled English (Ace)," arXiv Preprint Cmp-Lg/9603003.

[32] Dietzold,S., "Agile Knowledge Engineering and Semantic Web (AKSW)," 2012, <http://aksw.org/About.html>, Visited: 2/24/2013.

[33] "DIQA," 2013, <http://diqa-pm.com/en/Main_Page>, Visited: 10/17/2013.

[34] "SMWCon," 2013, <http://semantic-mediawiki.org/wiki/SMWCon>, Visited: 9/12/2013.

[35] Shortliffe, E. H., Davis, R., Axline, S. G., 1975, "Computer-Based Consultations in Clinical Therapeutics: Explanation and Rule Acquisition Capabilities of the MYCIN System," Computers and Biomedical Research**, 8**, (4), pp. 303-320.

[36] Maher, M. L., Sriram, D., and Fenves, S. J., 1984, "Tools and Techniques for Knowledge Based Expert Systems for Engineering Design," Advances in Engineering Software (1978)**, 6**, (4), pp. 178-188.

[37] Rychener, M., 1988, "Expert systems for engineering design," Access Online via Elsevier.

[38] Pratt, M. J., 2001, "Introduction to ISO 10303—the STEP Standard for Product Data Exchange," Journal of Computing and Information Science in Engineering**, 1**, (1), pp. 102-103.

[39] Kim, B. C., Mun, D., Han, S., 2011, "A Method to Exchange Procedurally Represented 2D CAD Model Data using ISO 10303 STEP," Computer-Aided Design**, 43**, (12), pp. 1717-1728.

[40] PTC, "PTC Products: Windchill," 2013, <http://www.ptc.com/products/windchill/>, Visited: 2/3/2013.

[41] "Teamcenter: Siemens PLM Software," 2013, <http://www.plm.automation.siemens.com/en_us/products/teamcenter/>, Visited: 10/21/2013.

[42] Sainter, P., Oldham, K., Larkin, A., 2000, "Product knowledge management within knowledge-based engineering systems," Proceedings of the ASME 2000 design engineering technical conference and computers and information in engineering conference.

[43] Sudarsan, R., Fenves, S. J., Sriram, R. D., 2005, "A Product Information Modeling Framework for Product Lifecycle Management," Computer-Aided Design**, 37**, (13), pp. 1399-1411.

[44] ANSYS, I., 2009, "ANSYS® Engineering Knowledge Manager™," ANSYS® 2.0 Release.

[45] Breindel, J., 2011, "Techniques for Industrial Implementation of Emerging Semantic Technologies,".

[46] Eddy, D., Krishnamurty, S., Grosse, I., 2012, "Toward Integration of a Semantic Framework With a Commercial PLM System," ASME 2012 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, ASME, Chicago, Illinois.

[47] McMahon, C., Lowe, A., Culley, S., 2004, "Waypoint: An Integrated Search and Retrieval System for Engineering Documents," Journal of Computing and Information Science in Engineering**, 4**, pp. 329.

[48] Li, Z., Raskin, V., and Ramani, K., 2008, "Developing Engineering Ontology for Information Retrieval," Journal of Computing and Information Science in Engineering**, 8**, pp. 011003.

[49] Li, Z., and Ramani, K., 2007, "Ontology-Based Design Information Extraction and Retrieval," Artificial Intelligence for Engineering Design, Analysis and Manufacturing**, 21**, (02), pp. 137.

[50] Li, Z., Yang, M. C., and Ramani, K., 2009, "A Methodology for Engineering Ontology Acquisition and Validation," Artificial Intelligence for Engineering Design, Analysis and Manufacturing**, 23**, (01), pp. 37.

[51] Verhagen, W. J., and Curran, R., 2010, "Knowledge-Based Engineering Review: Conceptual Foundations and Research Issues," New World Situation: New Directions in Concurrent Engineering, pp. 267-276.

[52] Patil, L., Dutta, D., and Sriram, R., 2005, "Ontology-Based Exchange of Product Data Semantics," Automation Science and Engineering, IEEE Transactions On**, 2**, (3), pp. 213-225.

[53] Codd, E. F., 1970, "A relational model of data for large shared data banks," Comm. ACM, pp. 377-387.

[54] Codd, E. F., 1979, "Extending the Database Relational Model to Capture More Meaning," ACM Transactions on Database Systems (TODS)**, 4**, (4), pp. 397-434.

[55] Hammer, M., and McLeod, D., 1979, On Database Management System Architecture.

[56] McLeod, D., and Heimbigner, D., 1980, "A federated architecture for database systems," Proceedings of the May 19-22, 1980, national computer conference, ACM, pp. 283-289.

[57] Heimbigner, D., and McLeod, D., 1985, "A Federated Architecture for Information Management," ACM Transactions on Information Systems (TOIS)**, 3**, (3), pp. 253-278.

[58] Stumme, G., and Maedche, A., 2001, "Ontology merging for federated ontologies on the semantic web," Proceedings of the International Workshop for Foundations of Models for Information Integration (FMII-2001), Citeseer, pp. 413-418.

[59] Noy, N. F., and Musen, M. A., 2000, "Algorithm and tool for automated ontology merging and alignment," Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-00). Available as SMI technical report SMI-2000-0831.

[60] Noy, N. F., and Musen, M. A., 2003, "The PROMPT Suite: Interactive Tools for Ontology Merging and Mapping," International Journal of Human-Computer Studies**, 59**, (6), pp. 983-1024.

[61] Chalupsky, H., 2000, "Ontomorph: A translation system for symbolic knowledge," PRINCIPLES OF KNOWLEDGE REPRESENTATION AND REASONING-INTERNATIONAL CONFERENCE-, Morgan Kaufmann Publishers; 1998, pp. 471-482.

[62] McPherson, J., Grosse, I. R., Krishnamurty, S., 2013, "Integrating Biological and Engineering Ontologies," ASME 2013 International Design Engineering Technical Concerences & Computers and Information in Engineering Conference, ASME, Portland, OR.

[63] Chandrasegaran, S. K., Ramani, K., Sriram, R. D., 2013, "The Evolution, Challenges, and Future of Knowledge Representation in Product Design Systems," Computer-Aided Design**, 45**, (2), pp. 204.

[64] Smith,B., "Lecture 1 on Biomedical Ontology - Semantics of Biodiversity Workshop," 2012, <http://www.youtube.com/watch?v=eLeJgxj6AMY;>, Visited: 1/14/2013.

[65] Gennari, J. H., Neal, M. L., Galdzicki, M., 2011, "Multiple Ontologies in Action: Composite Annotations for Biosimulation Models," Journal of Biomedical Informatics**, 44**, (1), pp. 146-154.

[66] Bodenreider, O., and Stevens, R., 2006, "Bio-Ontologies: Current Trends and Future Directions," Briefings in Bioinformatics**, 7**, (3), pp. 256-274.

[67] Ghazvinian, A., Noy, N. F., and Musen, M. A., 2011, "How Orthogonal are the OBO Foundry Ontologies," J Biomed Semantics**, 2**, (Suppl 2), pp. S2.

[68] Lloyd, C. M., Lawson, J. R., Hunter, P. J., 2008, "The CellML Model Repository," Bioinformatics**, 24**, (18), pp. 2122-2123.

[69] Hoehndorf, R., Dumontier, M., Gennari, J. H., 2011, "Integrating Systems Biology Models and Biomedical Ontologies," BMC Systems Biology**, 5**, (1), pp. 124.

[70] Novere, N. L., Finney, A., Hucka, M., 2005, "Minimum Information Requested in the Annotation of Biochemical Models (MIRIAM)," Nature Biotechnology**, 23**, (12), pp. 1509-1515.

[71] Rosse, C., and Mejino Jr, J. L. V., 2003, "A Reference Ontology for Biomedical Informatics: The Foundational Model of Anatomy," Journal of Biomedical Informatics**, 36**, (6), pp. 478-500.

[72] Bard, J. B. L., Malone, J., Rayner, T. F., 2008, "Minimal Anatomy Terminology (MAT): a species-independent terminology for anatomical mapping and retrieval," 11<SUP>th</SUP>Annual Bio-Ontologies Meeting, Citeseer, Toronto, Canada, pp. 41.

[73] Sayers, E. W., Barret, T., Benson, D. A., 2009, "Database Resources of the National Center for Biotechnology Information," Nucleic Acids Research**, 37**.

[74] "Semantic Web for Earth and Environmental Terminology (SWEET)," 2011, Visited: 1/11/2013.

[75] Sirin, E., Parsia, B., Grau, B. C., 2007, "Pellet: A Practical Owl-Dl Reasoner," Web Semantics: Science, Services and Agents on the World Wide Web**, 5**, (2), pp. 51-53.

[76] Kuiken, T., Rimmelzwaan, G., Van Amerongen, G., 2003, "Pathology of Human Influenza A (H5N1) Virus Infection in Cynomolgus Macaques (Macaca Fascicularis)," Veterinary Pathology Online**, 40**, (3), pp. 304-310.

[77] Horrocks, I., Patel-Schneider, P. F., Boley, H., 2004, "SWRL: A Semantic Web Rule Language Combining OWL and RuleML," W3C Member Submission**, 21**, pp. 79.

[78] O'Connor, M. J., and Das, A., 2009, "SQWRL: a Query Language for OWL," OWL: Experiences and Directions (OWLED), Fifth International Workshop, pp. 143.

[79] Fowler, D., Crowder, R. M., Guan, T., 2010, "Requirements for Semantic Web Applications in Engineering," pp. 1.

[80] "Protégé," 2013, <http://protege.stanford.edu/>, Visited: 2/6/2013.

[81] "Protege 3.4.8 (build 629)," <http://protege.stanford.edu/download/protege/3.4/installanywhere/Web_Installers/>, Visited: 11/18/2012.

[82] Ali, S. I., Jain, S., Patidar, B. L., 2012, "A Framework for Modeling and Designing of Intelligent and Adaptive Interfaces for Human Computer Interaction," International Journal of Applied Information Systems**, 1**, (2), pp. 20.

[83] Bailey, J. E., and Pearson, S. W., 1983, "Development of a Tool for Measuring and Analyzing Computer User Satisfaction," Management Science**, 29**, (5), pp. 530.

[84] Wanous, J. P., and Lawler III, E. E., 1972, "Measurement and Meaning of Job Satisfaction," Journal of Applied Psychology**, 56**, (2), pp. 95.

[85] Root, R. W., and Draper, S., 1983, "Questionnaires as a Software Evaluation Tool," CHI, ACM, pp. 83.

[86] Chin, J. P., Diehl, V. A., and Norman, K. L., 1988, "Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface," CHI.

[87] Nielsen, J., and Molich, R., 1990, "Heuristic Evaluation of User Interfaces," CHI, ACM, pp. 249.

[88] Molich, R., and Nielsen, J., 1990, "Improving a Human-Computer Dialogue," Computing Practices, ACM, **33,** pp. 338.

[89] Jeffries, R., Miller, J.R., Wharton, C., 1991, "User Interface Evaluation in the Real World: A Comparison of Four Techniques," HP Software and Systems Laboratory, HPL-91-03, New Orleans.

[90] Karat, C. M., Campbell, R., and Fiegel, T., 1992, "Comparison of Empirical Testing and Walkthrough Methods in User Interface Evaluation," CHI, pp. 397.

[91] Nielsen, J., and Landauer, T. K., 1993, "A Mathematical Model of the Finding of Usability Problems," INTERCHI, pp. 206.

[92] Lewis, J. R., 1995, "IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use," International Journal of Human-Computer Interaction**, 7**, (1), pp. 57-78.

[93] Likert, R., 1932, "A Technique for the Measurement of Attitudes." Archives of Psychology.

[94] Tullis, T. S., and Stetson, J. N., 2004, "A comparison of questionnaires for assessing website usability," Usability Professional Association Conference.

[95] Nielsen, J., Tscherning, C.C., Jansson, T.R., 2012, "Geodesy for Planet Earth," Springer, pp. 443-449, "Development and User Testing of a Python Interface to the GRAVSOFT Gravity Field Programs".

[96] "Qualtrics," 2013, <http://qualtrics.com/>, Visited: 10/30/2013.