Supporting Ontology Alignment Tasks By Organizing Ontology Mappings Using Edge Bundling

by

© *Muhammad Nasir*

A thesis submitted to the

School of Graduate Studies

in partial fulfilment of the

requirements for the degree of

Master of Science

Department of Computer Science

Memorial University of Newfoundland

May 2014

St. John's Newfoundland

Abstract

Ontologies are commonly used for knowledge representation and to exchange information between multiple applications. When similar information is represented by different ontologies, information sharing requires a mapping between corresponding pairs of entities. While ontology alignment algorithms have been developed to support such tasks, they generally do not offer entirely complete and precise mappings. Consequently, an important interactive aspect of the alignment process is the validation of automatically generated mappings and the addition of new mappings, by a knowledge manager. While visual interfaces exist to support these tasks, showing a large number of mappings can result in a significant amount of visual clutter. To address this issue, an edge bundling approach has been adapted to the constraints of an existing alignment interface. A user study was designed and conducted to evaluate the value of edge bundling in this context, with positive results for both mapping validation and addition tasks.

Acknowledgements

First of all, I would like to sincerely thank my supervisors, Dr. Joerg Evermann and Dr. Orland Hoeber, for their continued guidance, and support. They provided their insightful advices, showed how the academic research is conducted, helped in improving my critical thinking skills, and attention to detail. Most importantly, they constantly challenge me to do better in conducting high quality research and learning the skill of explaining the research better. I am very grateful to them and will always appreciate their invaluable guidance in improving my overall skill set.

I would like to thank the Department of Computer Science and Memorial University of Newfoundland for their generous resources without which this thesis would not have been possible. My research is financially supported by my supervisor's grant (NSERC Discovery Grants) and scholarship from the University's School of Graduate Studies and Department of Computer Science.

I would also like to thank the graduate students of the Department of Computer Science for providing their insightful comments and suggestions. All of my St. John's friends that made my transition here smooth, made my life fun and always provided moral support throughout the course of my research.

Last but definitely not the least, I owe my deepest gratitude to my loving parents, Muhammad Yakoob and Farida Yakoob, who are my ideals, best friends, strongest supporters, and always push me to strive for excellence and in becoming a better person each day.

Refereed publications from this thesis: Based upon the research work conducted in this thesis, a peer-reviewed publication has been made [58].

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Effective communication and collaboration among people, organizations, and software systems is the key to success in today's competitive world. However, such communication among software systems can be challenging due to their independent development. Software systems are built by people; due to disparate backgrounds and cultures, education, and languages, software developers may use different terminology to explain the same conceptual entity [80]. In addition, they may produce different hierarchical decompositions of difficult or complex domains within the software they create. This leads to different computer systems storing information in different formats, making it difficult for them to share and inter-communicate. Therefore, in order for such systems to share information, an effective mechanism is required that can provide translations between different data formats.

Ontologies provide a convenient mechanism to represent information about a domain by organizing their major concepts and forming hierarchical relationships between entities. They have become an important tool for sharing and re-using information among academia and industry [29], and their usage to support the sharing of information between multiple software systems has become increasingly important in recent years [18, 41, 43].

While communication and sharing is straight-forward when two systems use the same ontology, challenges exist when sharing is required between systems that use different ontologies to represent the same information. This may occur when systems are built in parallel, when one system is changed or updated, or when new sharing functionality is required. In these cases, information sharing requires ontology alignment (i.e., a mapping between the entities in the pair of ontologies) [42]. Such a mapping can then be used to automatically translate information from one ontology to the other. For example, in the Web service area, ontology alignments are used to translate one message format of an e-commerce application into another, completely automatically or semi-automatically [43]. This translation is needed to conduct business transactions between disparate databases, inventory control systems, and e-commerce systems. Thus, the challenge is to obtain accurate translations by resolving heterogeneity issues and discovering similarities among different ontologies representing comparable domains. The solutions researched and developed to meet this challenge are usually referred to as "ontology alignment approaches" in the literature [30].

Ontology alignment has been an active research area for over two decades [66]. It is still one of the most fundamental, time-consuming, labor-intensive, and crucial tasks in the information integration process [29]. Since the sizes of ontologies can range from a few concepts to several hundreds, manually aligning two ontologies is not practical when ontologies are large. Therefore, automatic ontology alignment algorithms are required that can automatically match entities between the two ontologies. Numerous ontology alignment algorithms exist to perform this task [66], with the output being a set of candidate mappings between entities across the pair of ontologies.

Fundamentally, ontology alignment is a difficult problem due to the complexities of human language. Human beings perceive the same things differently by nature [70]; thus when different developers design ontologies for the same domain, each may label and structure similar concepts and properties differently. For example, one ontology developer may label the price of the ticket in an ontology representing the domain of a travel agency management system as "price", but another developer may define the label as "cost" in a different ontology representing the same domain. Furthermore, in the hierarchical decomposition of the problem domain, one ontology may include the ticket price within a larger ticket entity, whereas another might include it within a travel itinerary. Usage of such different labellings and hierarchical decompositions of important elements of the concepts of an ontology may result in heterogeneity conflicts, and conceptual discrepancies [66]. The alignment process needs to resolve these to find the matches between the given ontologies.

Even though significant advances have been made as a result of natural language processing [36] and graph matching approaches [62], the end results of alignment algorithms are seldom certain and are often incomplete [29]. As a result, fully automatic alignment systems are not feasible. Since the human mind is well-suited to making sense of ambiguous and incomplete knowledge about the world, it may be beneficial to take advantage of such human capabilities by including them in the ontology alignment process [2].

Recently, researchers have started to analyze and give importance to various aspects of user involvement within ontology alignment processes [30]. Initially, using a matching algorithm, such systems produce candidate mappings between the given ontologies and present them to a knowledge manager. A common approach to address the uncertainty about these candidate mappings is to allow the knowledge manager to interactively validate the mappings, as well as add new mappings that were not detected by the automatic alignment algorithm. While a naïve approach might be to simply provide a list of mappings from which a knowledge manager can delete or add new mappings, doing so makes it difficult to consider the ontological structure during this process. This in turn might lead to incorrectly validated or missed mappings. A more effective approach is to present the ontologies and their mappings in a visual manner that is specifically designed to support the validation and addition tasks, and to provide interactive tools that aid in the completion of these tasks [28, 30].

The visual component of ontology alignment systems normally represents the two ontologies as tree structures on either side of the interface, and graphically depicts the mappings as edges between the corresponding entities. The tree structures provide an efficient way of representing the hierarchical nature of the ontologies, and allow for the navigation among large or complex ontologies. The representation of the mappings using edges allows the knowledge manager to visually trace the connection between entities in the ontologies.

Ontologies that are built to represent complex domain knowledge can be very large. When a semi-automatic alignment process is carried out on a pair of such ontologies using an alignment algorithm [66], the outcome is a significant number of candidate mappings that must be validated by the knowledge manager. The validation process includes tracing individual mappings, confirming the relationship, as well as identifying and adding new mappings. In order to complete the validation task, the knowledge manager has to process and manipulate each and every candidate mapping. Validating these mappings is the most fundamental but protracted task in the alignment process.

Since the visual space available to show the mapping edges is limited, problems may occur when representing a large number of mappings. When representing a relatively small set of mappings, the space can be used effectively, making it easy for knowledge managers to perceive and understand the mappings and compare the pair of ontologies. However, when real life ontologies are aligned, the number of mappings produced is great and presenting these mappings within the space available creates a visual mess.

These mappings are conventionally drawn as straight lines between the source and target ontologies trees, and in all but the simplest cases will include many edge crossings. Even with as few as 20 mappings, the usability of such interfaces quickly deteriorates due to visual clutter created by the edge crossings within the mapping region of the interface. As a result, the knowledge manager may experience difficulty in observing general patterns or trends within the mapping set, as well as in choosing and manipulating a particular mapping. Such clutter may also cause problems for finding a specific mapping or set of mappings.

While some studies have explored the use of curved lines to represent the mappings [31] or interactive highlighting to allow the knowledge manager to focus on a specific mapping [31, 67], the fundamental problem of trying to make sense of a visually cluttered representation remains. The potential solution for reducing the amount of visual clutter and improving the visual organization of mappings that is proposed in this research is to use edge bundling [45] on the representation of the mappings.

1.2 Approach

Edge bundling is the process of distorting the shapes of the edges in a graph to provide paths that are easier for the human eye to follow [45]. It has been proposed for providing an organized view of large graphs, grouping the graph edges by bundling their common pathways together [45].

In the context of ontologies, the bundling process can be considered as a potential solution for the visual clutter problem due to mapping edges crossings. By bundling the edges within an ontology mapping interface, much of the visual clutter within the mapping representations will be eliminated due to the clustering of the edges and the crossing of a few bundles rather than many individual edges. The grouping of mapping edges helps in providing an overview of the mappings results, which may help a knowledge manager to perform the mapping validation tasks with ease and increased performance. Instead of working on individual mappings, with bundling, the knowledge manager can work on mapping sets (i.e., those that are grouped into bundles).

Another advantage provided by edge bundling is the easy identification of potentially interesting areas within the source and target ontologies from which a large number of mappings are originating or terminating. Identifying these areas can help the knowledge manager in focusing on important concepts within the source and target ontologies. Due to the structure of ontologies, these areas may contain potential candidate concepts for finding and adding new relationships that were missed by the automatic alignment algorithm.

While edge bundling adds structure to the visualization of the mappings, it does so at the cost of introducing ambiguity. The drawback of using bundling is that the mapping edges are now grouped together making it difficult to trace a single mapping from source to destination. Since the knowledge manager is required to view each and every mapping before validating it, the same ability is also required while dealing with bundling. Therefore, in order to disambiguate the mappings contained within a bundle, some interaction mechanism is needed. As such, along with the implementation of edge bundling, an interactive focusing feature is also required, which can allow the knowledge manager to easily view and perform the relevant validation tasks for each mapping individually.

1.3 Research Questions and Methods

The primary goal of this thesis is to study the application of an edge bundling approach to ontology mapping representation. The following research questions will be addressed:

1) How can the edge bundling technique be adopted to the domain of ontology mapping representation?

2) What are the benefits and drawbacks of using edge bundling in the context of ontology mapping validation and addition tasks?

To answer these questions, the first step was to identify a state-of-the-art ontology alignment and visualization system, which could be used as a baseline system. The minimum requirements of this system were to handle the input of source and target ontologies, perform the automatic alignment of the ontologies, and finally present the candidate mappings within a visual interface. To address the first research question, after the identification of the baseline system, the next step was to study the constraints and requirements of the ontology mapping representation and modify the general bundling process to comply with these restrictions. In addition, to support the alignment related tasks (i.e., mapping verification and addition), interaction methods were developed for bundle disambiguation. The final step was to conduct an empirical evaluation to study the benefits of using edge bundling to support a knowledge manager's ontology alignment tasks. The results of this evaluation address the second research question.

1.4 Organization of Thesis

The remainder of the thesis is organized as follows. The following chapter outlines related work pertaining to automatic ontology alignment, ontology alignment interfaces, and edge bundling, along with a theoretical discussion on the proposed benefits of edge bundling. Chapter 3 explains the process of performing edge bundling within the baseline ontology alignment framework, provides the details regarding the implementation, and discusses the value of the approach in the context of the primary tasks of mapping validation and mapping addition. Chapter 4 provides the methodology, results, and discussion of the empirical evaluation, which was conducted to study, evaluate, and compare the benefits and drawbacks of the proposed method in comparison to the state-of-the-art baseline system. In Chapter 5, this thesis concludes with a summary of the primary contributions of this work along with an outline of future work.

Chapter 2

Related Work

This chapter begins by describing the concept of ontologies in detail, after which several common types of algorithms that are used in alignment systems will be discussed. Following that, the importance of knowledge managers in the alignment process will be described, which will lead to the discussion about the interfaces of the alignment systems to support knowledge managers. This section will also include a discussion about the value of visualization in designing such interfaces, as well as the theories and principles from the field of information visualization that apply to the representation of ontology mappings. A discussion on existing mapping representation approaches and issues will constitute the next section of the chapter. Details will also be provided on the chosen baseline system for this research.

2.1 Ontologies

"Ontology" is a term that originates from the field of philosophy where it is used to refer to the study of existence [39], as well as to describe categories of things and the nature of relationships that exists between these things. In the field of computer science, this term has evolved to represent the final product obtained when an abstract and simplified description of a domain is created [39]. The key concepts of the domain are represented using classes, and logical relationships are created among them by knowledge experts to reflect a common understanding of the overall domain.

Primarily, ontologies provide a standard vocabulary that can enable the sharing and re-use of domain knowledge [43], as well as allow automated computer reasoning and analysis [43]. Domain ontologies provide standard definitions of the domain concepts and the logical relationships that exist between them, which can be used by computers to extract common meaning automatically from the given information [81]. Using the capabilities of the domain ontology, large amounts of data can also be consistently classified and aggregated [81].

A traditional list-based graphical representation of a small ontology of a university domain is shown in Figure 2.1. The main concepts that define this domain are depicted as nodes and the relationships that exist between these concepts are shown in the form of lines connecting the relevant concepts. This is one example of a visual representation of an ontology; a more complete discussion on ontology visualization is presented in Section 2.3.2.

Ontologies have been used extensively in different application domains. In computer science, one of the important usage is in the area of the Semantic Web [43], in which they facilitate the information exchange between different systems, support query-answer services, provide knowledge bases that are re-usable, and enable the interoperability among various heterogeneous information systems [39]. They primarily provide semantics for the elements of a web document, as well as enable those

Figure 2.1: An example ontology of a university domain [29] generated using OWLViz [46] plugin in Protégé [60].

semantics to be used by intelligent information retrieval systems and applications [43].

There are many technologies and languages that have been introduced to specify ontologies. XML (eXtensible Markup Language) [6], is a data-describing language that uses tags to describe and define data on the Web in a structured fashion [18]. It is the basic underlying technology that can be used to write ontologies, which also provides the foundation for more specialized languages to author ontologies such as RDF (Resource Development Framework) [44] and OWL (Ontology Web Language) [79].

The idea behind these languages is to allow automatic processing of the content rather than showing presentable information to humans [43]. RDF [44] was developed to incorporate meaning to Web data elements by adding machine-readable metadata. It was the first language that was designed to support the Semantic Web [18]. RDF can be considered as a layer on top of the existing technologies that use XML. RDF Schema (RDF-S) [7] extends RDF and provides the basic structures required to create the fundamentals of an ontology, such as classes, properties, instances, and relationships (e.g., is-a, child-of) [18].

Currently, the most popular ontology development language is OWL, which was developed by the Web Ontology working group [14] under the WWW Consortium [79]. The OWL language supports more expressibility than RDF-S, providing the ability to define constraints and restrictions on the classes, attributes, and instances. Therefore it is an improved version of RDF-S, providing additional vocabulary with formal semantics [43].

Due to the widespread usage of ontologies, many currently exist that model and represent similar domains. Furthermore, different notions and terms can be used to describe the same things in the ontologies used within software systems. Software developers have a specific set of requirements and factors to consider before building an ontology for a software system. In addition, there is a possibility of human bias because of different perceptions and understandings of the meaning of a concept. Typically, a word or a set of words are employed to describe a concept, and it is possible that different combinations of words, synonyms, homonyms, and other relevant features of a language are used to define the same concept, which can be confusing. This independence in usage of language brings versatility and broad knowledge to an ontology, but may also lead to an ambiguous or incomplete definition about a concept.

To enable communication and sharing of information between multiple systems, discrepancies between their ontologies must be resolved to provide translation. Thus, information sharing requires ontology alignment, producing a mapping between the entities in the pair of ontologies [42]. In addition, alignment of ontologies representing a similar domain is required to provide a more integrated view of the domain. The emerging challenge is to resolve heterogeneity issues and discover similarities among different ontologies representing comparable domains. The approaches and systems developed to perform this challenging task are referred to as "ontology alignment" [30].

2.2 Ontology Alignment

Ontology alignment is a requirement for many ontology-dependent software systems in order for them to communicate with each other. For example, Semantic Web applications require effective and standard ways of communicating between different service providers [43]. These service providers expose characteristics of their services using ontologies. In order to exchange information, these ontologies need to be aligned [23]. A number of different types of alignment algorithms have been proposed in the literature to find mappings between the concepts of the given ontologies. Generally, a single or multiple types of alignment algorithms are used in over a hundred different proposed alignment systems [37]. Due to this widespread scope, an outline of the commonly used types of algorithms will be provided in this section. Furthermore, a brief discussion about systems that incorporate those types of algorithms will be presented after providing a formal definition of the ontology alignment.

2.2.1 Definition

Considering the source ontology as Os and the target ontology as Ot, these two ontologies are provided as input to the ontology alignment process, which produces the one-to-one or one-to-many mappings indicating the relationship that exists between the relevant concepts. With the help of these mapping relationships, it can be deduced that the corresponding ontology concepts represent the same conceptual entity.

The mappings are generally presented in the form of (Cs, Ct, i), where Cs and Ct represent the concepts from source and target ontologies while i shows the confidence level of similarity computed by the alignment algorithm. The range of the confidence level values is normally between 0 and 1. In order to be considered a mapping, the confidence level of the computed similarity must meet a pre-defined threshold value. A pictorial representation of this process is shown in Figure 2.2.

2.2.2 Algorithm Types

A number of different approaches exist for performing automatic ontology alignment. The most common approach used by the ontology alignment systems is string-based matching, which employs a name-matching approach to the labels of the ontology concepts. In these string-based methods, the computation of the similarity value between the concepts is based on the amount of relatedness among their names or labels. Certain string normalization techniques are applied to the input strings before the analysis of similarity is carried out (e.g., case folding, encoding, removal of blanks

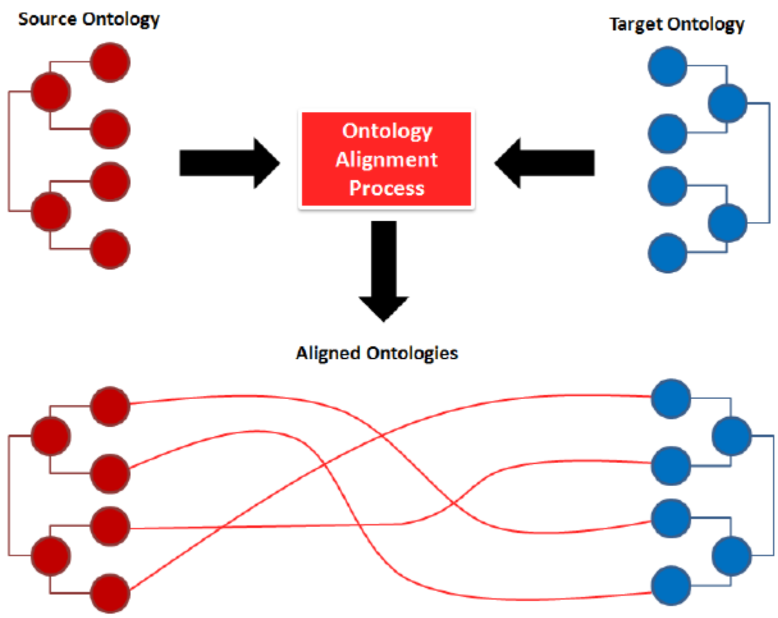


Figure 2.2: Typical flow of an ontology alignment process.

[37]). The outcome of the input texts to the string-based techniques can be reported as exact, which implies that they are equal, or approximate up to a level of confidence, which is obtained by using different similarity metrics. Examples of string comparison techniques include edit distance, soundex index, and prefix/suffix comparison [37]. Although string-based techniques provide support for determining concept similarity, they cannot overcome the complexity in the usage of a language to describe those concepts. Alignment systems that use string-based techniques include COMA [20] and COMA++ [1], OLA [27], Anchor-Prompt [61], and S-Match [36].

In order to overcome the difficulty faced by the string-based approaches, language-based approaches are added to complement the string-based approaches. They provide more advanced support for concept-label matching by incorporating methods to counter the complexity introduced by languages due to the use of synonyms, homonyms, and phrases. Some methods include tokenization, removal of stopwords, and reduction of a phrase or sentence to its stem words [37]. Ontology alignment systems that use the string-based techniques as well as language-based analysis of text include COMA [20], COMA++ [1], OLA [27], Anchor-Prompt [61], and S-Match [36].

The string-based and language-based analysis techniques offer a syntactic level of matching. However, the quality of alignment can be greatly improved by performing semantic matching analysis on the concept labels. Inclusion of domain-specific thesauri and lexical databases (e.g., WordNet [33]) for the matching strategy enables semantic-based comparison, which is often referred to as linguistic-based matching [37]. Since ontology concepts are organized in the form of hierarchical relationships, the usage of synonyms, homonyms, etc., (provided by a linguistic resource) in the matching strategy can be advantageous. Doing so also helps in determining the nature of the relationships that might exist among the concept labels, such as equivalence or generalization [37]. Systems that perform matching using linguistic resources include OLA [27], Cupid [55], and COMA [20].

The alignment strategy can be further improved by including type-based methods, which use information that defines the type of concept (e.g., integer, float, string, date, etc.) to find appropriate matches. In addition, the limits of the values on such concepts is used to calculate the similarity among two concept labels. For example, if a concept from an ontology is of type integer and has a range of 0 to 2500, it can be marked as relevant to a similar concept with a different label in a different ontology with the same type and range [37]. Examples of systems that also use this method are OLA [27] and COMA [20].

Along with above approaches, some ontology alignment systems also add techniques such as decision-theoretic approaches [72] that ask for the knowledge manager's feedback (i.e., accept or reject alignments) to select from the available alignment strategies (i.e., semantic, lexical, structural) for the given pair of ontologies [21]. For example, if the knowledge manager accepts the mapping suggestion that was found using lexical alignment strategy, its importance will increase for the next alignment task among other alignment approaches. In other words, the next time the knowledge manager will be provided with the suggestion produced by the lexical strategy on top of others. Using the knowledge manager's feedback in the alignment process is highly desirable, but it is difficult to achieve [9]. Systems that use decision-theoretic approaches are COMA [1] and QOM [23].

The current state-of-the-art method for matching is structural-based matching, which uses the hierarchical structure of the concept in addition to the information provided by the same single concept and following other trivial alignment strategies in order to compute mappings [37]. The main idea behind this type of matching is to consider the ontologies as graphs, and compare the hierarchical relationships of different concepts as sub-graphs using graph matching algorithms. For example, if the child concept of the parent concept is the same in terms of label and other attributes to a child of a different parent concept (even if the parent concepts labels are not similar to each other), they should be matched to each other.

In this dimension, similarity flooding [56] is a successful and common technique used in the context of structure matching. In this method, the idea is that if the nodes are similar to each other, their neighbors should also be similar. This approach propagates this similarity definition along the deeper levels of inheritance of the concepts being considered for matching. The Anchor Prompt [62] algorithm focuses on finding the related semantics in the ontologies using the structural similarity analysis.

Initially, the given ontologies are transformed into a labeled graph structure. The knowledge manager then selects some anchors (i.e., main entities) within the graph, or the system can also provide those automatically by performing lexical matching. The algorithm then analyzes the subgraph limited by these anchors and determines which classes frequently appear in similar positions on similar paths. These classes are likely to represent semantically similar concepts. The results produced by this approach are promising; therefore it is now used in many alignment systems as the underlying alignment algorithm [3, 31] including PROMPT [59], COMA [1], OLA [27], QOM [23], and RiMOM [54].

Since a single type of alignment algorithm cannot provide accurate and complete alignment results, multiple alignment algorithm types are incorporated to compute the alignments. Starting from basic alignment approaches such as string-based and language-based tools, alignments systems include combinations of different types of alignment algorithms to compute more accurate results. Table 2.1 presents the list of popular alignment systems along with the type of the algorithm that has been employed.

2.2.3 Human Involvement Within the Ontology Alignment Process

Integrating different ontologies and computing associations between their concepts is considered a difficult task. Despite being an active research area for many years [30], the full automation of the ontology alignment process that can produce quality

Table 2.1: List of popular alignment systems along with the types of algorithms employed within them to calculate mappings.

Alignment Systems	Algorithm Types				
	String & Language	Semantic	Type	Decision-theoretic	Structural
COMA	X	X	X	X	X
COMA++	X				
OLA	X	X	X		X
PROMPT	X				X
S-Match	X				
Cupid		X			
QOM				X	X
RiMOM					X
Anchor-Prompt	X				

mappings on its own is far from being possible [37]. Due to the complexity of human language and variations in how the same information may be encoded in different ontologies, the output of automatic ontology alignment algorithms may include errors or omissions [66]. Hence, there is a need to incorporate human decision-making within the ontology alignment process. The human mind is better suited to understanding the nuances of language than automatic algorithms, and is very effective at making decisions based on incomplete or conflicting information [74]. The main idea is to produce better quality mappings by taking advantage of a human's versatile knowledge about the domain and ability to interpret language, combined with the efficient ability of machines for storing large amounts of information and performing repetitive tasks [74].

Humans are good at developing novel solutions, adapting to the current situation, changing the strategy if the original solution fails, and making decisions [74]. On the other hand, machines can operate with vast amounts of data, recall information accurately, perform repetitive tasks quickly, and can maintain performance for a longer time even if carrying out multiple tasks simultaneously [74]. Therefore, it is an active area of research to combine these human capabilities with immense abilities of machines, and find a good balance between automation and manual work in performing tasks at hand [73].

Systems combining human and machine capabilities in the context of ontology alignment are referred to as semi-automatic approaches in the literature [28] and are gaining importance in the ontology alignment research domain. They are generally comprised of two main parts: first is the underlying alignment algorithm that is used by the system to find the candidate mappings set, and second is the design of the interface that is provided to the knowledge manager in order to support ontology alignment tasks.

The semi-automatic approaches build upon the work of automatic methods, providing information about the candidate mappings to allow the knowledge manager to either confirm or reject each mapping. This mapping validation process is more efficient than manual alignment due to the focused decision making that is supported by the suggestion of candidate mappings produced by the automatic algorithms. Since it may be possible for the automatic algorithms to miss important mappings, these semi-automatic approaches must also allow the knowledge manager to manually add mappings.

Semi-automatic approaches for ontology alignment have become crucial since automatic processes cannot satisfy the quality requirements of mapping related tasks [37, 29]. Involving humans in this process can make the alignment process more efficient, feasible, and useful [37]. With their versatile general knowledge (which is difficult to reproduce automatically in machines) along with the sophisticated and ample processing power of their visual system, humans bring invaluable advantages in solving problems [29]. Therefore, in addition to the development of the state-of-the-art alignment algorithms, careful consideration is also given to the design of the interface of these systems that facilitates the interaction of knowledge managers with these systems [37].

2.3 Ontology Alignment Interfaces

A wide range of interfaces have been proposed to support the human element of mapping validation and mapping addition within semi-automatic ontology alignment systems [28, 37]. A common theme among these interfaces is the graphical representation of both the ontologies themselves as well as the mappings. Rather than simply providing a textual list of candidate mappings, these interfaces take advantage of information visualization techniques to convey both the structure of the ontologies and the mapping information to the knowledge managers in a graphical format.

2.3.1 Information Visualization

Information visualization [82] is an interdisciplinary field that combines human-computer interaction, user interface design, cognitive psychology, computer graphics, and information systems to bridge the gap between humans and machines. The core idea is to build interactive visual methods to represent abstract data in order to amplify cognition and help in uncovering different aspects about the data. Information visualization enhances human cognition by using computer supported, interactive, and visual representations of abstract, non-spatial, semi-structured, or hierarchical data [49, 53, 82]. The visual methods developed for representations of abstract data take advantage of the swift processing capability of the human visual apparatus [82].

The visual perception system of humans is very powerful and quick in automatically interpreting certain visual features. Information visualization takes advantage of this quick processing ability to enhance the understanding and discovery of new insights about investigational data [53]. The information visualization field has been utilized to address the difficult problem of ontology alignment, which has resulted in a number of different ontology alignment interfaces [1, 20, 27, 31, 37, 59, 66].

Considering the amount and complexity of data that is available for different purposes, effective visualization representations for this data are required. These can be developed using relevant information visualization theories and principles to effectively view, evaluate, manipulate, and explore such data. Visual and interactive representations can be further enhanced by taking into consideration the user's expectations from the available information. The implementation of the ideas and the chosen design methods for the presentation of information, along with the evaluation of the product are also important aspects of information visualization.

In the information visualization domain, theories and principles that help in understanding the working of the human mind when perceiving objects, interpreting data, and inferencing logical conclusions out of visual representation of the presented information are also studied extensively. There are many well-defined theories and principles such as pre-attentive processing, Gestalt Laws, and colour theory, which can be effectively used to build useful visualizations.

One phenomenon that occurs constantly in the human visual system is referred to as pre-attentive processing. The visual system of the human brain operates naturally at a very fast rate, which allows fast and seemingly parallel processing of information as soon as it enters through the retina. By studying the visual elements and conditions that trigger pre-attentive processing, data attributes can be mapped to visual elements that highlight important aspects of the presented information to enable quick processing. The main notion is that the information that is required to be processed pre-attentively, needs to represented carefully so that it stands out from other information. Another important fact is that only a small number of items that are visually different from the rest can be pre-attentively processed. This number can be found by performing repeated experiments for the given information and by measuring the response time of finding the information from the rest. Failing that, the pre-attentive processing cannot be achieved as the human visual system will not be able to detect any difference quickly and focus on the important information, which will lead to the typical option of performing a serial search to look for the required information.

There are a number of different visual variables that can be pre-attentively processed (e.g., curvature, line orientation, line width shape, size, number, etc). A curved line representing a valuable element of information can be pre-attentively processed and quickly identified when all the other data near this curved line is shown as straight lines [82]. In terms of line width, the width of the line defining substantial information can be increased to enable its quick processing while keeping widths of the other lines constant. Similarly, a limited number of oval shapes can be processed swiftly among a large number of circle shapes on a given display. Furthermore, a huge square can be easily differentiated from the set of smaller squares. Similarly, among the sets of a certain number of circles (e.g, four circles) closely placed to each other distributed over a space, a set containing a smaller number of circles from the rest (e.g., only one) can be swiftly distinguished. Other visual variables activating pre-attentive processing are also available than can be used for presenting different kinds of information within information visualization systems [82].

In order to design effective information visualization systems for users, it is important to study and understand how humans perceive patterns and interpret information. In this direction, an important step was taken by a group psychologists in 1912 from the Gestalt school of psychology, who designed a series of experiments and performed a number of evaluations to examine pattern perception [82]. The results of these evaluations are commonly known as the *Gestalt Laws of Pattern Perception* in the field of information visualization.

The Gestalt Laws [50] fundamentally deal with two types of concepts: perception of relationships, and perception of foreground from background. In general, all the Gestalt Laws pertain to understand how the human mind perceives patterns. However, three Gestalt laws that focus on how humans automatically infer the existence of a relationship between things when they are near one another, visually similar to each other, and connected to one another, are relevant and important for this research.

The law of *proximity* suggests that objects that are positioned near each other are automatically perceived to be related. Due to the usefulness of this spatial proximity interpretation, it is commonly used as a design principle in information visualization systems [82]. Visual objects that represent data belonging to the same group can be placed nearby each other to facilitate the automatic perception about their relatedness. It is considered to be the simplest way to accentuate the relationships between different data elements and is commonly used to visually represent clusters [82].

As emphasized by the law of *similarity*, visual objects that are drawn similarly are perceived to be related [82]. This law is commonly used as a design principle when building user interfaces. For example, same-size black colour filled circles will be perceptually grouped, when placed in the same interface as same-size circles that are filled with red color. Therefore, similar type of visual techniques are generally used to show elements that represent related data.

The law of *connectedness* [82] has resulted in a commonly used design principle to show relationship between two objects. This law asserts that the simplest and most effective way of expressing that any kind of relationship exists between two graphical objects is by connecting them through a line [82]. This design principle has been used and extended into many graphical forms and structures that are used to show relationships between contained objects (e.g., graphs, trees, node-link diagrams).

The Opponent Process Theory of Colour [57] provides details about the interpretation of color by the human vision apparatus. It describes the process through which the human eye can perceive the difference between colors and the ways in which this raw information is processed within the human brain. The fact that we can perceive different colours, is a function of our cone sensitivity to different wavelengths of light. The ability for the mind to easily determine the difference between red and green, yellow and blue, and black and white is explained by the Opponent Process Theory of Colour.

According to this theory, the human brain combines the stimulus from the available three cones (long, medium, short) before processing them, which results in six elementary colours that are perceived as opponent pairs along three channels: red-green, yellow-blue, and black-white. We can see the differences between colours on these channels much more easily than with arbitrary colours. For example, a red object can be seen within a majority of green objects much better than if it is surrounded by yellow objects. Using the guidelines of this theory, the six different colours forming these channels are effective choices to encode the distinctness of different types data attributes [82]. Objects that are colored using these six primary colors are automatically perceived as different, and can be carefully used to represent different kinds of information on the same screen. This theory also provides guidelines to use colors to present information containing order and amount or intensity. For example, moving along the black-white channel, a darker color can be used to fill a shape or area representing a bigger amount or higher rank, and a lighter shade of the same color can be used to show a less amount or lower rank [76]. Similarly, range of negative values to positive values can be shown using the red-green colour channel.

Another important aspect of information visualization is the support of interaction. It is assumed that many information visualization systems today have basic interaction features such as zooming, filtering, and focusing. Interaction is typically integrated into the visual representations, which allows for a change in the presented information [85]. With the interaction support, users can administer and manipulate the presented information according to their understanding and requirements. Therefore, interaction techniques can improve the user's cognition, allowing them to easily process and investigate the data at a deeper level, and then make the relevant decisions. Information visualization systems without interaction still exist; they are basically static images and a very limited number of tasks can be performed using such systems [19]. In the absence of interaction support, the systems can easily be classified as not user-friendly and their value can be adversely affected, specifically those systems that deal with large datasets [85].

2.3.2 Representation of Ontologies

The field of information visualization provides theories and principles to deal with complex and large amounts of data. Since ontologies contain large collections of data structures that represent constructs of domains, visualization has an enormous potential in dealing with ontologies [53]. Consequently, many ontology visualization systems incorporate different visualization mechanisms for presenting ontologies.

The purpose of providing ontologies in a visual format is to allow knowledge managers to take advantage of their visual processing capabilities. With an effective visual representation, knowledge managers are able to readily perceive, interpret, and make sense of the features, relationships, and structure among the data [10]. Therefore, showing ontologies in graphical format enhances the knowledge managers' cognitive abilities to visually process the presented ontologies, which can help them in effectively performing the associated tasks.

Because the ontologies themselves are structured as a hierarchy of concepts, the logical method for representing them is in a tree structure [48]. This approach is used in virtually all of the prior research on ontology alignment interfaces [37, 53]. The structure of ontologies along with their relationships are shown as a hierarchy when tree-based representations are employed. Under this category, the systems generally present ontologies similar to the windows-explorer view [48]. Many systems that are specifically designed for ontology development fall in this group, including Protégé [60], OntoEdit [77], and OntoRama [24]. Often knowledge managers are already familiar with these methods, therefore these systems generally have high acceptability rate and usefulness [48]. Since the ontologies themselves may be much larger than what can fit on a regular computer screen even using the tree structure, interactive features such as node collapsing/expansion, vertical scrolling, and zooming are often implemented.

Tree-based representations are highly structured and regular in their information layouts, and are simple to implement and display. This representation offers a clear view of the class names and hierarchy, and the labels representing the meaning of the nodes. Since every node is individually positioned and allocated a defined space, tree-based representations generally have no problems in terms of overlapping and occlusion of node labels [48].

However, researchers argue that the tree-based representations typically make inefficient use of the available screen space [32]. The root side of the tree is left completely empty, leading to overcrowding in other parts. Even the trees of a hundred nodes often need multiple screens to be completely displayed, or require scrolling since only part of the diagram is visible at a given time. In addition, certain interaction issues, such as having to drag the scrollbars to navigate, lead to negative opinion about their usefulness [48] as it is difficult to keep track of the concerned nodes with the scrollbar when the tree is large. OntoViz [75] address this space-limitation problem by giving an option of interactively collapsing nodes. The knowledge manager can select the nodes which they would like to see and the selected concept is then shown with its sub-hierarchies or related nodes. In this direction, SpaceTree [64] uses expansion and contraction options for its sub-hierarchies to avoid the problem of space limitation. The evaluations suggest that SpaceTree [64] performs better in comparison with others in specific ontology-related tasks such as finding a particular node.

Some researchers have explored the use of unstructured graph representations for the ontologies [48, 53]. The lack of structure of these representations implies that they can be very compact and provides advantages in terms of available space on the display. However, they are less effective for the purposes of ontology alignment due to the fact that the hierarchical structure of the ontology is difficult to perceive and understand. Furthermore, it can be rather difficult to compare subsets of the ontological structure. Because of these problems and the prevalence of tree-based representations, in this work the tree-based representations have been adopted.

2.3.3 Representation of Mappings

In alignment systems that use tree-based representations, source and target ontology trees are normally shown on opposite sides of the screen while the middle region is used to represent mappings between the entities. In this direction, the source and target ontology tree placed are on the opposite sides of the screen in PROMPT [59, 62], however, the mapping representation is replaced by a list component instead of straight or curved lines, which displays all the produced mappings in a list and is shown in the middle.

The mapping between two concepts represents the association between two data elements. Since humans interpret two things as being related if they are connected by lines as suggested by the Gestalt Law of *connectedness* [50], the easiest method to make users infer and trace the connectedness between the two matched ontology concepts is by linking them via a line. The relatedness encoded using lines is automatically perceived and quickly processed through the pre-attentive capabilities of the human brain. This encoding can be used by systems to represent association among objects (i.e., mappings and matched concepts).

While early approaches used straight lines to connect associated entities between the ontologies, there are visual difficulties when following such connections due to the sharp corners that are created by edge crossings. For example, AgreementMaker [16] displays the hierarchical relationship between the ontology classes (i.e., the input ontologies) using the tree-based representation (see Figure 2.3). The source and target ontology trees are placed on the opposite sides of the screen, while straight lines are used to represent the mappings between the mapped class nodes [37]. One feature is that the amount of similarity in percentage among the mapped concepts is displayed along the mapping lines as a number, which can be noticed easily by the knowledge manager. However, visually encoding this information along the mapping line causes a visual mess. Another feature of the system is that mapping lines can be filtered according to the similarity threshold set by the knowledge manager. More information about an individual concept in the two given ontologies is made available separately in a detail window when the knowledge manager clicks on the node representing that concept. The knowledge manager can also add their own mappings by selecting any two nodes and then trigger the relevant operation in the menu.

Since the human eye can more readily follow curved lines [82], some ontology alignment interfaces now use curves to represent the mappings [1, 31]. COMA++ [1, 42] has a similar graphical user interface to AgreementMaker [16], showing the ontologies on the opposite sides of the screen (see Figure 2.4). However, mappings lines found by the underlying mapping algorithm are drawn using curved lines rather than straight. The features in terms of relevant mapping operations (i.e., validation and rejection) are also similar, such as allowing knowledge managers to define their own mappings for matches that the mapping algorithm missed. A confidence measure is associated with each mapping, which shows the level of confidence the mapping algorithm has for the correctness of the particular mapping.

The status of the mapping can also be conveyed by visual parameters of the curve. In some systems, colour is used to represent the difference between candidate mappings and confirmed mappings [1, 31]. Line style (e.g., solid, dashed) may also be used to convey this information [31], allowing the knowledge manager to readily identify features of individual mappings.

An open issue in this area is whether to visually represent mappings when either end of the mapping is not visible in the ontology representation. This situation may occur when a particular portion of one ontology is collapsed, or when an entity is not visible due to the scrolling of a large ontology. Some systems continue to show these mappings, with the end pointing to a collapsed node or being directed off the bottom or top of the display [31, 67]. Another method is to dynamically filter these mappings, adding them back in as the collapsed node is expanded or as the knowledge manager scrolls the ontologies [31].

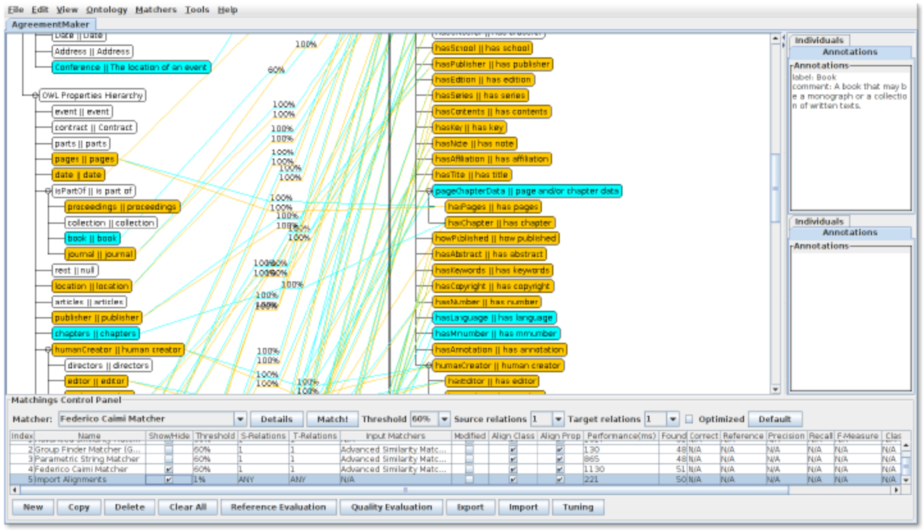


Figure 2.3: Screenshot of the interface of the AgreementMaker system [15, 16]

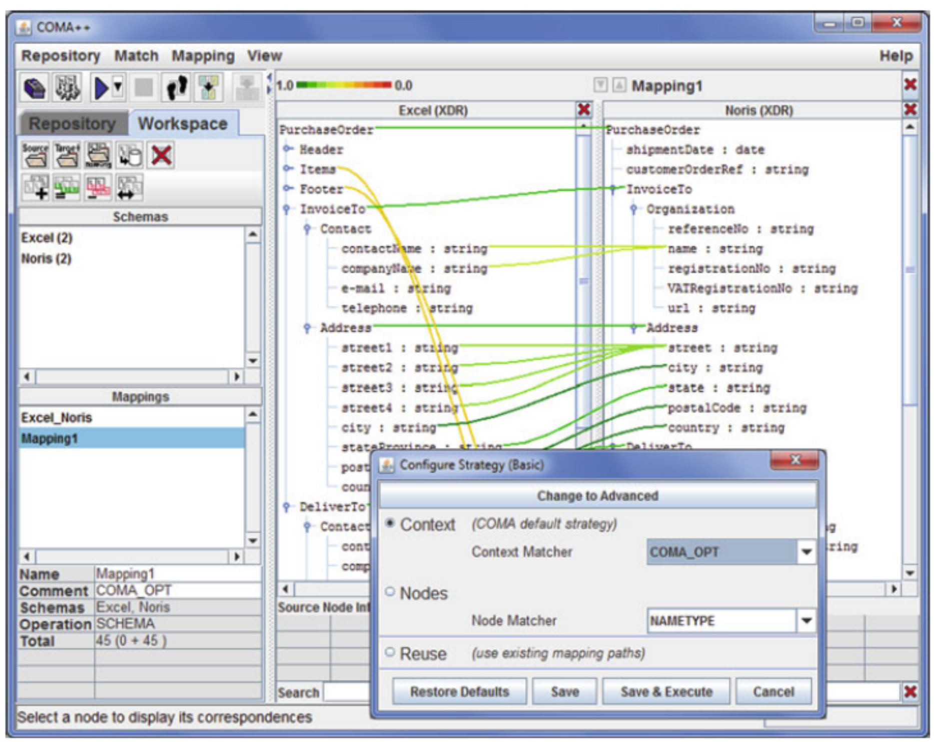


Figure 2.4: Screenshot of the interface of the COMA++ system [28]

2.4 Ontology Mappings Representation Issues

Improvements for the representation of ontology mappings has been given little attention in the literature compared to the development of alignment algorithms (Section 2.2.3) and ontology representations (Section 2.3.2). Knowledge managers of ontology alignment systems find two major shortcomings: a lack of availability of the basic visualization and navigation of mappings that relate two elements of the schemas, and the interface becoming easily disorganized when the ontologies and number of mappings are too large [67].

One of the fundamental problems with visually representing mappings with lines or curves is that edge crossings may occur when the order of the entities is not consistent between the ontologies. Although the use of curves to represent the mappings makes these edge crossings easier to follow, the problem is not solved with the use of curves. The candidate mappings produced by the alignments algorithms are generally very large in number. As such, when representing a realistic set of mappings, the edge crossings may cause a significant degree of visual clutter.

Clutter is an important concept in daily lives and is considered with great significance in the information visualization domain [69]. It is defined as a state in which surplus items or their representation or presentation style, may lead to a degradation of performance at some task [69]. For example, clutter can interfere with normal tasks such as searching for an important item (e.g., a research paper on a disorganized desk that has lots of other irrelevant papers laying around). In the language of user interface design, it refers to a situation that has the potential to create obstacles in performing a simple task such as visual searching. When designing user interfaces, it is one of the basic problems faced by systems that present extensive amount of information and have limited display space. Several attempts have been made to address this phenomenon and provide different mechanisms to measure visual clutter [69]. Many information visualization techniques also exist for addressing this problem [25].

In the context of ontology alignment interfaces, this clutter due to mapping edge crossings reduces the usability of the interface, making it difficult for the knowledge manager to visually trace a mapping from source to destination, or to make comparisons between mappings. While one approach might be to find an optimal ordering of entities in the ontology representations, there is no guarantee that such an ordering exists nor that it is an ordering that will make sense to the knowledge manager.

Most of the ontology alignment systems, including the ones discussed in Sections 2.2.3 and 2.3.3, provide very basic visualization tools and limited interaction options for manipulating mappings and to conduct alignment tasks. As the knowledge manager has to perform the exhausting task of validating candidate mappings produced by the system, there is a need for a better organization of ontology mappings combined with interaction support for ontology alignment tasks (i.e., mapping validation, and addition).

In this direction, BizTalk Mapper [67] attempts to employ advanced visualization approaches to lessen the cluttering problem, as well as many visual techniques to support the knowledge manager in performing mapping tasks. It supports different kinds of auto-scrolling. For example, given the source and target ontology trees presented on the opposite sides of the screen, when the knowledge manager selects an item, the relevant objects on the screen become prominent through highlight propagation. In addition, the interface is auto-scrolled to bring the highlighted mappings and the selected item in the middle of the screen. This way all the relevant links to the selected item are shown as much as possible. Another interactive feature is multi-select, which allows the manager to see multiple elements and the relations that exist between them. The mapping links are also bendable so that the managers can make changes in the formation of the mapping lines according to their needs.

Even with these features, when there are a large number of mappings, it remains difficult to avoid clutter in the representation. The fundamental problem is that each individual mapping continues to be represented by a straight or curved line within the interface. This causes significant edge crossings even with a small number of mappings edges that results into the introduction of visual clutter on the interface. An example of this problem can be seen in Figure 2.5. The proposed approach is addressing this fundamental issue through the improvement of the visual representation of the edges.

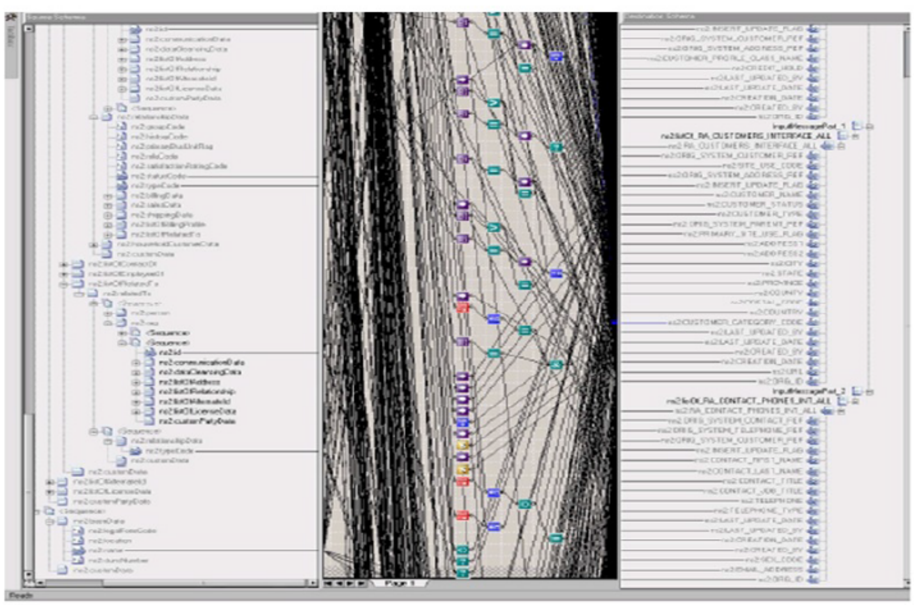


Figure 2.5: A snapshot highlighting the cluttering problem in BizTalk Mapper system [67].

2.5 Baseline System

CogZ [29, 31] is an interactive visual tool that was developed to improve the quality of a knowledge manager's interaction with the alignment system. It is a plugin of PROMPT [59], which uses the similarity flooding algorithm along with string-based and language-based algorithms (discussed in Section 2.2.3), to calculate the mappings between the provided source and target ontologies. It provides a similar interface to COMA++[42] and AgreementMaker [16], by representing source and target ontologies as trees and placing them in opposite sides of the screen. However, it improves the mapping representation by replacing straight lines with *Bezier curves* [84]. PROMPT is built within the framework of Protégé [60], which allows ontology inputs in popular file formats such as RDF and OWL, and facilitates the processing of ontology files into the format that can be processed by alignment algorithms in PROMPT.

The main interface of CogZ is shown in Figure 2.6. The marked areas "A" and "B" show the tree forms of the source and target ontologies, respectively. The "C" component shows the red-dotted and black-solid Bezier curves [84] that are used to represent the candidate and confirmed mappings, respectively. The top bar panel "D" shows buttons to execute different filter operations, and action buttons related to mapping decisions. Finally, the window marked as "E" is used to list the candidate mappings, completed or marked mappings, and neighbor nodes of the selected node. When the knowledge manager hovers over a mapping curve, a tool tip containing more details about the represented mapping is displayed. This method provides the manager with easy and quick access to the important information about the mapping, which is useful in making a correct mapping validation decision (accept or reject) [28].

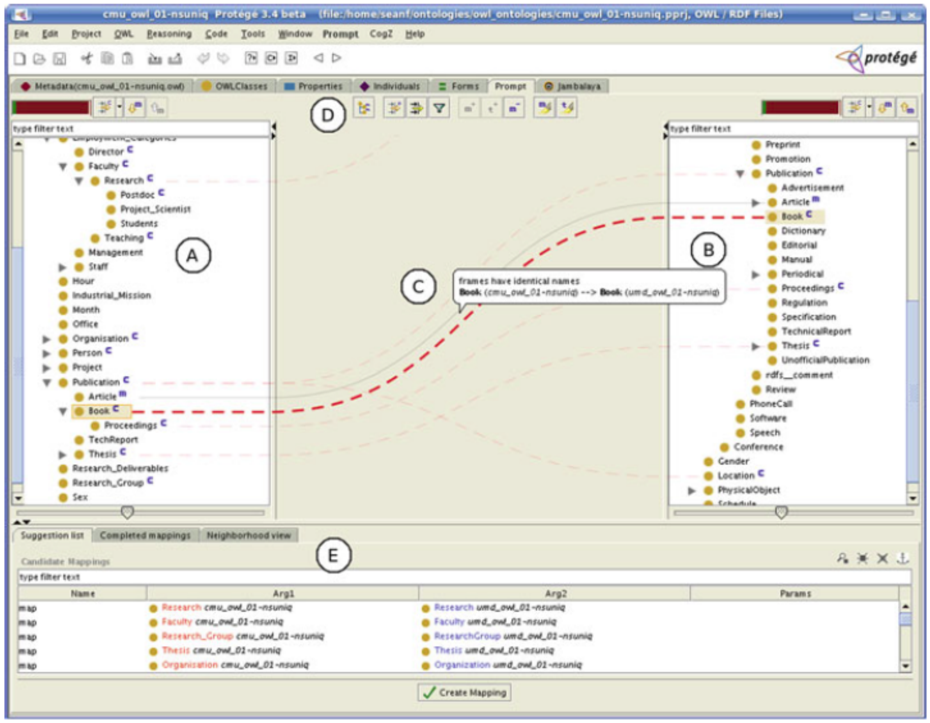


Figure 2.6: View of the main interface of the CogZ system [28].

The process of adding a missing mapping is very simple. After the knowledge manager has located any two unmapped but relevant concepts from the source and target ontologies, the manager is required to select both of the concepts. After the selection, the knowledge manager is required to click on the "m+" button on bar panel "D" to create the mapping between the selected concepts, and add the new mapping to the existing mapping set. Immediately upon the click, a new solid curve is drawn reflecting a mapping between the selected concepts.

The system also provides important features such as an incremental search on the source and target ontologies that are activated as soon as the knowledge manager starts typing in the search bar. Only those ontology concepts that match the keystrokes of the knowledge manager are left visible in the relevant tree window. It also provides a filtering operation on the ontologies as well as on the produced candidate mappings [28]. For example, the knowledge manager can filter the ontology tree concepts to only show those concepts for which mappings exist. In addition, the ontology trees can be filtered to show only those concepts that are connected through candidate or confirmed mappings.

The highlight propagation feature of CogZ allows the knowledge manager to focus on the selected-individual mapping by making all the other mappings semi-transparent [28]. The system also provides options such as semantic zooming [35] and fish-eye views [34] for the source and target ontology trees. The semantic zooming supports different size settings for the ontology trees, which can be set by knowledge managers according to their preference. Since the size of the ontology trees affect the number of entities visible on the screen at a given time, the number of relevant mappings that are visible also change accordingly. The fish-eye view helps in handling large ontologies by keeping the font-size constant for the selected concepts, while decreasing the font-size of the rest of the concepts according to their relatedness to the selected concepts [28]. This method allows the manager to see more of the ontologies in the same available space (see Figure 2.7).

The CogZ system [31, 29] incorporates many features that are designed to support the human decision processes, as well as employing some remedies to lessen the common problems faced by knowledge managers. It was developed to investigate the benefits of providing better support for ontology mapping to knowledge managers using the alignment system PROMPT [62], which contains current state-of-the-art mapping algorithms to calculate mappings but has a limited user interface. CogZ employs some of the features in BizTalk Mapper [67], but instead of the auto-scroll

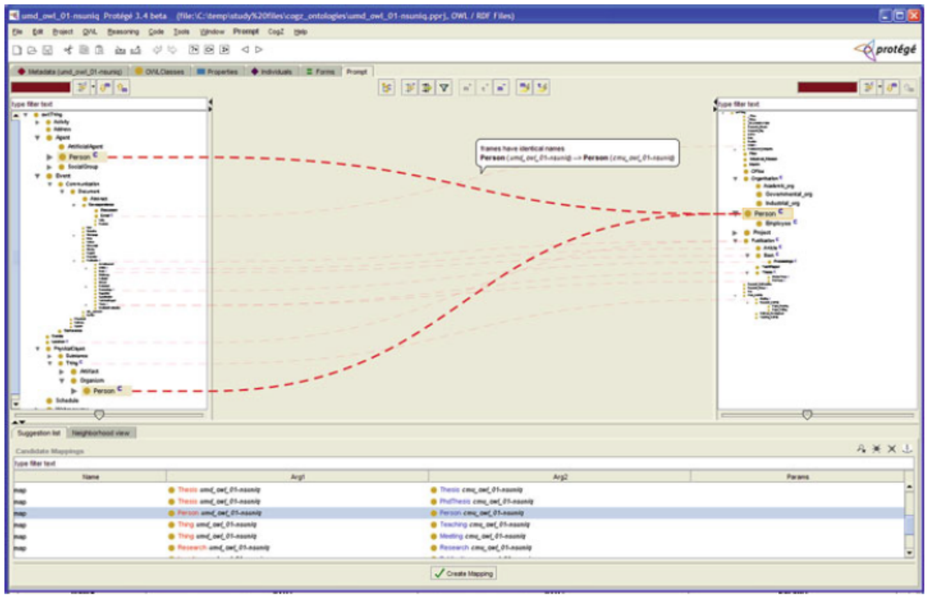


Figure 2.7: The fish-eye zoom feature of the CogZ system [28].

feature, it uses "user-guided" scrolling. For example, when the scrolling activity in any source or target tree is detected, the location of both trees is synchronized depending on the selected option. As a result of the interaction with a mapping, the appearance of ontology trees and the mapping are always kept concurrent.

Because of the above features, CogZ [31, 29] is considered as the state-of-the-art in terms ontology alignment interfaces. However, the system still faces the problem of visual clutter when the number of mappings to be shown is large (see Figure 2.8). Therefore, to improve the representation of ontology mappings over this state-of-the-art interface, CogZ was selected as a baseline system for this research. As an added benefit, it is an open-source project that facilitates the integration of new features.

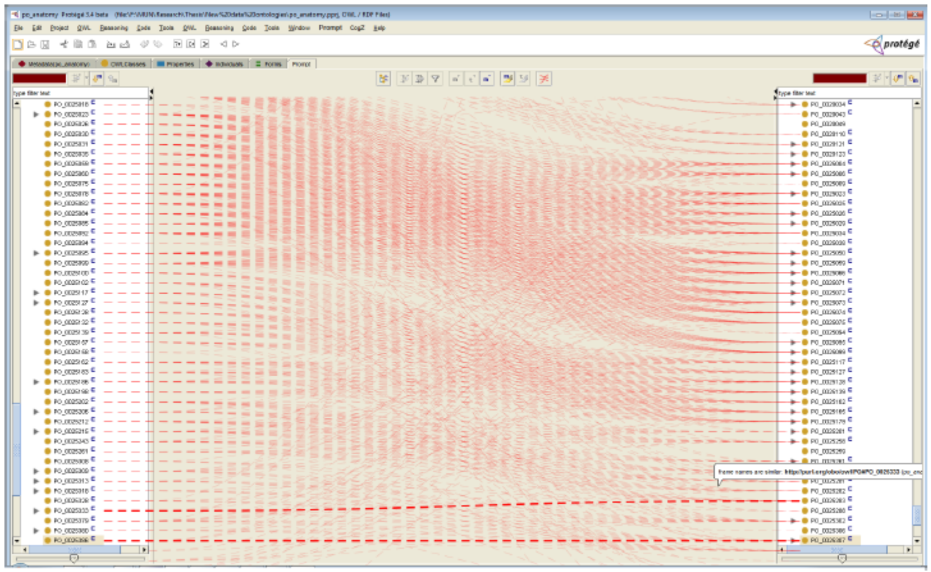


Figure 2.8: Results of presenting a large number of mappings on CogZ's interface [28]. The light red-color dotted edges represent the candidate mappings, edges turn darker when the knowledge manager hovers over them and the tooltip is also made visible.

2.6 Edge Bundling

The amount of visual clutter on a visual interface can potentially be decreased by grouping or categorizing the presented data items according to their similar characteristics [69]. Following this concept, edge bundling [45] was developed within the graph visualization research community as a mechanism for simplifying the visual representations of large and complex graphs. In such graphs, edge crossings may cause a significant amount of visual clutter, making it difficult to discern patterns in the graph structure. By bundling edges that have sources and destinations in common regions of the graph into clusters, the patterns within the graph can be made more apparent. That is, replacing many individual edges with a much smaller number of

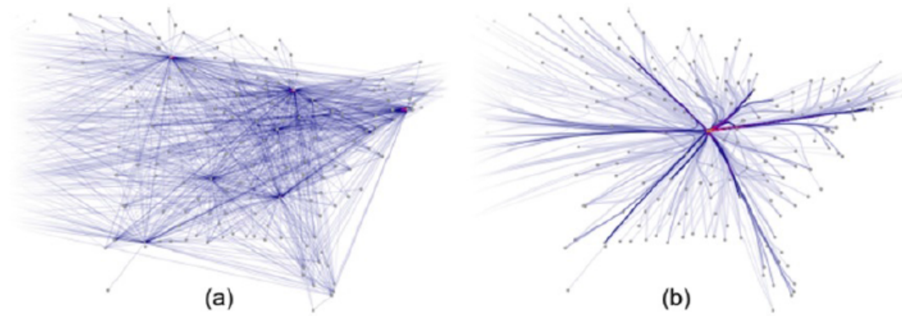


Figure 2.9: The snapshot of a system from [45] presenting the migration data between different parts of the world using (a) without edge bundling (b) with edge bundling.

bundles not only "cleans up" the clutter in the visual representation, but bundles can be more readily followed by the human eye.

An example of the edge bundling algorithm over a graph representing migration between different parts of the world is shown in Figure 2.9. Since there are too many edges connecting different nodes, it is difficult to determine any patterns from Figure 2.9(a). However, after the bundling is applied (Figure 2.9(b)) it is comparatively easier to comprehend the presented information and make the relevant inferences about the presented data.

2.6.1 Theoretical Foundation

Edge bundling provides a clear advantage in terms of efficient use of the available visual space on the screen and reducing visual clutter. Different theories from the domains of information visualization and cognitive psychology also support the notion behind the edge bundling. Therefore, in addition to the streamlining of the visual representation, there is also a theoretical foundation for the value of edge bundling.

2.6.1.1 Gestalt Laws

As introduced in Section 2.3.1, the Gestalt Laws are commonly used in the domain of information visualization to explain human perception of relationships among graphical entities [50, 82]. The connection between the related entities of the source and target ontologies is shown using mapping lines. Therefore, the connectedness will be automatically interpreted due to the law of *connectedness*. However, the notion of relations between the relevant concepts between ontologies can be conveyed via the other two principles (i.e., proximity and similarity).

The laws of *proximity* and *similarity* suggest that the human mind will judge elements to be related if they are near one another and if they look similar [50]. In other words, objects that appear closer and look visually similar to each other are automatically considered related to each other. These laws of human perception are repeatedly translated into several design principles for user interfaces. They can also be used to present different types of information in such a way that the intended understanding of relationships within the presented information on the interface can be automatically perceived by the user. These laws are also applicable for this research that provide support for such visual methods that can be employed to represent proximity and similarity.

The edge bundling process modifies segments of the individual mapping edges and brings them closer to each other to form bundles. The effect is that almost all of the individual mappings edges are grouped into a number of bundles on the screen. Following the law of *proximity*, the mapping edges present in these individual bundles will be interpreted and perceived as being related mappings because of their close proximity to one another. These also reflect similarity in subsets of the structures of the two ontologies. This will allow the knowledge managers to quickly understand, relate, and evaluate the mappings present in the bundles accordingly, which may help in performing the associated tasks more efficiently and effectively.

Since the position of edge segments are modified to be included in a bundle, their shapes have also modified and the final result is that the overall shape of the edges within the bundle are visually similar. According to the law of *similarity*, this will further enhance the impression of relatedness of these edges, which will help knowledge managers to use this information to their advantage while performing the relevant tasks. These perceptions of relationships among the edges lead to two additional theories regarding human reasoning: categorization theory [68] and cognitive load theory [12].

2.6.1.2 Categorization Theory

Categorization theory provides evidence regarding the cognitive process by which humans organize information into categories or classes [63]. It also explains the types of concepts and why humans classify things [63]. Categorization is fundamental to our thoughts, action, speech, and perception, as whenever we are analyzing any situation we are categorizing not only scenarios, but things and actions. Logical reasoning and conclusions about a situation are also closely tied to categorization [68]. Our natural capability to categorize knowledge helps us to deal with the information presented to us, and aids in our decision-making processes. The more effective a categorization can be, the better a person is able to perform in drawing a conclusion [68]. Without categorization humans will not be able to function completely, therefore it is important to study categorization as it is central to understanding our thinking process [52].

The basic principle for forming categories provided by the theory describes the functionality of categorization systems, and stresses that the fundamental task of the categorization system is to provide maximum information with the least possible cognitive load on the user [68]. Categorization can reduce the cognitive load associated with the storage and organization of knowledge. With effective categorization, the user can gain maximum information at minimum cognitive expense [63]. Therefore, problem solving systems can benefit by supporting the user's categorization efforts by automatically grouping related information. As such, categorization theory suggests that if information is presented such that it is already organized, the cognitive work required to understand and analyze the information will be reduced.

2.6.1.3 Cognitive Load Theory

Cognitive load theory [78] states that learning can be enhanced by the presentation of information, and explains the relationship between memory load and the ability to solve problems. Human working memory is limited and hence it can process only a limited number of items at a time. Although the theory focuses on the relationship between the human memory capacity and how it effects the problem-solving skills or strategies of users, interesting constructs and relationships can be inferred from this theory. It clearly illustrates that cognitive load resulting from information overload severely affects a user's performance.

Related to cognitive load theory, the *split-attention effect* [13] is a learning effect that is unavoidable when information is presented poorly. When multiple sources of information are presented it leads to increased cognitive load. This effect refers to the principle that these information sources first should be integrated before being used. Research studies have been conducted to study learners' abilities to process information [13, 65]. In one study, the learners were better able to easily process and understand the information that was presented in a more integrated function, when compared to other methods using more separated forms to present the same information [13]. Cognitive load theory suggests that if memory load can be reduced, decision-making speed and accuracy can be increased.

Based on these theoretical justifications, the proposed benefit of edge bundling is that it supports the interpretation of the relationships among the edges, aids the knowledge manager's mental categorization efforts, and allows the analysis of clusters of edges rather than individual edges. By considering the clusters, the cognitive effort in categorizing the edges is reduced, as is the memory load associated with decision-making tasks. The final outcome is an expectation of more efficient and effective analytical reasoning, decision-making, and problem-solving.

2.6.2 Edge Bundling Algorithm

Although there are a number of different methods for implementing edge bundling, a popular approach is to use a spring layout algorithm [45]. Spring-based algorithms have been used for many years for dynamic graph drawing [22]. As such, it is a natural approach to use within edge bundling. Spring-based edge bundling divides each edge into a number of small segments. The position of each segment is compared to the positions of all segments in the other edges, and a virtual spring is attached

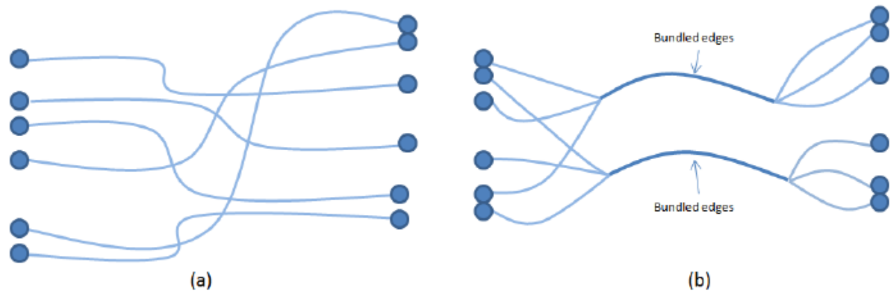


Figure 2.10: (a) Prior to the bundling process (b) After the bundling process

if the segments are sufficiently near one another. An iterative process minimizes all the forces acting on the springs by moving the connected edge segments closer to one another and away from each other. The end result is a grouping of edge segments and an associated bundling of the edges themselves. The visual effect is similar to the way electrical wires are merged together within a bundled cable, and separated (or frayed) at the end. A sample of the before and after results of the execution of this process is shown in Figure 2.10.

There are a number of factors that control the tradeoff between efficiency and effectiveness in the bundling process. Chief among these are the size of the segments, the distance threshold between segments that exert spring forces upon one another, and the number of iterations taken to minimize the spring forces.

If each edge is divided into a small number of large segments, the number of springs will be relatively small, and the spring forces can be minimized quickly. However, doing so can make the transition points between segments appear as sharp angles. Smaller segments give the impression of a curve rather than a series of connected line segments, making the path easier for the human eye to follow. Unfortunately, this results in a larger number of springs with consequent higher computational requirements. Thus, the tradeoff between the quality of the visual representation and the computational costs must be considered.

The distance threshold that dictates whether a virtual spring is or is not added between a pair of edge segments has an important outcome not only on the computational costs but also the final bundling outcome. If this threshold is made too small, the computational costs will be low (due to the small number of springs), and very little bundling will occur. If this threshold is made too large, the computational costs will be great, and a small number of very dense bundles will be generated. Carefully tuning this parameter in order to produce a sufficiently useful bundling at an acceptable computational cost is necessary.

Choosing the number of iterations taken to minimize the spring forces also requires consideration. Too few iterations and the bundles will not be well formed; too many and time is wasted on minor modifications of the segments with little visual improvement. Rather than controlling the precise number of iterations, it is more common to specify a threshold for changes between iterations that signals convergence, together with a limit for the maximum number of iterations.

2.6.2.1 Compatibility Measures

For graphs, the bundling process carried out using the spring layout algorithm usually results in high amounts of bundling (i.e., too many non-relevant edges join a bundle) [45]. Holten & van Wijk [45] described and implemented specific edge compatibility measures for complex graph structures, in order to counter the over-bundling problem. These compatibility measures use the different aspects of the edge bundling algorithm applicable mostly for general graphs [45].

To avoid bundling edges that are perpendicular to each other, the *angle compatibility* measure calculates the angle between two edges [45]. This value is maximum when the edges are parallel to each other, and is minimum if the edges are orthogonal to one another [45]. Using this compatibility measure, those edges that have an angle that approaches perpendicularity are discouraged from joining the bundle.

During the edge bundling process, it also possible that certain edges that are of different lengths can be bundled together. Including such edges in the bundle can distort the overall shape of the bundle. Therefore, the *scale compatibility* measure is developed to discourage such edges from joining a bundle based upon lengths of the edges in consideration for bundling [45]. Edges that are equal in length are allowed to be bundled and vice versa. Using this measure, the edges that are closer or equal in length to each other are favoured to be included in the same bundle.

The *visible compatibility* measure was introduced to filter out edges that are parallel, equal in length, and close together but still should not be bundled. The example of such edges is the opposite edges of parallelogram [45]. To calculate the visible compatibility value of the two edges, band of sight (i.e., a projection) is extended from one edge onto the extended line of other edge. The intersecting points of this band of sight are also calculated. So if these two edges are exactly parallel to each other the midpoint of the band of sight will exactly coincide with the midpoint of the second edge, which allows these edges to be bundled according to their visible compatibility. In other cases, this degree of compatibility depends upon how far (horizontally) these two edges are from each other. This measure is very effective in leaving those edges out from inclusion into bundles that are not visibly compatible.

In order to avoid bringing closer those edges that are far from each other in terms of distance the *position compatibility* measure was introduced. For any two candidate edges, the value of position compatibility is calculated by comparing the midpoints of these edges. This compatibility measure favours the edges when they are aligned [45], and the degree of this measure greatly decreases as the distance between the midpoints of the edges reaches infinity. This is the most important measure in all the compatibility measures as it ensures the basic condition of bundle compatibility of edges. Using this measure, those edges that are far from each other are discouraged from being bundled together, and a significant amount of computational processing time can be saved.

After all the values from the above different compatibility measures have been calculated, the next step is to multiply all the values, which results in a *bundling compatibility* value. This compatibility value can be used a global threshold value, which can be set to control the overall amount of bundling for the edges. Since different amounts of bundling can be obtained by varying this global threshold value, users are generally allowed to modify this value according to their requirements for the bundling process.

2.6.3 Benefits and Drawbacks of Edge Bundling

Within the context of ontology alignment, edge bundling can be considered a method for visually categorizing the mappings. That is, those edges that are grouped together may be considered a category due to their similar path between the two ontologies. Therefore, edge bundling can help in reducing the amount of visual clutter [69]. In addition to reducing visual clutter, it is expected that such an approach to representing the mappings will reduce the cognitive effort required to understand the mappings [78], which can lead to performance gains during the mapping validation and addition tasks.

While edge bundling can allow the structure of the edges within a graph to become more apparent, it does introduce a new problem. By clustering the edges within a bundle, ambiguity is introduced into the visual representation. That is, with edge bundling it is no longer possible to visually identify the precise source and destination of a given edge. Instead, all that can be done is to identify a number of possible candidate destination nodes for a given source node. Hence, whenever edge bundling is implemented in a setting that requires precise information to be extracted, interactive tools are required for the disambiguation of the bundles. The process of bundling temporarily hides some amount of information that is presented on the screen. However, the information is not removed and is available by using the interactive features developed.

Chapter 3

Edge Bundling within an Ontology Alignment Interface

3.1 Motivation

As discussed in the previous chapter, ontology alignment tools currently face problems in providing effective visualization of ontology mappings when the mappings are large in number. The source and target ontologies are normally presented as trees while the mappings between their entities are shown as lines connecting the concepts in the given ontologies [48]. Due to the limited available display space to show these mapping lines, a large number of edge crossings is inevitable. This situation leads to a significant degree of visual clutter that decreases the usability of the ontology alignment interfaces.

Since the ontology alignment tasks (i.e., mapping validation and addition) are already time-consuming and require a great amount of patience and effort, the presence of visual clutter due to mapping edge crossings further increases the difficulties for the knowledge manager in performing these tasks. A clear and organized view of the mappings will help in reducing the cognitive load and frustration experienced by the knowledge manager. Therefore, the main aspect of this research is to improve the representation and organization of mappings for an existing state-of-the-art ontology alignment system interface, with the goal of increasing the productivity, effectiveness, and satisfaction of knowledge managers.

To accomplish this, a promising approach from the domain of graph visualization referred to as edge bundling has been implemented within an existing ontology alignment system (CogZ [31]), with interactive support for conducting alignment tasks. In large graphs, the edge bundling is designed to reduce the amount of visual clutter by combining the common pathways of edges that are near one another; the goal within ontology alignments is the same. The extended system has been named as CogZ-e (or CogZ-extended).

The details of the visual and interactive features of CogZ-e will be described in this chapter, along with implementation details. A comprehensive example is also provided to illustrate the potential benefits of the system. The chapter will conclude with a discussion on the potential advantages, and drawbacks of using bundling in the domain of ontology mapping visualization.

3.2 Approach

3.2.1 Framework and Constraints of Mapping Representations

The prototype system that was developed to implement edge bundling is based on the open and extendable framework provided by CogZ [31]. This existing framework provides the core features required by an ontology alignment interface, including integration with existing state-of-the-art ontology alignment algorithms [62], representation of the pair of ontologies as zoomable and scrollable trees on opposite sides of the interface, rendering of the mappings as curves within the middle region of the interface, and basic interaction mechanisms that support mapping validation and mapping addition tasks. The extension integrates edge bundling into the process, and introduces new interaction mechanisms to support the disambiguation of edges within a bundle.

The edge bundling was designed specifically for graphs, however, it is now intended to be used for mapping edges. This new scenario is conceptually dissimilar to graphs. Therefore, the constraints of the mapping edge representations are required to be studied and analysed before utilizing the edge bundling for mapping edges.

As previously discussed, within an ontology alignment interface, the two ontologies to be matched are represented as vertical hierarchies, with the space between the two hierarchies used to show the mappings. Since the ontology organization is critical to understanding its structure, the nodes cannot be moved freely in order to facilitate bundling. As such the edge bundling is constrained to only manipulate the edges themselves, and not the source or destination nodes within the ontologies themselves.

In addition, when representing a mapping, it is possible that one concept from the upper part of the source ontology is mapped to the lower part of the target ontology. Due to the limited availability of visual space, these ontologies may be contained within scrollable panes. As such, it may be the case that one of these concepts are visible and the other is scrolled out of view, or neither of them are visible. Representing such a mapping is problematic. In addition, it would require significant amount of memory and computing resources. Applying edge bundling on all the edges including those that have off-screen sources and destinations may result in over-bundling, with many of the bundles connecting concepts that are not in the field of view. Therefore, the mapping edges whose source and target concepts are not visible (i.e., scrolled out of view or collapsed) are filtered from this set in order to ensure that the bundling is only performed on completely visible edges.

For the bundling approach to work effectively, the ontologies themselves need to be ordered and organized in some meaningful way. Typically, ontologies are ordered alphabetically based on the root nodes, however, other interesting organizations of the ontologies are possible. For example, they may be ordered based on ascending or descending order of the number of sub-classes that are present in a root node of sub-trees. Since examining how different organizations of ontologies could affect the results of the bundling algorithm is beyond the scope of this research, the default ontology organization that is provided by the baseline system was used without any alterations. The main view of the baseline system CogZ along with the default layout of sample source and target ontologies, is shown in Figure 3.1.

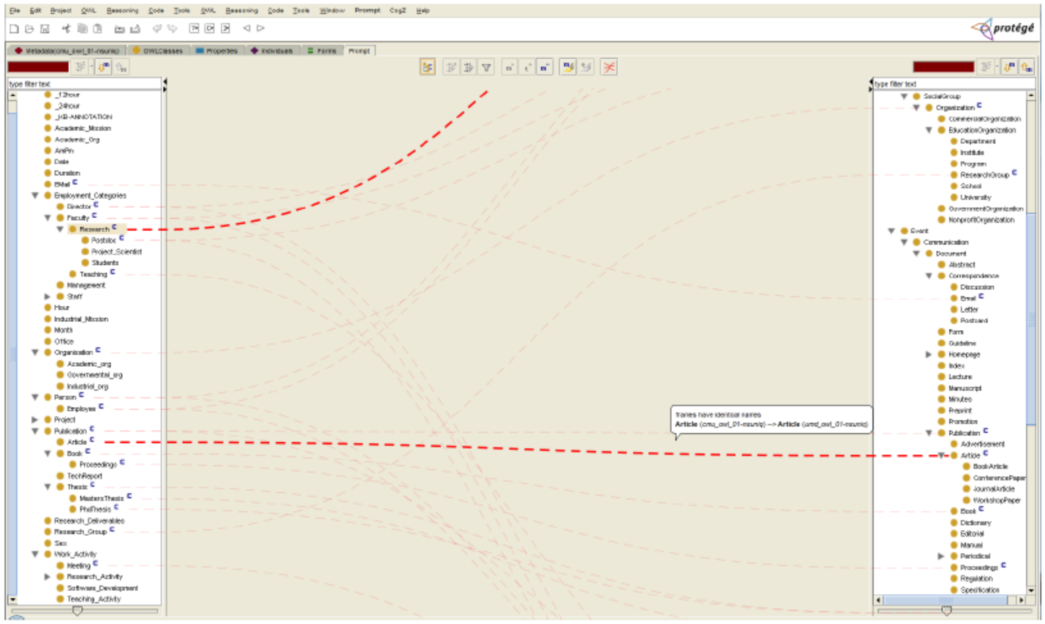


Figure 3.1: The main view of the prototype without bundling.

3.2.2 Edge Bundling Process

Following the edge bundling approach outlined in Section 2.6, the first step in the process is to extract the edges from the ontology alignment interface framework. This information includes the source and destination points (entities in the ontologies), the edge path, and the status of the mapping represented by the edge.

The next step of the process is the segmentation of each edge in the mapping set. As previously discussed in Section 2.6.2, there is a tradeoff between the number of the segments and the computational costs when performing edge bundling. While 1-pixel segments would provide the smoothest curves, doing so would result in an unsatisfactory performance. Rather than statically defining this parameter, the approach from Holten and van Wijk's work on edge bundling in complex graph structures [45] was followed, and the number of segments (and therefore the size of the segments) were dynamically adjusted. By starting with a small number of edge segments, course edge bundling can be done efficiently. Iteratively increasing the number of edge segments then allows finer and finer levels of bundling to be achieved.

In this case, the process was started with four segments for each edge. However, these segments are not defined over the entire length of the edge. Since it is important to be able to clearly identify the source and destination entities within the ontologies, it is not desired that the bundles to be formed are too close to the actual ontologies. As such, the segmented region of the edges is restricted to the middle 75% of the length of the edges.

Holten & van Wijk [45] describe a number of different measures for determining whether a pair of edge segments should be bundled (as discussed in Section 2.6.2.1). These measures take into account the complex nature of general graph structures, including the position of the edge segments, the lengths of the entire edges, the edge direction, and the angle of edge crossings. Given the restricted nature of the edges within an ontology alignment interface, where only similar-length undirected mapping edges are used, these measures do not produce a significant effect on the bundling results. This was also confirmed through preliminary experimentation, that many of these measures do not produce noticeable effects on the edge bundles. However, since the position compatibility measure directly works on the position of the edges, only this measure is still applicable to the given scenario, and plays a major role in the formation of edge bundles. As such, the developed approach only uses the position of the edge segments when determining the potential for bundling.

For each edge segment, the distance from its midpoint to the midpoint of all other segments in all other edges is calculated. If the distance between a pair of segments is below a distance threshold, a virtual spring is attached between these segments. This distance threshold is based on a ratio of the average length of edge segments to the average length plus the distance between their midpoints, such that longer edge segments are brought closer to more distant edge segments. In order to force non-connected and more distant edge segments further away from one another, virtual electrostatic forces are also modeled within the system. These forces avoid the situation of over-bundling, whereby all edges are pulled together because there are no forces pushing them apart.

The next step in the process is to attempt to minimize the forces exerted upon the edge segments due to the springs and electromagnetic forces. As edges are moved closer to one another, the spring forces will be reduced, but the electromagnetic forces will be increased. The sum of all forces being exerted on each edge segment is considered, and its location is adjusted in the direction of the force by a distance that is relative to the magnitude of the force. Since the movement of one edge segment in this manner may cause the forces exerted on other edges to change, the process of minimizing the forces within the system must be done over multiple iterations. A simple approach of statically limiting the number of iterations to 120 was followed.

At this point, a coarse-level of bundling will have been achieved based on only four segments for each edge. In order to smooth out the bundling, this whole process is repeated over 12 additional cycles, each time increasing the number of segments by one (and therefore reducing the length of each edge segment). This choice of the number of additional cycles (and therefore the number of additional segments) was determined through experimentation. The goal is to find a balance between time required to perform the bundling calculations and smooth bundle formation. When increasing the number of segments, the divisible points that are used are chosen at the location on the straight-line edge segments from the previous cycle. This incremental process is more efficient and effective than performing the bundling at a fine level of detail straight-away [45].

The final outcome is edge bundling performed over 16 segments on each edge. Since drawing the edges within the display using only these segments would not be graphically appealing, instead a curved line is fitted to the center points of each edge segment. Figures 3.2 and 3.3 provide before and after examples of the outcome of edge bundling.

3.2.3 Bundle Ambiguity and Disambiguation

Although the edge bundling produces a less-cluttered visual representation of the mappings, it has the side-effect of introducing ambiguity among the edges within the bundles. That is, for a mapping that starts at a given entity, if its edge is contained within a bundle, it is not possible to determine the exact ending entity from among those that are contained in the same bundle. This causes a problem during the mapping validation tasks since knowledge managers must examine and interact with mappings individually.

The solution to this problem is to provide interactive tools that support disambiguation. Since tooltips and single clicks on edges are already interactive aspects of the CogZ framework, the interaction for disambiguation must use an alternate mechanism. An interactive method is required to quickly view mappings that are present within a bundle along with options for selecting an individual or multiple

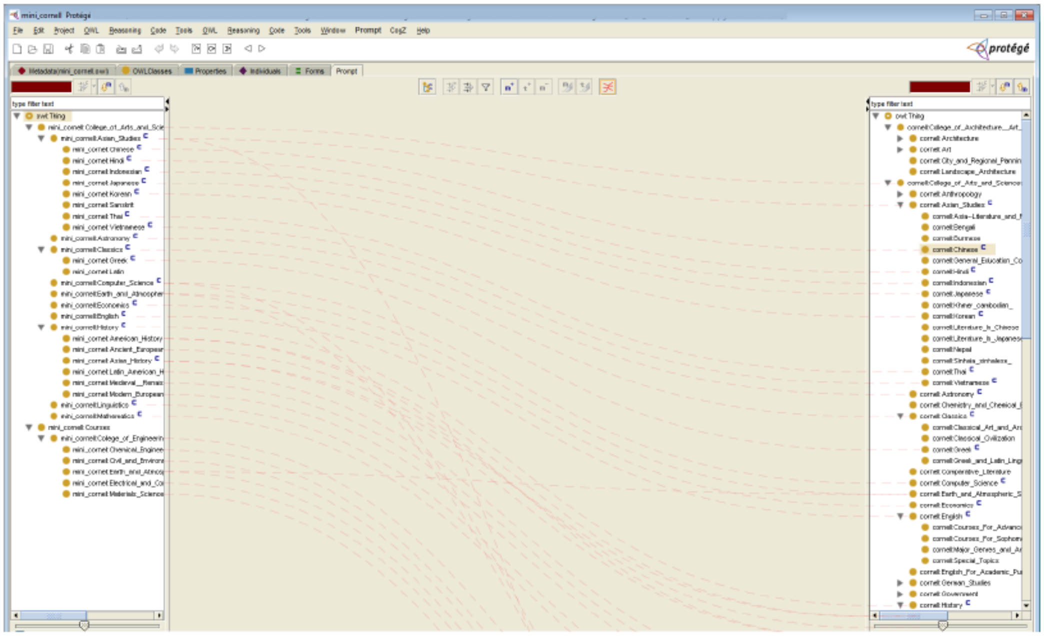


Figure 3.2: View of the baseline system with no bundling. The red dashed curves represent the candidate mappings generated by the system.

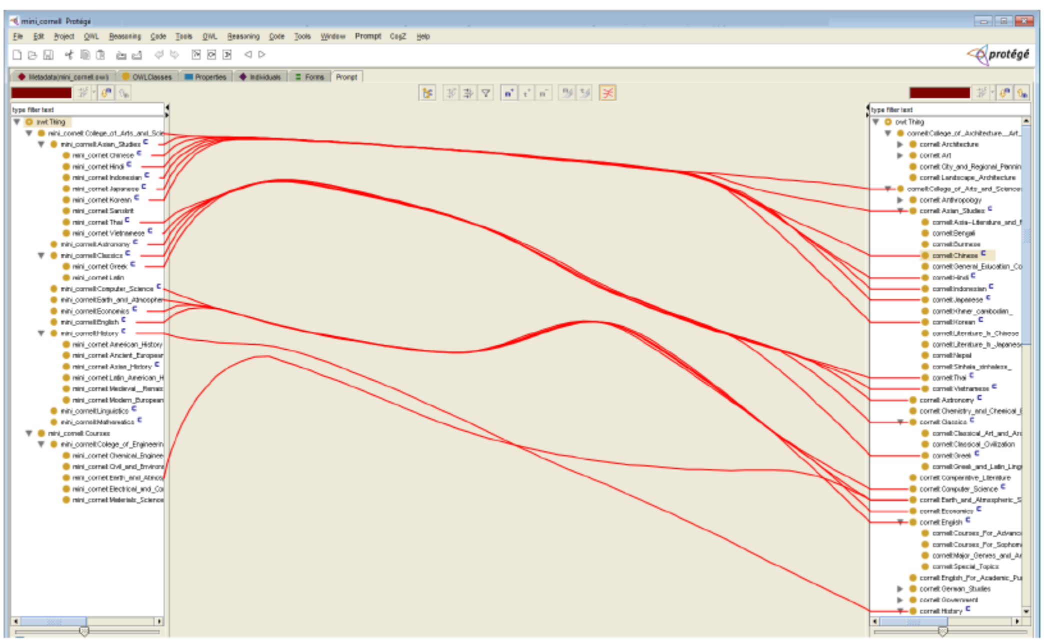


Figure 3.3: Application of edge bundling to the mapping edges. Note that mappings whose source and destination are not visible have been filtered.

mappings. Furthermore, differentiation of mappings based on their state (e.g., candidate or verified) needs to be presented accurately and distinctly on the screen. More details about the provided disambiguation options and process with examples will be provided in the sections that follow.

3.2.4 Support for Ontology Alignment Tasks

Since the mapping validation and mapping addition tasks are already very time-consuming and require a great amount of cognitive effort, performing these tasks while navigating and interpreting a complex and ill-structured visual interface can make them even more challenging. Edge bundling offers the promise of presenting a less complex and more structured view of the mappings, which can help in reducing the cognitive load of knowledge managers.

3.2.4.1 Mapping Validation

In the mapping validation task, the knowledge managers confirm or reject the candidate mappings produced by the ontology alignment algorithm by using their knowledge about the source and target ontologies. In the baseline system (CogZ), the knowledge manager has to select each mapping edge and click on the desired operation button (i.e., remove or confirm the mapping) for each selected edge. To complete the validation task, every mapping presented on the interface needs to examined by the knowledge manager.

Edge bundling helps by dividing the larger task into multiple smaller sub-tasks of validating the mappings within a given bundle. Reviewing the mappings within a bundle provides the knowledge managers with relevant context about the set of mappings as a group, and allows the other bundles to be ignored. Thus, tasks such as validating whether the set of mappings is internally consistent can be done by considering only the edges within the bundle, making efficient use of the knowledge managers' cognitive memory. Because of the focus that the bundling provides to the task, the degree of searching and swapping of information in and out of cognitive memory will be greatly reduced in comparison to an unstructured examination of the candidate mappings.

The details for the corresponding mappings that are encapsulated within the bundle can be seen when knowledge managers hover the mouse pointer over the bundle via an extended tooltip. This tooltip presents all the mappings that are contained in the bundle. The knowledge manager can simply press and keep the Alt key pressed on the keyboard to see the tooltip listing all the mappings that are present in the bundle (see Figure 3.4). Since the mappings in the bundle can be viewed individually using this tooltip, the mapping information can be disambiguated. The important information that could be helpful for the tasks at hand such as the number of mappings that are present in the bundle, and the presumed relatedness of the mappings due to presence in the bundle, can be easily processed and used through this tooltip.

Since edge bundling provides knowledge managers with the ability to work on a bundle of mappings rather than verifying them one by one, the knowledge manager must be able to perform the appropriate validation operations (confirm or reject) on the mappings of the selected bundle. Therefore, the ability of performing multiple validation operations simultaneously is required.

To meet this requirement, a pop-up window showing further details about the mappings contained in the selected bundle with the validation operations is provided.

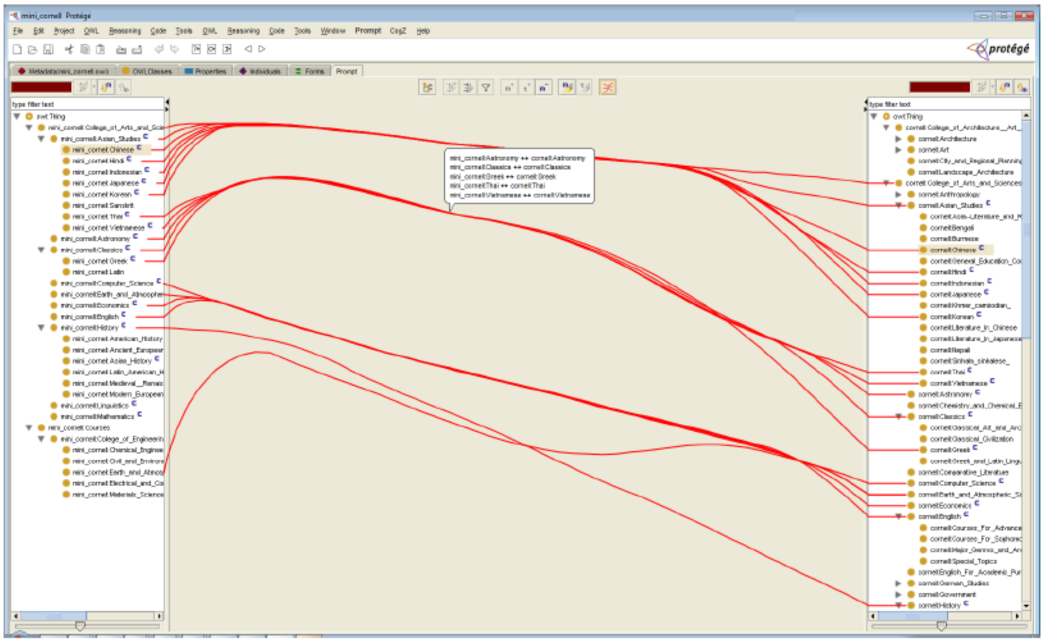


Figure 3.4: The tooltip showing the information about the mappings that are present in the bundle.

This window appears when the knowledge manager hovers the mouse pointer over a bundle and clicks on the bundle with the Alt key pressed. The window provides the same list of the bundle's mappings that were present in the tooltip with the ability to select multiple mappings, and buttons for validation operations such as confirm or reject the selected mappings. The pop-up window is shown in Figure 3.5.

Since the pop-up window appears in front of the mappings, the knowledge managers may lose the sense of the location of the mapping that they are currently working on. This issue was addressed by the development of a mapping highlight feature, which changes the colors of the mappings listed in the window depending upon their current state. When the knowledge manager selects a candidate mapping on the pop-up window, the color of the selected mapping edge is changed to black from

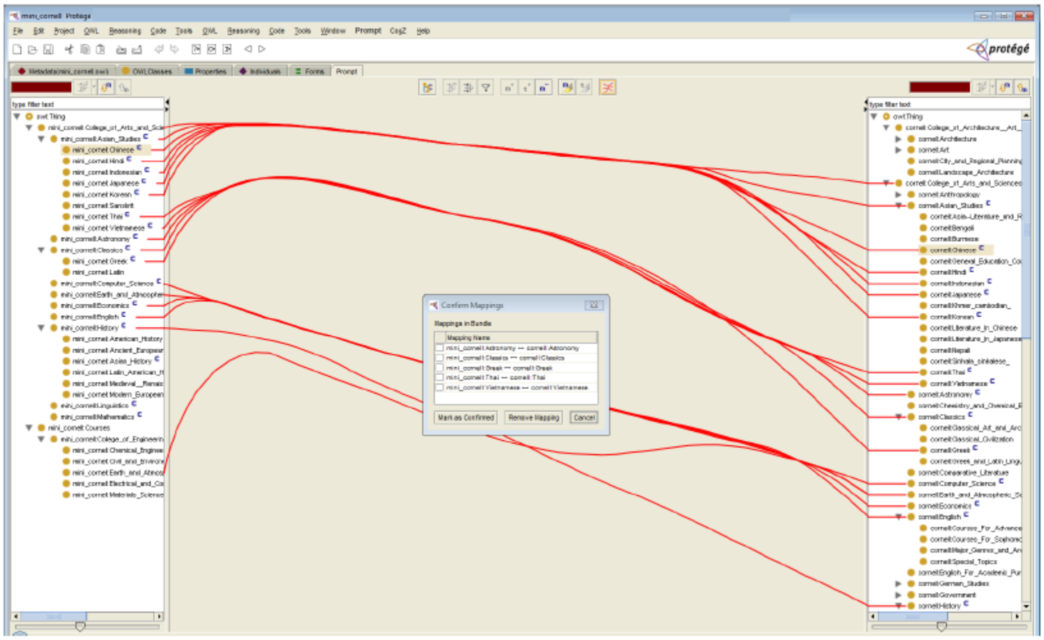


Figure 3.5: The pop-up window providing the mapping validation operations.

red. The change in the color of the mapping edge quickly grabs the knowledge manager's attention, allowing them to easily accommodate the information concerning the mapping location, while making a validation decision. The effect of this feature is illustrated in Figure 3.6.

When the knowledge manager confirms some mappings and their state is changed from candidate to confirmed, such mappings need to be shown visually different from the rest of the mappings in the bundles, as well as when the pop-up window or tooltip is visible. Whenever the knowledge manager confirms a candidate mapping inside a bundle their colour is changed to blue (as shown in Figure 3.7). The blue colour clearly distinguishes the confirmed mapping(s) from the rest which are in red colour on the light beige coloured background of the interface.

The mapping-highlight feature developed for the knowledge manager's assistance

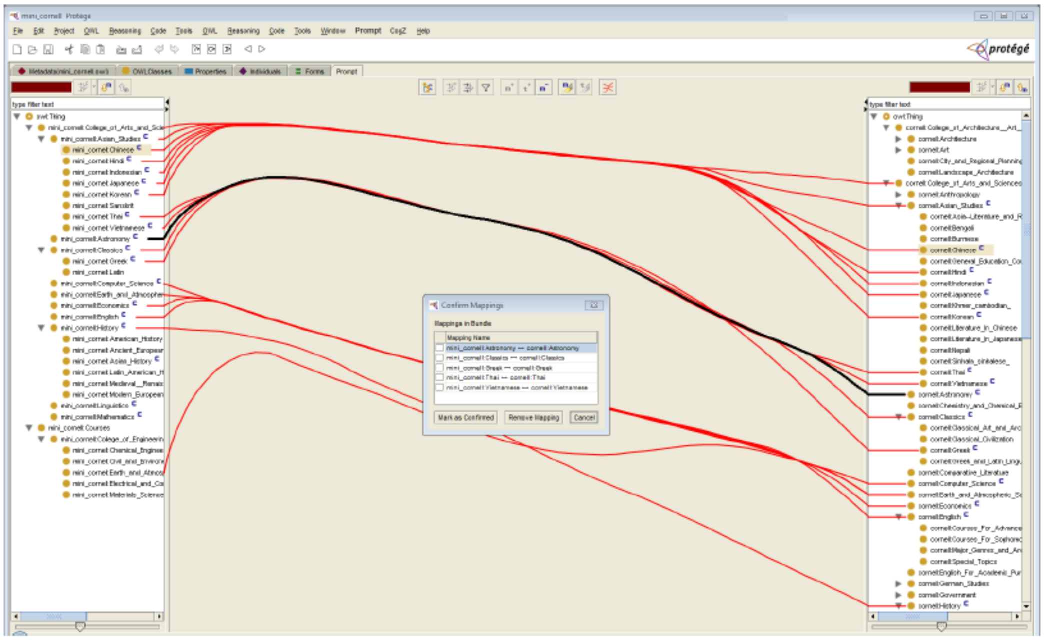


Figure 3.6: As per the knowledge manager's selection, the "mini-cornell:Astronomy" to "cornell:Astronomy" mapping's colour has been changed to black.

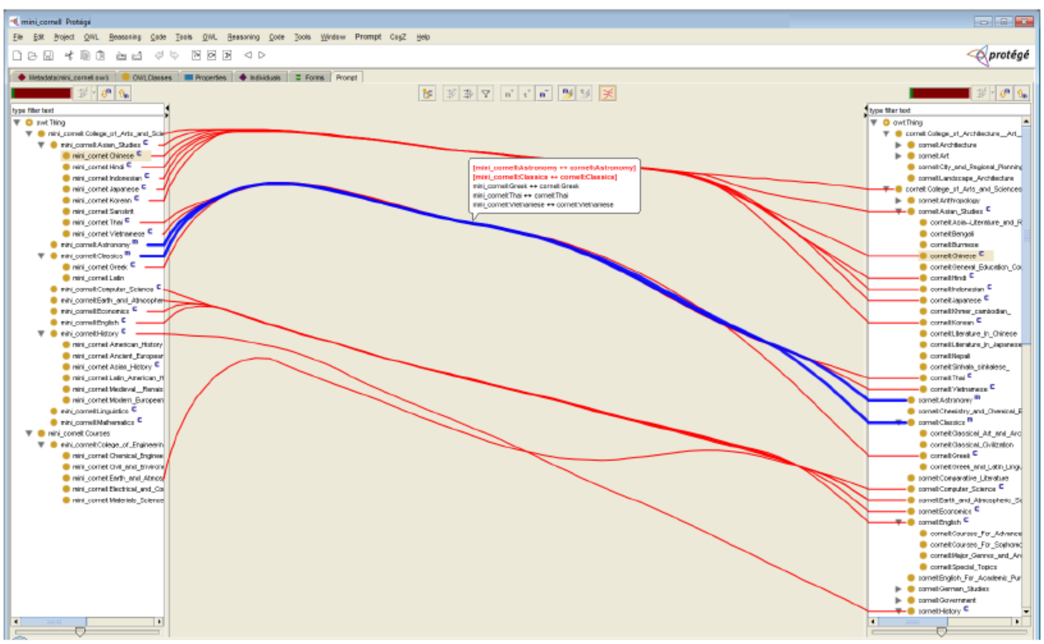


Figure 3.7: The tooltip shows the confirmed mappings in red colour with bold font.

was further extended to reflect the state of the mapping within the pop-up window as well. When the pop-up window for a bundle is visible that contains confirmed mappings, these mappings are listed in red colour with bold font instead of regular font and black colour. When the knowledge manager selects a confirmed mapping in the pop-up window, the mapping highlight feature also temporarily changes the colour of confirmed mappings from blue to green so that the knowledge manager can clearly see the confirmed mapping. This change in colour is important specifically when the number of confirmed mappings are more than one, and it is not easy to distinguish them even though they are already shown separately from the candidate mappings with a different colour (i.e. in blue colour). The implementation of this feature can be seen in Figure 3.8. A finite state diagram of mapping colour changes depending upon the current mapping state is shown in Figure 3.9 along with Table 3.1 listing the color encodings of the mappings depending on their state and different conditions.

The tooltip to view the mappings inside the bundle was also modified to present the details about the confirmed mappings (if present) differently from the candidate mappings. The modified tooltip functionality shows the details about the confirmed mapping in red colour instead of the black. The font used to describe the details of the confirmed mapping was also made bold to make it further distinguishable from the texts of candidate mappings that are also listed in the same tooltip, and for which regular font is used (see Figure 3.7).

3.2.4.2 Support for Mapping Addition

It is possible that the alignment algorithm may miss a mapping between two concepts. Therefore, the second step of the ontology alignment process is the mapping

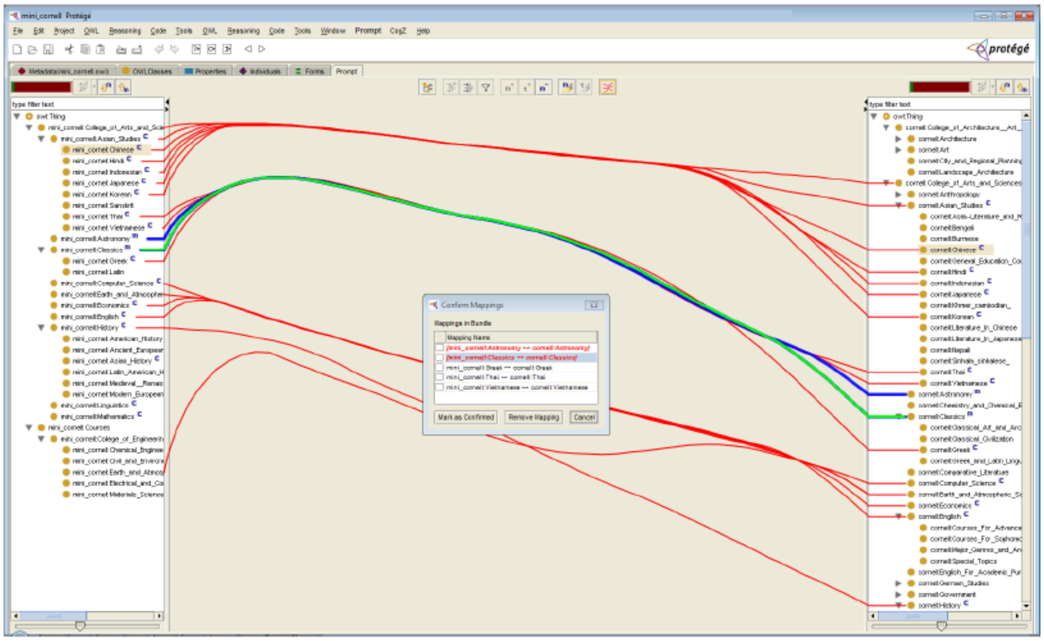


Figure 3.8: The confirmed mappings listed in the pop-up window are also presented using red colour with bold font following the scheme adopted in the tooltip, and when a confirmed mapping is selected the colour is changed to green from blue.

addition task. In this task, the knowledge manager has to identify such unmapped, but related concept pairs in the source and target ontologies and then manually create a mapping. This process typically requires visual scanning of the entire source and target ontologies.

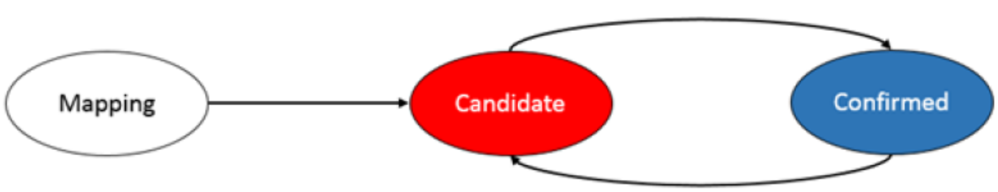


Figure 3.9: The colour transition of the mapping edge depending upon the current state.

Table 3.1: Summary of colour encodings for the mapping depending upon the state and location.

Mapping color encodings		
Within the interface		
Mapping State	Edge color	When highlighted
Candidate	Red	Black
Confirmed	Blue	Green
Within the extended tooltip		
Mapping State	Text color	Font attribute
Candidate	Black	Regular
Confirmed	Red	Bold
Within the pop-up window		
Mapping State	Text color	Font attribute
Candidate	Black	Regular
Confirmed	Red	Bold

The benefits of edge bundling may also be realized in the context of the mapping addition tasks. By clustering related mappings, the overall organization of how the mappings relate to the structure of the ontologies may be more easily perceived. When seeking potentially related, but as yet unmapped entities, the structure imposed by the bundles provides the knowledge managers with a relatively small space in which to search. Rather than searching the entire ontology for a corresponding entity, the knowledge managers can visually focus on areas around the existing bundles, thus reducing the cognitive load associated with this task.

Scanning for concepts that should be mapped in nearby areas improves on scanning concepts from top to bottom in both ontologies. For example, if the knowledge manager is looking to find a match for an unmatched concept in the source ontology, it is better to look for a potential match by following the nearby mappings of this unmatched concept around this concept. The potential match for the concept might be right below the target concept. This process to finding potential matches is more efficient and effective than the existing system, however, the underlying assumption here is that the potential match might be in the vicinity of existing mappings, as organized by the hierarchical structures of the ontologies. If this is the case, then the bundling will help to focus the search effort. A more specific example is provided in Section 3.4.

3.3 Implementation Details

3.3.1 Platform

The edge bundling prototype is integrated into the CogZ system, which is an open source system and a standard Java desktop application built on the Java Development Kit 1.6. The programming of the edge bundling process follows the implementation of the original algorithm in the JFlowMap [5] system for bundling graph edges.

3.3.2 System Architecture

As the prototype CogZ-e is built on the existing CogZ system, the architecture shown in Figure 3.10 includes the architecture of CogZ as well as the new modules that were developed. The modules shown in blue already exist in the baseline system, while the new developed modules are shown in maroon. The work flow as well as the interaction between the modules will be described here.

The system architecture of CogZ-e consists of four important modules. Protégé's [60] *ontology import* module is responsible for importing the current format of source ontology and creating data structures to facilitate the representation and manipulation of the ontology. PROMPT's [62] *alignment generator and target ontology importer* modules are incorporated inside Protégé as plugins, which carry out the tasks of importing the target ontology and calculating the alignments between the entities

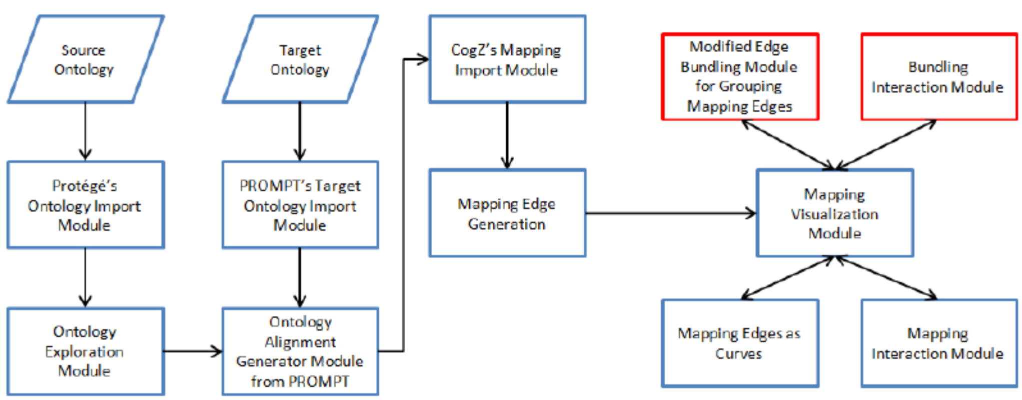


Figure 3.10: System Architecture of the prototype CogZ-e. The information flow from different modules has been marked by black arrows. The exchange of information is represented using red arrows. The new modules that have been developed are marked in maroon.

of the source and target ontologies. CogZ's [29] *mapping import* module takes alignments produced by the alignment generator module and generates graphical edges to present them on the screen. Finally, the *mapping visualization* module of CogZ portrays the mapping edges on the screen and provides the interaction options. The new modules *modified edge bundling* and *bundling interaction* that perform the bundling process on the mapping edges and provides the interaction support have been built inside the *mapping visualization* module.

3.3.3 Work Flow

The process starts when the knowledge manager has a set of two ontologies that need to be aligned to each other. The set typically consists of one source ontology and one target ontology. The first layer of the system is based on Protégé [60], which is mainly used to create, visualize and edit ontologies. Therefore, to start the process the knowledge manager provides the source ontology to the system. The source ontology is loaded into the system by the import mechanism of Protégé. The ontology-exploration module of Protégé can also be used to view the source ontology. The managers can view the source ontology using different plugins available in Protégé at this point, however, for the alignment task this step is generally skipped at this stage.

The next step is to initialize the PROMPT [62] plugin on the source ontology that leads to the target ontology import module of PROMPT. The ontology alignment generator module of PROMPT takes the target ontology as input, and calculates the mapping set comprised of candidate mappings between the concepts of the given source and target ontologies.

After the computation of the candidate mapping set, the control is then passed to the mapping import module of CogZ. This module is responsible for generating a mapping edge for every mapping present in the candidate mapping set. This set is provided as an input to the mapping visualization component of CogZ, to draw the mappings as Bezier curved lines. The mapping visualization component of CogZ also creates a mapping information tooltip for every mapping curve, which is visible whenever the mouse pointer hovers over the edge. The mapping visualization component of CogZ also presents the source and target ontologies on the opposite sides of the screen using tree-based representations and the generated mappings are shown within the space between the two ontology trees.

The mapping visualization module of CogZ is extended to include the new developed modules such as the modified edge bundling and bundling interaction modules. These modules are responsible for the bundling process and provide interaction options for bundles. When the bundling process is initiated, the mapping visualization component passes all the information pertaining to currently visible mappings to the bundling module. The bundling module processes all the provided mapping information and calculates the bundling formation. The bundle interaction module becomes automatically activated after the bundling process. When the knowledge manager confirms or rejects a mapping, the concerned change is passed on to the mapping visualization module that updates the state of the mapping in the mapping set. Similarly, when a new mapping is added, the new information is appended to the mapping set by the mapping visualization module.

3.4 Example

An example is provided here to illustrate the value and benefits that edge bundling can provide to a knowledge manager dealing with mapping validation and addition tasks. This example reflects how a knowledge manager can use the bundling capability of CogZ-e in order to see an organized view of mappings and perform the validation and addition tasks. The sample source and target ontologies used for this example describe the domain of academic conferences, and are available to be used freely [26]. The complete ontologies are listed in Appendix B.

As described in Section 3.3.3, the ontology alignment tasks start after both the source and target ontologies have been loaded and the mapping set has been generated. After the mappings are bundled and presented on the screen, the knowledge manager can start the first step of the alignment task, which is the mapping validation task. In this task, the knowledge manager starts confirming or rejecting the candidate mappings based on their knowledge about the ontologies, and the level of accuracy of the mappings calculated by the alignment algorithm.

After the knowledge manager has clicked the "Apply edge bundling" button, the bundling algorithm is initiated and takes the current mapping set as input. The algorithm produces bundles for the current visible mapping set, the results of which can be seen in Figure 3.11. It can be easily seen that instead of dealing with a random order of mapping edges before bundling, the knowledge manager now has to deal with roughly two bundles.

Since bundles provide a logical set of milestones or a sense of the number of steps towards finishing the mapping validation task, the knowledge manager may quickly

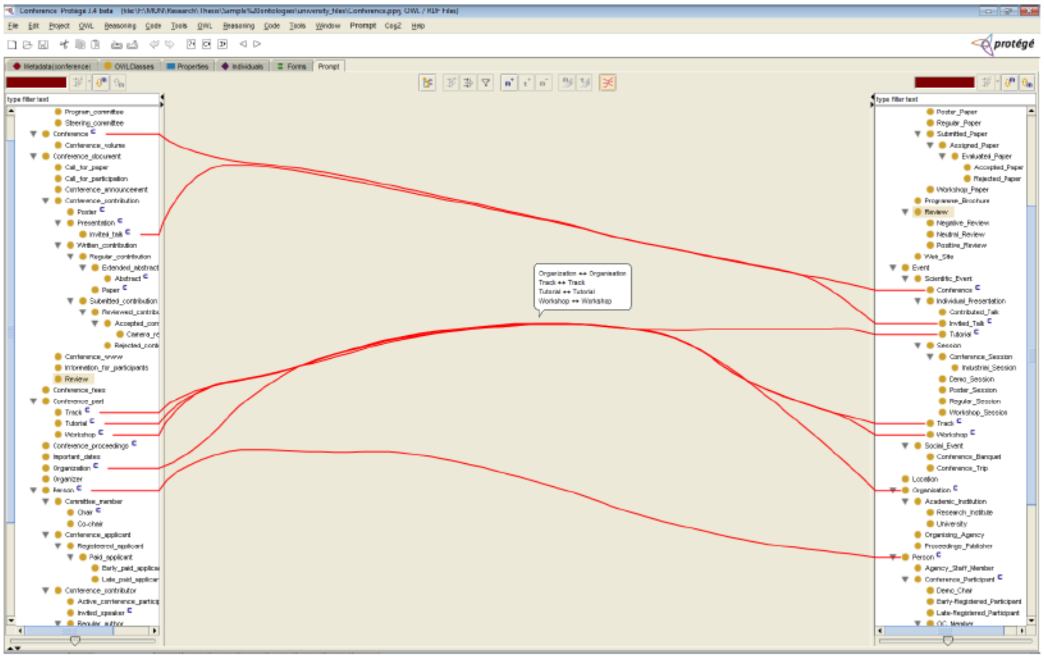


Figure 3.11: Extended tooltip listing all the mappings contained in the bundle.

identify different starting points and pick any bundle as a starting point for the validation task. This feature makes the validation tasks more manageable, allowing them to be completed in an organized manner. After the knowledge manager selects a bundle as a starting point and decides to work on it, the next step would be to see what mappings are contained in the bundle. Since the mapping edges are grouped because of the bundling, this information can be disambiguated by keeping the Alt key pressed, while the knowledge manager hovers over the bundle. The resulting tooltip allows the knowledge manager to see that there are four mapping edges merged into the bundle (as seen in Figure 3.11). By scanning the mapping list, the knowledge manager now decides to validate the mappings present in the bundle.

While keeping the Alt key pressed and with the tooltip being visible, the knowledge manager may click within the tooltip in order to initialize the pop-up window, which lists all the mappings shown in the tooltip. The pop-up window also allows the knowledge manager to view the mappings individually as well as perform single or multiple validation operations. The knowledge manager has found the "Organisation to Organisation" mapping to be correct. The knowledge manager can highlight the mapping (making the colour of edge representing the mapping to turn to black), and select the mapping to be marked as confirmed (as seen in Figure 3.12).

After pressing the "mark as confirmed" button, the pop-up window is closed, the state of the mapping "Organisation to Organisation" gets changed to confirmed from candidate, and the mapping edge colour has been changed to blue, which makes it clearly distinguishable from other mappings. This change is also reflected in the tooltip showing the mapping list of the same bundle but the text of the mapping is in bold font with red colour (as in Figure 3.13). This change in visual cues (i.e., different

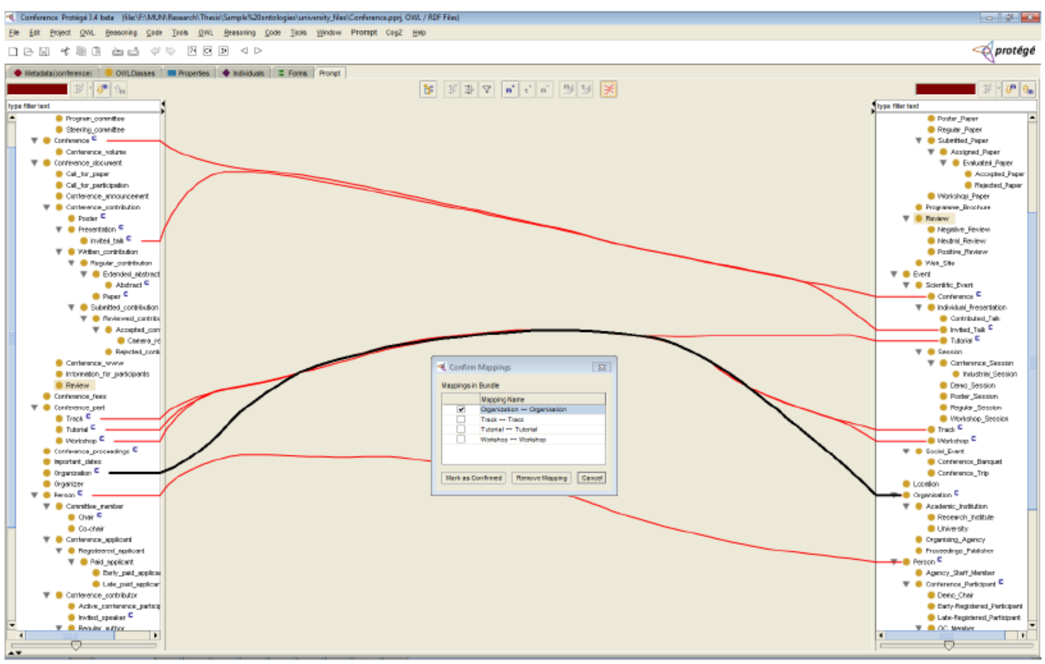


Figure 3.12: Pop-up window listing the mappings that were visible in the tooltip.

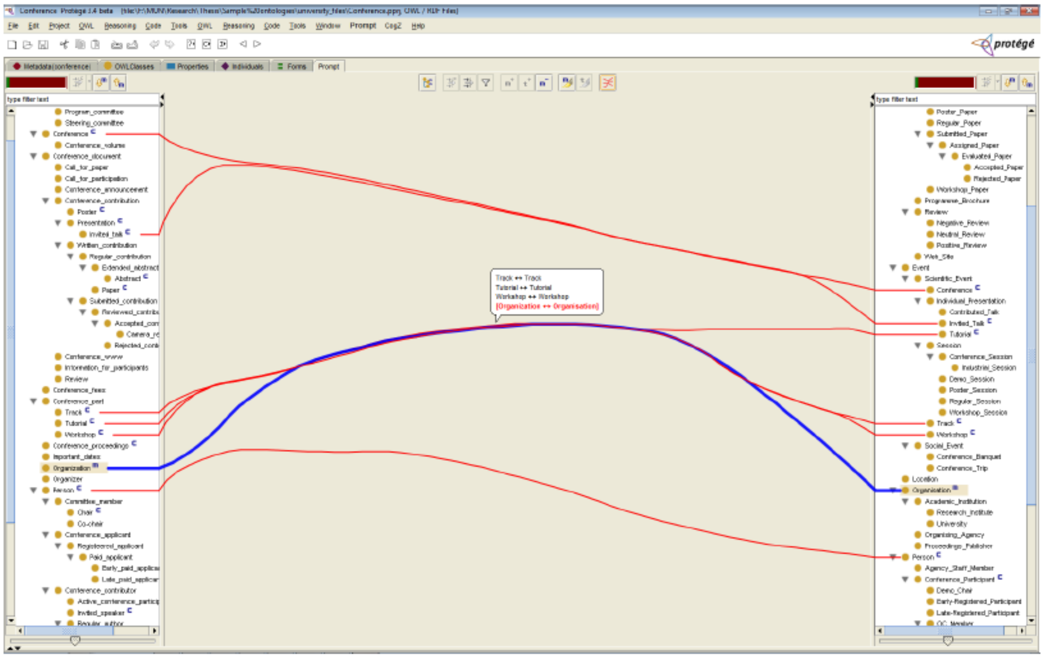


Figure 3.13: The confirmed mapping "Organisation to Organisation" is shown in blue colour with the text in the tooltip in red colour with bold font.

colour for mapping edge and text in the tooltip) to represent the confirmed mapping allows the knowledge manager to easily view the difference.

At this point, the knowledge manager decides to verify the rest of the mappings present in the bundle, therefore the pop-up window has been opened again. The difference of already confirmed mappings can also be seen in the pop-up window as those mappings are pushed down to the bottom of the list with their text colour changed to red and the font made bold (see Figure 3.14). All the remaining three mappings have now been selected and confirmed by the knowledge manager. Therefore, all the mappings of the initially selected bundle have been confirmed and the knowledge manager has finished working on that bundle (see Figure 3.15). The knowledge manager may now continue working on other bundles and validating mappings that are

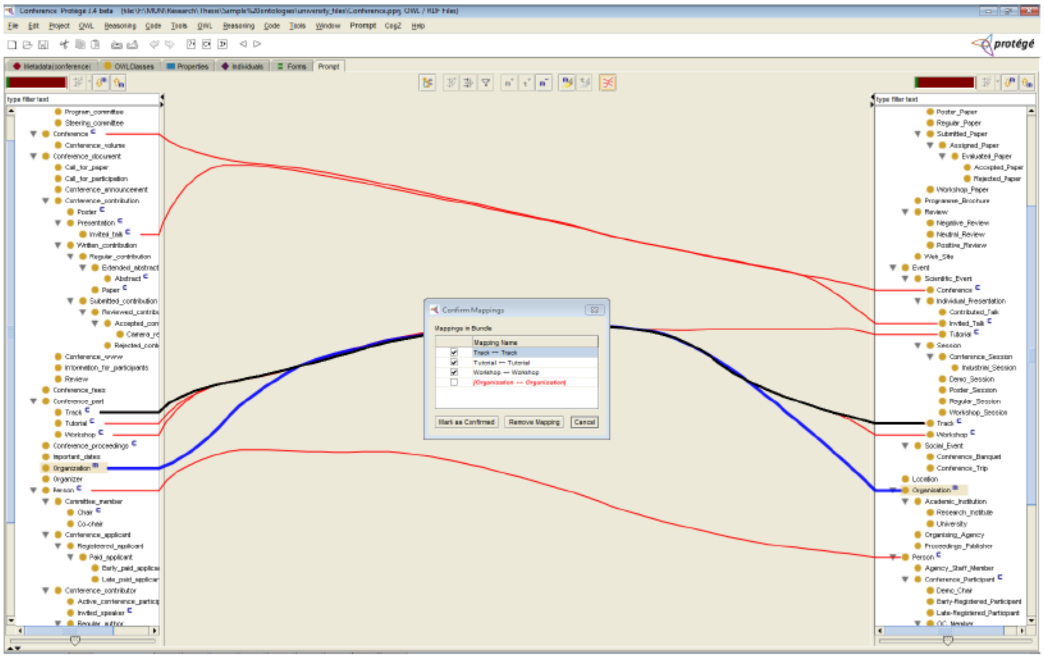


Figure 3.14: The confirmed mapping appearing at the bottom in red colour with bold text in the pop-up window along with the remaining candidate mappings.

present within these bundles.

It is quite possible that the alignment algorithm may miss matching some concepts that are relevant to each other. Therefore, the second and final step of the alignment task is the mapping addition task, in which the knowledge manager's task is to find any two concepts between the source and target ontologies that are relevant but no mapping exists between them, and create a mapping between the concepts.

Bundling allows the knowledge manager to focus on the areas within the source and target ontologies that are related. These areas (if present) are automatically highlighted by the bundles as one can easily see a group of mappings originating from source and ending at the target ontology. For adding a potential mapping, these areas can be helpful as there is a higher chance of finding new mappings in the nearby

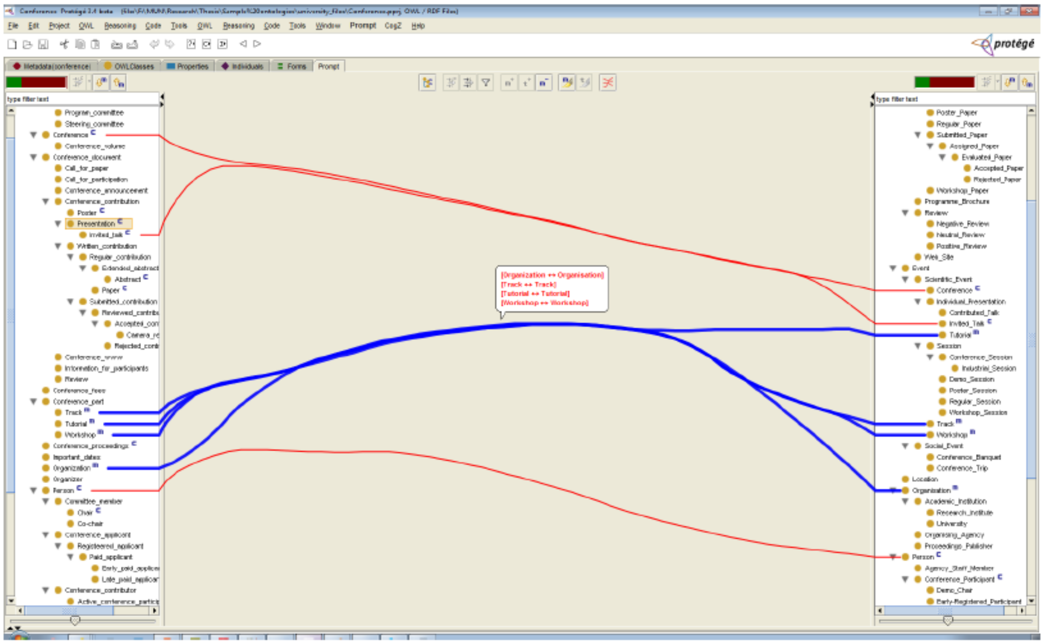


Figure 3.15: The snapshot of the prototype after the knowledge manager has confirmed all the mappings of the initial bundle.

region of these areas. It is still a better strategy for the knowledge manager to look for potential concepts (that should be mapped to each other) in nearby areas rather than scanning concepts from top to bottom in both ontologies in order to find anything useful. For example, if the knowledge manager is looking to find a match for the target ontology's concept named "Individual_Presentation" from the source ontology, it is better to look for a potential match by following the nearby mapping of the concept "Invited_Talk" (in the target ontology). The knowledge manager will find that the concept "Presentation" in the source ontology that is just above the "Invited_talk" concept that can be a potential match to the "Individual_Presentation" in the target ontology. Hence, this process is more efficient and effective than the existing system, which has no option but scanning through both ontologies to match the remaining concepts. The potential match that can be found by the knowledge manager is shown in Figure 3.16.

3.5 Discussion

Generally, when two real-life related ontologies are aligned, the process results in a large number of mappings, which, when portrayed as-is on an interface, causes visual clutter. As noted in Section 2.4, one of the main challenges of ontology alignment interfaces is to overcome the complexities of presenting the mappings in the limited space available on the visual screen. In this chapter, an approach for grouping and visually categorizing ontology mapping edges was described. This approach helps

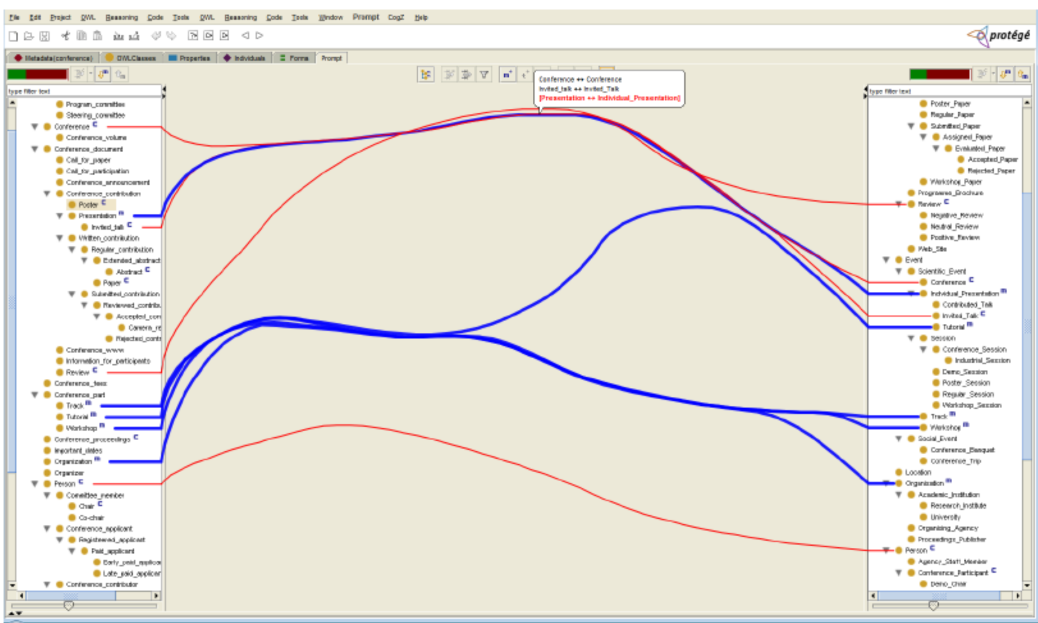


Figure 3.16: The newly created mapping between the source "Presentation" concept and the target "Individual_Presentation" concept is shown by the top blue line as well as in the tooltip.

in reducing visual clutter produced by the mapping edge crossings by incorporating an edge bundling algorithm, and provides interaction support to disambiguate the mapping information in the bundles and to conduct the mapping validation and addition tasks.

In the CogZ-e prototype, the usage and justification for applying the edge bundling algorithm as an approach for visually categorizing the mapping edges was guided by different cognitive theories, specially the categorization and cognitive load theories. They provide the connection between the reduction of cognitive load on a human being while processing information with the organization of the information being presented (see Section 2.6.1). Categorization theory provides evidence that humans tend to categorize in order to process the presented information to solve a problem. Thus, any problem solving system can benefit from categorizing information. The knowledge manager would be able to process information quickly if the categorization is done effectively, which can improve performance. By studying the principles of the categorization theory, the perceived categorization of an information system can be enhanced by employing different techniques to perform classification.

Cognitive load theory states that human working memory is limited and can only perform a limited number of tasks at a time. Similarly, cognitive load reflects the fact that when a high volume of information is presented to humans, they may experience difficulty in grasping the information and then performing the relevant task because of their limited working memory. The visual clutter concept also acknowledges that clutter in the presentation of information can create obstacles and thus lead to performance degradation.

Fundamentally, CogZ-e is a problem solving system (i.e., an ontology alignment system) that can benefit when the information (i.e., ontology mappings) is better categorized. This may reduce cognitive load on the knowledge manager who has to process the presented information (i.e., ontology mappings) in order to perform the relevant tasks (i.e. mapping validation tasks). In addition, the edge bundling process classifies or categorizes the information (i.e., edges) being given as input. Therefore, these theories can be considered as the theoretical foundations that support the potential benefits of using edge bundling approach as a categorization/organization method for ontology mappings.

As discussed in Section 2.6.1.1, the design choices and the interaction support for the prototype is developed by following the guidelines of information visualization theories and principles. The Gestalt Law of connectivity confirms that when the two objects are connected to each other by a curved or straight line, they are automatically perceived to have a relationship with each other. By following the guidelines of this Gestalt Law, curved lines have been used in the original system as well as in the prototype to link the source and target concepts that clearly shows that the concerned concepts are a match to each other. Therefore, mapping edges represent the connection between the matched entities without any confusion.

Furthermore, the Gestalt Laws of *proximity* and *similarity* provide evidence regarding the quick perception of relationships between any two objects by a human mind when they are near each other, and visually similar to one another. In other words, when two objects are positioned near one another or drawn similarly (using same colour or shape), they are automatically perceived as related. As explained by the pre-attentive processing phenomena, these relationships are automatically perceived by the human brain at great speeds. In this research, it was intended to use this capability of human mind to the knowledge manager's advantage while processing the mapping information.

When edges are grouped by the bundling algorithm, the mappings within their respective resulting bundles will be assumed as related because mappings are in proximity to one another. Since the overall shape of the mapping edges confined within a bundle is also similar, this further enforces the notion of the mappings being related. Hence, by looking at a bundle, the knowledge manager can perceive that the mappings present in the bundle are related, or at least the mappings are connecting similar concepts of the two ontologies. As discussed in Section 3.2.4.1, while making a decision about a particular mapping, if nearby mappings are also present they help the knowledge manager in making a more confident decision regarding the validation. Supported by the laws of proximity and similarity along with the pre-attentive phenomena, the knowledge managers are able to quickly perceive and process the related information because of bundles. It can reduce the time required to validate mappings that are present in the bundle, which in the end, can result in performance gains.

The principles from the Opponent Process Theory of Colour [57] were extensively used while making the colour choices in presenting different states of mapping edges during the process of conducting validation tasks. There are two different states of the mapping: candidate and confirmed. As it is already known, the resulting mapping set is a candidate mapping set after the bundling process, which implies that mappings produced at this point require the knowledge manager's confirmation or rejection. During the validation tasks, the mappings need to appear differently because of either transformation from one state to another or due to the temporary selection (by the knowledge manager) in different states. In other words, there are four conditions that require the mapping edges to be displayed visually distinguishable. These conditions are: when the mappings are in candidate state, when they are at confirmed state, when they are selected at the candidate stage, and when they are selected at the confirmed state. These four conditions may also occur at the same time on the interface.

By following these guideline of the theory to clearly show the two states of the mapping within or outside a bundle on the lighter beige coloured background of the interface, red was used for the candidate mappings and dark blue for the confirmed mappings. Since the background of the prototype was in lighter colour, the chosen colour blue was more suitable than the other available option of using green colour (due to the red-green channel). The existing baseline system also uses red and blue for highlighting candidate and confirmed mappings. Therefore, in order to maintain consistency between the two systems in terms of colour usage, the similar colour scheme was also adopted for the prototype system. When the knowledge manager selects a confirmed mapping within the pop-up window, the colour was changed from blue to green. According to the Opponent Theory of Colour, the blue colour that is used to show confirmed mappings over a beige (yellowish) coloured background would be clearly distinct. Similarly, the use of green for the highlighted confirmed mappings, red for the candidate mappings and black to highlight the candidate mappings will be clearly distinguishable.

The colours in the extended tooltip were also modified depending upon the state of the mapping being listed. The candidate mappings were shown in normal font with black colour whereas, the confirmed mappings were shown in bold font with red colour. Maintaining consistency between different parts of the interfaces is highly desirable [8], particularly, when the same information is presented at two different stages. For example, when knowledge managers see the list of mappings in the bundle via extended tooltip, the same mapping list is loaded in the pop-up window if they decide to work on that bundle. In order to maintain the consistency between these parts of the prototype's interface (i.e., the tooltip and the pop-window) as they both provide the same mapping information, the colour and font type of the mapping text in both the tooltip and the pop-up window has been kept the same (i.e., black colour with normal font for candidate mappings and red with bold font for confirmed mappings).

The interaction support is the fundamental feature of any information visualization system since it provides users more power over the control of the presented information. Since most of the mapping edges have become indistinguishable after the bundling process, the interaction options that can disambiguate the mapping information and provide details about mappings were required. During the alignment tasks, the knowledge manager needs to repeatedly view and interact with an individual mapping in order to perform the required operations. In order to accomplish this goal, the extended tooltip and pop-up window features were developed and implemented. The tooltip feature of the existing system was extended to provide details about the mappings presented in the bundle, whereas the pop-up window feature was implemented to support the validation operations. Like any other information visualization system, the interaction options were provided in the prototype in order to make the system more user friendly, effective, and easier to use.

With the modifications and support for the bundling approach from cognitive theories, along with the usage of theories and principles from information visualization in designing the prototype and the related features, it is expected that the approach could improve the organization of the mapping edges and may lead to the performance gain of a knowledge manager. The second research question from Chapter 1, asks: what are the benefits and drawbacks of using edge bundling in the context of ontology alignment tasks? In order to address this question and to validate the potential benefits and drawbacks of the edge bundling approach to the mapping validation and addition tasks, a user study evaluation was conducted, which will be explained in detail in Chapter 4.

Chapter 4

Evaluation

4.1 Purpose

There are a number of methods that have been proposed in the literature to evaluate ontology alignment systems [67]. However, the most common and accepted practice to evaluate any software systems involving human-computer interaction or information visualization is to conduct user studies in a controlled environment [11]. User studies provide researchers with valuable ways to examine the potential benefits of the proposed methods or systems and give insights regarding the system's behaviour in situations similar to real life use.

The design and execution of user-centered evaluations of ontology alignment systems is a difficult task because of the complexity associated with ontologies, the alignment methods, the nature of the typical tasks, and the amount of information that the user needs to comprehend and process to execute different tasks. It can be a challenging undertaking to design the tasks for the user study. The tasks need to reflect at least some of the activities that are performed by the actual users. At the same time, they should not be too complex for the available participants to perform in a limited time under the restricted atmosphere of a study. This is required to avoid any unnecessary conditions that may negatively affect the original purpose of the study.

Despite evaluating the broad range of activities involved in the ontology alignment task with the combination of other factors such as ontology sizes and complexities in a single user study, it is important to focus on certain aspects of the tasks involved in the ontology alignment while keeping the other conditions restricted. This limited scope of the study will counter some of the aforementioned challenges and difficulties, and will help in obtaining empirical data regarding the experiences, behaviours, and preference of knowledge managers while conducting most ontology alignment related activities using an ontology alignment system.

Since this proposed system (CogZ-e) is implemented within the CogZ framework, CogZ system was used as the baseline for comparison. The two systems employ different visual methods for mapping representation and interaction options to conduct the relevant tasks. It is anticipated that participants will be able to comprehend, use, evaluate, and compare both systems according to the benefits and drawbacks provided by each. The primary objective of this study is to evaluate the performance of the developed system (CogZ-e) for ontology mapping validation and addition tasks, in comparison with the baseline system (CogZ).

Ontology mapping validation and addition tasks are the most fundamental tasks among the other types of ontology alignment related tasks that are performed in an actual setting by the real users such as, ontology evaluation, alignment quality and accuracy evaluations, and addition/removal/updation of the input ontologies. Therefore, the study consists of these two fundamental tasks, which will also be conducted by participants in the same sequence in the study. A controlled environment for the study was also strictly maintained to avoid any irrelevant distractions that can affect the outcome of the evaluation of the main features of the system.

4.2 Hypotheses

Two sets of hypotheses were formulated based on the observations and knowledge about the two alignment system interfaces that are being studied here (i.e., the CogZ without bundling and the CogZ-e with bundling), and how they can affect the performance of the users while conducting the ontology alignment tasks. Both sets of hypotheses anticipate the efficiency, effectiveness, and subjective opinions of participants about the two interfaces while performing ontology alignment tasks. The first set is for the mapping validation tasks (H1) and the second set is for the mapping addition tasks (H2).

H1.1: *Users will take less time to complete mapping validation tasks with CogZ than CogZ-e.*

For mapping validation tasks, the user is expected to go through each mapping edge individually to validate (i.e., confirm or reject) the mapping. The baseline system CogZ presents the mappings as simple Bezier curved lines that the user can view, select, and then perform the validation. As such, for the mapping validation tasks it is expected that users will be able to select any mapping they want to verify one by one without much difficulty. However, in the case of CogZ-e, individual mapping edges have been combined into bundles, which introduces ambiguity in presenting the mapping information. In order to confirm or reject a mapping with CogZ-e, users are expected to interact with the bundles first before they can verify one or multiple mappings being shown in the tool-tip, and then in the pop-up window.

This ability to confirm multiple mappings at once can be regarded as a useful feature by users. However, the overhead of extra interaction (bundle interaction) with CogZ-e might consume more time as it is an additional step that the user has to perform before the mapping can be confirmed or rejected. While there are additional interaction features provided to support disambiguating within the bundle in CogZ-e, it is expected that using these interaction features will take additional time. In the case of CogZ, the mapping edges are separately drawn from each other, and thus no disambiguation is required. Therefore, it is anticipated that the users will be faster at mapping validation tasks with CogZ, due to the interaction overhead in CogZ-e.

H1.2: *In terms of accuracy, there will not be any significant difference between CogZ and CogZ-e while performing mapping validation tasks.*

Since both systems present the same data for making a decision about the validation of a mapping, it is expected that the accuracy will not be significantly varied between the two systems for performing validation tasks. In addition, these mappings are calculated by the automatic ontology alignment algorithms that are generally successful in finding mappings accurately especially when there is a exact entity to entity name match. The user is unlikely to invalidate a correct mapping, therefore, there will be no significant difference in terms of accuracy between the two systems.

H1.3: *The perceived level of difficulty of performing mapping validation tasks will be more in CogZ than CogZ-e.*

Since mappings are drawn separately from each other in CogZ, users have to select each mapping before confirming or rejecting it and then continue validating other mappings. As the general strategy to conduct these tasks is to follow a top-down approach, the users will likely do the same by verifying every candidate mapping one by one. This approach is quite traditional and may work for other scenarios, but it can be very inconvenient in the case when the number of candidate mappings are large in number. Since the individual mapping edges are transformed into bundles in CogZ-e, it provides a sense of a series of steps in the form of bundles that the user can finish to complete the overall validation task. Furthermore, the amount of information posed by individual mapping edges is also reduced to bundles, which encourages the efficient use of the user's cognitive memory and imposes less cognitive load. The structure added to the validation task due to bundles will help users in conducting the task more efficiently and effectively. In addition, the user can confirm one or multiple mappings simultaneously within a bundle, which can bring ease to the task at hand, leading to the impression that the perceived level of difficulty of performing mapping validation tasks is higher with CogZ when compared to CogZ-e.

H1.4: *For mapping validation tasks, users will be less confident in their results after using CogZ than CogZ-e.*

Since mappings are presented as-is with no particular order of organization in CogZ, users have to make a decision about a particular mapping with no additional help from the system. Using bundles, CogZ-e emphasizes the impression of relatedness between the mappings present within the bundle by drawing them similarly and placing them very near to each other (i.e., forming a bundle). This allows users to view related mappings at once, which can be useful while conducting validation tasks. It is important to note that the prototype does not calculate the proximity of the items in the inputted ontologies. In this respect, the functionality provided by the baseline system was used as-is and the bundling algorithm uses this information provided by the baseline system to perform the bundling process.

Since at least two or more mapping edges are contained in a bundle, these edges combined with relevant source and target entities may form an interesting area within the source and target ontologies from where multiple mapping edges are originating or terminating. Bundles emphasize these interesting areas and allow them to be automatically perceived by users. This information can aid the user in making a more informed validation decision about a mapping present in a bundle (i.e., in an interesting area) by looking at the other related mappings present in the same bundle, providing them more confidence about their validation decision. Thus, it is expected that users will have more confidence in results of their actions with CogZ-e compared to CogZ.

H1.5: *For mapping validation tasks, users will report that CogZ is less useful than CogZ-e.*

The systems CogZ-e and CogZ provide similar core functionality in terms of mapping validation with fundamental differences in visual methods used to present mappings and the interaction support provided to complete these tasks. In CogZ, users have no other option but to follow the potential strategy of manually scanning all presented mapping edges, and start verifying each mapping one by one. In the case of CogZ-e, the better categorization and organization of mappings could help users in verifying mappings quickly, in particular, the same-bundle mappings more easily, which could lead to their positive feedback regarding usefulness for CogZ-e. Since humans prefer to be presented with organized information that is relevant to a task, and bundles provide such information in an organized manner, it is expected that users will provide positive feedback in terms of usefulness for CogZ-e when conducting validation tasks. In addition, the absence of any help in CogZ for validation tasks may lead to users' negative feedback about the usefulness when compared to CogZ-e.

H1.6: *For mapping validation tasks, users will report that CogZ is similar in ease of use to CogZ-e.*

The process of validating a mapping is different in both systems (i.e., one by one validation in CogZ and disambiguation of mapping information with multiple mappings validation in CogZ-e), but is easy to follow in both systems. Therefore, the expectation is that users would have find the two systems being compared as similar regarding the ease of use.

H1.7: *For mapping validation tasks, users will prefer CogZ-e over CogZ.*

Due to the simple representation of mapping edges, it is expected that users will be faster using CogZ while conducting validation tasks. Even though mappings are grouped into bundles and require some disambiguation steps to view the individual mappings in CogZ-e, users' expected positive feedback regarding the perceived usefulness, and increased confidence in their results, as well as the decrease in the perceived difficulty of conducting validation tasks with CogZ-e will outweigh this drawback. Consequently, it is expected that due to the overall positive impression about CogZ-e, users will regard CogZ-e as a preferred ontology alignment interface over CogZ for mapping validation tasks.

H2.1: *Users will take less time to complete mapping addition tasks with CogZ-e than CogZ.*

Ontologies are developed to provide a hierarchical organization of the information of the domain they represent [40]. Thus, it can be expected that information is grouped in a logical manner. By bundling mapping edges, CogZ-e automatically highlights this grouped information that can be helpful for conducting mapping addition tasks. For example, if there is a mapping missing near a bundle, it is logical to start looking for the potential corresponding element near the opposite end of the bundle. The corresponding element may not be present, but this is still a preferred strategy to a usual top-down or random search strategy, and can be regarded as a useful feature.

Viewing mappings individually while performing the addition tasks is required. In order to see an individual mapping with CogZ-e, the user has to perform some disambiguation steps such as clicking on the bundle to open the small pop-up window containing the mappings in it, and then clicking on the mapping to see it separately (through mapping-highlight feature). These extra interaction steps could required some additional time, however, it is expected that the benefits of organization due to bundling will outweigh this disadvantage. On the other hand, CogZ represents individual mapping edges clearly with no extra steps required to view these mappings. Since no relevant help is provided in finding the potential concepts for mapping addition, it may lead to the typical approach of random scanning of individual concepts in the source and target ontologies to locate the missing match. This would require additional amount of time as no aid is provided to the user in order to complete this task.

Even though both systems have different visual representation and interactions methods, both require additional time at different stages of the process. In addition, the process of adding a mapping is similar in both systems after the potential concepts have been found. However, with the overall structure provided by bundling along with the prominent interesting areas, it is expected that users will be able to identify missing mappings more quickly with CogZ-e than CogZ.

H2.2: *In terms of accuracy, users will be more accurate with CogZ-e than CogZ while performing mapping addition tasks.*

In mapping addition tasks, the availability of related and background information to find or create a missing mapping is important. This can be helpful in making a more valid, informed, and accurate decision. Since the relevant mappings are most likely to be bundled together, the users will be able to exploit this related information more easily with CogZ-e. Therefore, it is expected that the users will be more accurate while performing mapping addition tasks with CogZ-e than CogZ.

H2.3: *For mapping addition tasks, the perceived level of difficulty of a given task will be less in CogZ-e than CogZ.*

By representing the overwhelming number of individual mapping edges on a limited screen-space by transforming them into a set of bundles, CogZ-e reduces the required cognitive load on users to process this information. The mappings present in the bundle can be automatically perceived by users as related, allowing them to intuitively process mappings with little cognitive effort. This helps in decreasing the perceived level of difficulty associated with the task at hand, making this approach advantageous. Therefore, it is expected that the perceived level of difficulty of mapping addition tasks would be lower in CogZ-e. In CogZ mapping edges are not well-organized and structured, which will make the mapping addition task more cumbersome and tedious to perform. Hence, it is expected that users would find the addition task more difficult to perform with CogZ.

H2.4: *For mapping addition tasks, users will be more confident in their results after using CogZ-e than CogZ.*

CogZ-e facilitates the process of locating any potential mappings after the bundling has been applied. The users can intuitively look for missing mappings among other concepts near already formed bundles. Since any new mapping created using this process can be created in the vicinity of already confirmed mappings, this would provide additional confidence to the user in terms of correctness of their actions. It is more consistent and a better strategy compared to random search to find additional mappings closer to the ones already available. CogZ-e helps users by making the relevant information available more prominently using bundles, which will allow users to make a more confident decision. Therefore, it is expected that this will be beneficial in terms of increasing their confidence in the results. In the case of CogZ, users have to locate the two potential related concepts in the given ontologies first and then create mappings. This lessens the users' confidence over their actions as they have selected both concepts by themselves with the absence of any relevant help from the system. Therefore, it is expected that user would be less confident in the results with CogZ than CogZ-e.

H2.5: *For mapping addition tasks, users will report that CogZ-e is more useful than CogZ.*

Since humans tend to organize the presented problem-related information into categories in order to understand a problem [63], it is expected that the mapping organization provided by CogZ-e would help users in locating the related concepts more easily, leading towards positive feedback regarding the usefulness of CogZ-e.

However, for CogZ, users have to adopt the typical strategy of manually scanning through the ontology trees and scattered mapping edges to consider potential missing mappings. The manual searching may lead to user's negative feedback about the usefulness of CogZ when compared to CogZ-e.

H2.6: *For mapping addition tasks, users will report that CogZ-e is similar in ease of use to CogZ.*

Once the related concepts for a potential mapping have been identified, the process of creating a mapping among them is similar in both CogZ-e and CogZ. After scanning through concepts in CogZ, users have to select the two potential concepts and then perform a mapping addition operation. In the case of CogZ-e, users can take advantage of the more organized mapping information but have to first disambiguate the bundling information before performing a mapping addition operation. Therefore, the expectation is the benefits versus the drawbacks of both systems will be balanced and users will report that CogZ-e is similar in ease of use to CogZ.

H2.7: *For mapping addition tasks, users will prefer CogZ-e over CogZ.*

Regardless of the absence of any potential performance gain in terms of time required to complete the mapping addition tasks, it is expected that users will value their increased ability to conduct the tasks more easily and accurately with CogZ-e. Furthermore, their positive feedback regarding the usefulness, and increased confidence on their actions for CogZ-e will lead to an overall positive impression about CogZ-e, resulting into the expectation that users might prefer CogZ-e over CogZ as an ontology alignment interface for mapping addition tasks.

4.3 Methodology

The user study was designed to evaluate the benefits and drawbacks of edge bundling for ontology alignment tasks. In order to allow direct comparisons between the two interfaces, a within-subjects design was employed. The participants used each interface only once, conducting ontology alignment tasks with two different pairs of test ontologies.

Since this enhancement was implemented within the CogZ framework, CogZ was used as the baseline comparison point. Within the study design, the interface was treated as the independent variable CogZ-e (with bundling) vs. CogZ (without bundling). The dependent variables consisted of the time-to-task completion, accuracy of the task outcomes, participants' perceived level of difficulty, confidence over their results, and their perceptions of usefulness and ease of use. At the end, open-ended feedback was also collected from the participants.

4.3.1 Tasks

In order to successfully evaluate ontology alignment interfaces, it is important to clearly define realistic ontology alignment tasks that can also be used in the study. The designed tasks should include the basic ontology alignment activities that are performed by the real users. The activities should also allow the direct testing of the two fundamental aspects of the system i.e., the visual representation methods used for presenting the information, and the interaction options provided. The results of the user's performance while conducting these carefully designed tasks in this manner, can then help in evaluating the effectiveness of these important aspects of the system.

To evaluate the two ontology alignment interfaces, the study is structured around the participants performing the two fundamental high-level ontology alignment tasks: i.e. mapping validation and mapping addition tasks. Since the logical flow of performing these alignment tasks is to first conduct mapping validation tasks and then mapping addition tasks, the same sequence of task execution is also maintained in the study. In addition, the activities involved in these tasks ascertains that participants interact with the two important aspects of the system (i.e., visual methods used to present mapping information, and the available mapping interaction options).

Since it is beneficial to provide context to participants so that they can relate to the situation being defined in the tasks and apply their own knowledge [51], a hypothetical scenario was provided for every task. Participants were asked to imagine themselves as data managers working for a data mining company. Asking the participants to put themselves in a situated task adds context to the activity they are to perform, even if they are not familiar with the details of the task.

For the mapping validation task, participants were asked to verify five out of a total fifteen candidate mappings. Following the completion of this task, with the same set of ontologies, participants performed the mapping addition task. For this task, participants were again given a similar activity in which they were asked to add five new mappings among the source and target concepts of the given ontology set, which they believe should be mapped to each other. The detailed task descriptions that were presented to participants are as follows:

Task 1: Validation of mappings

Imagine you are a data manager in a department of a data mining company which deals with the data integration. You have been hired to map two ontologies to each other which represent the domain of an academic conference. The mappings have already been computed and presented to you. Your task is to verify any 5 of the 15 mappings shown to you. Tell the investigator clearly which mapping you confirm, which mapping you believe should be removed and why they should be confirmed or removed.

Task 2: Add a mapping

Imagine you are a data manager in a department of a data mining company which offers services for data integration. You have been hired to map two ontologies to each other which represent the domain of an academic conference. The mappings have already been computed and presented to you. This time your task is to add a mapping link between two concepts which you believe are similar to each other and the mapping link is missing between them. Please add any 5 mappings that you feel are missing, and tell the instructor why the mapping should be added.

These ontology alignment tasks can be carried out using both interfaces. In addition, they were designed to also comply with the requirements of being moderately complex, be completed easily in the controlled environment of the user study, as well as to reflect a basic level of the tasks that are performed by the actual users with real systems and complex ontology datasets.

4.3.2 Selection of Datasets

The selection of the datasets was one of the important steps in the design of the study. The main eligibility criteria for the datasets is that they represent commonly known domains, and are similar to each other. Furthermore, these datasets should not be complex and vary significantly in terms of the numbers of contained concepts, the inheritance level among the concepts, and the number of generated mappings when aligned with another dataset. This is necessary to minimize any impact of datasets' dissimilarity and complexity as much as possible on the study results.

For the purposes of this study, two sample ontology sets (two ontologies in one set) representing commonly known domains were used. Both of these ontology sets have been made freely available on the Web to be used for research purposes. One ontology set represents the domain of a university as an organization [29], and the other defines the domain of an academic conference [26]. It was expected that since participants would be recruited from the student body, they would be knowledgeable about these domain areas.

The summary of important characteristics of each of the ontology dataset is listed in Table 4.1. It can be viewed that the two sets of test ontologies chosen for this study are similar sized in terms of the number of entities, depth of the hierarchical structure, number of generated mappings for the validation tasks, and number of mappings for the addition tasks. In particular, each ontology of the datasets on average contained around seventy five concepts, the depth of inheritance was one, the number of mappings produced between the two ontologies of a dataset was between fifteen to twenty, and the potential mappings to be found were between six to seven (see Table 4.1). The complete datasets are also available as Appendix B of this thesis.

Table 4.1: Summary of characteristics of the test ontology datasets

Domain	University dataset	Conference dataset
Number of entities	55 111	59 73
Hierarchical depth	4 4	4 3
No. of available mappings for validation	21	18
No. of potential mappings for addition	7	6

4.3.3 Procedures

As a first step of the study, a pre-study questionnaire was administered to record the educational background characteristics of every participant. Since participants' experience with ontologies could greatly affect the outcome of the study, these questionnaires were required to determine the participants' level of prior knowledge with ontologies or knowledge representation, and their prior experience of using any ontology alignment systems.

Ontologies and the related concepts are not commonly known, therefore in order to ensure a preliminary level of knowledge and experience, a brief training session was conducted prior to the beginning of the study. In this training, a short introduction about ontologies, ontology alignment, ontology mappings, and the overall problem domain was provided. In this training session, participants were also requested to perform some sample mapping validation and addition tasks using another test ontology set representing the domain of common things in one's daily life [26] (available in Appendix B). This step was taken to ensure that each participant had a preliminary understanding of both interfaces and the tasks that were to be performed in

Participant groups	Interface exposure with ontology set	
G_1	C with O_A	C_e with O_B
G_2	C_e with O_A	C with O_B
G_3	C with O_B	C_e with O_A
G_4	C_e with O_B	C with O_A

Table 4.2: Round-Robin rotation of tasks and interfaces. Let the CogZ system be C and CogZ-e as C_e and consider the used ontology datasets as O_A and O_B.

the study. In realistic scenarios, knowledge managers are overall familiar with the domain and the ontological structure of the source and target ontologies. Therefore, before beginning the study, all participants were also provided with a short briefing over the domains of the test ontology sets (i.e., university and academic conference) along with the ontological structures.

In order to minimize potential ordering effects of the interface exposure and the test ontology sets, a 2X2 Graeco-Latin square [38] design was adopted to systematically rotate the order of the interface exposure and the test ontology datasets (see Table 4.2). In order to mitigate any potential learning or other ordering effects on the results of the study, such as some participants' familiarity with one domain of an ontology set than the other, participants in the study were assigned to different groups in a round-robin fashion.

Following the pre-defined task order, for each test condition, participants were first asked to validate the mappings produced by the automatic algorithm. This task included confirmation of candidate mappings that are correct, and rejection of inaccurate mappings. After the completion of this task, participants were asked to add any additional mappings they think that were missed by the automatic algorithm. As explained earlier, this ordering of tasks was kept constant in the study design to mimic the normal process for conducting ontology alignment tasks in realistic scenarios.

During each task, the overall time to complete the task was noted. Immediately after each task completion, subjective measures of participants' confidence in performing the tasks and perceived task difficulty were measured using an in-task questionnaire. The same in-task questionnaire was also used to determine the participant's level of familiarity with the domain of the test ontology set.

The accuracy of the mapping validation and addition decisions taken by participant's were verified post-hoc by a panel of ontology experts after all the study tasks were completed. Following that, a post-study questionnaire was administered to measure perceptions of usefulness and ease of use about the ontology alignment interfaces for both the validation and addition tasks. With some modifications, the questionnaire design was borrowed from the guidelines of Technology Acceptance Model (TAM) [17] to calculate these measures.

The default TAM questionnaires provide effective feedback and are helpful for evaluating new technological systems against previously known systems. However, an issue with the default TAM questionnaires is that they require the participants to have enough prior context in order for them to correctly compare the system being evaluated with their past experience with a similar system being used in an analogous context. However, in this case, this assumption is not appropriate because participants have very little knowledge about the ontologies and, more importantly, have no prior experience in dealing with the ontology alignment systems and tasks. Thus, in order to accurately evaluate the features of the interfaces, the questions of the TAM instrument along with the available choices were modified. Rather than asking participants to provide answers for each interface on a Likert scale, it was asked that for each question they rank one interface over the other, or indicate that they are the same.

At the end, participants' were asked to indicate their preference for an interface based upon their overall experience developed while conducting all the tasks during the study with both interfaces. In addition, a debriefing session was conducted to allow participants to provide general comments regarding the two interfaces. The questions in this session included their opinions on the interfaces, the positive and negative features of the interfaces, and the major reasons for their preference for an interface. During the study the participants were also encouraged to follow the think-aloud protocol [47], which helped in analyzing the thinking process of the users as well as in discovering the difficulties participants were having in performing their assigned tasks.

4.3.4 Analysis

To compare the time to task completion measure of the two interfaces while conducting ontology alignment tasks, the overall times to complete the mapping validation tasks and addition tasks were used. Since these measures are quantitative in nature, they were analyzed via analysis of variance (ANOVA). Similarly, the accuracy of the mapping validation and addition decisions during the tasks were also analyzed using analysis of variance (ANOVA) [71]. The time to task completion and accuracy results of the alignment tasks are analyzed and reported independently for each type of task with the used test ontology dataset.

The data collected from the in-task questionnaires for the dependent variables for familiarity with ontology domains, and confidence in results is qualitative. Similarly, the aggregated responses from post-study questionnaires for the dependent variables for perceived usefulness, and perceived ease of use are also qualitative in nature. To determine that the two data samples obtained for each of these dependent variables come from distinct populations, and the samples do not affect each other, and thus are independent from each other, they were analyzed using pair-wise Wilcoxon-Mann-Whitney tests [4] .

Since the TAM instrument provides six questions that relate to usefulness, and six that relate to ease of use, the responses were aggregated for all participants based on these underlying constructs in order to gauge their overall perceptions. Furthermore, since it was wished that participants make a comparison between interfaces, but each interface was used only once with a different data set, it is not possible to analyze the responses by each topic domain separately.

To determine participants' preference for an interface the data samples were also obtained. Since these data samples are related pairs, the preference ranks were analyzed using pair-wise Wilcoxon signed rank test [83]. Tests that are found to be significant are represented in the bold font.

4.4 Evaluation Results

4.4.1 Participant Demographics

Twenty four participants were recruited from the senior undergraduate and graduate student population of the Computer Science, Engineering, and Business departments to participate in the study. Pre-study questionnaires were designed to deduce participants' prior experience regarding ontology alignment, along with their knowledge about the overall problem domain. From the responses, it was inferred that the participants belong to a relatively homogeneous group, with very little to no experience with ontologies themselves, and no prior experience using ontology alignment systems (see Table 4.3).

After completing each task, the participants were asked to report their familiarity with the domains of test ontology datasets. Since the familiarity with the domain of the test ontology datasets is independent of type of the task performed, the responses

Table 4.3: Features of the participant demographics.

Have you taken any ontology development or knowledge representation course?	Yes:20% No:80%
Have you used any structure or database or ontology mapping system before (for e.g., COMA++, SMatch, Protege, Anchor-Prompt)?	Yes:0% No:100%
How confident are you about your knowledge regarding ontologies? (asked after the training session)	Very confident: 4% Confident: 42% Neutral: 33% Unconfident: 17% Very unconfident: 4%

have been aggregated to facilitate further analysis of the results. The distribution of the responses can be seen in Figure 4.1. As can be seen from these results that the participants reported a normal range of familiarity levels, confirming the expectation that the participants would be knowledgeable about these domains (i.e., university and academic conference) of the test ontology datasets.

4.4.2 Time to Task Completion

The average time taken by participants to perform the mapping validation tasks is shown in Figure 4.2. As it is can be clearly seen, participants took significantly less time with CogZ-e interface using edge bundling to verify the mappings with both test ontology datasets. The results of a statistical comparison using ANOVA are reported in Table 4.4. The differences in completion time for the validation tasks over both ontology sets were found to be statistically significant.

In general, there are two competing aspects of edge bundling that can have an effect on the time taken to perform the mapping validation tasks. As previously

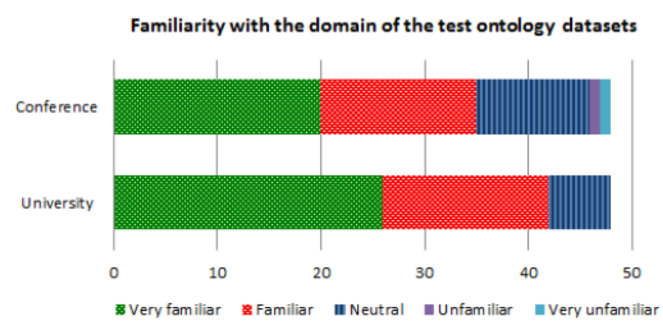


Figure 4.1: Distribution of the aggregated responses about familiarity with the domain of the test ontology datasets.

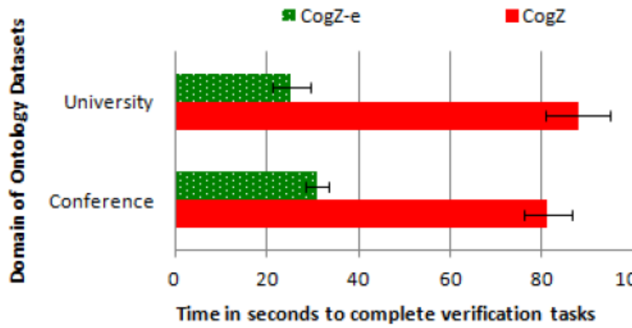


Figure 4.2: Average time taken to complete the validation tasks. Error bars are reflecting the standard error about the mean.

Table 4.4: Statistical analysis of the differences in the time to task completion of the validation tasks.

Ontology Domain	ANOVA
University	$\mathbf{F(1,\ 22) = 71.046, p < 0.0001}$
Conference	$\mathbf{F(1,\ 22) = 78.791, p < 0.0001}$

discussed, by bundling the edges, the clutter within the visual interface is reduced and the task of performing mapping validation becomes more structured and focused. This has the potential to increase the speed at which the tasks can be completed. However, there is an added interactive element of disambiguating a bundle that takes additional time. Under hypothesis H1.1, it was predicted that the required disambiguation steps due to bundles would negatively dominate the advantages of bundling. However, these results invalidate this hypothesis, which is promising in terms of time efficiency of participants and is even better than what was expected. From these results, it can be clearly seen that the time-saving effects of providing the mappings in bundles greatly outweighs the extra work required to disambiguate a bundle.

In addition to the fact that bundling provides a more structured approach toward mapping validation tasks, another potential reason for this outcome is the provided ability of validating multiple mappings at once, which is a consequent feature of using bundling (i.e., multiple mappings are grouped therefore multiple mappings can also be validated together at once). Using this feature, a significant amount of time can be saved, leading to the overall decrease in the amount of time taken by participants while conducting validation tasks compared to CogZ.

Since the primary focus of this research is on reducing the amount of information that was required to be processed by knowledge managers simultaneously, the mapping results were provided by the baseline system, and were not modified. The amount of error (if any) introduced in the validation results and the consequent factors are not the focus of the conducted research. In fact, the idea behind providing the mapping information to the analyst for validation is to address the potential for such mapping errors.

The results for the time taken to complete the mapping addition tasks are reported in Figure 4.3. These results are mixed, with the statistical analysis showing no significance between the data for either test ontology dataset (see Table 4.5). As a result, it is concluded that edge bundling had no effect on the time taken to perform the mapping addition tasks.

This result is contrary to the expected outcome (H2.1). It was anticipated that the structure provided by the edge bundling would have aided the participants in finding missing mappings. One possible explanation for not realizing a time savings in these tasks is that the added overhead of the interactive disambiguation may have been more prevalent during these addition tasks. During the mapping validation tasks,

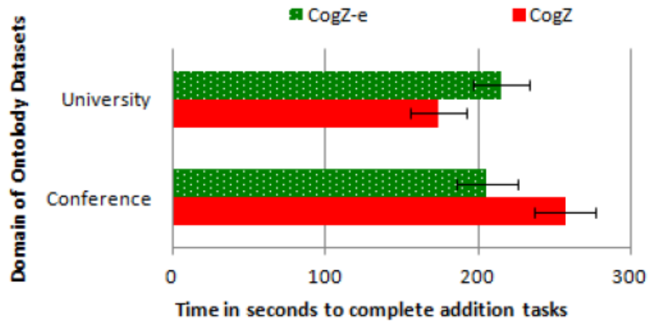


Figure 4.3: Average time taken to complete the addition tasks. Standard error about the mean is shown using the error bars.

Table 4.5: Statistical analysis of the differences in the time to task completion of the addition tasks.

Ontology Domain	ANOVA
University	$F(1,22) = 2.921, p = 0.101$
Conference	$F(1,22) = 2.077, p = 0.163$

this overhead was incurred once for each bundle since the disambiguation could be done in concert with the validation of all mappings within the bundle. However, for the mapping addition tasks, the participants may have incurred this time penalty of disambiguation repeatedly as they analyzed the bundles to determine an appropriate end point for a candidate entity in one of the ontologies. However, it should be noted that this added time did not result in significantly more time being taken, but instead balanced out the time savings as a result of the edge bundling itself.

4.4.3 Accuracy

After the participants finished the validation and addition tasks, the confirmed, rejected, and added mappings were examined by a panel of experts to ensure accuracy (correctness). The panel consisted of the two supervisors of this thesis including the author. The accuracy of the validation tasks was perfect (100%) for both tasks and both interfaces. This result shows that the visual encoding of mappings as edges, whether they be bundled or not, is an effective means of representing the mappings. Furthermore, it is an indication that the participants were careful in completing the mapping validation tasks. These results confirm the expectation that since candidate mappings are calculated by automatic alignment algorithms for both systems, there will be no significant difference in terms of accuracy for validation tasks (H1.1).

However, for the accuracy of the addition tasks, there were differences over both test domains, as shown in Figure 4.4. With the university ontology dataset, the participants' average accuracy was 90% with CogZ and 98.3% with CogZ-e. For the academic conference ontology dataset, participants were 86.6% accurate while using CogZ, and 96.6% accurate when using CogZ-e. In order to check significance of these results, ANOVA was carried out on both accuracy data samples. From the calculations of ANOVA between the interfaces (see Table 4.6), it was found that the improvements in accuracy as a result of using the bundling interface (CogZ-e) were statistically significant for both test ontology sets.

These results suggest that the structure imposed by the edge bundling actually helped participants to find appropriate mappings to add. That is, by using the bundles as a guide for finding missing mappings, participants were able to make better choices

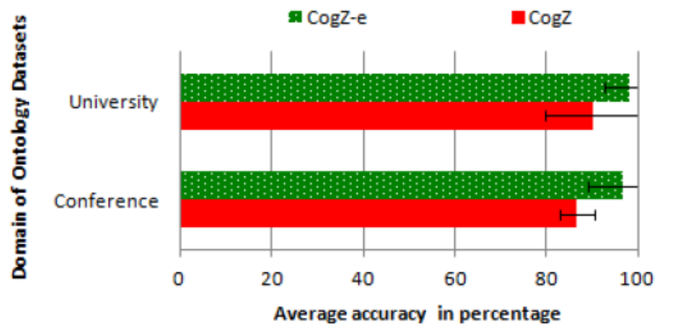


Figure 4.4: Average accuracy for addition tasks in percentage. Standard errors of mean of the data sample is also shown using the error bars.

Table 4.6: Statistical analysis of the differences in the accuracy of the addition tasks.

Ontology Domain	ANOVA
University	**$F(1, 22) = 5.851, p < 0.05$**
Conference	**$F(1, 22) = 5.210, p < 0.05$**

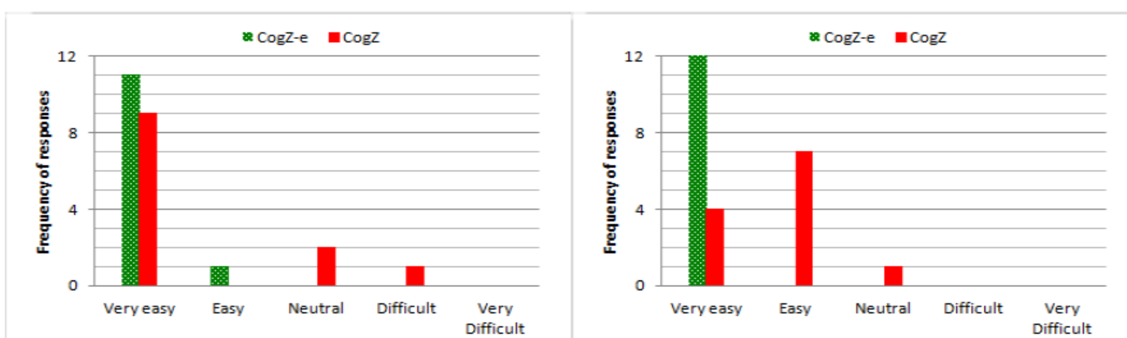
for such additions than using the non-bundled representation of the mappings. So, while the bundling did not improve the time with which such bundling addition tasks could be completed (as reported in the previous sub-section), it was able to improve the accuracy of the results. Therefore, there is empirical evidence to support hypothesis H2.2.

4.4.4 Perceived Difficulty of Given Tasks

After each task was completed, the participants' perceived level of difficulty regarding the performed tasks was recorded. The aim was to compare how this subjective measure varies for the same sequence of tasks while using the two interfaces with the test ontology datasets. The distribution of responses to this question for the validation tasks with the domain of test ontology datasets are shown in Figure 4.5. For both the university and academic conference ontology datasets, the average perceived difficulty was slightly lower with CogZ-e compared to CogZ.

In order to determine the statistical significance of these results, pair-wise Wilcoxon-Mann-Whitney tests were performed for every ontology dataset. It was found that, for mapping validation tasks, there was no significant difference among the average response to the perceived difficulty between CogZ and CogZ-e with the university domain ontology dataset (see Table 4.7). However, for the conference domain ontology dataset, it was found that participants found the validation task easier with CogZ-e.

It was expected that with reduced visual clutter on the screen due to bundling, the perceived level of difficulty for the time consuming validation task would be lower with CogZ-e. However, since this could only be shown for only conference domain ontology dataset, this result does not entirely confirm the expectation (H1.3). Although both interfaces provide very different methods of representing and interacting



(a) (b)

Figure 4.5: Distribution of responses for perceived difficulty for validation tasks (a) university domain (b) conference domain.

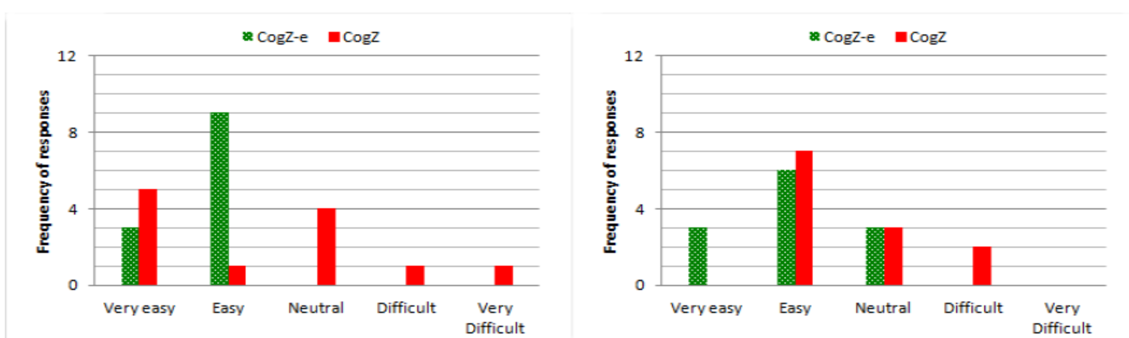
Table 4.7: Statistical analysis of the responses for the perceived difficulty for mapping validation tasks.

Domain of Ontology Dataset	Wilcoxon-Mann-Whitney tests
University	$Z = -1.201, p = 0.230$
Conference	$\mathbf{Z = -3.361, p < 0.001}$

with mappings, mixed results were found that indicate that perhaps prior knowledge could have influenced the results.

Similarly, for the mapping addition tasks, the average response to perceived difficulty with CogZ-e was slightly lower than CogZ (Figure 4.6). However, no significant difference was found when the pair-wise Wilcoxon-Mann-Whitney tests were performed for both the ontology datasets (see Table 4.8).

It was anticipated that the organized mapping information with CogZ-e will decrease the difficulty that is normally associated with mapping additions tasks (H2.3). However, from the above results, it can be inferred that participants did not find



(a) (b)

Figure 4.6: Distribution of responses for perceived difficulty for addition tasks (a) university domain (b) conference domain.

Table 4.8: Statistical analysis of the responses for the perceived difficulty for mapping addition tasks.

Domain of Ontology Dataset	Wilcoxon-Mann-Whitney tests
University	$Z = -0.919, p = 0.358$
Conference	$Z = -1.622, p = 0.975$

significant difference between the two interfaces being evaluated in terms of perceived difficulty of the task. Thus, CogZ-e neither increased nor decreased the perceived level of difficulty in performing the mapping addition tasks.

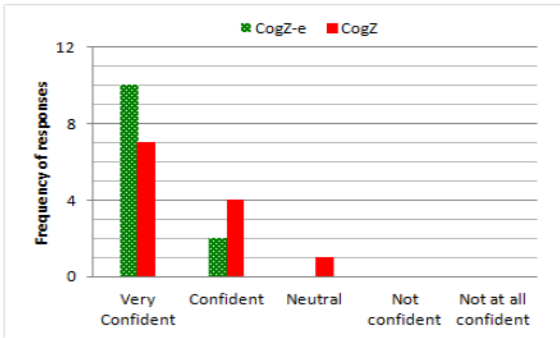
It is evident that regarding the validity of hypothesis H1.3, mixed results were obtained, such that, for the university domain of the test ontology set, there was no advantage for using CogZ-e but for the conference domain of the test ontology, CogZ-e helped participants' in conducting validation tasks with more ease. For the hypothesis 2.3 the validity could not be confirmed, implying that for mapping addition task, CogZ-e provides no benefit in terms bringing ease to the task. Although the sample shown in Figure 4.6 is good, the test results confirm that this sample is not inference to the whole population. The main reason for these results could be that participants indicated during the pre-study questionnaire that they have very little to no prior knowledge regarding ontologies or ontology alignment systems. Since participants lack the relevant knowledge about the overall problem domain and the fact that the core functionality provided by the two systems is similar, it is possible that participants do not have an adequate alternative to compare their current performance with their prior experiences with similar systems, which may have affected these results.

4.4.5 Perceived Confidence in Results

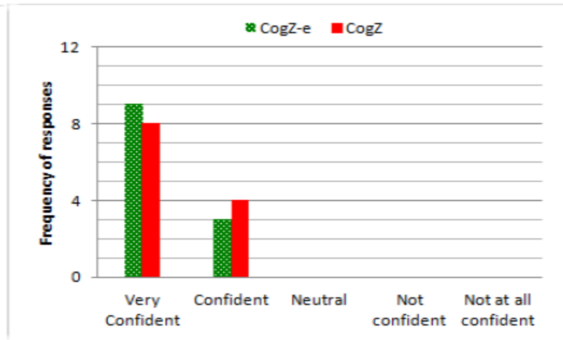
The measure of perceived confidence aims to compare how participants' confidence in their results or task outcomes changes while performing the given tasks using both interfaces and test ontology datasets. The distribution of responses to the perceived confidence can be seen in Figure 4.7 for the validation tasks. It is evident that similar responses were reported for both interfaces in terms of confidence for the mapping validation tasks. On the basis of the average, the confidence with CogZ-e was slightly higher when performing validation tasks compared to CogZ. However, the results of pair-wise Wilcoxon-Mann-Whitney tests indicate that these differences are not statistically significant (see Table 4.9). This implies that for both interfaces, the participants indicated high confidence, which is a sign of the value of visually representing the ontology alignment outcomes to the participant and allowing them to validate the results. These results also indicate that in the opinion of participants, both interfaces are similar in terms of perceived confidence over their actions when performing the validation tasks.

These results invalidate hypothesis H1.4, in which it was predicted that the benefits of viewing related information at once before carrying out a validation decision would increase participants' confidence on their mapping validation decisions. It is possible that due to the same core data being used by both systems and participants having to make their decisions based on the same data (i.e., mapping information), they were equally confident over their validation decisions, and usage of the different mapping representation neither increased nor decreased their perceived confidence.

For the mapping addition tasks, the average confidence regarding CogZ-e was



(a)



(b)

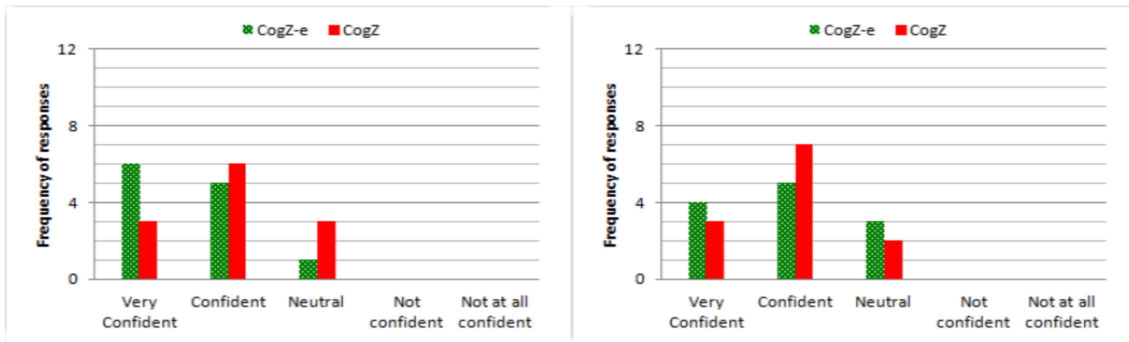
Figure 4.7: Distribution of responses for perceived confidence for validation tasks (a) university domain (b) conference domain.

Table 4.9: Statistical analysis of the responses for the perceived confidence on correctness for mapping validation tasks.

Domain of Ontology Dataset	Wilcoxon-Mann-Whitney tests
University	$Z = -1.382, p = 0.167$
Conference	$Z = -0.440, p = 0.660$

slightly higher than CogZ, indicating that the bundling interface has stronger positive results for both domains of the test ontology datasets (see Figure 4.8). In order to determine the significance of the results, pair-wise Wilcoxon-Mann-Whitney tests was carried out (see Table 4.10).

The tests show that there are no significant differences among the confidence over the two interfaces for performing the mapping addition task. Similarly, this result also does not support the relevant hypothesis (H2.4). This hypothesis predicted that CogZ-e will be able to increase participants' confidence in their mapping addition decisions since related information is grouped, creating interesting areas that help



(a) (b)

Figure 4.8: Distribution of responses for perceived confidence for addition tasks (a) university domain (b) conference domain.

Table 4.10: Statistical analysis of the responses for the perceived confidence on correctness for mapping addition tasks.

Domain of Ontology Dataset	Wilcoxon-Mann-Whitney tests
University	$Z = -1.411, p = 0.158$
Conference	$Z = -0.031, p = 0.975$

finding new mapping connections. However, these results suggest that participants were equally confident in their actions using both interfaces. As mentioned before, since the core data used by these two systems in terms of mapping addition information is the same, participants found little difference between the interfaces in terms of their confidence.

4.4.6 Perceived Usefulness and Ease of use

After the subjective reactions for every particular task were recorded, overall perception about the usefulness and ease of use of the interfaces being compared were collected via post-task questionnaire. At this stage, the participants would have conducted all the tasks using both interfaces with the two test ontology datasets, which enables them to effectively judge and compare the interfaces based upon their experience.

It was expected that due to the better visual representation and interaction support, participants would find CogZ-e more useful while conducting the mapping validation tasks (H1.5). It was also expected that since the process to validate a mapping is easy to follow in both systems, participants will not report any difference in terms of the ease of use (H1.6). The aggregate responses to the perceived usefulness and ease of use questions for the mapping validation tasks are illustrated in Figure 4.9. The results show that there is a strongly positive perception of both the usefulness and the ease of use of the edge bundling interface for the mapping validation tasks. Table 4.11 shows the results of a Wilcoxon signed rank test on this data sample, which confirms the statistical significance of the finding.

The results show that CogZ-e has been regarded as more useful and easy to use compared to CogZ for the mapping validation tasks. The hypothesis H1.5 has been supported by these results. However, H1.6 is not supported, which is better than the expected outcome. The reason could be that in addition to the organized mapping information, participants also acknowledged the feature of validating multiple mappings at once, which resulted in their additional positive feedback towards the ease of use of CogZ-e.

For the mapping addition tasks, it was predicted that participants will appreciate the grouping of the presented information using edge bundling, which will result in their positive feedback towards the perceived usefulness (H2.5), and they would find a

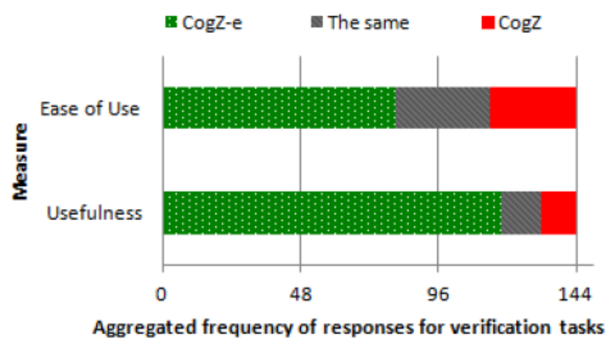


Figure 4.9: Aggregated responses for perceived ease of use and usefulness measures for the mapping validation tasks.

Table 4.11: Statistical analysis of the differences in the perceive ease of use and usefulness of the two interfaces for the mapping validation and addition tasks.

Task Type	Perceived Ease of Use	Perceived Usefulness
Mapping Validation	**Z = -4.841, p < 0.001**	**Z = -9.297, p < 0.001**
Mapping Addition	**Z = -5.327, p < 0.001**	**Z = -7.673, p < 0.001**

small difference between the interfaces in terms of the ease of use (H2.6). Figure 4.10 shows the results of the perceived usefulness and ease of use questionnaires for the mapping addition tasks. Similar to the case of mapping validation tasks, there is also an overall positive feedback for both the usefulness and ease of use of the edge bundling interface when conducting mapping addition tasks. The statistical significance of these results was confirmed through a Wilcoxon-Mann-Whitney tests using a pair-wise grouping of the interfaces (see results in Table 4.11).

The results depict that CogZ-e is superior to CogZ in terms of usefulness (H2.5) and ease of use (H2.6) for the mapping addition tasks that participants performed. Since these results confirm that CogZ-e not only received more favourable votes in

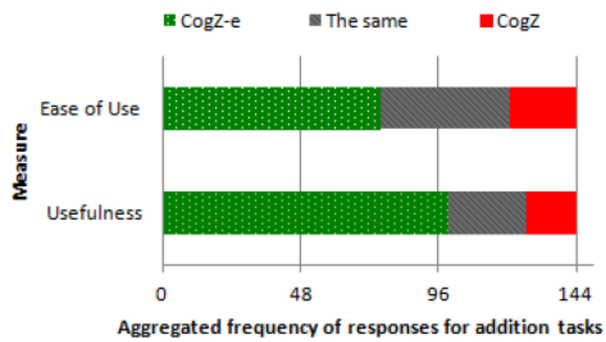


Figure 4.10: Aggregated responses for perceived ease of use and usefulness measures for the mapping addition tasks.

terms of usefulness and ease of use measure but also the results were statistically significant, the hypothesis H2.5 relating to usefulness was supported. However, hypothesis H2.6 predicting a small difference about the ease of use for both systems could not be supported, which is again better than the expected outcome. The reason may be that once participants are more familiar with CogZ-e, they were also able to use bundling with ease for addition tasks. They reported that the bundling also helps them in locating the potential mappings more easily because of the categorization of already-formed mappings.

These results provide an indication that the participants appreciated the less visually cluttered interface produced by the edge bundling, finding the interface both easier to use and more useful for the prescribed tasks. While edge bundling introduces a degree of ambiguity in the mapping representation, this ambiguity was not perceived as being detrimental to the ease of use nor the usefulness, perhaps due to the simple yet effective approach for disambiguating the bundles.

4.4.7 Preference

After the study was completed, participants were asked to give their preference for an ontology alignment interface. For the mapping validation tasks, twenty-two participants out of twenty-four indicated their preference for CogZ-e (92%) over CogZ. Similarly, twenty participants out of twenty-four indicated their preference for CogZ-e (83%) over CogZ for mapping addition tasks. A Wilcoxon signed rank test found statistical significance in the preference of CogZ-e over CogZ for mapping validation tasks ($\mathbf{Z = -4.082, p < 0.0001}$) as well as for mapping addition tasks ($\mathbf{Z = -3.266, p < 0.0001}$).

The hypotheses, H1.7 for mapping validation tasks and H2.7 for mapping addition tasks, predicted that due to the overall positive impression of CogZ-e, participants will prefer CogZ-e over CogZ for both kinds of tasks. The above results clearly indicate the supremacy of CogZ-e interface over CogZ for both fundamental tasks related to ontology alignment i.e. mapping validation and addition, and support the relevant hypotheses.

4.4.8 Open-ended Feedback

Participants were asked to provide their general opinion about the features of the interface, and their suggestions for future improvements. All of the feedback provided by the participants were about the visual representation and interaction tools of the CogZ-e interface. In general, the comments were overall positive about the systems and tasks with some criticism. For the CogZ-e interface, the general consensus was that it is useful and helpful for the mapping validation and addition tasks. It is a normal practice to qualitatively define individual participants and their comments rather than quantifying them and trying to fit them into a group. Since the data obtained was qualitative, participant numbers are specifically mentioned against their opinion in the following paragraphs to clearly define that different participants gave different opinions.

Participants appreciated the grouping of mapping information with CogZ-e in comparison to CogZ, and the interaction mechanisms provided for the bundle interaction. It was mentioned that having the groups of (most likely related) mappings within bundles, allowed them to view and make effective decisions easily. It was said that bundling increases focusing ability, provides overview, creates a neat interface, and helps in quickly validating relevant mappings (participant nos. 4, 7, 9, and 14). Furthermore, the ability to validate multiple mappings at once was highly acknowledged (participants 12 and 17). It was said that this feature greatly increases the efficiency of the system and increases the user's overall productivity. Few participants also mentioned that they liked the color encoding scheme used for edges to reflect different states of the mapping and for the mapping highlight feature (participants 4 and 17).

For the mapping addition tasks, the overall responses were positive. Participants numbers 5, 12, 16, and 20 mentioned that it is easier to follow and highlights the areas for potential new mappings, which allows better searching strategy for finding potential concepts compared to the top-down or random scanning approach. However, some participants mentioned that the requirement of extra bundle interaction before addition of the concept is troublesome and seems unnecessary (participant nos. 4, 10, 12 and 16). It was also mentioned that required repetition of disambiguation steps to follow an interesting mapping that is hidden within bundles, creates hurdles in conducting the mapping addition tasks (participant nos. 4 and 10). Some participants also indicated that for mapping addition tasks, the difficulty level with both systems was similar, and CogZ-e provides no significant advantage in terms of reducing the difficulty associated with this tasks.

4.4.9 Think-aloud Protocol Findings

During the study participants were encouraged to follow the think-aloud protocol. During which it was observed, that participants were able to effectively use CogZ-e's features for both types of tasks in general. More specifically for the validation tasks, participants were very quick in completing their tasks and were clearly acknowledging the benefits and ease of this task using CogZ-e. For addition tasks, it was observed that participants were having difficulty in dealing with the interaction overhead and were not pleased by the requirement of continuous disambiguation of mapping information.

Some of the participants provided suggestions to improve the system. It was mentioned that the requirement of keeping the Alt key pressed to view mappings needs to removed, and it can be limited to only when the user wants to work on the mappings. In addition, it was mentioned that an undo button for reversing any mapping validation and addition decision can be useful. Furthermore, improving the overall curvature of bundles will make the interface more visually appealing and user friendly.

4.5 Discussion

To evaluate the potential benefits and drawbacks of edge bundling (CogZ-e) in the context of ontology mapping visualization compared to a baseline system (CogZ), a user study was conducted in a controlled setting. In addition, the interface design decisions incorporated within CogZ-e were examined. This study was also designed to minimize any potential effects of bias by varying the order of the interface exposure along with the order of the test ontology datasets to the participants.

For the given tasks, a context was provided to participants to ensure their active engagement within the process and to mimic the actual conditions as much as possible. Minimum relevant background was ensured by recruiting participants from the computer science, engineering, and business administration departments of the university. In addition, to further ensure that every participant has a baseline understanding of the basic concepts associated with the study as well as the overall problem domain, a brief training session was conducted prior to beginning of the study.

It was anticipated that participants would be familiar with the domain areas used for the study i.e., university and academic conference, since they belong to the community of students at a senior level and they frequently interact with these areas directly or indirectly in their daily life. As expected, the level of familiarity as asked in the task questionnaire among participants with these domain areas was found to be equivalent. Therefore, the familiarity with the domains of ontology datasets was expected to have no significant negative or positive effect on the outcomes of the study.

To compare different aspects of the two interfaces while performing the two fun-

Table 4.12: Summary of the validation results of the hypotheses for both type of ontology alignment tasks.

Results CogZ-e (with bundling) Vs. CogZ (without bundling)		
No.	**Hypotheses**	**Final results**
H1.1	Users will take less time with CogZ than CogZ-e.	Rejected*
H1.2	In terms of accuracy, there will not be any significant difference between CogZ and CogZ-e.	Accepted
H1.3	The perceived level of difficulty of a given task will be more in CogZ than CogZ-e.	Mixed
H1.4	Users will be less confident in their results after using CogZ than CogZ-e.	Rejected
H1.5	Users will report that CogZ is less useful than CogZ-e.	Accepted
H1.6	Users will report that CogZ is similar in ease of use to CogZ-e.	Rejected*
H1.7	Users will prefer CogZ-e over CogZ.	Accepted
H2.1	Users will take less time with CogZ-e than CogZ.	Rejected
H2.2	In terms of accuracy, users will be more accurate with CogZ-e than CogZ.	Accepted
H2.3	The perceived level of difficulty of a given task will be less in CogZ-e than CogZ.	Rejected
H2.4	Users will be more confident in their results after using CogZ-e than CogZ.	Rejected
H2.5	Users will report that CogZ-e is more useful than CogZ.	Accepted
H2.6	Users will report that CogZ-e is similar in ease of use to CogZ.	Rejected*
H2.7	Users will prefer CogZ-e over CogZ.	Accepted
	*The outcome is better than expected	

damental tasks of ontology alignment, the time to task completion and accuracy of the decisions was measured, as well as the subjective measures about the perceived difficulty of tasks, participants' confidence over their results, perceived usefulness and ease of use were also recorded. The summary of the results of the hypotheses for both types of tasks is available in Table 4.12.

The analysis of time to task completion for mapping validation tasks provided better results than expected for CogZ-e, implying that the interaction overhead compared to CogZ was actually beneficial and improved participants' time efficiency for task completion. Using bundling, the facility of being able to view and analyze the subsets of related mappings easily and readily along with the option of validating multiple mappings at once, participants were able to make valid decisions with speed. The analysis also indicates that the edge bundling allowed participants to complete their tasks faster with the same 100% accuracy as non-bundling. These findings regarding the accuracy were also expected since the core data that is required to make the validation decisions is generated using the same automatic alignment algorithms and hence is the same in both systems.

Through open-ended feedback, it was also confirmed that the participants were able to complete validation tasks more quickly with more accuracy with CogZ-e was mainly because often related mappings were grouped together in a bundle that helped them in making a decision with ease, and the multiple mapping validation option. This feature had a clear advantage over CogZ, in which the user has to select each and every mapping to confirm or reject it without much help in terms of organization of the mapping results.

As predicted, the time to task completion for the mapping addition tasks was similar with both interfaces. The majority of the feedback reflected that CogZ-e provided an overview and a better organization to mapping results compared to CogZ. In addition, participants appreciated the effort of discouraging the following of the traditional top-down or random search approach to look for potential concepts in CogZ-e compared to CogZ, but the requirement of the additional interaction required to disambiguate the mapping information consumed more time. Therefore, despite providing a better mapping representation and organization, the extra interaction led to similar results with regards to the time efficiency of both systems for the mapping addition tasks. It can be inferred that for mapping addition tasks the benefits versus the drawbacks were balanced out in terms of time taken with both the interfaces. As such, superiority of any interface could not be determined. In terms of accuracy the outcome that CogZ-e performed better than CogZ for the addition tasks, was consistent with the expectation. In CogZ-e, participants reported that they were able to easily focus on the interesting areas (that are most likely to contain the potential concepts for addition) as they were automatically made prominent by bundling. This could be the reason participants made more appropriate and accurate mapping addition decisions with CogZ-e compared to CogZ.

It was observed that most of the participants were having difficulty in dealing with the bundling interaction required for the addition tasks. This interaction was required to first view current mappings present within a bundle, and then to see the source and destination concepts of a mapping. Although, it seems unnecessary to have the participant go through this interaction for addition tasks, it was required to keep the interface consistent in terms of the visual representation and interaction methods provided to conduct to both kinds of tasks. Changing this condition only for the addition task, could introduce other variables in the study, which can affect actual examined conditions or variables, making it difficult to draw any definite conclusions from the study results.

Another important aspect related to mapping validation task is the perceived difficulty associated with this task. Even though the results of the perceived difficulty are slightly more positive for CogZ-e implying that it offers modest advantage in terms of reducing perceived difficulty, they were not statistically significant. Similarly, for the perceived confidence on decisions taken for completing the given tasks, no improvement by using CogZ-e was verified based on statistical significance for validation tasks. The plausible reason for this is that since the core data (i.e. the candidate mappings computed by ontology alignment algorithm) is the same in both systems, there is a small chance of validating an incorrect candidate mapping, leading to no significant difference in terms of perceived confidence on results between the two systems.

As far as the confidence on the perceived difficulty and correctness of actions performed in a task is concerned, there was no significant difference for the addition tasks. Participants' feedback was slightly positive for both measures towards CogZ-e, however, these responses were not statistically significant. This result was contrary to what was expected. Since CogZ-e highlights prominent areas within the existing mapping set, it was anticipated that this will result in a reduction in the perceived difficulty associated with the addition task by default, and would increase participants' confidence on their mapping addition decisions. However, due to the interaction overhead with CogZ-e outweighed the benefits of structure provided by bundling and led to these results. Consequently, the relevant hypotheses could not be validated.

The comparative responses about the perceived usefulness and ease of use regarding the two systems, revealed some interesting results. Since at this stage participants have worked with both interfaces and performed all the tasks with both ontology sets, these results present a more accurate illustration of their experience during the study. As anticipated, due to overall benefits of bundling for both mapping validation and addition tasks, participants reported that they found CogZ-e more useful for conducting these two fundamental ontology alignment tasks compared to the baseline system. This implies that the provided structure to the validation tasks, and the accentuation of potential areas to find more appropriate mappings for the addition tasks, helped participants in conducting these tasks and increased the overall usability of the system. Participants reported that they found CogZ-e easier to use compared to CogZ, which is better than the expected result. It was evident that as participants become more familiar with bundling representation and interaction options, they were able to use them with ease for both types of tasks. This resulted in their positive feedback regarding the ease of use measure for the CogZ-e system compared to the baseline system. Although, there was some criticism regarding the interaction overhead posed by bundling, overall similar positive responses were reported for the mapping addition tasks as well. Participants appreciated the grouped information that was made more prominent in CogZ-e, which helped them in finding potential concepts to create a mapping with more ease. There was also a general consensus on the fact that once they get more accustomed with CogZ-e, the usefulness and ease of use measures was further increased with time for both types of tasks. In addition, the overall positive benefits provided by the proposed system for both types of tasks, the majority (92% for mapping verification tasks, and 83% for mapping addition tasks) of participants reported that they would prefer CogZ-e over CogZ. The results about this preference were also statistically significant, and consequently, the related hypotheses were supported.

By analyzing the overall results of this user study, it can be concluded that these results provide empirical evidence regarding the positive impacts of CogZ-e for the mapping validation and addition tasks. More importantly, it validates the fundamental assumption regarding the value of edge bundling to reduce visual clutter on the screen in the domain of ontology mapping representation. The results confirm that participants were able to interpret the relationships between the mappings easily when, through the edge bundling process, the mapping edges were placed near one another and drawn similar to each other. These findings validate the Gestalt laws of *proximity* and *similarity*, which predict that when relationships are enforced using proximity and similarity, they are automatically perceived by humans. In addition to understanding these relationships automatically, participants were able to also use them effectively, which resulted in performance gain in terms of speed for the mapping validation tasks, and increased accuracy for the mapping additions tasks. These results also confirms the benefit of organizing information as proposed by the categorization theory. Through edge bundling, the mapping information was categorized and grouped, which helped the participants in processing the information and making decisions more effectively and easily. Similarly, by processing the clusters of edges instead of individual edges, the effective management of the participants' limited working memory was possible, which reduced the cognitive load on participants, which helped them in making decisions more quickly and accurately. In addition to the positive findings with respect to efficiency and accuracy, the results were also favourable regarding the perceived usefulness, perceived ease of use and preference for the bundling interface for both types of ontology alignment tasks. These results validate the support provided by different theories for the value of edge bundling and the notion of using bundling for improving the visual representation of mappings on ontology alignment interfaces.

Even though the results of the conducted study are promising, and the majority of participants indicated their preference for CogZ-e over CogZ for both types of alignment tasks, real-world use may result in even better or worse or the same results. Since the study was carried out in a laboratory setting, this kind of controlled environment limits the ability of participants to get familiar with the system being examined. As such, participants had limited opportunity to become more skillful with the edge bundling given the limited time and other constraints. In a real-world scenario, actual users may have no such constraints, and they can take as much time as they need to become more acquainted with the proposed system. They can perform realistic tasks according to their professional requirements, which can provide more valuable insight into the performance and effectiveness of the interface. Therefore, this evaluation can be considered a preliminary study. A more detailed study with different types and complexity levels of ontology alignment tasks with bigger test ontology datasets, along with actual knowledge managers is required, to deduce more strong and generalizable conclusions.

Chapter 5

Conclusions and Future Work

The primary objective of this research has been to enhance the knowledge manager's experience and performance while conducting the two fundamental ontology alignment tasks (i.e., the mapping validation and mapping addition task). In terms of visual support provided by the current ontology alignment systems for conducting these tasks, the main issue is the presence of significant visual clutter on the interface due to the crossings of mapping edges. In the presence of this visual clutter, the complexity of conducting ontology alignment related tasks increases, and completing these complex tasks becomes harder for knowledge managers [67].

With the goal of reducing visual clutter on the interface of the ontology alignment system, an edge bundling technique from the domain of graph visualization was selected, modified, and implemented to improve the representation and organization of ontology mapping edges (Chapter 3). Several new interaction features were also developed for knowledge managers to view the mappings present within the bundles, and simultaneously conduct multiple mapping validations. At the end, a user study was conducted to address the research question regarding the benefits and drawbacks of using the edge bundling for the domain of ontology mapping representation, and in the context of ontology alignment tasks (Chapter 4). The primary contributions of this research and the potential future work directions are discussed in the remainder of this Chapter.

5.1 Research Contributions

The primary contributions of this work are the application of edge bundling within an ontology alignment interface and its evaluation in the context of supporting the core ontology alignment tasks. Both of these contributions are explained in the following sections.

5.1.1 Bundling Algorithm Modifications and Interaction Support

Through the study of commonly used ontology alignment systems, it was identified that effectively managing the display of a large number of ontology mapping edges on the limited space available is considered an important issue [67]. A poor and disorganized representation of mappings can result in a significant degree of visual clutter, which increases the complexity and the time required to complete the tasks, resulting in increased difficulty for knowledge managers who have to repeatedly perform such ontology alignment tasks using these interfaces for knowledge management [67]. In this direction and as one of the the primary contribution of this research, edge bundling was applied within an ontology alignment interface to address the issue of visual clutter. The edge bundling reduces mapping edge crossings by first grouping together individual edges into bundles, which has the effect of reducing the perception of edge crossings (since multiple edges within a bundle will produce a single edge crossing).

As the edge bundling is primarily a graph-based information visualization method, the fundamental research question was, *how can the edge bundling technique be adopted to the domain of ontology mapping representation?* To address this question, the edge bundling technique was modified and implemented in two parts. The first part contained the changes that are required to meet the constraints of the ontology mapping representation domain, and the second part provides the interaction support that is required to perform the mapping validation and addition tasks with bundles.

The first part was implemented in three steps: first, instead of incorporating the entire mapping edge as input for the segmentation, only the middle part of the edge was considered. In other words, the division of edge parts was limited to the 75% of the length of the edge. Second, if any of the edge's source or destination concepts are not visible at present, that edge is not shown, is flagged as ineligible and therefore, is excluded from being a candidate for any bundle. Finally, due to the limited scope of mapping representation within an ontology alignment interface, the only compatibility measure (provided by the original algorithm) that was used in the proposed system was the position compatibility measure that majorly affected the results of the bundling.

Once the mapping information is grouped together by edge bundling and the overview has been observed by knowledge managers to determine an appropriate starting point to conduct the relevant alignment task, the mapping information needs to be disambiguated and mappings are required to be viewed individually. Therefore, for the second part, interaction features were provided to disambiguate the mapping information contained in bundles, as well as to conduct the ontology alignment tasks. First, the tool-tip feature of the baseline system was extended to work with bundles, providing information about the mappings contained in the bundle. This tool-tip helps knowledge managers to quickly view the mappings inside the bundles and deciding which bundle they want to start validating first. Second, once the knowledge managers decide to validate the mappings present in a particular bundle, a pop-up window was provided with the mapping details and different validation options. This pop-up aids in clearly viewing the multiple mappings present in the bundle with the option of validating or rejecting multiple mappings at the same time. While the pop-up window is still visible, knowledge managers may need to view the location of a mapping listed in the pop-window. In order to meet this requirement, the selected mapping in the pop-up window was highlighted to help the knowledge manager to quickly identify the location of the mapping and make a more informed decision. Depending upon the current state, the selected mapping was highlighted using different colours that were chosen by following the Opponent Process Theory of Colour [57] that helps in selection of such colours that can aid the knowledge manager in understanding the different states of the mappings.

Different information visualization theories and principles along with cognitive theories support the value of edge bundling. The Gestalt Laws support the fundamental idea that is used by edge bundling that by placing the mapping edges near one another and drawing them similar to each other, it will help knowledge managers in automatically perceiving the relationships between mappings. The overview of the ontology mappings provided by edge bundling through categorization reduces the amount of information to be processed and reinforces the existence of the relationships between mappings present within a bundle, which may be advantageous as elaborated by the categorization theory. In addition, as supported by the cognitive load theory, this organization also allows efficient use of the knowledge manager's cognitive memory, which may lead to performance gains in terms of conducting the ontology alignment tasks. Edge bundling also aids in identifying interesting areas within the source ontology from where a large number of mappings originate by simply observing the bundles. Following the bundles, it can be easily deduced which sets of concepts in the source ontology have successfully found their counter-parts in the target ontology. Knowledge managers can easily locate these interesting areas, which may help in discovering new mappings as well as finding a match for the unmapped concepts. In addition, once the concepts in the bundles are validated, these can be ignored and knowledge managers can focus on the mappings that appear to be linked to vastly different areas within the ontologies.

5.1.2 User Study Findings

After the modification of edge bundling to meet the constraints of ontology mapping representation, the second research question that emerged was, *what are the benefits and drawbacks of using edge bundling in the context of ontology mapping validation and addition tasks?* In order to address this question, a user study was conducted to compare the performance of the ontology alignment system with edge bundling (CogZ-e) with the one without the edge bundling capability (CogZ) in a controlled laboratory environment.

The outcomes of the study reflect different aspects of participants' performance experiences while conducting the given tasks. The results of this study show that even though it was anticipated that participants will take more time with bundling while conducting validation tasks, instead they performed much better with bundling. The reason for this improvement was the ability to view related mappings via the bundles, and the ability to validate multiple mappings at once. For the mapping addition tasks, the expected outcome regarding the time taken could not be supported since participants took similar times with both interfaces. As explained in Section 4.4.2, participants were having difficulty interacting with bundles for the addition tasks. Although there was no such issue in the case of CogZ, no relevant aid was provided for conducting addition tasks. Therefore, this resulted in canceling out any performance gained due to the different benefits provided by both systems against their respective limitations.

By studying the outcomes of this study, it can be concluded that the bundling process has the potential of improving the knowledge manager's capability to conduct mapping validation tasks more efficiently and mapping addition tasks more accurately. It also affects positively knowledge managers' subjective opinions about the usefulness and ease of use of using edge bundling capable interfaces for ontology alignment systems for conducting both types of ontology alignment tasks. Although the corresponding hypotheses were rejected, these results are better than expected confirming that participants performed better than expected with the edge bundling system. These positive findings highlight the value of showing the high-level structure of the mappings via the bundles, and simultaneously reducing the visual clutter within the interface. Through the visual perception of similarity among the edges and bundles, impressions of relationships were implied. This in turn supported the cognitive activity of organizing the mappings, providing guidance to the tasks and reduced the cognitive load. Although such bundling has the side effect of making it difficult to identify the specific source and destination concepts within a bundle, this problem was mitigated through an interactive disambiguation feature. Overall, these results support the fundamental hypothesis regarding the value of edge bundling in the context of ontology mapping representation and the ontology alignment tasks.

5.2 Future Work

5.2.1 Further Improvement of the Proposed System

The results of the user study suggest that participants found the edge bundling useful and easier to use for both types of tasks. However, the improvement in terms of time taken to complete the addition tasks could not be supported mainly due to the extra disambiguation steps required to see the mappings inside the bundle. Certain improvements in terms of interaction could be done to improve this situation. For example, the requirement of keeping the Alt key pressed to view the extended tool-tip can be removed, and when participants want to work on the bundle, only then they press the Alt key to see the pop-up window containing the mappings inside the bundle. This will reduce the amount of interaction required not only for the mapping addition tasks but also for the validation tasks.

Similarly, other interaction features can be developed for mapping addition tasks to further improve the knowledge manager's performance. For example, based on the bundling results, potential areas for locating a match for a selected unmapped concept can be made more prominent by increasing the transparency of other parts of the source and target ontology trees. In addition to using the transparency technique, these areas can be further highlighted by using different colours. In addition, other visual channels such as line shape, blackened borders etc. can be used to for disambiguating mapping edges. Similarly, colour encodings like hierarchical parallel coordinates can be used to help resolving mapping information that is hidden due to the bundling process. This may allow the knowledge manager to process this information easily, and find the relevant match for the selected unmapped concept more quickly.

In terms of using consistent colours, it was a viable option to use the same colour for the tooltip as well as on the interface to represent confirmed mappings. In addition, usage of the blue colour will further improve the consistency of the current approach with its predecessor. Since these tooltips are shown over the area where all kinds of mappings with different colours are shown including the confirmed mappings in the blue colour, the idea was to use a different colour to reflect this fact in the tooltip. Therefore, the bold font with red colour in the tooltip helps in clearly identifying the confirmed mappings from the candidate mappings.

However, if the blue colour was also used to show detail about the confirmed mappings in the tooltip, there was a chance that the user might have an increased difficulty in seeing and distinguishing blue over blue compared to seeing red over blue. However, for future research, based on the colour theory, different colours can be used to represent different entities in the approach, and their advantages can be evaluated by performing a more detailed user study with prime focus on the usage of color on the performance of ontology alignment tasks of knowledge managers.

5.2.2 Study of Different Factors Affecting the Bundling Quality

Since the test ontology datasets used for this study were similar in terms of characteristics i.e. (the number of concepts, hierarchical depth, and complexity), it could be beneficial to also test the edge bundling using ontologies with different characteristics. Furthermore, studying how different sizes of ontologies and different sizes of mapping sets (produced after an alignment process) affect the overall bundling quality, and then evaluating the potential benefits and drawbacks of using edge bundling for these ontologies, can be advantageous. Examining these parameters will provide further insight into the value of the edge bundling in the context of ontology mapping representation and how it affects the performance of the knowledge managers while conducting the ontology alignment tasks.

As discussed in Section 3.2.2, the only edge compatibility applicable and used for the purposes of this research was the position compatibility measure. The other available compatibility measures that were not used for bundling were specific to graphs and provided little effect on the overall bundling results for ontology mapping representation domain. Since none of these compatibility measures consider the properties of the source and target ontologies, the exploration and development of other edge compatibility measures that take into account features of the ontologies themselves such as the semantic distance between concepts within the source and destination ontologies can be helpful.

The edge bundling process was applied to the default order of source and target ontologies provided by the baseline system without any alteration. Since the order of concepts within the ontologies effects the location of the mapping edges (which effects the bundling results), this research dimension can also be explored and studied to find any potential ordering within the source and target ontologies that can improve the bundling results. By using and developing edge compatibility measures that consider the features and properties of the source and target ontologies along with the studying of the effects of entity ordering within the ontologies on the bundling quality, a hybrid approach can be developed that combines ontology organization with edge bundling that may produce further better results.

5.2.3 Further Evaluations

For this research, the evaluation of the CogZ-e system was conducted in a controlled laboratory environment. These laboratory studies are useful for researchers to control the influence of other non-relevant variables on the study, allows focusing on evaluating specific features of the system being examined, and facilitating a more accurate and direct comparison [11]. However, the outcomes of subjective measures as a result of performing such studies are obtained through a small number of participants and pre-defined tasks, which limits its scope and ability to conclude more generalizable results. Therefore, other evaluation methods may be useful to gain insight into the potential benefits and drawbacks of the system on other interesting cognitive measures e.g, learning and cognitive load of the knowledge managers. Furthermore, the study can be extended to evaluate the different aspects of the system, and by also manipulating other important variables e.g., different sizes of ontologies, different sizes of mapping sets, and different orders of the ontology trees.

Furthermore, the CogZ-e system can be tested in real world scenarios through field trials. In this way, the system can be evaluated by actual knowledge managers performing real tasks with real ontology datasets. The outcomes of such study may provide more firm evidence regarding the specific and overall performance of the system and knowledge managers conducting the alignment tasks using the system, which can be used to bring further improvements to the system.

Bibliography

[1] D. Aumueller, H.-H. Do, S. Massmann, and E. Rahm. Schema and ontology matching with COMA++. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 906–908. ACM, 2005.

[2] P. A. Bernstein and S. Melnik. Model management 2.0: manipulating richer mappings. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 1–12. ACM, 2007.

[3] P. A. Bernstein, S. Melnik, and J. E. Churchill. Incremental schema matching. In *Proceedings of the International Conference on Very Large Data Bases*, pages 1167–1170. VLDB Endowment, 2006.

[4] S. Boslaugh and P. Watters. *Statistics in a Nutshell: A Desktop Quick Reference.* O'Reilly Media, 2008.

[5] I. Boyandin, E. Bertini, and D. Lalanne. Jflowmap. `http://boyandi.net/works/2010/10/01/jflowmap/`, last viewed on 02 February 2011.

[6] T. Bray, J. Paoli, C. Sperberg-McQueen, E. Maler, and F. Yergeau. Extensible markup language (XML). *World Wide Web Journal*, 2(4):27–66, 1997.

[7] D. Brickley and R. Guha. Resource description framework (RDF) schema specification. `http://www.w3.org/TR/PR-rdf-schema`, 1999.

[8] C. M. Brown. *Human-Computer Interface Design Guidelines.* Intellect Books, 1998.

[9] Z. Cao, K. Li, and Y. Liu. Putting feedback into incremental schema matching. In *Proceedings of the WRI World Congress on Software Engineering*, pages 332–336. IEEE Computer Society, 2009.

[10] S. K. Card, J. D. Mackinlay, and B. Shneiderman. *Readings In Information Visualization: Using Vision To Think.* Morgan Kaufmann, San Francisco, 1999.

[11] S. Carpendale. Evaluating information visualizations. In A. Kerren, J. T. Stasko, J.-D. Fekete, and C. North, editors, *Information Visualization*, volume 4950 of *Lecture Notes in Computer Science*, pages 19–45. Springer Berlin / Heidelberg, 2008.

[12] P. Chandler and J. Sweller. Cognitive load theory and the format of instruction. *Cognition and Instruction*, 8(4):293–332, 1991.

[13] P. Chandler and J. Sweller. The split-attention effect as a factor in the design of instruction. *British Journal of Educational Psychology*, 62(2):233–246, 1992.

[14] D. Connolly. Web-Ontology working group. `http://www.w3.org/2001/sw/WebOnt/`, last viewed on 02 February 2011.

[15] I. Cruz. AgreementMaker. `http://agreementmaker.org/wiki/index.php/`, last viewed on 02 December 2012.

[16] I. Cruz, W. Sunna, N. Makar, and S. Bathala. A visual tool for ontology alignment to enable geospatial interoperability. *Journal of Visual Languages & Computing*, 18(3):230–254, 2007.

[17] F. D. Davis. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3):319–340, 1989.

[18] J. De Bruijn. Using ontologies: enabling knowledge sharing and reuse on the semantic web. Technical Report DERI-2003-10-29, Digital Enterprise Research Institute (DERI), Austria, 2003.

[19] A. Dix and G. Ellis. Starting simple: adding value to static visualisation through simple interaction. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, pages 124–134. ACM, 1998.

[20] H.-H. Do and E. Rahm. COMA: a system for flexible combination of schema matching approaches. In *Proceedings of the International Conference on Very Large Data Bases*, pages 610–621, 2002.

[21] S. Duan, A. Fokoue, and K. Srinivas. One size does not fit all: Customizing ontology alignment using user feedback. In *Proceedings of the International Semantic Web Conference*, pages 177–192. Springer, 2010.

[22] P. Eades. A heuristic for graph drawing. *Congressus Numerantium*, 42:149–160, 1984.

[23] M. Ehrig and S. Staab. QOM - quick ontology mapping. In *Proceedings of the International Semantic Web Conference*, pages 683–697. Springer, 2004.

[24] P. Eklund, N. Roberts, and S. Green. OntoRama: Browsing RDF ontologies using a hyperbolic-style browser. In *Proceedings of the First International Symposium on Cyber Worlds*, pages 405–411. IEEE Computer Society, 2002.

[25] G. Ellis and A. Dix. A taxonomy of clutter reduction for information visualisation. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1216–1223, 2007.

[26] J. Euzenat. Ontology alignment evaluation initiative. http://oaei.ontologymatching.org, last viewed on 28 February 2013.

[27] J. Euzenat, D. Loup, M. Touzani, and P. Valtchev. Ontology alignment with OLA. In *Proceedings of the EON Workshop in the International Semantic Web Conference*, volume 128, pages 59–68. CEUR-WS, 2004.

[28] S. M. Falconer and N. F. Noy. Interactive techniques to support ontology matching. In Z. Bellahsene, A. Bonifati, and E. Rahm, editors, *Schema Matching and Mapping*, pages 29–51. Springer, Berlin, 2011.

[29] S. M. Falconer and M.-A. Storey. A cognitive support framework for ontology mapping. In *Proceedings of the International Semantic Web Conference*, pages 114–127, 2007.

[30] S. M. Falconer and M.-A. Storey. Ontology mapping - a user survey. In *Proceedings of the International Workshop on Ontology Matching*, pages 49–60, 2007.

[31] S. M. Falconer and M.-A. Storey. CogZ: cognitive support and visualization for semi-automatic ontology mapping. In *Proceedings of the International Conference on Biomedical Ontology*, pages 1–2, 2009.

[32] J.-D. Fekete and C. Plaisant. Interactive information visualization of a million items. In *Proceedings of IEEE Symposium on Information Visualization*, pages 117–124. IEEE, 2002.

[33] C. Fellbaum. Wordnet. In R. Poli, M. Healy, and A. Kameas, editors, *Theory and Applications of Ontology: Computer Applications*, pages 231–243. Springer Netherlands, 2010.

[34] G. W. Furnas. Generalized fisheye views. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 16–23. ACM, 1986.

[35] G. W. Furnas and X. Zhang. MuSE: a multiscale editor. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*, pages 107–116. ACM Press, 1998.

[36] F. Giunchiglia, P. Shvaiko, and M. Yatskevich. S-match: an algorithm and an implementation of semantic matching. In C. J. Bussler, J. Davies, D. Fensel, and R. Studer, editors, *The Semantic Web: Research and Applications*, volume 3053 of *Lecture Notes in Computer Science*, pages 61–75. Springer Berlin / Heidelberg, 2004.

[37] M. Granitzer, V. Sabol, K. W. Onn, D. Lukose, and K. Tochtermann. Ontology alignment - a survey with focus on visually supported semi-automatic techniques. *Future Internet*, 2(3):238–258, 2010.

[38] D. A. Grant. The Latin square principle in the design and analysis of psychological experiments. *Psychological Bulletin*, 45(5):427–442, 1948.

[39] T. Gruber. Ontology. In L. Liu and T. M. Özsu, editors, *Encyclopedia of Database Systems*, pages 1963–1965. Springer-Verlag, 2009.

[40] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[41] N. Guarino. Formal ontology and information systems. In *Proceedings of the International Conference on Formal Ontology in Information Systems*, pages 3–15, 1998.

[42] D. H. Hai. *Schema Matching and Mapping-based Data Integration: Architecture, Approaches and Evaluation.* VDM Verlag, Saarbrücken, Germany, 2007.

[43] J. Hendler. OWL Web ontology language. http://www.w3.org/TR/owl-features/, last viewed on 02 February 2011.

[44] J. Hendler. RDF/XML syntax specification. `http://www.w3.org/TR/REC-rdf-syntax/`, last viewed on 02 February 2011.

[45] D. Holten and J. J. van Wijk. Force-directed edge bundling for graph visualization. *Computer Graphics Forum*, 28(3):983–990, 2009.

[46] M. Horridge. Owlviz - a visualisation plugin for the protégé owl plugin. `http://www.co-ode.orgldownloads/owlvizl`, 2005.

[47] R. Jaaskelainen. Think-aloud protocol. In Y. Gambier and L. v. Doorslaer, editors, *Handbook of Translation Studies*, volume 1, pages 371–373. John Benjamins Publishing, 2010.

[48] A. Katifori, C. Halatsis, G. Lepouras, C. Vassilakis, and E. Giannopoulou. Ontology visualization methods - a survey. *ACM Computing Surveys*, 39(4):10:1–10:43, 2007.

[49] T. Khazaei and O. Hoeber. Metadata visualization of scholarly search results: Supporting exploration and discovery. In *Proceedings of the International Conference on Knowledge Management and Knowledge Technologies*, pages 21:1–21:8. ACM, 2012.

[50] W. Köhler. Gestalt psychology. *Psychological Research*, 31(1):18–30, 1967.

[51] B. Kules and R. Capra. Designing exploratory search tasks for user studies of information seeking support systems. In *Proceedings of the ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 419–420, 2009.

[52] G. Lakoff. *Women, Fire, and Dangerous Things.* University Of Chicago Press, 1990.

[53] M. Lanzenberger, J. Sampson, and M. Rester. Visualization in ontology tools. In *Proceedings of International Conference on Complex, Intelligent and Software Intensive Systems*, pages 705–711, 2009.

[54] J. Li, J. Tang, Y. Li, and Q. Luo. RiMOM: A dynamic multistrategy ontology alignment framework. *IEEE Transactions on Knowledge and Data Engineering*, 21(8):1218–1232, 2009.

[55] J. Madhavan, P. A. Bernstein, and E. Rahm. Generic schema matching with cupid. In *Proceedings of the International Conference on Very Large Data Bases*, volume 1, pages 49–58, 2001.

[56] S. Melnik, H. Garcia-Molina, and E. Rahm. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In *Proceedings of the International Conference on Data Engineering*, pages 117–128, 2002.

[57] K. Mullen. The contrast sensitivity of human color vision to red-green and blue-yellow chromatic gratings. *American Journal of Optometry and Physiological Optics*, 359(1):381–400, 1985.

[58] M. Nasir, O. Hoeber, and J. Evermann. Supporting ontology alignment tasks with edge bundling. In *Proceedings of the International Conference on Knowledge Management and Knowledge Technologies*, (to appear in 2013).

[59] N. F. Noy. Ontology management with the PROMPT plugin. In *Proceedings of International Protégé Conference*, pages 1–4, 2004.

[60] N. F. Noy, R. W. Fergerson, and M. A. Musen. The knowledge model of Protégé-2000: combining interoperability and flexibility. In R. Dieng and O. Corby, editors, *Knowledge Engineering and Knowledge Management Methods, Models, and Tools*, volume 1937 of *Lecture Notes in Computer Science*, pages 17–32. Springer Berlin / Heidelberg, 2000.

[61] N. F. Noy and M. A. Musen. Anchor-PROMPT: Using non-local context for semantic matching. In *Proceedings of the Workshop on Ontology and Information Sharing at the International Joint Conference on Artificial Intelligence*, pages 63–70, 2001.

[62] N. F. Noy and M. A. Musen. The PROMPT suite: Interactive tools for ontology merging and mapping. *International Journal of Human-Computer Studies*, 59(6):983–1024, 2003.

[63] J. Parsons. An information model based on classification theory. *Management Science*, 42(10):1437–1453, 1996.

[64] C. Plaisant, J. Grosjean, and B. B. Bederson. SpaceTree: Supporting exploration in large node link tree, design evolution and empirical evaluation. In *Proceedings of IEEE Symposium on Information Visualization*, pages 57–64, 2002.

[65] L. M. Quiroga, M. E. Crosby, and M. K. Iding. Reducing cognitive load. In *Proceedings of the International Conference on System Sciences*, pages 1–9, 2004.

[66] E. Rahm and P. A. Bernstein. A survey of approaches to automatic schema matching. *The VLDB Journal*, 10(4):334–350, 2001.

[67] G. G. Robertson, M. P. Czerwinski, and J. E. Churchill. Visualization of mappings between schemas. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 431–439, 2005.

[68] E. Rosch and B. B. Lloyd. *Cognition and Categorization*. Lawrence Erlbaum, Hillsdale, NJ, 1978.

[69] R. Rosenholtz, Y. Li, and L. Nakano. Measuring visual clutter. *Journal of Vision*, 7(2):1–22, 2007.

[70] B.-A. Scharfstein. *A Comparative History of World Philosophy: From the Upanishads to Kant*. State University of New York Press, 1998.

[71] H. Scheffé. *The Analysis of Variance*. John Wiley and Sons Inc., New York, 1959.

[72] F. Shi, J. Li, J. Tang, G. Xie, and H. Li. Actively learning ontology matching via user interaction. In *Proceedings of the International Semantic Web Conference*, pages 585–600. Springer-Verlag, 2009.

[73] B. Shneiderman. Inventing discovery tools: Combining information visualization with data mining. In K. P. Jantke and A. Shinohara, editors, *Discovery Science*, volume 2226 of *Lecture Notes in Computer Science*, pages 17–28. Springer Berlin / Heidelberg, 2001.

[74] B. Shneiderman and C. Plaisant. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley, Boston, 5th edition, 1997.

[75] M. Sintek. OntoViz tab: Visualizing Protégé ontologies. `http://protegewiki.stanford.edu/wiki/OntoViz`, last viewed on 02 February 2011.

[76] M. C. Stone. Representing colors as three numbers. *IEEE Computer Graphics and Applications*, 25(4):78–85, 2005.

[77] Y. Sure, J. Angele, and S. Staab. OntoEdit: Guiding ontology development by methodology and inferencing. In R. Meersman and Z. Tari, editors, *On the Move to Meaningful Internet Systems: CoopIS, DOA, and ODBASE*, volume 2519 of *Lecture Notes in Computer Science*, pages 1205–1222. Springer Berlin / Heidelberg, 2002.

[78] J. Sweller. Cognitive load during problem solving: effects on learning. *Cognitive Science*, 12(2):257–285, 1988.

[79] The W3C Group. World Wide Web consortium. `http://www.w3.org/`, last viewed on 28 February 2013.

[80] M. Uschold and M. Gruninger. Ontologies: principles, methods and applications. *The Knowledge Engineering Review*, 11:93–136, 1996.

[81] R. L. Walls, B. Athreya, L. Cooper, J. Elser, M. A. Gandolfo, P. Jaiswal, C. J. Mungall, J. Preece, S. Rensing, B. Smith, et al. Ontologies as integrative tools for plant science. *American Journal of Botany*, 99(8):1263–1275, 2012.

[82] C. Ware. *Information Visualization: Perception For Design*. Morgan Kaufmann, San Francisco, 2nd edition, 2004.

[83] R. F. Woolson. Wilcoxon signed-rank test. In P. Armitage and T. Colton, editors, *Encyclopedia of Biostatistics*, volume 6, pages 4739–4740. John Wiley, New York, 1998.

[84] F. Yamaguchi and F. Yamaguchi. *Curves and Surfaces in Computer Aided Geometric Design*. Springer-Verlag Berlin, 1988.

[85] J. S. Yi, Y. A. Kang, J. Stasko, and J. Jacko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1224–31, 2007.

Appendix A

User Study

This appendix includes the formal approval received from the Interdisciplinary Committee on Ethics in Human Research (ICEHR) for the user evaluation, the extension of the approval, and the user study documents.

MEMORIAL UNIVERSITY

Interdisciplinary Committee on Ethics in Human Research (ICEHR)

Office of Research - IIC2010C
St. John's, NL Canada A1C 5S7
Tel: 709 864-2561 Fax: 709 864-4612
www.mun.ca/research

ICEHR Number:	**2012-318-SC**
Approval Period:	**March 5, 2012 – March 31, 2013**
Funding Source:	Pending
Responsible Faculty:	Dr. Orland Hoeber Department of Computer Science
Title of Project:	***A comparison of unbundled and bundled edges in the context of ontology alignment software***

March 5, 2012

Ms. Muhammad Nasir
Department of Computer Science
Memorial University of Newfoundland

Dear Mr. Nasir:

Thank you for your email correspondence of March 2, 2012 addressing the issues raised by the Interdisciplinary Committee on Ethics in Human Research (ICEHR) concerning the above-named research project.

The ICEHR has re-examined the proposal with the clarification and revisions submitted and is satisfied that concerns raised by the Committee have been adequately addressed. In accordance with the *Tri-Council Policy Statement on Ethical Conduct for Research Involving Humans (TCPS2)*, the project has been granted *full ethics clearance* for one year from the date of this letter.

If you intend to make changes during the course of the project which may give rise to ethical concerns, please forward a description of these changes to Mrs. Eleanor Butler at icehr@mun.ca for the Committee's consideration.

The TCPS2 requires that you submit an annual status report on your project to the ICEHR, should the research carry on beyond March 31, 2013. Also to comply with the TCPS2, please notify us upon completion on your project.

We wish you success with your research.

Yours sincerely,

M. Shute, Th.D.
Chair, Interdisciplinary Committee on
Ethics in Human Research

MS/eb

copy: Supervisor – Dr. O. Hoeber, Department of Computer Science

Office of Research Services, Bruneau Centre for Research & Innovation

Nasir, Muhammad

From: smmercer@mun.ca
Sent: April-11-13 2:48 PM
To: Nasir, Muhammad
Cc: Dr. Orland Hoeber (Supervisor); Evermann, Joerg; Mercer, Susan
Subject: ICEHR Clearance 2012-318-SC - EXTENDED

MEMORIAL UNIVERSITY

Interdisciplinary Committee on Ethics in Human Research (ICEHR)

Dear Mr. Nasir ,

Thank you for your response to our request for an annual status report advising that your project will continue without any changes that would affect ethical relations with human participants.

On behalf of the Chair of ICEHR, I wish to advise that the ethics clearance for this project has been extended to Renewal Due-2014/03/31. The *Tri-Council Policy Statement on Ethical Conduct for Research Involving Humans* (TCPS2) requires that you submit an annual status report to ICEHR on your project, should the research carry on beyond Renewal Due-2014/03/31. Also, to comply with the TCPS2, **please notify us upon completion of your project**.

ICEHR Ref. No.	2012-318-SC
Project Title:	*(2012-318-SC) A comparison of unbundled snd bundled edges in the context of ontology alignment software*
PI:	**Mr. Muhammad Nasir** **School of Graduate Studies**
Supervisor:	
Clearance expiry date:	Renewal Due-2014/03/31

We wish you well with the continuation of your research.

Sincerely,
Susan Mercer
Secretary, ICEHR

ORS reference only - 20121347

Informed Consent Form

Title: *A comparison of unbundled and bundled edges in the context of ontology alignment software*

Researcher: *Muhammad Nasir, Dept. of Computer Science, Memorial University of Newfoundland; muhammad.nasir@mun.ca*

You are invited to take part in a research project entitled "*A comparison of straight edges and edge bundling in the context of ontology alignment software*".

My name is Muhammad Nasir and I am a graduate student in Computer Science. As part of my Master's thesis, I am conducting research under the supervision of Dr. Hoeber and Dr. Evermann.

This form is part of the process of informed consent. It should give you the basic idea of what the research is about and what your participation will involve. It also describes your right to withdraw from the study at any time. In order to decide whether you wish to participate in this research study, you should understand enough about its risks and benefits to be able to make an informed decision. This is the informed consent process. Take time to read this carefully and to understand the information given to you. Please contact the researcher, *Muhammad Nasir,* if you have any questions about the study or for more information not included here before you consent.

It is entirely up to you to decide whether to take part in this research. If you choose not to take part in this research or if you decide to withdraw from the research once it has started, there will be no negative consequences for you, now or in the future.

The user study will help in analysing the value of visualization techniques for the ontology alignment domain. The primary objective of this study is to determine the benefits of the edge-bundling technique to ontology mapping tasks. Also it will help in making the studied technique better and more effective, by understanding the feedback you provide while conducting the study.

An ontology is a formal representation of knowledge about a domain. It is composed of sets of concepts and their relationships within a domain. Today, many ontologies overlap in content and different ontologies exist for similar domains. In order for such domains to interact effectively with each other, the ontologies representing them need to be aligned.

In this study, you will be asked to complete a training task as well as four mapping related tasks using two different ontology mapping software interfaces and two different pairs of ontologies. You will be given a description of the source and target ontologies and information about the mappings that you are to verify. You will then be asked to perform this verification task using the assigned interface, followed by an adding task where you can add additional mappings based on your understanding of the ontologies.

Your activities during the study will be video and audio recorded (over your shoulder, focusing on the computer screen, keyboard, and mouse), and you will be asked to explain what you are doing and what

Page **1** of **3**

you are thinking while performing the tasks. Three different sets of questionnaires will be administrated in this study: a pre-study questionnaire to ask about your ontology and knowledge representation experience, in-study questionnaires following each task, to assess your familiarity level with the domain of ontologies being used, to evaluate your confidence level of the results while performing each task, as well as your experience regarding the level of task difficulty, and a post-study questionnaire to capture your feelings and experiences with using the interfaces. In the end, a debriefing session will be conducted to ask about your general comments about the interfaces.

In a nutshell, you will be required to perform the training task, two mapping verification tasks, two mapping addition tasks, and answer pre-task, in-task, and post-task questionnaires. These activities will require approximately 60 minutes of your time, and will be conducted in the User Experience Lab in the Department of Computer Science at Memorial University.

You will be compensated $10 for participation in this study, regardless of your performance or ability to complete the tasks. Participation in this could be a great experience for you in terms of studying and observing research methods in practice. This might help you in improving your own research (at present or in the future). Given the importance of ontologies and ontology alignment research, the results of this study will provide empirical evidence regarding the benefits of using the edge-bundling techniques as a novel way of organization of the mappings results.

You can withdraw your participation in this study at any time, and your decision to participate in this study, and your subsequent involvement in it, will have absolutely no bearing on any other dealings you have with Mr. Nasir, Dr. Hoeber, or Dr. Evermann.

Knowledge of your identity is not required. You will not be asked to write your name or any identifying information on the research questionnaires. The video and audio recordings will be used for analysis purposes and the comments you make relevant to the assigned tasks or to the use of the software will be transcribed and identified only using a participant ID. The original raw data will only be accessed by the investigators (Mr. Nasir, Dr. Evermann, or Dr. Hoeber). All research materials will be held confidential by the Principle Investigator and kept in a secure on-campus location and on password-protected computers for a period of five years, after which it will be destroyed in accordance with University policy.

The proposal for this research has been reviewed by the Interdisciplinary Committee on Ethics in Human Research and found to be in compliance with Memorial University's ethics policy. If you have ethical concerns about the research (such as the way you have been treated or your rights as a participant), you may contact the Chairperson of the ICEHR at icehr@mun.ca or by telephone at 709-864-2861.

The results of the user study will be used for analysis and discussion in Mr. Nasir's M.Sc. thesis as well as in conference and/or journal publications. You are welcome to study the results after they have been published. You can obtain copies of the results in this study, upon completion, by contacting Dr. Hoeber, in care of the Department of Computer Science, Memorial University.

You are welcome to ask questions at any time during your participation in this research. If you would like more information about this study after its completion, please contact: *Muhammad Nasir at muhammad.nasir@mun.ca.*

Consent:

Your signature on this form means that:

- You have read the information about the research.
- You have been able to ask questions about this study.
- You are satisfied with the answers to all your questions.
- You understand what the study is about and what you will be doing.
- You understand that you are free to withdraw from the study at any time, without having to give a reason, and that doing so will not affect you now or in the future.
- You understand that should you choose to withdraw from the study, any data collected from you up to this point will be deleted/destroyed.

If you sign this form, you do not give up your legal rights and do not release the researchers from their professional responsibilities.

Your signature:

I have read and understood what this study is about and appreciate the risks and benefits. I have had adequate time to think about this and had the opportunity to ask questions and my questions have been answered.

☐ I agree to participate in the research project understanding the risks and contributions of my participation, that my participation is voluntary, and that I may end my participation at any time.

A copy of this Informed Consent Form has been given to me for my records.

Participant Name (please print legibly): ______________________________

Signature : ______________________________

Date : ______________________________

Researcher's Signature:

I have explained this study to the best of my ability. I invited questions and gave answers. I believe that the participant fully understands what is involved in being in the study, any potential risks of the study and that he or she has freely chosen to be in the study.

______________________________ ______________________________

Signature of Principal Investigator Date

Investigators:

Mr. Muhammad Nasir
M.Sc. Student
Department of Computer Science
Memorial University
muhammad.nasir@mun.ca

Dr. Orland Hoeber
Assistant Professor
Department of Computer Science
Memorial University
hoeber@mun.ca

Dr. Joerg Evermann
Associate Professor
Faculty of Business Administration
Memorial University
jevermann@mun.ca

Introduction to Ontologies and training material

Introduction:

Ontology is a formal representation of knowledge about a domain. It is composed of sets of concepts and their relationships within a domain. It defines the terms used to describe and represent an area of knowledge. They have become a popular way of sharing and re-using knowledge among academia and industry.

Example: University-domain Ontology

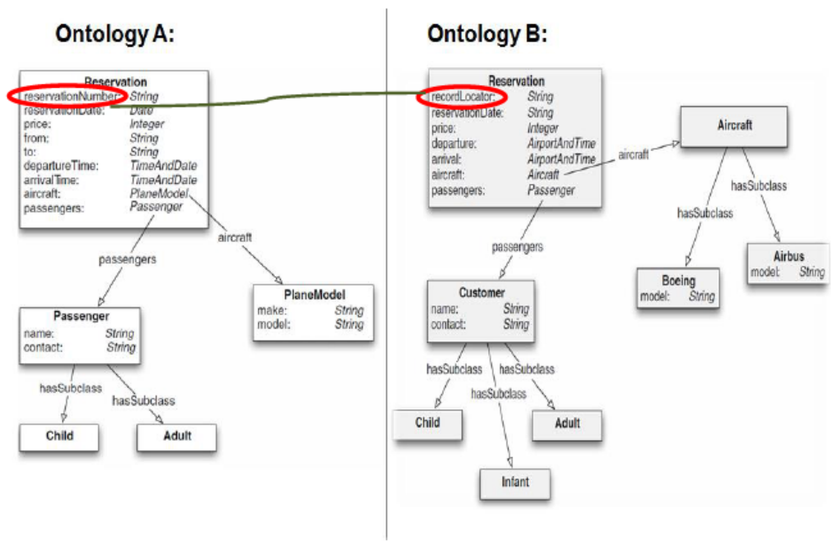
A university is an institution of higher education and research, which grants academic degrees in a variety of subjects. A university is a corporation that provides both undergraduate education and postgraduate education. The word university means "community of teachers and scholars." Members are usually students, staff, faculty members and others. One of the pictorial representations of university ontology is shown below.

Ontology Issues :

Many ontologies now-a-days over lap in content. Different ontologies exists for similar domains. The common issues are language ambiguities (same term but different meanings) etc. For two domains to interact with each other their underlying ontologies needs to aligned.

Ontology Alignment :

Ontology alignment is the process of determining correspondences between concepts among two different ontologies. A mapping or link is created when two concepts in two different ontologies, are found to be similar to each other.



Concept 'reservationNumber' in Ontology A (source) is similar to concept 'recordLocator' in Ontology B (target) therefore a mapping link can be created among them. The green curve shows the mapping link.

Commonly, ontology mappings are produced by an ontology alignment system at hand and the user has to verify them. Due to complex nature and significant size of current ontologies, these systems produce a large number of candidate mappings. This results into a visual mess and the user has to deal with. This study intends to examine the effects of using a visual technique edge-bundling in mapping visualization. Edge-bundling technique helps in visually organizing the mappings. It wraps up the mappings which share a part of their pathway between the source and target ontology, thus forming a bundle. To fully understand the nature of this study, the following sample tasks which will be performed prior to the actual study. This will aid in understanding the nature of the problem as well as will provide a realistic scenario as being faced by a data manager.

Training Ontology Domain = General things

Task Scenario 1 – Locate data relationships

You have been provided with a two different ontologies which relate to the cooking and recreational domain. Imagine you are an ontology merging expert and want to find which data elements in the source ontology are linked or related to another element in the target ontology.

Your tasks are as follows:

1. Find any three (3) data nodes in source ontology and their links to other elements in target ontology as well as their mapping relation by using the **CogZ** interface.
   For example: Data element in source ontology (O1) is "Internet" and it relates to another element "Internet" (via a line) in target ontology (O2) and the relation is equivalent then the mapping would be O1: Internet _=_ O2: Internet
   Identify those and indicate them to the evaluator.

___

___

___

In-Task Questionnaires

1. How confident are you in your finding of mapping links for this task?

1	2	3	4	5
(Very confident)		(Neutral)		(Very unconfident)

2. How difficult did you find it to complete this task?

1	2	3	4	5
(Very easy)		(Neutral)		(Very difficult)

Task Scenario 2 – Locate data relationships

You have been provided with a two different ontologies which relate to the cooking and recreational domain. Imagine you are an ontology merging expert and want to find which data elements in the source ontology are linked or related to another element in the target ontology.

Your tasks are as follows:

1. Find any three (3) data nodes in source ontology and their links to other elements in target ontology as well as their mapping relation by using the **CogZ-e** interface.
   For example: Data element in source ontology (O1) is "United Kingdom" and it relates to another element "United Kingdom" (via a line) in target ontology (O2) and the relation is equivalent then the mapping would be O1: United Kingdom _=_ O2: United Kingdom
   Identify those and indicate them to the evaluator.

___

___

___

In-Task Questionnaires

1. How confident are you in your finding of mapping links for this task?

1	2	3	4	5
(Very confident)		(Neutral)		(Very unconfident)

2. How difficult did you find it to complete this task?

1	2	3	4	5
(Very easy)		(Neutral)		(Very difficult)

Task Scenario 3 – Create a mapping link

Imagine you are an ontology domain expert and want to add any mappings between the source and target ontology concepts, which the alignment algorithm has missed. Find any two concepts in the given source and target ontologies that you believe should be mapped and create a mapping link between them using the **CogZ** interface. Write the names of the source and target ontology concept as well as the nature of relationship you have created among them. Add any two mappings.

In-Task Questionnaires

1. How confident are you in adding a mapping for this task?

1	2	3	4	5
(Very confident)		(Neutral)		(Very unconfident)

2. How difficult did you find it to complete this task?

1	2	3	4	5
(Very easy)		(Neutral)		(Very difficult)

Task Scenario 4 – Create a mapping link

Imagine you are an ontology domain expert and want to add any mappings between the source and target ontology concepts, which the alignment algorithm has missed. Find any two concepts in the given source and target ontologies that you believe should be mapped and create a mapping link between them using the **CogZ-e** interface. Write the names of the source and target ontology concept as well as the nature of relationship you have created among them. Add any two mappings.

In-Task Questionnaires

1. How confident are you in adding a mapping for this task?

1	2	3	4	5
(Very confident)		(Neutral)		(Very unconfident)

2. How difficult did you find it to complete this task?

1	2	3	4	5
(Very easy)		(Neutral)		(Very difficult)

Participant ID:

Pre-study questionnaire and user study tasks

Pre-Study Questionnaire

Please answer the following questions in regards to your background. Circle the answer that best describes you or your opinion.

1. Have you ever taken any ontology development and/or knowledge representation course?

   Yes No

2. What is your area of research/focus (if any)?

   ____

3. How confident are you about your knowledge regarding ontologies?

1	2	3	4	5
(Very confident)		(Neutral)		(Not at all confident)

4. Have you used any structure or database or ontology mapping/matching system before (for e.g., COMA++, SMatch, Protege, Anchor-Prompt)?

   Yes No Don't know

User Study Tasks

Task 1: Verification of mappings with CogZ

Imagine you are a data manager in a department of a data mining company which deals with the data integration. You have been hired to map two ontologies to each other which represent the domain of an academic conference. The mappings have already been computed and presented to you. Your task is to **verify any 5 of the 15 mappings** shown to you. Tell the instructor clearly which mapping you confirm, which mapping you believe should be removed and why they should be confirmed or removed.

In-Task Questionnaires

1. How familiar are you with the academic conference domain?

1	2	3	4	5
(Very familiar)		(Neutral)		(Very unfamiliar)

2. How confident are you that the mappings you have verified for this task are correct?

1	2	3	4	5
(Very confident)		(Neutral)		(Not at all confident)

3. How difficult did you find it to complete this task?

1	2	3	4	5
(Very easy)		(Neutral)		(Very difficult)

Task 2: Add a mapping with CogZ

Imagine you are a data manager in a department of a data mining company which offers services for data integration. You have been hired to map two ontologies to each other which represent the domain of an academic conference. The mappings have already been computed and presented to you. This time your task is to **add a mapping link between two concepts** which you believe are similar to each other and the mapping link is missing between them. Please **add 5 any mappings** that you feel are missing and tell the instructor why the mapping should be added.

In-Task Questionnaires

1. How familiar are you with the academic conference domain?

1	2	3	4	5
(Very familiar)		(Neutral)		(Very unfamiliar)

2. How confident are you that the mappings you have added for this task are correct?

1	2	3	4	5
(Very confident)		(Neutral)		(Not at all confident)

3. How difficult did you find it to complete this task?

1	2	3	4	5
(Very easy)		(Neutral)		(Very difficult)

Task 3: Verification of mappings with CogZ-Extended (CogZ-e)

Imagine you are a data manager in a department of a data mining company which offers services for data integration. You have been hired to map two ontologies to each other which represent the domain of an university. The mappings have already been computed and presented to you. Your task is to **verify any 5 of the 15 mappings** shown to you. Tell the instructor clearly which mapping you confirm, which mapping you believe should be removed and why they should be confirmed or removed.

In-Task Questionnaires

1. How familiar are you with the university domain?

1 (Very familiar)	2	3 (Neutral)	4	5 (Very unfamiliar)

2. How confident are you that the mappings you have verified for this task are correct?

1 (Very confident)	2	3 (Neutral)	4	5 (Not at all confident)

3. How difficult did you find it to complete this task?

1 (Very easy)	2	3 (Neutral)	4	5 (Very difficult)

Task 4: Add a mapping with CogZ-Extended (CogZ-e)

Imagine you are a data manager in a department of a data mining company which offers services for data integration. You have been hired to map two ontologies to each other which represent the domain of a university. The mappings have already been computed and presented to you. This time your task is to **add a mapping link between two concepts** which you believe are similar to each other and the mapping link is missing between them. Please **add 5 any mappings** that you feel are missing and tell the instructor why the mapping should be added.

In-Task Questionnaires

1. How familiar are you with the university domain?

1	2	3	4	5
(Very familiar)		(Neutral)		(Very unfamiliar)

2. How confident are you that the mappings you have added for this task are correct?

1	2	3	4	5
(Very confident)		(Neutral)		(Not at all confident)

3. How difficult did you find it to complete this task?

1	2	3	4	5
(Very easy)		(Neutral)		(Very difficult)

Note:
CogZ → without bundling
CogZ-e → with bundling

POST-TASK QUESTIONNAIRE

The following set of questions relate to your experiences using **CogZ** and **CogZ-e** as your interface for ontology **mapping verification tasks**.

INSTRUCTIONS: Please answer the following questions by circling the option that you feel is appropriate.

For Mapping Verification Tasks

Question	Please circle only one in each row		
Which interface enabled you to accomplish the mapping verification task more quickly?	CogZ	CogZ-e	They are the same
Which interface improved your mapping verification performance?	CogZ	CogZ-e	They are the same
Which interface increased your mapping verification productivity?	CogZ	CogZ-e	They are the same
Which interface enhanced your effectiveness when performing mapping verification?	CogZ	CogZ-e	They are the same
Which interface made it easier to verify mappings?	CogZ	CogZ-e	They are the same
Which interface did you find more useful for mapping verification?	CogZ	CogZ-e	They are the same

Question	Please circle only one in each row		
Which interface did you find easier to learn to operate?	CogZ	CogZ-e	They are the same
In which interface you find it easier to get the system to do what you wanted it to do?	CogZ	CogZ-e	They are the same
With which interface was your interaction more clear and understandable?	CogZ	CogZ-e	They are the same
Which interface did you find to be more flexible to interact with?	CogZ	CogZ-e	They are the same
With which interface was it easier for you to become skillful in using its features?	CogZ	CogZ-e	They are the same
Which interface did you find easier to use?	CogZ	CogZ-e	They are the same

INSTRUCTIONS: Please rate how strongly you agree or disagree with each of the following statements by circling the appropriate number.

Question	Please circle only one in each row				
How difficult was it to verify mappings with **CogZ?**	1 Very easy	2 Easy	3 Neutral	4 Difficult	5 Very difficult
How difficult was it to verify mappings with **CogZ-e?**	1 Very easy	2 Easy	3 Neutral	4 Difficult	5 Very difficult

Note:
CogZ → without bundling
CogZ-e → with bundling

POST-TASK QUESTIONNAIRE (CONTINUE)

The following set of questions relate to your experiences using **CogZ** and **CogZ-e** as your interface for ontology **mapping addition tasks**.

INSTRUCTIONS: Please answer the following questions by circling the option that you feel is appropriate.

For Mapping Addition Tasks

Question	Please circle only one in each row		
Which interface enabled you to accomplish the mapping addition task more quickly?	CogZ	CogZ-e	They are the same
Which interface improved your addition performance?	CogZ	CogZ-e	They are the same
Which interface increased your mapping addition productivity?	CogZ	CogZ-e	They are the same
Which interface enhanced your effectiveness when performing mapping addition?	CogZ	CogZ-e	They are the same
Which interface made it easier to add mappings?	CogZ	CogZ-e	They are the same
Which interface did you find more useful for mapping addition?	CogZ	CogZ-e	They are the same

Question	Please circle only one in each row		
Which interface did you find easier to learn to operate?	CogZ	CogZ-e	They are the same
In which interface you find it easier to get the system to do what you wanted it to do?	CogZ	CogZ-e	They are the same
With which interface was your interaction more clear and understandable?	CogZ	CogZ-e	They are the same
Which interface did you find to be more flexible to interact with?	CogZ	CogZ-e	They are the same
With which interface was it easier for you to become skillful in using its features?	CogZ	CogZ-e	They are the same
Which interface did you find easier to use?	CogZ	CogZ-e	They are the same

INSTRUCTIONS: Please rate how strongly you agree or disagree with each of the following statements by circling the appropriate number.

Question	Please circle only one in each row				
How difficult was it to find potential concepts for addition of a mapping with **CogZ**?	1 Very easy	2 Easy	3 Neutral	4 Difficult	5 Very difficult
How difficult was it to find potential concepts for addition of a mapping with **CogZ-e**?	1 Very easy	2 Easy	3 Neutral	4 Difficult	5 Very difficult

POST-TASK QUESTIONNAIRE (CONTINUE)

Please rank your mapping verification interface preference (order from 1 to 2):
____________ CogZ
____________ CogZ-e (with Edge-bundling)

Please rank your mapping addition interface preference (order from 1 to 2):
____________ CogZ
____________ CogZ-e (with Edge-bundling)

Please make any other comments about the ontology mapping verification and/or mapping addition interface.

____

____

____

____

____

____

____

____

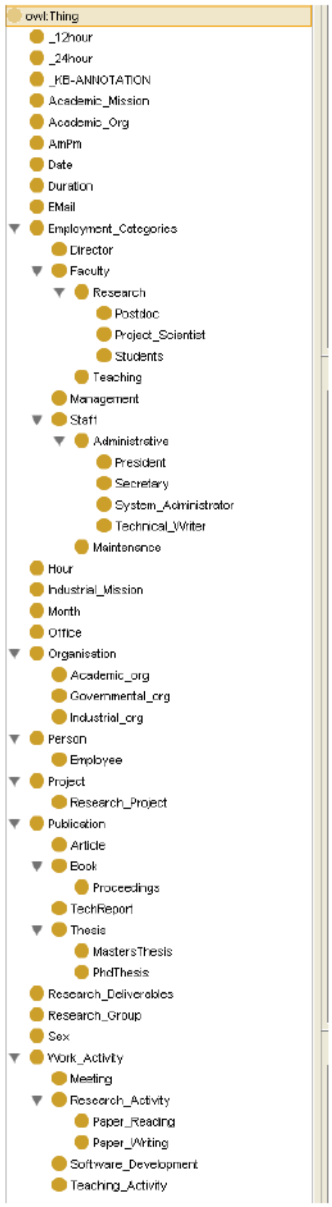
____

Appendix B

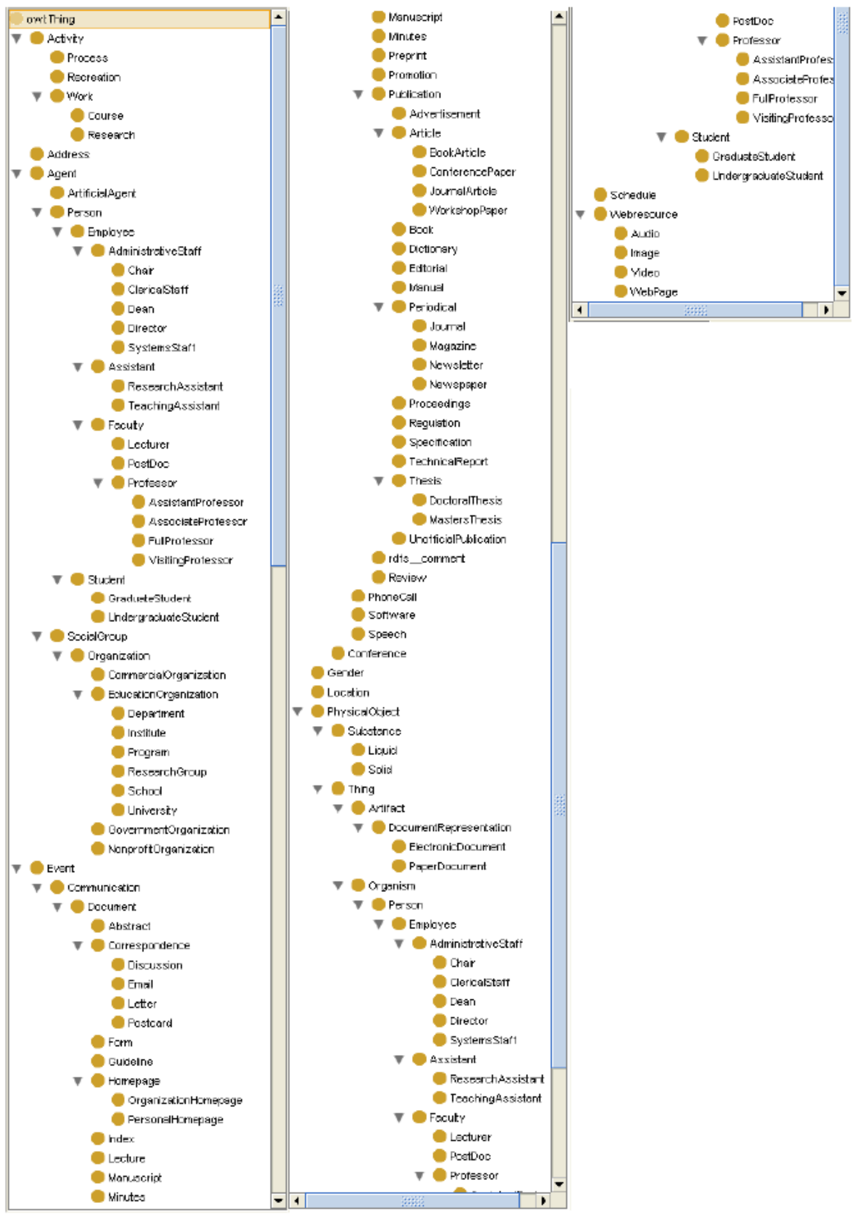
Test Ontology Datasets

This appendix includes the snapshots of the test ontology datasets used in the user study.

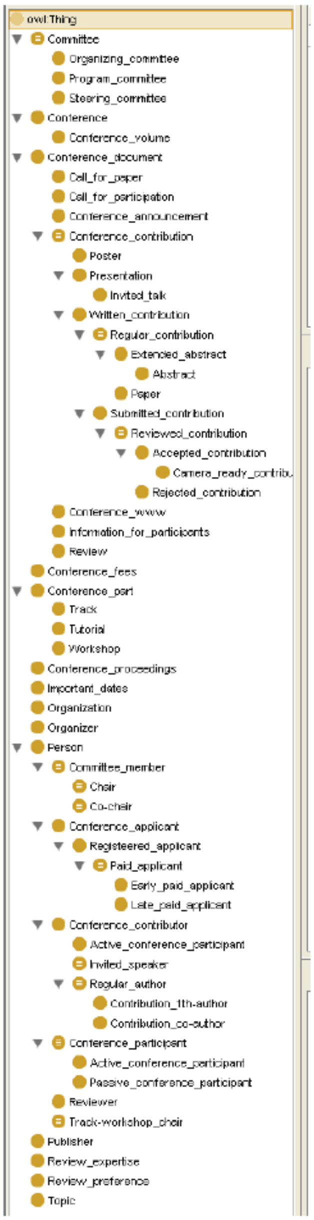
University Ontology A



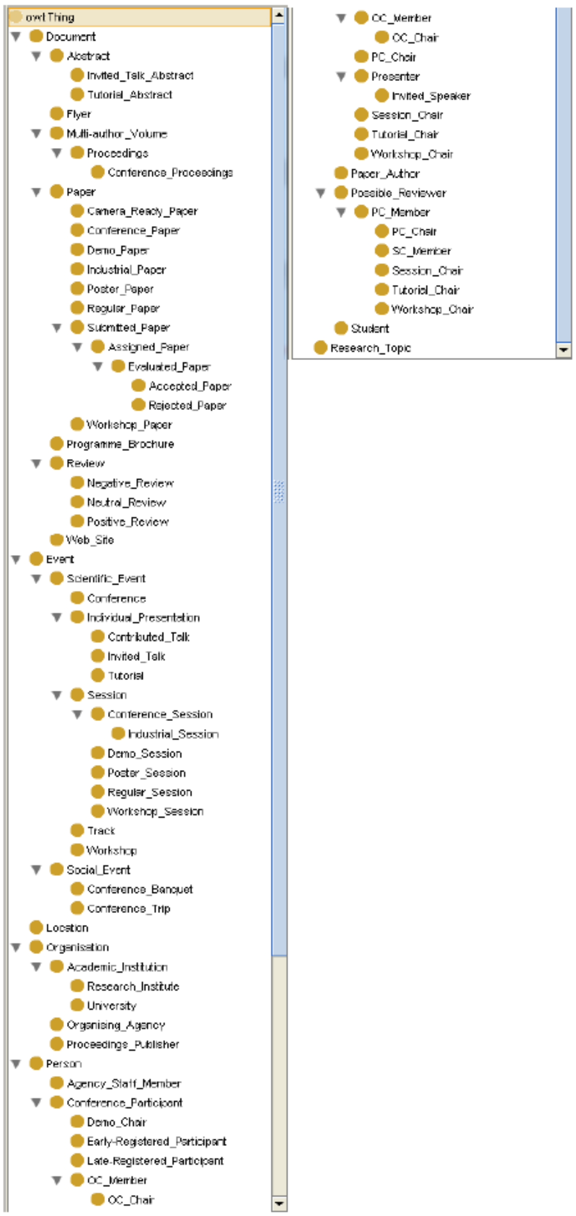
University Ontology B



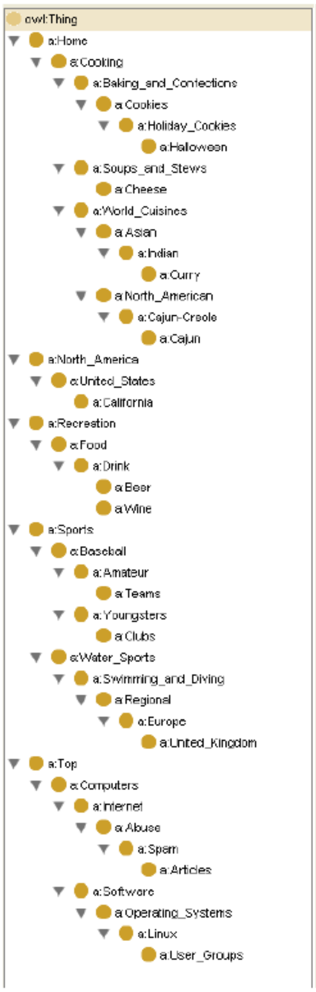
Conference Ontology A



Conference Ontology B



Common Things Ontology A



Common Things Ontology B