Ontology research and development, Part 2 - A review of ontology mapping and evolving.
Ding, Y., & **Foo, S.** (2002). *Journal of Information Science*, 28(5), 375-388.

Ontology Research and Development
Part 2 – A Review of Ontology Mapping and Evolving

Ying Ding
Division of Mathematics and Computer Science
Vrije Universiteit, Amsterdam
(ying@cs.vu.nl)

Schubert Foo
Division of Information Studies
School of Communication and Information
Nanyang Technological University, Singapore
assfoo@ntu.edu.sg

Abstract

This is the second of a two-part paper to review ontology research and development, in particular, to ontology mapping and evolving. Ontology is a formal explicit specification of a shared conceptualization. Ontology itself is not a static model so that it must have the potential to capture changes of meanings and relations. As such, mapping and evolving ontologies is part of an essential task of ontology learning and development.

Ontology mapping is concerned with reusing existing ontologies, expanding and combining them by some means and enable a larger pool of information and knowledge in different domains to be integrated to support new communication and use. Ontology evolving, likewise, is concerned with maintaining existing ontologies and extending them as appropriate when new information or knowledge is acquired.

It is apparent from the reviews that current researches on semi-automatic or automatic ontology research in all the three aspects of generation, mapping and evolving have so far achieved limited success. Expert human input is essential in almost all cases. Achievements have been largely made in the forms of tools and aids to assist the human expert. Many research challenges remain in this field and many of such challenges need to be overcome if the next generation of semantic web is to be realized in future.

Keywords Ontology, ontology mapping, ontology evolving, ontology learning

1. Introduction

Ontology is a formal explicit specification of a shared conceptualization (Gruber, 1993). It provides a shared and common understanding of a domain that can be communicated across people and application systems. Ontology itself is not a static model. It must have the potential to capture the changes of meaning (Fensel, 2001).

Ontologies are developed to provide the common semantics for agent communication. When two agents need to communicate or exchange information, the pre-requisite is that a common consensus has to form between them. This leads to the need to map two ontologies. For example, in business to business (B2B) e-commerce applications, the mapping among different classification standards (such as UNSPSC[1] and ecl@ss[2]) turns out to be not trivial.

This paper reports the second part of the survey on ontology research and development. The first part introduced the subject of ontology and focused on the state-of-the-art techniques and work done on semi-automatic and automatic ontology generation, as well as the problems facing these researches (Ding and Foo, 2001). This second part focuses on the current status of ontology mapping and ontology evolving researches.

An introduction of ontology mapping is first presented followed by a review of a number of different ontology mapping projects to illustrate the approaches, techniques, resulting mapping and problems associated with the resulting ontologies. This is followed by a review of related works on ontologies evolving using a similar framework of presentation.

2. Ontology mapping

Effective use or reuse of knowledge is essential. This is especially so now due to the overwhelming amount of information that is been continually generated, which in turn, has forced organizations, business and people to manage their knowledge more effectively and efficiently. Simply combining knowledge from distinct domains creates several problems, for instance, different knowledge representation formats, semantic inconsistencies, and so on. The same applies to the area of ontology engineering.

Upon ontologies generation, ontology engineers subsequently face the problem of how to reuse these existing ontologies, and how to map various different ontologies to enable a common interface and understanding to emerge in order to support communication among existing and new domains. As such, ontology mapping has been turned out to be another important research area for ontology learning.

Sofia Pinto, Gomez-Perez & Martins (1999) provided a framework and clarified the meaning of the term "ontology integration" to include that of ontology reuse, ontology

[1] http://eccma.org/unspsc/
[2] http://www.eclass.de/

merging and ontology alignment along with tools, methodology and applications as shown in Table 1.

Table 1. Examples of ontology integration techniques and applications

1. Integration of ontologies by building a new ontology and reusing other available ontologies (Ontology Reuse)		
Tools	Ontologies built	Methodology
Ontolingua server	PhySys, Mereology ontology, KACTUS, Standard-Units ontology, etc.	Integration of the building blocks and foundational theories.
2. Integration of ontologies by merging different ontologies into a single one that "unifies" all of them (includes Ontology Merging and Ontology Alignment)		
Tools	Ontologies built	Methodology
ONIONS	SENSUS, Agreed-Upon-Ontology (Skuce, 1997)	Manually, brainstorming, ONIONS.
3. Integration of ontologies into applications		
Tools	Ontologies built/Applications	Methodology
KACTUS	CYC, GUM, PIF, UMLS, EngMath, PhySys, Enterprise Ontology, Reference ontology	Manual method, brainstorming, ONIONS

Note: The various tools, ontologies and methodology presented in the Table will be discussed in subsequent sections.

Noy & Musen (1999) clarified the difference between ontology alignment and ontology merging and noted that "in ontology merging, a single ontology is created which is a merged version of the original ontologies, while in ontology alignment, the two original ontologies exist, with links established between them". There are several ways to carry out ontology mapping as in the ways the resulting mapping are represented. Mappings can be represented as conditional rules (Chang & Garcia-Molina, 1998), functions (Chawathe *et. al.*, 1994), logic (Guha, 1991), or a set of tables and procedures (Weinstein & Birmingham, 1998), and so on.

Ontology mapping has been addressed by researchers using different approaches:

- One-to-one approach, where for each ontology a set of translating functions is provided to allow the communication with the other ontologies without an intermediate ontology. The problem with this approach is the computing complexity (e.g., OBSERVER (Mena, et al., 1999))

- Single-shared ontology. The drawbacks of dealing with a single shared ontology are similar to those of any standards (Visser & Cui, 1998).

- Ontology clustering where resources are clustered together on the basis of similarities. Additionally, ontology clusters can be organized in a hierarchical fashion (Visser & Tamma, 1999).

Figure 1 shows a very simple example of ontology mapping to illustrate the process of mapping an Employee ontology and a Personnel ontology from different departments of the same company. A different UnitOfMeasure exists in these two ontologies so that the mapping rule of UnitConversion is needed to secure the right mapping.

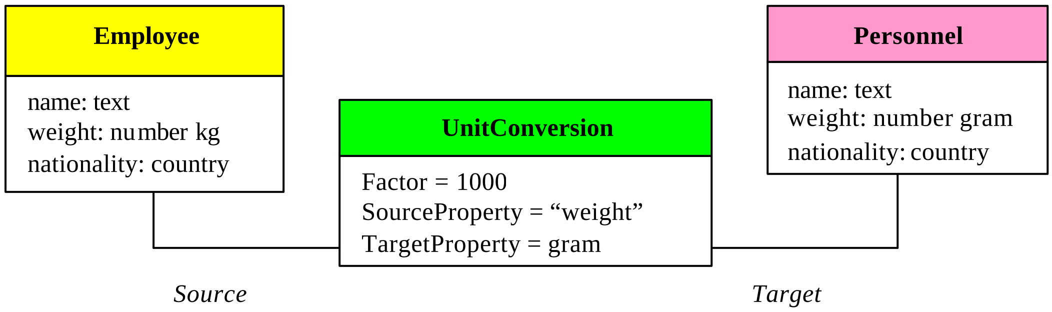

Figure 1. Simple example of ontology mapping

3. Ontology mapping projects

Many existing ontology mapping projects have been carried out and reported in the literature. The following sections provide a comprehensive update of the state of development of ontology mapping through these projects. Information about each project along with the important features associated with the project are provided and highlighted.

3.1 InfoSleuth's reference ontology

InfoSleuth (Fowler et al, 1999) can support construction of complex ontologies from smaller component ontologies so that tools tailored for one component ontology can be used in many application domains. Examples of reused ontologies include units of measure, chemistry knowledge, geographic metadata, and so on. Mapping is explicitly specified among these ontologies as relationships between terms in one ontology and related terms in other ontologies.

All mappings between ontologies are made by a special class of agents known as "resource agents". A resource agent encapsulates a set of information about the ontology mapping rules, and presents that information to the agent-based system in terms of one or more ontologies (called reference ontologies). All mapping is encapsulated within the resource agent. All ontology are represented in OKBC (Open Knowledge Base Connectivity) and stored in an OKBC server by a special class of agents called ontology agents, which could provide ontology specifications to users (for request formulation) and to resource agents (for mapping).

3.2 Stanford's ontology algebra

In this application, the mapping between ontologies has been executed by ontology algebra (Wiederhold, 1994) that consists of three operations, namely, intersection, union and difference. The objective of ontology algebra is to provide the capability for interrogating many largely semantically disjoint knowledge resources. Here, articulations (the rules that provides links across domains) can be established to enable the knowledge interoperability. Contexts, the abstract mathematical entities with some properties, were defined to be the unit of encapsulation for well-structured ontologies (McCarthy, 1993), which could provide guarantees about the knowledge they export, and contain the inferences feasible over them. The ontology resulting from the mappings between two source ontologies is assumed to be consistent only within its own context, known as an articulation context (Jannink et. al., 1998). Similar works can be found in McCarthy's work and the CYC[3] (Cycorp's Cyc knowledge base) project. For instance, McCarthy (1993) defined context as simple mathematical entities used for the situations in which particular assertions are valid. He proposed to use the lifting axioms to state that a proposition or assertion in the context of one knowledge base is valid in another. The CYC (Guha, 1991) use of microtheories bears some resemblance to this definition of context. Every microtheory within CYC is a context that makes some simplifying assumptions about the world. Microtheories in CYC are organized in an inheritance hierarchy whereby everything asserted in the super-microtheory is also true in the lower level microtheory.

Mitra, Wiederhold & Kersten (2000) used ontology algebra to enable interoperation between ontologies via articulation ontology. The input to the algebra is the ontology graphs. The operators in the algebra include unary operators like filter and extract, and binary operators include union, intersection and difference (as in normal set operators):

- The union operator generates a unified ontology graph comprising of the two original ontology graphs connected by the articulation. The union presents a coherent, connected and semantically sound unified ontology.
- The intersection operator produces the articulation ontology graph, which consists of the nodes and the edges added by the articulation generator using the articulation rules among two ontologies. The intersection determines the portions of knowledge bases that deal with similar concepts.

[3] http://www.cyc.com/

- The difference operator, to distinguish the difference of two ontologies (O1-O2) is defined as the terms and relationships of the first ontology that have not been determined to exist in the second. This operation allows a local ontology maintainer to determine the extent of one's ontology that remains independent of the articulation with other domain ontologies so that it can be independently manipulated without having to update any articulation.

They also built up a system known as ONION (Ontology compositION) which is an architecture based on a sound formalism to support a scalable framework for ontology integration. The special feature of this system is that it separated the logical inference engine from the representation model of the ontologies as much as possible. This allowed the accommodation of different inference engines. This system contains the following components:

- Data layer which manages the ontology representations, the articulations and the rule sets involved and the rules required for query processing.
- Viewer (which is basically a graphical user interface).
- Query system.
- Articulation agent that is responsible for creating the articulation ontology and the semantic bridges between it and the source ontologies. The generation of the articulation in this system is semi-automatic.

The ontology in ONION is represented by the conceptual graph and the ontology mapping is based on the graph mapping. At the same time, domain experts can define a variety of fuzzy matching. The main innovation of ONION is that it uses articulations of ontologies to interoperate among ontologies and it also represents ontologies graphically which could help in separating the data layer with the inference engine.

Mitra, Wiederhold & Jannink (1999) generated SKAT (Semantic Knowledge Articulation Tool) to extract information from a website by supplying a template graph. It also can extract structural information from an ontology that could be used to create a new ontology. Noy & Musen (1999) developed SMART which is an algorithm that provides a semi-automatic approach to ontology merging and alignment. SMART assists the ontology engineer by prompting to-do lists as well as perform consistency checking.

3.3 AIFB's formal concept analysis

The ontology learning group in AIFB (Institute of Applied Informatics and Formal Description Methods, University of Karlsruhe, Germany) through Stumme, Studer & Sure (2000) preliminarily discussed steps towards an order-theoretic foundation for maintaining and merging ontologies and articulated some questions about how a structural approach can improve the merging process, for instance:

- Which consistency conditions should ontologies verify in order to be merged?

- Can the merging of ontologies be described as a parameterized operation on the set of ontologies?
- How can other relations beside the *is-a* relation be integrated?
- How can an interactive knowledge acquisition process support the construction of the aligning function?
- How can meta-knowledge about concepts and relations provided by axioms be exploited for the aligning process, and so on.

They proposed the Formal Concept Analysis (Ganter & Wille, 1999) for the merging and maintaining ontologies that offers a comprehensive formalization of concepts by mathematising them as a unit of thought constituted of two parts: its extension and its intension. Formal Concept Analysis starts with a formal context defined as a triplet K : = (G, M, I), where G is a set of objects, M is a set of attributes, and I is a binary relations between G and M. The interested reader can refer to Ganter & Wille (1999) for a more detailed account of this technique.

3.5 ECAI2000's method

A number of automatic ontology mapping researches were reported in the Ontology Learning Workshop of ECAI 2000 (European Conference on Artificial Intelligence). It is well known fact that discovering related concepts in a multi-agent system with diverse ontologies is difficult using existing knowledge representation languages and approaches.

Williams & Tsatsoulis (2000) proposed an instance-based approach for identifying candidate relations between diverse ontologies using concept cluster integration. They discussed how their agents represent, learn, share, and interpret concepts using ontologies constructed from Web page bookmark hierarchies. The concept vector represents a specific Web page and the actual semantic concept is represented by a group of concept vectors judged to be similar by the user (according to the meaning of the bookmark). The agents use supervised inductive learning to learn their individual ontologies. The output of this ontology learning is semantic concept descriptions (SCD) represented as interpretation rules. They built up one system to fulfill this purpose called DOGGIE, which could apply the concept cluster algorithm (CCI) to look for candidate relations between ontologies. The experimental results have demonstrated the feasibility of the instance-based approach for discovering candidate relations between ontologies using concept cluster integration. However, here they assume all the relations are only general *is-a* relations. This method could be very useful for ontology merging.

Tamma & Bench-Capon (2000) presented a semi-automatic framework to deal with inheritance conflicts and inconsistencies while integrating ontologies. This framework represents ontologies by a frame-based language where the classical set of slot facets is extended to encompass other information in order to associate with each attribute a degree of strength thus permitting to deal with default conflicts and inconsistencies. The framework includes several steps:

- Check the class and slot name's synonyms.
- If a name mismatch is detected, the system proceeds both bottom-up and top-down trying to relate classes
- If an inconsistency is detected then the priority functions for the inheritance rules are computed on the grounds of both the rankings of probabilities and the degree of strength
- Subsequently, the system provides domain experts with a list of suggestions that are evaluated according to a priority function. The final choice is always left to the domain experts, but the system provides them with a set of possible choices including information concerning how and when the attribute changes.

Uschold (2000) pointed out the global reference ontology will be the perfect candidate for the ontology mapping of the local ontologies. Different user communities can view the global reference ontology from their own preferred perspectives through mapping and projecting. The basic idea is to define a set of mapping rules to form a perspective for viewing and interacting with the ontology. Different sets of mapping rules enable the ontology, or a portion of it to be viewed from three different perspectives: viewing the global ontology using own local terminologies; viewing a selected portion of the ontology; and viewing at a higher level of abstraction.

3.5 ISI's OntoMorph

The ISI's OntoMorph system aims to facilitate ontology merging and the rapid generation of knowledge base translators (Chalupsky, 2000). It combines two powerful mechanisms to describe KB transformations. The first of these mechanisms is syntactic rewriting via pattern-directed rewrite rules that allow the concise specification of sentence-level transformations based on pattern matching, and the second mechanism involves semantic rewriting which modulates syntactic rewriting via semantic models and logical inference. The integration of ontologies can be based on any mixture of syntactic and semantic criteria.

In syntactic rewriting process, input expressions are first tokenized into lexemes and then represented as syntax trees, which are represented internally as flat sequences of tokens and their structure only exists logically. OntoMorph's pattern language and execution model is strongly influenced by Plisp (Smith, 1990). The pattern language can match and de-structure arbitrarily nested syntax trees in a direct and concise fashion. Rewrite rules are applied to the execution model.

For the semantic rewriting process, OntoMorph is built on top of the PowerLoom knowledge representation system, which is a successor to the Loom system. Using semantic import rules, the precise image of the source KB semantics can be established within PowerLoom (limited only by the expressiveness of first-order logic).

3.6 KRAFT's ontology clustering

Visser *et. al.* (1999) proposed a set of techniques to map one-to-one ontologies in the KRAFT project:

- Class mapping: maps a source ontology class name to a target ontology class name;
- Attribute mapping: maps the set of values of a source ontology attribute to a set of values of a target ontology attribute; or maps a source ontology attribute name to a target ontology attribute name;
- Relation mapping: maps a source ontology relation name to a target ontology relation name, and
- Compound mapping: maps compound source ontology expressions to compound target ontology expressions.

Following this, Visser & Tamma (1999) suggested the concept of 'ontology clustering' to integrate heterogeneous resources. Ontology clustering is based on the similarities between the concepts known to different agents. The ontology clustering was represented in the hierarchy fashion. The ontology on top of the hierarchy is known as the application ontology that is used to describe the specific domain so that it is not reusable. The application ontology contains a relevant subset of WordNet concepts with senses selected from WordNet. A new ontology cluster is a child ontology that defines certain new concepts using the concepts already contained in its parent ontology. Concepts are described in terms of attributes and inheritance relations, and are hierarchically organized. This approach has been applied to the domain of international coffee preparation.

3.7 Heterogeneous database integration

In the ontology community, the concept of ontology has been extended to encompass a very broad scope. Many classification systems, catalogues, indexes have also been referred to ontology (or more specificity in this context as lightweight ontology). A database scheme is another of such lightweight ontology. In the database community, the problems concerning of the integration of heterogeneous database were raised long time ago. Initial research in heterogeneous databases was largely directed towards the issues of resolving schema and data heterogeneity conflicts across multiple autonomous databases (Kim & Seo, 1991), and of developing a global schema to provide integration (Bright et. al., 1994; Castano & De Antonellis, 1997; Palopoli et. al., 2000). Some of these researches are worthy of mention since they offer results that could indicate potential solutions for ontology mapping and merging.

Batini *et. al.* (1986) provided a comprehensive survey of different schema integration techniques. They define schema integration as the activity of integrating schemata of existing or proposed databases into a global, unified schema. The five steps to schema

integration described include pre-integration, comparison, conformation, merging, and restructuring.
While structural integration has been well defined in Sheth and Larson (1990) and Batini et. al. (1986), the treatment from a semantic perspective is not that easy. In an attempt to reconcile the semantic and schematic perspectives, Sheth and Kashyap (1992) presented a semantic taxonomy to demonstrate semantic similarities between two objects and related this to a structural taxonomy. Various types of semantic relationships have been discussed in the literature. Many terms such as semantic equivalence, semantic relevance, semantic resemblance, semantic compatibility, semantic discrepancy, semantic reconciliation, and semantic relativism have been defined. There is no general agreement on these terms. There are a number of projects addressing the semantic issues of database integration, for example, FEMUS (Swiss Institute of Technology), ETHZ, COIN (Context technology Interchange Network, MIT).

At present, intelligent integration has been applied to heterogeneous database integration. Two major ideas are proposed in the literature: The first is via *agents* (Bordie, 1992). Typical systems are RETSINA[4] (an open multi-agent system (MAS)) and InfoSleuth. The second is based on the concept of *mediators* (Wiederhold, 1994a) that provide intermediary services by linking data resources and application programs. Examples are the TSMISS at Stanford University. Both information agents and mediators require domain knowledge that is modeled in some kind of common vocabulary (ontology).

Palopoli et. al. (2000) present two techniques to integrate and abstract database schemes. Scheme integration is to produce a global conceptual scheme from a set of heterogeneous input schemes. Scheme abstraction groups objects of a scheme into homogeneous clusters. Both assume the existence of a collection of inter-schema properties describing semantic relationships holding among input database scheme objects. The first technique uses inter-schema properties to produce and integrate schema. The second one takes an integrated scheme as the input and yields an output of the form of an abstracted scheme. The difficulties encountered in achieving schema integration have highlighted the importance of capturing the semantics embedded in the underlying schemata. There is a general agreement that integration can be achieved only with a good understanding of the embedded semantics of the component databases. Srinivasan, Ngu & Gedeon (2000) introduced a conceptual integration approach that exploits the similarity in meta-level information on database objects to discover a set of concepts that serve as a domain abstraction and provide a conceptual layer above existing legacy systems

3.8 Other ontology mappings

Hovy, E. (1998) described several heuristics rules to support the merging of ontologies. For instance, the NAME heuristic compares the names of two concepts, the DEFINITION heuristics uses linguistic techniques for comparing the natural language definitions of two concepts, and the TAXONOMY heuristic checks the closeness of two concepts to each other.

---

[4] http://www-2.cs.cmu.edu/~softagents/retsina.html

The Chimaera system (McGuinness et al., 2000) can provide support for merging of ontological terms from different knowledge sources, for checking the coverage and correctness of ontologies and for maintaining ontologies. It contains a broad collection of commands and functions to support the merging of ontologies by coalescing two semantically identical terms from different ontologies and by identifying terms that should be related by subsumption or disjointness relationships.

PROMPT (Noy & Musen, 2000) is an algorithm for ontology merging and alignment that is able to handle ontologies specified in OKBC compatible format. It starts with the identification of matching class names. Based on this initial step an iterative approach is carried out for performing automatic updates, finding resulting conflicts, and making suggestions to remove these conflicts. PROMPT is implemented as an extension to the Protégé 2000 knowledge acquisition tool and offers a collection of operations for merging two classes and related slots.

Li (Li, 1995) identifies similarities between attributes from two schemas using neural networks. Campbell and Shapiro (1995) described an agent that mediates between agents that subscribe to different ontologies. Bright *et, al.* (1994) use a thesaurus and a measure of “semantic-distance” based on path distance to merge ontologies. Kashyap and Sheth (1996) define the “semantic proximity” of two concepts as a tuple encompassing contexts, value domains and mappings, and database states. The resulting analysis yields a hierarchy of types of semantic proximity, including equivalence, relationship, relevance, resemblance, and incompatibility.

Lehmann & Cohn (1994) require that concept definitions of the ontologies should include more specialized definitions for typical instances, and assume that the set relation between any two definitions can be identified as either equivalence, containment, overlap or disjointness. OBSERVER (Mena, et al., 1999) combines intensional and extensional analysis to calculate lower and upper bounds for the precision and recall of queries that are translated across ontologies on the basis of manually identified subsumption relations.

Weinstein & Birmingham (1999) compared concepts in differentiated ontologies, which inherit definitional structure from concepts in shared ontologies. Shared, inherited structure provides a common ground that supports measures of "description compatibility." They use description compatibility to compare ontology structures represented as graphs and identify similarities for mapping between elements of the graphs. The relations they find between concepts are based on the assumption that local concepts inherit from concepts that are shared. Their system was evaluated by generating description logic ontologies in artificial words.

Borst & Akkermans (1997) used the term ‘ontology mapping’ to describe the process that an entity of a primary ontology is furthered differentiated through the application of a secondary ontology. The mapping exists between the secondary and the primary ontology and constraints on a secondary ontology restrict the application of its elements. The result of ontological mappings is a set of ontological commitment which could be considered as a new ontology.

3.9 Ontology mapping: Summary of observations

The summary of the aforementioned research of ontology mapping along the dimensions of mapping rules, resulting ontology, application areas, assisting tools and systems are summarized in Table 2.

From these reviews, it became evident that most of such mappings depend very much on the inputs of human experts. Although some tools are available to facilitate the mapping, the limited functions they could provide are class or relation name checking, consistency checking, to-do list recommendation, and so on. Ontology mapping is not a simple one-to-one mapping (link the class name, relation name, attribute name from one ontology to another) but on the contrary, demands substantial deeper checking and verification for inheritance, consistency of the inference, and so on. Furthermore, ontology mapping could be complicated by many-to-one, one-to-many or many-to-many relations either within one domain or one that transcends across different domains (ontology clustering).

Ontology mapping could also be viewed as the projection of the general ontologies from different point of views, either according to the different application domains or various tasks or applications (Uschold, 2000). Much remains to be done in the area of semi-automatic or automatic ontology mapping. However, when we tackle and overcome the problems facing automatic ontology generation, we are very much moving towards providing a solution for ontology mapping at the same time.

Table 2 Summary for ontology mapping

Project	InfoSleuth	Stanford	AIFB	ECAI2000	ISI	KRAFT	Others
Mapping rules	. reference ontology . agent mapping (reference agent, ontology agent)	. ontology algebra .articulation . context (articulation context) . lifting algorithm . microtheory . graph mapping . fuzzy mapping	. order-theoretic foundation . formal concept analysis . theoretical discussion (no specific methods)	. concept cluster integration . supervised inductive learning . default conflicts and inconsistency checking based on the features provided by frame-based language . ontology projection	. syntactic rewriting . semantic rewriting (based on semantic models or logical inference)	. one-to-one ontology mapping . ontology clustering	. heuristic rules . logical subsumption or disjointness . PROMPT . neural networks for identifying similarity between attributes . path distance (semantic distance) . semantic proximity . description compatibility
Results for final ontology	. represented in OKBC . stored in OKBC server	. the final ontology is assumed to be consistent only within its own context . represented in conceptual graph	No.	. semantic concept descriptions . represented in frame-based language	. PowerLoom (first-order logic)	. hierarchy fashion	. OKBC . conceptual graph . description logic . a set of ontological commitment
Semi-automatic/automatic?	Manually (or semi-automatically)	Manually (or semi-automatically)	No.	Manually (or semi-automatically)	Manually (or semi-automatically)	Manually (or semi-automatically)	Manually (or semi-automatically)
Application area	Multi-agent system	No.	No.	. multi-agent system	No.	International coffee preparation	. database
Assisting tool (system)	No.	. ONION . SKAT . SMART (consistency checking and to-do list)	No.	. DOGGIE	. OntoMorph	No.	. Chimaera . OBSERVER
Others				. assume all relations are general is-a relations		. WordNet	

4. Ontology evolving (maintenance)

The second half of the paper examines ontology evolving (the preferred term used in ontology literature over ontology maintenance). Ontology evolving requires the clarity of structure of ontologies so as to guarantee the accurate gauge of the maintenance. Some of the researches on semi-automatic or automatic ontology evolving have been carried out recently, but almost all of them are at their early stages and require manual intervention.

Fowler *et. al.* (1999) mentioned the need to maintain different versions of the same ontology in InfoSleuth agent architecture is it is being scaled up. Both TOVE (Gruninger & Fox, 1995a) and METHODOLOGY (Fernandez-Lopez, Gomez-Perez & Juristo, 1997) focus on the ontology maintenance, but manually. The main difference is that METHODOLOGY focuses on comprehensively addressing the maintenance stage of the life cycle of ontology whereas TOVE utilizes more formal techniques to address a more limited number of maintenance issues. Uschold (2000) also pointed out the importance of the ontology maintenance between local and global ontologies, especially the importance of standardizing the concepts and relations by using the unique identifier codes and names.

4.1 Stanford's algebraic methodology

Jannink & Wiederhold (1999) maintained the ontology using the same methodology as that employed to generate their ontologies – ontology algebra. When the knowledge base changes, they use the S operator, which provides a simple method for assessing the contents of a context to reveal terms with missing end tags in the data. For instance, $S_{len(hw)div20}$(dictionary) returns the entries of the dictionary, grouped by length of the head words. Once the errors were identified, the rules to convert terms with missing end tags are added. By maintaining statistics with the S operator on the process of extracting the relevant parts of the dictionary, rules can be updated straightforwardly. The congruity measure within the algebraic framework significantly simplified the process of identifying and handling the changes among the ontologies. Wiederhold (1994a) implored that "it is important that new systems for our networks are designed with maintenance in mind". Keeping ontologies small enables semantic agreements among the people using them with little lag time.

4.2 AIFB's method

Stumme, Studer & Sure (2000) also used the same method in their ontology mapping for ontology maintenance. They pointed out that current maintenance of ontologies is tedious and time-consuming. Domain expert can easily loose orientation in large ontologies. Tool support is needed to make reasonable suggestions to the expert or automate certain tasks based on some pre-defined principles. They considered that future research in this direction should include the integration of approaches for generating ontologies from text with linguistic methods, and for evaluating them with data mining techniques. They proposed the Formal Concept Analysis for ontology maintenance. Formal Concept Analysis (Ganter & Wille, 1999) offers formalization by mathematizing the concept as a unit of thought constituted of two parts: its extension and its intension.

4.3 ECAI 2000

Faatz, Kamps & Steinmetz (2000) presented an algorithm to determine similarities between text documents and used the strict supervised learning algorithm to improve the already-existing ontology based on the similarities or clusters of the relevant knowledge. First, they chose the plain-text document from online newswire or other web documents, then indexed them according to the controlled vocabularies and the already-built-up ontology (manually). They used the vector space model to normalize each document adding some background knowledge from the manually built-up ontology to enrich the whole document. After that, they used multi-linear regression to cluster or match the similarity among the documents and let the domain experts or the ontology engineers to decide whether these knowledge learned from the similarity matching or clustering could be used to maintain the current ontology. They concluded the paper with several areas for future researches: 1) confirmation of the proposed idea by experimenting with a certain amount of new vocabulary; 2) improving the results by introducing an additional qualitative tagging of keywords in the vector representation; and 3) attempting to find new ways to automatically detect the ontological relations.

Roux, et al. (2000) present a method to insert new concepts in an existing information extraction system based on conceptual graph architecture. In this system, an ontology is a two-fold set that contain passive information, the lattice, combined with active information, the graph patterns. This system has adopted a two-levels architecture:

- A linguistic analysis component that is mainly based on the Finite State Transducer technology. It consists of a Part-of-Speech tagger and a Robust Parser. Text is processed in several steps: tokenization, morphological analysis, disambiguation using an Hidden Markov Model technique, error corrections, and finally a contextual lookup to identify gene names. Syntactic dependencies are then extracted from that output.
- A knowledge based processing component based on conceptual graphs. The syntactic dependencies extracted at the linguistic level are used to build the semantic representation.

When a new word appears, pattern matching and syntactic dependencies were used to detect this new word from the Web document and insert it to the lattice. As the lattice comprises concepts that are connected to each other along semantic paths, this poses the requirement to categorize this new concept in order to find its correct slot in the lattice. This is based on the projection of customized conceptual sub-graphs with typed concepts on specific nodes to detect certain configuration of verbs that will assign their position in the lattice. However, this algorithm is currently under development and future tests are needed to validate its usefulness and applicability.

Todirascu *et. al.* (2000) proposed a system to acquire new concepts and to place them into the existing domain hierarchies (domain ontology). When a new document is added to the index base, it is processed by the Part-of-Speech technique. From here, the most frequent content words (noun, adjective, etc.) and their contexts were extracted. For each context, a concept description is built and classified in the existing hierarchy. Sense tagging assigns words and syntagms with their Description Logic description. Partial semantic descriptions are combined by heuristic rules to encode syntactic knowledge. In order to limit the size the domain ontology, they encountered some major problems in that the selection of high frequency words always ends as the very general concepts and inconsistent conceptual descriptions of concepts are very common.

Agirre *et. al.* (2000) used the topic signatures and hierarchical clusters to tag a given occurrence of a word with the intended concept to enrich very large ontologies from the Web. In this work, the main obstacle to get clean signatures comes from the method to link concepts and relevant documents from the web. Some filtering techniques have to be applied in order to get documents with less bias and more content. Cleaner topic signature opens the avenue for interesting ontology enhancement, as they provide concepts with rich topical information. For instance, similarity between topic signatures could be used to find out topically related concepts, so that the clustering strategy could be extended to all other concepts.

4.4 Ontology server

The core architecture of OntoSeek is an ontology server (Guarino, Masolo & Vetere, 1999). The server provides an interface for applications willing to access or manipulate an ontology data model (a generic graph data structure), and facilities for maintaining a persistent LCG (Lexical Conceptual Graph) database. End users and resource encoders can access the server to update the LCG database encoded in markup language, such as HTML or XML.

Uschold & Gruninger (1996) extensively discussed the ontology server provided by the Knowledge Systems Laboratory at Stanford University. To some degree, they already found the importance of the ontology server for the foreseeable-future ontology maintenance research. The main function of the ontology server is to create, edit, evaluate, publish, maintain, and reuse ontologies. The particular significance is the ability to support the collaborative works through the Web. Duineveld *et. al.* (1999) also conducted a complicated comparison of the current available ontology editing tools from different points of views among which include the client/server support of tools to facilitate the collaborative ontology generation and the ontology maintenance issue. Uschold & Gruninger (1996) list the improvement of ontology server in contrast to the original Ontolingua system as follows:

- It is a remote computer server available on the Web.
- It provides an extensible library of sharable reusable ontologies with suitable protections for proprietary, group and private work.

- There is an extensive browsing capability, which allows convenient viewing of ontologies.
- It has extended the original representation language to support decomposition of ontologies into modules and assembling new ontologies from existing modules from the library.
- It provides explicit support for collaborative work. This includes the concept of session to which multiple parties can be attached; parties are automatically informed of each other's activities.
- It has an application programmer's interface (API) which allow remote applications to query and modify ontologies stored on the server overt the Internet.
- As well as translating into multiple output language, it also allows multiple input languages.

4.5 Ontology evolving: Summary of observations

A summary of the aforementioned researches in the area ontology evolving along the dimensions of evolving or maintenance rules and future research requirements are shown in Table 3.

Table 3. Summary of ontology evolving or maintenance

Project	Stanford	AIFB	ECAI2000	Ontology server	Others
Evolving or maintenance rules	. ontology algebra . congruity measure . keeping ontologies small . bear the maintenance in mind when design or generate the ontology	. order-theoretic foundation . formal concept analysis . theoretical discussion (no specific methods)	. similarity measures based on the strict supervised learning . concept clustering . lattice and semantic path identification . partial semantic description (description logic) . topic signature & hierarchical cluster	. facilitate browse and editing, .client/server accessing . online collaboration	. TOVE (formal techniques) . METHODOLOGY (maintenance stage of the life cycle of ontology) . standardising concepts and relations (unique identifier codes and names)
Semi-automatic/automatic?	Manually (Semi-automatic)	-	Manually (Semi-automatic)	Manually (Semi-automatic)	Manually
Future researches	-	. generating ontology automatically from the text based on linguistic methods . evaluate it with data mining techniques	. try additional qualitative tagging in the vector representation . automatic detect ontological relations	-	-
Assisting tool (system)	-	-	-	. OntoSeek	-

5. Discussion: Ontology and its role in bringing Semantic Web to its full potential

The World Wide Web (WWW), with its continuous explosive growth of information, has caused a severe information overload problem. It is increasingly difficult to find, access, present, and maintain the information required by a wide variety of users. This is so since the information content is presented primarily in natural language. A wide gap has emerged between the avaiable information available and the tools that are in place to support users information seeking and use.

Many new research initiatives and commercial enterprises have been set up to enrich available information with machine-processable semantics. Such support is essential for "bringing the web to its full potential". Tim Berners-Lee, Director of the World Wide Web Consortium, referred to the future of the current WWW as the "*Semantic Web*" - an extended web of machine-readable information and automated services that extend far beyond current capabilities. The explicit representation of the semantics underlying data, programs, pages, and other web resources, will enable a knowledge-based web that provides a qualitatively new level of service.

A key enabler for the semantic web is on-line ontological support for data, information and knowledge exchange. Given the exponential growth of the information available online, automatic processing is vital to the task of managing and maintaining access to that information. Used to describe the structure and semantics of information exchange, *ontologies* is seen to play a key role in areas, such as knowledge management, B2B e-commerce and other such burgeoning electronic enterprises.

Currently the Semantic Web has attracted much interest from various communities. Many interesting and important research issues related to the Semantic Web have been addressed through ontology development and research and a number of promising advances have been made:

- Ontology Language: W3C (www.w3c.org) and EU IST projects (OntoKnowledge (www.ontoknowledge.org), OntoWeb (www.ontoweb.org) and Wonderweb (wonderweb.semanticweb.org)) are working together with the objective of deriving efficient ontology languages. Current candidate lanuages include those of RDF (www.w3.org/RDF), DAML+OIL (www.w3c.org/2001/sw), Topic Maps (www.topicmaps.org), and so on. (More information on these languages can be found in the semantic web portal at http://www.semanticweb.org)

- Ontology tools: Many tools, both research-based or commercial entities, are avaialbe to support ontology editing, ontology library management, ontology mapping and aligning, ontology inferening. (More information and a list of such tools can be found in http://www.ontoweb.org/sig.htm)

- Digital Libraries: XML or its variants are already widely-adopted to tag the metadata of e-journals or museum collections in many digital library projects. Documents or information are represented as much as possible in the machine-understandable format. The EU IST project –HeritageMap (http://sca.lib.liv.ac.uk/collections/HeritageMap/HeritageMapB.html) is a good example of such a development.

- Pioneering Business applications: Although the Semantic Web from the business-application point-of-view is far from reality, ontologies have been deployed for many applications such as corporate intranets and knowledge management (for example: www.ontoknowlege.org), e-commerce (for example: mkbeem.elibel.tm.fr/home.html), web portals and communities (for example: cweb.inria.fr). A limited number of successful scenarios for ontology-based applications can be found in http://www.ontoweb.org/download/deliverables/D21_Final-final.pdf

- Web Services is a development that deals with the limitation of the current web by aiming to transform the web from a collection of information into a distributed device of computation. In order to do so, appropriate description means for web services need to be developed. These are based on semi-formal natural language descriptions so that there is a need for human programmers to provide the links between services. A related development is the development of the Semantic Web enabled Web Services (www.cs.vu.nl/~dieter/wsmf) that is aimed to provide mechanization in service identification, configuration, comparison and combination. A key enabler is ontologies that provide the terminology used by such services.

Although such developments are promising, fundamental challenges remain at the root of ontology research and development.

6. Conclusion

Ontology research and development has gained substantial interest of late as researchers grapple with the information overload problem and the need to better organize and use information in order to support information retrieval and extraction to deliver the right information, in the right amount and at the right time. Researchers are diverse and come from the fields of library and information science, computer science, artificial intelligence, e-commerce and knowledge management. Ontologies have a higher level of applicability over the traditional computing and information science techniques in its ability to define relationships, deeper semantics, enhanced expressiveness, all of which collectively serve as the enabler for the next generation of semantic web to become a reality.

The need for manual intervention in all the three aspects of ontology generation, mapping and evolving attest to the complex nature of ontology research and its associated unresolved problems. Many useful tools and techniques have surfaced and these serve as useful aids to human experts to create, use and maintain existing ontologies. Nonetheless, many remaining research challenges need to be resolved in order for ontology to be applied on scale envisaged in future. A focus on tackling ontology generation is the first necessary step to be taken since this has direct implications and applications in the area of ontology mapping and evolving. So far, no significant breakthrough in semi-automatic and automatic ontology generation has been reported but the interest and work done by many well-known research institutions and agencies augurs well for this area of research in future.

References:

Agirre, E., Ansa, O. Hovy, E. & Martinez, D. (2000). Enriching very large ontologies using the WWW. In. *Proceedings of the First Workshop on Ontology Learning (OL-2000) in conjunction with the 14th European Conference on Artificial Intelligence (ECAI 2000)*. August, Berlin, Germany.

Batini, C., Lenzerini, M., & Navathe, S.B. (1986). A comparative analysis of methodologies for schema integration. *Computing Surveys*, 18(4), 323-364.

Borst, P. & Akkermans, H. (1997). An ontology approach to product disassembly. *EKAW97*. Sant Feliu de Guixols, Spain, October.

Bright, M.W., Hurson, A.R. Pakzad, S. (1994). Automated resolution of semantic heterogeneity in multidatabases. *ACM Transactions on Database Systems*, 19(2), 212-253.

Bordie, M. (1992). The promise of distributed computing and the challenges of legacy information systems. In *Proceedings of IFIP DS-5 semantics of interoperable databases systems.* Lorne, Australia: Chapman & Hall.

Campbell, A. E. & Shapiro, S.C. (1995). Ontologic mediation: An overview. *IJCAI95 Workshop on Basic Ontological Issues in Knowledge Sharing*, Montreal, Canada.

Castano, S., De Antonelli, V. (1997). Semantic dictionary design for database interoperability. In. *Proceedings of the International Conference on Data Engineering (ICDE97)*, IEEE Computer Society, Birmingham, UK.

Chalupsky, H. (2000). OntoMorph: A translation system for symbolic knowledge. In. *Proc. 7th Intl. Conf. On Principles of Knowledge Representation and Reasoning (KR2000)*, Breckenridge, Colorado, USA, April 2000.

Chang, C.C.K. & Garcia-Molina, H. (1998). Conjunctive constraint mapping for data translation. *Third ACM Conference on Digital Libraries*, Pittsburgh, USA.

Chawathe, S., Garcia-Molina, H., Hammer, J., Ireland, K., Papakonstantinou, Y., Ullman, J. & Widom, J. (1994). The TSIMMIS project: Integration of heterogeneous information sources. In. *IPSJ Conference*, Tokyo, Japan.

Ding, Y. and Foo, S. (2002). Ontology research and development: Part 1 – A review of ontology generation. *Journal of Information Science*, 28(2).

Duineveld, A.J., Stoter, R., Weiden, M.R., Kenepa, B., Benjamins, V.R. (1999). WonderTools? A comparative study of ontological engineering tools. *KAW 99: Twelfth Workshop on Knowledge Acquisition, Modeling and Management*, Alberta, Canada, October, 1999.

Faatz, A., Kamps, T. & Steinmetz, R. (2000). Background knowledge, indexing and matching interdependencies of document management and ontology maintenance. In. *Proceedings of the First Workshop on Ontology Learning (OL-2000) in conjunction with the 14th European Conference on Artificial Intelligence (ECAI 2000)*. August, Berlin, Germany.

Fensel, D. (2001). Understanding is based on Consensus. In *Semantic Web Panel*, *WWW10*, HongKong, May, 2001.

Fernandez-Lopez, M., Gomez-Perez, A. & Juristo, N. (1997). METHODOLOGY: From ontological art towards ontological engineering. In., *AAAI-97 Spring Symposium on Ontological Engineering*. Stanford University, March 24-26.

Fowler, J., Nodine, M., Perry, B., & Bargmeyer, B. (1999). Agent-based semantic interoperability in Infosleuth. *Sigmod Record*, 28, 1999.

Ganter, B. & Wille, R. (1999). *Formal concept analysis: Mathematical Foundations*. Springer, Berlin-Heidelberg.

Gruber, T.R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5, 199-220.

Gruninger, M. & Fox, M.S. (1995a). Methodology for the design and evaluation of ontologies. In. *IJCAI95 Workshop on Basic Ontological Issues in Knowledge Sharing*. Montreal, Canada. August 19-20.

Guarino, N., Masolo, C., & Vetere, G. (1999). OntoSeek: Content-based access to the Web. *IEEE Intelligent Systems*, May/June, 70-80.

Guha, R.V. (1991). *Contexts: A formalization and some applications*. Ph.D. Thesis, Stanford University.

Hovy, E. (1998). Combining and standardizing large-scale: practical ontologies for machine learning and other uses. In. *Proc. 1st Intl. Conf. On Language Resources and Evaluation*, Granada, Spain, May 1998.

Jannink, J. & Wiederhold, G. (1999). Ontology maintenance with an algebraic methodology: A case study. In. *Proceedings of AAAI workshop on Ontology Management*, July, 1999.

Jannink, J., Srinivasan, P., Verheijen, D. & Wiederhold, G. (1998). Encapsulation and Composition of Ontologies. In. *Proc. AAAI Workshop on Information Integration, AAAI Summer Conference*, Madison WI, July 1998.

Kashyap, V. & Sheth, A. (1996). Semantic and schematic similarities between database objects: A context-based approach. *International Journal on Very Large Databases*, 5(4), 276-304.

Kim, W. & Seo, J. (1991). Classifying schematic and data heterogeneity in multidatabases systems. *IEEE Computer*, 24(2), 12-18.

Lehmann, F. & Gohn, A.G. (1994). The EGG/YOLK reliability hierarchy: Semantic data integration using sorts with prototypes. In., *Third International ACM Conference on Information and Knowledge Management (CIKM-94)*, New York, ACM Press.

Li, W.S. (1995). Knowledge gathering and matching in heterogeneous databases. In. *AAAI Spring Symposium on Information Gathering from Distributed, Heterogeneous Environment.* Palo Alto, California,

McCarthy, J. (1993). Notes on formalizing context. *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, AAAI, 1993.

McGuinness, D.L., Fikes, R., Rice, J. & Wilder, S. (2000). An environment for merging and testing large ontologies. In. *Proc. 7th Intl. Conf. On Principles of Knowledge Representation and Reasoning (KR2000)*, Colorado, USA, April 2000.

Mena, E., Illarramendi, A., Kashyap, V. & Sheth, A.P. (1999). OBSERVER: An approach for query processing in global information systems based on interoperation across pre-existing ontologies. In. *Proceedings of the First IFCIS International Conference on Cooperative Information Systems (CoopIS'96)*, Brussels, Belgium, June 19-21, 1996

Mitra, P., Wiederhold, G. & Kersten, M. (2000). A Graph-Oriented Model for articulation of Ontology Interdependencies. In *Proceedings of Conference on Extending Database Technology, (EDBT 2000)*, Konstanz, Germany, Mar. 2000.

Mitra, P., Wiederhold, G. & Jannink, J. (1999). Semi-automatic Integration of knowledge Sources. In *Proceedings of Fusion 99.* Sunnyvale CA, July, 1999.

Noy, N. F. & Musen, M.A. (1999). *SMART: Automated support for ontology merging and alignment*. SMI Report Number: SMI-1999-0813.

Noy, N.F. & Musen, M.A. (2000). PROMPT: Algorithm and tool for automated ontology merging and alignment. In. *Proc. 17th Natl. Conf. On Artificial Intelligence (AAAI2000)*, Austin, TX, July/August 2000.

Palopoli,L., Pontieri, L., Terracina, G. & Ursino, D. (2000). Intensional and extensional integration and abstraction of heterogeneous databases. *Data & Knowledge Engineering*, 35, 201-237.

Roux, C., Proux, D., Rechenmann, F. & Julliard, L. (2000). An ontology enrichment method for a pragmatic information extraction system gathering data on genetic interactions. In. *Proceedings of the First Workshop on Ontology Learning (OL-2000) in conjunction with the 14th European Conference on Artificial Intelligence (ECAI 2000)*. August, Berlin, Germany.

Sheth, A. & Larson, J. (1990). Federated database systems for managing distributed, heterogeneous and autonomous databases. *ACM Computing Surveys*, 22(3).

Sheth, A. & Kashyap, V. (1992). So for (schematically), yet so near (semantically). In. *Proc. IFIP DS-5 Semantics of Interoperable Database Systems.* Lorne, Australia: Chapman & Hall.

Smith, D.C. (1990). *Plisp Users Manual.* Apple Computer, August, 1990.

Sofia Pinto, H., Gomez-Perez, A. & Martins, J.P. (1999). Some issues on ontology integration. In. *Proceedings of IJCAI-99 Workshop on Ontologies and Problem-Solving Methods: Lessons Learned and Future Trends*, in conjunction with the Sixteenth International Joint Conference on Artificial Intelligence, August, Stockholm, Sweden.

Srinivasan, U.; Ngu, A.H.H. & Gedeon, T. (2000). Managing heterogeneous information systems through discovery and retrieval of generic concepts. *Journal of the American Society for Information Science*, 51(8), 707-723.

Stumme, G., Studer, R. & Sure, Y. (2000). Towards an order-theoretical foundation for maintaining and merging ontologies. In. *Proc. Referenzmodellierung 2000*, Siegen, Germany, October 12-13, 2000.

Tamma, V. A. M. & Bench_Capon, T.J.M. (2000). Supporting different inheritance mechanisms in ontology representations. In. *Proceedings of the First Workshop on Ontology Learning (OL-2000) in conjunction with the 14th European Conference on Artificial Intelligence (ECAI 2000)*. August, Berlin, Germany.

Todirascu, A., Beuvron, F., Galea, D. & Rousselot, F. (2000). Using description logics for ontology extraction. In. *Proceedings of the First Workshop on Ontology Learning (OL-2000) in conjunction with the 14th European Conference on Artificial Intelligence (ECAI 2000)*. August, Berlin, Germany.

Uschold, M. (2000). Creating, integrating and maintaining local and global ontologies. In. *Proceedings of the First Workshop on Ontology Learning (OL-2000) in conjunction* *with the 14th European Conference on Artificial Intelligence (ECAI 2000)*. August, Berlin, Germany.

Uschold, M. & Gruninger, M. (1996). Ontologies: principles, methods, and applications. *Knowledge Engineering Review*, 11(2), 93-155.

Visser, P.R.S. & Cui, Z. (1998). On accepting heterogeneous ontologies in distributed architectures. In. *Proceedings of the ECAI98 workshop on applications of ontologies and problem-solving methods*. Brighton, UK, 1998.

Visser, P.R.S. & Tamma, V.A.M. (1999). An experience with ontology clustering for information integration. In. *Proceedings of the IJCAI-99 Workshop on Intelligent Information Integration in conjunction with the Sixteenth International Joint Conference on Artificial Intelligence*, Stockholm, Sweden, July 31, 1999.

Visser, P.R.S., Jones, D.M., Beer, M., Bench-Capon, T.J.M., Diaz, B. & Shave, M.J.R. (1999). Resolving Ontological Heterogeneity in the KRAFT Project. In., *DEXA-99, 10th International Conference and Workshop on Database and Expert Systems Applications,* Florence, Italy, August, 1999.

Weinstein, P.C. & Birmingham, P. (1998). Creating ontological metadata for digital library content and services. *International Journal on Digital Libraries*, 2(1), 19-36.

Weinstein, P.C. & Birmingham, P. (1999). Comparing Concepts in Differentiated Ontologies. In. *Proceedings of the Twelfth Workshop on Knowledge Acquisition, Modeling and Management (KAW99).* October 1999, Banff, Alberta, Canada.

Wiederhold, G. (1994). An algebra for ontology composition. In. *Proceedings of 1994 Monterey Workshop on formal Methods* (pp. 56-61), U.S. Naval Postgraduate School, Monterey CA, 1994.

Wiederhold, G. (1994a). Interoperation, mediation, and ontologies. In. *Proceedings International Symposium on Fifth Generation Computer Systems (FGCS), Workshop on Heterogeneous Cooperative Knowledge-Bases*, Vol.W3, pp.33--48, ICOT, Tokyo, Japan, Dec. 1994.

Williams, A.B. & Tsatsoulis, C. (2000). An instance-based approach for identifying candidate ontology relations within a multi-agent system. In. *Proceedings of the First Workshop on Ontology Learning (OL-2000) in conjunction with the 14th European Conference on Artificial Intelligence (ECAI 2000)*. August, Berlin, Germany.