The Role of Ontologies in eCommerce

Y. Ding[2], D. Fensel[2], M. Klein[1], B. Omelayenko[1], and E. Schulten[1]

[1] Vrije Universiteit Amsterdam VUA, Division of Mathematics and Informatics,
De Boelelaan 1081a, NL-1081 HV Amsterdam, The Netherlands

[2] Leopold-Franzens Universität Innsbruck, Institut für Informatik,
Technikerstrasse 25, A-6020 Innsbruck, Austria

Abstract. Web technology is starting to penetrate many aspects of our daily life and its importance as a medium for business transactions will grow significantly during the next few years. In terms of market volume, B2B will be the most interesting area where new technology will lead to drastic changes in established customer relationships and business models. Simple and established one2one trading relationships will be replaced by open and flexible n2m relationships between customers and vendors. However, this new flexibility in electronic trading also creates serious challenges for the parties who want to realize it. The main problem is the heterogeneity of information descriptions used by vendors and customers. Product descriptions, catalog formats, and business documents are often unstructured and non-standardized. Intelligent solutions that help to mechanize the process of structuring, standardizing, aligning and personalizing are key requisites to successfully overcoming the current bottlenecks of eCommerce and enabling its further growth. This paper discusses the main problems of information integration in this area and describes how ontology technology can help solve many of them.

Introduction

eCommerce in business-to-business (B2B) is not a new phenomenon. Initiatives to support electronic data exchange in the business processes between different companies already existed in the 1960s. In order to exchange business transactions the sender and receiver have to agree on a common standard (a protocol for transmitting the content and a language for describing the content). In general, the automation of business transactions has not reached the expectations of its propagandists. Establishing an eCommerce relationship requires a serious investment and it is limited to a predefined number of trading partners. It also is limited to a specific type of extranet that needs to be set up for mechanizing the business relationships.

Web-enabled eCommerce helps users contact a large number of potential clients without running into the problems associated with implementing numerous communication channels. However, enabling flexible and open eCommerce requires contending with other serious problems. One has to deal with the question of heterogeneity in the *product*, *catalogue*, and *document* description standards of the trading partner. The effective and efficient management of different styles of description becomes a key obstacle for this approach.

Web-enabled eCommerce needs to be open to a large numbers of suppliers and buyers. Its success is closely related to its ability to mediate a large number of business transactions. Web-enabled eCommerce provides its users with one key advantage - they can communicate with many customers through a single communication channel. This open, flexible, and dynamic channel reduces the number of special-purpose communication links for its user community. However, in order to provide this service, there must be solutions that solve the significant normalization, mapping, and updating problems for the clients. A successful approach has to deal with numerous aspects. It has to integrate various hardware and software platforms and provide a common protocol for information exchange. However, the real problem is the openness, heterogeneity and dynamic nature of the exchanged content. There are at least three levels at which this heterogeneity arises: the *content* level, the level of *product catalogs structures*, and the level of *document structures*.

- The actual **content** of the exchanged information needs to be modelled. Many different ways to categorize and describe products have evolved over time. Vendors often have their own way of describing their products. Structuring and standardizing the product descriptions, ensuring the different players can actually communicate with each other, and allowing customers to find the products they are looking for are significant tasks in B2B eCommerce.
- eCommerce is about the electronic exchange of business information. Product descriptions are just one element, but they are the building blocks of an electronic **catalog**, together with information about the vendor, the manufacturer, the lead- time etc. Furthermore, a catalog provider needs to include quality control information, such as the version, date, and identification number of the catalog. If two electronic catalogs are involved, the structure of these catalogs must be aligned as well.
- The next step in the process is the actual use of the catalog. A buyer may want to send a purchase order, after retrieving the necessary information from a catalog. The vendor has to reply with a confirmation, and then the actual buying process begins. A common language is needed in order for the buyer and the vendor to read and process each other's **business documents**. Marketplace software designers like Commerce One developed their structures based on xCBL[1]. This provides a large collection of document structures reflecting different aspects of a trading process. Aligning these document structures with other document definitions from, for example, Ariba (cXML[2]), is not certainly a trivial task.

[1] http://www.xcbl.org

[2] http://www.cXML.org

The first type of mismatch that arises primarily concerns with the real-world semantics of the exchanged information. People describe the same products in different ways. The second and third types arise in relation to the syntactical structure of the exchanged information. These problems are more serious, reflecting the dynamic nature of eCommerce. New players arise, new standards are proposed, and new products and services enter the marketplace. No static solution can deal with this constantly changing and evolving situation. Given the requirements there is only one IT technology available that can provide at least a partial solution - ontology. This technology and its promises for eCommerce are examined in the reminder of this paper.

Ontology-based solution paths. Ontologies (cf. [Fensel, 2001]) are a key enabling technology for the semantic web. They interweave human understanding of symbols with machine-processability. Ontologies were developed in Artificial Intelligence to facilitate knowledge sharing and reuse. Since the early nineties, ontologies have become a popular research topic and a subject of study by several Artificial Intelligence research communities, including Knowledge Engineering, Natural Language Processing and Knowledge Representation. More recently, the concept of ontology has spread to other fields, such as intelligent information integration, cooperative information systems, information retrieval, electronic commerce, and knowledge management. The reason ontologies are becoming so popular is primarily due to what they promise: a shared and common understanding of a domain that can be communicated between people and application systems. In essence, Ontologies are formal and consensual specifications of conceptualizations that provide a shared and common understanding of a domain, an understanding that can be communicated across people and application systems. Ontologies glue together two essential factors that help to bring the Web to its full potential:

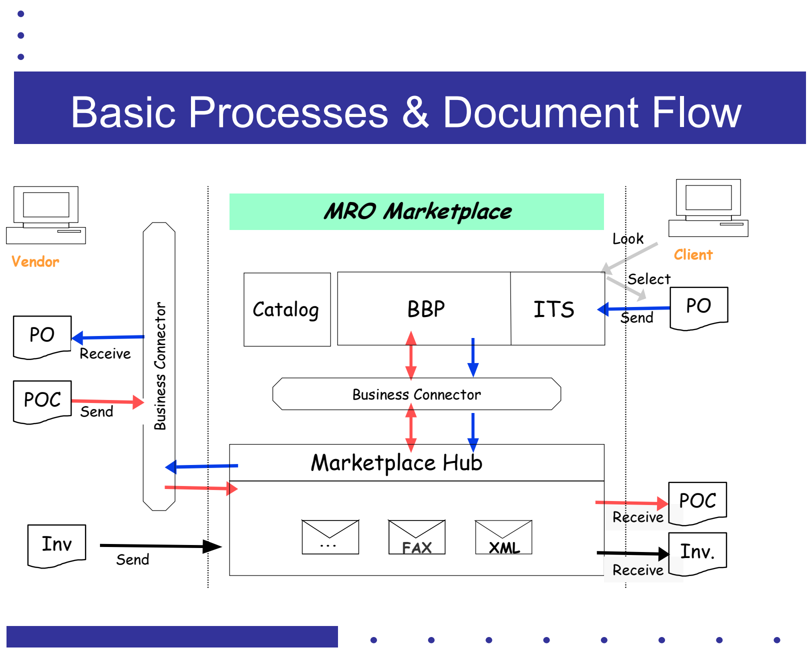
- Ontologies define formal semantics for information that allows information processing by a computer.
- Ontologies define real-world semantics that make it possible to link machine-processable content with meaning for humans based on consensual terminologies.

The latter aspect makes ontology technology especially interesting. Ontologies must have a *network architecture* and be *dynamic*. Ontologies deal with heterogeneity in space and development in time. Ontology is networks of meaning where, from the very beginning, heterogeneity is an essential requirement. Tools for dealing with conflicting definitions and strong support in interweaving local theories are essential in order to make this technology workable and scalable. Ontologies are used as a method of exchanging meaning between different agents. They can only provide this if they reflect an inter-subjectual consensus. By definition, ontologies can only be the result of a social process. For this reason, ontologies cannot be understood as a static model. An ontology is as much required for the exchange of meaning as the exchange of meaning may influence and modify an ontology. Consequently, evolving ontologies describe a process rather than a static model. Indeed, evolving over time is an essential requirement for useful ontologies. As daily practice constantly changes, ontologies that mediate the information needs of these processes must have strong support in versioning and must be accompanied by process models that help to organize consensus.

Contents of the paper. The structure of this paper reflects the issues discussed above. In Section 2, we explore the role of standardization in eCommerce, as openness cannot be achieved without agreements. In Section 3 and 4, we explain the need for heterogeneity in these descriptions. Section 3 focuses on heterogeneity in space (i.e. on aligning standards), and Section 4 focuses on heterogeneity in time (i.e. on evolving these standards). Section 5 covers an aspect we have not yet mentioned - that Ontologies are structures for describing actual content. This section also describes methods and tools to allow this in a scalable and economic fashion. Finally, conclusions are provided in Section 6.

Openness: Harmonization and Standardization in eCommerce

A fundamental premise - and the major economic drive - behind eCommerce is that labor intensive and time consuming human interactions can be replaced with (semi-) automated internet-enabled processes. Looking at actual eCommerce solutions, we see rather simple applications for the final customers, such as product search and selection without the help of a sales representative. There are slightly more sophisticated solutions between enterprises, such as server-to-server communication for enterprise inventory management. Despite these solutions, the slower-than-expected adoption of electronic buying and the bankruptcy of many dotcoms point to the complexity of replacing the human element. Of course, this is not difficult to understand. In the human world, dialog is structured by grammatical, semantic, and syntactic rules that are expressed in a shared context of social and cultural conventions. The young eCommerce world is lacking this rich consensual background, and we are still far from achieving the vision of a Universe of Network-Accessible Information - as the W3C defines the Web. The need for consensus in a trading community arises on many different levels, which is reflected in the different areas of focus of these harmonization initiatives. Fig. 2 illustrates the basic processes and documents exchanged through an e-marketplace based on SAP technology. Depending on the level of sophistication, the Business Connector allows integration with the back-end system of the business partners and the billing process is automated through the marketplace. Looking from a business perspective, we first encounter the level of the basic building blocks of any commercial transaction; the descriptions of the products and services themselves. Clearly, without agreement on the name of an item to be bought or sold, any degree of transaction automation becomes quite complex. We then arrive at the level where these descriptions are represented in an electronic catalogue. The catalogue requires specific content and an agreed format because the many-to-many communication in an electronic marketplace presupposes a shared catalogue. Finally, there is the level where the electronic catalogue is actually used. Here the business processes and the business



Basic Processes & Document Flow

MRO Marketplace

Vendor

Client

Look

Select

Send

PO

Catalog

BBP

ITS

Business Connector

PO

Receive

POC

Send

Business Connector

Marketplace Hub

...

FAX

XML

Inv

Send

POC

Receive

Inv.

Receive

Fig. 2 Basis processes and documents exchanged through an e-marketplace.

documents involved have to be aligned. Consider the straightforward example of purchasing a non-stock item, such as writing paper, through an electronic marketplace. The business partners at a minimum need to be able to exchange a Purchase Order and a Purchase Order Confirmation and in a more sophisticated application. the Billing process, Order Status Tracking, and the Goods Receipt Process are included as well. Hence, business processes and documents throughout the supply chain are involved in this alignment process.

We will now discuss standardization and harmonization initiatives that have a significant impact on the development of electronic business. First, Table 1. provides a summary and classification of the product and service standards.

Table 1. Survey of Product and Service Standards

Name	Design perspective	Main classification concept	Major use	Domain
ecl@ss, www.eCl@ss.de	Supply side	Material of construction.	Building blocks for electronic catalogues in B2B marketplaces.	Intending to cover services and products, but current focus on products. The automotive and the electrical industry are strongly represented.
HS: Harmonised System, www.wcoomd.org	Supply side	Material of construction.	Collection of customs duties and international trade statistics.	Intending to cover services and products, but strong focus on products.
NAICS/NAPCS: North American Industry Classification System/ North American Product Classification System, www.census.gov	NAICS: supply side NAPCS: demand side	NAICS: Production process. NAPCS: not yet decided	NAICS & NAPCS: Statistics on a.o. productivity, unit labor.	NAICS: intending to cover services and products, but strong focus on products. NAPCS: intending to cover services and products, first focus will be on services because they have in the past been neglected by classification systems.
RosettaNet, www.rosettanet.org	Supply side	Product category	Building blocks for electronic catalogues in B2B marketplaces	Products in IT industry, automotive industry, consumer electronics and telecommunications industries.
SYMAP/CVP: Système d'Information pour les Marchés Publics / Common Procurement Vocabulary, www. simap.eu.int	Supply side	Industry of origin	Purchasing in public sector.	Intended to cover services and products, but focus on products.

Table 1. Survey of Product and Service Standards

Name	Design perspective	Main classification concept	Major use	Domain
UNSPSC: United Nations Standard Products and Services Codes, www.un-spsc.net	Supply side	Product category	Building blocks for electronic catalogues.	Intending to cover services and products, but currently very shallow.

These content standards are complemented by proposals for the alignment of business processes. Examples are: BizTalk, www.biztalk.org and www.microsoft.com/biztalk/; Commerce XML: cXML, www.cxml.org; Electronic Business XML: ebXML, www.ebxml.org; Open Buying on the Internet Consortium OBI, www.openbuy.org; Open Applications Group Integration Specification: OAGIS, www.openapplications.org; Organization for the Advancement of Structured Information Standards: OASIS, www.oasis-open.org; Rosettanet, www.rosettanet.org; UN/CEFACT, www.unece.org/cefact; and XML Common Business Library: xCBL, www.xcbl.org.

In order to evlevate electronic business beyond the buying and selling of mere commodities such as a desktop computer or a CD, customers need a generic classification system with a high level of detail. It is clear that the current classification systems are built for different purposes, with different classification concepts and structures, and cover different domains. Some do not provide the level of detail required for an electronic catalogue, others neglect the important area of services, and most are developed from a supply instead of a demand perspective. In short, a universal product and service classification system that is useful for a customer dealing with an electronic catalog does not exist. Therefore, the question of the compatibility between these classification systems is a crucial one. This will be addressed in the following section.

Sophisticated electronic commerce also presupposes that the business processes of the engaged partners are aligned and that the related business documents and catalogues are standardized. We can see the major industry players in this field recognize the importance of consensus and harmonization, and they increasingly ensure compliance with international independent bodies such as the W3C and ebXML.

In an ideal world, all electronic commerce between businesses would utilize one universal standard covering the issues on all the levels that we have discussed in this chapter. Nevertheless, for at least two reasons, this does not look feasible in the real world. First, because business requirements and technology possibilities alter at a rapid rate, and therefore, standards will always be in development. Second, businesses will not wait decades for a global standard to 'arise'. Indeed, notwithstanding the lack of proper standards, many enterprises already engage in electronic business in different ways, utilizing different languages. Multilinguality is not a problem itself; instead, it often allows creativity and refreshing diversity. However, things get trickier when lacking the means for translation. This is exactly the case in many parts of B2B electronic commerce and brings to mind the biblical building of the Tower of Babel.

Flexibility: Alignment of Standards

The heterogeneity of eCommerce cannot be captured by one standard and personalization is needed anyway. Therefore, scalable mediation service between different standards is essential. We will now describe how Ontology mapping methods can contribute a solution to this problem, focusing on the alignment of business documents and product classifications.

Alignment of Document Standards

The B2B area operates with substantial number of different business documents. There are several non-XML plain text document standards already accepted and widely used by the industry. The first is the well-known EDIFACT format approved by the United Nations Economic Commission for Europe.[3] An EDIFACT document is presented with complicated formatted text not understandable by a non-specialist. Several text wrappers able to translate an EDIFACT catalog into XML are now available. For example, the XML-EDIFACT[4] wrapper transfers EDIFACT documents into their XML representation and vice versa. Another non-XML standard is ISO 10303 [ISO, 2000] (also known as STEP) that is an International Standard for the computer-interoperable representation and exchange of product data. It contains a rich set of modeling primitives that allow building hierarchical product specifications. ISO has developed an XML syntax for STEP that is now

[3] http://www.unece.org/trade/untdid/welcome.htm

[4] http://www.xml-edifact.org/

being standardized as part 28 of the ISO 10303 specification.

Table 2. A fragment of the xCBL and cXML formats.

CatalogSchema	PunchOutOrderMessage
SchemaVersion> 1.0	BuyerCookie> 342342ADF
SchemaStandard> UNSPSC	ItemIn
SchemaCategory	quantity> 1
CategoryID> C43171801	ItemID
ParentCategoryRef> C43170000	SupplierPartID> 1234
CategoryName> Computers	ItemDetail
CategoryAttribut	UnitPrice
AttributeName> Processor Speed	Money
CatalogData	currency> USD
Product	Money> 1250
SchemaCategoryRef> C43171801	Description> Armada M700 PIII 700
ProductID> 140141-002	UnitOfMeasure> EA
Manufacturer> Compaq	Classification
CountryOfOrigin> US	domain> SPSC
ShortDescription> COMPAQ Armada	Classification> 43171801
AM700PIII 700	ManufacturerPartID> 140141-002
LongDescription> This light, …	ManufacturerName> Compaq
ObjectAttribute	
AttributeID> Warranty, years	
AttributeValue> 1	
ProductVendorData	
PartnerRef> Acme_Laptops	
VendorPartNumber> 12345	
ProductPrice	
Amount> 1250	
Currency> USD	
(a) xCBL	(b) cXML

In addition to legacy standards, there exist a number of recently processed XML standards. Besides the common serialization language of XML, they significantly differ from the underlying document models.

One typical example of these differences is the diverse ways to represent a list of products in a purchase order when the products are grouped per transaction or in delivery order where the products are grouped per container. Document integration requires regrouping of the records.

Conceptually equivalent properties can be named and re-grouped in different ways. For example, consider the fragments of the document structures represented in Table 2. for (a) xCBL[5] and (b) cXML[6] standards. The tags in the figure represent the elements of the structure and roughly correspond to the XML tags, which describe the instance documents. The values of the tags are displayed in the italics to illustrate the intended meaning of the tags. Graphical tags nesting represent the part-of relation. We see the structures provide slightly different representations for very

[5] http://www.commerceone.com/solutions/business/content.html

[6] http://www.ariba.com/

similar content. Both standards introduce internal product IDs and import the manufacturer's product IDs and names. They also contain pricing information, product descriptions, and a reference to a certain content standard.

Finally, the documents tend to be substantially different in capturing and representing is-a relations. For example, the fact that an address is either a physical address or a legal address (both are subclasses from a generic address) can be represented as tag nesting (making a tag sequence <! ELEMENT Address (PhysicalAddress | LegalAddress)>) explicitly capturing the is-a relationship at the schema level or with a certain attribute value assigned to element (<!ATTLIST Address type (Physical | Legal) #REQUIRED>) where value "Physical" being assigned to attribute type would specify that the address is a physical one. The second way encodes the is-a relation with attribute values at the level of values. The Ontology-mediated business integration framework [Omelayenko, 2002(b)] specifically addresses these issues by performing three steps of document integration.

First, document conceptual models are extracted from document DTDs, explicitly representing objects with string (#PCDATA) properties. This can be done automatically following existing work [Mello and Heuser, 2001]. It is important to mention that element and attribute names tend to be reused in DTDs with different associated meaning. For example, tag value may represent several completely different values if assigned to different elements (price value and document revision value). These specific cases should be separated during the model extraction.

Second, these document models are mapped to a mediating unified conceptual model. This is done by means of RDFT mapping meta-ontology that specifies maps between conceptual models in RDF Schema consisting of bridges. Each bridge represents a certain relation between the concepts being mapped and this relation is then interpreted by inference engine that uses these bridges. The bridges link (several) source and (several) target roles, where each role stands for either a class, a property being attached to a specific class, or property value. Such bridge structure allows dealing with the heterogeneity in the modeling described above.

Third, the conceptual models and RDFT maps can then be easily converted to Prolog (See Figure 3 for a sample) to perform different reasoning tasks like validation checking for the maps.

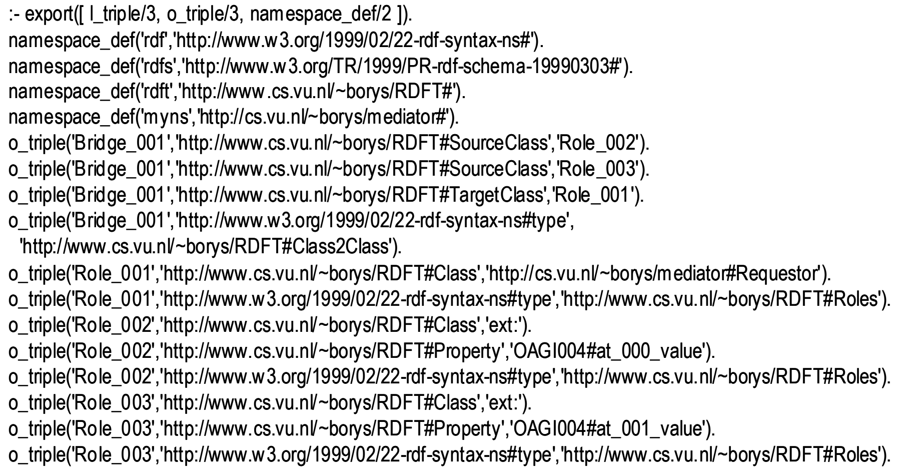
To summarize, the document needs to be integrated stepwise via a mediating conceptual model to overcome the tremendous heterogeneity in underlying document models. The maps linking these models need to be capable of dealing with these differences and inference can be used to analyze the maps.

Alignment of Content Standards

Different eCommerce applications naturally use different content standards. For example, the UNSPSC standard mentioned earlier is primarily targeted at vendor's needs, while the ecl@ss standard largely represents buyer's needs. Therefore, different content standards need to be aligned and mapped in a scalable and efficient way [Fensel, 2001].

Mapping the content standards by specifying pairs of equivalent categories is not always possible due to different principles used to aggregate the products into categories of the same abstraction level. For this reason, for example, mapping UNSPSC to ecl@ss includes creating many-to-many bridges regrouping the products to categories. There are also prominent examples of aligning specific content standards to more generic ones. These mappings are manually created and verified, and sometimes have normative status. We can point to the UNSPSC crosswalk files linking it to NAICS and several other standards used for reporting and statistical purposes. Another example is mapping RosettaNet standard that specifies 445 categories and 2660 attributes for the electronic components to UNSPSC. Rosetta Net is specific in describing these components, but it does not cover concepts left beyond the primary focus. The mapping links only 136 UNSPSC elements out of more than 17,000 - most of which belong to the bottom level in the UNSPSC hierarchy - and thus expanding these 136 categories with all the Rosetta Net classes and attributes. The specific standards are very precise in describing the items on which they are focused. At the same time, they are even shallower than the generic standards in describing the things that lay beyond their focus.

Essentially, the content standards can be seen as lightweight ontologies containing hierarchies of classes with (possibly) several attributes attached to each class. They still have quite limited expressiveness to be regarded as logical theories, and thus form a simple playground for Ontology mapping and integration techniques. There exist several approaches for representing the maps between different ontologies ranging from UML-based representations like CWM [CWM, 2001] to those based on mapping ontologies represented in RDF Schema like RDFT [Omelayenko, 2002(b)] or MAFRA [Maedche et al., 2002]. However, the standards represent little formal semantics with no explicitly represented axioms or formal relations. As a result, it is difficult to perform inference over the standard and maps



```
:- export([ l_triple/3, o_triple/3, namespace_def/2 ]).
namespace_def('rdf','http://www.w3.org/1999/02/22-rdf-syntax-ns#').
namespace_def('rdfs','http://www.w3.org/TR/1999/PR-rdf-schema-19990303#').
namespace_def('rdft','http://www.cs.vu.nl/~borys/RDFT#').
namespace_def('myns','http://cs.vu.nl/~borys/mediator#').
o_triple('Bridge_001','http://www.cs.vu.nl/~borys/RDFT#SourceClass','Role_002').
o_triple('Bridge_001','http://www.cs.vu.nl/~borys/RDFT#SourceClass','Role_003').
o_triple('Bridge_001','http://www.cs.vu.nl/~borys/RDFT#TargetClass','Role_001').
o_triple('Bridge_001','http://www.w3.org/1999/02/22-rdf-syntax-ns#type',
  'http://www.cs.vu.nl/~borys/RDFT#Class2Class').
o_triple('Role_001','http://www.cs.vu.nl/~borys/RDFT#Class','http://cs.vu.nl/~borys/mediator#Requestor').
o_triple('Role_001','http://www.w3.org/1999/02/22-rdf-syntax-ns#type','http://www.cs.vu.nl/~borys/RDFT#Roles').
o_triple('Role_002','http://www.cs.vu.nl/~borys/RDFT#Class','ext:').
o_triple('Role_002','http://www.cs.vu.nl/~borys/RDFT#Property','OAGI004#at_000_value').
o_triple('Role_002','http://www.w3.org/1999/02/22-rdf-syntax-ns#type','http://www.cs.vu.nl/~borys/RDFT#Roles').
o_triple('Role_003','http://www.cs.vu.nl/~borys/RDFT#Class','ext:').
o_triple('Role_003','http://www.cs.vu.nl/~borys/RDFT#Property','OAGI004#at_001_value').
o_triple('Role_003','http://www.w3.org/1999/02/22-rdf-syntax-ns#type','http://www.cs.vu.nl/~borys/RDFT#Roles').
```

Fig. 3 RDFT Map in Prolog.

between them, as well as to specify formal interpretation of the maps. The categories are mainly interpreted in terms of product descriptions classified to each specific category. The categories possess mostly extensional information and they are interpreted in terms of instance data. Hence, any formal way of mapping the standards should be augmented with instance processing techniques linking the maps to actual product descriptions. A case study described in [Omelayenko, 2002(a)] presents the use of two Naïve-Bayes classifiers trained on two datasets that employ instance information for this problem.

To summarize, manual mapping of content standards is possible in some cases leaving quite a demand for automated mapping techniques. The categories are primarily interpreted in terms of instance product descriptions; the standards are lacking formal relations and axioms and as a result ontology-based mapping approaches should be improved by machine learning algorithms.

Dynamics: Versioning of Standards

The dynamic and open character of eCommerce requires that classification standards, as described in Section , are extended or adapted when new products or services arise. However, this presents new problems such as how to manage classification hierarchies that change over time, in such a way that the old and new versions can be used intermixed. If no special arrangements are taken, the evolution of standards might cause operability problems that will seriously hamper eCommerce applications. Solutions are required to allow changes to classification standards without making their present use invalid. In this section, we will first look at what typical changes in the UNSPSC classification system. We will then describe the requirements for a change management system, and explain some methods and tools for the versioning of ontologies.

Changes in UNSPSC

The high change rate of the classification hierarchies and the way in which those changes are handled is a serious threat for electronic commerce. For example, an examination of UNSPSC reveals:

- There were 16 updates between 31 January 2001 and 14 September 2001;
- Each update contained between 50 and 600 changes;
- In 7,5 month, more than 20% of the current standard is changed!

Although some parts of the UNSPSC schema might be more stable than other parts, it is clear this number of changes cannot be ignored. Such a high change rate can quickly invalidate many of the actual classifications of products. For example, the product “Binding elements” in version 8.0 is removed from the standard and three new products are added in version 8.1 (“Binding spines or snaps”, “Binding coils or wire loops”, and “Binding combs or strips”). This means that all products that were classified as “Binding elements” are unclassified under the new version. This is a serious problem because of the high costs for producing the right classifications for products. Moreover, if companies use local extensions of the standard they have to adapt these extensions to new versions as well. A versioning mechanism that allows partly automatic transformation of data between content standard versions is essential.

An effective versioning methodology should take care of the different types of changes in ontologies, as those might have different effects on the compatibility of data that is described by them [Klein & Fensel, 2001]. An analysis of differences between several versions of content standards yielded the following typical changes: class-title changes, additions of classes, relocations of classes in the hierarchy (by moving them up or down in the hierarchy or horizontally), relocations of a whole subtree in the hierarchy, merges of two classes (in two variants: two classes become one new class, or one class is appended to the other class), splits of classes, and pure deletions. However, current versioning techniques for content standards are often quite simple. In UNSPSC, for example, all changes are encoded as additions deletions, or edits (title changes). This means the relocation of a subtree is specified as a sequence of “delete a list of classes” and “add a list of classes”.

Requirements for content standard versioning

The need to cope with changing data structures is not new in computer science. Much of the research in database technology has focused on the topic of database schema evolution. However, while there are quite a few similarities between Ontology versioning and database schema evolution, there are also many differences (For a detailed discussion, see [Noy & Klein, 2002]). An important difference is that with ontologies the distinction between data and schema is not as clear as it is in databases. Ontologies themselves - and not just the data - are often used in applications, (i.e. as controlled vocabularies, or navigation structures). The UNSPSC standard, for example, might be used in an application to structure the website of sales company. In addition, ontologies are even more distributed by nature than are databases. We often have a clear picture of the locations where changes might have an impact on distributed databases. However, with content standards like UNSPSC the author of the Ontology has absolutely no clue as to which applications use the Ontology. It is not possible to synchronize changes with all users.

Due to these differences, the traditional distinctions [Roddick, 1995] between evolution (new schemas that are backward compatible) and versioning (multiple views of the data via different versions) and between reading and updating compatibility are not very relevant to ontology versioning. Changes to ontologies will occur and some are likely to cause incompatibilities. Therefore, versioning methodologies for ontologies cannot guarantee prevention of any information loss. However, it should make the effects of changes explicit. The management of changes is the key issue in support for evolving ontologies.

The mechanisms and techniques to manage those changes to ontologies should aim at achieving maximal interoperability with existing data and applications. This means that it should retain as much information and knowledge as possible without deriving incorrect information. This methodology should feature the following:

- **Identification mechanism**: for every use of a concept or a relation, a versioning framework should provide an unambiguous reference to the intended definition

- **Change specification mechanism**: the relation of one version of a concept or relation to other versions of that construct should be made explicit, both by specifying the ontological relation (i.e. subclass of) and the intention of the change (i.e. replacement)
- **Transparent access**: methods for rendering a valid interpretation to as much data as possible (i.e. automatically translating and relating the versions and data sources to the maximum possible extent).

Ontology comparison techniques can help companies find and describe the differences between the new versions of the standards and the old versions that were used to classify data. Descriptions of the semantics of the discovered changes can facilitate the transformation of data classification. For example, in the most trivial case, it can specify that a new version is a combination of two other classes; all products that were classified under the old classes can then be classified under the new class. Complicated specifications of the logical consequences, possibly with approximations, will further decrease the negative effects of the evolution of content standards.

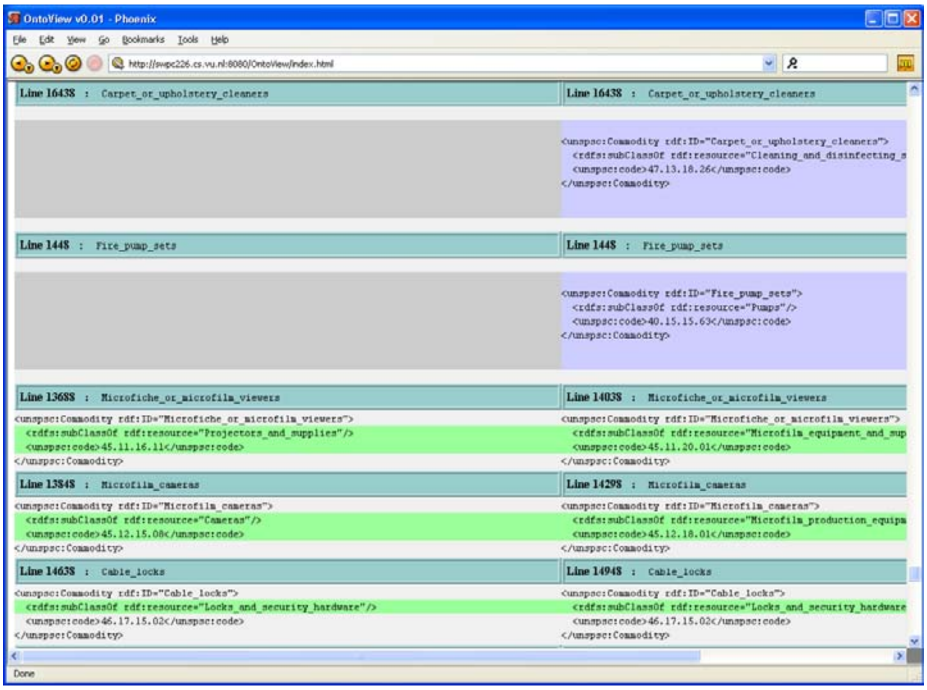
Tools for ontology versioning

OntoView [Klein et al., 2002] is a change management tool for ontologies. The main function of OntoView is to provide a transparent interface to arbitrary versions of ontologies. To achieve this it maintains an internal specification of the relation between the different variants of ontologies. This specification consists of three aspects: the meta-data about changes (author, date, time etc.), the conceptual relations between versions of definitions in the ontologies, and the transformations between them. This specification is partly derived from the versions of ontologies themselves, but it also uses additional human input about the meta-data and the conceptual effects of changes.

To help the user to specify this information, OntoView provides the utility to compare versions of ontologies and highlight the differences. This helps in finding changes in ontologies, even if those have occurred in an uncontrolled way (i.e., possibly by different people in an unknown order). The comparison function is inspired by UNIX diff, but the implementation is quite different. Standard diff compares file version at line-level, highlighting the lines that textually differ in two versions. OntoView, in contrast, compares version of ontologies at a structural level, showing which definitions of ontological concepts or properties are changed.

The comparison function distinguishes between the following types of change:

- Non-logical change (i.e. in a natural language description). These are changes in the label of a concept or property, or in comments inside definitions.
- Logical definition change. These changes in the definition of a concept affects its formal semantics. Examples of such changes are alterations of subclass statements or changes in the domain or range of properties. Additions or deletions of local property restrictions in a class are also logical changes.


OntoView v0.01 - Phoenix
File Edit View Go Bookmarks Tools Help
http://swpc226.cs.vu.nl:8080/OntoView/index.html
Done

```
Line 16438 : Carpet_or_upholstery_cleaners                     Line 16438 : Carpet_or_upholstery_cleaners

                                                                 <unspsc:Commodity rdf:ID="Carpet_or_upholstery_cleaners">
                                                                   <rdfs:subClassOf rdf:resource="Cleaning_and_disinfecting_s
                                                                   <unspsc:code>47.13.18.26</unspsc:code>
                                                                 </unspsc:Commodity>

Line 1448 : Fire_pump_sets                                      Line 1448 : Fire_pump_sets

                                                                 <unspsc:Commodity rdf:ID="Fire_pump_sets">
                                                                   <rdfs:subClassOf rdf:resource="Pumps"/>
                                                                   <unspsc:code>40.15.15.63</unspsc:code>
                                                                 </unspsc:Commodity>

Line 13688 : Microfiche_or_microfilm_viewers                    Line 14038 : Microfiche_or_microfilm_viewers
<unspsc:Commodity rdf:ID="Microfiche_or_microfilm_viewers">     <unspsc:Commodity rdf:ID="Microfiche_or_microfilm_viewers">
  <rdfs:subClassOf rdf:resource="Projectors_and_supplies"/>       <rdfs:subClassOf rdf:resource="Microfilm_equipment_and_sup
  <unspsc:code>45.11.16.11</unspsc:code>                          <unspsc:code>45.11.20.01</unspsc:code>
</unspsc:Commodity>                                             </unspsc:Commodity>
Line 13848 : Microfilm_cameras                                  Line 14298 : Microfilm_cameras
<unspsc:Commodity rdf:ID="Microfilm_cameras">                   <unspsc:Commodity rdf:ID="Microfilm_cameras">
  <rdfs:subClassOf rdf:resource="Cameras"/>                       <rdfs:subClassOf rdf:resource="Microfilm_production_equipm
  <unspsc:code>45.12.15.08</unspsc:code>                          <unspsc:code>45.12.18.01</unspsc:code>
</unspsc:Commodity>                                             </unspsc:Commodity>
Line 14638 : Cable_locks                                        Line 14948 : Cable_locks
<unspsc:Commodity rdf:ID="Cable_locks">                         <unspsc:Commodity rdf:ID="Cable_locks">
  <rdfs:subClassOf rdf:resource="Locks_and_security_hardware"/>   <rdfs:subClassOf rdf:resource="Locks_and_security_hardware
  <unspsc:code>46.17.15.02</unspsc:code>                          <unspsc:code>46.17.15.02</unspsc:code>
</unspsc:Commodity>                                             </unspsc:Commodity>
```

Fig. 4 The result of a comparison of two version of the UNSPSC hierarchy in OntoView.

- Identifier change. This is when a concept or property is given a new identifier (i.e. a renaming).
- Addition of definitions
- Deletion of definitions

Each type of change is highlighted in a different color, and the altered lines are printed in boldface. An example of the visual representation of the result of a comparison is shown in Figure 4. For this picture, a subset of the two versions of the UNSPSC classification was used (i.e. segment 40 till 49 of UNSPSC version 8.0 and 8.4). The figure shows two classes that are added to the new version, two that are moved in the hierarchy (with another superclass and a different code), and one in which the superclass has changed.

The comparison function also allows the user to characterize the conceptual implication of the changes. For the first three types of changes, the user is given the option to label them either as "identical" (i.e., the change is an explication change) or as "conceptual change". In the latter case, the user can specify the conceptual relation between the two versions of the concept, for example, by stating the property "*Stamp_pads*" in version 8.4 is a subset of "*Ink_or_stamp_pads*" in version 8.0.

Another function is the possibility to analyze the effect of changes. Changes in ontologies do not only affect the data and applications that use them, but they can also have unintended, unexpected, and unforeseeable consequences in the ontology itself. The system provides some basic support for the analysis of these effects. First, on request it can highlight the places in the ontology where changed concepts or properties are used. For example, if a property "*hasChild*" is changed, it will highlight the definition of the class "*Mother*", which uses the property "*hasChild*". This function can also exploit the transitivity of properties to show the propagation of possible changes through the ontology. A foreseen second effect analysis feature is the connection to FaCT, which allows checking the formal consistency of the suggested conceptual relations between different versions of definitions.

When an ontology does not have persistent identifiers for concepts, there is another task involved in comparing the two versions - finding the mappings between concepts in the two versions. This task is closely related to the task of ontology alignment in general. PromptDiff [Noy & Musen, 2002] is a tool that integrates different heuristics for comparing ontology versions. PromptDiff uses heuristics similar to those that are used to provide suggestions for ontology merging in Prompt [Noy & Musen, 2000]. Figure 4 shows the differences that are detected between version 8.0 and 8.4 of the UNSPSC classification (ignoring the persistent EGCI code). The tool lists the concept names in the two versions, whether their name is changed (and the reason behind this conclusion), and whether the structure is changed.

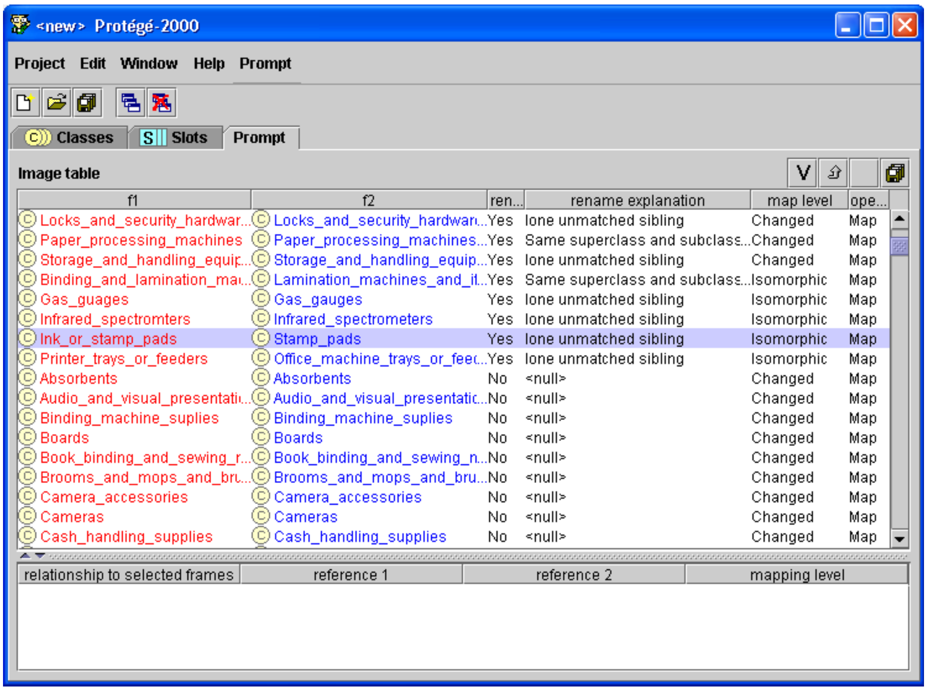
Grounding of Standards

eCommerce is about buying and selling actual products and services. These goods need to be classified and described in terms of standardized categorizations for reasons of reporting and searching. In this section, we portray the prototype of automatic classification of product description in the B2B marketplace (called GoldenBullet) to realize a semi-automatic way to populate ontologies in eCommerce.

GoldenBullet: Automatic classification of product description

Finding the right place for a product description in a standard classification system such as UNSPSC is not a trivial task. Each product must be mapped to corresponding product category in UNSPSC to create the product catalog. Product classification schemes contain huge number of categories with far from sufficient definitions (i.e. over 15,000 classes for UNSPSC), and millions of products must be classified according to them. This requires tremendous effort and the product classification stage takes altogether up to 25 percent of the time spent for content management [Fensel et al., 2002(a)].

GoldenBullet is a software environment targeted to support product classification according to certain content standards. It is currently designed to automatically classify the products based on their original descriptions and existenting classification standards (such as UNSPSC). It integrates different classification algorithms from the information retrieval and machine learning, and some natural



f1	f2	ren...	rename explanation	map level	ope...
Locks_and_security_hardwar...	Locks_and_security_hardwar...	Yes	lone unmatched sibling	Changed	Map
Paper_processing_machines	Paper_processing_machines...	Yes	Same superclass and subclass...	Changed	Map
Storage_and_handling_equip...	Storage_and_handling_equip...	Yes	lone unmatched sibling	Changed	Map
Binding_and_lamination_ma...	Lamination_machines_and_it...	Yes	Same superclass and subclass...	Isomorphic	Map
Gas_guages	Gas_gauges	Yes	lone unmatched sibling	Isomorphic	Map
Infrared_spectromters	Infrared_spectrometers	Yes	lone unmatched sibling	Isomorphic	Map
Ink_or_stamp_pads	Stamp_pads	Yes	lone unmatched sibling	Isomorphic	Map
Printer_trays_or_feeders	Office_machine_trays_or_fee...	Yes	lone unmatched sibling	Isomorphic	Map
Absorbents	Absorbents	No	<null>	Changed	Map
Audio_and_visual_presentati...	Audio_and_visual_presentatic...	No	<null>	Changed	Map
Binding_machine_suplies	Binding_machine_suplies	No	<null>	Changed	Map
Boards	Boards	No	<null>	Changed	Map
Book_binding_and_sewing_r...	Book_binding_and_sewing_n...	No	<null>	Changed	Map
Brooms_and_mops_and_bru...	Brooms_and_mops_and_bru...	No	<null>	Changed	Map
Camera_accessories	Camera_accessories	No	<null>	Changed	Map
Cameras	Cameras	No	<null>	Changed	Map
Cash_handling_supplies	Cash_handling_supplies	No	<null>	Changed	Map

relationship to selected frames	reference 1	reference 2	mapping level

Fig. 5 The result of a comparison of two version of the UNSPSC hierarchy in PromptDiff.

language processing techniques to pre-process data and index UNSPSC to improve classification accuracy. The system helps to mechanize an important and labor-intensive task of content management for B2B eCommerce.

We will first describe the main components. A *wrapper factory* gathers various wrappers to convert the raw data description from external formats (Database, Excel, XML-like, formatted plain text,...) into internal formats, and subsequently convert the final results to preferable output formats (Database, Excel, XML-like, plain text,...) or user-designed formats. No matter how the data are imported manually or automatically, before they are passed to be pre-processed, they are *validated* by the GoldenBullet data validator. Basic validation is checked. For instance, to see if a description is too long or too short, or the Product ID is missing or incorrect. The validated product data will be pre-processed before the automatic classification has been performed. Some of the *Natural Language Processing* algorithms have been implemented into GoldenBullet. The product data will be stemmed (grouping different words with the same stems) and tagged (extracting noun-phrases). Furthermore, UNSPSC is also being pre-processed (stemmed and tagged) to make sure that noisy words or information have been screened out. A stop word list has been generated, updated and extended during the whole process. The *learning algorithm* has been embedded in GoldenBullet; the classification rules and instances learned during the online or offline learning procedure are stored in the system to enrich UNSPSC and the classification rule base. Thus, the loop of the entire system has been formed and the system can be self-improved. The more data it processes, the more intelligence it gains. Currently, GoldenBullet can handle English and French product data.

The essence of GoldenBullet is its ability to automatically classify product descriptions. This requires two important properties: (1) Intelligence in classification: We implemented and evaluated various classification strategies; (2) Knowledge in the domain: We acquired and used ten thousands of manually classified product data to learn from it. To satisfy the above two requirements, the following algorithms have been implemented in GoldenBullet:

- The standard Vector space model (VSM, [Salton et al., 1975]) has been applied to represent document (in our case product description) and existing categories (in our case UNSPSC). The category (UNSPSC) can then be assigned to a document (product) when the cosine similarity between them exceeds a certain threshold.
- Another algorithm implemented here is based on the k-Nearest Neighbor method (KNN). The algorithm uses the set of pre-classified examples directly to classify an example, passes the whole set of training examples, searches for the most similar one, and then assigns the class to the new example that equals to the class of the most similar one.
- The Naïve-Bayes classifier (NB, [Mitchell, 1997]) was also employed to learn and train our pre-classified data and ten thousands of manually classified product data from the vendors

VSM was adopted to find the match between UNSPSC commodities and product descriptions. We implemented two strategies. Both treat an unclassified product description as a query; however, they differ in what they use as a document collection:

- The first takes each commodity as a document. The examples are used to enrich the commodity description. Essentially, we extract words from pre-classified product data and add them to the word list describing the commodity.
- The second takes each pre-classified product description as a document. We use VSM to retrieve the instance that best fits to a new product description and infer the UNSPSC code of the latter from the known UNSPSC code of the former.

Content management has to structure, classify, re-classify, and personalize large volumes of data to make product descriptions automatically accessible via B2B market places. GoldenBullet applies the information retrieval and machine learning metaphor to the problem of automatically classifying product descriptions according to the existent product classification standards. Furthermore, GoldenBullet will challenge other existing problems in the B2B marketplace, such as mapping and reclassifying product descriptions according to different product classification standards, personalizing the marketplace view to divergent customers, and offering flexible input and output services.

Conclusions

No technology can survive without convincing application areas. However, the reader should also be aware about the time span of innovation. For example, it took the Internet 30 years before it was hidden by its killer application, the World Wide Web. Lets hope we need less than a generation for the next killer. Ontology technology certainly has promising potential in areas such as knowledge management, Enterprise-Application Integration, and eCommerce.

eCommerce in business to business (B2B) is not a new phenomenon. However, Internet-based electronic commerce provides a much higher level of *openness, flexibility,* and *dynamics* that will help to optimize business relationships. This type of eCommerce technology may change the way business relationships are established and performed. In a nutshell, web-enabled eCommerce helps its users to contact a large number of potential clients without running into problems associated with implementing numerous communication channels. This enables virtual enterprises that are form in reaction to demands from the market and vica versa it enables to brake large enterprises up into smaller pieces that mediate their eWork relationship based on eCommerce relationships. In consequence, flexible and open eCommerce has to deal with serious problems (cf. [Fensel et al., 2002(a)]).

1) **Openness** of eCommerce cannot be achieved without standardization. Such a lesson can be learned from the success of the web; however, the requirements on standardization are much higher here. We also require standardization of the actual content exchanged, which goes far beyond the requirements of standardizing protocols and document layouts (i.e., we require ontologies).

2) **Flexibility** of eCommerce cannot be achieved without multi-standard approaches. It is unlikely that a standard will arise that covers all aspects of eCommerce that is acceptable for all vertical markets and cultural contexts. Nor would such a standard free us from the need to provide user-specific views on it and the content it represents.

3) **Dynamic** of eCommerce requires standards that act as living entities. Products, services, and trading modes are subject of high change rates. An electronic trading device must reflect the dynamic nature of the process it is supposed to support.

Given these requirements only ontology technology can promise to provide at least a partial solution.

References

[CWM, 2001]
CWM, “Common Warehouse Model Specification”, Object Management Group, 2001. http://www.omg.org/cwm/.

[Davis et al., 2002]
J. Davis, D. Fensel, and F. van Harmelen (eds.): *Towards the Semantic Web: Ontology-*

Driven Knowledge Management, Wiley, 2002.

[Ding et al., 2002]

Y. Ding, M. Korotkiy, B. Omelayenko, V. Kartseva, V. Zykov, M Klein, E. Schulten, and D. Fensel: GoldenBullet in a nutshell. In *Proceedings of the 15th International FLAIRS Conference,* AAAI Press, May 16-18, 2002 (in press).

[Fensel, 2001]

D. Fensel: *Ontologies: Silver Bullet for Knowledge Management and Electronic Commerce*, Springer-Verlag, Berlin, 2001.

[Fensel et al., 2001]

D. Fensel, Y. Ding, B. Omelayenko, E. Schulten, G. Botquin, M. Brown, and A. Flett: Product Data Integration for B2B E-Commerce, *IEEE Intelligent Systems*, 16(4), 2001.

[Fensel et al., 2002(a)]

D. Fensel, B. Omelayenko, Y. Ding, M. Klein, A. Flett, E. Schulten, G. Botquin, M. Brown, and G. Dabiri: *Intelligent Information Integration in B2B Electronic Commerce*, Kluwer Academics Publishers, Boston/Dordrecht/London, 2002.

[Fensel et al., 2002(b)]

D. Fensel, J. Hendler, H. Lieberman, and W. Wahlster (eds.): *Spinning the Semantic Web: Bringing the World Wide Web to its full Potential,* MIT Press, Boston, 2002.

[ISO, 2000]

Standard, I. S. O., "Integrated generic resource: Fundamentals of product description and support", International Standard ISO 10303-41, Second Edition, 2000.

[Klein et al., 2002]

M. Klein, A. Kiryakov, D. Ognyanov, and D. Fensel: Ontology versioning and change detection on the web. In *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW02)*, Siguenza, Spain, October 2002.

[Klein & Fensel, 2001]

M. Klein and D. Fensel: Ontology versioning on the Semantic Web. In *Proceedings of the First Semantic Web Working Symposium*, Stanford, July 2001.

[Maedche et al., 2002]

A. Maedche, B. Motik, N. Silva, R. and Volz: MAFRA - A MApping FRAmework for Distributed Ontologies. In A. Gomez-Perez and R. Benjamins (eds.), *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW-2002)*, Springer-Verlag, LNCS 2473, Siguenza, Spain, October 2002.

[Mello and Heuser, 2001]

R. Mello and C. Heuser: A Rule-Based Conversion of a DTD to a Conceptual Schema. In H. Kunii et al. (eds.), *Conceptual Modeling - ER'2001*, Springer, LNCS 2224, November 27-30, 2001, pp. 133-148.

[Mitchell, 1997]

T. Mitchell, *Machine Learning*, McGraw Hill, 1997.

[Noy & Klein, 2002]

N. F. Noy and M. Klein: Ontology Evolution: Not the Same as Schema Evolution. SMI technical report SMI-2002-0926, 2002.

[Noy & Musen, 2000]

N. F. Noy and M. Musen: PROMPT: Algorithm and tool for automated ontology merging and alignment. In Proceedings of the 17th Nat.Conf. on Artificial Intelligence (AAAI-2000), 2000.

[Noy & Musen, 2002]

N. F. Noy and M. Musen: PromptDiff: A Fixed-Point Algorithm for Comparing Ontology Versions. In *Proceedings of the Eighteenth National Conference Artificial Intelligence (AAAI-02)*, Edmonton, Alberta. AAAI Press. 2002.

[Omelayenko, 2002(a)]

B. Omelayenko: Integrating Vocabularies: Discovering and Representing Vocabulary Maps. In *Proceedings of the First International Semantic Web Conference (ISWC-2002)*, Springer, LNCS (in press), June 2002.

[Omelayenko, 2002(b)]

B. Omelayenko: RDFT: A Mapping Meta-Ontology for Business Integration. In *Proceedings of the Workshop on Knowledge Transformation for the Semantic for the Semantic Web* at the 15th European Conference on Artificial Intelligence (KTSW-2002), July 2002.

[Roddick, 1995]

J. F. Roddick: A survey of schema versioning issues for database systems, *Information and Software Technology*, 37(7):383–393, 1995.

[Salton et al., 1975]

G. Salton, A. Wong, and C. S. Yang: A vector space model for automatic indexing, *Communications of the ACM*, 18(7): 613-620, 1975.