8109103 Ohjelmistotuotannon teoria

Ontology in Information Systems

22.03.2004

Mika Viinikkala
mika.viinikkala@tut.fi

TABLE OF CONTENTS

1. INTRODUCTION

This document discusses about ontology in information systems discipline. Until recently, there have been only one meaning for the term ontology. Originally, ontology is an old branch of philosophy that, in brief, aims to establish the truth about reality by finding an answer to the question: what exists. Ontology, in its original meaning, aims at the objective description of reality and any domain of objects. Recently, the term has been increasingly used by the members of information systems discipline. However, the role and meaning that ontology seems to have in information systems discipline is wholly distinct from its role in philosophy.

This document discuses about the original ontology and then describes the process how ideas of ontology were adopted by members of information systems discipline in order to better understand the main subject of this document.

In information systems discipline, in use of the ‘term’ ontology, some kind of anarchy has existed until these days. The term has been used in various associations and various definitions have been declared for the term. However, this document tries to benefit from attempts to harmonize the meaning of the term ontology while describing what exactly is meant by ontology in context of information systems. Also, a distinction to philosophical ontology is made.

The document also describes for what purposes an ontology or ontologies can be used. In addition, some languages for creating ontologies as well as some attempts to create an ontology are also introduced. However, the subject is theoretical and it concerns many fields of sciences thus making the subject not easy. This document stays at rather abstract level. In order to have more understanding about ontology in information systems, it would be sufficient to examine one of the existing ontology languages in detailed level and to apply the ideas of ontology to various real world problems.

In the end, it is not a surprise why ontology came into the field of information systems. Representation is the essence of the things that are called information systems. As Weber argues, information systems represent the histories of things in the real world in terms of the ways we have chosen to conceive these things. [Weber, 1997] We need things that represent other things because our use of information systems is id often the most cost effective way to know about the histories of things in the real world. It is more efficient to observe representations of things that the things directly. According Weber, actually, the term ‘information system’ that is used for these things that process representations is a misnomer. They are not systems of information but systems of representations. [Weber, 1997] And what comes about representations, ontology has certainly something to do with it.

2. PHILOSOPHICAL ONTOLOGY

In this chapter, philosophical ontology is discussed. Its historical backgrounds and fundamental ideas are introduced. The original meaning of ontology is clarified in order to create better understanding of ontology in information systems.

2.1 History of ontology

Ontology as a branch of philosophy *is the science of what is, of the kinds and structures of objects, properties, events, processes and relations in every area of reality*. Ontology is often used by philosophers as a synonym for *metaphysics* – a term which was used by early students of Aristotle to refer to what Aristotle himself called *first philosophy*. Sometimes ontology is used in a broader sense, to refer to the study of what *might* exist, where metaphysics is used for the study of which of the various alternative possibilities is true of reality. [Smith]

The word ontology comes from the Greek *ontos* for being and *logos* for word. [Sowa, 2001] The term ontology was itself coined in 1613, independently, by two philosophers, Rudolf Göckel, in his *Lexicon philosophicum* and Jacob Lorhard, in his *Theatrum philosophicum*. The first occurrence in English appears in Bailey's dictionary of 1721, which defines ontology as '*an Account of being in the Abstract*'. [Smith]

The more traditional term is Aristotle's word *category*, which he used for classifying anything that can be said about anything. He presented ten basic categories in the first treatise in his collected works. [Sowa, 2001] The oldest known categorization was drawn in the 3rd century AD by the Greek philosopher Porphyry in his commentary on Aristotle's categories. Figure 1 shows a version of the *Tree of Porphyry*, as it was drawn by the 13th century logician Peter of Spain. It illustrates the subcategories under Substance, which is called the *supreme genus* or the most general supertype. Despite its age, the Tree of Porphyry has many features that are considered quite modern. Following is Porphyry's description: *Substance is the single highest genus of substances, for no other genus can be found that is prior to substance. Human is a mere species, for after it come the individuals, the particular humans. The genera that come after substance, but before the mere species human, those that are found between substance and human, are species of the genera prior to them, but are genera of what comes after them.*

Aristotle introduced the term *differentia* for the properties that distinguish different species of the same genus. Substance with the differentia material is Body and with the differentia immaterial is Spirit. The technique of *inheritance* is the process of merging all the differentiae along the path above any category: LivingThing is defined as animate material Substance, and Human is rational sensitive animate material Substance. Aristotle's method of defining new categories by *genus* and *differentiae* is fundamental to artificial intelligence, object-oriented systems, the semantic web, and every dictionary from the earliest days to the present. [Sowa, 2001]

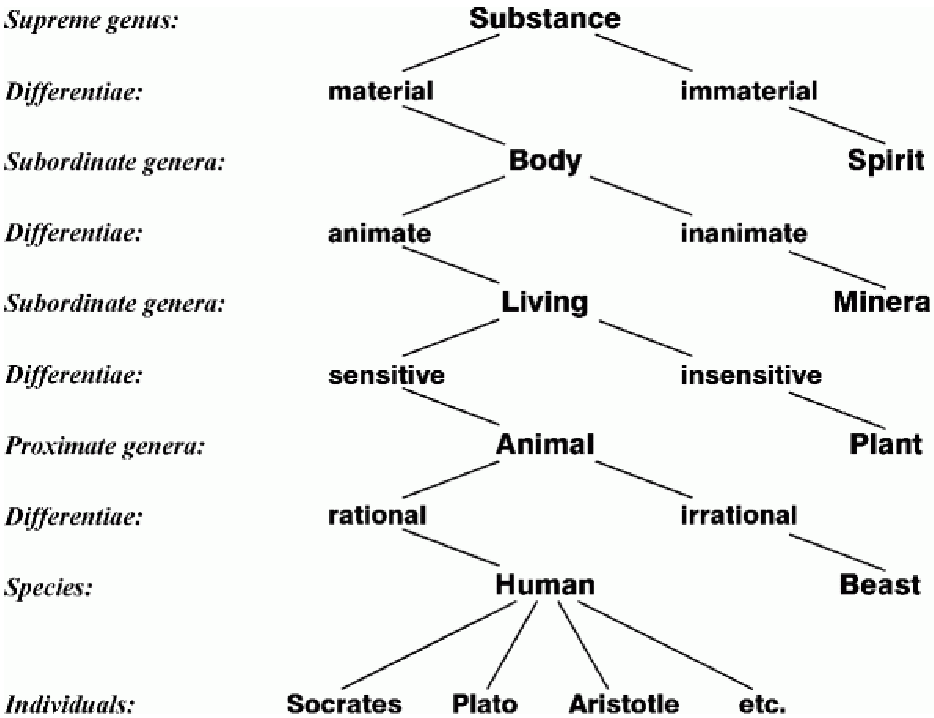

Figure 1: The tree of Porphyry [Sowa, 2001]

2.2 Goals of philosophical ontology

The goal of ontology is to provide a definitive and exhaustive classification of entities in all spheres of being. [Smith] The classification should be definitive in the sense that it can serve as an answer to such questions as: *What classes of entities are needed for a complete description and explanation of all the goings-on in the universe? What classes of entities are needed to give an account of what makes true all truths?* The classification should be exhaustive in the sense that all types of entities should be included in the classification, including also the types of *relations* by which entities are tied together to form larger wholes. That is, philosophical ontology aims for classification and description and the targets for description are things in any domain and the relations existing among them.

It is important to notice that the goal of philosophical ontology is truth. For example, collecting the knowledge of the denizens of any sphere would not be sufficient to produce an ontology of that sphere. Philosophical ontology accounts for the knowledge and the conceptual framework of the sphere. However, part of the task involves distilling true from false beliefs in the analysis of the nature of the sphere. [Zúñiga, 2001]

2.3 Approaches to ontology

Different schools of philosophy offer different approaches to ontology. One large division is made between so called *substantialists* and *fluxists*, that is, those who conceive ontology as a substance- or thing- (or continuant-) based discipline and those who favor an ontology centered on events or processes (or occurrents), respectively.

Another large division is between what might be called *adequatists* and *reductionists*. Adequatists seek a taxonomy of the entities in reality at all levels of aggregation, from the microphysical to the cosmological, and including also the middle world (the *mesocosmos*) of human-scale entities in between. Reductionists see reality in terms of some one privileged level of existents. That is, they seek to establish the ‘ultimate furniture of the universe’ by decomposing reality into its simplest constituents, or they seek to ‘reduce’ in some other way the apparent variety of types of entities existing in reality.

Sometimes, a division between formal and material (or regional) ontology is made. In philosophy, formal ontology is domain-neutral; it deals with those aspects of reality, such as parthood and identity, which are shared in common by all material regions. Material ontology deals with features, such as mind or causality, which are specific to given domains. [Smith]

2.4 Ontology and science

Ontology is a descriptive enterprise. That is, it distinguishes from the special sciences not only in its radical generality but also in its goal or focus: *it seeks not explanation but rather taxonomy and description*. Ontology is mostly qualitative. In contrast, science is very largely quantitative. [Smith]

To describe something and create effective representations it is an advantage to know something about the things and processes that are tried to represent. Smith calls this *the Ontologist’s Credo.* The attempt to satisfy this credo has led philosophers to be maximally opportunistic in the sources they have drawn upon in their ontological explorations of reality. These have ranged all the way from the preparation of commentaries on ancient texts to reflection on our linguistic usages when talking about entities in domains of different types. Increasingly, however, philosophers have turned to science. Ontologists assume that one generally reliable way to find out something about the things and processes within a given domain is to see what scientists say. [Smith]

3. ONTOLOGY IN INFORMATION SYSTEMS

In recent years, the term ontology has gained currency in the field of computer and information science. The amount of publications and conferences on the topic of ontology has increased enormously. The term has become popular especially in domains such as knowledge engineering, natural language processing, cooperative information systems, intelligent information integration, and knowledge management.

In philosophical ontology, there is, in principle at least, only a one goal: to establish the truth about reality by finding an answer to the question: what exists? In the information systems discipline, in contrast, an ontology is a software artifact or formal language designed with a specific set of uses and computational environments in mind. In other words, an ontology is often something that is ordered by a specific client in a specific context and in relation to specific practical needs and resources.

In this chapter, links between philosophical ontology and information systems ontology are described by clarifying how ontology and its ideas were noticed by information system scientists. An effort is made to describe definitions of ontology in information systems. Uses of ontologies in information system discipline are also discussed. Finally, some examples of ontologies and ontology languages that are related to information systems are briefly introduced.

3.1 How ideas of philosophical ontology became in information systems discipline

At the early stages of information systems discipline, *knowledge-bases* were common. They were often crafted in ways that both reflected common sense human knowledge in a declarative way and took advantage of the powers of the particular automated reasoning system used. [Smith and Welty, 2001] In those days, the field of artificial intelligence was marked by debates between the so-called *proceduralists* and *declarativists*. These groups had different answers in the following question: What is the relative significance of *process* and *content* (or of *procedures* and *data*) in the project of modelling intelligent reasoning and constructing intelligent machines? [Smith]

Proceduralists believed that the way to create intelligent machines was by instilling into a system as much *knowledge how* as possible, via ever more sophisticated programs. [Smith] That is, they considered the reasoning mechanisms themselves to be the important scientific challenge, and the knowledge-bases to be nothing more than examples having no intrinsic significance.

Declarativists, on the other side, believed that intelligent machines would best be arrived at by instilling into a system a maximum amount of content, of *knowledge that* – knowledge in the form of representations. [Smith] That is, they argued that the knowledge bases themselves ought to be subject to scientific inquiry. They pointed out the high degree of arbitrariness characterizing existing knowledge-bases, and to the lack of rigor applied in their development.

In the database community, it was also discovered that after database technology begun to stabilize, the far more important problem *conceptual modeling* still remained. The declarative and the procedural elements of computer system were seen as representations by some members of the database community. Programs are representations of processes and data structures are representations of objects or things. The early years of database conceptual modeling were marked for the most part by ad hoc and inconsistent modeling, leading to the many practical problems of database integration we face today. [Smith and Welty, 2001]

Independently of these developments, yet another subfield of computer science, namely software engineering, encouraged in particular by advances in object-oriented languages, began to recognize the importance of what came to be known as *domain modeling*. The increasing size and complexity of programs meant increasing difficulties in maintaining such programs and putting them to new uses.

The idea was to build declarative representations of the procedures in a way which would allow application systems to re-use program elements. This field, too, was severely debilitated by a lack of concrete and consistent formal bases for making modeling decisions. [Smith and Welty, 2001]

The step from here to ontology in something like the traditional philosophical sense of this term is then relatively easy. The data analyst realizes the need for declarative representations which would have as much generality as possible in order to maximize the possibility of reusability. At the same time, these representations must correspond as closely as possible to the things and processes they are supposed to represent. Thus he starts asking questions like: *What is an object/process/attribute/relation? How do they depend on each other? How are they related?* [Smith and Welty, 2001]

Most members of the information systems discipline chose not to consider the work of the much older overlapping field of philosophical ontology, preferring instead to use the term ‘ontology’ as an exotic name for what they had been doing all along – knowledge engineering. This resulted in an unfortunate skewing of the meaning of the term as used in the AI and information systems fields, as work under the heading ontology was brought closer to logical theory, and especially to logical semantics, and it became correspondingly more remote from anything which might stand in a direct relation to existence or *reality*. Some may argue that this meaning is appropriate for a computer system – that it is enough to define the kinds of structures of objects, properties, events, processes and relations *that exist in the system*. On the other hand, many are now arguing that the very lack of grounding in external reality is precisely what created the problems, so pressing for the industry today, of *legacy systems integration.* How can we make older systems with different conceptual models but overlapping semantics work together, if not by referring to the common world to which they all relate? [Smith and Welty, 2001]

The significance of the term grew, and as the disparate fields of knowledge engineering, conceptual modeling, and domain modeling began to converge and discover each other, so, too, did the range of variations in its meaning.

3.2 Various interpretations of the information systems ontology

The first credible attempt to define the term ontology was Tom Gruber’s contribution in 1993. [Gruber, 1993] However, his definition, *“an ontology in a specification of a conceptualization”*, left room for too many possible interpretations, and despite the attempts to clarify and formalize the definition further, new meanings of the term ontology continued to proliferate.

Welty, Lethmann, Gruniger, and Uschold reported in 1999 on a wide spectrum of information artifacts that had been at some time classified as ontologies. [Welty et al., 1999] The results of their work are shown in figure 2.

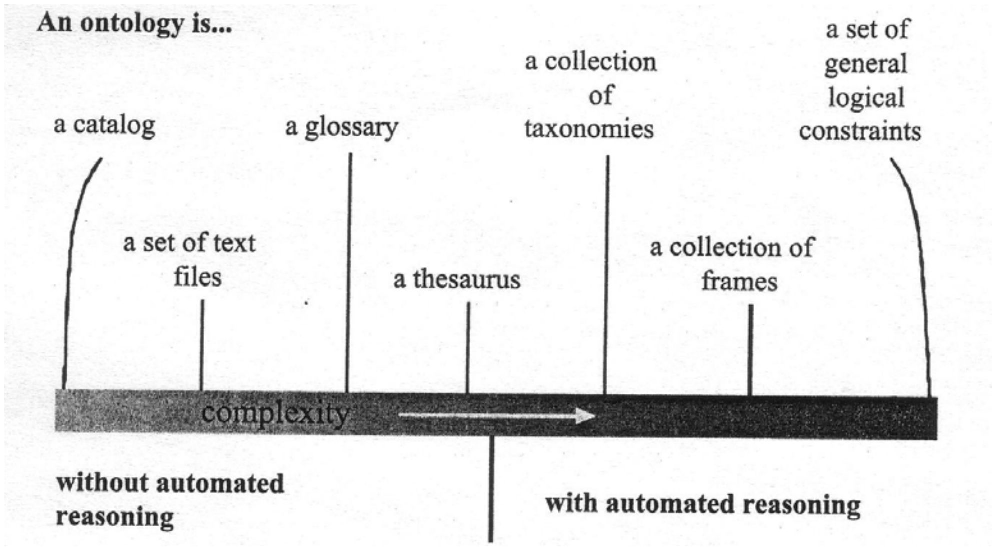

Figure 2: Various interpretations of information systems ontology [Smith and Welty, 2001]

Information systems as simple as catalogs, in which each product type has a unique code, have been dubbed as ‘ontologies’. A catalog, in a sense, the ontology of the tings a company sells. A slightly more complex information system may provide simple natural language descriptions of terms, thus imposing some structure on the text. Thesauri are standardized information systems that provide, in addition to description of terms, also relations to other more general or more specific terms within a common hierarchy. The fields of knowledge representation, database development, and object oriented software engineering all employ ontologies conceived as taxonomies in which properties of more general classes are inherited by the more specific ones. Frame-based systems provide, in addition to taxonomic structure, relations between objects and restrictions on what and how classes of objects can be related to each other. Finally, the most expressive and complex information system ontologies use the axioms of full first order, higher order, or modal logic. All these types of information systems satisfy Grubers’s definition, and all are now common bedfellows under the rubric of ‘ontology’. [Smith and Welty, 2001]

3.3 Searching a common reference ontology

In the AI community, ontology building is seen as a process of extending the frontiers of what can be represented in a systematic fashion in a computer. Boundaries of automation are tried to be radically extended by means of ‘artificial intelligence’.

In information systems discipline, in particular in the data modeling and knowledge representation communities, however, the goal is to integrate the automated systems we already have. Here the problems faced by ontologists are presented by the heterogeneous systems used, for example, in the different parts of a large enterprise.

In this sense, the most important task for the new information systems ontology pertains to what might be called the Database Tower of Babel problem. Different groups of data- and knowledge-base system designers have for historical and cultural and linguistic reasons their own idiosyncratic terms and concepts by means of which they build frameworks for information representation. [Smith] Different databases may use identical labels but with different meanings. Alternatively the same meaning may be expressed via different names. As ever more diverse groups are involved in sharing and translating ever more diverse varieties of information, the problems standing in the way of putting such information together within a larger system increase geometrically.

It was therefore recognized that systematic methods must be found to resolve the terminological and conceptual incompatibilities which then inevitably arise. Initially, such incompatibilities were resolved on a case-by-case basis. Gradually, however, it was recognized that a common reference ontology – a shared taxonomy of entities – might provide significant advantages over such case-by-case resolution. [Smith]

The potential advantages of such ontology for the purposes of information management are obvious. Each group of data analyst having a database would need to perform the task of making its terms and concepts compatible with those of other such groups having other databases only once. This would be established by making its terms and concept compatible with a single shared canonical backbone language, a sort of ontological Esperanto.

Thus, great advantages would be achieved if all databases were made compatible with just one common ontology built around a consistent, stable and highly expressive set of category labels. The prospect would arise of leveraging the thousands of person-years of effort that have been invested in creating separate database resources in such a way as to create, in more or less automatic fashion, a single integrated knowledge, that would fulfil an ancient philosophical dream of an encyclopedia comprehending all knowledge within a single system.

However, the obstacles standing in the way of the extension of such an ontology to the level of categorical details which would be required to solve the real-world problems of database integration are unfortunately prodigious. They are analogous to the task of establishing a common ontology of world history. This would require a neutral and common framework for all descriptions of historical facts, which would require in turn that all events, legal and political systems, rights, beliefs, powers, and so forth, be comprehended within a single, perspicuous list of categories. [Smith] However, the idea about common ontology has been proved useful when applied in a smaller scale, as will be described later.

3.4 What is an information systems ontology

There is no common agreement about what ontology is in the information systems literature. However, it would be safe to say that information systems ontology is a formal language designed to represent a particulate domain of knowledge. The purpose for information systems ontologies is functional. That is, an information systems ontology is designed for one or more purposes that arise in the effort to computerize as much information as possible, an effort that characterizes today's business and educational environments. [Zúñiga, 2001]

Here, a distinction related to the term 'ontology' must be emphasized. That is, the distinction between ontology as a sphere of investigation, and a particular ontology that results from an ontological investigation. The former in a branch of philosophy occupied with the description of existence and reality. The latter is either someone's ontology, or an ontology of this or that domain of objects – for example an ontology of works of art. It can be also said, that in information systems discipline, the term 'ontology' commonly means a particular ontology while, in contrast, in philosophy the term 'ontology' usually means a sphere of investigation.

Moreover, as already mentioned earlier, formal ontology consists in the general investigation and description of the properties of objects in the world and the relations existing between different sorts of objects. Material ontology, by contrast, is occupied with only a particular domain of objects, and it consists in the description of the structure of the domain and the relations of the objects therein. There is another distinction that must be considered: that between the word 'formal' meaning general, and the word 'formal' in the sense of a deductive system and its symbolism. The latter meaning is the one most often used in the information community. In philosophy, the relevant sense of 'formal' for ontology addresses the fact that the description resulting from the investigation is general and thus applicable to all material spheres or domains of reality. [Zúñiga, 2001]

As mentioned earlier, the first credible attempt to define the term ontology was Tom Gruber's contribution in 1993. [Gruber, 1993] In context of information systems discipline, he defines the term ontology as an *explicit specification of a conceptualization.* That is, an ontology is a description (like a formal specification of a program) of the concepts and relationships that can exist for an agent or a community of agents. As he says, this definition is consistent with the usage of ontology as set-of-concept-definitions, but more general. And it is certainly a different sense of the word than its use in philosophy. [Gruber]

However, he argues further that what is important is what an ontology is *for*. Ontologies are designed for purpose of enabling knowledge sharing and reuse. In this context, he argues, an ontology is a specification used for making ontological commitments. For pragmatic reasons, an ontology is written as a set of definitions of formal vocabulary. This is not the only way to specify a conceptualization, but it has some desired properties for knowledge sharing e.g. semantics independent of reader and context. Practically, an ontological commitment is an agreement to use a vocabulary i.e. ask queries and make assertions in a way that is consistent with respect to the theory specified by an ontology. Agents that commit to an ontology are build. Ontologies are designed in order to enable knowledge sharing with and among agents. [Gruber]

A number of other definitions have been published after the information systems discipline and philosophy meet each other by means of ontology. Zúñiga captures the relevant differences between philosophical ontology and ontology in information systems aiming for convergence of existing definitions. According to Zuniga, the objective of philosophical ontology is description, and its commitment is to truth. Philosophical ontology is an investigation into the nature and existence of thing in the world, and this investigation must not be entirely dependent on our knowledge of things in the world. Philosophical ontology aims at the objective description of any domain of objects. In contrast, an information systems ontology in an axiomatic theory made explicit by means of a specific formal language. The IS ontology is designed for at least one specific or practical application. Consequently, it depicts the structure of a specific domain of objects, and *it accounts for the* intended meaning of a formal vocabulary or protocols that are employed by the agents of the domain under investigation. [Zúñiga, 2001]

3.5 Uses of ontology in information science

The project of building one single ontology, even one single top-level ontology, which would be at the same time non-trivial and also readily adopted by a broad population of different information systems communities, has largely been abandoned. The reasons for this can be summarized as follows. The task of ontology-building proved much more difficult than had initially been anticipated. Difficulties including in it can be said to be at leas in part identical to those with which philosophical ontologists have been grappling for some 2000 years. Moreover, the information systems world itself is very often subject to the short time horizons of the commercial environment. This means that the requirements placed on information systems themselves change at a rapid rate, so that theoretically grounded work on ontologies conceived as modules for translating between information systems has been unable to keep pace. [Smith]

However, work in ontology in the information systems world continues to flourish. The principal reason for this lies in the fact that its focus on classification and on constraints on allowable taxonomies and definitions has proved useful in ways not foreseen by its initial progenitors [Guarino and Welty, 2000]. Automation requires a higher degree of accuracy in the description of its procedures, and ontology is a mechanism for helping to achieve this. The attempt to develop terminological standards, which means the provision of explicit specifications of the meanings of terms, loses nothing of its urgency in application domains such as medicine or air traffic control, even when the original goal of a common ontology embracing all such domains has been set to one side. [Smith]

Ontology also goes by other names, so that the building of ontologies has much in common with work on what are still called *'conceptual schemes'* in database design, or on *'models of application domains'* in software engineering, or on *'class models'* in object-oriented software design. The designers of large databases are increasingly using ontological methods as part of their effort to impose constraints on data in such a way that bodies of data derived from different sources will be rendered mutually compatible from the start. Ontological methods are used also in the formalization of standards at the level of metadata. There the goal is to provide in systematic fashion information about the data with which one deals, for example as concerns its quality, origin, nature and mode of access. [Smith]

Ontological methods may have implications also for the writing of software. If you have gone to the trouble of constructing an ontology for purposes of integrating existing information systems, this ontology can itself be used as a basis for writing software that can replace those old systems, with anticipated gains in efficiency.

Ontological methods have been applied also to the problems of extracting information for example from large libraries of medical or scientific literature, or to the problems of navigation on the Internet. The semantic web aims to use ontology as a tool for answering the issue of huge diversity of sources from which Internet content is derived. Here, even a small increase of ontological rigor may provide significant benefits to both producers and consumers of on-line information. [Smith]

Ontological methods have been applied also in the domain of natural language translation, for example as aids to parsing and disambiguation. Nirenburg and Raskin have developed a methodology for what they call 'ontological semantics', which seeks to use ontological methods as the basis for a solution to the problem of automated natural language processing. Here ontology is conceived as a 'constructed world model' and it would provide the framework for unifying the needed knowledge modules within a comprehensive system. Thus they use ontology as the central resource for extracting and representing meaning of natural language texts, reasoning about knowledge derived from texts as well as generating natural language texts based on representations of their meaning. [Nirenburg and Raskin, 2001]

Efforts continue to be made to use ontology to support business enterprises [Uschold *et al.,* 1998. Obrst *et al.,* 2001]. A large international banking corporation with subsidiaries in different countries throughout the world probably seeks to integrate the information systems within its separate parts in order to make them interoperable. Here again a common ontology is needed in order to provide a shared framework of communication, and even here, within the relatively restricted environment of a single enterprise, the provision of such a common ontology may be no easy task, in virtue of the fact that objects in the realms of finance, credit, securities, collateral and so on are structured and partitioned in different ways in different cultures. [Smith]

One intensely pursued goal in the information systems ontology world is that of establishing methods for automatically generating ontologies. Hakimpour and Geppert, for example, have introduced an approach that is based on ontologies and that answers the problem of resolving semantic heterogeneity in schema integration. [Hakimpour and Geppert, 2001] The approach is applied to integrate schemas from different communities, where each such community is using its own vocabulary, which is dubbed an 'ontology'. The approach is based on merging these ontologies based on similarity relations among concepts of different ontologies. They present formal definitions of similarity relations that are based on intensional definitions. Similarity relations among concepts of ontologies are detected in order to generate the merged ontology semi-automatically. The merged ontology is then used to derive an integrated schema. The resulting schema can be used as the global schema in a federated database system. This process is illustrated in figure 3.

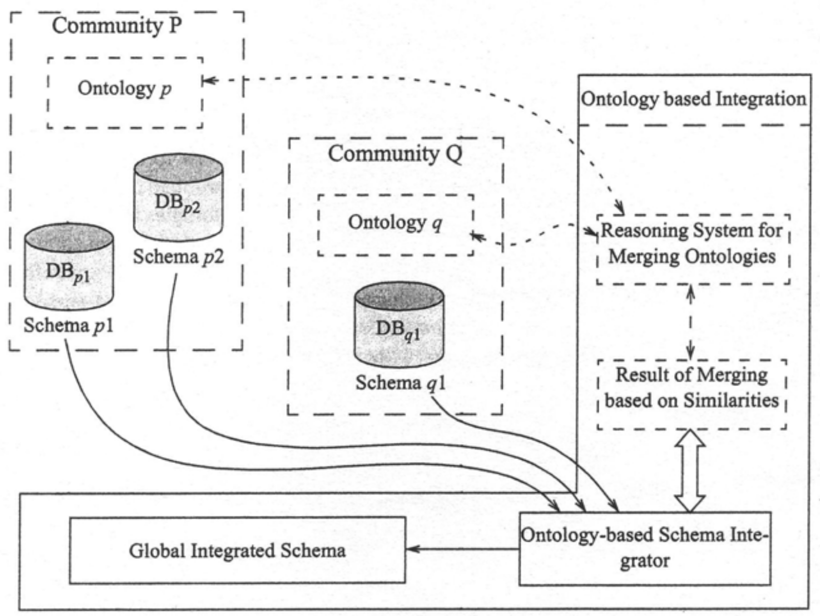

Figure 3: Global schema generation based on a common ontology produced by integration of ontologies [Hakimpour and Geppert, 2001]

3.6 Examples of information system ontologies

In this subchapter, examples of ontologies and ontology languages are introduced briefly.

3.6.1 KIF

Knowledge Interchange Format (KIF) language is an important milestone in the development of ontology as a solution to the problems of knowledge sharing. Originally, the language was not conceived for ontological purpose. KIF is a variant of the language of the first-order predicate calculus, motivated by the goal of developing an expressive, flexible, computer- and human-readable medium for exchanging knowledge bases. The existence of such a language means that each system, provided its syntax is translatable into that of KIF, can internally handle data in its own ways and communicate with its human users in yet other ways, but with the guarantee that the results of the system's operations will be automatically compatible with those of other systems likewise structured in such a way as to be compatible with KIF. More information about KIF can be found from [KIF].

3.6.2 Ontolingua

On the basis of KIF, Tom Gruber and his associates at the Stanford Research Institute developed a more serviceable language for ontology representation known as Ontolingua [Gruber 1992, 1995], designed to serve as a lingua franca for those involved in building ontologies. Ontolingua is built up on the basis of KIF 3.0, but it has a very distinctive purpose. Where KIF is conceived as an interface between knowledge representation systems, Ontolingua is intended as an interface between ontologies (analogous, again, to Esperanto). It provides an environment and a set of software tools designed to enable heterogeneous ontologies to be brought together on a common platform via translation into a single language. [Smith] More information about Ontolingua can be found from [Ontolingua].

3.6.3 Cyc

One of the most influential information systems ontology projects is that of Cyc [Lenat and Guha, 1990], which grew out of an effort initiated by Doug Lenat, one of the pioneers of AI research, to formalize common-sense knowledge in the form of a massive database of axioms covering all things, from governments to mothers. Cyc started as a research project in the early 80's. In 1995 Lenat created a company, Cycorp, charged with the task of developing further the technology and its applications. Cyc is intended to be able to serve as an encyclopedic repository of all human knowledge. ('Cyc' comes from en-cyc-lopedia.) As such it purports to provide a medium for the representation of facts and the inscription of rules about all existing and imaginable things. More information about this subject can be found from [Cyc].

3.6.4 OWL

Web Ontology Language (OWL) is part of the growing stack of W3C Recommendations related to the Semantic Web. [McGuinness and van Harmelen, 2004] The Semantic Web is a vision for the future of the Web in which information is given explicit meaning, making it easier for machines to automatically process and integrate information available on the Web.

The Semantic Web will build on XML's ability to define customized tagging schemes and RDF's flexible approach to representing data. The first level above RDF required for the Semantic Web is an ontology language what can formally describe the meaning of terminology used in Web documents. If machines are expected to perform useful reasoning tasks on these documents, the language must go beyond the basic semantics of RDF Schema.

XML provides a surface syntax for structured documents, but imposes no semantic constraints on the meaning of these documents. XML Schema is a language for restricting the structure of XML documents and also extends XML with datatypes. However, it can not be considered as an ontology language.

RDF is a datamodel for objects (resources) and relations between them. It provides a simple semantics for this datamodel, and these datamodels can be represented in an XML syntax. RDF Schema is a vocabulary for describing properties and classes of RDF resources, with a semantics for generalization-hierarchies of such properties and classes. It is recognized as an ontology language, but it is still too restricted.

OWL has more facilities for expressing semantics than XML, RDF, and RDF-S. It can be used to explicitly represent the meaning of terms in vocabularies and the relationships between those terms. Thus, it is intended to be used for creating ontologies that are representations of terms and their interrelationships. Also, OWL was designed to satisfy the requirements that are not satisfied by various efforts that preceded it. That is, OWL is a revision of the DAML-OIL web ontology language, which was an effort to merge DAML that was developed by group of largely US researchers and OIL that was developed by group of European researchers.

OWL provides three expressive sublanguages designed for use by specific communities of users: OWL Lite supports creating classification hierarchies and enables simple constraints. For example, while it supports cardinality constraints, it only permits cardinality of values 0 or 1. Moreover, it has a lower formal complexity than its more expressive relatives. OWL DL is named due its correspondence with description logics, a field of research that has studied the logics that form the formal foundation of OWL. OWL DL provides maximum expressiveness while retaining computational completeness and dedicability. That is, all conclusions are guaranteed to be computable and all computations will finish in finite time. OWL DL includes all OWL language constructs, but they can be used only under certain restrictions. OWL full is union of OWL syntax and RDF. OWL Full is meant for users who want maximum expressiveness and the syntactic freedom of RDF with no computational guarantees.

Furthermore, OWL Full can be viewed as an extension of RDF, while OWL Lite and OWL DL can be viewed as extensions of a restricted view of RDF. [McGuinness and van Harmelen, 2004]

4. SUMMARY

Ontology as a branch of philosophy *is the science of what is, of the kinds and structures of objects, properties, events, processes and relations in every area of reality.* The aim of philosophical ontology is to provide a classification of entities in all spheres of being. All types of entities should be included in the classification, including also the types of *relations.*

In philosophical ontology, there is, in principle at least, only a one goal: to establish the truth about reality by finding an answer to the question: what exists. In the information systems discipline, in contrast, an ontology is a software artifact or formal language designed with a specific set of uses and computational environments in mind. In other words, an ontology is often something that is ordered by a specific client in a specific context and in relation to specific practical needs and resources.

The objective of philosophical ontology is description, and its commitment is to truth. Philosophical ontology is an investigation into the nature and existence of thing in the world, and this investigation must not be entirely dependent on our knowledge of things in the world. Philosophical ontology aims at the objective description and the targets for description are things in any domain and the relations existing among them. In contrast, an information system ontology is an axiomatic theory made explicit by means of a specific formal language. The information system ontology is designed for at least one specific or practical application. Consequently, it depicts the structure of a specific domain of objects, and *it accounts for the* intended meaning of a formal vocabulary that is employed by the agents of the domain under investigation. In other words, an information systems ontology is a description of the concepts and relationships that can exist for an agent or a community of agents.

Different groups of data- and knowledge-base system designers have for historical and cultural and linguistic reasons their own idiosyncratic terms and concepts by means of which they build frameworks for information representation. Different databases may use identical labels but with different meanings. Alternatively the same meaning may be expressed via different names. There have been attempts to build one single ontology or even one single top-level ontology for purposes of information systems discipline. However, the task of building such ontology has been proved much difficult than had initially anticipated. Instead, in particular application domains, such as medicine or traffic control, ontologies have been successfully applied. Automation requires a higher degree of accuracy in the description of its procedures, and ontology is a mechanism for helping to achieve this.

Building of ontologies has much in common with work on what are still called *'conceptual schemes'* in database design, or on *'models of application domains'* in software engineering, or on *'class models'* in object-oriented software design. Ontological methods are increasingly used by large database designers to impose constrains on data in such way that bodies of data derived from different sources will be rendered mutually compatible from the start.

Ontological methods have been applied also to the problems of extracting information for example from large libraries of medical or scientific literature, or to the problems of navigation on the Internet. The semantic web aims to use ontology as a tool for answering the issue of huge diversity of sources from which Internet content is derived. Here, even a small increase of ontological rigor may provide significant benefits to both producers and consumers of on-line information and services.

Ontological methods have been applied also in the domain of natural language translation, for example as aids to parsing and disambiguation. Further, a business enterprise, such as a large banking corporation, aim to create a common ontology that is needed in order to provide a shared framework of communication within the enterprise.

REFERENCES

[Cyc] http://www.cyc.com (7.3.2004)

[Gruber, 1992] Gruber, T. R.: *Ontolingua: A Mechanism to Support Portable Ontologies*, Stanford Knowledge Systems Laboratory, 1992.

[Gruber, 1993] Gruber T. R.: *A Translation Approach to Portable Ontology Specifications*, Knowledge Acquisition, 5, 1993, pp. 199–220.

[Gruber, 1995] Gruber T. R.: *Toward Principles for the Design of Ontologies Used for Knowledge Sharing*, International Journal of Human and Computer Studies, 43(5/6), 1995, pp. 907–928.

[Gruber] Gruber T. R.: “What is an Ontology?”, http://www-ksl.stanford.edu/kst/what-is-an-ontology.html (7.3.2004)

[Guarino and Welty, 2000] Guarino N. and Welty C.: *A Formal Ontology of Properties*, Knowledge Engineering and Knowledge Management: Methods, Models and Tools. 12th International Conference (EKAW 2000), Berlin/New York, Springer, 2000, pp. 97–112.

[Hakimpur and Geppert, 2001] Hakimpour F., Geppert A.: *Resolving Semantic Heterogenity in Schema Integration: an Ontology Based Approach*, Formal Ontology in Information Systems, Proceedings of the Second International Conference (FOIS '01), October 17-19, Ogunquit, Maine, New York, ACM Press, 2001, pp. 297-308.

[KIF] http://www-ksl.stanford.edu/knowledge-sharing/kif/

[Lenat and Guha, 1990] Lenat D. B. and Guha R. V.: *Building Large Knowledge-Based Systems*: Representation and Inference in the Cyc Project, Reading, MA: Addison-Wesley, 1990.

[McGuinness and van Harmelen, 2004] McGuinness D. L., van Harmelen F.: OWL Web Ontology Language Overview, W3C Recommendation, http://www.w3.org/TR/2004/REC-owl-features-20040210/, February 10, 2004.

[Nirenburg and Raskin, 2001] Nirenburg S., Raskin V.: *Ontological semantics, formal ontology, and ambiguity*, Formal Ontology in Information Systems. Proceedings of the Second International Conference (FOIS '01), October 17-19, Ogunquit, Maine, New York, ACM Press, 2001, pp. 151-161.

[Obrst *et al.*, 2001] Obrst L., Wray R. E., and Liu H.: *Ontological engineering for B2B E-commerce*, Formal Ontology in Information Systems. Proceedings of the Second International Conference (FOIS '01), October 17-19, Ogunquit, Maine, New York, ACM Press, 2001, pp. 117-126.

[Ontolingua] http://www.ksl.stanford.edu/software/ontolingua/ (7.3.2004)

[Smith] Smith B.: http://ontology.buffalo.edu/smith/articles/ontologies.htm (7.3.2004)

[Smith and Welty, 2001] Smith B., Welty C.: *FOIS Introduction*, *Ontology: Towards a New Synthesis*, Formal Ontology in Information Systems. Proceedings of the Second International Conference (FOIS '01), October 17-19, Ogunquit, Maine, New York, ACM Press, 2001.

[Sowa, 2001] Sowa J. F.: Building, sharing, and merging ontologies. http://www.jfsowa.com/ontology/ontoshar.htm, 2001. (7.3.2004)

[Uschold *et al.* 1998] Uschold M., King M., Moralee S., and Zorgios Y.: *The Enterprise Ontology*, The Knowledge Engineering Review, 13, 1998.

[Weber, 1997] Weber R.: *Ontological Foundations of Information Systems*, Coopers & Lybrand, Australia, 1997, 212 p.

[Welty *et al*., 1999] Welty, C. Lehmann F., Gruniger G., and Uschold M.: *Ontology: Expert Systems All Over Again?*, Invited panel at AAAI-99: The National Conference on Artifical Intelligence, Austin, Texas, 1999.

[Zúñiga, 2001] Zúñiga G. L.: *Ontology: Its Transformation From Philosophy to Information Systems*, Formal Ontology in Information Systems, Proceedings of the Second International Conference (FOIS '01), October 17-19, Ogunquit, Maine, New York, ACM Press, 2001, pp. 187-197.