Semantic Enterprise Content Management

to appear in Practical Handbook of Internet Computing, CRC Press

Mark Fisher and Amit Sheth*
Semagix, Inc.
*Also, LSDIS Lab, Computer Science, University of Georgia

Abstract

The emergence and growth of the Internet and vast corporate intranets as information sources has resulted in new challenges with regard to scale, heterogeneity, and distribution of content. Semantics is emerging as the critical tool for enabling more scalable and automated approaches to achieve interoperability and analysis of such content. This chapter discusses how a Semantic Enterprise Content Management system employs metadata and ontologies to effectively overcome these challenges.

Keywords: Semantic Web, Enterprise Content Management, Content Management System, Semantic Technology, Automatic Classification, Semantic Metadata, Ontology, Metadata Annotation, Metadata Extraction, Metadata Enhancement, Knowledge Discovery, Text Analytics

Introduction

Systems for high-volume and distributed data management were once confined to the domain of highly technical and data-intensive industries. However, the general trend in corporate institutions over the past three decades has led to the near obsolescence of the physical file cabinet in favor of computerized data storage. With this increased breadth of data-rich industries, there is a parallel increase in the demand for handling a much wider range of data source formats with regard to syntax, structure, accessibility, and physical storage properties. Unlike the data-rich industries of the past, which typically preferred to store their data within the highest possible degree of structure, many industries today require the same management capabilities across a multitude of data sources of vastly different degrees of structure. In a typical company, employee payroll information is stored in a database, accounting records are stored in spreadsheets, internal company policy reports exist in word-processor documents, marketing presentations exist alongside white papers and web-accessible slideshows, and company financial briefing and technical seminars are available on-line as a/v files and streaming media.

Thus, the "Information Age" has given rise to the ubiquity of Content Management Systems (CMS) for encompassing a wide array of business needs from Human Resource Management to Customer Resource Management, invoices to expense reports, and presentations to emails. This trend has affected nearly every type of enterprise – financial institutions, governmental departments, media and entertainment management - to name but a few. Moreover, the growth rate of data repositories has accelerated to the

point that traditional CMS no longer provides the necessary power to organize and utilize that data in an efficient manner. Furthermore, CMS are often the backbone of more dynamic internal processes within an enterprise such as content analytics, and more public face of an enterprise as seen through its enterprise portal. Result of not having a good CMS would mean lost or misplaced files, inadequate security for highly-sensitive information, non-viable human resource requirements for tedious organizational tasks, and in the worst case it may even lead to unrecognized corruption or fraud perpetrated by malevolent individuals who have discovered and exploited loopholes which will undoubtedly exist within a mismanaged information system.

Current demands for business intelligence require information analysis that acts upon massive and disparate sources of data in an extremely timely manner, and the results of such analysis must provide actionable information that is highly relevant for the task at hand. For such endeavors, machine processing is an indisputable requirement due to the size and dispersal of data repositories in the typical corporate setting. Nonetheless, the difficulty in accessing highly relevant information necessitates an incredibly versatile system that is capable of traversing and "understanding" the meaning of content regardless of its syntactic form or its degree of structure. Humans searching for information can determine with relative ease the meaning of a given document, and during the analytical process will be unconcerned, if not unaware, of differences in the format of that document (e.g. web-page, word processor document, email). Enabling this same degree of versatility and impartiality for a machine requires overcoming significant obstacles, yet, as mentioned above, the size and distribution of data leaves no choice but to confront these issues with machine-processing. A human cannot possibly locate relevant information within a collection of data that exceeds millions or even billions of records, and even in a small set of data, there may be subtle and elusive connections between items that are not immediately apparent within the limits of manual analysis. By applying advanced techniques of semantic technology, software engineers are able to develop robust content management applications with the combined capabilities of intelligent reasoning and computational performance.

"Content," as used throughout this chapter, refers to any form of data that is stored, retrieved, organized, and analyzed within a given enterprise. For example, a particular financial institution's content could include continuously updated account records stored in a Relational Database Management System (RDBMS), customer profiles stored in a shared file system in the form of spreadsheets, employee policies stored as web pages on an intranet, and an archive of email correspondence among the company employees. In this scenario, several of the challenges of CMS are apparent.

This chapter will focus on three such challenges, and for each of these, we will discuss the benefits of applying **semantics** to create an enhanced CMS. Throughout we will emphasize that the goal of any such system should be to increase overall efficiency by maximizing return on investment (ROI) for employees who manage data, while minimizing the technical skill level required of such workers, even as the complexity of information systems grows inevitably in proportion to the amount of data. The trends that have developed in response to these challenges have propelled traditional CMS into the realm of semantics where quality supersedes quantity in the sense that a small set of highly relevant information offers much more utility than a large set of irrelevant information. Three critical enablers of semantic technology - **classification**, **metadata**,

and **ontologies** - are explored in this chapter. Finally we show how the combined application of these three core components may aid in overcoming the challenges as traditional content management evolves into semantic content management.

Primary Challenges for Content Management Systems

1. Heterogeneous Data Sources

First, there is the subtle yet highly complicated issue that most large-scale information systems comprise heterogeneous data sources. These sources differ structurally and syntactically [Sheth 1998]. Retrieving data from a RDBMS, for instance, involves programmatic access (such as ODBC) or, minimally, the use of a query language (SQL). Likewise, the HTML pages that account for a significant portion of documents on the Internet and many intranets are actually composed of marked-up, or *tagged*, text (tags provide stylistic and structural information) that is interpreted by a browser to produce a more human-readable presentation. One of the more challenging environments is when the transactional data needs to be integrated with documents or primarily textual data. Finally, a document created within a word-processing application is stored as binary data and is converted into text using a proprietary interpreter built into the application itself (or an associated "viewer"). Some of the applications, such as Acrobat, provide increasing support for embedding manually entered metadata in RDF and/or based on the Dublin Core metadata standard (to be discussed later). A system which integrates these diverse forms of data in a way that allows for their **interoperability** must create some normalized representation of that data in order to provide equal accessibility for human and machine alike. In other words, while the act of reading an email, a web page, and a word-processor document is not altogether different for a human, a machine is "reading" drastically different material in regard to structure, syntax, and internal representation. Add to this equation the need to manage content which is stored in rich media formats (audio and video files), and the difficulty of such a task is compounded immensely. Thus, for any system that enables automation for managing such diverse content, this challenge of interoperability must be overcome.

2. Distribution of Data Sources

Inevitably, a corporation's content is not only stored in heterogeneous formats, but its data storage systems will likely be distributed among various machines on a network, including desktops, servers, network file-systems, and databases. Accessing such data will typically involve the usage of various protocols (HTTP, HTTPS, FTP, SCP, etc.). Security measures, such as firewalls and user-authentication mechanisms, may further complicate the process of communication among intranets, the Internet, and the World Wide Web. Often, an enterprise's business depends not only on proprietary and internally generated content, but also subscribed syndicated content, or open source and publicly available content. In response to these complexities, an information management system must be extremely adaptable in its traversal methods, highly configurable for a wide variety of environments, and non-compromising in regard to security.

Increasingly, institutions are forming partnerships based upon the common advantage of sharing data resources. This compounds the already problematic nature of data distribution. For example, a single corporation will likely restrict itself to a single database vendor in order to minimize the cost, infrastructure, and human resources required for maintenance and administration of the information system. However, a corporation should not face limitations regarding its decisions for such resource sharing partnerships simply based on the fact that a potential partner employs a different database management system. Even after issues of compatibility are settled, the owner of a valuable data resource will nevertheless want to preserve a certain degree of autonomy for their information system in order to retain control of its contents [Sheth and Larson 1990]. This is a necessary precaution regardless of the willingness to share the resource. Understandably, a corporation may want to limit the shared access to certain views or subsets of its data, and even more importantly, it must protect itself given that the partnership may expire. Technologies within the growing field of Enterprise Application Integration are overcoming such barriers with key developments in generic transport methods (XML, IIOP, SOAP and Web Services). These technologies are proving to be valuable tools for the construction of secure and reliable interface mechanisms in the emerging field of Semantic Enterprise Information Integration (SEII) systems.

3. Data Size and the Relevance Factor

The third, and perhaps most demanding, challenge arises from the necessity to find the most relevant information within a massive set of data. Information systems must deal with content that is not only heterogeneous and distributed but also exceptionally large. This is a common feature of networked repositories (most notably the World Wide Web). A system for managing, processing, and analyzing such data must incorporate filtering algorithms for eliminating the excessive "noise" in order for users to drill down to subsets of relevant information. Such challenges make the requirements for speed and automation critical. Ideally, a CMS should provide increased quality of data management as the quantity of data grows. In the example of a search engine, increasing the amount of data available to the search's indexing mechanism should enable an end user to find not only more but *better* results. Unfortunately it is all too often the case that an increased amount of data leads to exactly the opposite situation where the user's results are distorted due to a high number of false positives. Such distortion results from the system's combined inabilities to determine the contextual meaning of its own contents or the intentions of the end user.

Facing the Challenges: The Rise of Semantics

The growing demands for integrating content, coupled with the unfeasibility of actually storing the content within a single data management system, have given rise to the field of Enterprise Content Management (ECM). Built upon many of the technical achievements of the Document Management (DM) and CMS communities, the applications of ECM must be more generic with regard to the particularities of various data sources, more versatile in its ability to process and aggregate content, more powerful in handling massive and dynamic sources in a timely manner, more scalable in response

to the inevitable rise of new forms of data, and more helpful in providing the most relevant information to its frontend users. While encompassing each of these features, an ECM system must overcome the pervasive challenge of reducing the requirements of manual interaction to a minimum. In designing the functional specifications for an ECM application, system architects and developers focus upon any management task that has traditionally been a human responsibility and investigate the possibilities of devising an automated counterpart. Typically the most challenging of these tasks involve text analytics and decision-making processes. Therefore, many developments within ECM have occurred in parallel with advances in the Artificial Intelligence (AI), lexical and natural language processing, and data management and information retrieval communities. The intersection of these domains has occurred within the realm of semantics.

Enabling Interoperability

The first two challenges presented in the previous section, heterogeneity and distribution, are closely related with regard to their resulting technical obstacles. In both cases, the need for interoperability among a wide variety of applications and interfaces to data sources presents a challenge for machine processibility of the content within these sources. The input can vary widely, yet the output of the data processing must create a normalized view of the content so that it is equally usable (i.e. *machine-readable*) in an application regardless of source. Certain features of data storage systems are indispensable for the necessary administrative requirements of their users (e.g. automated backup, version-tracking, referential integrity), and no single ECM system could possibly incorporate all such features. Therefore, an ECM system must provide this "normalized view" as a portal layer, which does not infringe upon the operational procedures of the existing data infrastructure yet provides equal access to its contents via an enhanced interface for the organization and retrieval of its contents.

While this portal layer exists for the frontend users, there is a significant degree of processing required for the backend operations of data aggregation. As the primary goal of the system is to extract the most relevant information from each piece of content, the data integration mechanism must not simply duplicate the data in a normalized format. Clearly such a procedure would not only lead to excessive storage capacity requirements (again this is especially true in dealing with data from the World Wide Web), but would also accomplish nothing for the relevance factor. One solution to this predicament is an indexing mechanism that analyses the content and determines a subset of the most relevant information, which may be stored within the content's **metadata** (to be discussed in detail later). Because a computer typically exploits structural and syntactic regularities, the complexity of analysis grows more than linearly in relation to the inconsistencies within these content sources. This is the primary reason that many corporations have devoted vast human resources to tasks such as the organization and analysis of data. On the other hand, corporations for which data management is a critical part of the operations typically store as much data as possible in highly-structured systems, such as Relational Database Management Systems (RDBMS), or for smaller sets of data, spreadsheet files may be used. Still other corporations have vast amounts of legacy data that is dispersed in unstructured systems and formats – such as the individual

file-systems of desktop computers in a Local Area Network or email archives or even in a legacy CMS which no longer supports the needs of the corporation.

The Semantic Web

Ironically, the single largest and rapidly growing source of data – the World Wide Web – is a collection of resources that are extremely non-restrictive in terms of structural consistency. This is a result of the fact that a majority of these exist as HTML documents, which are inherently flexible with regard to structure. In hindsight this issue may be puzzling and even frustrating to computer scientists who, in nearly every contemporary academic or commercial environment, will at some point be confronted with such inconsistencies while handling data from the World Wide Web. Nevertheless the very existence of this vast resource is owed largely to the flexibility provided by HTML, as this is the primary enabling factor for non-specialists who have added countless resources to this global data repository. The guidelines for the HTML standard are so loosely defined that two documents which appear identical within a browser could differ drastically in the actual HTML syntax. While this presents no problem for a human reading the web page, it can be a significant problem for a computer processing the HTML for any purpose beyond the mere display in the browser. With XML (eXtensible Markup Language), well-formed structure is enforced, and the result is increased consistency and vastly more reliability in terms of machine-readability. Additionally, XML is customizable (extensible) for any domain-specific representation of content. When designing an XML Schema or DTD, a developer or content provider outlines the elements and attributes which will be used and their hierarchical structure. The developer may specify which elements or attributes are required, which are optional, their cardinality, and basic constraints on the values. XML, therefore, aids considerably in guaranteeing that the content is machine-readable since it provides a template describing what may be expected in the document. XML also has considerably more semantic value since the elements and attributes will typically be named in a way that provides meaning as opposed to simple directives for formatting the display.

For these reasons, the proponents of the "Semantic Web" have stressed the benefits of XML for web-based content storage as opposed to the currently predominant HTML. XML has been further extended by the Resource Description Framework (RDF, described in the section on "ontologies" below), which enables XML tags to be labeled in conjunction with a referential knowledge representation. This in turn allows for machine-based "inferencing agents" to operate upon the contents of the web. Developed for information retrieval within particular domains of knowledge, these specialized agents might effectively replace the web's current "search engines." These are the concepts which may transform the state of the current World Wide Web into a much more powerful and seemingly intelligent resource, and researchers who are optimistic about this direction for the web propose that it will not require a heightened technical-level for the creators or consumers of its contents [Berners-Lee et al., 2001]. It is true that many who upload information to today's web use editors that may completely preclude the need to learn HTML syntax. For the Semantic Web to emerge pervasively, analogous editors would need to provide this same ease of use while infusing semantic information into the content.

Core Components of Semantic Technology

1. Classification

Classification is, in a sense, a coarse-level method of increasing the relevancy factor for a CMS. For example, imagine a news content provider who publishes 1000 stories a day. If these stories were indexed en masse by a search engine with general keyword searching, it could often lead to many irrelevant results. This would be especially true in cases where the search terms are ambiguous in regard to context. For example, the word "bear" could be interpreted as a sports-team's mascot or as a term to describe the current state of the stock market. Likewise, names of famous athletes, entertainers, business executives, and politicians may overlap – especially when one is searching by last-name only. However, these ambiguities can be reduced if an automatic classification system is applied. A simple case would be a system that is able to divide the set of stories into groups of roughly a couple hundred stories each within five general categories, such as World News, Politics, Sports, Entertainment, and Business. If the same keyword searches mentioned above were now applied within a given category, the results would be much more relevant, and the term "bear" will likely have different usage and meaning among the stories segregated by the categories.

Such a system is increasingly beneficial as the search domain becomes more focused. If a set of 1000 documents were all within the domain of Finance and the end-users were analysts with finely tuned expectations, the search parameters might lead to unacceptable results due to a high-degree of overlap within the documents. While the layman may not recognize the poor quality of these results, the analyst, who may be particularly interested in a merger of two companies, would only be distracted by general industry reports that happen to mention these same two companies. In this case the information retrieval may be extremely time-critical (even more critical cases exist – such as national security and law enforcement). A highly specialized classification system could divide this particular set of documents into categories such as "Earnings", "Mergers", "Market Analysis", etc. Obviously such a fine-grained classification system is much more difficult to implement than the earlier and far more generalized example. Nevertheless, with a massive amount of data becoming available each second, such classification may be indispensable. Several techniques of classification may be used to address such needs, including statistical analysis and pattern matching [Joachims, 1998], rule-based methods [Ipeirotis et al., 2000], linguistic analysis [Losee, 1995], probabilistic methods employing Bayesian theory [Cheeseman and Stutz., 1996], and machine-learning methods [Sebastiani 2002] including those based on Hidden Markov Models [Frasconi et al., 2002]. In addition, ontology-driven techniques, such as named-entity and domain-phrase recognition, can vastly improve the results of classification [Hammond et al., 2002]. Studies have revealed that a committee-based approach will produce the best results since it maximizes the contributions of the various classification techniques [Sheth et al., 2002]. Furthermore, studies have also shown that classification results are significantly more precise when the documents to be classified are tagged with metadata resources (represented in XML) and conform to a predetermined schema [Lim and Liu, 2002].

2. Metadata

Metadata can be loosely defined as "data about data." For a discussion of enterprise applications and their metadata-related methodologies for infusing Content Management Systems with semantic capabilities, and to reveal the advantages offered by metadata in semantic content management, we will outline our description of metadata as progressive levels from the perspective of increasing utility. These "levels of metadata" are not mutually exclusive; on the contrary, the accumulative combination of each type of metadata provides a multi-faceted representation of the data including information about its syntax, structure, and semantic context. For this discussion, we use the term "document" to refer to a piece of textual content – the data itself. Given the definition above, each form of metadata discussed here may be viewed in some sense as data *about* the data within this hypothetical document. The goal of incorporating metadata into a CMS is to enable the end-user to find actionable and contextually relevant information. Therefore, the utility of these types of metadata is judged against this requirement of contextual relevance.

A. Syntactic Metadata

The simplest form of metadata is *syntactic* metadata, which provides very general information, such as the document's size, location, or date of creation. Although this information may undoubtedly be useful in certain applications, it provides very little in the way of context determination. However, the assessment of a document's relevance may be partially aided by such information. The date of creation or date of modification for a document would be particularly helpful in an application where highly time-critical information is required and only the most recent information is desired. For example, a news agency competing to have the first release of breaking news headlines may constantly monitor a network of reports where the initial filtering mechanism is based upon scanning only information from the past hour. Similarly a brokerage firm may initially divide all documents based on date and time before submitting to separate processing modules for long-term market analysis and short-term index change reports. These attributes, which describe the document's creator(s), modifier(s), and times of their activity, may also be exploited for the inclusion of version-tracking and user-level access policies into the ECM system. Most document types will have some degree of syntactic metadata. Email header information provides author, date, and subject. Documents in a file-system are tagged with this information as well.

B. Structural Metadata

The next level of metadata is that which provides information regarding the *structure* of content. The amount and type of such metadata will vary widely with the type of document. For example, an HTML document may have many tags, but as these exist primarily for purposes of formatting, they will not be very helpful in providing contextual information for the enclosed content. XML, on the other hand, offers exceptional capabilities in this regard. While it is the responsibility of the document

creator to take full advantage of this feature, structural metadata is generally available from XML. In fact, the ability to enclose content within meaningful tags is usually the fundamental reason one would choose to create a document in XML. Many "description languages," which are used for the representation of knowledge, are XML-based (some will be discussed in the section on *ontologies* below). For determining contextual relevance and making associations between content from multiple documents, structural metadata is more beneficial than merely syntactic metadata, since it provides information about the topic of a document's *content* and the items of interest within that content. This is clearly more useful in determining context and relevance when compared to the limitations of syntactic metadata for providing information about the document itself.

C. Semantic Metadata

In contrast to the initial definition of metadata above, we may now construct a much more pertinent definition for *semantic* metadata as: "data which may be associated explicitly or implicitly with a given piece of content (i.e. a document) and whose relevance for that content is determined by its ontological position (its context) within one or more domains of knowledge." In this sense, metadata is the building block of semantics. It offers an invaluable aid in classification techniques, it provides a means for high-precision searching, and, perhaps most importantly, it enables interoperability among heterogeneous data sources.

How does semantic metadata empower a Content Management System to better accomplish each of these tasks? In the discussion that follows, we will provide an in-depth look at how metadata can be leveraged against an ontology to provide fine-grained contextual relevancy for information within a given domain or domains. As we briefly mentioned in the discussion of classification techniques in the previous section, the precision of classification results may be drastically augmented by the use of domain-knowledge. In this case, the method is *named entity recognition*.

Named entity recognition involves finding items of potential interest within a piece of text. A named entity may be a person, place, thing, or event. If these entities are stored within an ontology, then a vast amount of information may be available. It is precisely this *semantic* metadata that allows for interoperability across a wide array of data storage systems, because the metadata that is extracted from any document may be stored as a "snapshot" of that document's relevant information. The metadata contained within this snapshot simply references the instances of named-entities, which are stored in the ontology. Therefore, there is a rich resource of information available for each named-entity including synonyms, attributes, and other related entities. This enables further "linking" to other documents on three levels: those containing the same explicit metadata (mention the exact same entities), those containing the same metadata implicitly (such as synonyms or hierarchically related named-entities), and those related by ontological associations between named-entities (one document mentions a company's name while another simply mentions its ticker symbol). This process in effect normalizes the vastly different data sources by referencing the backend ontology, and while this exists "behind the scenes," it allows for browsing and searching within the front-end portal layer.

D. Metadata Standards

The use of metadata for integrating heterogeneous data [Bornhövd, 1999; Snijder, 2001] and managing heterogeneous media [Sheth and Klas, 1998; Kashyap et al., 1995] has been extensively discussed, and an increasing number of metadata standards are being proposed and developed throughout the information management community to serve the needs of various applications and industries. One such standard which has been well accepted is the Dublin Core Metadata Initiative (DCMI). Figure 1 shows the 15 elements defined by this metadata standard. It is a very generic element set flexible enough to be used in content management regardless of the domain of knowledge. Nevertheless, for this same reason, it is primarily a set of syntactic metadata as described above; it offers information about the document but offers very little with regard to the structure or content of the document. The semantic information is limited to the "Resource Type" element, which may be helpful for classification of documents, and the inclusion of a "Relation" element, which allows for related resources to be explicitly associated. In order to provide more semantic associations through metadata, this element set could be extended with domain-specific metadata tags. In other words, the Dublin Core metadata standard may provide a useful parent class for domain-specific document categories.

Element	Description
Title	A name given to the resource.
Contributor	An entity responsible for making contributions to the content of the resource.
Creator	An entity primarily responsible for making the content of the resource.
Publisher	An entity responsible for making the resource available.
Subject and Keywords	The topic of the content of the resource.
Description	An account of the content of the resource.
Date	A date associated with an event in the life cycle of the resource.
Resource Type	The nature or genre of the content of the resource.
Format	The physical or digital manifestation of the resource.
Resource Identifier	An unambiguous reference to the resource within a given context.
Language	A language of the intellectual content of the resource.
Relation	A reference to a related resource.
Source	A Reference to a resource from which the present resource is derived.
Coverage	The extent or scope of the content of the resource.
Rights Management	Information about rights held in and over the resource.

Figure 1: Dublin Core Metadata Initiative as described in the Element Set Schema [DCMI, 2002]

The Learning Technology Standards Committee (LTSC), a division of the IEEE, is developing a similar metadata standard, known as Learning Object Metadata (LOM). LOM provides slightly more information regarding the structure of the object being

described, yet it is slightly more specialized with a metadata element set that focuses primarily upon technology-aided educational information [LTSC, 2000].

The National Library of Medicine has created a database for medical publications known as MEDLINE, and the search mechanism requires that the publications be submitted according to the PubMed XML specification (the DTD is located at: http://www.ncbi.nlm.nih.gov/entrez/query/static/PubMed.dtd). Once more, the information is primarily focused upon authorship and creation date, but it does include an element for uniquely identifying each article, which is helpful for indexing the set of documents. This would be particularly helpful if some third-party mechanism were used to traverse, classify, and create associations between documents within this repository.

The next section will demonstrate how **ontologies** provide a valuable method for finding implicit semantic metadata in addition to the explicitly mentioned domain-specific metadata within a document [see figure 2]. This ability to discover implicit metadata enables the annotation process to proceed to the next level of **semantic enhancement,** which in turn allows the end-user of a semantic CMS to locate contextually relevant content. The enhancement of content with non-explicit semantic metadata will also enable analysis tools to discover non-obvious relationships between content.

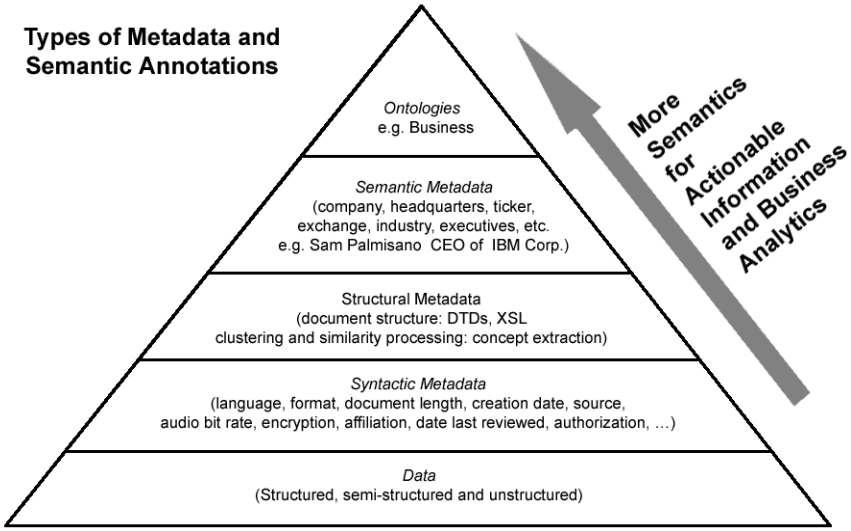

Figure 2: Filtering to highly relevant information is achieved as the type of semantic annotations and metadata progress toward domain-modeling through the use of ontologies.

3. Ontologies

Although the term "**ontology**" originated in Philosophy where it means the "study of Existence" (*ontos* is the Greek word for 'being'), there is a related yet more pragmatic and concrete meaning for this term in Computer Science; an ontology is a representation of a domain of knowledge. To appreciate the benefits offered by an ontological model within a content management system, we will convey the intricacies and the features of such a system in comparison with other, more basic forms of knowledge representation. In this way, the advantages of using an ontological model will be presented as a successive accumulation of its forebears. The use of ontologies to provide underpinning

for information sharing, heterogeneous database integration, and semantic interoperability has been long realized [Gruber, 1991; Kashyap and Sheth, 1994; Sheth, 1998; Wache et al., 2001].

A. Forms of Knowledge Representation

The simplest format for knowledge representation is a **dictionary**. In a sense, a dictionary may be viewed as nothing more than a table where the "terms" are the keys and their "definitions" are the values. In the most basic dictionary – disregarding etymological information, example sentences, synonyms and antonyms – there are no links between the individual pieces of knowledge (the "terms"). Many more advanced forms of knowledge organization exist, yet the differences are sometimes subtle and thus terminology is often misused (http://www.kmconnection.com/C_and_R_definitions.htm). From a theoretical viewpoint, when antonyms and synonyms are included, one is dealing with a **thesaurus** as opposed to a dictionary. The key difference is a critical one and one that has massive implications for network technologies: the pieces of knowledge are *linked*. Once the etymological information is added (derivation) and the synonyms are organized hierarchically (inheritance), the thesaurus progresses to the next level, **taxonomy**. The addition of hierarchical information to a thesaurus means, for instance, that no longer is "plant" simply synonymous with "flower," but a flower *is a type of* (or *subclass of*) plant. Additionally, we know that a tulip *is a type of* flower. In this way, the relations between the pieces of knowledge, or **entities**, take the form of a tree structure as the representation progresses from thesaurus to taxonomy. Now, with the tree structure, one may derive other forms of association besides "is a subclass/is a superclass"; for example, the tulip family and the rose family are both subclasses of flower, and therefore they are related to each other as siblings. Despite this, a basic taxonomy limits the forms of associativity to these degrees of relatedness, and although such relationships can create a complex network and may prove quite useful for certain types of data analysis, there are many other ways in which entities may be related. In an all inclusive knowledge representation, a rose may be *related to* love in general or Valentine's Day in particular. Similarly, the crocus may be associated with spring, and so on. In other words, these associations may be emotional, cultural, or temporal. The fundamental idea here is that some of the most interesting and relevant associations may be those that are discovered or traversed by a data-analysis system utilizing a reference knowledgebase whose structure of entity relationships is much deeper than that of a basic taxonomy; rather than a simple tree, such a knowledge structure must be visually represented as a web. Finally, in adding one last piece to this series of knowledge representations, we arrive at the level of **ontology** which is most beneficial for *semantic* content management. This addition is the *labeling* of relationships; the associations are provided with contextual information. From the example above, we could express that "a rose-symbolizes-love" or "a crocus-blooms in-Spring." Now these entities are not merely associated but are associated *in a meaningful way*. Labeled relationships provide the greatest benefit in cases where two types of entities may be associated in more than one way. For example, we may know that Company A is associated with Company B, but this alone will not tell us if Company A is a competitor of Company B, or if Company A is a subsidiary of Company B, or vice versa.

B. Ambiguity Resolution

Returning to the flower example above, we will present an even greater challenge. Assuming an application uses a reference knowledgebase which is in the form of a general but comprehensive ontology (such as the lexical database, WordNet, described below), determining the meaning of a given entity, such as "plant" or "rose," may be quite difficult. It is true that there is a well-defined instance of each word in our ontology with the meanings and associations as intended in the examples outlined previously. Still, the application also may find an instance of "plant," which is synonymous with "factory," or a color known as "rose." To resolve such ambiguities, the system must analyze associated data from the context of the extracted entity within its original source. If several other known flowers whose names are not used for describing colors were mentioned in the same document, then the likelihood of that meaning would become evident. More complex techniques may be used, such as linguistic analysis, which could determine that the word was used as a noun, while the color, "rose" would most likely have been used as an adjective. Another technique would rely upon the reference ontology where recognition of associated concepts or terms would increase the likelihood of one meaning over the other. If the document also mentioned "Valentine's Day," which we had related to "roses" in our ontology, this would also increase the likelihood of that meaning. Programmatically, the degree of likelihood may be represented as a "score" with various parameters contributing weighted components to the calculation. For such forms of analysis, factors such as proximity of the terms and structure of the document would also contribute to the algorithms for context determination.

C. Ontology Description Languages

With steadily growing interest in areas such as the Semantic Web, current research trends have exposed a need for standardization in ontology representation. For semantic content management, such standardization would clearly be advantageous. The potential applications for knowledge sharing are innumerable, and the cost benefit of minimizing redundancy in the construction of comprehensive domain ontologies is indisputable. Nevertheless, there are two key obstacles for such endeavors. First, the construction of a knowledge model for a given domain is a highly subjective undertaking. Decisions regarding the granularity of detail, hierarchical construction, and determination of relevant associations each offer an infinite range of options. Second, there is the inevitable need for combining independently developed ontologies via intersections and unions, or analyzing subsets and supersets. This integration of disparate ontologies into a normalized view requires intensive heuristics. If one ontology asserts that a politician *is affiliated with* a political party while another labels the same relationship as "politician *belongs to* party," the integration algorithm would need to decide if these are two distinct forms of association or if they should be merged. In the latter case, it must also decide which label to retain. Although the human ability to interpret such inconsistencies is practically instinctual, to express these same structures in a machine-readable form is another matter altogether.

Among the prerequisites of the Semantic Web that are common with those of semantic content management is this ability to deal with multiple ontologies. In one

sense, the Semantic Web may be viewed as a *global* ontology that reconciles the differences among *local* ontologies and supports query processing in this environment [Calvanese et al., 2002]. Such query processing should enable the translation of the query terms into their appropriate meanings across different ontologies in order to provide the benefits of semantic search as compared to keyword-based search [Mena et al., 2000]. The challenges associated with ontology integration vary with regard to the particularities of the task. Some examples are the reuse of an existing ontological representation as a resource for the construction of a new ontology, the unification or merging of multiple ontologies to create a deeper or broader representation of knowledge, and the incorporation of ontologies into applications that may benefit from their structured data [Pinto et al., 1999].

Recently there have been many key developments in response to these challenges of ontology assimilation. XML lies at the foundation of these *ontology description languages*, since the enforcement of consistent structure is a prerequisite to any form of knowledge model representation that aspires to standardization. To evolve from the structural representations afforded by XML to an infrastructure suitable for representing a semantic network of information requires the inclusion of capabilities for the representation of associations. One of the most accepted candidates in this growing field of research is the Resource Description Framework (RDF) [W3C, 1999] and its outgrowth RDF-Schema (RDF-S) [W3C, 2003].

RDF-S provides a specification that moves beyond ontological representation capabilities to those of ontological modeling. The addition of a "schema" brings object-oriented design aspects into the semantic framework of RDF. In other words, hierarchically structured data models may be constructed with a separation between class-level definitions and instance-level data. This representation at the "class-level" is the actual schema, also known as the *definitional* component, while the instances constitute the factual, or *assertional*, component. When a property's class is defined, constraints may be applied with regard to possible values, as well as which types of resource a particular instance of that property may describe.

The DARPA Agent Markup Language (DAML), in its latest manifestation as DAML+OIL (Ontology Inference Layer), expands upon RDF-S with extensions in the capabilities for constructing an ontology model based on constraints. In addition to specifying hierarchical relations, a class may be related to other classes in disjunction, union, or equality. DAML+OIL provides a description framework for restrictions in the mapping of property values to datatypes and objects. These restriction definitions outline such constraints as the required values for a given class or its cardinality limitations (maximum and minimum occurrences of value-instances for a given property). The W3C Web Ontology Working Group (WebOnt) has created a Web Ontology Language, known as OWL (http://www.w3.org/TR/owl-ref/), which is derived from DAML+OIL and likewise follows the RDF specification.

The *F-Logic* language has also been used in ontology building applications. F-Logic, which stands for "Frame Logic," is well suited to ontology description although it was originally designed for representing any object-oriented data model. It provides a comprehensive mechanism for the description of object-oriented class definitions including "object identity, complex objects, inheritance, polymorphic types, query methods, encapsulation, and others" [Kifer et al., 1990]. The OntoEdit tool, developed at

the AIFB of the University of Karlsruhe, is a graphical environment for building ontologies. It is built upon the framework of F-Logic, and with the OntoBroker "inference engine" and associated API, it allows for the importing and exporting of RDF-Schema representations. F-Logic also lies at the foundation of other systems that have been developed for the integration of knowledge representation models through the transformation of RDF. The first of these "inference engines" was SiLRI (Simple Logic-based RDF Interpreter [Decker et al., 1998]), which has given way to the open source transformation language, TRIPLE [Sintek and Decker., 2002], and a commercial counterpart offered by Ontoprise GmbH [http://www.ontoprise.com].

D. Sample Knowledgebases

Several academic and industry-specific projects have led to the development of shareable knowledgebases as well as tools for accessing and adding content. One such knowledgebase is the lexical database, WordNet, whose development began in the mid-1980s at Princeton University. WordNet is structured as a networked thesaurus in the form of a "lexical matrix," which maps *word forms* to *word meanings* with the possibility of many-to-many relationships [Miller et al., 1993]. The full range of a thesaurus' *semantic relations* may be represented in WordNet. The set of all word meanings for a given word form a *synset*. The synset may represent any of the following lexical relations: synonymy (same or similar meaning), antonymy (opposite meaning), hyponymy/hypernymy (hierarchical *is a*/*has a* relation), and meronymy/holonymy (*has a part*/*is a part of* relation). Additionally, WordNet allows for correlations between morphologically inflected forms of the same word, such as plurality, possessive forms, gerunds, participles, different verb tenses, etc. While WordNet has been a popular and useful resource as a comprehensive thesaurus with a machine-readable syntax, it is not a formal ontology since it only represents the lexical relations listed above and does not provide contextual associations. It is capable of representing that a "branch" is synonymous with a "twig" or a "department" within an institution. If the first meaning is intended, then it will reveal that a branch is part of a tree. However, for the second meaning, it will not discover the fact that an administrative division typically has a chairman or vice president overseeing its operations. This is an example of the labeled relationships required for the representation of "real world" information. Such associations are lacking in a thesaurus but may be stored in an ontology. WordNet has still been useful as a machine-readable lexical resource and, as such, is a candidate for assimilation into ontologies. In fact, there have been efforts to transform WordNet into an ontology with a greater ability to represent the world as opposed to merely representing language [Oltramari et al., 2002].

In the spirit of cooperation which will be required for the Semantic Web to succeed, the Open Directory Project is a free and open resource, and on the website (http://www.dmoz.org/), it claims to be the "largest, most comprehensive human-edited directory of the Web." The directory structure is designed with browsing in mind as opposed to searching and is primarily a hierarchical categorization of web resources, which allows multiple classifications for any given resource. Therefore, it is not an ontology, rather it may be loosely referred to as a taxonomy of web resources which have been manually, and therefore subjectively, classified. It is maintained by volunteers who each agree to supervise a category. While this undoubtedly raises questions with regard

to the authority and consistency of the resource, its success and growth are promising signs for the future of Semantic Web.

The National Library of Medicine has developed an ontology-driven system, known as the Unified Medical Language System (UMLS), for the assimilation, organization, and retrieval of medical information. Intended for integration of data sources ranging from biology, anatomy, and organic chemistry to pharmacology, pathology, and epidemiology, it provides an invaluable resource for medical researchers and practitioners alike (http://www.nlm.nih.gov/research/umls/).

Since many of the researchers and institutions involved in the creation of these and other large knowledgebases are constantly striving for increased shareability, it is feasible that the level of standardization will soon enable the construction of a single high-level Reference Ontology that integrates these various domains of knowledge [Hovy, 1997].

Applying Semantics in ECM

Toolkits

Any semantic CMS must be designed in a generic way that provides flexibility, extensibility, and scalability for customized applications in any number of potential domains of knowledge. Because of these requirements, such a system should include a toolkit containing several modules for completing these necessary customization tasks. The user of such a toolkit should be able to manage the components outlined in the previous section. For each task the overall goal should be to achieve the optimum balance between configurability and automation. Ideally, these tasks are minimally interactive beyond the initialization phase. In other words, certain components of the system, such as content extraction agents and classifiers, should be fully automated after the configurable parameters are established, but a user may want to tweak the settings in response to certain undesired results. The highest degree of efficiency and quality would be achieved from a system that is able to apply heuristics upon its own results in order to maximize its precision and minimize its margin of error. For example, if in the early stages, a user makes adjustments to a particular data extractor or classification module after manually correcting a document's metadata associations and category specification, the system could recognize a pattern in the adjustments so that future occurrences of such a pattern would trigger automatic modifications of the corresponding configuration parameters.

The classification procedure should be configurable in terms of domain specification and granularity within each domain. Additionally, if such a feature is available, the user should be able to fine-tune the scoring mechanisms involved in the interaction of multiple classifier methods. If the classification module requires training sets, the method for accumulating data to be included in these sets should be straightforward. Creation of content extraction agents should also be handled within a user-friendly, graphical environment. Because tweaking parameters for the crawling and extraction agents may be necessary, the toolkit should include a straightforward testing module for these agents that produces feedback to guide the user in constructing these rules for gathering metadata as precisely as possible. The same approach should be taken when designing an ontology-modeling component. Due to the inherent complexity of an ontological

knowledge representation and the importance of establishing this central component of the system, it is critical that this component provide an easily navigable, visual environment. A general description of the requirements for ontology editing and summary of several tools that address these needs may be found in [Denny, 2002]. Finally, an important feature for any content management system is some form of auditing mechanism. In many industries, there is a need for determining the reliability of content sources. Keeping track of this information aids in the determination of the "reliability" of content. Once again, the World Wide Web is the extreme case where it is very difficult to determine if a source is authoritative. Likewise, tracking the date and time of content entering the system is important, especially for the institutions where timeliness has critical implications – news content providers, law enforcement, financial services, etc.

The Karlsruhe Ontology and SemanticWeb Tool Suite (KAON) is an example of a semantic content management environment. It consists of a multi-layered architecture of services and management utilities [Bozsak et al., 2002]. This is the same set of tools which contains the OntoEdit GUI environment for ontology modeling, which has been described in the discussion of ontology description languages above. KAON has been developed with a particular focus on the Semantic Web. Another suite of tools is offered by the ROADS project, which has been developed by the Access to Networked Resources section of eLib (the Electronic Libraries Programme). ROADS provides tools for creating and managing information portals, which they refer to as "subject gateways" [http://www.ilrt.bristol.ac.uk/roads/].

Semantic Metadata Extraction

Traditionally, when dealing with heterogeneous, dispersed, massive and dynamic data repositories, the overall quality or relevance of search results within that data may be inversely proportional to the number of documents to be searched. As can be seen from any major keyword-based search engine, as the size of data to be processed grows, the number of false-positives and irrelevant results grows accordingly. When these sources are dynamic (the World Wide Web again being an extreme case), the resulting "links" may point to nothing at all, or - what is often even worse for machine processing applications - they may point to different content than that which was indexed. Therefore two major abilities are favorable in any system that crawls and indexes content from massive data repositories: the extraction of the semantic metadata from each document for increased relevance, and an automated mechanism, so that this extraction will maintain reliable and timely information. The complexity of metadata extraction among documents of varying degrees of structure presents an enormous challenge for the goal of automation.

A semantic ECM toolkit should provide a module for creating extractor agents, which act as *wrappers* to content sources (e.g. a web-site or file-system). The agent will follow certain rules for locating and extracting the relevant metadata. Obviously this is not a trivial task when dealing with variance in the source structure. While the World Wide Web offers the greatest challenges in this regard, it is understandably the most popular resource for extraction. Several extraction wrapper technologies have focused upon crawling and retrieving data from web pages such as the WysiWyg Web Wrapper

Factory (W4F), which provides a graphical environment and a proprietary language for formulating retrieval and extraction rules [Sahuget and Azavant, 1999]. ANDES is a similar wrapper technology that incorporates regular expressions and XPath rules for exploiting structure within a document [Myllymaki, 2001]. Semi-automatic wrapper-generation is possible with the XML based XWRAP toolkit, which enables interactive rule formulation in its test environment. Using an example input document, the user selects "semantic tokens," and the application attempts to create extraction rules for these items, but since the structure of input documents may vary considerably, the user must enter new URLs for testing and adjust the rules as necessary [Liu et al., 2000]. Likewise, S-CREAM (Semi-automatic CREAtion of Metadata) allows the user to manually annotate documents and later applies these annotations within a training mechanism to enable automated annotation based on the manual results. The process is aided by the existence of an ontology as a reference knowledgebase for associating the "relational metadata" with a given document [Handschuh et al., 2002]. A fully automatic method for extracting and enhancing metadata is not only the preferred method for the obvious reason that it minimizes manual supervision of the system, but such a method is also most flexible in that it will be equally integrated into a push or pull data aggregation environment [Hammond et al., 2002]. Although the majority of research in crawling and extraction technologies has been undertaken in academic institutions, commercial metadata extraction products have been developed by corporations such as Semagix [Sheth et al., 2002] and Ontoprise (http://www.ontoprise.com).

Semantic Metadata Annotation

It has been stressed that achieving interoperability among heterogeneous and autonomous data sources in a networked environment requires some ability to create a normalized view. Minimally this could be a "metadata snapshot" generated by semantic annotation. If an ontology that is comprehensive for the domain at hand exists in the backend and the interfacing mechanisms for handling distributed data of various formats reside on the frontend, then after filtering the input through a classifier to determine its contextual domain, the system will be able to apply "tagging" or "markup" for the recognized entities. Because of the inclusion of the classification component, the tagged entities would be contextually relevant. An advanced system would also have the ability to *enhance* the content by analyzing known relationships between the recognized entities and those that should be associated with the entity due to implied reference. For example, a story about a famous sports personality may or may not mention that player's team, but the metadata enhancement process would be able to include this information. Similarly a business article may not include a company's ticker symbol, but a stock analyst searching for documents by ticker symbol may be interested in the article. If the metadata enhancement had added the ticker symbol, which it determined from its relationship with the company name, then the analyst would be able to find this article when searching with the ticker symbol parameter alone. No keyword-based search engine would have returned such an article in its result set due to the fact that the ticker symbol's value is simply not present in the article. Implied entities such as these may be taken for granted by a human reading a document, but when a machine is responsible for the analysis of content, an ontology-driven classifier coupled with a domain-specific metadata annotator will enable the user to find highly relevant information in a timely

manner. Figure 3 shows an example of semantic annotation of a document. Note that the entities are not only highlighted, but the types are also labeled.

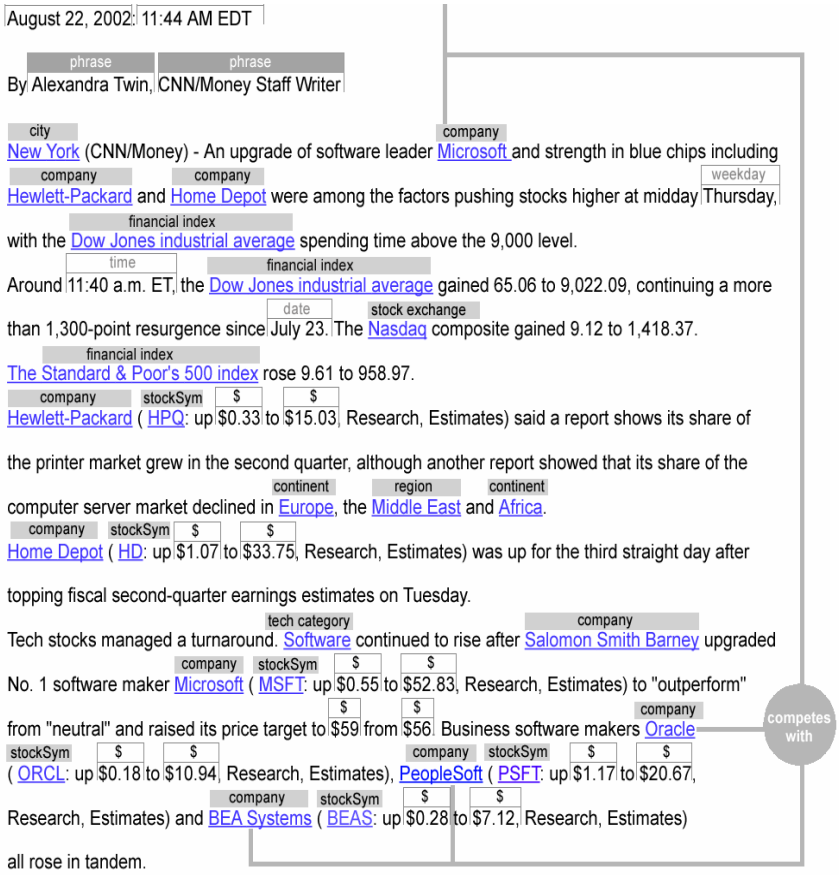

Figure 3: Example of semantic metadata annotation.
Note that named-entities, currency values, and dates and times are highlighted and labeled according to their classification. Also relationships between entities are labeled. This information comes from a reference ontology (Semagix, Inc.).

Semantic Querying

Two broad categories account for a majority of human information gathering: searching and browsing. Searching implies a greater sense of focus and direction while the connotations of "browsing" are that of aimless wandering with no predefined criteria to satisfy. Nevertheless, it is increasingly the case that browsing technologies are employed to locate highly precise information. For example, in law enforcement, a collection of initial evidence may not provide any conclusive facts, yet this same evidence may reveal non-obvious relationships when taken as a starting point within an ontology-driven semantic browsing application. Ironically, searching for information with most keyword-based search engines typically leads the user into the process of browsing before finding the intended information if indeed it is found at all.

The term "query" is more accurate for discussing the highly configurable type of search that may be performed in a semantic content management application. A query consists of not only the search term(s), but also a set of optional parameter values. For

example, if these parameters correspond to the same categories that drive the classification mechanism, then the search term(s) may be mapped into the corresponding entities within the domain-specific ontology. The results of the query thus consist of documents whose metadata had been extracted and contained references to these same entities. In this manner, semantic querying provides much higher precision than keyword-based search owing to its ability to retrieve contextually relevant results. Clearly semantic querying is enabled by the semantic ECM system that we have outlined in this chapter. It requires the presence of a domain-specific ontology and the processes that utilize this ontology: the ontology-driven classification of content, and the extraction of domain-specific and semantically enhanced metadata for that content.

To fully enable custom applications of this semantic querying in a given enterprise, a semantic ECM system should also include flexible and extensible APIs. In most cases, the users of such a system will require a custom front-end application for accessing information that is of particular interest within their organization. For example, if an API allows for the simple creation of a dynamic web-based interface to the underlying system, then the application will appeal to a wide audience without compromising its capabilities. While APIs enable easier creation and extension of the ontology, visualization tools offer a complimentary advantage for the browsing and viewing of the ontology on a schema and/or instance level. Figure 4 shows one such tool, the Semagix Visualizer (http://www.semagix.com).

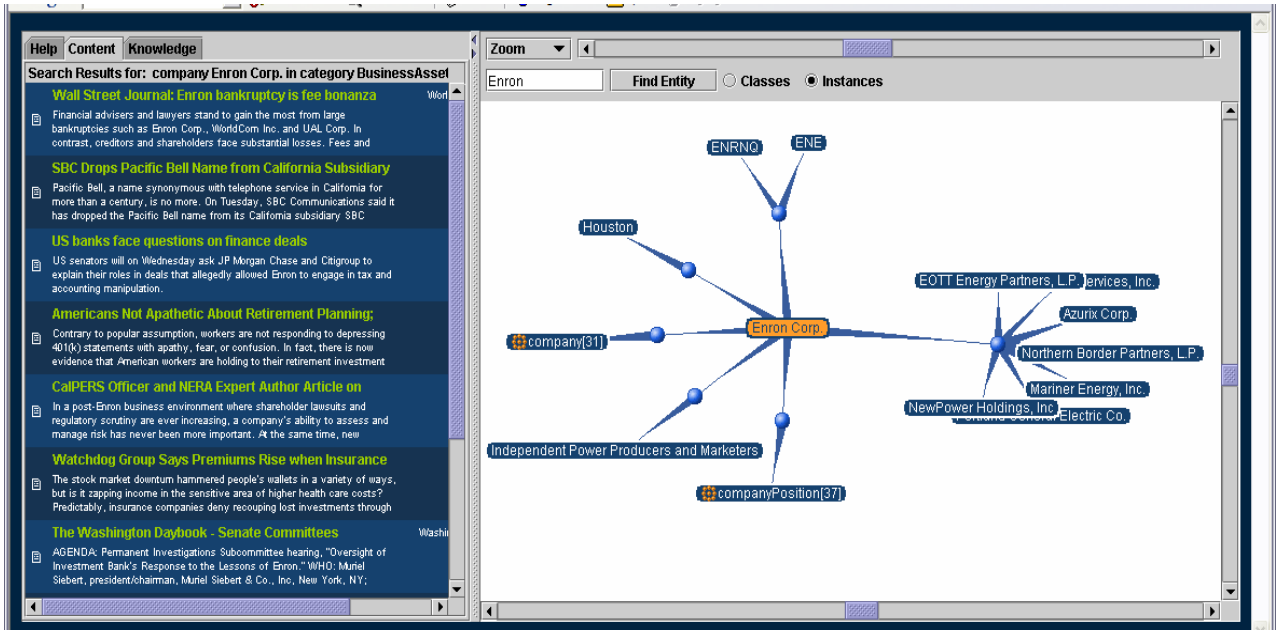

Figure 4: An example of an ontology visualization tool.
The Semagix Visualizer provides a navigable view of the ontology on the right-hand side while the left-hand panel displays either associated documents or more detailed knowledge.

Knowledge Discovery

It has been stressed that machine processing is indispensable when dealing with massive data sources within humanly insurmountable time constraints. Another major benefit of machine processing related to semantic content management is the ability to discover non-obvious associations within that content. For example, while manually sifting through documents or browsing files, it is highly unlikely that one would happen to discover a relationship between two persons which consisted of a chain of three or more associations. For example, in a law enforcement scenario where two suspects "personA" and "personB" are under investigation, it may be important to know that personA lived in the same apartment complex as the brother of a man who was a co-worker of a woman who shared a bank account with personB. Similarly complex associations may be pertinent for a financial institution processing credit reports or a federal agency doing a background check for job applicants.

Obviously the exact definition of such scenarios will differ considerably dependent upon the application. Therefore, any semantic content management system that aims to support automated knowledge discovery should have a highly configurable module for designing templates for such procedures. It would be necessary for a user to determine which types of entities may be meaningfully associated with each other, which relationships are important for traversal between entities, and possibly even a weighted scoring mechanism for calculating the relative level of association for a given relationship. For example, two people working for the same company would most likely receive a higher "weight" than two people living in the same city. Nevertheless, the procedure could be programmed to handle even more advanced analytics such as factoring in the size of the company and the size of the city so that two people living in New York City would receive very little "associativity" compared with two people in Brunswick, Nebraska.

Other less directed analysis may be employed with very similar processing. For example, when dealing with massive data repositories, knowledge discovery techniques may find associations between entities that were mentioned together in documents more than 10 times (or some predetermined threshold) and flag these as "related" entities to be manually confirmed. This type of application may be applied for finding non-obvious patterns in large data sets relatively quickly. As a filtering mechanism, such a procedure could significantly amplify timeliness and relevance, and in many cases these results would have been impossible to obtain from manual analysis regardless of the time constraints. A framework of complex semantic relationships is presented in [Sheth et al., 2003], and a formal representation of one type of complex relationships called *semantic associations* is presented in [Anyanwu and Sheth, 2003].

Conclusion

Enterprise information systems comprise heterogeneous, distributed, and massive data sources. Content from these sources differs systemically, structurally, and syntactically, and accessing that content may require using multiple protocols. Despite these challenges, timeliness and relevance are absolutely required when searching for

information, and therefore the amount of manual interaction must be minimized. To overcome these challenges, a system for managing this content must achieve interoperability, and the key to this is semantics. However, enabling a machine to read in documents of varying degrees of structure from heterogeneous data sources and "understand" the meaning of each document in order to find associations among those documents is not a trivial task.

Advanced classification techniques may be employed for filtering data into precise categories, or domains. The domains should be defined as metadata schemas, which basically outline the items of interest that may occur within a document in a given category (such as "team" in the sports domain or "ticker symbol" in the business domain). Therefore, each piece of content may be annotated (or "tagged") with the instances of these metadata classes. As a collection of semantic metadata, a document can become significantly more machine-readable than in its original format. Moreover, the excess has been removed so that only the contextually relevant information remains.

This notion of tagging documents with the associated metadata according to a pre-defined schema is fundamental for the proponents of the Semantic Web. Metadata schemas also lie at the foundation of most languages used for describing ontologies. An ontology provides a valuable resource for any semantic content management system, since the metadata within a document may be more or less relevant depending upon its location within the referential knowledgebase. Furthermore, an ontology may be used to actually enrich the metadata associated with a document by including implicit entities which are closely related to the explicitly mentioned entities in the given context.

Applications that make use of ontology-driven metadata extraction and annotation are becoming increasingly popular within both the academic and commercial environments. Because of their versatility and extensibility, such applications are suitable candidates for a wide range of content management systems, including Document Management, Web Content Management, Digital Asset Management, and Enterprise Application Integration. The leading vendors have developed refined toolkits for managing and automating the necessary tasks of semantic content management. As the visibility of these products increases, traditional content management systems will be superseded by systems that enable heightened relevance in information retrieval by employing ontology-driven classification and metadata extraction. These semantic-based systems will permeate the enterprise market.

References

Kemafor Anyanwu and Amit Sheth, "The ρ Operator: Discovering and Ranking Associations on the Semantic Web," Proceedings of the Twelfth International World Wide Web Conference, Budapest, Hungary, May 2003.

Tim Berners-Lee, James Hendler, and Ora Lassila, "The Semantic Web, A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities", Scientific American, May 2001.

Christof Bornhövd. "Semantic Metadata for the Integration of Web-based Data for Electronic Commerce", International Workshop on Advance Issues of E-Commerce and Web-Based Information Systems, 1999.

E. Bozsak, M. Ehrig, S. Handschub, Hotho, et al: KAON -- Towards a Large Scale Semantic Web. In: K. Bauknecht, A. Min Tjoa, G. Quirchmayr (Eds.): Proc. of the 3rd Intl. Conf. on E-Commerce and Web Technologies (EC-Web 2002), 2002, 304-313.

Diego Calvanese, Giuseppe De Giacomo, and Maurizio Lenzerini. "A framework for ontology integration", In Proc. of the First Semantic Web Working Symposium, pages 303-316, 2001.

Peter Cheeseman and John Stutz. "Bayesian Classification (AutoClass): Theory and Results", in Advances in Knowledge Discovery and Data Mining, Usama M. Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, & Ramasamy Uthurusamy, Eds., pages 153-180, 1996.

DCMI. Dublin Core Metadata Initiative, 2002. URL: http://dublincore.org/2002/08/13/dces

Stefan Decker, Dan Brickley, Janne Saarela, Jurgen Angele. "A Query and Inference Service for RDF", in Proceedings of the W3C Query Languages Workshop (QL-98), Boston, MA, December 3-4, 1998.

M. Denny. Ontology Building: A Survey of Editing Tools, 2002. available at: http://www.xml.com/pub/a/2002/11/06/ontologies.html

Paolo Frasconi, Giovanni Soda, Alessandro Vullo. "Hidden Markov Models for Text Categorization in Multi-Page Documents", Journal of Intelligent Information Systems, vol. 18, no. 2-3, pages 195-217, 2002.

Thomas Gruber. The Role of Common Ontology in Achieving Sharable,
Reusable Knowledge Bases. in Principles of Knowledge Representation and Reasoning, James Allen, Richard Fikes, and Erik Sandewall, eds, Morgan Kaufman, San Mateo, CA, pages 601-602, 1991.

Brian Hammond, Amit Sheth, and Krzysztof Kochut, "Semantic Enhancement Engine: A Modular Document Enhancement Platform for Semantic Applications over Heterogeneous Content," in Real World Semantic Web Applications, V. Kashyap and L. Shklar, Eds., IOS Press, 2002.

Siegfried Handschuh, Steffen Staab, and Fabio Ciravegna, "S-CREAM: Semi-automatic Creation of Metadata," in 13th International Conference on Knowledge Engineering and Knowledge Management, October 2002.

Eduard Hovy. "A Standard for Large Ontologies," Workshop on Research & Development Opportunities in Federal Information Services, Arlington, VA, MAY 1997. Available at: http://www.isi.edu/nsf/papers/hovy2.htm

Panagiotis Ipeirotis, Luis Gravano, and Mehran Sahami, "Automatic Classification of Text Databases through Query Probing," in Proceedings of the ACM SIGMOD Workshop on the Web and Databases, May 2000.

Thorsten Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," in Proc. of Tenth European Conference on Machine Learning, pages137-142, 1998.

Vipul Kashyap and Amit Sheth, "Semantics-based Information Brokering," in Proceedings of the Third International Conference on Information and Knowledge Management (CIKM), pages 363-370, November 1994.

Vipul Kashyap, Kshitij Shah, and Amit Sheth, "Metadata for building the MultiMedia Patch Quilt," in Muitimedia Database Systems: Issues and Research Directions, S..Jajodia and V.S.Subrahmaniun, Eds., Springer-Verlag, 1995, pp. 297-323.

Michael Kifer, Georg Lausen, and James Wu. Logical Foundations of Object-Oriented and Frame-Based Languages. Technical Report 90/14, Department of Computer Science, State University of New York at Stony Brook (SUNY), June 1990.

Ee-Peng Lim, Zehua Liu, and Dion Hoe-Lian Goh. **"**A Flexible Classification Scheme for Metadata Resources", in Proceedings of Digital Library – IT Opportunities and Challenges in the New Millennium, Beijing, China, July 8-12, 2002.

Ling Liu, Calton Pu, and Wei Han. "XWRAP: An XML-Enabled Wrapper Construction System for Web Information Sources", in Proc. Int'l Conf. Data Engineering, pages 611-621, 2000.

Robert M. Losee and Stephanie W. Haas, "Sublanguage Terms: Dictionaries, Usage, and Automatic Classification," in Journal of the American Society for Information Science, 46(7), 1995, pp. 519-529.

LTSC. Draft Standard for Learning Object Metadata, 2000. IEEE Standards Department. URL: http://ltsc.ieee.org/doc/wg12/LOM_WD6-1_1.doc

Eduardo Mena, Arantza Illarramendi, Vipul Kashyap, Amit Sheth. "OBSERVER: An Approach for Query Processing in Global Information Systems based on Interoperation across Pre-existing Ontologies", in Conference on Cooperative Information Systems. pages 14-25, 1996.

George Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine Miller. Introduction to WordNet: An On-line Lexical Database. August 1993.

Jussi Myllymaki. "Effective Web Data Extraction with Standard XML Technologies", in World Wide Web, pages 689-696, 2001.

Alessandro Oltramari, Aldo Gangemi, Nicola Guarino, Claudio Masolo. Restructuring WordNet's Top-Level: The OntoClean approach, 2002.

H. Sofia Pinto, Asuncion Gomez-Perez, Joao P. Martins. Some Issues on Ontology Integration, 1999.

Fabrizio Sebastiani, "Machine learning in automated text categorization", in ACM Computing Surveys, 34 (1), March 2002.

Arnaud Sahuget and Fabien Azavant. "Building Lightweight Wrappers for Legacy Web Data-Sources Using W4F." Proc. Int'l Conf. Very Large Data Bases, pages 738-741, 1999.

Amit Sheth, "Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics", in Interoperating Geographic Information Systems, M. Goodchild, M. Egenhofer, R. Fegeas, and C. Kottman, Eds., Kluwer Publishers, 1998.

Amit Sheth and Wolfgang Klas, Eds., Multimedia Data Management: Using Metadata to Integrate and Apply Digital Data, McGraw Hill, 1998.

Amit Sheth, Clemens Bertram, David Avant, Brian Hammond, Krzysztof Kochut, and Yash Warke, "Semantic Content Management for Enterprises and the Web," *IEEE Internet Computing*, July/August 2002.

Amit Sheth, I. Budak Arpinar, and Vipul Kashyap, "Relationships at the Heart of Semantic Web: Modeling, Discovering, and Exploiting Complex Semantic Relationships," Enhancing the Power of the Internet Studies in Fuzziness and Soft Computing, M. Nikravesh, B. Azvin, R. Yager and L. Zadeh, Springer-Verlag, 2003 (in print).

Amit Sheth and James Larson, "Federated database systems for managing distributed, heterogeneous, and autonomous databases," ACM Computing Surveys, 22 (3), September 1990, pp. 183-236.

Michael Sintek and Stefan Decker. "TRIPLE – A Query, Inference, and Transformation Language for the Semantic Web", International Semantic Web Conference, Sardinia, June 2002.

Ronald Snijder. "Metadata Standards and Information Analysis: A Survey of Current Metadata Standards and the Underlying Models", Electronic resource, available at http://www.geocities.com/ronaldsnijder/, 2001.

York Sure, Juergen Angele, Steffen Staab. “OntoEdit: Guiding Ontology Development by Methodology and Inferencing”, Proceedings of the International Conference on Ontologies, Databases and Applications of Semantics ODBASE 2002.

Holger Wache, Thomas Vögele, Ubbo Visser, Heiner Stuckenschmidt, Gerhard Schuster, Holger Neumann, Sebastian Hübner. “Ontology-Based Integration of Information A Survey of Existing Approaches”, in IJCAI-01 Workshop: Ontologies and Information Sharing, H. Stuckenschmidt, ed., pages 108-117, 2001.

W3C. Resource Description Framework (RDF) Model and Syntax Specification, 1999. URL: http://www.w3.org/TR/REC-rdf-syntax/

W3C: RDF Vocabulary Description Language 1.0: RDF Schema, 2003. URL: http://www.w3.org/TR/rdf-schema/