

# Semantic Content Management for Enterprises and the Web

Amit Sheth\*, Clemens Bertram, David Avant, Brian Hammond, Krzysztof Kochut\*, Yashodhan Warke  
Voquette, Inc.

{amits, clemensb, davida, brianh, kochut, yashw}@voquette.com

\*Also LSDIS Lab, Computer Science, University of Georgia, amit@cs.uga.edu

## 1 Introduction

The Semantic Web [Be99], some researchers hope, might have an even bigger impact than what the WWW has achieved. This requires that data or content—whether Web pages or anything exchanged and displayed on Intranets and the Internet—be “semantically” annotated so that the meaning of data is expressed such that programs can *understand* it [Be99, FM01, BHO01]. The primary benefit of the vision of the Semantic Web is that it juxtaposes semantics and the Web. Semantics, with meaning and use of data, brings information closer to human thinking and decision-making. Together, these force us to simultaneously deal with the complexity of modeling, reasoning and perceptions to support semantics, with the huge scale and heterogeneity of all imaginable kind needed to deal with the Web.

Researchers in diverse areas have studied semantics for a long time. We have seen a steady progress from syntax, to representation and structure, and to semantics [S98], in the ways we approach and solve the challenges of finding, integrating and using information of diverse types and from diverse sources. Businesses have noticed the importance of semantics, too, in several ways. This has involved, among other things, development of taxonomies or ontologies and metadata standards of interest to an enterprise or industry, organization of content according to such taxonomies, annotation of content with metadata — especially contextually relevant or domain specific metadata, analysis of content for patterns or its mining to identify relationships between data from different sources, etc. Applications have ranged from improving search and personalization, organizing content for enterprise and industry portals, and improving syndication of content.

The key to a semantic approach and technology is agreement among humans, embodied in terms of ontological commitments and knowledge sharing through shared used of ontologies. Achieving such agreements amidst a very broad, pan-Web scale is difficult and expensive, which is why there are few very large ontologies. Developing successful business models for activities that span the entire Web is also difficult. During the recent upheaval in the Internet and digital content market, companies have increasingly shifted their focus to serving medium and large enterprises, rather than consumers and world wide audiences, as noticed by the decimation of B2C and large declines of B2B businesses, but relatively stronger showing of enterprise software players. Thus, for both technological and business reasons, we find that businesses developing solutions based on semantic or Semantic Web technologies focus on enterprise software markets. These technologies are also extending or finding specific applications in what is also considered to be Content Management and Knowledge Management markets, both of which are currently a few hundreds of millions of dollars large. Over 25 companies claim to offer semantic technologies or products and services enabling the Semantic Web (see: <http://business.semanticweb.org>).

In this article, we describe the *Semantic Content Organization and Retrieval Engine* (SCORE) technology in depth, and use it as the basis of describing some of the key components in building Semantic Web solutions. This is also an example of technology, which originated from academia, in this case the Large Scale Distributed Information Systems Lab (LSDIS) at the University of Georgia, and was licensed to start a company, *Taalee*, Inc. Taalee was later acquired by Voquette, Inc. [V], which now provides commercial products and services based on the SCORE technology. The focus of this paper is on the patented SCORE technology [SAB01] and what it has to offer now and in the near term. Broad-based commercial adoption and

deployment of some of the -current research performed in the rapidly growing Semantic Web community will require (a) further maturing of current proposals and more universal acceptance of a standard (e.g., RDF(S), DAML+OIL) for annotating Web pages and other content, (b) progress in technologies for creation, maintenance and sharing of ontologies, and (c) improved performance and scalability of inferencing techniques.

Four core capabilities that the semantic technology discussed in this paper addresses, but that are not widely seen in today's technologies are discussed next.

**Semantic Search and Personalization:** Consider the current generation of search engines. It is not possible to convey to the search engine whether the word “Palm” that you are interested in is the name of a company (i.e., query “Company: Palm”), name of a technology (i.e., “Technology: Palm” as in OS), or the name of a product (i.e., “Product: Palm” as in PDA). Today's search engines do not typically know the exact context of searched entities. For example, it is difficult to decide whether terms like Palm are discussed in a business context or a technology context. Some of them may be able to limit the search to “Palm” in the category of technology, but they do not know whether the sub-context is Palm OS or Palm PDA (without the explicit mention of the words “OS” and “PDA” in the document). A more difficult scenario would be to answer a query on movies that are directed by Robert Redford, but not in which he acted with another director. The difficulty arises due to the fact that keywords “director” and “Robert Redford” could be found even in documents in which Robert Redford has starred in a movie directed by another director. Automatic classification can partially address this problem if it can classify documents into the categories “Business” and “Technology”. Such techniques are, however, impractical to automatically subcategorize Movies content based on direction and acting.

**Semantic Metadata:** Besides ontologies, which provide classification and the terminological basis for contextual reasoning, the use of metadata is another key component in a semantic technology. Broadly speaking, we can divide metadata into two types – syntactic metadata and semantic metadata. Syntactic metadata are metadata that describe non-contextual information about content, e.g. language, length, date, audio bit-rate, format, etc. Such metadata offer no insight ‘about the content’. Semantic metadata, on the other hand, describe domain-specific information about content (in the right context). For example, if the content is from the Business domain, the relevant semantic metadata could be company name, ticker symbol, industry, sector, executives, etc., whereas if the content is from the Baseball domain, the relevant semantic metadata could be player, team, league, coach, game, score, etc. Metadata elements that offer more insight ‘about the document’ fall under the semantic metadata category. Ontology can provide the context for semantic metadata.

**Semantic Normalization:** Normalization plays an important role in dealing with semantic heterogeneity associated with multiple sources of content. We discuss two aspects of normalization. The first relates to associating (as much as possible) the same metadata for content belonging to the same domain or category regardless of source and format. For example, consider an article in a NewsML feed from Bloomberg and a PDF article posted at the CBS MarketWatch web site. If both articles contain an analyst's equity research report, then same type of metadata (such as the name of the company being reported on, the name of the analyst, the stock exchange that the stock is traded on, and so on) need to be associated for both. The second aspect of normalization of metadata refers to homogenizing the multiple names of a single entity into one uniform (canonical) name. For example, Yahoo's founder, David Filo, is referred to as ‘Chief Yahoo’ within the company, but as ‘Yahoo Founder’ in literature outside of Yahoo. Both these names refer to the same person (entity), but if a keyword search is made on either of these, results pertaining to the other will not show up. A strong support for mapping techniques is needed to support normalization.

**Semantic Association:** Consider an application to support a Financial Advisor, who analyzes stocks for his clients. Suppose this analyst were evaluating Intel Corporation (ticker: INTC), and that the application displayed a stock screen for INTC showing all items such as Company News, Market Commentary, Analyst Reports, Earnings News, etc. Here, semantic technology may be able to infer that a report that was recently released on the Semiconductor sector is of interest to our equity analyst because INTC is an important stock in the Semiconductor sector. Hence, the stock screen may then include this report, providing the Financial Advisor with information he did not specifically ask for, but that he needs to know. Note that there are two possible variations in technical solutions of this type — one is to determine with some probability that the report on the Semiconductor sector may have some relevancy to the *keyword* Intel, another is to know with certainty that the report is about the Semiconductor *sector*, and is associated with Intel, the *company* because that company is part of the Semiconductor sector. The former may be possible with some statistical and learning methods, while the latter would require some sort of ontology or domain model and a specific knowledgebase involving concepts of industry sectors and companies, and a relationship between these two concepts. In a similar example, consider an intelligence analyst researching Bin Laden. Semantic association between Bin Laden and Al Qaeda (with the relationship that the former is the leader of the latter) and similar relationships between Mohammed Atta and Al Qaeda should allow the semantic technology to provide the analyst with the ability to quickly associate content about Bin Laden and Mohammed Atta.

The above discussion identifies some of the key capabilities a comprehensive semantic technology may have. The following example brings these capabilities together. Consider a search on Tiger Woods. The traditional approach is to look for pages containing “Tiger Woods”, “Tiger” AND “Woods”, “Tiger” OR “Woods” (or even “Wood”), and other ways a keyword search is done using information retrieval techniques (word collocations, frequencies, etc.). One search engine does a particularly good (albeit self-fulfilling or perpetuating) job for some types of search needs, by considering trustworthiness, importance or popularity of a Web site or source. It would do a particularly good job of leading the user to an official site, a home page and fan pages for “Tiger Woods,” whatever that “Tiger Woods” is. Some marry directories or categorization with search, so that the search for “Tiger Woods” would be limited to a particular category, possibly segregating results better. To us, the real semantic solution starts by identifying that “Tiger Woods” is not a set of closely occurring words or a phrase, but that it identifies a Person with that name (perhaps Class modeling a Person, with Name as its property). Semantic technique can further reveal that it is likely either a Golf Player or a Spokesperson (who advertises for some companies and some products). If a content involving Person Tiger Woods is in Golf context, we would try to know about such relevant information as the tournament name, the golf course, etc. mentioned in or inferable from that content. If the content involving Tiger Woods was in the Advertisement context, we would try to know about the Companies or Products represented, etc. Semantics then further helps to establish which of the two context is of more interest to the users, and once that is established, easily help the user explore or search relevant relationships (e.g., in the context of Golf, the tournament this player played, the locations and golf courses those tournaments are held, the sponsors of a given tournament, players this golfer played with, or in the context of Advertisement, the companies whose products this Person represents, the agent of this Person, the advertisement agencies used, example advertisements, etc.). It would also be possible to easily associate or access an individual content item through those relationships.

After presenting the system architecture for SCORE, we discuss some of its key components. We emphasize new technical contributions and achievements within each of the components. The article ends with a discussion on some of the more advanced semantic applications, for which SCORE-based products are commercially deployed today.

## 2 System Architecture Overview

Ontologies, or their substitutes- taxonomies that involve hierarchical arrangement of categories, play a central part in most semantic technologies. Ontologies also become the basis for syntactic and (more importantly)

semantic metadata, which can be used for annotating or tagging the content. Knowing a context of content can help in identifying, which semantic metadata to extract or assign. An automatic classification technology can help in determining a context. A document can be classified into one or more categories, and semantic metadata corresponding to one or more contexts can be extracted or created.

We divide ontology into two related components – the WorldModel, which can be seen as the definitional component, and the Knowledgebase, which can be seen as the assertional component. As with the specification of ontologies, the definition of the WorldModel and Knowledgebase involves domain experts and understanding of eventual application requirements and cannot as well as should not be automated.

The Knowledgebase reflects that subset of the real world, for which we are creating semantic applications, and is an important part of the solution. It allows for providing value-added semantic metadata (e.g., since a particular document talks about the terrorist “Bin Laden”, the corresponding organization is “Al Qaeda” even if it is not explicitly mentioned in that document), or making semantic associations (i.e., instantiating relationships, such as between industry and company).

Syntax and structure of content (used interchangeably with data and documents), such as a Web site’s page templates or an XML feed (more specifically, the associated DTDs), provide an important basis for obtaining valuable metadata. The metadata or information extraction technique is another important component of a semantic technology. Finally, there is a need to have a query processing system, with appropriate inferencing or reasoning capability, and preferably a comprehensive suite of APIs that not only supports semantics-enabled search and other traditional applications (such as directory, personalization, syndication, and so on), but also enables rapid building of highly customized enterprise applications.

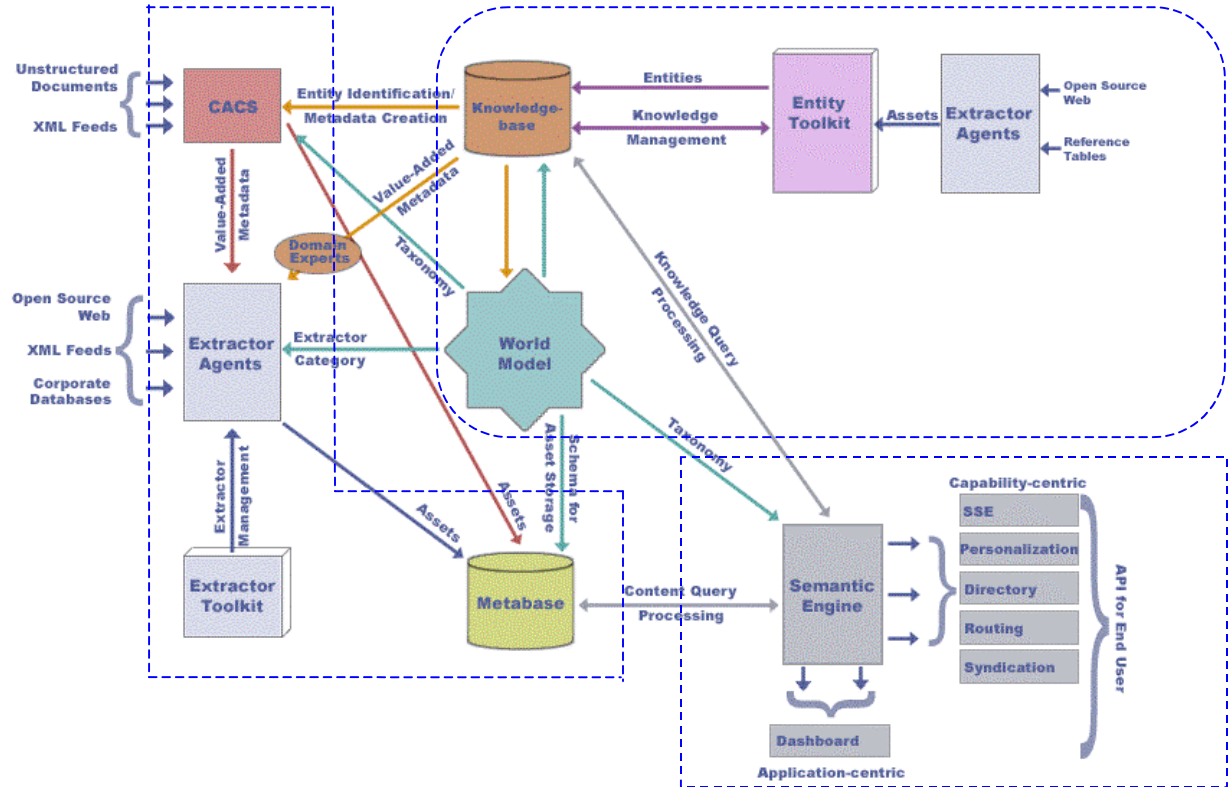


Figure 1: SCORE System Architecture

Operation of SCORE technology involves a process consisting of three independent activities shown by dashed areas in Figure 1. They cooperate through knowledge and metadata sharing. All internal data exchange in SCORE is in XML. While RDF is not currently used, the functionality is similar and a move to RDF would technically be a straightforward effort.

The first activity shown in the top right part of Figure 1 involves the definition of the WorldModel and Knowledgebase using graphical toolkits, creation and execution of knowledge extraction agents to manage the Knowledgebase based on trusted knowledge sources. Different parts of the Knowledgebase can be populated from different trusted knowledge source. The second activity shown on the left part of Figure 1 is that of content processing, including classification and extraction of metadata out of content. This activity results in creating metadata that is organized according to the WorldModel definition, and stored in a database (referred to as Metabase). Notice that SCORE utilizes two types of software agents – knowledge extractors for Knowledgebase creation and maintenance, and content extractors for metadata extraction. Each knowledge or content source could be heterogeneous (e.g., an XML or other metadata standard based feed, Web site, database, or documents in various formats), internal or external to the enterprise, and accessible in push (as in content feeds or database exports) or pull (as in Web site or document access) modes. Given the exploding size of the available content, it is important that all components and processes of a scalable semantic technology (including Knowledgebase creation and maintenance, classification, and metadata extraction) be as much automated as possible, as is the case with SCORE.

The final activity, as shown on the bottom right part of Figure 1, is that of supporting semantic applications. Semantic Engine is a highly scalable, very high-performance query processor. It does not support inferencing mechanisms similar to those found in some AI or logic-based systems; rather, it provides limited inferencing based on traversal of relationships in the ontologies. It provides APIs for rapidly building traditional and customized applications. Building GUIs for such applications is easy as all results are in XML with well-defined DTDs.

### 3 Semantic Metadata Extraction

Crawling and information extraction technologies are encountered today in wide varieties (See Appendix/Sidebar: 9.1 Crawler and Extraction Technologies). The advantage of SCORE's approach to metadata extraction as compared to other approaches is the combination of six key capabilities:

1. It can exploit information from structured (databases, XML feeds) and semi-structured (HTML) content as well as identify metadata in unstructured sources (continuous text).
2. It can identify both domain-specific and domain-independent metadata
3. It can enhance the extracted information using the Knowledgebase
4. The Knowledgebase itself can be continuously updated using the same extraction technology, thus avoiding the problem of static, obsolete dictionaries.
5. Extractor agents are rapidly created and maintained by non-programmers using a graphical toolkit
6. Extractor agents are deployed in a distributed multi-agent environment, require only a JVM-enabled machine, and execute on demand (for pushed content) or are scheduled to run independently (for pulled content)

At the heart of the system lies the WorldModel (Figure 1, center-middle), the driving component of context-sensitive metadata extraction. The WorldModel contains a hierarchy of categories or domains that each possess a set of (inheritable) attributes. The underlying idea is that documents belonging to different domains have a different set of interesting, "domain-specific" properties, while another set of properties is common to documents belonging to other domains. For instance, in a business news story, items of interest (that could be indexed) are company names, ticker symbols, industries and analysts, whereas in a baseball report, items to index could be team and player names, leagues and locations. All stories, however, will have a source, creation date, title, description, and other domain-independent properties. Those generic attributes are

associated with a top-level category and are inherited by all other categories. The collection of all metadata about one piece of content is referred to as an Asset.

The extraction process is driven by the WorldModel and is governed by a set of extraction and enhancement rules. Using the Extractor Toolkit – a Java-based GUI – one defines Extractor Agents for a particular source of information, e.g., a NewsML feed or a Web site. After deciding on one of the WorldModel categories, the agent creator specifies where in the source to find values for the attributes of that category – for each attribute one or more regular expression-based rules. Thus, a number of attributes can be populated by applying the rules to the source text by exploiting the structure of the document. Obviously, the more structured the source, the better the extraction result. Only in few cases, though, will it be possible to populate *all* attributes just by examining the structure. For this purpose, the Categorization and Auto-Cataloging System (CACS, see also Chapter 5) is used to find any relevant proper nouns within the raw text.

CACS makes heavy use of the Knowledgebase (see Figure 1, center-top). The Knowledgebase consists of Entities and Relationships. Entities are classified according to a hierarchical Entity-class tree; one Entity can belong to multiple Entity classes. For example, one branch of the class tree can contain persons with subclasses politician, artist, sportsPerson; sportsPerson might be further subclassified into coach, athlete, and so on. The level of detail depends solely on the requirements of the application. The Entity “Jesse Ventura” could belong to the Entity classes “politician” and “athlete”. One could also say, that he “plays two roles”. Every Entity can have aliases or synonyms that help the system deal with various spellings of a name (“Taleban” vs. “Taliban”) or even nicknames (“The Rock”).

The Knowledgebase also defines Relationships between Entity classes that the Entities can participate in, where Relationships defined for a superclass are inherited by its subclasses. In our example, it could contain the Relationship instances “Jesse Ventura holdsOfficeOf Governor”, an instance of “politician holdsOfficeOf politicalOffice”, and “David Letterman interviewed Jesse Ventura”, an instance of “person interviews person”. The design and number of Relationships is, once again, dependent on the deployment environment.

CACS has the ability to recognize Entities in continuous text by their canonical form, nickname, or common alias (“Bill” vs. “William”). A baseball Extractor agent could ask CACS to populate the attribute “players” by scanning an article for baseball player Entities. Relationships in the Knowledgebase can be used to resolve ambiguities between competing “candidate” Entities found in the source text. For instance, the occurrence of the word “Gateway” in the source text of a business report may lead to several candidate company Entities (companies that have the word “Gateway” in their names). However, if the story were to mention a company executive, then the “CompanyExecutive worksFor Company” Relationship would be used to eliminate the false Entity matches.

Enhancement Rules use Relationships to populate other attributes with values that cannot be found in the text. For instance, a baseball player’s name is sufficient to derive both the “team” and – using the inferred value for the “team” attribute – the “league”. Similarly, a ticker symbol leads to “company name”, “industry”, and “sector”.

In certain cases, it is possible to use the classification result that CACS provides to populate further attributes. For example, “Equity” documents may be classified into the topics “analysis”, “IPO”, “earnings”, “market commentary”, and “mergers”. Instead of introducing a number of subcategories for the “Equity” category, an attribute “topic” could be added as a means of distinguishing these types of documents; this attribute would then get its value from CACS’s classification output.

## 4 Automatic Classification and Metadata Extraction

### 4.1 The link between Classification and Metadata Extraction

Automatic classification is a well-researched field. Here, we will mainly explore a strategy of combining multiple classification techniques as well as the close tie between classification and metadata extraction. The best way to show this connection is via a simple example:

Example 1:

*A lot of VC's have decided that now is a good time to invest in the technology sector, sensing that the turning point is just around the next corner. **John Smith** recently invested in **Voquette, Inc** in anticipation of a large return on a modest investment.*

Example 2:

*Many sports commentators are convinced that the **World Series** hinges on a single man this year. **John Smith** has had an incredible number of home runs this year and is being heralded the "**Babe Ruth of BeanTown**".*

Our classification system might deduce that example 1 is a Business text and example 2 is a Baseball text. The extracted Entities are shown in red.

An Entity named "John Smith" is found in each example; however, it is unlikely that the two Entities are really the same person. As in many metadata extraction examples, there is an ambiguity. Even if the two texts referred to the same person, John Smith, the attributes (hence the metadata) and the Relationships (hence semantic associations) for them would be different due to difference in contexts.

### 4.2 Resolving Metadata Extraction Ambiguities

There are many strategies to resolve ambiguities. Consider the following methods:

- Use all extracted Entities and use knowledge about their Relationships to each other to try to resolve the ambiguity.
- Associate each Entity with a category of an arbitrary classification scheme and use the classification result of the text to provide clues to help resolve ambiguity.

Each method has its own strengths and weaknesses. Relying on the Relationships between Entities is a powerful method, but what if **all** of the discovered Entities are ambiguous? Rather than using a simple technique such as starting with unambiguous Entities and iteratively disambiguating the others, a more computationally expensive method must be used, such as constructing a flow network with the interconnections between all of the extracted Entities and choosing that network which maximizes flow. There are also bound to be cases where all of the extracted Entities have no prior known (direct) Relationships, which further complicates the matter.

Assigning each Entity to a category in our classification scheme is a direct and easily expandable method, but relies on the correct assignment of Entities to one or more categories and is then dependent upon correct text classification. There may also be cases where the ambiguous Entity has matches, which **all** fall within the predicted category.

Instead of relying on either of these methods, a better strategy is a combination of both. The classification can be used to decrease or eliminate some ambiguities, and the Relationships can then be used to further disambiguate remaining ambiguous Entities. This merging of techniques also provides several additional benefits:

- Extracted Entities can be used to help to correctly classify a document.
- Classification allows the use of domain specific NLP techniques to discover new Relationships between Entities.

### 4.3 Classification by Committee

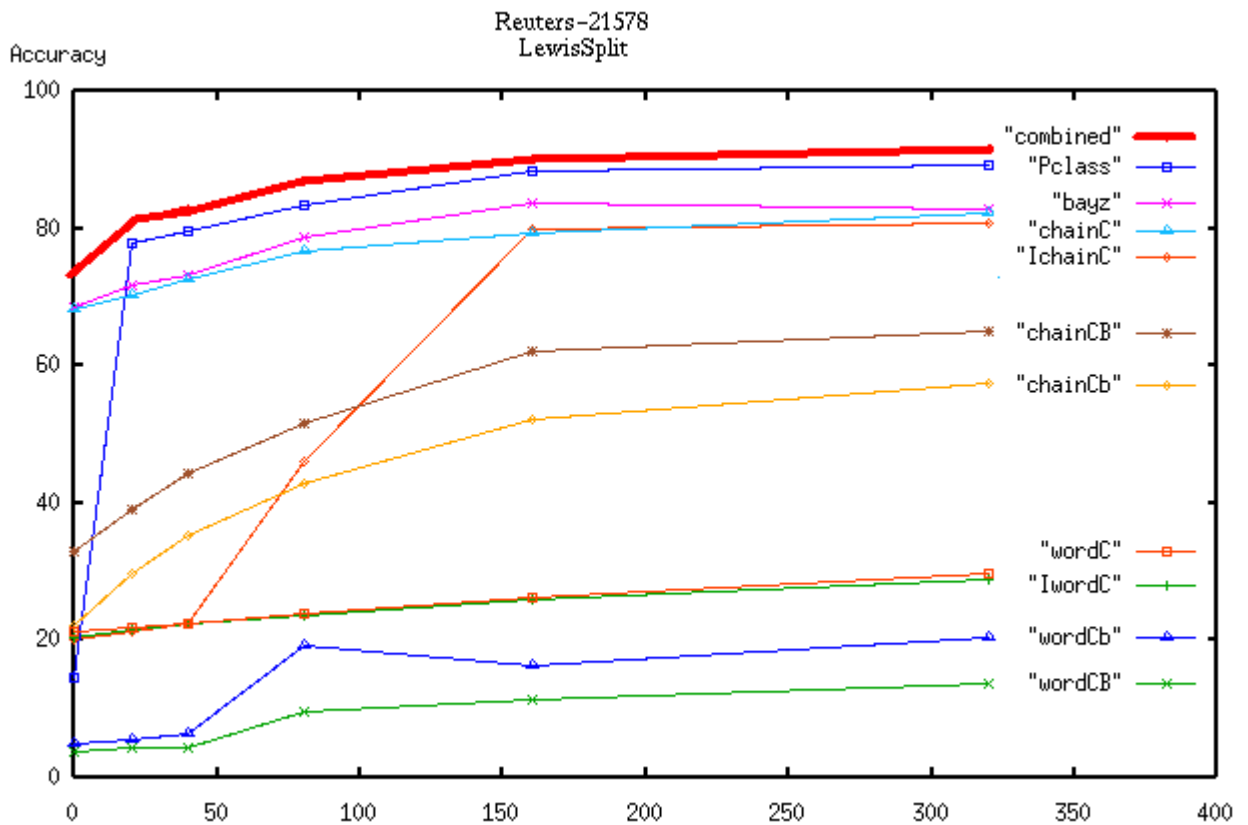
Numerous classification methods have been researched, designed, and implemented by a wide variety of developers in both the academic and corporate realms [LC, LJ, LT] (See Appendix/Sidebar 9.2 : Automatic Classification). There is much debate over which method or combination of methods work best. Rather than taking a "one size fits all" approach it is possible to combine disparate methods into a *Classification Committee*.

The basic idea behind a Classification Committee is to exploit synergy between classification technologies such that the accuracy of the whole is greater than the accuracy of the individual classifiers that make up the committee. This approach works best when the individual classifiers use very disparate techniques.

SCORE uses a committee of various approaches including probabilistic (Bayesian), learning (Hidden Markov Models (HMM)), as well as knowledge-based techniques. Both Entity recognition and use of domain phrases fall under the last category; the former uses Entity classes found in the text to derive the classification result, the latter relies on either handcrafted or otherwise derived phrases that are significant for the various categories, for instance "birdie, putting, tee" for Golf or "loss per share" for Earnings.

Since the various classifiers do not usually have the same accuracy a weight is attached to each classifier, which is used to scale the results before combining them. These weights are determined at training time according to a variant of Larkey's and Croft's "Weighted Linear Combination" approach [LC].

Below are some results based on the Reuters-21578 text categorization test collection:



This test was performed using a different threshold for the minimum number of documents per category in the training sets under the LEWISSPLIT, which produced the category counts at various thresholds as shown in Table 1.

This was done to illustrate the effect of the number of training documents on the accuracy of various classifiers and clearly shows that the Classification Committee consistently outperforms the individual classifiers at every resolution.

For completeness, Table 2 shows a brief explanation of each classifier in the committee.

Threshold	Category Count	Name	Feature	Type
0+	115	wordCb	word	Bayesian
20+	44	wordCB	word	Bayesian
40+	31	wordC	word	Bayesian
80+	18	IwordC	word	Bayesian (inverted)
160+	10	chainCb	chain	HMM
320+	7	chainCB	chain	HMM
		chainC	chain	HMM
		IchainC	chain	HMM (inverted)
		bayz	word	Bayesian
		Pclass	domain terms	Knowledge Based

**Table 1: Category Count**

**Table 2: Classification Committee**

## 5 Semantic Search Engine

Extracted and enhanced metadata that are stored according to the above mentioned WorldModel can be used with great efficiency and can yield extremely high quality search results because they provide the basis for “search in context”. Many irrelevant search results that are typical of common search engines stem from lack of context for ambiguous words. Although great improvements have been made, the highest precision can be achieved by attribute search where the user specifies the category and one or more attribute values, similar to the “advanced search” of domain-specific sites like Amazon (book search by title, author, ISBN, etc.). Unlike Amazon, however, the SCORE supports category-specific attributes for all categories involved in the extensible WorldModel and for content not limited to that stored in a database stored with the Web server.

SCORE’s Semantic Engine (SSE) creates a main-memory index of the metadata, which has two important strong points:

- a) In addition to supporting phrase search and exclusion, it retains attribute information, so that the above described attribute search with all its advantages is possible.
- b) By virtue of accessing a main-memory index instead of a database or index stored on a file system query processing is extremely fast (see also the chapter on performance and scalability), on the order of hundreds of nanoseconds per request.

The current implementation of the SSE enables incremental index updates. That means, newly found Assets can be indexed and made searchable within less than one minute, a typical requirement for near-realtime environments. Queries are posed either against the Metabase or the Knowledgebase. Traversal of relationships in the Knowledgebase leads to a limited form of inferencing.

SSE provides a HTTP API that is used as the basis for all Semantic Applications (see Chapter 7 on Development of Semantic Applications). All query results are in XML, and allows for easy creation of content or knowledge browsing or searching applications, as well as more customized applications.

## 6 Performance, Extensibility and other operational issues

The SCORE technology is meant to provide very high performance, be highly scalable and very robust. We present some of the key data to demonstrate that it can meet the most demanding requirements of an enterprise and even pan web applications.

*Performance Data* (based on a dual Pentium III 766MHz processor, 2 GB RAM, running RedHat Linux with Apache as web server and MySQL as light-weight database):

Queries per server / hour	1,000,000
Query Response Time (light load)	<1ms to 10ms
Query Response Time (heavy load)	10ms to 300ms
Semantic Associations created / hour	10,000
Main-Memory Index Update Frequency (incremental indexing)	1 minute (near real-time)
Extracted Assets / second / extractor (processing time)	1-3

### *Scalability:*

The main-memory index holds metadata of about 4.5 Million documents per server in the above-mentioned configuration. If more data needs to be stored, the index can be seamlessly distributed over any number of servers.

Minimizing human involvement by automating most of the work is a key factor in scaling up extraction and maintenance of the Knowledgebase. Three full-time Extractor writers can write and maintain a few hundred Web-based Extractors, assuming that the extracted sources change no more than a few times per year. If, at some point in the future, content were made available as XML (e.g., RDF), this number would increase manifold as only the final presentation of the data will change but not the underlying data format.

An up-to-date Knowledgebase is crucial to many enterprises. SCORE allows Extractor agents to be scheduled to regularly check for new or expired information and require very little manual effort. Extractors can run on different workstations and require only a Java Virtual Machine to run. Any number of extractors can run concurrently as the execution environment supports a distributed agent infrastructure.

### *Extensibility:*

The Classification Committee that is used by CACS can be extended by existing classifiers with little programming effort. The XML and HTTP based infrastructure easily supports various interfaces and API based interoperability with other components and systems.

Robustness is critical as critical enterprise operations depend on such continuous availability of score. There are many aspects of creating a robust software, which we do discuss here due brevity. As a historical information, a data center hosted SSE was continually operational without any down-time for over 400 days until we needed to make version upgrade.

## 7 Development of Semantic Applications

Two main APIs are available to create front-end applications: the Semantic Search Engine and the Knowledgebase API. Web application creators will usually follow the steps below:

1. create a user interface for the input
2. generate KB and MB queries (API calls) from the input
3. run the queries and assemble the (XML) output
4. use XSLT to present the results to the user

In this section, we briefly present three SCORE powered applications. They show how to use pure Metabase search as well as assembling multiple queries against Knowledgebase and Metabase and presenting them to the end-user through XSLT.

Choose a category to search in:

View a quick tutorial of Taalee Semantic Search. You'll need the free **Macromedia Flash 4 plug-in**.

**Taalee Semantic Search™** empowers Internet users to explore digital media content on the Web. **Taalee** incorporates a versatile search technology. This customizable interface blends an easy-to-use structured query, and yields precise and timely results. For more information, visit the **Taalee** website.

**YOUR SEARCH RESULTS**

Your search results will be divided into levels of precision, starting with the most likely. Some suggestions about items found in other categories may be included. There will be icons telling you if the media is audio (🔊) or video (📺). Clicking an underlined title will take you to a Rich Media Reference™ page with metadata about the media. Here you will find all the information that accompanies the media or, in some cases, other information that we have found. Clicking the buttons will launch the appropriate player and play the media. There may be more than one button for each media type, because they will play different bandwidth streams. We may not be able to label the button with the bandwidth, because the source site may not provide it. You must have the appropriate plugins installed in your browser to play the audio or video.

**REQUIREMENTS**

You must have the appropriate plugins installed in your browser to play the audio or video. A browser above version 4.0 is recommended.

**Terms of Use:** Taalee search pages are available for your personal use only. You are forbidden from reformatting the contents and search results and displaying them elsewhere without Taalee's explicit permission. No one may use automated queries, intelligent agents, software robots, or any other automated systems to extract information from the Taalee databases. Taalee Semantic Engine, Taalee Semantic Search, Rich Media Reference, and Taalee Semantic Asset Manager are trademarks of Taalee, Inc. All contents Copyright 2000 by Taalee, Inc. All rights reserved.

Search results in **All** for oracle

[Query time: 0.049 seconds](#)

oracle found in: [Business](#), [Tech Product](#), [Music Video](#), [Movie](#)

[Query time: 0.049 seconds](#)  
Page Download Time: 0.011 seconds

**BACK**

### 7.1 Semantic Search (pure MB search)

SCORE's Semantic Search is a service that enables the end user to search for relevant content in a simple and user-friendly manner. The search is provided through customized Search APIs that enables the user to search, either contextually (search in the domain of choice as a specific attribute value) or as a 'dumb' (traditional) keyword search (dumb search box allowing user to enter his keyword non-contextually, with the system returning results categorized/aggregated by the domains in which any matches occur). The user can choose to search from any number of custom domains. Search results for an "ALL" search (dumb keyword search) on the keyword results in the display of the various domains in which the keyword appeared in some context.

The results are classified into three relevancy buckets:

*Exact Matches* – exact context (AND),  
*Close Matches* – part of the query satisfied (OR),  
*Related Matches* – dumb keyword search, no attribute context, and each result displays the associated metadata.

Search results in **Business** for gates , ellison

195 results found

[Query time: 0.004 seconds](#)

**Exact match for your search...**

1. [Microsoft Corp.](#)  
 Has Oracle's Larry Ellison Surpassed Bill Gates as the World's Richest Man? (Computer Software & Services, Development Tools,...)  
 Source : ON24 PostedDate : 04/28/2000  
 company : Microsoft Corporation symbol : MSFT

**Close match for your search...**

1. [Applied Materials Launches Innovative Nitridation...](#)  
 Applied Materials, Inc. (Nasdaq:AMAT) today introduces the DPN (Decoupled Plasma Nitridation) chamber, a critical process...  
 Source : BusinessWire PostedDate : 11/28/2001  
 company : Applied Materials, Inc.; Thermal Systems, Inc. symbol : AMAT; THSI

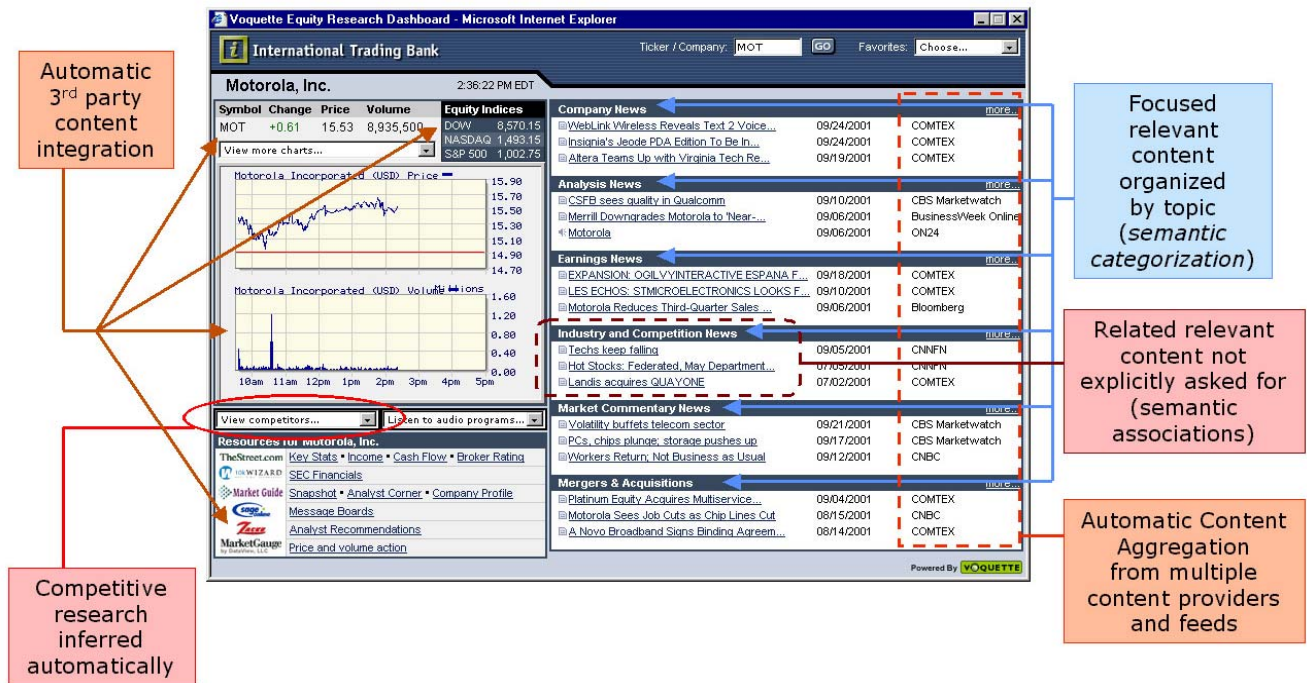
2. [Synopsys Physical Synthesis Integrated Into NEC's...](#)  
 Synopsys, Inc. (Nasdaq:SNPS), today announced that its Chip Architect and Physical Compiler(TM) products have been integrated...  
 Source : BusinessWire PostedDate : 11/27/2001  
 company : Synopsys, Inc.; NEC Corp. symbol : SNPS; NIPNY

3. [AOL discusses online shopping trends](#)  
 AOL E-Commerce Patrick Gates speaks with Allen Wan about online shopping trends on Black Friday, the biggest shopping day of...  
 Source : CBS PostedDate : 11/23/2001  
 company : America Online, Inc.

4. [Atmel Introduces New Embedded Memory Blocks to Support...](#)  
 Atmel(R) Corporation (Nasdaq:ATML) announced today a complete set of new 0.35 micron and 0.25 micron matrices to convert most...  
 Source : BusinessWire PostedDate : 11/21/2001  
 company : Atmel Corp.; Xilinx, Inc.; Altera symbol : ATML; XLNX; ALTR Corp.

5. [New Floating Point Cores from Nallatech extend...](#)  
 Nallatech, the leading high-performance systems solution provider, has launched a fully IEEE-754 compliant single precision...  
 Source : BusinessWire PostedDate : 11/20/2001  
 company : Xilinx, Inc. symbol : XLNX

**BACK**

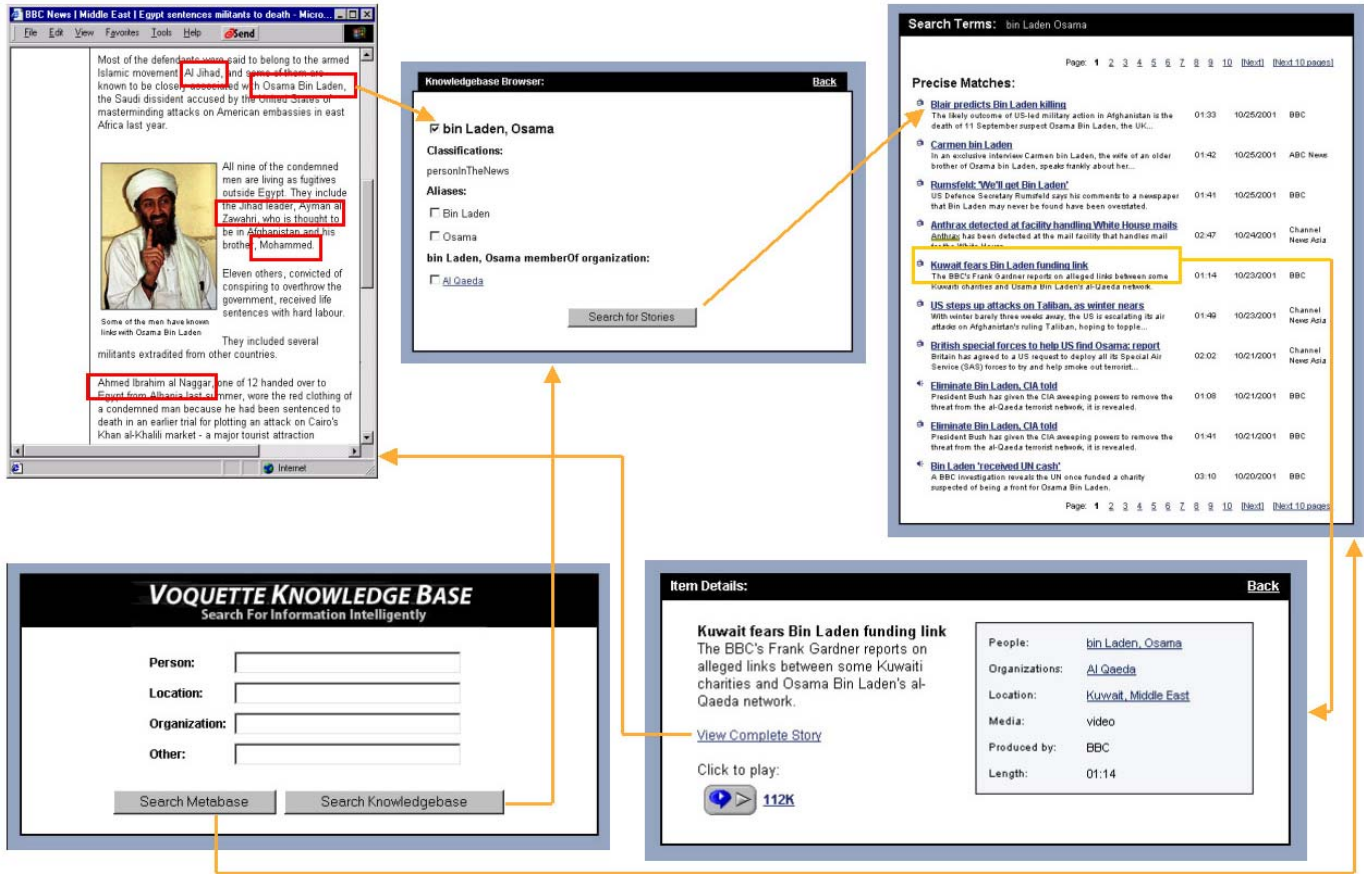


## 7.2 Analyst Workbench (Integration of KB and MB Search)

The Financial Dashboard is a knowledge-based, Semantic Web application, primarily for use by professional enterprise users like Financial Advisors, who feel the need to obtain a 'complete picture' view of their area of research, or a tool for detailed analysis of items of business interest (e.g., for Financial/Equity domain, company, industry, competition and so on). The Financial Dashboard displays content relevant to the user, duly categorized into meaningful categories in an automated fashion, with an ability to automatically use the knowledge base to fetch relevant content intelligently. It demonstrates our ability to aggregate and integrate content of all kinds (HTML, Audio, Video, etc.) from multiple sources (including third party content). In a nutshell, the Research Dashboard is a powerful application that enables the enterprise user to obtain all the possible information that is relevant to his needs without explicitly asking for all of it. It supports the following.

- Ability to *automatically* normalize the feed across multiple content sources (obtain, if possible, same metadata for content of the same category regardless of representation from different sources)
- Ability to aggregate and integrate content from multiple sources (including third party content and proprietary corporate content) and multiple forms of media (audio, video, HTML, PDF etc.) in one application
- Ability to demonstrate highly value-added semantic associations between financial Entities by means of an extensive growing financial knowledgebase (like - Cisco Systems competes with Nortel Networks, Juniper, 3COM etc.) to deliver relevant assets not specifically asked for, and for research applications (show report on Semiconductor sector downgrade when researching Intel Corporation)
- Content automatically categorized into multiple domains of great relevance (Market commentary to financial enterprise users)
- Ability to let the customer define the domains it would want to display the content by (If the customer wants a new domain called 'Technology News', we can incorporate that)
- Near real-time freshness of content

### 7.3 Integrated KB Browsing and Querying



The Knowledge Browser is a set of APIs that allow the user to co-relate the rich knowledge in SCORE’s system, and provide relevant content for the sake of performing intelligent analysis. The Knowledge Browser has the ability to pinpoint the exact classification of a given Entity (person, organization, etc.) identified in a document or that the user wishes to search for (in the desired context) and provides valuable Relationships of the searched Entity to other related Entities. The user can link and navigate between the Entities, thereby facilitating effective knowledge inferencing. Knowledge browsing makes use of the semantic associations in SCORE’s rich Knowledgebase.

In addition to knowledge navigation, the user can also view related content on any Entity of interest. Customized user-friendly GUIs can be created to display the metadata about the content, thus enabling the user to get a ‘complete picture’ of relevant content. Such relevant content can then be again scanned for Entities of interest and used for performing knowledge inferencing, thus completing the ‘information loop’ on knowledge browsing and content browsing. Examples of advanced applications of the above type include intelligence analysis and travel security applications.

It has taken a significant time for research to find its ways to commercial applications, in part due to performance, scalability and maintainability limitations of knowledge-based and AI technologies. SCORE is one of the new generation technologies that show success in dealing with these challenges and in applying semantic technology to real-world applications (See Appendix/Sidebar: 9.3 Commercial Offerings). It also shows the possibility of achieving grander visions such as that of the Semantic Web.

## 8 References

[Be99] Tim Berners-Lee (with Mark Fischetti), Weaving the Web, The original design and ultimate destiny of the World Wide Web, Harper, 1999.

[BHO01] Berners-Lee, T., Hendler, J. and Lassila, O., "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.

[FM01] D. Fensel and M. Musen, Eds. "The Semantic Web: A Brain for Humankind," IEEE Intelligent Systems, March/April 2001.

[S98] A. Sheth, "Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics", in Interoperating Geographic Information Systems. M. F. Goodchild, M. J. Egenhofer, R. Fegeas, and C. A. Kottman (eds.), Kluwer, Academic Publishers, 1998, pp. 5-30.

[SAB01] A. Sheth, D. Avant and C. Bertram, "System and Method for Creating Semantic Web and Its Applications in Browsing, Searching, Profiling, Personalization and Advertisement, US Patent # 6,311,194), October 30, 2001.

[V] Content Taalee/Voquette, [www.taalee.com](http://www.taalee.com) or [www.voquette.com](http://www.voquette.com)

[LC] Larkey, L.S. and Croft, W.B., "Combining classifiers in text categorization," Proceedings of SIGIR-96, 19-th ACM International Conference on Research and Development in Information Retrieval, 1996, pp.289-297.

[LJ] Li, Y.H. and Jain, A.K., "Classification of text documents," The Computer Journal 41, 8, 1998, pp. 537-546.

[LT] Liere, R. and Tadepelli, P., "Active learning with committees for text categorization," Proceedings of AAAI-97, 14-th Conf. of the American Association for Artificial Intelligence, 1997, pp. 591-596.

### About Authors

Dr. **Amit Sheth** founded Taalee, Inc in August 1999 and managed it as its CEO. Since its acquisition by Voquette inc. in June 2001 he has served as CTO and SrVP. He is the director of Large Scale Distributed Information Systems Lab at the University of Georgia and a Professor of Computer Science. He is widely recognized for his work in federated database systems, semantics heterogeneity and interoperability in distributed information systems, and workflow management. SCORE is the third major commercialization of his research.

**Clemens Bertram** is the Director of Engineering at Voquette, and has designed and supervised development of most of the SCORE components.

**David Avant** is a Senior Engineer and a key force behind the extraction technology.

**Brian Hammond** is a Senior Engineer and a key force behind the CACS and SSE.

Dr. **Krzysztof Kochut** is the Chief Architect who has conceived SSE and supervised as well as contributed key parts of CACS and SSE. He is also a Professor of Computer Science at the University of Georgia.

**Yashodhan Warke** is the Director of Product Development.



## 9 Appendices/Sidebars

### 9.1 Comparison chart of Crawler and Extraction Technologies

	Crawling range	Categorization	Extracted / indexed features	Other	Attribute Search
Conventional crawlers (Harvest, ...)	All, no restriction	None	Full text, title	--	Keywords
Advanced crawlers (Google, RetrievalWare, ...)	All, some restriction possible	None	Full text, some structured metadata (title, creationDate, ...)	--	Keywords, title, URL, creationDate
SCORE	Focused	Extensible Taxonomy	Domain-independent metadata (title, creationDate, ...), domain-specific metadata (according to WM)	Metadata enhancement by inference (using KB)	Full category/attribute search, keywords
Classification Software (Autonomy, Applied Semantics, Stratify)	None	Fixed / Extensible / inferred (clustering) Taxonomy	Full text	--	Keywords
SingingFish	Web	Flat, Small Taxonomy	Domain-independent metadata	A/V	Keywords, category

### 9.2 Automatic Classification

One of the most efficient ways of organizing information is classification of that information in one or more categories, possibly in a hierarchy. Doing so in an automated way enables end users to find the information they need easily and efficiently. Many different classification techniques are in use today that vary in the approach used to perform classification. A review of these techniques is provided below.

**Rules-based Techniques:** Rules-based techniques allow the user to precisely define the criteria by which documents are classified. Of course, this necessitates the use of well-trained people with domain expertise to write rules to ensure accuracy in the classification results.

**Learning Techniques:** Learning techniques involve the process of training the classification system using a set of reference documents, and subsequently using this trained system to classify new documents. In other words, the system intelligently “learns” how to classify new documents, based on observations it makes during training. Pattern matching is another method of classification by learning where the system identifies patterns in sample documents and makes predictions about unseen text.

**Clustering Techniques:** Clustering techniques automatically index and cluster similar concepts together. Such methods usually employ statistical and linguistic algorithms, to cluster together documents with similar

content. Products using clustering methods are more dependent on the underlying data for their categories and work in a prescriptive fashion as opposed to a descriptive fashion in the case of learning methods.

**Knowledge base Techniques:** Knowledge-base techniques rely on the use of rich knowledge bases to identify entities and domain relationships between entities for classification. Knowledge base techniques are a novel approach that does not depend on the abilities of the system for classification. Instead the use of high quality knowledge bases for look-up assures accurate classification results.

**Today's Classification Solutions Classified:**

*Rules-based:* Verity

*Learning:* Inxight, Mohomine, Sageware, Autonomy

*Clustering:* Cartia, Semio

**Classification Committee:**

SCORE provides the only commercial semantic solution that employs a state-of-the-art approach by combining multiple classification approaches together – classification committee – to improve classification results over individual classification approaches. SCORE uses a hybrid of probabilistic, learning and knowledgebase techniques to bring out the best of all the three approaches, resulting in more accurate classification.

Suggested additional reading:

D. Michie, D.J. Spiegelhalter, C.C. Taylor (eds), Machine Learning, Neural and Statistical Classification, IJIS Horwood, 1994. Now at: <http://www.amsta.leeds.ac.uk/~charles/statlog/>

K. Adams, "Word Wranglers: Automatic Classification Tools Transform Enterprise Documents from 'Bags of Words' to Knowledge Resources." Intelligent KM, January 2001.

<http://www.intelligentkm.com/feature/010101/feat1.shtml>

### 9.3 Commercial Offerings

[Note to the editors: This table looks at the capabilities of products from some of the companies listed at [business.semanticweb.org](http://business.semanticweb.org). If the table is included in the article, it will be updated again before the publication. If the table seems like commercial advertisement, it can be either removed entirely, or redone to remove the names of the specific companies, and present only the capabilities and general support for those capabilities in the commercial products.]

One way to compare capabilities of semantic technologies is to look at key capabilities of some of the products claiming to enable the Semantic Web. All companies and their URLs are listed at <http://business.semanticweb.org>.



Company	Voquette	Clearforest	Inxight	Applied Semantics	Autonomy
Automatic aggregation and integration of content from multiple sources	Yes	No	No	No	Yes
Ability to automatically do related inferencing	Yes	Limited	No	Limited	Limited
Ability to model content structure according to enterprise's view of the world	Yes	No	No	No	Yes
Automatic Creation and Maintenance of Knowledge Base for any domain	Yes	Limited	No	Limited (proprietary Knowledge Base)	Limited
Ability to automatically create relevant semantic metadata via classification	Yes	No	No	No	No
Tag Creation Based on explicit text	Yes	Yes	No	Yes	Yes
Tag Enhancements	Yes	No	No	Limited	Limited
Ability to automatically normalize content feed tags	Yes	No	No	Limited	No
Contextual Search (order of magnitude better than keyword-search)	Yes	No	Limited	Limited	Limited
Unstructured Content	Yes	Yes	Yes	Yes	Yes
Semi-Structured Content	Yes	Yes	No	No	Yes
Structured Content	Yes	Yes	No	No	No
Text Processing	Yes	Yes	Yes	Yes	Yes
Audio/Video Processing	Limited	No	No	No	Yes