TopQuadrant

helping enterprises envision, architect and plan e-Business solutions

TopQuadrant Technology Briefing

Semantic Technology

Version 1.2

March 2004

Table of Contents

Tables

Figures

1. Semantic Technology

1.1 What is Semantic Technology?

We define semantic technology as a software technology that allows the meaning of and associations between information to be known and processed at execution time. For a semantic technology to be truly at work within a system there must be a knowledge model of some part of the world that is used by one or more applications at execution time.

How is it distinguished from more conventional applications?

- Semantic technologies represent meaning through connectivity. The meaning of terms, or concepts, in the model is established by the way they connect to each other.
- A semantic model expresses multiple viewpoints.
- Semantic models represent knowledge about the world in which the system operates. Several interconnected models could be used to represent different aspects of the knowledge. The models are consultable (accessible) by applications at run time.
- A semantic application uses knowledge models in an essential way as part of its operation. Use of a model is often referred to as "reasoning over the model". Reasoning can range from a very simple process of graph search to intricate inferencing over the model.
- Semantic applications are thin because they work with “smart” data. All the business rules logic is held in the models shared across applications.

Figure 1 shows a simplest form of a semantic model, a taxonomy. The model describes government concepts that are part of Federal Enterprise Architecture (FEA). In a taxonomy connections between terms exist, but are not named. Therefore, the structure itself becomes a way to identify the nature of relationships. Taxonomies are hierarchies that establish “parent-child” relationship between its concepts.

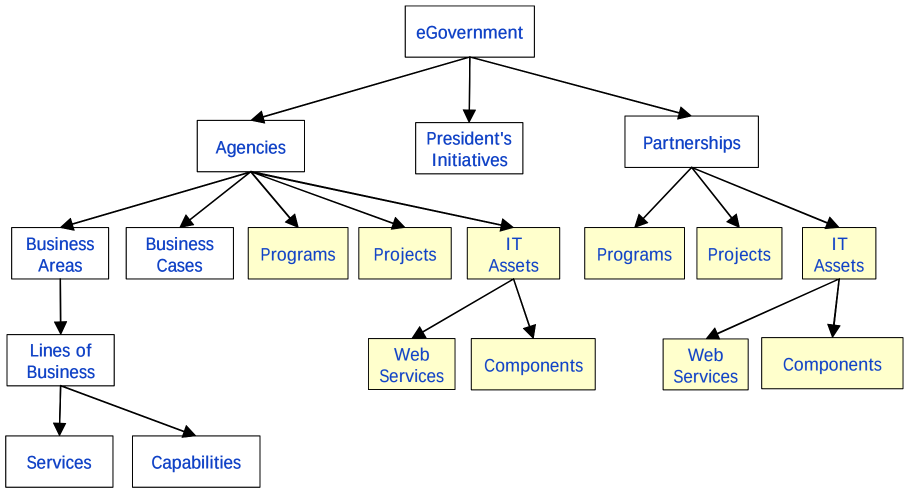


Figure 1: Example of a Taxonomy for e-Government

Because of the hierarchical nature of a taxonomy, some concepts have to be grouped under more then one category.

Figure 2 shows a richer model where relationships are explicitly named and differentiated. This model is called an ontology. Because the relationships are specified there is no longer a need for a strict structure. The model becomes a network of connections. New knowledge could be inferred by examining the connections between concepts. For example, the model below could be used to infer that a specific IT component has been developed in support of a given president's initiative. The model also identifies agencies that partnered in a development of this component.

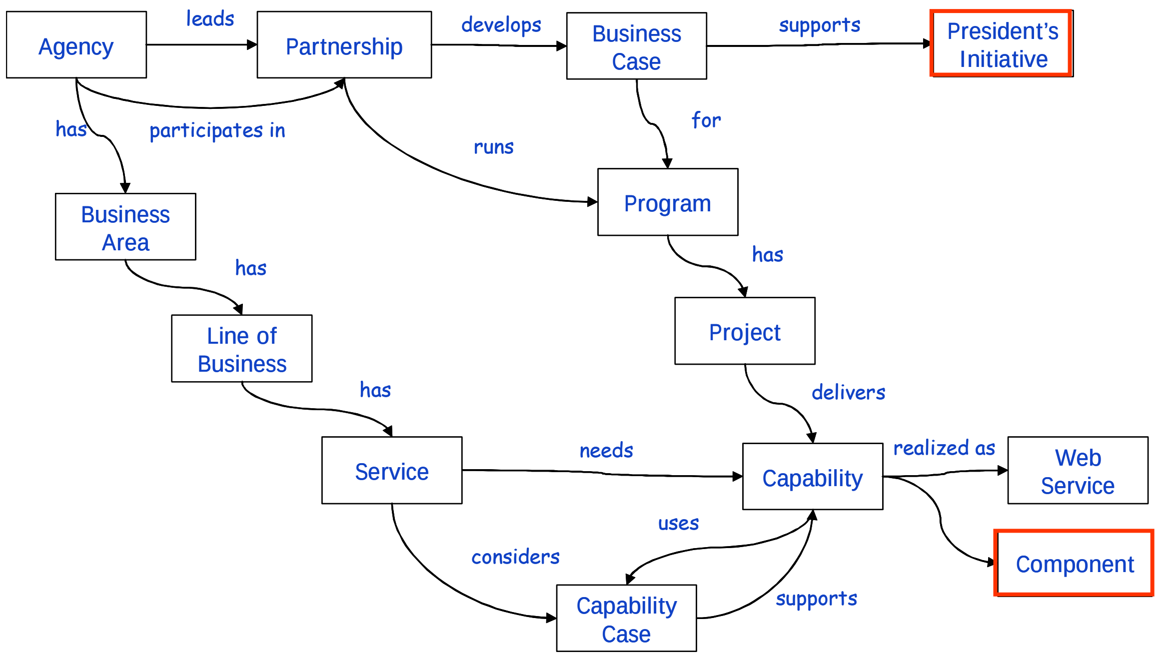


Figure 2: Part of the FEA Capabilities Manager Ontology Model

Simple ontologies are just networks of connections; richer ontologies include rules and constraints governing these connections as illustrated in Figure 3. The model shows how business cases have to be constructed with compliance to the FEA models. The schema for business cases required to be submitted by federal agencies is called Exhibit 300.

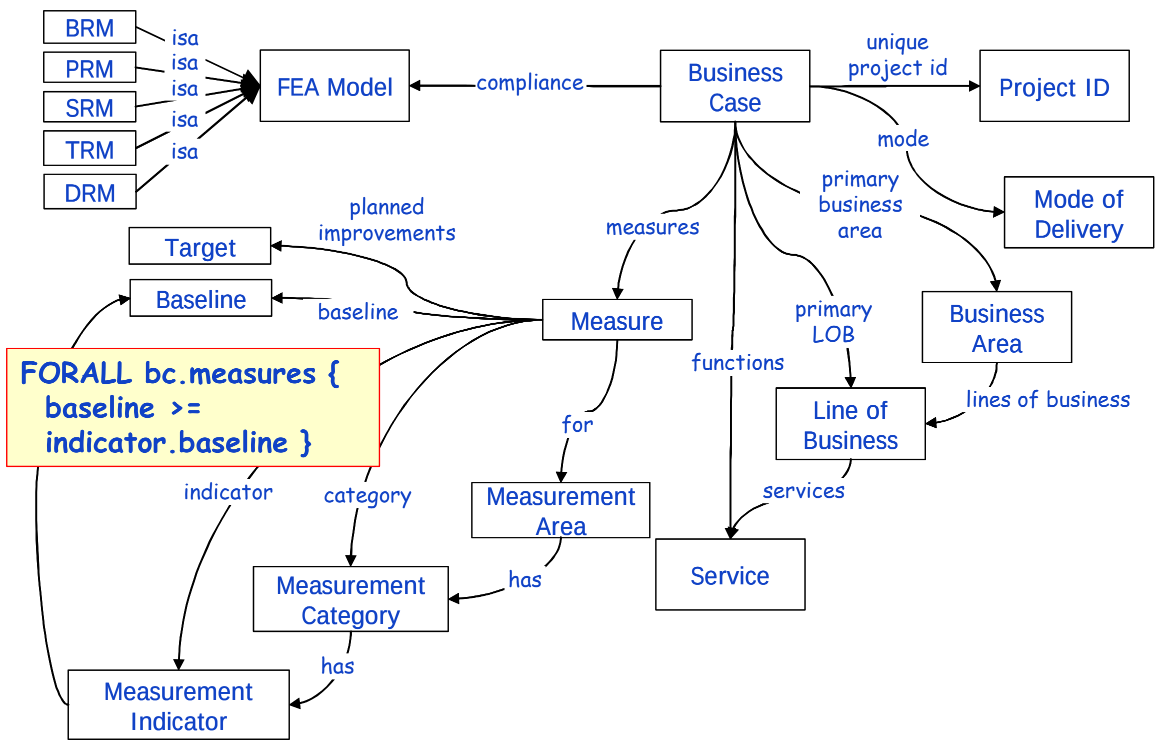


Figure 3: Example of a Rule for Exhibit 300 Measures

A simple rule for checking baseline values of measurement indicators is illustrated above. What the rule says is that the baseline values of all measures must be greater than or equal to the baseline values of their respective indicators. An example, from the FEA Project Management Office "Additional Guidance On The FE-Related Requirements in OMB Circular A-11" document, is shown in the table below.

Table 1: Example of a Business Measure Baseline

Fiscal Year	Measurement Area	Measurement Category	Measurement Indicator	Baseline	Planned Improvements	Actual Results
2005	Mission and Business Results	International Affairs and Commerce	# of US Exporters entering new market	5,386		

1.2 How Knowledge Models are different from other Software Models

A model describes how concepts and phenomena are similar and how they differ - the commonality and variability of concepts in a chosen area of interest. This area of interest is sometimes also referred to as a domain of discourse. The most commonly used models in software engineering are object and data models.

An object model is a representation of domain of interest using classes that encapsulate both, data and behavior. The object systems are built using 4 key ideas:

- classes as a way to organize similarities and differences among objects using inheritance hierarchies
- the power of encapsulation
- the notion of object identities that allow individual instances be distinguished, referenced and messaged
- the polymorphism that insulates objects from changes in other objects through dynamic binding mechanisms

Object models differ from ontology models in two important ways:

- **Modeling Intent**

  An object model is a specification of how a set of entities can encapsulate data and invoke behaviors on one another. The intent of the model is to provide realizable software where object behaviors become fragments of code. An ontology model is a specification of what is known in a region of interest. There is no need supply an execution mechanism because it is provided by the systems that work with ontologies. Object models and ontology models serve different purposes and should be seen in co-existence rather than in competition.

- **Richness of Constructs for Modeling Knowledge**

  Because ontologies capture knowledge as opposed to provide a basis for executable code, ontology modeling languages provide finer distinctions between relationship types and class expressions. They include:

- **Properties As First Class Constructs –** in an object model properties are local to each class. In ontology model properties exist independently of classes. They can be unified across the model and re-used in different classes. This enables powerful reasoning capabilities across classes and instances.
- **Relationships Among Properties –** in object modeling associations have limited semantics, they simply denote relationships between classes. In ontology models it is possible to create subclasses of relationships and describe different qualities of relationships. For example, a transitive relationship transfers a relationship across a connected chain of entities – if A is bigger than B and B is bigger than C and "bigger than" is transitive than A is bigger than C.
- **Class Expressions** – in an object model a relationship can only be expressed between one class and another class. In ontology model a relationship can be constrained to be valid for a set of different classes. In fact, ontology modeling languages offer a full range of set operators.
- **Class as a Viewpoint as well as a Specification –** in an object model classes are specifications of how instances will behave and manage operations. The role of ontology model is more general. It can act as a specification of instances class membership or as a means of knowledge discovery. In ontology model instances can carry sets of properties that allow them to be viewed as members of several classes at the same time. In fact, their membership in a class is dynamic and depended on the value of the properties.

A design of a relational database begins with a logical model of entities and relationships in a domain. We can regard this model as a simple ontology. Simple, because it lacks finer distinctions of relationship types and class expressions described above. The model is translated into physical tables. Relationships become names of columns or names of connecting tables. The knowledge about the nature of relationships is captured only in documentation and in the memory of people who have developed the models and is not available to the applications that work with the databases.

In a relational database many applications can share the same database, but in reality the schema of the data is typically fine-tuned to the needs of specific application. In a data model, each table in the schema dictates what this collection of records has in common. Differences are represented both by individual records, as well as record types. The relationships are held in special index tables and are not explicitly defined. Another schema (a different database) denotes a commonality of collection of records that are outside of the first schema's area of concern. It could also express a different view point on the same set of records as the first schema.

Semantic models are intended as a way for different agents (applications and/or people) to interoperate and to share meaning. The variations and commonalities semantic models represent are not of a single entity or stakeholder. By definition semantic models support multiple viewpoints. This makes them especially suitable for solving interoperability problems.

1.3 Standard Languages for Knowledge Modeling

What languages can be used for knowledge or semantic modeling? By now, we all have heard of HTML and XML. A few important developments preceded HTML, but many have occurred since XML became popular. What we are witnessing today is the emergence of standards for the semantic WEB. These and other important influences from AI, Software Engineering and Process Modeling make up what we are illustrating in Figure 4 as "The Tree of Knowledge Technologies"

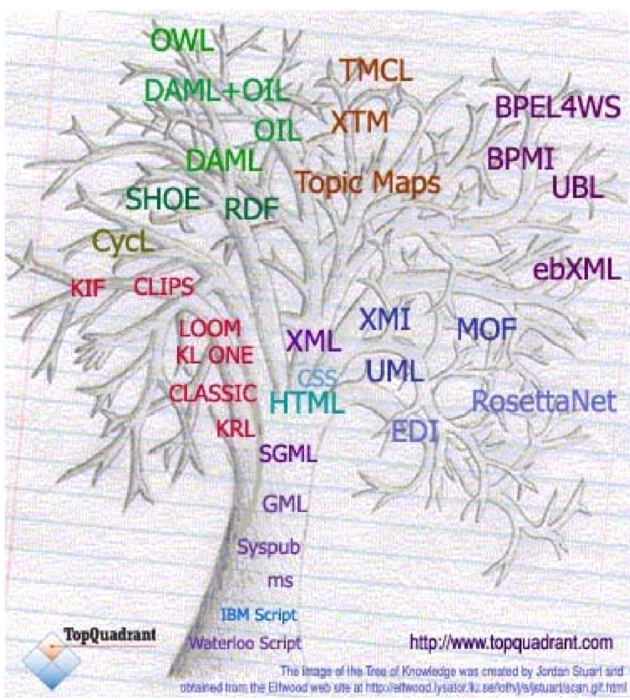


Figure 4: Tree of Knowledge Technologies

1.3.1 The History and the Current State

The current state of the art on representing and using ontologies has grown out of several efforts that started in the 1980s. Back then, KL-ONE was the most influential of the frame-based representation languages; it allowed for the representation of categories and instances, with inheritance of category properties, and a formal logic for expressing the meaning of properties and categories. At about the same time, rule-based systems were a promising technology. The NASA-sponsored C-Language Integrated Production System (CLIPS) became a de-facto standard for building and deploying rule-based systems.

The Knowledge Interchange Format (KIF), and it accompanying translation tool Ontolingua, were developed to allow knowledge to be shared among these different efforts, and provided the capability to translate knowledge bases in one representation language to another. These languages were ahead of their time. As a result, they have remained largely within the purvey of academia, gaining little commercial support.

With the advent of the World Wide Web, and the acceptance of XML as a de-facto standard for representation of information on the web, ontology efforts joined in. An early project at the University of Maryland produced SHOE, a system for expressing ontologies in XML, and marking up web pages with ontology-based annotations. Many of the ideas from this work made it into the World Wide Web Consortium (W3C) proposal for the Resource Description Framework (RDF) Language.

The DARPA Agent Markup Language (DAML) is built on RDF providing particular logical relationships that standardize the semantics of inferences that can be made over the information in a resource description. The DAML effort drew much of the formal semantics for its logical approach from a parallel effort called OIL (Ontology Inference Layer), which encoded the semantics of Description Logic into an XML-based language. The joining of the two efforts resulted in DAML+OIL language. It allows for a strict interpretation of the statements, so that reasoning agents can collaborate in their use of ontologies. DAML+OIL became a foundation for W3C Web Ontology Language (OWL).

While we have seen some use of UML as a knowledge language and a few MOF (Meta Object Framework) based integration solutions, RDF-based languages have the most potential for success. Table 2: provides a high level view of standards and an indication of the marketplace adoption.

Table 2: View of Knowledge Modeling Standards and Marketplace Adoption

	KIF/OKBC/ CG/CycL	UML	Topic Maps / XTM	RDF(S)	OWL
Description	*Legacy AI Languages*	*Universal Modeling Language*	*Topic Maps/XML Topic Maps*	*Resource Description Framework*	*Web Ontology Language*
Governance	ANSI American Natio /Other	OMG Object Management Group	ISO Internat OASIS	W3C	W3C
Years since proposed	*>10*	*>5*	*>5*	*>3*	*1 or less*
What is coming next		UML 2.0 (including OCL) MOF 2.0, Queries and Views	Constraint Language – TMCL, Query Language	RDF Query Language	Semantic Web Services-OWL-S, Rules-OWL-L
Commercial Support (as a KRL)	*2 or less vendors*	*2 or less vendors*	*5 or less vendors*	*More than 10 vendors*	*5 or less vendors*

1.3.2 XML-based Knowledge (Ontology) Modeling Languages

XML is being used to represent hierarchies of data. To go beyond hierarchies and simple taxonomies requires different kind of standards. The standards below represent convergence of conceptual modeling (AI heritage) and mark up languages (HTML and XML heritage):

ISO/IEC 13250 Topic Maps

Topic Maps defines a method of using SGML to represent networks of concepts to be superimposed on content resources (documents of various types), providing a means to represent, navigate, and query the network itself, rather than the full text of a document collection. ISO Topic Maps is an approach for representing topics, their occurrences in documents, and the associations between topics.

XTM is an XML serialization of Topic Maps.

Standard Status = Released

There are 2 commercial vendors that offer Topic Maps tools and a number of open source or research implementations. The Topic Maps standard has been developed in an effort parallel to RDF-based ontology languages. Convergence is not likely, but interoperability is possible. Several approaches for mapping between Topic Maps and RDF have been published. Topic Maps are applicable for building indices over *information objects* that represent unstructured information.

Topic Map community is planning to create TMCL (Topic Maps Constraints Language) which will be much the same to Topic Maps as OWL is to RDF.

RDF/S

The **Resource Description Framework** [W3C-RDF] defines a model and XML syntax to represent and transport metadata. RDF integrates a variety of applications from library catalogs and world-wide directories to syndication and aggregation of news, software, and content to personal collections of music, photos, and events using XML as interchange syntax. The RDF specifications provide a lightweight ontology system to support the exchange of knowledge on the Web.

Standard Status = Released

The Resource Description Framework (RDF) is a foundation for representing and processing metadata; it provides interoperability between applications that exchange machine-understandable information on the Web.

RDF Schema, RDF's vocabulary description language, is an extension of RDF. It provides mechanisms for describing groups of related resources and the relationships between these resources. RDF Schema does the same thing for RDF that DTD and XML Schema do for XML.

Standard Status = Released

RDF is making good inroads in terms of vendor support. Commercially available tools range from development environments to RDF databases to semantic integration and search/categorization solutions.

DAML+OIL and OWL

DAML + OIL is a semantic markup language for Web resources. It builds on earlier W3C standards such as RDF and RDF Schema, and extends these languages with richer modeling primitives. DAML+OIL was built from the original DAML ontology language DAML-ONT (October 2000) in an effort to combine many of the language components of OIL.

A DAML+OIL knowledge base is a collection of RDF triples. DAML+OIL prescribes a specific meaning for triples that use the DAML+OIL vocabulary.

The W3C Web Ontology Working Group (WebOnt) has been tasked with producing a web ontology language extending the reach of XML, RDF, and RDF Schema. This language, called OWL, is based on the DAML+OIL web ontology language. The only substantive changes from DAML+OIL are the removal of qualified number restrictions, the ability to directly state that properties can be symmetric; and the removal of some unusual DAML+OIL constructs, particularly restrictions with extra components. There are also a number of minor differences, including a number of changes to the names of the various constructs.

There are three levels of OWL defined (OWL Lite, OWL DL and OWL Full) with progressively more expressiveness and inferencing power. These levels were created to make it easier for tool vendors to support a specified level of OWL.

Standard Status = Released

DAML+OIL and OWL both depend on RDF/S semantics. Thus, the development of these standards is presently a fairly interlocking sequence. Today a number of vendors offer DAML+OIL support. As OWL matures we expect to see them moving from DAML+OIL to OWL.

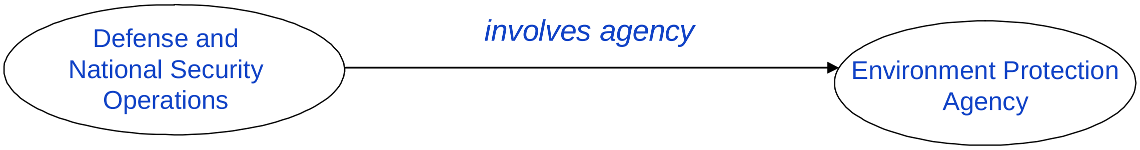
1.3.3 Differences and Similarities

Different approaches to semantic technology are distinguished by the different ways knowledge representation languages express the connections between concepts:

- Taxonomies and Thesauri have very simply connection
- RDF and Topic Maps have somewhat more complex ones:
    - RDF has very formal connections,

- o while Topic Maps have intuitive ones
- DAML and OWL have very powerful logical connections

A small example below shows a part of the FEA knowledge model, an RDF statement describing one of the relationships between Defense and National Security Operations and Environmental Protection Agency.




```
<FEA:Service rdf:about="&FEA;Anti-Terrorism"
                 rdfs:label="Anti-Terrorism">
                <FEA:service_of rdf:resource="&FEA;Defense and National Security
Operations"/>
                <FEA:involves_agency rdf:resource="&FEA;Department of Commerce"/>
                <FEA:involves_agency rdf:resource="&FEA;DoJ"/>
                <FEA:involves_agency rdf:resource="&FEA;Environment Protection Agency"/>
                <FEA:involves_agency rdf:resource="&FEA;FEMA"/>
                <FEA:involves_agency rdf:resource="&FEA;General Services Administration"/>
                <FEA:involves_agency rdf:resource="&FEA;State"/>
                <FEA:involves_agency rdf:resource="&FEA;Transportation"/>
                <FEA:involves_agency rdf:resource="&FEA;Treasury"/>
</FEA:Service>
```

Figure 5: RDF Example

1.4 Applications of Semantic Technology

Semantic technology can be applied in a number of different situations. The key to getting value out of it is picking the most appropriate application area. The table below lists a number of capabilities known to be successfully delivered by semantic technology. For each, we identify the reason why semantic technology is a good fit for implementing the capability. Alternative technical approaches are also described. The common downside many of the alternative approaches share is lack of scalability and flexibility needed to support the solution as the new information sources, new users and new applications are added or new requirements become important. Another words, they are simple to implement and work well in well bounded situations, but do not grow well. One exception is neural networks and other machine learning algorithms. In many cases this technology is complementary to semantic -- knowledge representation based -- technology, and could be used together very successfully.

Therefore, one of the key success criteria for implementing semantic technology is picking an area where the situation is fairly complex and/or extensibility of the solution is important. On the other hand, such situations are often perceived by companies as mission-critical. The tolerance to risk associated with new technology is low. As the number of early adopters' success stories starting to grow, they pave the road to broader adoption.

Table 3: Semantic Capabilities

Capability Intent	Semantic Technology Fit	Other Approaches
Answer Engine		
To provide a direct reply to a search questions as opposed to returning a list of relevant documents. It interprets a question asked in a natural language, checks multiple data sources to collect knowledge nuggets required for answering the question and may even create an answer on the fly by combining relevant knowledge nuggets.	Interpretation of questions using domain knowledge. Aggregation and composition of the answer. Also see Generative Documentation below.	Identifying frequently asked questions and posting answers to them.

Capability Intent	Semantic Technology Fit	Other Approaches
Automated Content Tagger		
To provide semantic tags that allows a document or other work-product to be "better known" by one or more systems so that search, integration or invocation of other applications becomes more effective.	Tags are automatically inserted based on the computer analysis of the information, typically using natural language analysis techniques. A predefined taxonomy or ontology of terms and concepts is used to drive the analysis.	Machine learning approaches based on statistical algorithms such as Bayesian networks.
Concept-based Search		
To provide precise and concept-aware search capabilities specific to an area of interest using knowledge representations across multiple knowledge sources both structured and un-structured.	Knowledge model provides a way to map translation of queries to knowledge resources.	Dictionary of synonyms and domain specific jargon could provide an approximation to concept-based search.
Connection and Pattern Explorer		
Discover relevant information in disparate but related sources of knowledge, by filtering on different combinations of connections or by exploring patterns in the types of connections present in the data.	Inferences over models to identify patterns using the principles of semantic distance.	Statistical algorithms such as Bayesian networks. Technologies could create visualization of complex data, thereby facilitating pattern discovery by humans or potentially by machine vision algorithms.

Capability Intent	Semantic Technology Fit	Other Approaches
Content Annotator		
Provide a way for people to add annotations to electronic content. By annotations we mean comments, notes, explanations and semantic tags.	Knowledge model is used to assist people in providing consistent attribution of artifacts.	Using fix templates for each type of artifact.
Context-Aware Retriever		
To retrieve knowledge from one or more systems that is highly relevant to an immediate context, through an action taken within a specific setting -- typically in a user interface. A user no longer needs to leave the application they are in to find the right information.	Knowledge model is used to represent context. This “profile” is then used to constrain a concept-based search.	Machine learning techniques based on statistical algorithms could be used to “understand” the context.
Dynamic User Interface		
To dynamically determine and present information on the web page according to user's context. This may include related links, available resources, advertisements and announcements. Context is determined based on user's search queries, web page navigation or other interactions she has been having with the system.	A model of context and a memory of activities are used to control UI generation.	Using XML interaction mark up languages and XSLT against a set of predetermined dialog choices.

Capability Intent	Semantic Technology Fit	Other Approaches
Enhanced Search Query		
To enhance, extend and disambiguate user submitted key word searches by adding domain and context specific information. For example, depending on the context a search query "jaguar" could be enhanced to become "jaguar, car, automobile", "jaguar, USS, Star Trek", "jaguar, cat, animal" or "jaguar, software, Schrödinger".	Knowledge models are used to express the vocabulary of a domain.	A dictionary of synonyms and domain specific jargon can be used.
Expert Locator		
To provide users with convenient access to experts in a given area who can help with problems, answer questions, locate and interpret specific documents, and collaborate on specific tasks. Knowing who is an expert in what can be difficult in an organization with a large workforce of experts. Expert Locator could also identify experts across organizational barriers.	The profiles of experts are expressed in a knowledge model. This can then be used to match concepts in queries to locate experts.	Simple profile-based approaches using fixed templates. Alternatives usually give poor results because of the lack of support for determining semantic distance and semantic similarity.
Generative Documentation		
Maintain a single source point for information about a system, process, product, etc., but deliver that content in a variety of forms, each tailored to a specific use. The format of the document, and the information it contains, is automatically presented as required by each particular audience.	Knowledge model is used to represent formatting and layout. Semantic matching is a key component of the solution.	Manual repurposing of the information. Creation of special one-to-one repurposing programs.

Capability Intent	Semantic Technology Fit	Other Approaches
Interest-based Information Delivery		
Filter information for people needing to monitor and assess large volumes of data for relevance, volatility or required response. The volume of targeted information is reduced based on its relevance according to a role or interest of the end user. Sensitive information is filtered according to the "need to know".	A profile of each user's interests is expressed in a knowledge model. This is then be used to provide "smart" filtering of information that is either attributed with meta-data or has knowledge surrogates.	Rules and collaborative filtering could be used for personalization.
Navigational Search		
Use topical directories, or taxonomies, to help people narrow in on the general neighborhood of the information they seek.	A Taxonomy that takes into account user profiles, user goals and typical tasks performed is used to drive a search engine. To optimize information access by different stakeholders, multiple inter-related taxonomies are needed. Taxonomies and ontologies are used to suggest related subjects.	

Capability Intent	Semantic Technology Fit	Other Approaches
Product Design Assistant		
To support the innovative product development and design process, by bringing engineering knowledge from many disparate sources to bear at the appropriate point in the process. Possible enhancements to the design process that result include rapid evaluation, increased adherence to best practices and more systematic treatment of design constraints.	Knowledge models are used to express design constraints and best practices.	Expert systems.
Semantic Data Integrator		
Systems developed in different work practice settings have different semantic structures for their data. Time-critical access to data is made difficult by these differences. Semantic Data Integration allows data to be shared and understood across a variety of settings.	A common knowledge model is used to provide one or more unified views of enterprise data. Typically this is done by using mapping. Rules are executed to resolve conflicts, provide transformations and build new objects from data elements.	One to one mappings and transformation of data sources.
Semantic Form Generator and Results Classifier		
To improve the data collection process and data input analysis by providing knowledge-driven dynamic forms.	A knowledge model is used to intelligently guide the user through data capture. The results are automatically classified and analyzed according to the model	Pre-defined forms.

Capability Intent	Semantic Technology Fit	Other Approaches
Semantic Service Discovery and Choreography		
Service Oriented Architectures enable increased reuse of existing services and the dynamic automation of processes through service composition and choreography.	Knowledge models are used to enhance the functionality of service directories. Invocation methods, terminology and semantic description allow the dynamic discovery of services by machines.	Predefine what services will be used by a process
Virtual Consultant		
Offer a way for customers to define their individual goals and objectives, and then show them what products and services can help them meet those goals. Understanding customer's goals and requirements through a questionnaire or dialog establishes a profile that helps you communicate effectively with them now and in the future.	A knowledge model of users and their work within a domain is used to provide intelligent guided support of interactive sessions.	Canned dialogs and responses.

1.5 Application Architecture

How does semantic technology fit into overall architecture of business applications? Figure 6 depicts typical application architecture.

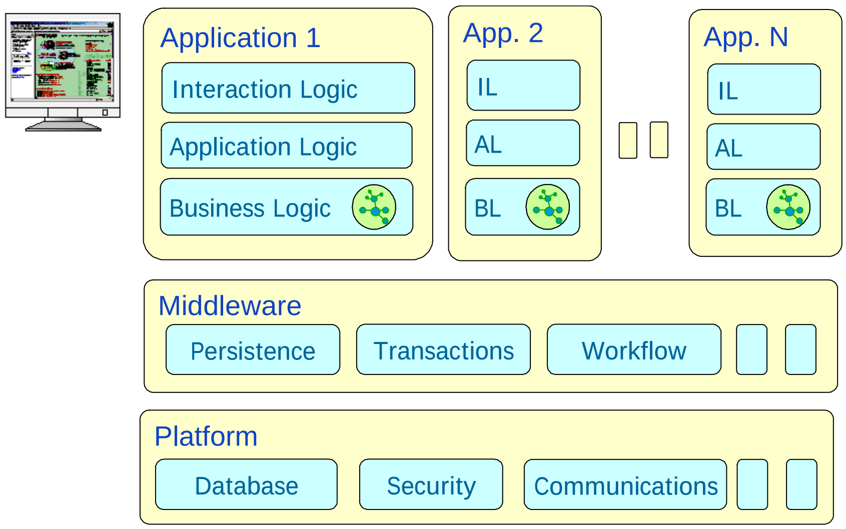


Figure 6: Typical Application Architecture

Semantic technology could be used to encapsulate business domain knowledge used by many applications. This means that the applications would become thinner as they no longer need to have their own representation of business logic. Instead they would need to have a way to consult a knowledge model. Such access is made possible through the use of semantic engines. Figure 7 shows a modified architectural view with each application having a semantic interface (SI).

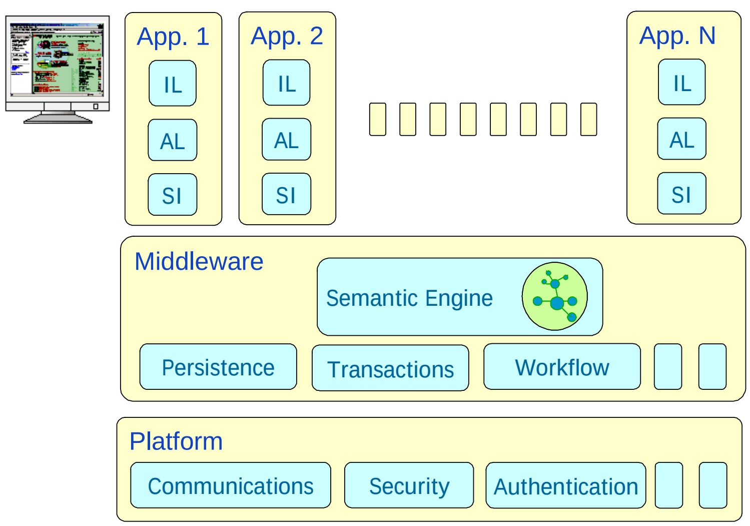


Figure 7: The role of Semantic Engine in the Application Architecture

This architectural approach ensures interoperability between diverse set of applications that operate in the same or related business domains. The interoperability is achieved by using a common set of models describing business concepts and their relationships as illustrated in Figure 8. This architecture can support interoperability between new types of applications built to work with semantic models as well as legacy applications that can connect to the model without changes to their existing logic.

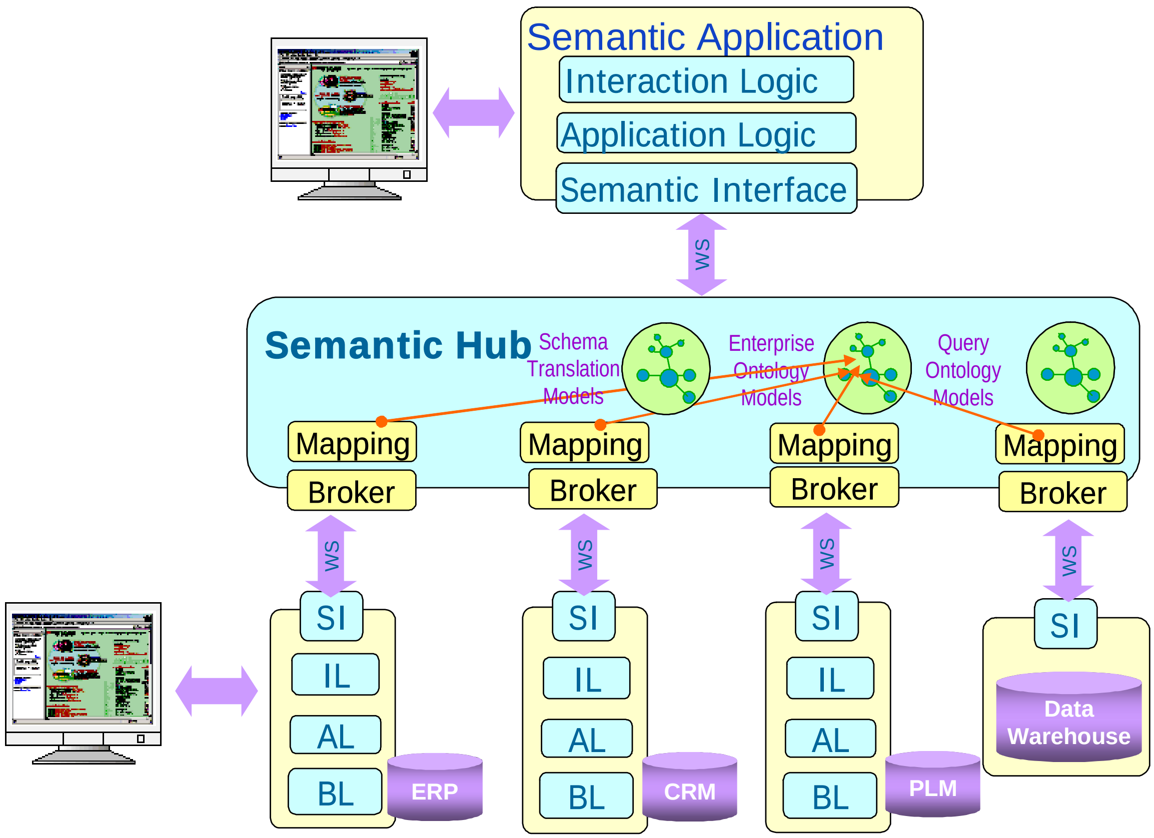


Figure 8: Architecture for Semantic Interoperability

The architecture described in Figure 8 distinguishes different types of ontologies. Semantic engines require not only ontology models for the domains of interests, but also models that express knowledge required to construct queries against different data resources and to achieve interoperability with different application protocols.

In the next section we will mention several vendors that provide commercial realization of semantic engines.

2. Semantic Integration, Strategies and Tools

2.1 Executive Summary

A growing number of semantic technology vendors are responding to the critical need to manage and integrate large numbers of disparate applications and data sources present in today's enterprise. This briefing is focused on the use of semantic technology to integrate structured data and applications and includes analyzes of offerings from 9 leading vendors.

The most common current solution to integration and translation is field to field mapping. Schemas from two data sources are imported and fields are mapped to each other. This approach doesn't scale well as the number of maps grows exponentially with each new data source. Enterprises working with this technology often discover that creating correct maps is a challenge because it requires that the person doing each mapping has an in depth knowledge of both data sources, which is rarely possible.

Semantic technologies offer a new way to integrate data and applications. Before making mappings, a model (or an ontology) of a given business domain is defined. The model is expressed in a knowledge representation language and it contains business concepts, relationships between them and a set of rules. By organizing knowledge in a discrete layer for use by information systems, ontologies enable communication between computer systems in a way that is independent of the individual system technologies, information architectures and applications.

Compared to one-to-one mappings, mapping data sources to a common semantic model offer a much more scaleable and maintainable way to manage and integrate enterprise data. The "common business model" terminology used here may remind readers of the enterprise data and process modeling initiatives. These initiatives have proven to be long on cost and resources and short on ROI. Does the use of semantic integration solutions depend on an enterprise-wide modeling effort? We don't believe so. In fact, we recommend a targeted way to start by situating your first semantic integration solution within a specific project, as opposed to having it as a separate initiative. The model has to be large enough to provide value – sufficient to integrate specific data or applications. It doesn't need to be enterprise-wide. Using knowledge representation approaches based on W3C standards ensures open, future proof implementations where models can be expanded, interlinked, merged and federated.

Semantic technologies are proving to offer enterprises competitive advantage. With the growing adoption of XML and the attendant need to reconcile meanings across different vocabularies, these technologies are becoming increasingly important. Beyond managing and connecting disparate enterprise data, key future capabilities include intelligent web services discovery and orchestration.

Now is the right time to begin developing the expertise in modeling and learning more about semantic technologies. As forecast by Gartner: “By 2005, lightweight ontologies will be part of 75 percent of application integration projects. The relative scarcity of skills in semantic modeling and the unification of information models may be the greatest challenge. Beyond initial development, the need for ongoing information-management processes at the enterprise level will severely tax most enterprises”[1].

To begin understanding and responding to these challengers, learning more about RDF/S and OWL is an important suggested step. Likewise, acquiring methodologies for modeling and information management is recommended.

2.2 A Need to Integrate and a Need to Manage

Integration is arguably the most pressing and expensive IT problem faced by companies today. A typical enterprise has a multitude of legacy databases and corresponding applications. The disconnected systems problem is the result of mergers, acquisitions, abundance of “departmental” solutions and simply implementation of many silo applications created for a specific purpose.

We know of a bank with over 40 different call center systems, a financial services company with more than 1,000 databases and a manufacturing company with over 2,000 CAD/CAM systems. These systems contain valuable information and often are still good for supporting specific tasks they were intended for. Unfortunately, the information they contain can not be leveraged by other systems without a considerable effort. When the changes in business needs or available technology require modifications to these applications to provide additional capabilities and to streamline workflows, integration and extension become a very expensive undertaking. Simply tracking all the enterprise data sources and their relationship to each other is proving to be a challenge. In fact, many IT organizations spend up to 80% of their budgets maintaining the legacy systems leaving limited funds to support new business opportunities or to satisfy new regulatory requirements.

Many companies have been moving to XML to take advantage of standards based integration.

[1] Gartner, "Semantic Web Technologies Take Middleware to the Next Level", 8/2002

However, XML doesn't capture the contextual meaning (or semantics) of the data. And a growing number of "standard" XML dialects (currently over 400) intended to standardize business vocabularies make the need for a semantic translation layer even more apparent.

2.2.1 The Most Common Solution Strategy

The most common solution to data integration and translation is field to field mapping. Schemas from two data sources are imported and fields are mapped to each other. Rules can be defined to split or concatenate fields or to perform other simple transformations. Once this is done the tool can do data translations either directly at run time or by generating code that will perform the transformations. There are a number of tools on the market that support this approach. Vendors include IBM and Microsoft. Some of the tools have been available for nearly a decade, but the adoption has been slow for a number of reasons:

- Field to field mapping works on a small scale. However, the number of maps grows exponentially with each new data source. Maintenance and evolution become a problem since any change in the schema of one data source will require you to redo multiple maps.
- Enterprises working with this technology often discover that creating correct maps is a challenge. It requires that the person responsible for each mapping has an in depth knowledge of both data sources, which is rarely possible. As a consequence, mapping mistakes are quite common.
- Mapping and translating between two schemas that are using a different design paradigm (i.e., different degree of normalization or nesting) can be very difficult. There is more then one way to design a schema. Performance considerations may result in de-normalized database schemas. When schemas are expected to change, designer may opt for a reflective design. Some XML schemas are deeply nested, others are shallow. Mapping between relational (RDBMS) and hierarchical (XML) stores can suffer from significant impedance mismatch of the models.
- Direct mapping may fail in the situations requiring more conceptual and conditional transformations.

Is there a better solution?

2.2.2 Semantic Solutions

Semantic technologies offer a new way to integrate data and applications. Before making mappings, an ontology (or a model) of a given business domain is defined. It can be "jump started" by importing data schemas. The model is expressed in a knowledge representation language and it contains business concepts, relationships between them and a set of rules. This is the knowledge that the users of the systems want to store and access, rather than the data that implements that knowledge. The knowledge model is then mapped to fields in databases, XML Schema elements, or operations, such as SQL queries or sets of screen interactions. This approach solves many maintenance, evolution and schema compatibility problems.

The key ingredients that make up an ontology are a vocabulary of basic terms, a precise specification of what those terms mean and how they relate to each other. The term 'ontology' has been used in this way for a number of years by the artificial intelligence and knowledge representation community, but is now becoming part of the standard terminology of a much broader community including object modelers and XML users. By organizing knowledge in a discrete layer for use by information systems, ontologies enable communication between computer systems in a way that is independent of the individual system technologies, information architectures and applications. As a common model an ontology helps in the management of enterprise data sources.

Once the data sources are mapped to the model it can be used as an enterprise data management tool and to transform and validate data at design or run time. We can also envision future applications composed of very thin components that dynamically change their behavior based on the interactions with the business knowledge embedded in the model.

The distinct advantage of knowledge representation languages as ways to express the model is that they are optimized for capturing relationships between concepts and for defining generic and specific rules (assertions) that logical reasoning can be based on. Some examples of such rules are:

- If A is a part of B and B is a part of C then A is a part of C
- If a person has blood-contact with someone at risk of an HIV infection risk, then they are at risk of an HIV infection
- If John wrote a paper on semantic integration, he knows about semantic integration

The attraction of logic as a technology for supporting semantic integration stems from the capability of logical languages to express relationships in generic ways, and the availability of sophisticated automated systems for finding combinations of related items that satisfy certain constraints. The variants of logic used for semantic integration (including Horn logic (prolog), frame logic, and description logic) differ primarily in the expressiveness of the logic and the tractability of the reasoning system. Another technology that provides similar capabilities is "means-ends analysis", which grew out of a different research background. Some vendors (Celcorp) base their integration products on this technology. Using models of knowledge, semantic engines can make inferences and create dynamic (on the fly) relationships between different concepts.

The model in the Figure 9 shows a unified view of billing and contractual databases. The blue arrows indicate explicitly defined relationships, while yellow arrows indicate derived ones. The derived relationships were established by the system based on the defined rules some of which are also shown in the figure below. For example:

- The rule "If customer is subject to a contract and invoice is billed to the customer then invoice is subject to a contract" has resulted in establishing a dynamic runtime connection between an invoice and a customer
- The rule "If contract has terms and invoice is subject to the contract then invoice is subject to each term" has built on the connection inferred by applying the previous rule and established connections between an invoice and the specific terms of the contract.

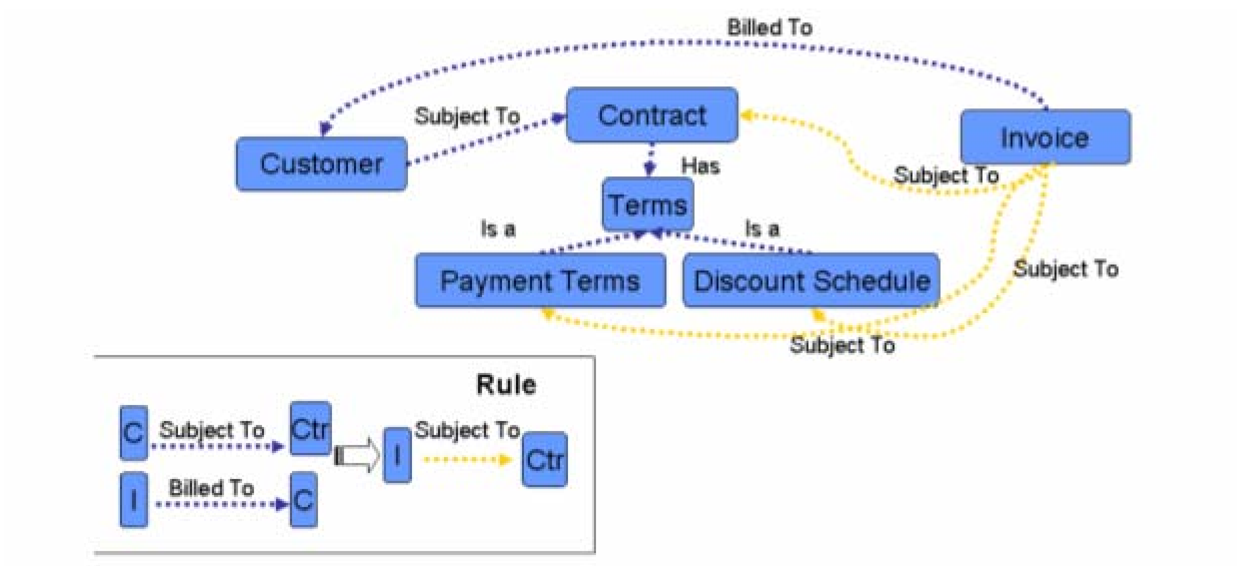


Figure 9: Illustration of a Unified View of Billing and Contractual Databases

Some ideas behind semantic models or ontologies for integration may remind you of metadata management. It is, in fact, based on the similar concepts. However, proponents of semantic integration argue that the use of W3C standard knowledge representation languages gives them distinct advantages:

- ***Open Standards.*** Using knowledge representation approaches based on W3C standards ensures open, future proof implementations where models can be expanded, interlinked, merged and federated.
- ***Rich Semantics.*** Knowledge representation languages offer support for richer and more precise semantics then UML, a standard language behind meta-data repositories. W3C languages like RDF (resource description framework), RDF Schema and the new Web Ontology Language (OWL) have been specifically designed to capture relationships between concepts and to define generic and specific rules (assertions) with the precision that logical reasoning needs.
- ***Native to the Web.*** RDF and OWL are serialized in XML and are, therefore, native to the Web. W3C sees semantic standards as a fundamental enabler for the next phase of web solutions.

2.3 Semantic Integration Vendors

Table 4 describes several companies offering semantic integration solutions. Most of the vendors in this emerging technology field are relatively young (less then 5 years old), privately held companies. Many are capitalizing on the research work that started in early 1990s.

Table 4: Overview of Semantic Integration Vendors

Vendor Name	Product Name	Description	Year Founded	Company
Celcorp	Celware	**Engine**: Server and Real-time Planner integrate applications streamlining users' workflow where multiple systems must be accessed in order to perform a task. The software uses intelligent agent technology based on proprietary extensions to the "Plan Domain	1990	Celcorp is privately held and based in Santa Monica, California. The company was originally established in Canada and has been offering business integration software for sometime. It has a number of reference clients.

Vendor Name	Product Name	Description	Year Founded	Company
		Model and the Graph Plan Algorithm." **Modeling**: Models are automatically generated by running Celware Recorder, a design time tool.		
Contivo	Enterprise Integration Modeling (EIM) Server	**Engine**: Server includes a Semantic Dictionary containing enterprise vocabularies, such as various XML, EDI, and ERP standards; a Thesaurus with synonyms that match business concepts; and a Rules Dictionary that governs the field level data transformation. **Modeling**: Modeling (mapping) is done using Contivo Analyst tool. Some pre-built maps are available.	1998	Contivo is a privately held company with offices in Palo Alto, California. Contivo's corporate investors include industry leaders BEA Systems, TIBCO Software and webMethods. Venture capital investors include BA Venture Partners, Voyager Capital and MSD Capital LP. It has received a 3rd round of funding in January 2003.
Digital Harbor	PiiE™	**Engine:** PiiE Fusion Server sits on top of J2EE application servers, EAI middleware, emerging web services to fuse back-office systems for use by front-office workers. Uses object-oriented Business Ontology. Includes a workflow engine. **Modeling**: Modeling is done using	1997	Digital Harbor is a privately held company headquartered in Reston, Virginia with offices in Provo, UT and Bangalore, India. Digital Harbor got its start in the U.S. Defense Intelligence Community and has spent over $35M on R&D since 1997. Digital Harbor is a profitable company. It has received $10M of funding from Insight Partners.

Vendor Name	Product Name	Description	Year Founded	Company
		PiiE Enterprise Designer.		
Modulant	Contextia Product Suite	**Engine**: Contextia Dynamic Mediation uses a central description of enterprise data called Abstract Conceptual Model (ACM) to enable disparate applications exchange information by transforming messages at runtime. It reconciles semantic conflicts among disparate applications and data sources. **Modeling**: Modeling is done using Contextia™ Interoperability Workbench capturing the meaning, relationships, and context of data elements of all source and target applications, and mapping them to ACM. The mapping specifications and ACM are then used by the Modulant Contextia Dynamic Mediation to transform data from source to target at runtime. The Interoperability Workbench accepts a variety of inputs for mapping and modeling, including XML schemas, native schemas, database tables, and delimited files.	2000	Modulant was founded in 2000, and subsequently merged with Product Data Integration Technologies (founded in 1989) in order to develop commercially-deployable software based on PDIT's proprietary technology and patent-pending methodology. Modulant is a private, venture-backed company whose existing investors include Sandler Capital Management, Guardian Partners and First Lexington Capital. Modulant's world-wide headquarters is in Charleston, SC, with additional offices in Long Beach and San Francisco, CA, Chicago, IL, Dallas, TX, Washington, DC, London, England and Stockholm, Sweden.
Network	Cerebra	**Engine**: Cerebra Inference Engine	2000	Founded in late 2000 to

Vendor Name	Product Name	Description	Year Founded	Company
Inference	Platform	creates dynamic connections between different ontologies using reasoning based on description logic. While Cerebra can work with the central model its value proposition is based on the assertion that only a few key connections between disparate schemas are needed. Cerebra can dynamically infer the rest of the connections thereby minimizing mapping efforts. **Modeling**: Modeling is done using Cerebra Construct, a MS Visio based graphical modeling tool.		commercialize a description logic reasoner from the University of Manchester. The company is headquartered in London, UK with plans to open US offices. Network Inference is backed by Nokia Ventures.
Ontology Works	IODE	**Engine**: IODE utilizes a central description of enterprise data to determine answers to complex queries. Each link in the enterprise ontology is mapped to a query in the "ontology database"; this can either be a warehoused database created as part of the ontology engineering process, or a mediated connection to a legacy database. Solutions to queries in the ontology are build using the rules and relations in the ontology, so that the "proof" of the result can	1998	The company is privately held and has offices in Maryland and Arkansas. In the first quarter of 2000, it completed development of an initial version (V 1.0) of its tool set and secured its first customer.

Vendor Name	Product Name	Description	Year Founded	Company
		be translated in a simple fashion into a program that runs over the databases, to determine the correct answer. **Modeling**: Modeling can be done using UML tools, translated into a proprietary Ontology Works Language.		
Ontoprise	OntoBroker	**Engine**: Data integration is done via a several step process that includes importing data schemas from existing databases, and using OntoMap to map concepts and relations from one ontology to the next. These mappings are translated into F-Logic statements, so that Ontobroker can reason over the combined ontology results in data references in the original data sources. **Modeling**: Modeling is done using OntoEdit and OntoMap. Two more tools are needed to complete this picture, which are a rule editor and a rule debugger, both of which are currently in the proposal stage. The rules state the actual connections between the newly	1999	The Ontoprise® GmbH is venture capital backed; it achieved a break even point in 2002. The company is headquartered in Germany. Ontoprise was founded as a spin off of the University of Karlsruhe which implemented the first version of technology in 1992.

Vendor Name	Product Name	Description	Year Founded	Company
		merged concepts, and are susceptible to bugs; hence they must be viewable and debuggable.		
SchemaLogic	SchemaServer	**Modeling**: SchemaServer captures and communicates data definitions (enterprise schema) used across all applications and languages. To help create the active repository of schema and metadata, SchemaServer imports existing schema, taxonomy and classification criteria from databases, applications or content management systems. It supports distributed, collaborative management of enterprise taxonomy. SchemaServer manages the associations and links among the separate schemas by providing the tools necessary to model, map, and describe the multiple relationships.	2001	Privately held company founded by ex-Microsoft employees. Located in Redmond, WA.
Semagix	Semagix Freedom	**Engine**: Freedom Metabase stores both semantic and syntactic metadata related to content items in either custom formats or one or more defined multiple metadata	2002	The Semagix name came into being in 2002, but the company started life in 1996, under the name of Protégé. Protégé was a management firm that incubated and launched several

Vendor Name	Product Name	Description	Year Founded	Company
		formats such as PRISM, Dublin Core, and SCORM. At any point in time, a snapshot of the Metabase (index) resides in the main memory (RAM), so that retrieval of assets is accelerated using the patented Semantic Query Server. The Semantic Enhancement Server (SES) classifies aggregated content into the appropriate topic/category (if not already pre–classified), and subsequently performs entity extraction and content enhancement with semantic metadata from the Freedom ontology. **Modeling**: Freedom has its own modeling tool. One of the strength of the product is its ability to populate ontologies by importing content from various sources.		companies in the Enterprise Content Management space. In August 2002, the company acquired Voquette, a metadata management company, to become Semagix. Semagix is privately held. It has three offices in the United States and Europe, with headquarters in Central London. R&D is performed in Georgia.
Unicorn Solutions	Unicorn System	**Engine**: The Unicorn is a design time tool and a script generator for integration with third party engine, such as WebMethods. **Modeling**: The Unicorn tool imports schemas from multiple data sources including XML, RDBMS,	2001	The company is privately held. It is headquartered in New York City with R&D in Israel. Unicorn's investors include: Jerusalem Global Ventures, Bank of America Equity Partners, Intel Capital, Israel Seed Partners, Tecc-IS and Apropos.

Vendor Name	Product Name	Description	Year Founded	Company
		COBOL, IMS, and EDI. They are then mapped to a central enterprise model (ontology). Mapping supports creation of data transformation rules. Unicorn can generate transformation scripts as executable SQL, XSLT, and Java Bean code.		

Other companies worth mentioning in this category include IGS (www.igs.com) and MetaMatrix (www.metamatrix.com) that have UML and MOF based approaches to integration, Miosoft (www.miosoft.com) that offers a highly scaleable run time data validation and consolidation platform based on a central model with a rich set of rules, as well as Vitria (www.vitria.com), an EAI vendor that incorporates business vocabularies.

The market for semantic integration is expected to grow fairly quickly fueled by the needs of enterprises and by the growing maturity of the AI (Artificial Intelligent) technologies that underlie many of these solutions. TopQuadrant estimates[2] that semantic technology is projected to grow from less than $2.0 billion in 2004 to around $63.0 billion in 2010 with a compound annual growth rate approaching 70%. Given the significant ROIs being realized by early adopters of semantic solutions reported here, we believe that level of growth is both reasonable and sustainable.

2.4 Capabilities of Semantic Integration Platforms:

We have identified the following as key capabilities offered by semantic integration solutions:

Management of Data Concepts and Schemas

- Creating and publishing shared vocabularies of business concepts
- Cataloging data assets, including their schemas and other metadata.

[2] For more information see Topquadrant's special report entitled "Business Value of Semantic Technology".

- Formally capturing the semantics of corporate data by mapping database and message schemas to the ontology
- Importing a variety of standard data definition formats
- Supporting model management and evolution

Data Transformation

- Generating scripts and transformations to copy or move the data from one data source to another

Dynamic Code Generation

- Generating executable code such as SQL, XSLT and Java
- Generating "wrappers" for data sources
- Embedding of business rules in models
- Automatic updates after change in the model and schemas

Semantic Data Validation

- Using inference rules to validate integrity of the data based on a set of restrictions. The inference rules will automatically identify inconsistencies when querying for information.

Run-time Support

- Scaleable semantic engine that supports high volume of real time queries

Orchestration of Web Services

- Integration broker
- Intelligent discovery and orchestration (composition and chaining) of web services

Table 5 compares capabilities currently offered by each of the vendors.

Table 5: Comparison of Capabilities Offered by Vendors

	Management of Data Concepts and Schemas	Data Transformation	Dynamic Code Generation	Semantic Data Validation	Run-time Support	Web Services Orchestration
Celcorp Celware	-	-	Yes	-	Yes	-
Contivo EIM Server	Yes	Yes	Yes	-	-	-
Digital Harbor PiiE	Yes	Yes	Yes	-	Yes	-
Modulant Contextia Product Suite	-	Yes	-	-	Yes	-
Network Inference Cerebra Platform	Some	-	-	Yes	Yes	-
Ontology Works IODE	-	Yes	Yes	Yes	Yes	-
Ontoprise Ontobroker	-	-	-	Yes	Yes	-
SchemaLogic SchemaServer	Yes	Yes	-	-	-	-
Semagix Freedom	Yes	-	-	-	Yes	-
Unicorn System	Yes	Yes	Yes	-	Some	-

Table 6 provides a detailed look at each product and its support for open standards.

Table 6: Maturity and Standards Compliance

Product	Product Adoption and Usage	Knowledge Representation	Reasoning Capabilities	Interfaces	Support for Web Services Standards
Celcorp Celware	Mature product, offers a unique approach to application integration. The product is being repositioned to “ride a semantic web”. Company has a number of reference customers in the financial services industry.	Proprietary, planning to go to RDF in 2003.	Based on proprietary extensions to the "Plan Domain Model and the Graph Plan Algorithm."	Import: Screen scraping, SQL statements	
Contivo Enterprise Integration Modeling (EIM) Server	Relatively mature, has a number of reference customers. Focused on complementing webMethods and	Proprietary on top of relational database, evaluating RDF	None evident, integration with a reasoning engine would be hard to implement until support for	Import: XML Schema, RDB (Oracle only), flat files Export: XML Schema (XSLT), EAI	XML, SOAP, WSDL

Product	Product Adoption and Usage	Knowledge Representation	Reasoning Capabilities	Interfaces	Support for Web Services Standards
	Tibco.		RDF is offered	(WebMethods, TIBCO), Java	
Digital Harbor PiiE Fusion Environment	Relatively mature, has a number of reference customers. Particularly strong in government sector.	Proprietary, object oriented, investigating support for OWL	Rules engine based on JESS	Import and Export: XML and other formats	XML, SOAP
Modulant Contextia Product Suite	Relatively mature, has a number of reference customers. Focused on government, STEP customers.	XML, proprietary, evaluating RDF	None evident	Import: XML, RDB, flat files, STEP 21 files Export: XML	XML, SOAP
Network Inference Cerebra Platform	New, currently in beta. Initial focus on biotechnology.	RDF, DAML+OIL, OWL	Description Logic	Import: XML Schema, RDB (JDBC), RDF/S, DAML+OIL Export: XML Schema (XSLT), RDF/S, DAML+OIL	XML, SOAP, WSDL
Ontology Works IODE	Relatively mature, has a number of	Proprietary	Robust, based on a	Import: UML, RDF/S	XML

Product	Product Adoption and Usage	Knowledge Representation	Reasoning Capabilities	Interfaces	Support for Web Services Standards
	reference customers in government.		proprietary Ontology language OWL (a variant of KIF, not related to w3c standard by the same name)	Export: RDB (Oracle, DB2), DDB, RDF/S, XML	
Ontoprise Ontobroker	Relatively mature semantic engine has a number of reference customers. New to the integration market.	RDF, DAML+OIL, OWL support planned	F-Logic	Import: RDB, RDF/S, DAML+OIL, XML Schema Export: RDF/S, DAML+OIL	XML
SchemaLogic SchemaServer	New. The product can unify structured and unstructured data management. Focuses on helping existing customers of Portal and Content Management	XML, Proprietary	No	Import: RDF, XML Schema Export: ?	XML, SOAP

Product	Product Adoption and Usage	Knowledge Representation	Reasoning Capabilities	Interfaces	Support for Web Services Standards
	products.				
Semagix Freedom	The company is new, but the underlying technology is relatively mature having been in development for several years. A number of reference customers are available.	Proprietary, planning to support RDF in 2004	Sophisticated conclusions about data are drawn using query engine rather than inferencing engine	Import: variety of document formats including spreadsheets, word documents, etc. Freedom offers a framework for building custom importers Export: ?	XML
Unicorn System	Relatively new, focused on enterprise data management. First customer implementations are in progress.	RDF, DAML+OIL, OWL support planned	A third party reasoning engine could be integrated with this standards-based tool	Import: RDB (Oracle 7i/8i/9i, MS SQL Server 7/2000, DB2), XML Schema, UML (via adopter), ERWin, RDF/S, DAML+OIL Export: RDF/S, DAML+OIL, SQL	XML

Product	Product Adoption and Usage	Knowledge Representation	Reasoning Capabilities	Interfaces	Support for Web Services Standards
				Transformation Scripts, XSLT	

Figure 10 compares how these solutions are positioned within the semantic integration space. The vertical axis represents a vendor's ability to integrate disparate information based on semantics. The horizontal positioning represents a vendor's solution focus. The vertical axis represents a progression – the higher positioning indicates more powerful semantic capabilities. The horizontal line doesn't end with an arrow because, unlike the vertical axis, it is not intended to represent a progression of capabilities. The right most position of a vendor indicates that its major strength is in "Integration and Orchestration". The vendor may also offer some support, but not the full functionality, in the areas of "Management of Data Concepts and Schemas", "Validation" or "Run-time".

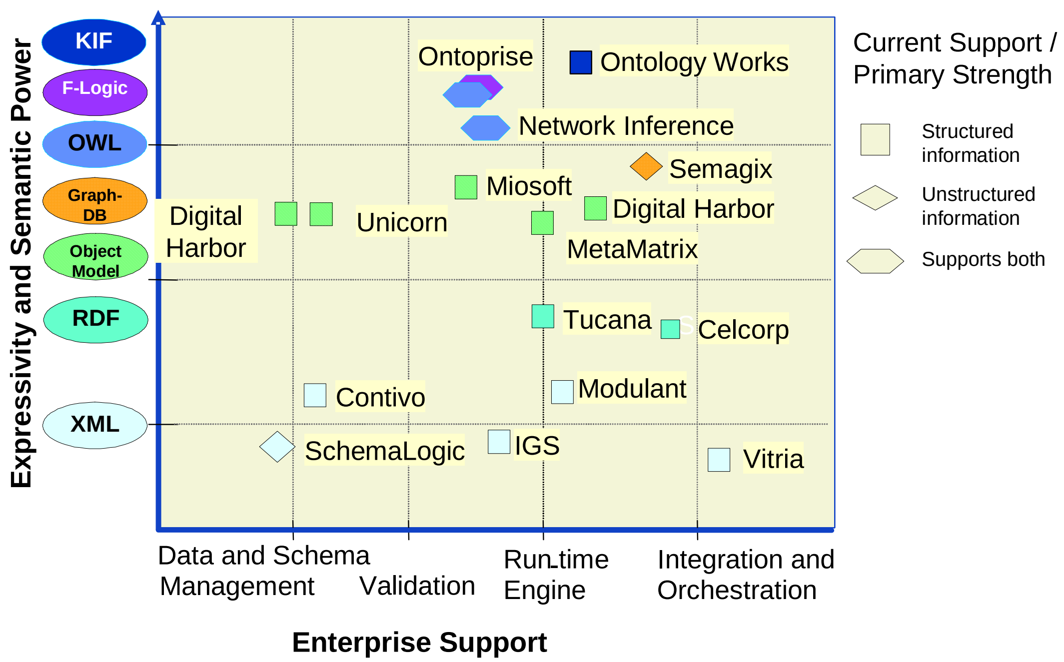


Figure 10: Positioning of Vendor's Solutions within the Semantic Integration Space

2.5 Recommendations for Getting Started:

The "common business model" terminology used by some vendors may remind readers of this report of the enterprise data and process modeling initiatives. These initiatives have proven to be long on cost and resources and short on ROI. Does the use of semantic integration solutions depend on an enterprise-wide modeling effort? We don't believe so. In fact, we recommend a targeted start by situating your first semantic integration solution within a specific project, as opposed to having it as a separate initiative. The model has to be large enough to provide value – sufficient to integrate specific data or applications. It doesn't need to be enterprise-wide. Using knowledge representation approaches based on W3C standards ensures open, future proof implementations where models can be expanded, interlinked, merged and federated.

You may be implementing or enhancing a CRM, portal or a supply chain solution. Any of these projects can be a good starting ground for the semantic integration. It could be used to help you with the data migration or to actually serve as an integration broker. Start with a limited model necessary to support your project. Grow it as needed. Using open standards based technology will enable you to leverage this model with other tools and projects.

Now is the right time to begin developing the expertise in modeling and learning more about semantic technologies. As forecast by Gartner: "By 2005, lightweight ontologies will be part of 75 percent of application integration projects. The relative scarcity of skills in semantic modeling and the unification of information models may be the greatest challenge. Beyond initial development, the need for ongoing information-management processes at the enterprise level will severely tax most enterprises"[3].

To begin understanding and responding to these challengers, learning more about RDF/S and OWL is an important suggested step. Likewise, acquiring methodologies for modeling and information management is recommended.

2.5.1 About Vendor Selection

Vendors covered in this issue have different strengths as well as different industry and problem focus areas. Choosing the right product will depend on:

- How well it integrates with your data and content sources, infrastructure and applications

[3] Gartner, "Semantic Web Technologies Take Middleware to the Next Level", 8/2002

- The degree to which you need run time support
- Product’s support for the industry specific XML schemas and vocabularies
- Vendor’s flexibility and interest in evolving the product to support your requirements

About the Authors

Irene Polikoff is an Executive Partner with TopQuadrant. She is an editor and the main writer of TopQuadrant's quarterly Technology Briefings.

Irene has over 15 years of experience in business application development and deployment, consulting, software development and strategic planning. Irene has held a number of executive positions at IBM. She was Senior Development Manager and Project Executive for worldwide consultant's tooling and methods. Most recently she was a Principal in the national Knowledge, Content Management and Portals Practice in IBM Global Services. Ms. Polikoff was part of the team that developed and deployed a world-wide project management method for IBM Global Services.

Prior to IBM, Ms. Polikoff held IT management positions at Fortune 500 companies where she was responsible for development and deployment of enterprise-wide mission critical information systems. Irene has a background in Operations Research and a strong interest in technologies for software innovation.

Dean Allemang is a Senior Consultant with TopQuadrant who has contributed content to sections of this report.

Dr. Allemang specializes in innovative applications of knowledge technology, and brings to TopQuadrant over 15 years experience in research, deployment and development of knowledge-based systems. Prior to joining Top Quadrant, Dr. Allemang was the Vice-President of Customer Applications at Synquiry Technologies, where he helped Synquiry's customers to understand how the use of semantic technologies could provide measurable benefit in their business processes.

Dr. Allemang has filed two patents on the application of graph matching algorithms to the problems of semantic information interchange. In the Technology Transfer group at Swiss Telecom, he co-invented patented technology for high-level analysis of network switching failures. He is a co-author of the Organization Domain Modeling method, which addresses cultural and social obstacles to semantic modeling, as well as technological ones.

Dr. Allemang combines a strong formal background (M.S. in Mathematics, University of Cambridge, PhD in Computer Science, Ohio State University) with years of experience applying knowledge-based technologies to real business problems. Dr. Allemang is a lecturer in the Computer Science Department at Boston University.

Companies interviewed for this report:

Celcorp - www.celcorp.com

Contivo - www.contivo.com

Digital Harbor – www.digitalharbor.com

enLeague Systems – www.enleague.com

Network Inference – www.networkinference.com

MetaMatrix - www.metamatrix.com

Miosoft – www.miosoft.com

Ontology Works – www.ontologyworks.com

Ontoprise – www.ontoprise.com

Tucana Technologies - http://www.tucanatech.com

Unicorn – www.unicorn.com

Additional TopQuadrant Technology Briefings are Available

- Dictionary of Search
- Modeling Techniques
- Semantic Solutions for Search and Self Service
- Interoperability

To access these papers, please visit our web site at www.topquadrant.com

About TopQuadrant

TopQuadrant, Inc. is a leading consultancy focused on the emergence of semantic web technologies and solutions. We provide a comprehensive combination of knowledge, resources, and services for semantic solutions from research to implementation. As business consultants and technologists, TopQuadrant acts as a trusted intermediary to help enterprises envision, architect, plan and realize knowledge-based solutions that deliver significant ROI. Our professionals bring expertise in artificial intelligence, object technology, adaptive systems, ontology engineering, knowledge and content management, publishing and media, semantic web and grid computing, and methodologies for knowledge, software and systems engineering.

TopQuadrant has developed a set of unique tools, methodologies, and services to jump-start successful building of semantic solutions that include:

- ***TopConnexion™,*** a multi-company knowledge service that conducts research; publishes case studies, technical assessments, and whitepapers; and produces workshops and conferences.
- ***TopDrawer™,*** a comprehensive knowledgebase of semantic technology capability cases — application solution patterns (e.g., for ontology-based knowledge applications).
- ***TopMind™,*** executive briefings on semantic technology; hands-on trainings in semantic web standards, languages, tools and ontology development.
- ***Solution Envisioning,*** scenario-driven workshops to explore system options and design future solutions through analogies and examples using Capability Cases.
- ***Semantic Solution Development Services,*** including optimal technology and vendor selection, ontology development, and full implementation of semantic solutions.

With a proven track record in the practical application of knowledge technologies, TopQuadrant helps clients transition to next generation, semantically integrated systems while sustaining and optimizing their investments in current technologies.