Understanding and Adopting Semantic Web Technology

by John Kuriakose

The industrial model that promoted manufacturing and engineering is slowly being phased out to usher in a new digital economy powered by developments in networking and information management. The old model placed controls on everything, including information flow, and permitted sharing of knowledge only through hierarchies of roles within the organization. The new business context places an emphasis on information availability, integration, and analytics and aims to foster collaborative communities whose shared contribution helps the organization respond to business opportunities in an efficient and effective manner.

Information and communication technology plays a vital role in this new context to enable these collaborative groups to interact and make sense of information from diverse data sources. The new context requires the effective deployment of technology for capture, storage, and dissemination of knowledge to aid analysis and interpretation.

BUSINESS VALUE: WHY CONSIDER SEMANTIC WEB TECHNOLOGY?

Business strategy and action are very much based on specific knowledge about the business context layered above general knowledge about a business domain and customer preferences. One aspect of strategy development also involves gleaning precise insights from existing data and previous results. This function completes the lifecycle shown in Figure 1 by providing feedback for further strategy development based on actual data. This assessment and feedback function is primarily characterized by information aggregation, integration, and summarization.

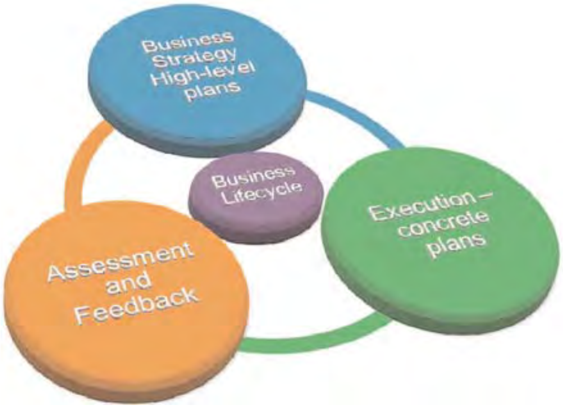

Business Strategy High-level plans
Business Lifecycle
Execution – concrete plans
Assessment and Feedback

Figure 1 — Role of SWT in deriving insights from existing data for assessment and feedback.

The same abstract principle is also applicable for IT strategy and management. In both these contexts, the feedback loop requires a high-level description of the entities, their states, and the relations between them. This description acts as a "lens" that represents how a decision maker chooses to see and interpret the dynamics between various stakeholders, the business entities, and their relationships with the organization and between each other. This model of the business world forms the basis for all interpretations that guide business thought and investment. Much of this model and its underlying general context are either implicit or at best expressed in natural language.

I see an opportunity to apply Semantic Web technology (SWT) to realize and accelerate this vision of a connected collaborative enterprise to promote business agility. Agility in this sense will be achieved through the deployment of smart solutions that can bring about deep integration across diverse data sources and also offer methods for deriving abstract or summarized views from this massive data store based on stakeholder concerns. These relevant views of actual data will give key decision makers the support needed to adjust and align execution plans for better results.

SEMANTIC WEB TECHNOLOGY

The Semantic Web[1] is a recent research project that aims to build the infrastructure to support the creation of a machine-readable Web (Web 3.0), where agents will collaborate to exchange information and make useful inferences to support and enrich human activity on the Web. SWT has evolved out of the Semantic Web research project and provides the right primitives for building the next wave of interesting enterprise applications with the ability to relate, integrate, abstract, and reason on data.

The use of metadata to achieve data exchange,[2] interoperability, and discovery[3] has been known and applied previously. According to ISO/IEC 11179-1, metadata is:

> ... the information and documentation which makes data understandable and shareable for users over time. Data remains useable, shareable, and understandable as long as the metadata remain accessible.[4]

The ISO standard also states that it is the obligation of organizations producing data to make available the metadata that supports the formal interpretation and use of this data.

Creating computer-based systems that can understand the meaning of data requires a formal representation of the meaning and context of data using "semantic metadata." Formal representations of knowledge have been the focus of knowledge representation and reasoning research[5-7] for more than 20 years within the artificial intelligence community.

For our discussion, "ontology" is the semantic metadata formally represented in the form of the concepts, relations, and rules that constitute a shared understanding of a domain.[8] An ontology language describes the general or background knowledge in a domain in terms of concepts and relations, while a data language such as the Resource Description Framework (RDF) describes specific instances or individuals in that domain. Ontology languages such as the Web Ontology Language (OWL)[9] and data languages such as the RDF[10] form the core of SWT. RDF relies on a simple triple-based data model to capture specific data related to individual elements in the domain.

A triple, or a fact, is a basic unit of knowledge storage and is at the core of a semantic data model. It is composed of three elements: a subject, a relation, and an object. The subject and the object in the triple refer to some individual elements in the domain. Each element is an instance of a concept in the domain terminology. These facts along with the elements constitute specific knowledge that varies, while the concept constitutes the background knowledge in the domain and is applicable to all instances. For example, "Account" and "Balance" are concepts in the Banking domain, and 256665:Account and USD 995.25:Balance are instances of those concepts. "256665:Account hasBalance 995.25:Balance" represents a fact using the *hasBalance* relation.

SWT provides the interoperable standards-based infrastructure to build, query, and use semantic metadata to augment our understanding of data within the enterprise. This application of ontologies and reasoning to make sense of structured and unstructured data is what we refer to as the semantic enterprise within the enterprise space.

Meaning and Consequence in Semantic Models

Current languages and technology for storing and managing data, whether XML- or RDBMS-based, capture the structure of data in some syntax. However, what is missing is the meaning of the data element, which is absolutely required for automated interpretation of data. SWT describes the precise meaning of data elements by associating them to semantic metadata. The semantic metadata expresses concepts that are interconnected by various kinds of relations between them (see Figure 2). The meanings for concepts (e.g., Customer, Account, Invoice, Payment) in a semantic model are derived from these relations to other concepts in the context. Thus each concept has a semantic space that includes a set of related concepts. The key relations

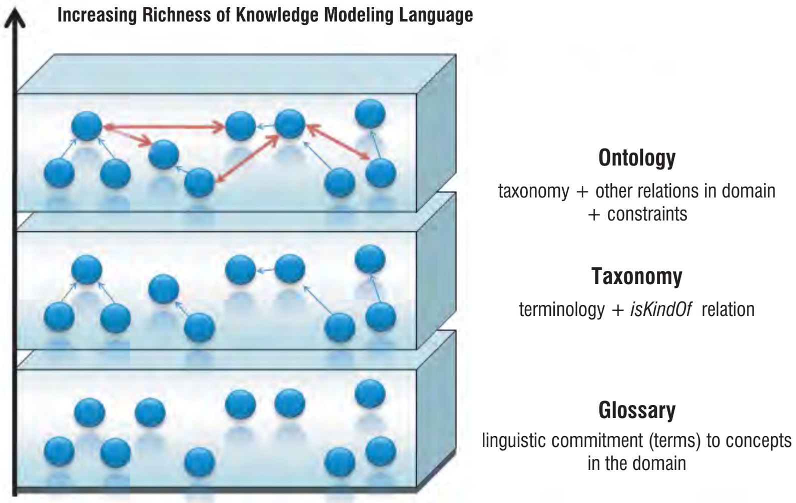

Increasing Richness of Knowledge Modeling Language
Ontology
taxonomy + other relations in domain + constraints
Taxonomy
terminology + *isKindOf* relation
Glossary
linguistic commitment (terms) to concepts in the domain

Figure 2 — Varying richness of metadata relations.

used to define the semantic space of a concept are as follows:

- *isKindOf*. Some concept is a subconcept of, or fully contained within, the semantic space of some other concept.
- *disjointWith*. Some concept is completely separate from some other concept (does not share semantic space at all).
- *equivalentTo*. Some concept completely overlaps the semantic space of some other concept.

There may be other relations in the domain of knowledge as well, such as *marriedTo*, *childOf*, *employedAt*, and so on.

Current languages and technology for information management also lack the ability to compute consequence. An assertion in semantic models is what has been explicitly stated to be true. Consequence is what follows or what is computed (by deductive reasoning) from existing assertions or other consequences. The key requirement is that the system always maintains logical consistency. The process of computing the consequence from what has been stated earlier is known as inference. This ability to compute consequence is the key differentiator between SWT and standard data technology. It is this inference that is applied to either verifying data against the rules described by the ontology or even making new inferences on the basis of relations that were not expressed in the triples but could be concluded from them.

Figure 3 shows an example of a simple semantic model and the consequences derived:

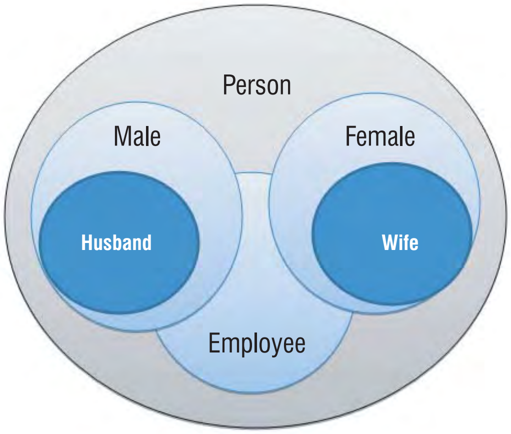


Figure 3 — Semantic space: the precise meaning of concepts based on context.

- A Person is always either a Male or Female (in other words, both Male and Female are kinds of Persons) — *isKindOf* relation.
- Male is completely distinct and separate from Female — *disjointWith* relation.
- Some Males are Husbands and some Females are Wives — *isKindOf* relation.
- Some Persons are Employees. Therefore some Males and some Females are Employees — *logical consequence.*
- Some Husbands and some Wives may be employees — *logical consequence.*

Key Characteristics

SWT is characterized by five major features or capabilities:

- **Flexible data model.** This is based on triples that supports modeling multidimensional relationships between data elements in a schema-independent manner. The triple-based data model is structured and does not require a schema, though it may use one.
- **Computational reasoning.** SWT provides the ability to compute logical conclusions from existing data (what is already known) by combining the ontology or semantic metadata as a precise and explicit expression of the background knowledge of the enterprise concepts.
- **Information integration.** SWT can relate and integrate data elements from diverse data sources and domains into a single model using bridge relations between multiple ontologies. The ontology constraints also serve to derive equivalence relations between individual data elements.
- **Information summarization.** The SWT stack includes a declarative rules language used to define abstractions (high-level concepts) as the precise decomposition from other detail concepts (low-level concepts) across multiple ontologies.
- **Information query.** SWT also defines an ontology-based query language that enables users to ask questions based on the concepts and relations in the ontology. The query focuses on the domain entities and relations without any reference to how the data is actually stored and organized at the physical level. A query constructed using terms in the user's vocabulary is answered by using the ontology to translate it to other concepts and relations that have actual data or instances associated with them.

Major Components of SWT

Figure 4 shows the major components that will be required in deploying SWT solutions in the enterprise. The key parts of this stack — the basic triple-based data model, the ontology language, and the query language (based on triples) — have been standardized by the W3C.

Key Tasks Involved in the Deployment of SWT

As shown in Figure 5, enterprise adopters of SWT must understand and plan for three distinct tasks. *Semantic modeling* involves the creation of semantic metadata in languages such as OWL, describing various units or aspects of the business. *Data population* involves translating structured enterprise data (RDBMS, XML, etc.) to RDF triples associated with the semantic metadata. This task is more complex when dealing with unstructured text, since it requires information extraction (extracting structured triples) using natural language–processing techniques and then refining the results with assistance from a domain expert. The final step is conceiving and implementing specific *information applications* that will embed and use the SWT components and to search, query, and reason on this data annotated with semantics.

BLUEPRINT FOR A SEMANTIC ENTERPRISE

A semantic computing platform is the architectural realization of the application of SWT to the enterprise context (see Figure 6). The platform provides the foundation on which applications that exploit semantic metadata will be built. SWT evolved out of the need to explicitly describe meaning and context for existing data. This implies that it has application within the enterprise wherever data is currently stored and used. Typical applications of a semantics platform to the enterprise are outlined in the next section.

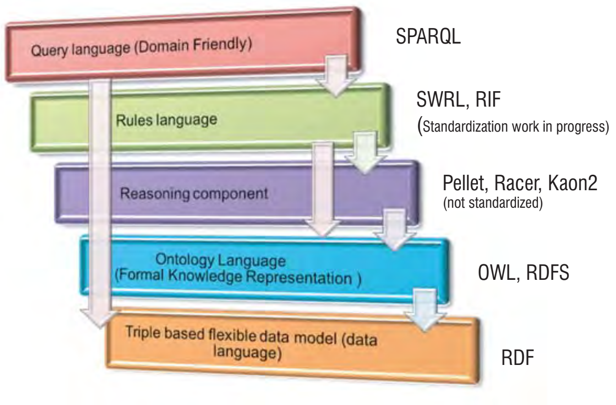
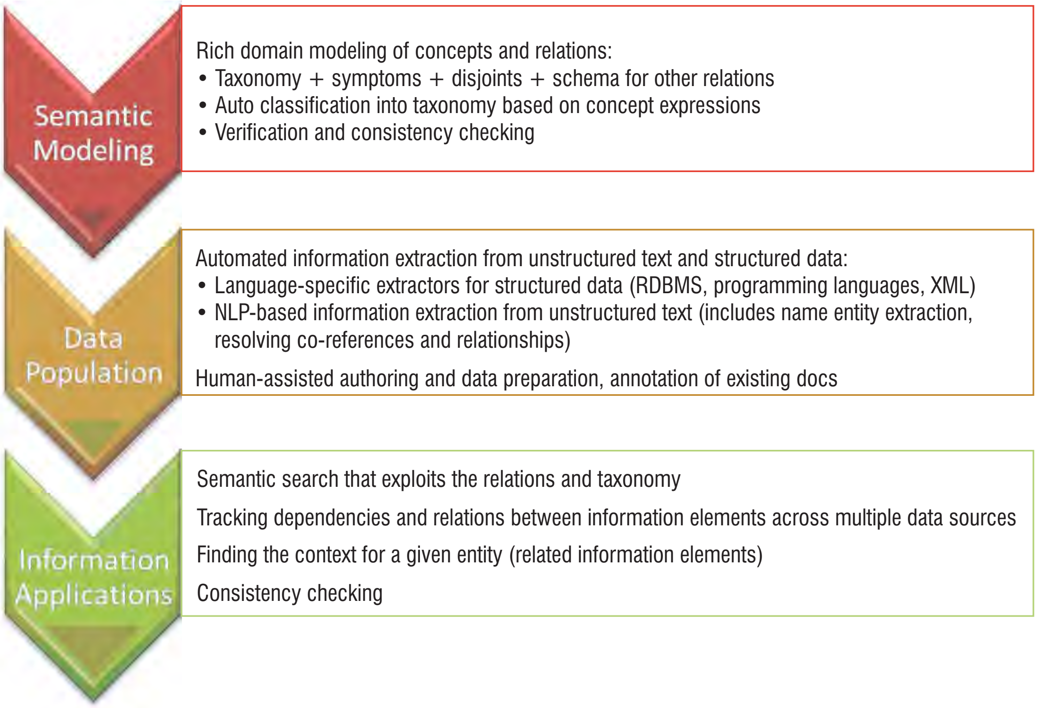

Query language (Domain Friendly)
SPARQL
Rules language
SWRL, RIF
(Standardization work in progress)
Reasoning component
Pellet, Racer, Kaon2
(not standardized)
Ontology Language
(Formal Knowledge Representation)
OWL, RDFS
Triple based flexible data model (data language)
RDF

Figure 4 — Semantic technology stack showing components and languages.



Rich domain modeling of concepts and relations:
- Taxonomy + symptoms + disjoints + schema for other relations
- Auto classification into taxonomy based on concept expressions
- Verification and consistency checking

Automated information extraction from unstructured text and structured data:
- Language-specific extractors for structured data (RDBMS, programming languages, XML)
- NLP-based information extraction from unstructured text (includes name entity extraction, resolving co-references and relationships)

Human-assisted authoring and data preparation, annotation of existing docs

Semantic search that exploits the relations and taxonomy

Tracking dependencies and relations between information elements across multiple data sources

Finding the context for a given entity (related information elements)

Consistency checking

Figure 5 — Process for SWT adoption.

USEFUL APPLICATIONS

Enterprise Information Management

SWT can deliver significant value when applied to business intelligence (BI) and information search solutions in enterprise information management. Ontologies support better communication, explanation, and prediction, as well as better mediation between data representations. Current technology for managing data is mature in handling the operational requirements of scale and performance within the enterprise context. RDBMS and XML together cover a large portion of structured organizational data. These technologies are known to scale and support querying on the underlying data store.

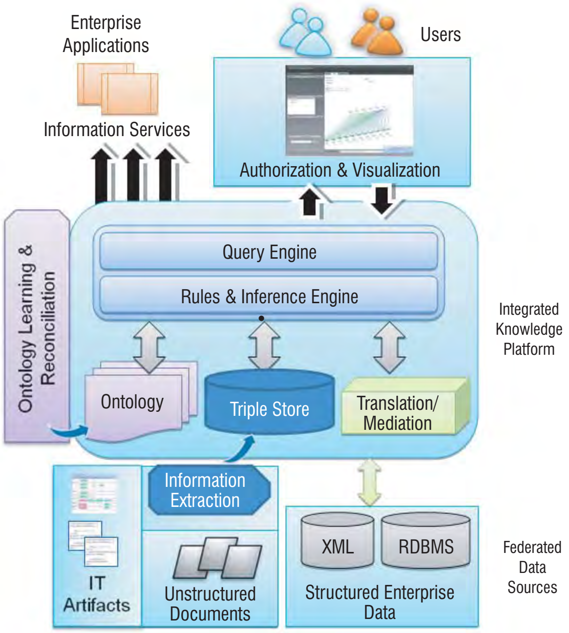


Figure 6 — Semantic enterprise architecture.

However, the scale and complexity of the current business context demand solutions beyond simple storage and retrieval — solutions such as mining and discovery of new relations from existing data and flexible views on an integrated IT portfolio database. Customer relationship management (CRM), fraud detection, compliance management, and many mining applications require this additional capability to reveal new insights from existing data.

Semantic Search

Current search technology is based on statistical occurrences of search keywords in a document corpus. This approach suffers from two primary drawbacks: it has low precision and is extremely sensitive to the actual words entered by the users. It has no understanding of the context of the keyword. Further, current search technology returns documents and not information content.

Semantic search within our scope is search over a formal knowledge base that includes the ontology and the individual data elements in RDF. Population of the knowledge base involves converting existing structured data into triple-based representation (RDF) and extraction of structured information triples from unstructured text. Once the knowledge base is populated, semantic search exploits the ontology relations to find related information content. In the example shown in Figure 7, a knowledge base is populated with a micro-ontology for the movie and mobile content domain. The search term used is "DiCaprio ringtone"; however, there is no ringtone associated with Leonardo DiCaprio, and so occurrence-based search simply cannot be considered. The ontology-powered search first looks for paths between the concepts (Person and Ringtone) involved in the ontology and then retrieves the correct RDF triples based on those paths.

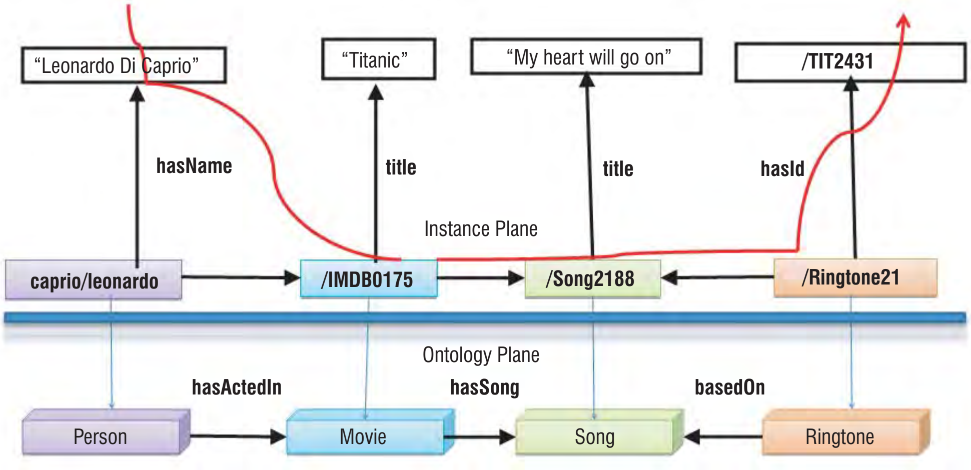


Figure 7 — Semantic information search.

Semantic Access to Structured Enterprise Data

Existing methods to access structured data are intrinsically coupled with the technology choice at the physical data level. For example, we use SQL for access to relational databases and XQuery/XPath for access to XML data. What we want is to enable information access that is free from the data management technology used in the physical layer. SWT permits analytic queries on data to be expressed against the conceptual data model as opposed to expressing queries against a physical data model (using tables, columns, joins, etc.).

Ontologies as rich information models provide the required expressivity and concrete language for conceptual information modeling in the enterprise. This implies that any existing database (relational, object, etc.) schema can be mapped and transformed into an OWL-based ontology definition. Ontology-based query language satisfies the exact requirements for semantic data access against the semantic data model expressed in ontologies.

Once the ontology has been extracted and mapped to the schema, there are two major options for dealing with the data in databases. One can either transform the data as triples into the semantic RDF store or simply use the mapping between the ontology and the physical data to mediate data access by translating semantic queries into the physical layer (in this case, SQL). It is possible to accommodate both these approaches within a single platform based on specific considerations for each database. The ontology defines the business terminology, while the data is maintained in the RDBMS. By establishing mappings between the ontology concepts and the physical schema, one can enable semantic querying. A useful feature of semantic querying is that it presents a unified business model across multiple databases and other data sources.

Smart Business Intelligence

The next generation of BI platforms will be powered by SWT and distinguishable by their use of semantic data models and schema-less data warehouses with greater flexibility and adaptability. This in turn will result in lower TCO and improved ROI. These systems will exploit SWT's information integration feature, establishing the relations between data elements that refer to the same concept but are expressed in distributed data sources, possibly using different languages.

Current BI platforms primarily employ a data warehouse based on a customized unified schema designed based on specific analytic needs. Existing data has to be cleansed and transformed (if necessary) before being moved into the warehouse. The RDF-based data model does not require a schema, and all attributes are explicitly modeled using relations. Information integration

Table 1 — A Comparison of Semantic and Relational Data Models

Semantic Data Model	**Relational Data Model**
Data represented at conceptual level.	Data represented at physical level.
Data is based on a flexible data model that can express any relationship — the schema is captured using expressive concept or ontology language.	Data is constrained by a rigid schema. Schema language has primitive expressivity compared to ontology language.
Stores information — data with context. Meaning is formally expressed and explicit.	Stores data — implicit informal meaning. Relations are translated as columns or constraints.
Captures subsumption (*isKindOf*) relations between concepts and relations.	No support for *isKindOf* relation — subsumption hierarchy is missing.
Domain-friendly language is used to express queries; everything is explicit.	Queries are bound by the schema, and explicit low-level joins have to be specified.
Rules language expresses new concepts and relations as expressions over existing ones.	Concepts and relations are limited to what is defined in the schema. No derivations — absence of rule support to define intentional concepts and relations.
Capable of making inferences on existing data by leveraging ontology.	No inference capability.
Information access is domain dependent and schema independent — ideal for information integration across diverse sources.	Information access is bound by the schema — difficult to merge and reconcile. Ideal for managing controlled data. Reasonable scalability and query performance.

from multiple operational databases is achieved by triplifying the data into an RDF-based triple store or mediating the data through the ontology. Further, by describing precise mapping rules between various stakeholder ontologies, semantics-powered BI will deliver automated information summarization from low-level operational data stores that will provide relevant information support while enabling drill-down into details.

Enterprise Application Integration

The information modeling, integration, and query features of the platform can be leveraged to improve IT-business alignment and enterprise application integration by integrating models and artifacts from business process management (BPM) and service-oriented architecture (SOA) into a single knowledge repository.

Enterprise Architecture

SWT can be an effective means of creating a machine-processable description of the various entities, functions, and relations between elements across all layers in a traditional enterprise architecture (EA) model. The technology permits slicing and dicing through the functions and layers to compute various views into the EA model from multiple stakeholder perspectives. For example, the platform can be deployed to create an integrated knowledge repository of IT and business artifacts and models, including business process models, entity models, physical database schema, use case models, application source code, and configuration files. The information integration and summarization capability of the platform will help multiple stakeholder concerns to be satisfied at varying levels of detail and scope from the single repository.

Software Engineering and Information Systems Development

SWT provides the infrastructure to create integrated knowledge repositories that import information from requirements, architecture, and design and from application programs and databases. This repository and the features of the technology then form the basis for semi-automatic traceability and impact analysis in software engineering.

Active Repositories in IT Management and Software Engineering

The current software engineering landscape, characterized by distributed teams, aggravates problems with informal knowledge management. Enterprise architects and software engineering teams struggle to cope with multiple scattered representations of data. Within the enterprise, we have data duplicated across multiple databases, IT applications, and business units. The software engineering context complicates this further by scattering business concepts across programming languages, modeling artifacts, and XML documents. This duplication of data across databases, artifacts, and languages increases overall IT costs, affects customer service, and increases maintenance effort. The main pain points are:

- The problem of semantic scatter (lack of integration between artifacts)
- Too much effort in impact analysis and system appreciation
- Poor knowledge management and reuse
- The challenge of knowledge transition across geographies
- Knowledge lost in employee turnover

Hidden or misunderstood relationships in the IT portfolio also lead to error-prone decision making. The following questions represent stakeholder concerns regarding an IT portfolio in a financial services organization:

- Which service returns the current balance of a Trading Account?
- What business processes rely on the historical price query service?
- What use cases in a specific IT application deal with Foreign Currency Accounts?
- How many customer-facing applications will be affected (directly or indirectly) if the payments server is down?

In the current scenario, these questions can be answered only by employees who are deeply involved in the design and implementation of the databases and applications in question. In order to support the proper scale and transition in an enterprise, we need an explicit representation of the implicit knowledge that is now restricted to a few experts.

Semantic technology can be employed to build an integrated knowledge repository for better insight into the IT portfolio. Multiple and diverse concerns and vocabularies from various stakeholders are represented using multiple ontologies that are bridged and reconciled. Process models, use case models, application code, and version history of software artifacts are primary data sources that are extracted into the RDF store using custom-built extractors to provide a multidimensional perspective (see Figure 8). This integrated repository supports querying across process definitions, entities, application code, and data. Some of the potential benefits are improved productivity and reduced cost of quality due to better visibility into the dependencies. The repository also delivers views at varying levels of abstraction for multiple stakeholders from the basic data by using declarative rules to precisely define the mapping between high-level concepts and detail data (see Figure 9).

CHALLENGES IN SEMANTIC TECHNOLOGY DEPLOYMENT

While SWT offers compelling benefits, there are a number of challenges in deploying semantic technology into the enterprise:

- **There is a mismatch between semantic technology and existing data technology.** The rules language within the semantic technology stack and SQL are both based on the abstract logic-based language Datalog. However, the complete integration of rules into the semantic stack is still a work in progress due to differences in primary assumptions between semantic technology and Datalog. SWTs, especially OWL and RDF, are primarily designed to operate at the Web scale without any central control. They presuppose incomplete distributed data that is not centrally controlled and therefore assumes no unique name and relies on "open-world" reasoning. Existing database technology operates in a complete and controlled environment, where it is safe to assume unique names for individuals and "closed-world" reasoning. Both paradigms provide the same results when facts are known and expressed in a knowledge base. However, when dealing with negated conditions, results differ. For example, consider the definition of a "childless couple" as a husband and wife who have no children. It is possible that the facts about children for some couples are not captured in the knowledge base. In this scenario, closed-world reasoning simply assumes that what is not mentioned in the knowledge base is not true, thus implying that all such couples are in fact childless. Presented with the same knowledge base, open-world reasoning concludes only that these couples *may* be childless.

- **The effectiveness of solutions depends largely on the quality of ontologies.** Semantic models involve some social agreement about the words used to describe concepts and relations in any domain. The translation of existing expertise and knowledge into machine-processable semantic models is an error-prone manual activity. Instead of humans devising semantic models, it is also possible to apply machine learning methods to "learn" concepts from existing data and documents. In either case, the quality of the semantic models will ultimately drive the value derived from information integration and search applications.

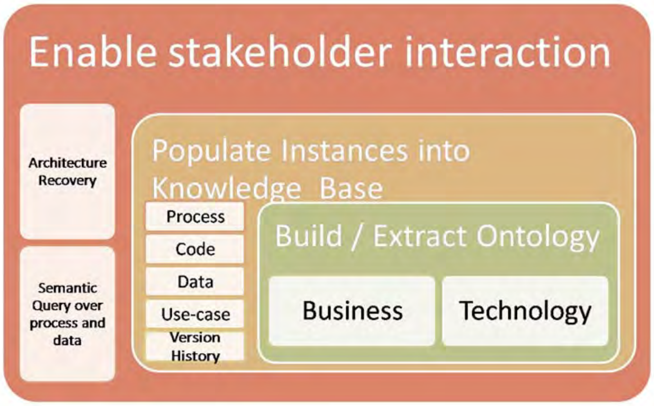

Enable stakeholder interaction
Architecture Recovery
Semantic Query over process and data
Populate Instances into Knowledge Base
Process
Code
Data
Use-case
Version History
Build / Extract Ontology
Business
Technology

Figure 8 — Integrated knowledge base for the IT portfolio.

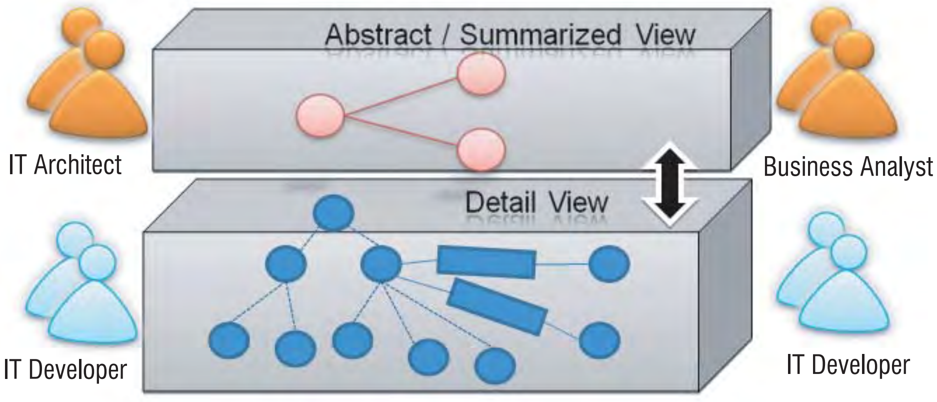


Figure 9 — The integrated knowledge repository uses declarative rules to define the mapping between high-level concepts and detail data.

- **Ontology creation in OWL presents a high barrier to entry due to the learning involved.** Semantic modeling involves understanding the various constructs of knowledge representation. The foundations of the modeling language constructs are borrowed from a subset of first-order predicate logic known as description logic.[11] This can be intimidating for modelers, so emerging semantic modeling tools hide the complexity by providing a graphical environment to model and maintain logical consistency. However, the task still demands some skill and initial learning about constructs of knowledge representation.
- **Integrating and aligning multiple ontologies is a nontrivial task.** An enterprise will require many ontologies to cover the various products, geographies, operating functions, units, and information categories. There will be some overlap between these ontologies, so discovering and expressing the precise overlap and mapping between various ontologies or semantic models within an enterprise is absolutely essential for deriving value from the technology.
- **User interaction with a knowledge base requires better visualization technology.** Enabling human-computer interaction through SWT-based knowledge repositories requires new techniques for visualizing data relationships. The ontology and the triple-based data store represent massive information graphs. The challenge here is to enable users to see what is of interest to them in terms of concepts and relations that express information at the right level of granularity.

CONCLUSION

SWT is ready for enterprise deployment. It has clearly moved out of academic and research contexts into actual industrial use. There is some literature[12, 13] that offers use cases and guidance for IT managers and architects. There are also reports from early adopters of this technology across all industry segments, ranging from pharmaceuticals to healthcare, banking, insurance, telecommunications, and retail.

I recommend starting small with a clear problem definition and set of use cases rather than attempting to go "big bang" with an enterprise-wide ontology modeling activity. This may require partnering with vendors to assess the applicability criteria, define the architecture and phased technology induction and training plan, provide tool support, and, finally, supply the implementation and maintenance services associated with SWT.

ENDNOTES

[1] Berners-Lee, Tim, James Hendler, and Ora Lassila. "The Semantic Web." *Scientific American*, May 2001.

[2] Brunnermeier, Smita B., and Sheila A. Martin. *Interoperability Cost Analysis of the US Automotive Supply Chain*. Prepared by Research Triangle Institute for National Institute for Standards and Technology (NIST), March 1999.

[3] Dublin Core Metadata Initiative (http://dublincore.org).

[4] "ISO/IEC 11179-1:2004: Information technology — Metadata Registries (MDR) — Part 1: Framework." International Organization for Standardization (ISO), 2004.

[5] Baader, Franz, Diego Calvanese, Deborah. L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider (eds). *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[6] Brachman, Ronald, and Hector Levesque. *Knowledge Representation and Reasoning*. Morgan Kaufmann, 2004.

[7] Sowa, John F. *Knowledge Representation: Logical, Philosophical and Computational Foundations*. Brooks/Cole Publishing, 1999.

[8] McGuinness, Deborah L. "Ontologies Come of Age." In *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential*, edited by Dieter Fensel, Jim Hendler, Henry Lieberman, and Wolfgang Wahlster. MIT Press, 2003.

[9] McGuinness, Deborah L., and Frank van Harmelen (eds). "The Web Ontology Language Overview: W3C Recommendation 10 February 2004." W3C, 2004 (www.w3.org/TR/owl-features).

[10] McBride, Brian. "The Resource Description Framework (RDF) and Its Vocabulary Description Language RDFS." In *Handbook on Ontologies: International Handbook on Information Systems*, edited by Steffen Staab and Rudi Studer. Springer, 2004.

[11] Baader et al. See 5.

[12] "Semantic Web Case Studies and Use Cases." W3C, 1994-2009 (www.w3.org/2001/sw/sweo/public/UseCases).

[13] Pollock, Jeffrey T. "Enterprise Semantic Web in Practice." Oracle, 2007 (http://me.jtpollock.us/pubs/2007.05-Pollock.STC.2007.pdf).

John Kuriakose is a Principal Architect with the Center for Knowledge-Driven Information Systems (CKDIS) labs at Infosys Technologies Ltd., India. The CKDIS labs is part of SETLabs, the applied research and development unit at Infosys, and it aims to create the basic reusable primitives that will help Infosys clients transform themselves from being data-centric enterprises to being knowledge-centric enterprises. Mr. Kuriakose has 17 years' experience in the IT industry, and his current research focuses on the application of semantic technology in creating next-generation, knowledge-driven IT solutions. He can be reached at john_kuriakose@infosys.com.