

MarkLogic Semantics: The Why and How of Linked Data

July 2014

Table of Contents

Introduction	1
Why the Semantic Web?	2
How Semantic Web Technology Works	3
MarkLogic Semantics	5
Semantics in Practice	7
Additional Resources	9
Appendix	10

The Semantic Web is a universal framework to describe and link data so that it can be better understood and searched holistically, allowing both people and computers to see and discover relationships in the data.

Introduction

The Age of Information and the associated growth of the World Wide Web has brought with it a new problem: *how to actually make sense of all the information available*. The overarching goal of the Semantic Web is to change that.

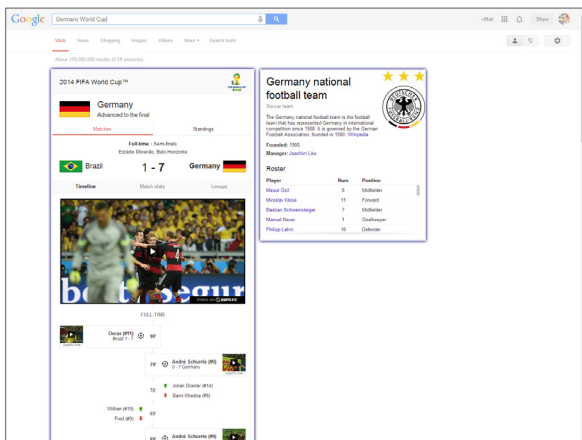
Semantic Web technologies accomplish this goal by providing a universal framework to describe and link data so that it can be better understood and searched holistically, allowing both people and computers to see and discover relationships in the data.

This “Linked Data,” as it is commonly known, is written as triples—the combination of a subject, predicate, and an object. Linked together, these triples form a graph-like representation of data without hierarchy, are machine readable, and can be used to infer new facts about the world. The standard language for writing triples is RDF (Resource Description Framework), and the standard query language is SPARQL (pronounced “sparkle”).¹

MarkLogic is an Enterprise NoSQL database platform that in addition to being a documents and data store, is also a native RDF triple store—providing the unique ability to store and query documents, data, and triples all in the same database. And, MarkLogic has the enterprise features organizations need—including ACID transactions, scalability and elasticity, and government-grade security.

Today, organizations are using MarkLogic Semantics to aggregate and link disparate data sources, create sophisticated search applications, dynamically publish content, and perform more efficient ETL processes.

Understanding Semantics With Google



Google uses Linked Data to automatically provide “rich snippets” of information based on RDF markup in Web pages.

For example, a search for “Germany World Cup” shows top-level results driven by semantics. The data resides on other Web pages, but is readable by Google—including metadata about the team, live match results, and video content.

¹ Triple stores can form graph-like representations, but are not graph databases. Refer to the Appendix for more information about the similarities and differences.

Why the Semantic Web?

Semantics is an evolution of the Web. It provides a standard format for Linked Data that is not too different from the once revolutionary idea of using HTML and HTTP standards to link documents on the Web. The Semantic Web goes further, by providing a universal framework to describe and link *data*, not just documents. This framework for Linked Data solves a number of existing problems by allowing data to be understood holistically, in the context of relationships with other data. MarkLogic allows you to capitalize on the structure of Linked Data.

Existing Problems

- **Problem: The Web is built to link documents, but not data.**
The Web is a network of HTML documents linked together using HTTP. With this simple framework, the Web has unleashed information like nothing else before. But, information is locked in the Web pages where it was published. And, the confusion is compounded by the sheer increase in data volumes. For this reason, a Google search can deliver millions of results and yet still fail to answer the question asked.
- **Problem: There is no context for understanding data.**
Consider the word “cook”—the computer does not know whether you mean a chef, the act of cooking, or the Cook Islands. And, even if the computer did know that you meant a chef, it would not know that you would also be interested in the particular restaurants that the chef works at in a particular city.
- **Problem: Applications create walled gardens within organizations.**
Applications have historically been built on relational databases with a specific use in mind, creating walled gardens of data that prevent the data from being used for anything beyond its original design. For example, just imagine the difficult task of trying to mashup data from bank statements, mobile phone usage, weather data, and a friend list from Facebook. Similar examples get replayed again and again within organizations around the world.

Semantic Web As Solution

- **Solution: Linking data using a universal standard.** Using RDF as a standard to link data creates a structure that allows discovery of facts that can be universally understood. This means that an application can communicate with another application without a human middle-man. A perfect example is the Google search that returns top-level facts that the user wants to know rather than a list of links to documents.
- **Solution: Linking data within ontologies.**
Semantic ontologies provide context. Ontologies—collections, categories, hierarchies, or taxonomies—relate data by defining different classes for events, people, or things. For example, consider a classification such as plants, which has sub-classifications such as flowers and shrubs. Then consider a “rose” in this context, which means the flower, and not the actress Rose Byrne. In addition to helping build better navigation and search experiences, ontologies are also helpful in publishing more relevant content and making sense of metadata.
- **Solution: Linking data together to be searched holistically.**
Semantics is predicated on the relationships *between* data, which makes it an ideal tool to link and search across both structured and unstructured data by using its standard query language, SPARQL. This is particularly useful in creating sophisticated queries that span multiple data sets. An example would be, “Provide all of the health insurance beneficiaries that earned over \$100,000 and lived in Atlanta, GA in the year 2010”—combining data about insurance, income, geography, and time.

How Semantic Web Technology Works

Traditionally, computers have had a very difficult time understanding context and meaning. Unlike computers, our brains associate different thoughts in order to come up with the “full picture” of something, whether it’s an abstract concept or a concrete inference. But, computers have a lot more difficulty connecting the dots, particularly with information that is not so structured. The Semantic Web addresses this challenge, using a standard for Linked Data that makes computers smarter.


Understanding Linked Data

Linked Data is written as *triples*—the combination of a subject, predicate, and an object—and linked together, these structured triples form a graph-like representation without hierarchy, are machine readable, and can be used to infer new facts about the world. The standard language for writing triples is RDF, and the standard query language is SPARQL.

DATA is stored in Triples, expressed as:

Subject	: Predicate	: Object
John Smith	livesIn	London
London	isIn	England

QUERY with SPARQL, gives us a simple look up... and more!
Find people who live in (a place that's in) England



Based on W3C standards, triples are written using RDF (Resource Description Framework), and queried using SPARQL.

RDF and SPARQL Standards

RDF (Resource Description Framework) is the common data format for Linked Data, and using this RDF standard liberates data from the containers it comes in, making it available for more automated processes. The international standards organization W3C recommends RDF, and has been guiding the standards for RDF since 2004. RDF is based on using HTTP URIs to lookup and describe resources.

Example of RDF

```
<http://example.org/dir/js> <http://xmlns.com/foaf/0.1/name> "John Smith" .
<http://example.org/dir/js> <http://xmlns.com/foaf/0.1/livesIn> "England" .
```

W3C also defines SPARQL as a standard query language for RDF. SPARQL was first defined as the standard semantics query language in 2008, and according to W3C Director, Tim Berners-Lee, “Trying to use the Semantic Web without SPARQL is like trying to use a relational database without SQL.”²

Example of SPARQL

```
SELECT ?person ?place
WHERE
{
  ?person <http://example.org/LivesIn> ?place .
  ?place <http://example.org/IsIn> "England" .
}
```

² For more information, visit the MarkLogic Developer website: developer.marklogic.com/learn/semantics-exercises/sparql-101.

RDF Provides Context

Using RDF at scale provides the ability to link disparate data sources in context—context from documents and data, the domain data lives in, and the world at large.

When these data sources are leveraged together, organizations have a framework in which to better understand their data. It is a framework that can be continually built-on, and just as the World Wide Web of documents grows, so does the World Wide Web of data. The Semantic Web continues to expand at an exponential rate as governments and organizations choose to store their data as RDF and contribute datasets to the world of Linked Open Data.

Below are examples of data types in “context”:

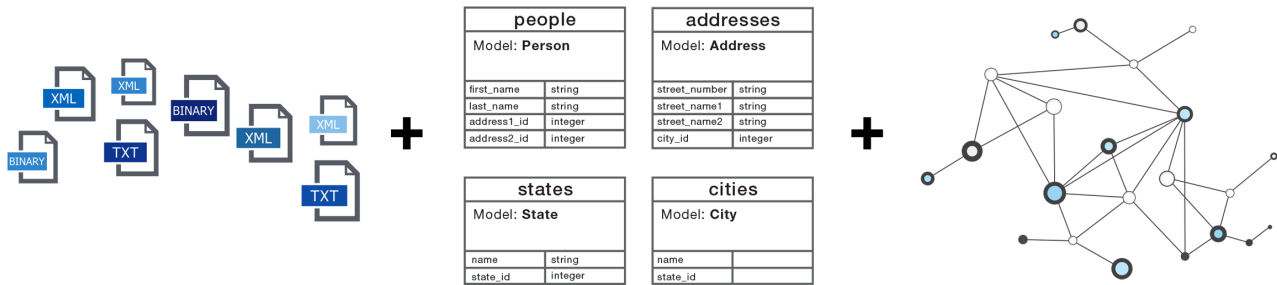
Context	Data Type
Documents and Data	Structured and unstructured data within an organization: <ul style="list-style-type: none">• XML and JSON documents• Free text with <i>entities</i> (Proper Nouns, e.g., the person Richard Nixon, the product Advil, the company IBM), and <i>events</i> (Nixon went to China, IBM acquired Cloudata)• Document metadata (categories, author, publish date, source)
Specific Domains	Shared data specific to an industry or organizations: <ul style="list-style-type: none">• A pharmaceutical company's drug ontology• SNOMED CT clinical healthcare terminology• Dublin Core Metadata Initiative for media and publishing resources• FIBO (Financial Industry Business Ontology)
World at Large	Billions of facts about the world at large that are often shared freely: <ul style="list-style-type: none">• DBpedia – Wikipedia as semantics data. Includes facts such as “Einstein was born in Germany” or “Ireland’s currency is the Euro.” DBpedia contains almost 2.5 Billion pieces of information stored as RDF triples and is growing rapidly.³• GeoNames – Geographical data such as “Doha is the capital of Qatar” and Doha is located at 25.2867° N, 51.5333° E)• Linked Open Data – Thousands of freely available data sets all interconnected.⁴

³ More Information is available on DBpedia at <http://wiki.dbpedia.org/About>.

⁴ The interactive Linked Open Data Cloud Diagram is available at <http://lod-cloud.net/>.

MarkLogic Semantics

To simplify data storage and provide one unified, easy-to-use solution, MarkLogic includes a native RDF Triple Store that can be queried with SPARQL—all right inside MarkLogic.⁵



Documents + Data + RDF

MarkLogic is the only Enterprise NoSQL database platform that can store and query a combination of documents, data, and triples. With a single platform, users have flexibility in choosing the data model that works best to store their data, and provides the ability to query across everything holistically.

Holistic Search

Triples can be embedded in documents, triples can refer to documents, or triples can connect documents. In any case, users can search across the data with a single query. In the real-world, having this flexibility is critical. For example, on Data.gov.uk, there are 10 different primary data formats, including documents, XML, CSV, and RDF. But, only one percent of the data is in RDF. Only MarkLogic can store and query the XML, CSV, and RDF data together *within the same database*.

Speed and Scale

MarkLogic has a specialized triple index to ensure querying triples is fast. MarkLogic also has a triple cache to better manage the use of memory to ensure optimal performance at scale. Some triple stores insist you store the whole triple store index in-memory, but MarkLogic uses memory-mapped index files to maintain speed without the limitations of physical memory.

Both of these features—the specialized triple store and the triple cache—make MarkLogic a scalable, elastic, high performance triple store. With other triple stores, volume quickly becomes an issue. Some triple stores claim to scale in clustered systems, but for parallel query only—that is, they can have three node clusters but only if each node has the same data on it. MarkLogic’s shared-nothing architecture supports elasticity and scalability.

MarkLogic can store 1 billion triples per node, at about 350 bytes per triple, and can scale to hundreds of billions of triples.

⁵ Refer to the Appendix for a full list of MarkLogic Semantics features.

Enterprise Features

MarkLogic's Triple Store comes with all of the features MarkLogic has built and proven over the past decade—including ACID transactions, scalability and elasticity, high availability and disaster recovery, government-grade security, and performance monitoring tools. With semantics in particular, government-grade security gives users the ability to define exactly which users are able to see which triples by using Role Based Access Control (RBAC).

Complementary Semantic Technologies

Organizations often use MarkLogic as the database platform to store and search RDF triples, while using additional technologies for ontology management, text analysis, semantic enrichment, and creation of triples.



Semaphore, Smartlogic's content intelligence software platform, enables the rapid development of ontologies and performs semantic analysis to describe and extract information from content of all types and sources. Semaphore compliments MarkLogic's capabilities by automatically applying consistent tags as content is loaded, standardizing metadata, performing fact and entity extraction, and enhancing MarkLogic's search capabilities with ontologies.

Smartlogic is an official MarkLogic technology partner. Find more information at <http://www.smartlogic.com>.



Temis' flagship platform, Luxid[®], identifies and extracts targeted information to semantically enrich content with domain-specific metadata. The integration of Luxid[®] and MarkLogic delivers seamless semantic enrichment of data stored in MarkLogic with the Luxid[®] domain-specific and multilingual annotation process. This enables organizations to build powerful, scalable applications—combining semantics with real-time database agility to make massive volumes of unstructured content easier to exploit.

Temis an official MarkLogic technology partner. Find more information at <http://www.temis.com>.



Protégé is a free, open-source platform that provides a suite of tools to construct domain models and knowledge-based applications with ontologies. Protégé is focused on ontologies within the medical industry, and is managed by the Stanford Center for Biomedical Informatics Research. Healthcare organizations use MarkLogic as their RDF triple store and Protégé for ontology management.

More information at <http://protege.stanford.edu/>.

This list by no means all inclusive. It highlights some complementary technologies that are currently in use, but there are many other technologies that would benefit from using MarkLogic as a database platform to store and search across documents, data, and triples.

Semantics in Practice

MarkLogic Semantics Use Cases

The below examples describe some ways in which organizations use MarkLogic Semantics today:

- **Aggregate and link disparate data sources** – Use MarkLogic to store documents, data, and triples—all in the same database. Store your data natively as RDF triples, or go further by using triples as the glue to link disparate data sources. You can also use triples to “decorate” existing data with annotations that describe it
- **Improve search navigation** – Improve search navigation by providing context through the use of ontologies. Even if the data is not natively stored as RDF, it can be semantically enriched so that a user can discover facts and information more intuitively. MarkLogic partners with organizations such as Smartlogic that manage ontologies
- **Provide holistic search and discovery** – Use MarkLogic to create combination queries that span documents, data, and triples. Enable granular searches, such as, “everything an analyst said about a company in relation to an employee that worked there at a specific time”
- **Dynamically publish content** – Use MarkLogic as the database to enable automated publishing of content in real-time (e.g., event data such as scores, team profiles, news articles, etc.) based on semantic relationships that helps re-purpose interrelated content objects according to an ontological domain-modeled information architecture
- **Complete ETL processes faster** – Use RDF to map data for faster ingestion. Map any number of physical representations of a data element across various ETL source and targets to a common semantic definition

Solutions Across Industries



Publishing & Media

Implement Dynamic Semantic Publishing to automatically publish content across thousands of Web pages, as the BBC does on bbc.com/sport/



Financial Services

Use semantics to support fraud detection, pre-trade analytics and decision support, regulatory compliance, data provenance, KYC, and reference data management⁶



Government

Intelligence, law enforcement, and fraud investigators and analysts use semantics to discover connections and patterns in facts and documents



Enterprise

Procurement agents are creating decision support tools to rationalize purchasing decisions, vendors, and bid management



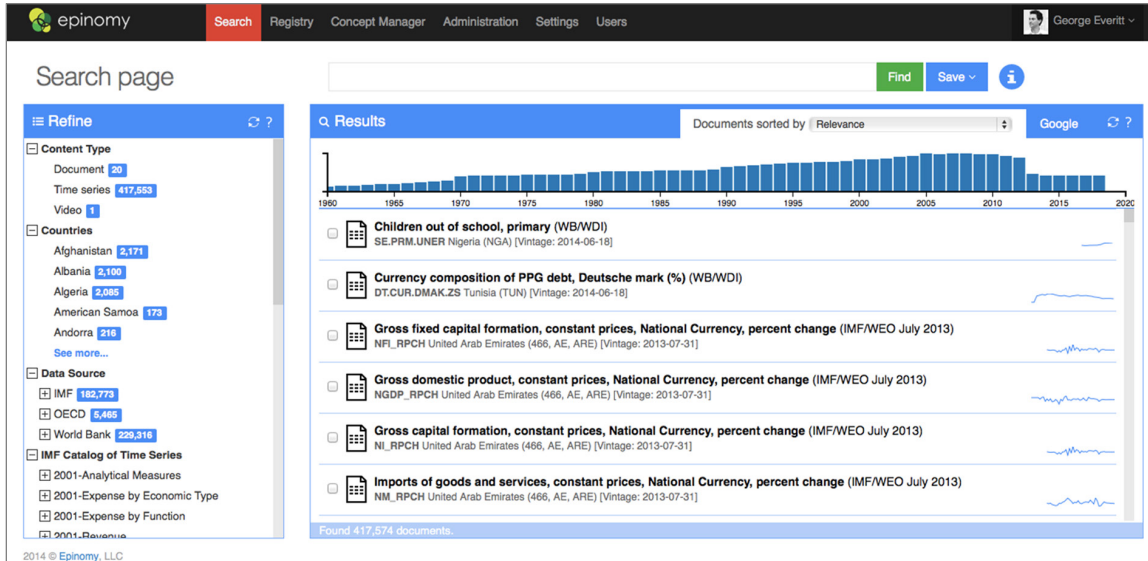
Healthcare

Pharmaceutical companies and governing organizations are using facts and documents together to assess risks and to decide which drug trials to invest in

⁶ To learn more, watch the Webinar recording on MarkLogic’s website, [Semantics in Financial Services](#).

Production Example

Applied Relevance, Epinomy



Applied Relevance created an application on MarkLogic called Epinomy, a time series search engine that combines the best full-text search engine and business analytics for time series data. Time series data is the accumulation of measurements taken at successive points in time spaced at uniform time intervals, and is the most common form of structured data.

The challenge Epinomy has addressed is figuring how to combine time series data with other unstructured and constantly changing data such as global economic indicator data. For example, the World Bank publishes data for poverty, inflation, and GDP in a format called SKOS SDMX Data Cube format, a triples format for tracking economic indicators and doing statistical analysis. But, there is lots of other economic data that is not already formatted for easy analysis. With relational databases, this challenge is difficult and even impossible to solve but with MarkLogic Semantics, new data can be incorporated in days, not months.

Consider the difficulty in trying to search across various data sources for a common term such as “Euro zone.” It means something different from “European Union”, “Europe OECD”, or “Europe.” Or what about a term such as “Small States,” which is different from “Least Developed Countries,” “Lower Middle Income,” or “Low & Middle Income.” Semantics provides the ability to map all of these terms so that a user can perform natural language searches.

Semantics also allows the application to quickly create facets without pre-defining what they should be. Facets, or the categories of results typically grouped down a left-hand column on a webpage, are created in Epinomy completely using triples. It happens dynamically on the fly, is dependent on the content loaded, and is presented fast to the user.

Another challenge is when the same economic data is released multiple times. These multiple “vintages” of the same data would typically be a headache to deal with. Semantics handles the various vintages of data by simply creating new sets of triples tagged as “vintage.” And, the natural language search was also designed so that a search can specifically return those vintage values.

To learn more about Applied Relevance’s Epinomy and how they used MarkLogic, watch the presentation, [A Field Guide to MarkLogic Semantics](#).

Additional Resources

MarkLogic provides numerous resources to get you started with semantics. Please visit marklogic.com or contact us at sales@marklogic.com.

Resources

Presentation: Overview of MarkLogic Semantics

marklogic.com/resources/marklogic-semantics-mlw14/

Presentation: A Field Guide to MarkLogic Semantics

marklogic.com/resources/field-guide-marklogic-semantics/

Presentation: MarkLogic Semantics - Under the Hood

marklogic.com/resources/marklogic-semantics-hood/

Semantics Developer's Guide

marklogic.com/guide/semantics

Appendix

RDF Triple Stores versus Graph Databases

A frequently asked question is how RDF triple stores are different from graph databases. There are many similarities, and when looking at a network or “link chart” style of data visualization, it is often impossible to know what type of database is even being used because they can produce similar looking end user experiences.

To summarize, both graph databases and triple stores are designed to store Linked Data. RDF is a specific kind of Linked Data that is queried using SPARQL, so it is fair to say that RDF triple stores are a kind of graph database. But, there are some subtle but important differences that are described below.

How They Are Similar

- Graph databases and RDF triple stores focus on the relationships between the data. Data points are called nodes, and the relationship between data points are called edges
- A web of nodes and edges can be put together into interesting visualizations—a defining characteristic of graph databases and triple stores

How They Are Different

- RDF and SPARQL are W3C standards whereas graph databases use ad hoc standards that are in development. One graph database, Neo4J, stores RDF triples and uses SPARQL but generally focuses on its proprietary language, Cypher. Other graph databases support G, GraphLog, GOOD, SoSQL, BiQL, SNQL, and more.
- RDF triple stores focus solely on storing rows of RDF triples and although they can act like property graphs, graph databases can manage a wide variety of graphs, including undirected graphs, weighted graphs, hypergraphs, etc.
- Graph databases are node, or property, centric whereas RDF triple stores are edge-centric. RDF triple stores are really just a list of graph edges, many of which are 'properties' of a node and not critical to the graph structure
- RDF triple stores provide inferences on data and are optimized for aggregate queries whereas graph databases do not do inferencing (e.g., an inference would be, “If John lives in London, and London is in England, then John lives in England”) but are better optimized for graph traversals (degrees of separation or shortest path algorithms)
- RDF triple stores are more synonymous with the “Semantic Web” and the standardized universe of knowledge being stored as RDF triples on DBpedia and other sources, whereas graph databases are seen as less universal and more purpose-built for specific applications

MarkLogic Semantics Features

MarkLogic 7 Semantics

- Store hundreds of billions of triples
- Query across documents, data, and triples
- Triple index for sub second search results
- Triple cache for high performance across large clusters
- Bulk-load of triples via MarkLogic Content Pump
- Provenance and reification by adding metadata
- SPARQL 1.0+ over REST or XQuery
- Search with arguments, variable bindings, or forest-ids
- XQuery helper modules for serializations, transitive closures
- Updates, aggregates via MarkLogic APIs
- Semantic Enrichment with partners (Temis, SmartLogic, NetOwl, TSO)
- Enterprise Features: ACID transactions, scalability and elasticity, HA/DR, government-grade security, monitoring and performance tools

MarkLogic 8 Semantics

Everything in MarkLogic 7 Semantics, plus:

- SPARQL 1.1
- Graph traversal and discovery
- Automatic inference
- Basic visualizations
- SPARQL from JavaScript, Node.js



About MarkLogic

For more than a decade, MarkLogic has delivered a powerful, agile, and trusted Enterprise NoSQL database platform that enables organizations to turn all data into valuable and actionable information. Organizations around the world rely on MarkLogic's enterprise-grade technology to power the new generation of information applications. MarkLogic is headquartered in Silicon Valley with offices in Washington D.C., New York, Chicago, London, Frankfurt, Utrecht, and Tokyo. For more information, please visit www.marklogic.com.

© 2014 MarkLogic Corporation. All rights reserved. This technology is protected by U.S. Patent No. 7,127,469B2, U.S. Patent No. 7,171,404B2, U.S. Patent No. 7,756,858 B2, and U.S. Patent No 7,962,474 B2. MarkLogic is a trademark or registered trademark of MarkLogic Corporation in the United States and/or other countries. All other trademarks mentioned are the property of their respective owners. [SS-MLIH-13-06]

999 Skyway Road, Suite 200, San Carlos, CA 94070 ›US: +1 650 655 2300 ›INT'L: +1 877 992 8885
sales@marklogic.com › www.marklogic.com