Over the years, Ontotext's leading semantic graph database GraphDB has helped organizations in a variety of industries with their data and knowledge management challenges. Enterprises and organizations in Healthcare, the Financial Services sector and Media & Publishing, as well as Government agencies, already use GraphDB to make sense out of diverse sets of data and get deeper insights.

Along with building up a stable customer base of knowledge graph users in various industries throughout the years, GraphDB has also generated a thriving community. Experts in linked data, semantic technology and triplestores are testing GraphDB's capabilities and performance, and applying it in various technical use cases. They share their insights and experience in numerous blog posts and tutorials, which continue to contribute to the growing community.

Comparing GraphDB to Other Graph Databases

In his post, **Angus Addlesee, Machine Learning Engineer and Researcher at Wallscope**, compares the performance of several linked data triplestores, using RDF and SPARQL queries he has created. Addlesee deployed several triplestores on the same server, one at a time, with the default settings where possible.

Exploring each triplestore's performance for six different queries, the expert found that Ontotext's GraphDB was the fastest triplestore out of the five tested to complete the simplest queries. GraphDB took just 16.7ms to complete two simple queries. In several more complex queries, GraphDB was also the fastest at retrieving entities.

> Overall, GraphDB is very easy to deploy and load data, has a pretty interface, and loading data is very intuitive, the expert says.

GraphDB allows users to visually explore data, which makes query construction faster and easier. According to Addlesee's observations, users can also see instantly if there is a spelling error somewhere if the query is not working.

> I really appreciate the ability to explore my data during development and

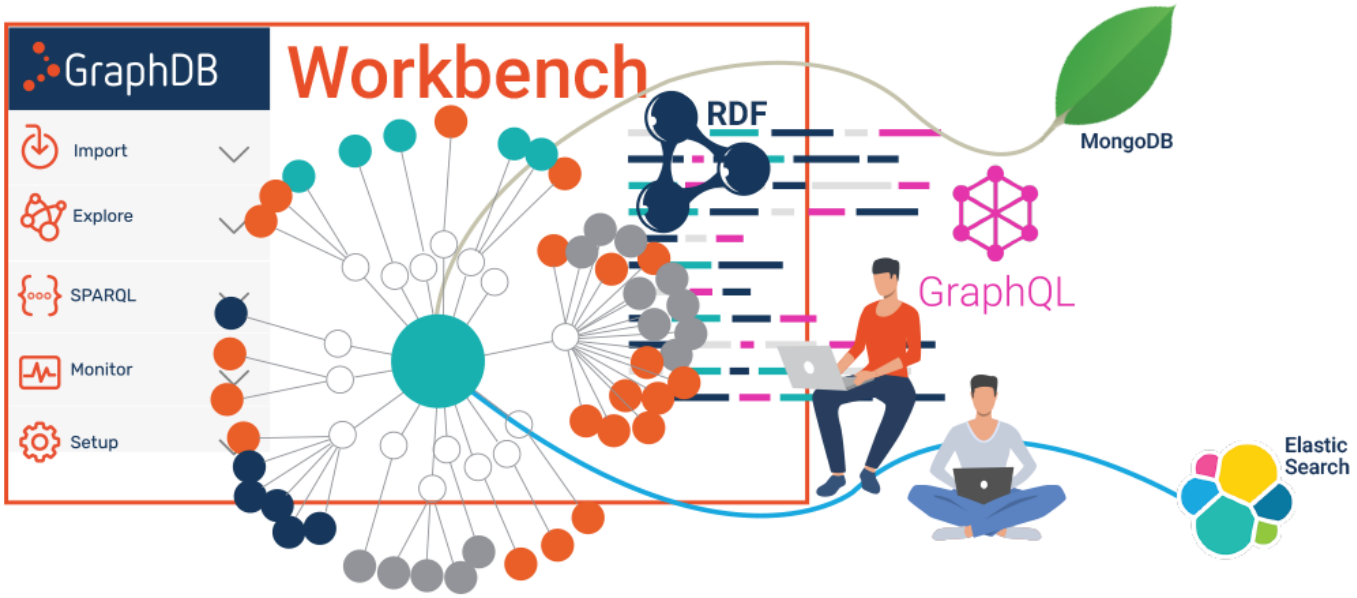
Copyright: Ontotext New York | Sofia | Basel
Reach out to us: info@ontotext.com

ontotext

GraphDB is by far the winner for this, Addlesee writes.

The Handshake Between GraphQL & GraphDB's RDF Model

Paul Wilton, Managing Director at technical consultancy Data Language, has written a post discussing the move of graph database vendors, including Ontotext, to start offering GraphQL – a popular graph query language that developers use, know and like.



Ontotext Platform (Ontotext's knowledge graph platform for building enterprise solutions), for example, uses GraphQL to lower the barrier of entry to knowledge graph data and at the same time provides the richness and expressivity of SPARQL.

> Ontotext has taken the next logical step and created a 'Semantic Object Service' that automatically generates GraphQL schemas bound to the underlying RDF data model in GraphDB, moving us towards the ontology ubiquity I described above, where your domain ontologies are expressed in the knowledge graph, and across your wider data-estate, including your GraphQL schemas. Nice work, Wilton writes.

Copyright: Ontotext New York | Sofia | Basel
Reach out to us: info@ontotext.com

Wilton also observes that Ontotext’s GraphDB is fully compliant with the RDF/SPARQL standard and has highly compliant and configurable OWL / RDFS forward-chaining reasoning, which materializes inferred knowledge at write time. Moreover, GraphDB supports unparallelled full-text-search capability using Lucene, or via bi-directional connectors to external SOLR or ElasticSearch clusters. The graph database can also materialize post-inference fully structured JSON or JSON-LD objects into ElasticSearch indices or MongoDB and supports geospatial querying and other extensions. Last but not least, it has a rich visual query and management workbench UI.

> One killer feature of GraphDB, which other graph databases lack, is the post-inference connectors to ElasticSearch and MongoDB which unlock some outstanding technical and data architecture patterns out of the box, Wilton says.

Unlocking The Power of GraphDB in Tutorials

Developers are also testing various use cases to get the most out of their data using GraphDB.

Copyright: Ontotext New York | Sofia | Basel
Reach out to us: info@ontotext.com

How to Transform Tabular Data into Linked Data with GraphDB's OntoRefine

Angus Addlesee has written a brilliant how-to post introducing readers to a quick tutorial on how to transform tabular data into linked data using GraphDB's OntoRefine. OntoRefine (based on OpenRefine) is integrated in GraphDB Workbench and enables users to easily filter their data, edit inconsistencies, convert the data into RDF, and import it into a repository. This post is a step-by-step tutorial – from getting started to cleaning your data to constructing a knowledge graph and loading your data.

How to Use GraphDB for Data Reconciliation

Taking the suggestion of Ontotext's CEO Atanas Kiryakov, Addlesee has written another tutorial to get users and developers started on how to use GraphDB for data reconciliation. In his easy-to-follow way, he explains how to set up and load tabular data in OntoRefine, how to construct the graph, reconcile the data and explore the new graph. The tutorial is an example of how to use DBpedia to enrich the data in GraphDB.

Copyright: Ontotext New York | Sofia | Basel
Reach out to us: info@ontotext.com

ontotext

> We started with a small list of car manufacturer names but, by using GraphDB and DBpedia, we managed to extend this into a small graph that we could gain actual insight from, Addlesee writes.

How to Set Up a GraphDB Cluster

Another valuable resource is this guide by **Ellis Pritchard, Editor of Agrimetrics Dev Yard**, on how to set up a GraphDB cluster.

> Setting up a cluster in GraphDB is pretty simple if one follows five easy steps to create, connect and configure repositories, Pritchard says.

How to Use GraphDB to Power an Angular App

Elvin Dechesne, a Senior Solution Architect with a passion for AI and knowledge graphs, has also written a tutorial in two parts about how front-end developers can use GraphDB to power an Angular app. (See Part One and Part Two). It explains how to get started, what OntoRefine is and how to use it, how to clean up data and how to transform data.

> GraphDB has a cool feature which enables you to visualize all connections, says Dechesne.

How to Query Geospatial Data in GraphDB

GeoSPARQL is a standardized extension for #SPARQL, which specifies how to make queries about spatial information (i.e. ask about distances in miles or kilometeres, while GeoSPARQL does the necessary trigonometry to provide the right answers). The second part of a tutorial by **Bob DuCharme, a technical writer at CCRi** – a machine learning and geospatial research company, looks into how GraphDB handles GeoSPARQL distance over Open Street Map data. (See Part One)

Copyright: Ontotext New York | Sofia | Basel
Reach out to us: info@ontotext.com

ontotext

> […] I tried several triplestores and data sources and never quite got there. When I tried it recently with Ontotext's free version of GraphDB, it all turned out to be quite easy, says Bob DuCharme.

Last But Not Least

The latest major release of Ontotext's graph database, GraphDB 9.0 has introduced another cool feature – it has **open-sourced its front-end and engine plugins**. In this way, Ontotext aims to foster developers' collaboration even more as well as to meet customer demands for bespoke data management and analytics tools.

The open-source plugins and workbench allow users to take full control in creating prototypes and solutions. They can now fully enjoy increased transparency, stability and security while solving their data and knowledge management challenges.

Ready to try it yourself?

Copyright: Ontotext New York | Sofia | Basel
Reach out to us: info@ontotext.com