Technical Analyst Kit

COMPLEXIBLE

Bloor

MarketUpdate

Graph and RDF databases 2015

Market basics

Graph databases represent a significant growth area. Indeed, research suggests that it is the fastest growing segment of the database market. There are arguably three reasons for this growing interest and each pertains to a particular sub-sector of the graph market. The first is that graph databases are designed to handle many-to-many relationships, which relational databases are not, so graph products have particular advantages in operational and transactional environments where this is the case. The second is with respect to semantics, which is of growing importance in its own right, and which graphs handle with ease. The third is that many graph analytic algorithms require a significant degree of iteration, which is not available with MapReduce and, where many-to-many relationships are involved, are not easily parallelised in conventional warehousing environments.

Not surprisingly, as an emerging market there are vendors coming at this space from a variety of different directions: with purpose-built native graph stores or with graph implementations on top of other types of database or file store, and with products that are aimed at different functional requirements.

As a result the graph database market is not homogeneous. Previous attempts to classify the market have distinguished between "graph databases" on the one hand and "graph compute engines" on the other, where the former tend to be more operationally focused and the latter are targeted at data warehousing and analytics. However, we believe that this is too simplistic and think that it is appropriate to distinguish between RDF (resource description framework) databases, which are often targeted at semantic applications or environments that involve semantics, and graph databases, which are less semantically oriented. Further, when the original two-tier distinction was proposed (by Neo4j) there were no graph databases per se in the data warehousing space. That is no longer true, so we need a new description. We have opted for RDF databases, operational graph databases and analytic graph databases. There is, of course, overlap between these categories, as explained in the following brief descriptions:

- **RDF databases.** Often semantically focused. Often, but not always, based on non-graph underpinnings (including relational databases). For use in operational environments but have inferencing capabilities. Require indexes even in transactional environments. Often ACID compliance.
- **Operational graph databases.** Tend to be native graph stores or built on top of a NoSQL platform. Focused at transactions (ACID) and operational analytics. No absolute requirement for indexes though these will typically be offered in order to improve query performance.
- **Analytic graph databases.** Some vendors focus on solving "known knowns" problems (the majority) where both entities and relationships are known, while others are more focused on known unknowns and even unknown unknowns. Multiple approaches characterise this area with different architectures including both native and non-native stores, different approaches to parallelisation, and the use of advanced algebra.

Figure 1: The highest scoring companies are nearest the centre. The analyst then defines a benchmark score for a domain leading company from their overall ratings and all those above that are in the champions segment. Those that remain are placed in the Innovator segment if their innovation rating is over 2.5 and Challenger if it is less than 2.5. The exact position in each segment is calculated based on their combined innovation and overall score.

A further distinguishing factor is in the languages supported by the different vendors. Most vendors support SPARQL (SPARQL Protocol and RDF Query Language), which is a W3C standard declarative language, but users often prefer to employ other options.

There are three other declarative languages available from different vendors, including one that is an extended form of SQL. Most graph products also support traditional languages such as Java while there are also specialised graph traversal languages such as Gremlin. A number of vendors with triple stores targeted at semantic processing support OWL (Web Ontology Language – so named because Owl in Winnie the Pooh misspelled his name as WOL).

For a detailed discussion of the types and architectures and uses of graph products see the Bloor Research Spotlight paper: ***"All about graphs: a primer"***.

Market trends

The biggest trend is simply towards graph products in general, as more and more products appear on the market. In particular, more of the major vendors are getting involved in this market. IBM, for example, has triple store support in DB2, has a graph database known as System G in R&D, and is embedding third party graph databases (at least two) into forthcoming products. Informatica and others are also embedding graph databases. Microsoft too has been researching graph technology. Similarly, MarkLogic is touting its triple store capabilities. Teradata has been offering graph analytics based on its Aster Data platform for some time, Oracle has been similarly active both for semantics and analytics, and SAP has introduced a graph engine into HANA. A corollary to this burgeoning interest in graphs is that there is not enough space in the market for all the vendors that are in it. Over time we can expect many of the current incumbents to either be acquired or to disappear altogether.

A secondary trend, which may be more of a marketing concept than a technical reality (depending on the vendor), is for RDF databases to make themselves over as operational graph databases. That is, the suppliers of these products are trying to expand beyond the confines of semantic processing to offer more general purpose operational graph processing, for example, by adding support for property graphs. Precisely what sort of operational graph applications these will be suitable for will depend on the product. For example, some products only support eventual consistency, which will not be suitable for certain types of transactional applications.

The market is split three ways in terms of SPARQL support: true believers, vendors (usually major ones) that support it because they think should but don't really care, and those that positively think that there are better alternatives. In practice, both KEL (Knowledge Engineering Language from Lexis Nexis) and Cypher (Neo4j) are more advanced in terms of functionality and performance, though that does not mean that these are without flaws either. Moreover, while both of these are open source they only work with their respective databases and until and if either of these is made to work more widely, then SPARQL will dominate the field for declarative languages. At present there is still scope for competition to arise because SPARQL is quite limited. You cannot, for example, add a time stamp to a relationship because this requires predicate attributes, which are not supported (though you can define relevant additional triples, which means that the graph expands). Recommendations have been made to the W3C that this be incorporated in the next version of this standard. If these sorts of additions can be added sooner rather than later then SPARQL will come to dominate this space.

Finally, it is worth commenting on the use of graph algorithms in analytics. Some vendors, such as Teradata, Franz and Oracle (through Parallel Graph Analytics: PGX) offer pre-built algorithms for analytics such as page ranking, finding shortest path, predicting future edges and so on. However, most suppliers rely on TinkerPop, which is a developer group working on an open source stack for graphs. Among its offerings are Gremlin, the Blueprints API and Furnace. The last of these is a graph algorithms package though it is only suitable for property graphs (that is, graphs that allow properties to be associated with the vertices and edges of the graph). There are around a hundred known graph algorithms but only a relative handful are available in a pre-built fashion at present. We expect this number to grow significantly.

Vendors

We have not attempted to analyse every graph or RDF product on the market, not least because there are so many of them. Those that are covered here have been included based on our own judgement and based on recommendations from Bloor Research subscribers. Because it is important to understand the positioning of the various offerings and their focus, the following provides a brief outline of those vendors/products that are included here.

Complexible

Complexible Inc. (which used to be known as Clark & Parsia after the founders of the company) are the developers of Stardog. Stardog is (currently) an RDF database with strong support for SPARQL and OWL (it supports all of OWL 2) and the company has embedded the Lucene search engine into Stardog. The database is ACID compliant and supports two-phase commit. A focus is on (model-driven) integration and analytics. The database uses query time reasoning that does not require the materialisation of inferences. It has a built-in optimiser for SPARQL. A major feature is that it provides graph versioning so that you can track changes to a graph, both for auditing and analysis purposes. In the company's forthcoming 3.1 release Stardog will be adding property graph and graph traversal capabilities along with support for TinkerPop and Gremlin, as well as graphing algorithms.

Comments

As a general principle we prefer native implementations to ones based on other database platforms. Moreover, in our view, NoSQL implementations are preferable to relational ones. However, both of these statements are dependent on the application. If you are exploring known-knowns in a purely analytic environment then storing edges and vertices in relational tables should provide perfectly acceptable performance. In other environments, for example when supporting transactional and operational processing, we would expect relational products in particular to perform poorly compared to native and even NoSQL-based implementations.

With respect to the various products shown on the following Bullseye Chart, we have sometimes used product names and sometimes vendor names. In general we have used the name with which we believe readers will be most familiar. It should be noted that SPARQL Server from Algebraix is currently in beta and the final release version may differ from that which is currently available. Titan, similarly, has yet to reach 1.0 status. The graph engine in SAP HANA is also new and, as yet, relatively unproven, although it shows promise. As will be seen the various product/vendors are colour coded so that we are comparing apples with apples. Even so, the following additional comments are relevant:

- **RDF databases –** both Oracle and Stardog are placed in this category despite the fact that Oracle provides analytics and Stardog will shortly be introducing graph database capabilities. They have been scored bearing these facts in mind.
- **Operational graph databases –** InfiniteGraph frequently does not compete with either Titan or Neo4j (or any other graph provider for that matter). This is because of its focus on very large, whole graph applications, especially where these graphs change rapidly.
- **Analytic graph databases –** both Cray and Teradata are outliers in this category: Cray because it focuses on environments where there are a lot of unknowns and Teradata for the exactly opposite reason, because it focuses on known-known analytics. The other products in this category are more general-purpose.

It should be clear from the above that some of these apples are eating apples and some of them are cooking apples and some may even be crab apples! We could have used seven different colours in our Bullseye Chart but that would have been over complicated. Just bear in bear these distinctions when viewing the following chart.

Conclusion

As has been noted there are lots of open source and development projects within the graph space. We have focused on those that we believe to be enterprise-ready. That is to say, we expect features such as high availability, resilience, security, scalability and performance as well as features that are specific to the graph and RDF markets.

With the exception of IBM, which has not yet got its act together with respect to graphs, all of the products included in this Market Update have significant strengths. The difficulty for potential users is identifying the particular types of use case for which each product is most suitable. This is one of the reasons why this is a rather longer Market Update than is typical: because we have wanted to give some indication as to the focus areas of the different vendors. As always, ultimately users should conduct proofs of concept both with respect to functionality and performance.

Bloor

2nd Floor
145–157 St John Street
LONDON EC1V 4PY
United Kingdom

Tel: **+44 (0)20 7043 9750**
Web: **www.BloorResearch.com**
email: **info@BloorResearch.com**