CAMBRIDGE SEMANTICS
THE SMART DATA COMPANY®

ANZOGRAPH® DB for Software Developers

How graph analytics databases provide the foundation for AI and advanced analytics and represent a new growth opportunity for software developers

Learn how the AnzoGraph DB, a highly scalable and fast graph analytics database, can empower boundless applications in the graph, AI, and machine learning revolution. AnzoGraph DB partners are creating the next wave of on-premises SaaS, cloud, and Kubernetes-based applications with the powerful AnzoGraph DB graph database.

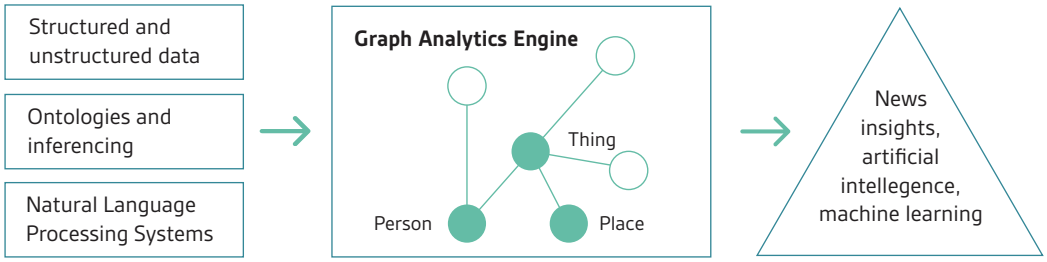
ANZOGRAPH® DB

Analytics in an Untamed Data World

Not so long ago, much of the world's data sat in structured databases, ready and waiting to deliver reports and analytics. Data engineers knew the data format of the data stored, what kinds of reports needed to run, and how long it would take to deliver clean data for analytics or reporting. They could even take data from sources that had varying degrees of structure and put it into a highly structured box, namely a database table. Database engineers and architects spent a lot of time learning about schemas and talking about managing metadata.

Today, you want to be able to use data from everywhere in order to get deeper insights with advanced analytics and machine learning. However, you may not know what data is available, and therefore, defining a traditional schema is more problematic than ever. Much of the world's valuable data is contained in human-readable natural language or semi-structured formats and it is that data that represents an even greater opportunity for valuable insights. Connected information is more important than ever, linking people, places and things in the real world in order to see patterns and communities.

Not only are customers asking software developers to handle new data sources, but they are also asking for new types of analytics as well. Most companies have evolved their analytical strategy from spreadsheets and BI-style analytics to performing data science, graph algorithms, and machine learning analytics. Performing advanced analytics on unconventional data sources has the biggest market growth potential for software developers looking to provide insights.



The good news is that graph databases are designed to overcome the challenges of performing analytics outside of rigid schemas and metadata management. They can allow ontologies and given knowledge sets to help you perform new and exciting types of analytics that were difficult or unreachable before. Let us not forget that what matters in these new untapped data sources are the entities and their relationships - who is a friend; linkages between customer, products and it networks; etc. Graph databases, specifically AnzoGraph DB, can solve these analytical challenges at scale and with speed.

We think that the emerging world of AI and machine learning offer workloads that are well-suited for graph databases. We believe this for two main reasons. First, data scientists in this field are often harmonizing diverse structured data and unstructured data points. By connecting from more diverse sources like these, you can provide deeper and more insightful analytics. Second, much of the work done by data scientists is well-suited for graph databases. Many of the machine-based algorithms are graph algorithms such as community detection algorithms, path finding algorithms, or similarity or centrality algorithms.

In this white paper, you will learn about the power of graph databases and how they can be used to drive new value to your customers when you build analytical applications.

Uses of Graph

By leveraging the power of a highly scalable fast graph analytics database, software developers can not only easily harmonize new streams of untamed data, but they can also build a Knowledge Graph with common business understanding of the data. The Knowledge Graph can be used by other software developers and end customers to allow for data discovery and the use of a variety of analytics capabilities to build analytical applications or get new insights. Below are examples of some of the value being seen in applications, from customer retention to compliance to traffic optimization, and in industries from retail to healthcare to telecommunications.

- **Data Harmonization and Knowledge Graphs -** Organizations are using graph databases to build Knowledge Graphs to provide common business understanding to the data harmonized from diverse sources. Knowledge Graphs stores entities and relationships in data, and allow users to search, analyze and use this connected data to accelerate vital new discoveries. Researchers freely "pivot" their analytics to ask new, ad hoc questions without being limited by rigid database schemas. Knowledge Graphs are replacing passive metadata and master data management solutions and dictionaries, and they provide much more value.
- **Entity Resolution -** A common problem with data from diverse sources is how different entities, such as names, places, addresses, and phone numbers, which can all be written slightly differently, can be matched and harmonized. Graph provides an easy way to visualize such entities and relationships, and to harmonize and resolve discrepancies.
- **Unstructured Data Analytics -** Combined with NLP, graph database offers a free-form repository to store the output of NLP, which is often formatted in RDF triples and used for data discovery and analytics.
- **Key Influencer Analytics -** Analyze all customer data to find key opinion leaders. Gain new insight into each customer's likes and dislikes in relation to other customers with similar location, similar demographics, etc. Discover new correlations between customers with inferencing, for more personalized and engaging customer experiences.
- **Recommendation Engine Analytics -** Recommendation engines are perfect in a graph when you want to make use of algorithms and data to recommend the most relevant items to a particular user.
- **Fraud Analytics -** Use Graph to detect fraudulent trading patterns and transactions in real-time. Semantically identify and understand the intricate relationships between entities and transactions, including the many individuals and organizations involved with those transactions.
- **Path Optimization Analytics -** Use Graph to analyze how the nodes connect and interact with each other and obtain powerful insights.
- **Social Analytics -** Use Graph to track of social networks and understand influence.

- **AI and Machine Learning -** Using graph algorithms, custom algorithms, and the knowledge provided by ontologies, graphs can provide decision support and help with predictions. Graph databases let you connect data, while providing rich data and metadata for deeper and more insightful analytics. They are also the basis for many of the machine-based algorithms and graph algorithms such as community detection algorithms, pathfinding algorithms, similarity, or centrality algorithms.

Graph analytics databases represent the next evolution of databases to address the growing amounts of diverse data and machine-based insights. These databases allow you to do the analytics of traditional databases, such as BI-style analytics, while providing access to new analytics capabilities and the ability to deal with diverse data. As software engineers build new analytical applications they should select the right engine to achieve the results for their customers. If not, you may find yourself performing unnatural acts with row-oriented databases that weren't designed to handle connected information.

Connected Data with AnzoGraph DB

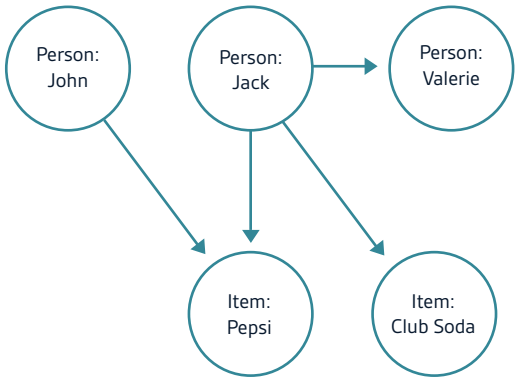
One of the biggest benefits that graph databases deliver is that it is much easier to use data without having to think too much about how it will be used. Rather than having to put data into square boxes, as you must do in traditional RDBMS, graphs have a different approach. Graph databases, specifically RDF triple stores like AnzoGraph DB, deal with data that's configured in SUBJECT-PREDICATE-OBJECT also known as triples. Of course, the facts are in a format that is specified by the RDF specification, but essentially, you'll see facts like:

- John is a person
- Jack is married to Valerie
- John buys Pepsi
- John is the son of Jack

Sales		
Customer	Item	Time
0001	1A	20:14
0001	1A	21:15
0003	2A	21:16
0002	1A	21:16
0002	5C	21:16

Inventory	
Description	
	SKU
Pepsi	1A
-	2A
-	-
Cup Noodles	
	5C

Customer	
Name	CustID
John	0001
Jack	0002
Ted	0003
Ken	0004
Valerie	0005



Traditional databes store data to efficiently store facts, but relationships must be rebuilt with JOINs and other inexact techniques

Graph databases store both facts and the relationships between facts, making certain types of analysis more intuitive and faster to compute for both humans and machines

In this system, you do not have to know anything ahead of time about what you want to store and what type of analytics you want to run. Facts about John can be added at any time. If new data comes along about John on any subject, it can be stored in a triple and not a separate table. Separate tables and joins with graph databases do not need to be created. You can get a lot of value from triple stores while avoiding much of the complex ongoing admin and maintenance required by relational databases (RDBMS) and traditional data management tools.

With all this linked information you can see what linked fact may be important to your analytics task. You can see what any given customer purchased, how influential their purchasing decisions are, etc. It allows you to go deeper or wider to find out who lives in New York, who buys BMWs, and who is Andrew's father.

When Triples Aren't Enough

It can get more complex than that, however. When graph databases first emerged on the market, there were two types: RDF Triple Stores and Labelled property graphs. While RDF followed a recognized standard, set up by some of the same people who set standards on the web, it had limitations for the amount of complexity possible. For example, it was easy to store relationships, like John is a friend of Sue, in a triple store. However, if you wanted to store when John and Sue became friends or some detail about either John or Sue, it meant adding more triples, leveraging quads for provenance, or using other methods to effectively store this information. In the past, it was easier to use a labeled property graph instead of a triple store because you could add those properties as part of the triple.

Modern triple stores like AnzoGraph DB, however, offer you the best of both worlds. If you need more complexity in AnzoGraph DB, you can use properties like those on a labeled property graph. AnzoGraph DB supports the new emerging standard from the W3C for properties called RDF*/SPARQL*. Properties can be used to identify, for example, when John bought the BMW, or how much he likes the brand.

AnzoGraph DB also supports the leading labeled property graph language, OpenCypher, on the same database and is currently in an early trial phase. So on the same database, you can use RDF with labeled properties based on the emerging RDF*/SPARQL* standards or use OpenCypher.

AnzoGraph DB and Its Value to Your Application

AnzoGraph DB is a solution that is simple, cost-effective, and offers great performance for performing analytics on connected data. At the core of the AnzoGraph DB is an MPP database written in C++ that is based on the new RDF*/SPARQL* open graph database standards and OpenCypher standards (beta signup) on the same database. Unlike many of the graph stores on the market, AnzoGraph DB provides an analytical graph and offers ISVs and their customers horizontal scalability where database servers can be added incrementally to deal with more data and to provide better performance. ISVs and their customers can start small and grow the solution as the business grows.

Additional benefits:

Analytics

- SPARQL functions that provide powerful analytics in a syntax that is similar to SQL
- Complete analytical functions like Aggregates, Federated Query, ORDER BY and offsets, Hash Functions, Count, Avg, Min, Max and more
- Inferences, also known as reasoning, to understand unidirectional, bilateral and familiar relationships.
- Extensive graph analytical functions like Page Rank, Shortest Path, All Path, Label Propagation, Weakly Connected Components, K neighborhood and Counting Triangles, as well as APIs you can use to create and deploy your own analytical functions.

Scalability and Deployment

- An MPP clustered approach to storing and analyzing big data, offering superior query and analytic performance
- Flexibility and scalability to easily ramp up when workloads increase
- Deployable on-premises, cloud, and container (Docker)
- The fastest loading of any leading database, allowing you to instantiate a database, load and analyze data in a matter of seconds

Beyond a Standard SPARQL Endpoint

- An SDK that supports C++, JAVA and many other languages when building a custom application.
- Support for properties in RDF, under the new W3C standard.
- OpenCypher and Bolt support (Beta signup)

Analytics In AnzoGraph DB: All-in-one analytics

In order to see the value of AnzoGraph DB, it's important to dive into the analytical capabilities of the platform. While the market is filled with solutions that can do analytics, often certain categories of analytics that are necessary for an application are missing or non-performant.

SPARQL 1.1 Standard Analytics

AnzoGraph DB implements the standard SPARQL forms and functions described in the W3C SPARQL 1.1 Query Language specification. Like SQL, SPARQL is a declarative language that can perform all of the analytics that a SQL system can do. For example, in both languages, SELECT statements and WHERE clauses can be used to analyze data, as well as ORDER BY, LIMIT and OFFSET commands, to name a few. However, since graph databases store data in triples in a simple SUBJECT-PREDICATE-OBJECT data model, it is optimized for SPARQL, which was was designed to query data that uses this data model to better analyze the relationships of data.

So why not just use SQL? To support relationship queries, it is necessary to supersede the more familiar SQL language with SPARQL or a similar language. SPARQL and others adopt the FOAF vocabulary, devoted to linking people and information. The additional functions provider the ability to identify

influencers, identify patterns in a supply chain, find friends or spousal information, manage parent/child relationships in people and in companies, define classes and subclasses, and so much more.

AnzoGraph DB is the only graph database that supports both RDF*/SPARQL* standards and OpenCypher (beta signup). Use the language you know to get the results you need. If programmers are more familiar withCypher from using Neo4j, they can continue to use it with AnzoGraph DB. This dual-language support is unique in that AnzoGraph DB supports two well-known standards in a scalable analytical database. It opens a world of opportunity in being able to use tools meant for the RDF/SPARQL ecosystem or the OpenCypher/LPG ecosystem.

Data Warehouse-Style BI Analytics

An analytical database needs to provide data warehouse-style BI analytics for reporting and ad-hoc analysis. Having performant aggregate functions and the capability to FILTER and JOIN is crucial to some applications. AnzoGraph DB has it all. AnzoGraph DB also provides a full list of built-in Excel-type functions.

AnzoGraph DB's value proposition is scale and speed for conducting online analytics, as well as the speed at which data can be loaded for such analytics.

Our engineers put this aspect of AnzoGraph DB to the test with an industry benchmark. For example, TPC-H is an analytical benchmark designed to compare data warehouse performance. In our documentation, you can see the 22 TPC-H benchmark queries converted to SPARQL. These data warehouse queries run faster in AnzoGraph DB than on many relational databases.

Data warehouse-style queries can be a challenge for most graph databases in the market today. If your application uses aggregate functions or performs deep analysis of a lot of data, you should put it to the test before you adopting a database.

Graph Algorithms

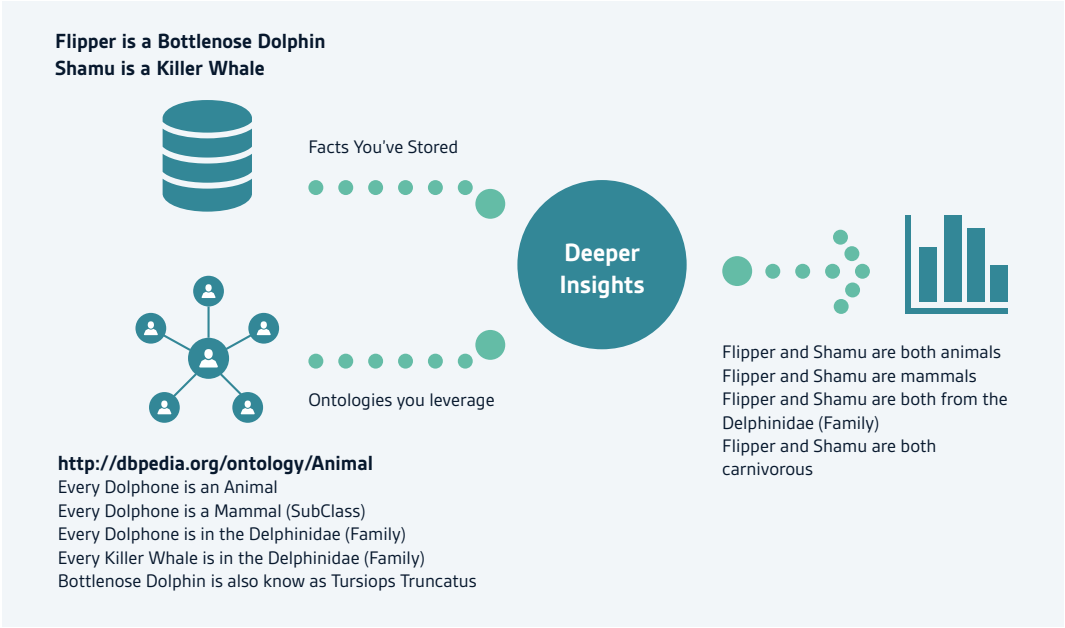
Graph databases like AnzoGraph DB are known for graph algorithms. Graph algorithms, including PageRank, Shortest Path, All Paths and many others, help you solve your data analytics challenges at scale with ease. For example, PageRank helps you understand things like who is the most influential person or what network router gets the most traffic. PageRank outputs a positive rational number, not necessarily linked to a graph-type visualization.

These types of calculations are comparably simple on graph databases and very efficient. If you were to try a PageRank in a relational database, it is possible, but you would likely end up with a multi-step iterative process that would perform calculations on an entire database, join two or more tables, and calculate aggregates like SUM. It is a very taxing process on a SQL database that would need to be repeated time and time again when new data is introduced. It is just easier and more efficient in a graph database.

Graph algorithms are being used by companies to get machine-based insights. For example, PageRank is used to find the key opinion leader in a sales situation, fraud use cases, and other networked scenarios where you want the machine to identify who is a key influencer. AnzoGraph DB's scale and speed allows software developers to analyze larger amounts of data , and provides much needed better performance than other graph databases.

Inferencing and Ontologies

Many RDF databases support logical reasoning during application runtime to answer queries about facts that have not been explicitly saved. Anzograph DB includes inferencing capabilities, written to the W3C RDFS+ specification. Inferencing capabilities can create new relationships and insight based on the vocabularies or ontologies in the existing data. The diagram below is an example from the W3C Semantic Web inference documentation.



This example is a simple one, leveraging the animal ontology. However, imagine you had relatively few facts in your database. Using known external ontologies, you could infer a great deal more.

There are many, many ontologies to help you gain insight. Another popular one is the FIBO ontology or Financial Industry Business Ontology. FIBO is a formal model of the legal structures, rights and obligations contained in the contracts and agreements that form the foundation of the financial industry. It can go far in helping you manage data and gain insight in a financial services organization.

Ontologies exist for many vertical applications. For example, there are automotive, financial services, oil and gas, health care, and many others. All ontologies are based on the Web Ontology Language (called OWL) and can easily generate new inferred relationships according to the OWL rules. You can also create your own ontologies.

User-defined Analysis and Out-of-box Data Science Functions

If you have not seen the exact kind of analytics that you need for your project, Anzograph DB offers the capability to define your own functions that leverage the power of the graph. AnzoGraph DB exposes a number of extension points through which developers can customize and extend the system. The extension point interfaces and the user code that implements them are called user-defined extensions (UDX).

Currently, AnzoGraph DB offers C++ and JVM APIs that developers can use to implement extensions. With this capability, you can design the following functions:

- **User-Defined Functions (UDF):** Create custom analytic functions, such as functions that concatenate values or convert integers to alternate currencies.
- **User-Defined Aggregates (UDA**): Create aggregate functions, such as functions that compute the arithmetic mean or calculate the average number from a list of maximum and minimum values. Unlike a UDF, which returns a distinct value each time it is applied, a UDA aggregates the collection of values to which it is applied to a single summary value.
- **User-Defined Services (UDS):** Create local SPARQL endpoints.
- **User-Defined Tables (UDT):** A function that is repeatedly invoked within a query to generate the rows of a table on-the-fly.

Cambridge Semantics has also developed a few user-defined functions and they are available to you. These functions are data science functions and include:

- **Correlations -** This type of analysis is useful when a data scientist wants to establish if there are possible connections between variables/factors. Correlation is not causation, but scientists use this for investigating possible connections between variables. For example, it is well-known that there is some correlation between height and weight. There may be a correlation between the time of day and the amount of energy generated by a solar panel.
- **Distributions -** Back in grade school, your teacher may have graded you on the Bell curve. This is perhaps the most simplistic and common type of distribution analysis. However, there are quite a few different types. Financial analysts and investors often use a distribution when analyzing the returns of a security or of overall market sensitivity and volatility.
- **Entropy -** Outside of the context of analytics, "entropy" is a term meaning a lack of order or predictability. You therefore might guess that an algorithm that tests entropy is one that is looking for predictable and unexpected results. In a way, it detects how remarkable an ongoing analysis is. For example in IoT, you might expect a device to send a consistent value or set of values. When the device varies from the norm, you can capture that with an algorithm.
- **Profiling -** There are some differences between data profiling, a system where you are looking for errors in your data, and profiling analysis, where you are trying to use several factors/variables to describe a 'profile' of a person, place or thing. Your company might have better success by marketing certain product and price points to categories of customers and profiling is where you'd start. Segmentation based on profiling analysis provides superior results. Profiling analysis is also great for recommendation engines. Additionally, profiling analysis may be used for IoT. Connected devices might be exhibiting a certain profile when they are about to fail, for example.

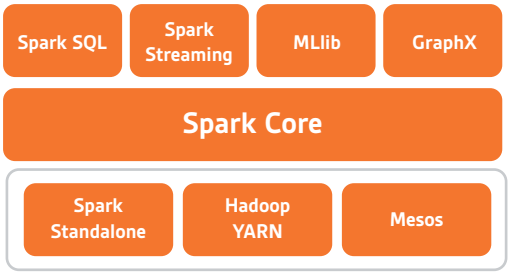
You can read more about how you would create your own algorithms and functions in the UDX section of our documentation.

Performance in AnzoGraph DB

When researching new database engines, you'll find that there are many levels of performance among graph database providers and even those non-database solutions like Spark GraphFrames. Solutions will vary depending upon whether the platform is meant to support transactions or analytics. Let's look at some of the features that are provided in AnzoGraph DB.

Single Unified Platform

AnzoGraph DB is a single installable application. As a result, you may find it easier to manage than some solutions on the market. For example, Apache Spark has seen a massive spike in adoption by enterprises mainly because Apache Spark is cheap to start and fairly easy to begin to use. However, it becomes very difficult when Spark applications start to slow down or fail.



For software developers, it's important to remember that a Spark application is always made up of multiple components including Spark SQL, GraphX or GraphFrames, YARN, Mesos or others that must be stitched together properly and tuned for your workload. When a query slows down or fails, it may not always be evident which component is at fault. Performance optimization is made easier by a single application.

Getting started with the AnzoGraph DB is easy. It supports the W3C standards SPARQL and RDF. Cambridge Semantics has already implemented the next generation of the SPARQL language by implementing property graphs. As well, we added support for OpenCypher and Bolt protocol (Beta signup), easing the process of migrating existing application code from other graph databases to AnzoGraph DB.

Developing with the AnzoGraph DB is easy to build into enterprise software, SaaS, and appliance-based applications:

- Supports SPARQL endpoint applications
- User-defined functions for JAVA and C++ come standard with AnzoGraph DB
- Leverage cloud-based or container-based replication, elasticity, and failover to save development time and improve performance and high-availability
- Online developer support and resources are available in StackOverflow

Massively Parallel

There are many solutions on the market that scale by letting you add nodes to a cluster. However, it's important to understand whether the cluster scales to meet the needs of concurrency and data insertion, or whether it scales to meet the needs of faster running queries on large data sets. Consider the kind of scale needed by your application and your workload.

Made for Analytics

The AnzoGraph DB has been specifically designed to support analytic workloads, which primarily comprise querying (often ad-hoc) and the insertion of new data (in bulk or on a constant "trickling" basis for real-time analytics). Because it is primarily memory-based and MPP AnzoGraph DB can deliver orders of magnitude faster performance than many relational databases when handling query-intensive analytic workloads.

To help quantify the advantage for you, the following tables contain performance benchmark results conducted by AnzoGraph DB and are based on industry-standard benchmarks:

Environment

High-powered Amazon Cloud Servers

Amazon EC2 R4.8xlarge (40)
Persistence disabled

- Environment selected for memory availability and processing power

Same Test, More Triples

Load 105.5 Billion Triples
Query Data warehouse-style queries

- 105,509,846,071 Triples
- Run 22 queries converted from TPC-H into SPARQL

Results

	Scale Factor 25 - 1 Node	Scale Factor 1000 - 40 Nodes
Data	2.6 Billion	105 Billion
Load	8 Min	18 Min
Queries	2 Min	4 Min

- Data loaded in under 18 minutes
- All queries completed in under 4 minutes

AnzoGraph DB loads new data at very high speeds as well. In fact, it took a world record-breaking, 4 minutes to load 105 billion triples of TPC-H data (generated retail sales data) into the AnzoGraph DB running in the Google cloud. AnzoGraph DB achieves fast loading by leveraging all the nodes for data loading.

Optimizations

In addition to standard SPARQL, OpenCypher (Beta signup), and custom user-defined functions, AnzoGraph DB supports some data warehouse optimizations that are important for analytics. When considering a graph database, consider how useful these functions would be in accomplishing your goals:

- **Named Views -** With named views, you can create virtual views or materialized views of data from one or more graphs. This can serve several purposes. If you'd like to hide the underlying complexity of data from users by offering a simplified view, named views can help. Or if you need to mask sensitive information from some users, you can use the named views feature to accomplish this.

- **Named Queries -** If you commonly use the same complex query in your analytics, you can create query definitions that you can reference as subqueries in other queries. This simplifies the writing of complex queries for your users of AnzoGraph DB.

- **Conditional Expressions -** Conditional expressions are not part of the SPARQL specification. However, they are included with AnzoGraph DB. CASE expressions enable you to add IF/THEN/ELSE logic to a query.

- **Windowed Aggregates -** Aggregate functions can be compute-expensive, especially on other graph databases that are meant for transactions rather than analytics. AnzoGraph DB is not only fast and powerful for aggregate functions, but it also supports windowed aggregates. Here you can compute aggregate values on a particular partition or window of the result set. This is often done based on time, as in computing hourly, daily or monthly reporting. Functions include AVG, COUNT, MAX/MIN, NTILE, PERCENTILE, PRODUCT, QUARTILE, ROW_NUMBER, and SUM

- **Advanced Grouping -** AnzoGraph DB supports advanced reports using grouping set extensions in the GROUP BY clause. For example, you might wish to summarize financial data by product, by time-period, and by city to compare actual and budget expenses. With AnzoGraph DB, you can use familiar functions like CUBE and ROLLUP to optimize reporting with advanced grouping.

Flexible Deployment

Most modern databases allow you to deploy on-premises, or on cloud infrastructure. These are table stakes for modern-day software. However, you can easily define and install our Massively Parallel Processing (MPP) graph database in a Kubernetes cluster on-demand to meet the scale and size of the data. Should those needs change, you can quickly spin up and down AnzoGraph clusters with the latest versions. By integrating AnzoGraph in a microservices architecture, it represents a game-changer for database deployment. On-Demand, data-driven scale-out deployments are just easier in Kubernetes with its tooling like helm and operators.

After the analysis is done, there is no need to keep the twitter data around, just the results. By being able to quickly spin up a containerized analytics environment that loads fast, computes quickly, and hibernates in seconds, companies can realize amazing cost savings over static, bare metal, or cloud. It is the new way to do analytics.

Deliver Highly Differentiated Analytic Capabilities to Customers

The performance advantage explained above enables companies that embed the AnzoGraph DB to really stand out from the competition and deliver higher value to customers. There are several areas where functionality tends to be improved and differentiated with AnzoGraph DB:

- Ad-hoc access to years of historic data, not just weeks or months
- Analyze data at any level of granularity, not just pre-calculated summary data
- Warehouse-style loading and analysis

Lower Costs and Increase Profitability with the AnzoGraph DB

Another benefit of building applications on the AnzoGraph DB is the cost savings. Here's how the AnzoGraph DB affects cost compared with the other data management options:

Category	Products that fall into this category	Natively support NLP output	Unified platform	Scale to meet big data analytics needs	Built-in Inferencing	Graph Algorithms
RDBMS	Oracle, MySQL, etc.	○	●	◔	◔	●

OLTP Graph Databases	Neo4j, Amazon Neptune and most other graph databases					
SPARK and GraphFrames	Spark					
Document Stores	MongoDB, Cassandra					
AnzoGraph	AnzoGraph DB					

- **Traditional databases (e.g. Oracle, MySQL, SQLServer) –** Although generally offered as a single platform, traditional databases don't naturally deal with disparate data sources and ontologies as easily as AnzoGraph. Certain analytical databases in this category will scale, while most do not. You may have to perform unnatural acts in order to perform graph analytics like pagerank, shortest path or perform inferencing. With the new forms of data analysis that your customers require, time and money will be spent for non-performant results. Nor do these database scale to allow for the depth of analysis available from AnzoGraph DB.

- **OLTP graph databases (e.g. Neo4j and AWS Neptune) –** OLTP graph databases do well on graph algorithms and simplified management, but the graph algorithms have limits when it comes to deep analysis of a lot of data. Not all offer inferencing and OWL support. You'll find that data loading is too slow for many applications.

- **SPARK and GraphFrames -** GraphFrames is a package for Apache Spark that is growing in popularity and provides some graph functions. In our benchmarking, the SPARK/GraphFrames stack is not as performant as native graph database. SPARK with GraphFrames is not simply one engine, but is actually several technologies in a stack. In order to perform graph analytics in SPARK, you may need YARN, Mesos and/or Spark Scheduler for management of the jobs and clusters, the SPARK core engine with the GraphFrames libraries, and some languages like SCALA, Python and R. These multiple technologies form a stack and you must provision appropriately for resources. If you want to visualize or extract and load data, additional technologies may be needed.

- **NoSQL Document Stores -** Document stores like MongoDB, MarkLogic and Cassanadra work by indexing documents.

Licensing and Deploying AnzoGraph-Based Solutions

Commercial software applications are delivered to customers in a variety of ways, and the AnzoGraph DB offers the deployment and licensing flexibility required to be built into:

- Installed software solutions
- In-house hosted solutions
- Cloud-hosted solutiovns
- Containerized solution

The AnzoGraph DB runs on industry-standard AWS, Google or Microsoft Azure servers. AnzoGraph DB runs on the Amazon Cloud, allowing solution providers to bring new offerings to market faster and less expensively without any in-house data center costs. Cambridge Semantics offers flexible licensing programs to ISVs that are designed to support your licensing model and foster mutual growth and profitability.

Additional Resources

Cambridge Semantics offers a free trial that is full-featured and allows you to develop applications using the AnzoGraph DB engine. You can take advantage of built-in features, or build your own using our User-defined extensions. If you would like to learn more about AnzoGraph DB, download documentation, or evaluate it yourself, please visit www.cambridgesemantics.com/anzograph.

Summary

While there are many reasons that developers of commercial analytic software and appliances should embed AnzoGraph DB, here are four for you to consider:

Scale and Performance

Guaranteed service level agreements (SLAs) competitors cannot match.

- Spin up AnzoGraph DB instances quickly in the cloud or in Kubernetes as you need them.
- Load data quickly using parallel loading capabilities.
- Start small and scale up as your data volume needs increase
- Execute analytical queries in seconds.
- Complete analytics from zero to completion without having resources idle.

Competitive Differentiation

Deliver analytical functionality that other databases cannot economically support.

- Handle your typical data warehouse-style analytics while leveraging the graph database for performing analytics that are difficult or impossible in traditional solutions.
- Use inferencing and ontologies to add a new level of intelligence and enhance the data.
- Understand the relationships in your data and provide support for knowledge graphs.
- More easily and economically perform certain algorithms like PageRank and Shortest Path, or whatever algorithm you want to build.

Profitability and Time to Market

Build and run solutions faster and at a lower cost by eliminating:

- Having to manage open source specialized data management layers
- Time and cost of designing your own database
- Any in-house data center startup costs at all, by using the AnzoGraph DB on the cloud or in Docker containers

Licensing & Packaging Flexibility

Innovate successful business models with AnzoGraph DB's flexible licensing options:

- Customized licensing to fit your business model and support your growth
- Gives you the freedom to deliver hosted, installed, or appliance solutions