Varigence Documentation

Cover Page sub header

Table of Contents

Chapter 1

Get Up To Speed With Data Warehouse Automation

1. What is a Data Warehouse?

   A data warehouse (DW or DWH), also known as an enterprise data warehouse (EDW), is a system used for reporting and data analysis, and is considered a core component of business intelligence. DWs are central repositories of integrated data from various sources.

   A data warehouse is a key enabler of BI. It is 'fed' data by operational systems (such as a transaction database, customer database, call centre system) to create one repository of information. And this repository is then used for different activities like reporting and analytics.

   The important fact is that a transactional database doesn't lend itself to analytics. To effectively perform analytics, you need a data warehouse. A data warehouse is a database of a different kind: an OLAP (online analytical processing) database. A data warehouse exists as a layer on top of another database or databases (usually OLTP databases). The data warehouse takes the data from all these databases and creates a layer optimized for and dedicated to analytics.

   A database designed to handle transactions isn't designed to handle analytics. It isn't structured to do analytics well. A data warehouse, on the other hand, is structured to make analytics fast and easy.

   It serves as corporate memory, collecting the body of history that makes time-series and trend analysis possible.

   The data warehouse also organises and structures data to make it understandable and useful for consumption by many different business stakeholders. This business intelligence gives organisations the edge, making them more competitive, more customer focused, more profitable.

   - Data warehouse is an information system that contains historical and commutative data from single or multiple sources.
   - A data warehouse is subject oriented as it offers information regarding subject instead of organization's ongoing operations.
   - In Data Warehouse, integration means the establishment of a common unit of measure for all similar data from the different databases
   - Data warehouse is also non-volatile means the previous data is not erased when new data is entered in it.
   - A Datawarehouse is Time-variant as the data in a DW has high shelf life.
   - There are 5 main components of a Datawarehouse. 1) Database 2) ETL Tools 3) Meta Data 4) Query Tools 5) DataMarts
   - These are four main categories of query tools 1. Query and reporting, tools 2. Application Development tools, 3. Data mining tools 4. OLAP tools
   - The data sourcing, transformation, and migration tools are used for performing all the conversions and summarizations.
   - In the Data Warehouse Architecture, meta-data plays an important role as it specifies the source, usage, values, and features of data warehouse data.

2. What is Data Warehouse Automation?

   Data warehouse automation or DWA refers to the process of accelerating and automating the data warehouse development cycles, while assuring quality and consistency.

   Data Warehouse Automation (DWA) is a modern approach to data warehousing. An evolution from traditional ETL, it provides automation and optimizations from designing the warehouse, to generating ETL code, to quickly applying updates, all leveraging best practices and proven design patterns.

DWA is the core of devops in the world of analytics, it links together the design and implementation of analytical environments into repeatable processes and should lead to increased data warehouse and data mart quality, as well as decreased time to implement those environments.

Data Warehouse Automation describe the automation of the following:

1. Simplified capture of the Data Warehouse **Design**.
2. Automated **Build** (i.e. Generate Code)
3. Automated **Deployment** of code to the Server
4. Automated **Batch** execution of the ETL code on the Server.
5. Automated **Monitoring and Reporting** of the Batch execution.

Generally this is achieved through a Data Warehouse Automation tool.

3. The Evolution of DWA

In the past, designing a data warehouse has taken too long to complete. Executing numerous semi-automated steps results in a data warehouse that was limited and inflexible. The DWA solution is to automate the data warehouse through every step in the life-cycle, thus reducing the efforts required to manage it.

The DWA solution comprises a central repository of design patterns, which encapsulate architectural standards as well as best practices for data design, data management, data integration, and data usage.

DWA increases developer productivity by automating and accelerating the routine tasks associated with building and managing decision support infrastructures, and gradually improves the time and reduces the repetitive manual efforts associated with each step of a data warehouse lifecycle.

A Data Warehouse provides additional benefits over other solutions (i.e. Self Service BI). For example:

- The ability to keep history,
- A single source of the truth,
- End user productivity,
- Reduced risk of reliance on key individuals,
- Data augmentation,
- Query performance etc.

The problem isn’t the Data Warehouse concept, its still an extremely useful method of managing information. The problem has been execution of Data Warehouse development.

This is where Data Warehouse Automation comes in. It doesn’t throw out the idea of a Data Warehouse in search of a better way, instead it directly addresses the real problem, being the execution of Data Warehouse development.

4. The Limitations of Not Automating

Today’s data warehouse must meet fast-changing business requirements, support new data sources, and deliver solutions in quick iterations. Traditional ETL tools lack the agility, ease of use and level of automation needed to answer these requirements because they are:

- SLOW. DWA will dramatically reduce your development time.

- INFLEXIBLE: DWA responds to changing business requirements quickly and easily; it's agile.

- BOGGED down in CODE. DWA allows you to focus on what really matters, reporting and analytics instead of being stuck in ETL code.

- OFTEN OUT of DATE: DWA tools produce tested, high performance, complete and readable code, fast.

- Difficult to STANDARDISE: DWA produces consistent code, naming standards etc. Developers come and go, but as long as they keep using the same tool, its easy for one developer to understand the work of another.

  So, using DWA your business can make changes much later in the development process and change can occur more frequently with less disruption, waste and rework. This efficiency is not only a joy, it saves time, resources and money. In a traditional data warehouse build, it is especially difficult to get complete and correct requirements due to the linear development process. Automation also brings quality benefits through standards enforcement and standardising the development processes.

  The agility of the automated data warehouse is not limited to its ability to change in the warehouse development process, it can also handle changes in business requirements. Responding to change in real time and without the delay of lengthy projects is the essence of business agility.

Chapter 2

Data Warehouse Landscape Options

Business users, application owners and IT executives must realize, in an increasingly software-defined data center (SDDC) world which includes networks, servers, storage and databases, it is expedient to deploy specialized, purpose-built databases best suited for the task. This represents the new normal for enterprise database deployment. One size does not fit all and while more traditional, relational-centric DBs are unlikely to disappear in the next decade, the preeminence of the RDBMS as the central warehouse for enterprise data is quickly waning as data structures and tools to ingest, manage, analyze and visualize new semantic or schema-less text and data formats become dominant. Enterprises of all sizes and in every industry who still rely on internal IT for supporting their own applications need to follow the lead of cloud service providers (CSP) and other organizations with web-scale infrastructures: Select the best database(s) for the job. In this regard, the NoSQL space has much to offer.

In the ever-evolving world of enterprise IT, choice is generally considered a good thing – albeit having too many choices can create confusion and uncertainty. For those application owners, database administrators and IT directors who pine for the good old days when one could count the number of enterprise-class databases (DBs) on one or two hands, the relational-database-solves-all-our-data-management-requirements days are long gone. Note: The clear trend for non-relational database deployment is for enterprises to acquire multiple DBs based on application-specific needs, what could be referred to as software-defined database adoption.

Thanks to the explosion of Big Data throughout every industry sector and requirements for real-time, predictive and other forms of now indispensable transactions and analytics to drive revenue and business outcomes, today there are more than 50 DBs in a variety of categories that address different aspects of the Big Data conundrum. Welcome to the new normal world of NoSQL – or, Not only Structured Query Language – a term used to designate databases which differ from classic relational databases in some way.

Evolution of NoSQL

In the beginning, there was SQL (structured query language). Developed by IBM computer scientists in the 1970s as a special-purpose programming language, SQL was designed to manage data held within a relational database management system (RDBMS). Originally based on relational algebra and tuple relational calculus, SQL consists of a data definition language and a data manipulation language. Subsequently, SQL has become the most widely used database language largely due to the popularity of IBM, Microsoft and Oracle RDBMSs.

NoSQL DBs started to emerge and become enterprise-relevant in the wake of the open-source movement of the late 1990s. Aided by the movement toward Internet-enabled online transaction processing (OLTP), distributed processing leveraging the cloud and the inherent limitations of relational DBs, including lack of horizontal scale, flexibility, availability, findability and high cost, use of NoSQL databases has mushroomed.

Amazon's instantiation of DynamoDB is considered by many as the first large-scale, or web-scale, production NoSQL database. To quote author Joe Brockmeier, who now works for Red Hat, "Amazon's Dynamo paper is the paper that launched a thousand NoSQL databases." Brockmeier suggests that the "paper inspired, at least in part, Apache Cassandra, Voldemort, Riak and other projects."

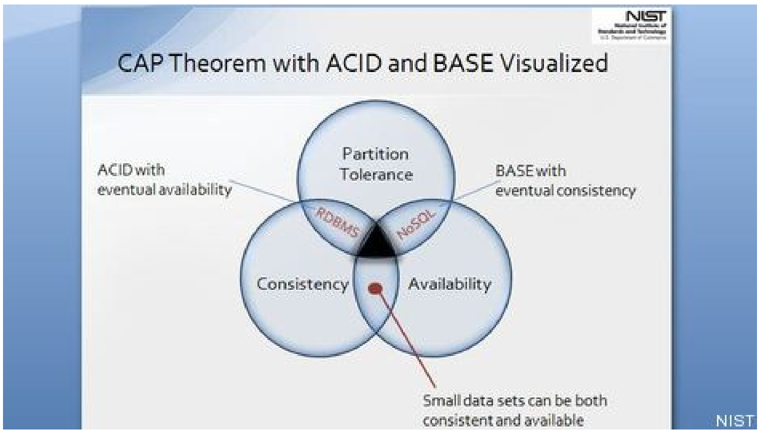
According to Amazon CTO Werner Vogels, who co-authored the paper entitled Dynamo: Amazon's Highly Available Key-value Store, "DynamoDB is based on the principles of Dynamo, a progenitor of NoSQL, and brings the power of the cloud to the NoSQL database world. It offers customers high availability, reliability, and incremental scalability, with no limits on dataset size or request throughput for a given table." DynamoDB is the primary DB behind the wildly successful Amazon Web Services business and its shopping cart service that handles over 3 million "checkouts" a day during the peak shopping season.

As a result of the Amazon DynamoDB and other enterprise-class NoSQL database proof points, it is not uncommon for an enterprise IT organization to support multiple NoSQL DBs alongside legacy RDBMSs. Indeed, there are single applications that often deploy two or more NoSQL solutions, e.g., pairing a document-oriented DB with a graph DB for an analytics solution. Perhaps the primary reason for the proliferation of NoSQL DBs is the realization that one database design cannot possibly meet all the requirements of most modern-day enterprises – regardless of the company size or the industry.

The CAP Theorem

In 2000, Berkeley, CA, researcher Eric Brewer published his now foundational CAP Theorem (consistency, availability and partition tolerance) which states that it is impossible for a distributed computer system to simultaneously provide all three CAP guarantees. In May 2012, Brewer clarified some of his positions on the oft-used "two out of three" concept.

- Consistency (all nodes see the same data at the same time)
- Availability (a guarantee that every request receives a response about whether it was successful or failed)
- Partition Tolerance (the system continues to operate despite arbitrary message loss or failure of part of the system).


CAP Theorem with ACID and BASE Visualized
NIST
Partition Tolerance
ACID with eventual availability
BASE with eventual consistency
RDBMS
NoSQL
Consistency
Availability
Small data sets can be both consistent and available
NIST

According to Peter Mell, a senior computer scientist for the National Institute of Standards and Technology, "In the database world, they can give you perfect consistency, but that limits your availability or scalability. It's interesting, you are actually allowed to relax the consistency just a little bit, not a lot, to achieve greater scalability. Well, the Big Data vendors took this to a whole new extreme. They just went to the other side of the Venn diagram, and they said we are going to offer amazing availability or scalability, knowing that the data is going to be consistent eventually, usually. That was great for many things."

ACID vs. BASE

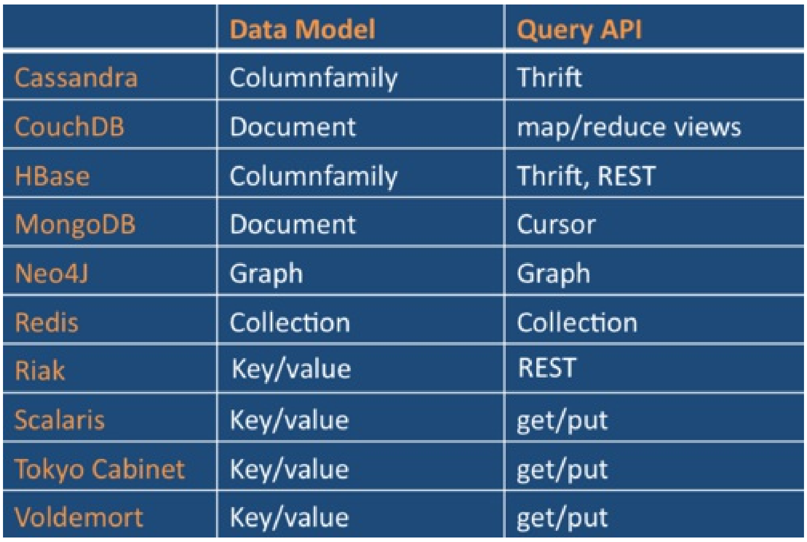
In most organizations, upwards of 80% of Big Data is in the form of "unstructured" text or content, including documents, emails, images, instant messages, video and voice clips. RDBMSs were designed to manage "structured" data in manageable fields, rows and columns such as dates, social security numbers, addresses and transaction amounts. ACID (Atomicity, Consistency, Isolation, Durability) is a set of properties that guarantees database transactions are processed reliably and is a necessity for financial transactions and other applications where precision is a requirement.

Conversely, most NoSQL DBs tout their schema-less capability, which ostensibly allows for the ingestion of unstructured data without conforming to a traditional RDBMS data format or structure. This works especially well for documents and metadata associated with a variety of unstructured data types as managing text-based objects is not considered a transaction in the traditional sense. BASE (basically available, soft state, eventually consistent) implies the DB will, at some point, classify and index the content to improve the findability of data or information contained in the text or the object.

Increasingly, a number of database cognoscenti believe NoSQL solutions will or have overcome the "ACID test" as availability is said to trump consistency – especially in the vast majority of online transaction use cases. Even Eric Brewer argued recently that bank transactions are BASE not ACID because availability = $.

NoSQL Database Categories

As will be seen in the following section, NoSQL DBs simultaneously defy description and define new categories for NoSQL databases. Indeed, many NoSQL vendors possess capabilities and characteristics associated with more than one category, making it even more difficult for users to differentiate between solutions. A good example is the following taxonomy provided by Cloud Service Provider (CSP) Rackspace, which classifies NoSQL DBs by their data model.

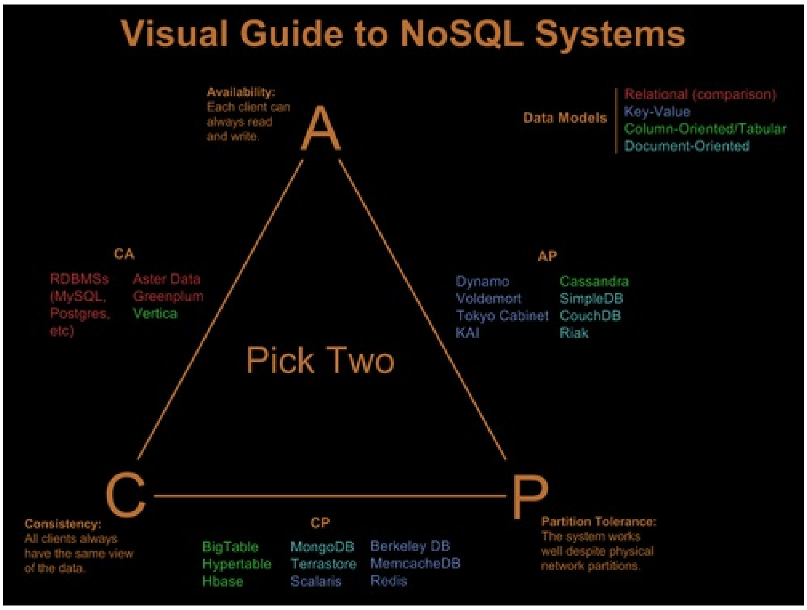


	Data Model	Query API
Cassandra	Columnfamily	Thrift
CouchDB	Document	map/reduce views
HBase	Columnfamily	Thrift, REST
MongoDB	Document	Cursor
Neo4J	Graph	Graph
Redis	Collection	Collection
Riak	Key/value	REST
Scalaris	Key/value	get/put
Tokyo Cabinet	Key/value	get/put
Voldemort	Key/value	get/put

Note: : In the original slide, Riak is depicted as a "Document" data model. According to Riak developer Basho, Riak is actually a key-value data model and its query API (application programming interface) is the popular web REST API as well as protocol buffers.

The chart above represents the five major NoSQL data models: Collection, Columnar, Document-oriented, Graph and Key-value. Redis is often referred to as a Column or Key-value DB, and Cassandra is often considered a Collection. According to Technopedia, a Key-Value Pair (KVP) is “an abstract data type that includes a group of key identifiers and a set of associated values. Key-value pairs are frequently used in lookup tables, hash tables and configuration files.” Collection implies a way documents can be organized and/or grouped.

Yet another view, courtesy of Beany Blog, describes the database space as follows:



“In addition to CAP configurations, another significant way data management systems vary is by the data model they use: relational, key-value, column-oriented, or document-oriented (there are others, but these are the main ones).

- Relational systems are the databases we’ve been using for a while now. RDBMSs and systems that support ACIDity and joins are considered relational.
- Key-value systems basically support get, put, and delete operations based on a primary key.
- Column-oriented systems still use tables but have no joins (joins must be handled within your application). Obviously, they store data by column as opposed to traditional row-oriented databases. This makes aggregations much easier.
- Document-oriented systems store structured ‘documents’ such as JSON or XML but have no joins (joins must be handled within your application). It’s very easy to map data from object-oriented software to these systems.”

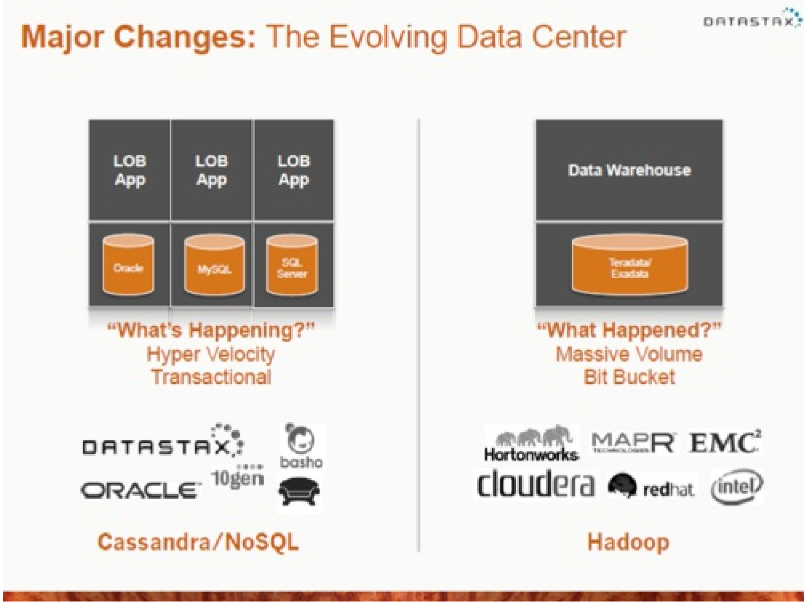
Beany Blog omits the Graph database category, which has a growing number of entrants in the space, including; Franz Inc., Neo4j, Objectivity and YarcData. Graph databases are designed for data whose relations are well represented as a graph, e.g., visual representations of social relationships, road maps or network topologies and representation of "ownership" for documents within an enterprise for legal or ediscovery purposes.

Hadoop and NoSQL

The Hadoop Distributed File System (HDFS) is an Apache open-source platform that enables applications, such as petabyte-scale Big Data analytics projects, to potentially scale across thousands of commodity servers such as Intel standard x86 servers, dividing up the workload.

HDFS includes components derived from Google's MapReduce and Google File System (GFS) papers as well as related open-source projects, including Apache Hive, a data warehouse infrastructure initially developed by Facebook and built on top of Hadoop to provide data summarization, query and analysis support; and Apache HBase and Apache Accumulo, both open-source NoSQL DBs, which, in the parlance of the CAP Theorem, are CP DBs and are modeled after the BigTable DB developed by Google. Facebook purportedly uses HBase to support its data-driven messaging platform while the National Security Agency (NSA) supposedly uses Accumulo for its data cloud and analytics infrastructure.

In addition to the HBase, MarkLogic 7 and Accumulo native integrations of HDFS, several NoSQL DBs can be used in conjunction with HDFS, whether they are open source and community supported or proprietary in nature, including Couchbase, MarkLogic, MongoDB or Oracle's version of NoSQL based on the Berkeley open-source DB. As Hadoop is inherently a batch-oriented paradigm, additional DBs to handle in-memory processing or real-time analysis are needed. Therefore, NoSQL – as well as RDBMS – solution providers have developed connectors for allowing data to be passed between HDFS and their DBs.



Major Changes: The Evolving Data Center

LOB App | LOB App | LOB App

Oracle | MySQL | SQL Server

"What's Happening?"
Hyper Velocity
Transactional

Cassandra/NoSQL

Data Warehouse

Teradata/ Exadata

"What Happened?"
Massive Volume
Bit Bucket

Hadoop

The slide above, courtesy of DataStax, illustrates how NoSQL and Hadoop solutions are transforming the way both transactional and analytic data are handled within enterprises with large volumes of data to manage both in real-time, or near real-time, and post-processing or after data is updated or archived.

NoSQL DB Funding and Growth

A recent note written by Wikibon's Jeff Kelly, Hadoop-NoSQL Software and Services Market Forecast 2012-2017, gives a good indication of how well funded and fast growing the market for RDBMS alternatives has become.

"The Hadoop/NoSQL software and services market reached $]$3.48 billion in 2017, a 45% CAGR [compound annual growth rate] during this five-year period." Kelly forecasts the NoSQL portion of the market to reach nearly $2 billion by 2017.

Kelly's research also indicates that the top ten companies in the space, measured in amount of funding dollars, received more the $600 million over the last 5 years, with funding increasing dramatically over the years. The top-funded NoSQL DB companies – in order of total funding amount – include DataStax (Cassandra), MongoDB, MarkLogic, MapR, Couchbase, Basho (creator of Riak), Neo Technology (creator of Neo4j) and Aerospike.

21 for 2020: NoSQL Innovators

As previously mentioned, there are now more than 50 vendors that have entered the NoSQL DB software and services space. As is the case with most nascent technology markets, more companies will emerge and others will buy their way into the market, fueling the inevitable surge of consolidation.

Oracle has publicly committed to its Berkeley DB open-source version of NoSQL, while IBM offers support for Hadoop and MongoDB solutions as part of its InfoSphere information management platform as well as Hadoop enhancements for its PureData System, and Microsoft supports a variety of NoSQL solutions on its Windows Azure cloud-based storage solution. Suffice to say, the big three RDBMS vendors are pragmatic about the future of databases. Sooner or later, expect them all to make NoSQL acquisitions.

Meanwhile, here is a short list of companies anticipated to disrupt the database space over the next 5 to 7 years arranged in somewhat different categories from the above NoSQL taxonomies and based more on use case within the enterprise than on data model.

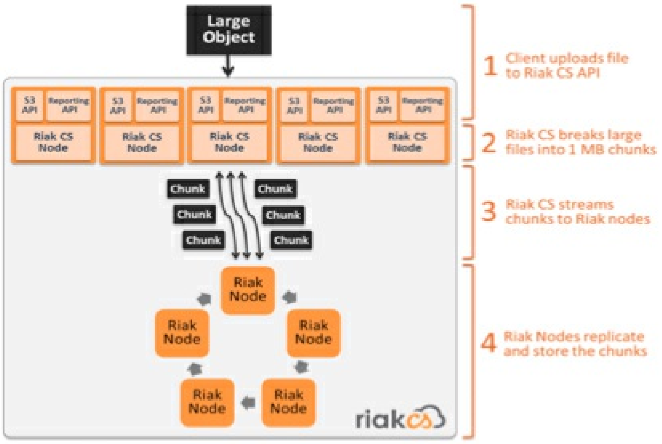
This group is also distinguished by added capabilities or functionality beyond just providing a simple data store with the inclusion of analytics, connectors (interoperability with other DBs and applications), data replication and scaling across commodity servers or cloud instances.

DISTRIBUTED DATABASES

This group is made up of four venture-backed, pure-play NoSQL vendors and two NoSQL alternative solutions in NuoDB and Virtue-Desk. Distributed databases are critical for web-scale or cloud-based applications where multiple nodes are needed to ensure availability and partition tolerance – an AP solution in CAP Theorem jargon. These databases also have capabilities consistent with other categories of DBs, including relational and graph. Use case examples include ecommerce applications for Amazon and other web stores, gaming programs such as Angry Birds, Netflix, and applications for government agencies and scientific research organizations.

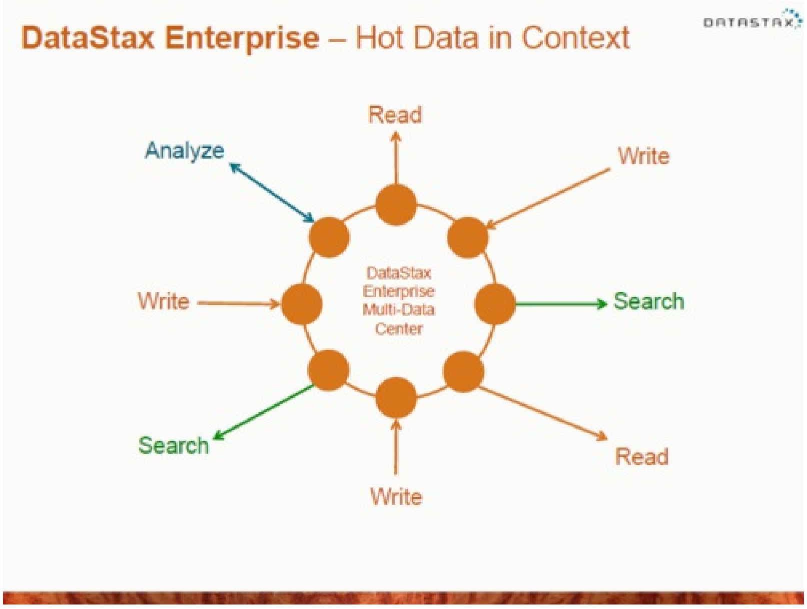
Basho

Basho is the progenitor and primary curator of Riak, a non-relational open-source DB (under the Apache license since 2011) based on Amazon Dynamo. With Riak CS (cloud storage) release 1.4 in March 2013, Basho boosts its distributed cloud enablement capabilities to allow customers to support well over 100 nodes, making it easier to distribute data. Riak CS VS (virtual store) also supports multi-tenant object stores for CSPs looking to extend their portfolios. Basho boasts a 6x performance advantage over the most popular NoSQL DBs, superior availability and data replication characteristics compared with any kind of database, and dramatic TCO (total cost of ownership) over traditional RDBMSs – as much as 80% savings. According to CTO Justin Sheehy, "One of Riak CS's key draws is that customers can use it to build private or public cloud storage that is API-compatible with Amazon S3."


Large Object
S3 API
Reporting API
Riak CS Node
Chunk
Riak Node
riak cs
1 Client uploads file to Riak CS API
2 Riak CS breaks large files into 1 MB chunks
3 Riak CS streams chunks to Riak nodes
4 Riak Nodes replicate and store the chunks

DataStax

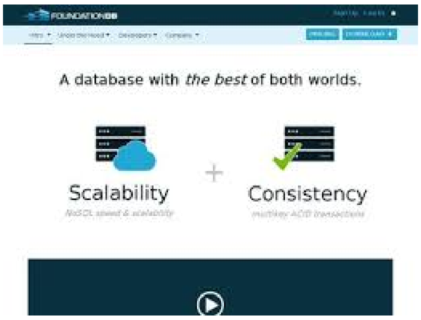
DataStax is laser focused on providing its customers with availability and performance when powering their online applications. The release of its Enterprise (DSE) 3.1 Cassandra-based platform offers more powerful tools and easier deployment capabilities that allow up to 10x more Cassandra data per node. According to CEO Billy Bosworth, DSE 3.1 improves "enterprise confidence" by providing analytics and search capabilities for online applications that rely on real-time data, such as tracking and displaying shopper search trends, with the ability to personalize data within the same cluster. DataStax is aggressively addressing security for the bulk of its customer use cases, including integration with Kerberos, and embracing industry standards such as Hive and ODBC for directing data between systems. A recent large investment round will help accelerate company growth worldwide.



DataStax Enterprise – Hot Data in Context

FoundationDB

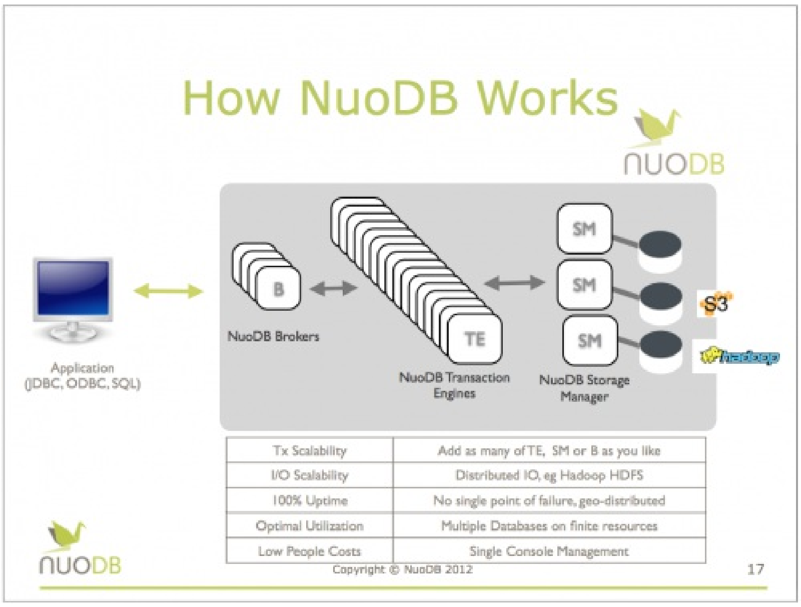
FoundationDB is a new entrant into the NoSQL space that emphasizes scalability and consistency while supporting ACID transactions and giving up "perfect" availability. Community and enterprise versions became generally available in August 2013. Co-founder David Rosenthal describes the process as "turning many computers into a single powerful database. By using a shared-nothing distributed architecture, FoundationDB scales out by adding more machines to a cluster rather than just scaling up by increasing capacity of a single machine." Utilizing a key-value store data model, FoundationDB has been providing its more than 2k beta users with multiple "real transactions" without sacrificing speed. Service contracts are priced starting at $99 per month per server process including support for document, graph and SQL and can be effectively used for both operational (OLTP) analytic (OLAP) workloads.



A database with *the best* of both worlds.

NuoDB

NuoDB describes itself as "the only distributed cloud database that is 100% SQL, 100% ACID and 100% elastically scalable." Launched in February 2013, NuoDB targets web-scale distributed processing analytics and OLTP use cases offering a single "logical" database across several virtual machines or geographies while maintaining the high performance expected from scale-up DB solutions. NuoDB touts its easy administration that allows users to "seamlessly" switch between on-premise and cloud deployments running on Azure or AWS and in Windows or Linux configurations. While not an in-memory DB, NuoDB's architecture allows for up to 100 transaction engines, which NuoDB states doubles the throughput with every added engine. The DB synchronizes storage nodes across all storage engines so each node has a full copy of the data to provide built-in high availability and replication capabilities.



How NuoDB Works

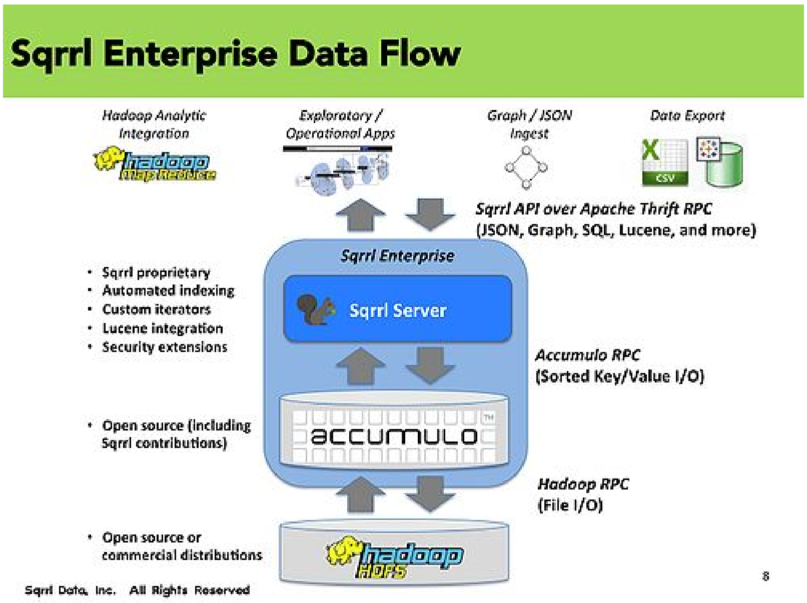
Tx Scalability	Add as many of TE, SM or B as you like
I/O Scalability	Distributed IO, eg Hadoop HDFS
100% Uptime	No single point of failure, geo-distributed
Optimal Utilization	Multiple Databases on finite resources
Low People Costs	Single Console Management

Copyright © NuoDB 2012

17

Sqrrl

Sqrrl] is the creator of Sqrrl Enterprise, “a highly secure, massively scalable Big Data platform powered by Apache Accumulo and Hadoop.” The genesis of Accumulo starts in 2008 with the National Security Agency’s (NSA) search for a petabyte-scale, BigTable-like (Google) data store with flexible schemas and fine-grained security features (aka “cell-level security”) that could very quickly handle both structured and unstructured data running on commodity hardware. Sqrrl extends Accumulo “with additional data ingest, security, and real-time analytical features.” Sqrrl Enterprise also cuts across multiple NoSQL categories by adding document store and graph store capabilities to Accumulo. Use cases include fraud and risk applications in government, financial services, healthcare, and cyber security. Accumulo is widely deployed and used throughout the U.S. Department of Defense and Intelligence agencies, and now has users across the Federal Government and in various commercial sectors.



Sqrrl Enterprise Data Flow

- Sqrrl proprietary
- Automated indexing
- Custom iterators
- Lucene integration
- Security extensions

- Open source (including Sqrrl contributions)

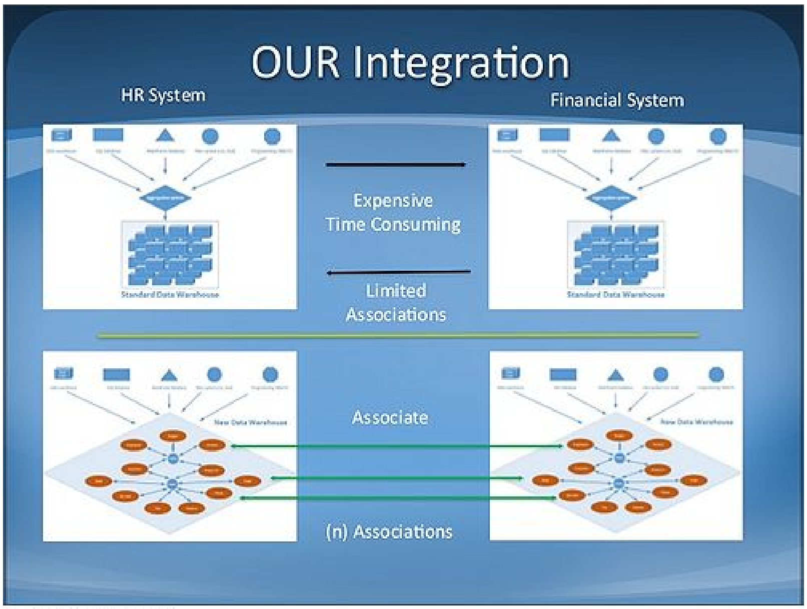
- Open source or commercial distributions

Sqrrl Data, Inc. All Rights Reserved

8

Virtue-Desk

Virtue-Desk provides what it refers to as an “associative” database, which it claims is 100x faster than SQL on reads. Dubbed AtomicDB, the solution has no views, no indexes, no tables and no whitespace, and there are no queries to write. Used for more than a decade by the U.S. Navy to keep track of 70 million parts and patented in 2011, AtomicDB extends its practical use to healthcare and financial services applications, including multi-petabyte-scale document stores, and is intended to be used by business analysts and other non-technical users. The system does not “hold” data but “associations” with the data, and “the data is the structure” left resident in its existing location with application-class libraries acting as APIs. CTO Jean Michel LeTennier states, “AtomicDB is much simpler and more cost effective to implement than traditional RDBMSs or even NoSQL solutions.”



OUR Integration

HR System

Financial System

Standard Data Warehouse

Expensive
Time Consuming

Limited
Associations

Standard Data Warehouse

New Data Warehouse

Associate

New Data Warehouse

(n) Associations

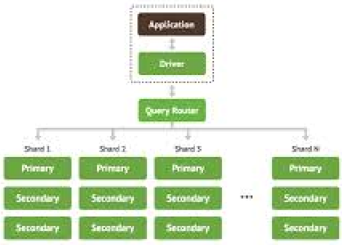
Wednesday, August 28, 13

DOCUMENT-ORIENTED DATABASES

As the category name implies, DBs in this group – also referred to as document stores – are optimized to handle documents and other forms of unstructured or semi-structured data such as emails, instant messages and the like. As with non-relational DBs in other categories, document-oriented DBs can also possess attributes associated with other DB categories, including distributed nodes, graph capabilities and near real-time analytics.

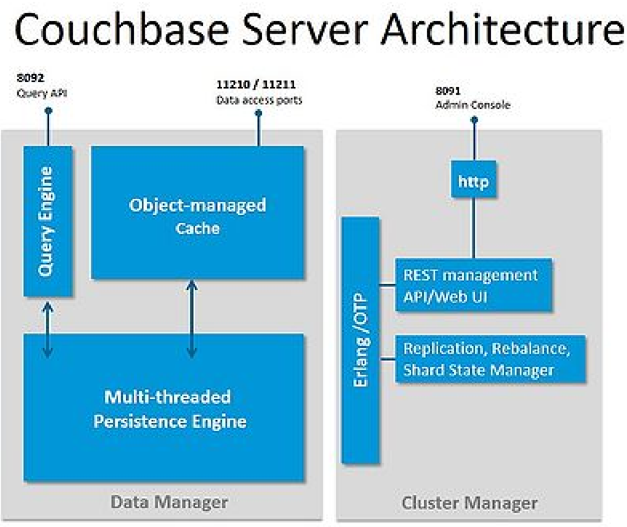
MongoDB (Formerly 10gen)

MongoDB, Inc. is the developer of MongoDB, which has the largest community of any open-source database distribution. MongoDB executives attribute the large following to a few key attributes: ease of solving both easy and hard problems, agile development, sufficient for most workloads and a transparent business model that makes it easy to do business. MongoDB's enterprise version includes Kerberos authentication, SNMP support and user training. Both community and enterprise versions feature a JSON data model with dynamic schemas for improved document handling, auto-sharding of objects to enable horizontal scaling, replication and availability, and rich document querying and search capabilities. MongoDB is deployable on-premise, in the cloud or as a hybrid solution, is supported by IBM services and is the most popular NoSQL DB on Amazon AWS. MongoDB announced October 4, 2013 that it secured $150 million, the largest funding round ever for any Database vendor – NoSQL or otherwise.



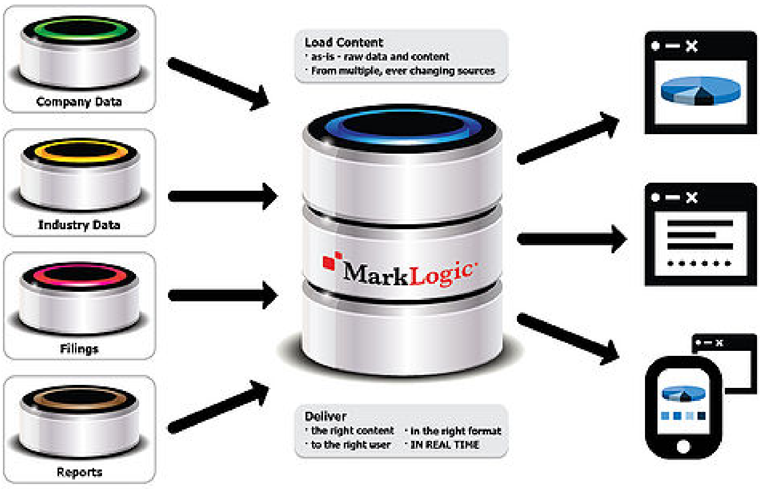
Couchbase

Couchbase is laser focused on being "the platform of choice for the most demanding web and mobile applications at the world's largest enterprises." Couchbase Server is a NoSQL document database optimized for interactive web and mobile applications. A flexible JSON data model makes it easy to modify applications without being constrained by a fixed database schema. "With sub-millisecond, high-throughput reads and writes, Couchbase Server delivers consistent high performance for web and mobile apps. It is easy to scale out, and supports live cluster topology changes with zero downtime." Couchbase recently announced its JSON anywhere mobile strategy with the first NoSQL database for mobile. A recent funding round brings total money raised to $55 million. Funds will be used to further expand international sales and marketing operations and support key strategic product initiatives.



MarkLogic

MarkLogic is the market share leader in the Hadoop/NoSQL market segment as measured by Wikibon. A dozen years ago, MarkLogic embraced XML and XQuery as document markup and access standards for multi-terabyte scale collections. Today, MarkLogic Server ingests a variety of other document formats, including PDF and JSON, is "schema-agnostic" and has developed or partners with a variety of query, search and analytics programs to find information within enterprises' document stores. Some clients report that they have replaced SQL with XQuery although MarkLogic supports both, as well as keyword and faceted searches, enabling non-technical users to more easily find information within documents or search meta-data associated with images, sound and multimedia files. MarkLogic supports ACID transactions and has developed a REST API, a native Hadoop bi-directional connector and semantic indexing, search and query capabilities along with other enhancements to support its latest 7.0 release. (Click here to view an independent assessment of MarkLogic 7.)



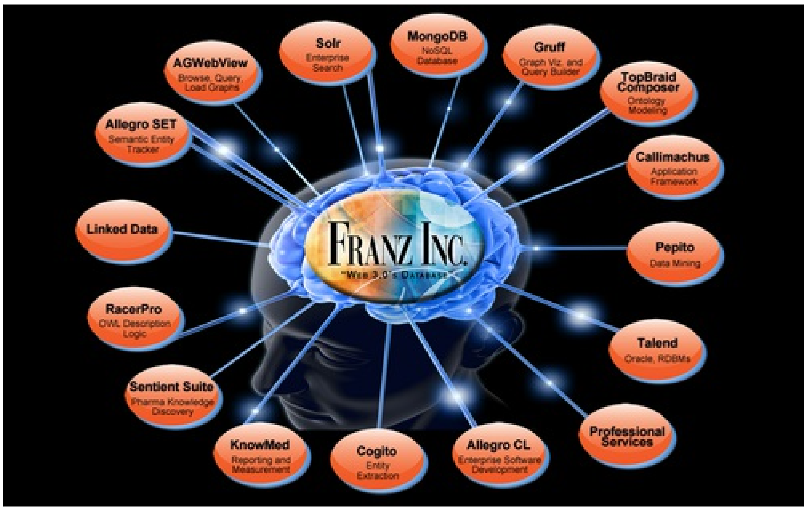
GRAPH NoSQL DATABASES

While databases outside this category, such as Sqrrl, support graph capabilities, these 4 providers specialize in this segment. Some graph DB vendors are also appropriate for use cases beyond purely graph database apps, including YarcData, which has an in-memory discovery analytics capability. Graph DBs are often paired with other types of databases to dramatically improve performance and relevance for e-commerce, fraud detection or knowledge-based applications.

Franz Inc.

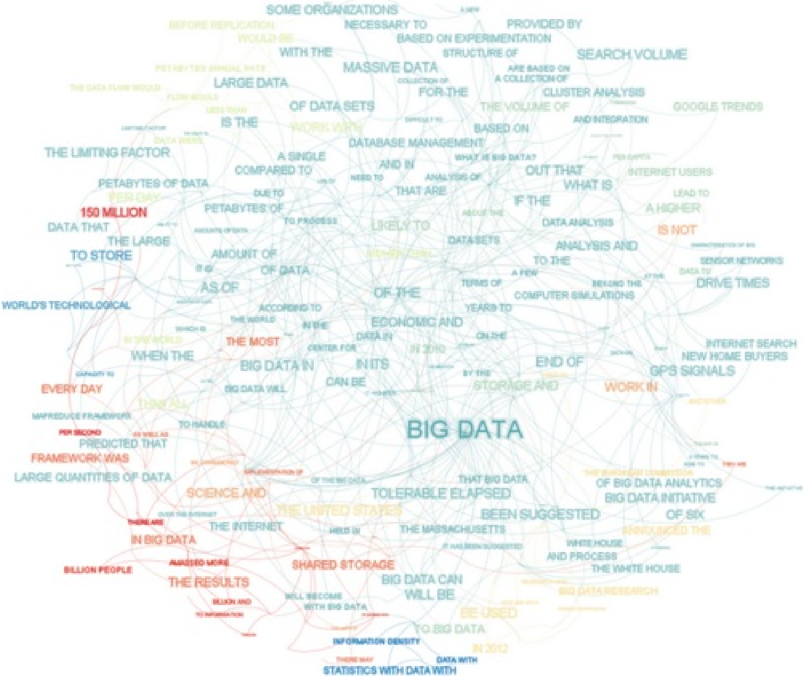
Franz Inc. has "expert knowledge in developing and deploying Semantic Web technologies (i.e., Web 3.0) and providing Common Lisp (programming language)-based tools that offer an ideal environment to create complex, mission-critical applications." AllegroGraph and Allegro CL, Franz's object-oriented development system, with AllegroCache are distinct scalable platforms used by startups and

Fortune 100 companies for knowledge-based applications or for social media analysis. “AllegroGraph is a modern, enterprise, high-performance, persistent graph database. It uses efficient memory utilization in combination with disk-based storage, enabling it to scale to billions of quads while maintaining superior performance.” AllegroGraph supports SPARQL, Jena, Sesame, ACID Compliant, RDFS++, and Prolog reasoning from numerous client applications. Franz also provides training, services and support for Lisp-based programming environments and has built connectors to MongoDB and other popular databases as well as to search and BI (business intelligence) tools.



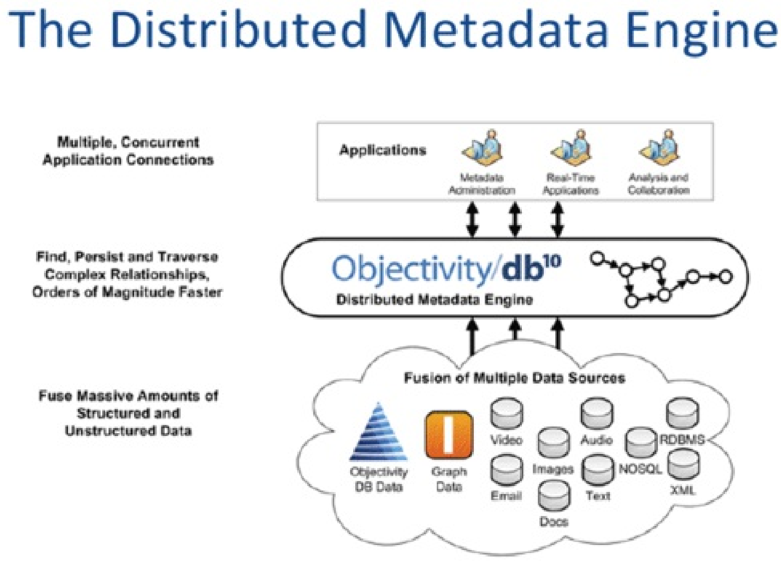
Neo Technology

Neo Technology developed, open-sourced and now supports Neo4j, which has the largest ecosystem of any graph database, with over 500,000 downloads. Its enterprise version supports high-availability clustering, ACID requirements and delivers what Neo4j’s CEO Emil Eifrem refers to as a “run-time, real-time transaction environment” for OLTP and other mission-critical use cases. Eifrem believes the most powerful cognitive model for developing relationships between seemingly disparate data types is the whiteboard, and the Neo4j graph model mimics that whiteboard friendliness. “Query performance with connected data sets can literally be 1,000 times faster than traditional DBs because it’s a native graph database.” Social networking, identity & access management, geo routing, dependency analysis and fraud detection apps have all adopted graph DBs due to their speed and ease of use. In Eifrem’s view, the need for fast, intuitive, visually compelling applications is driving their growth. Neo4j also works well with several NoSQL DBs.



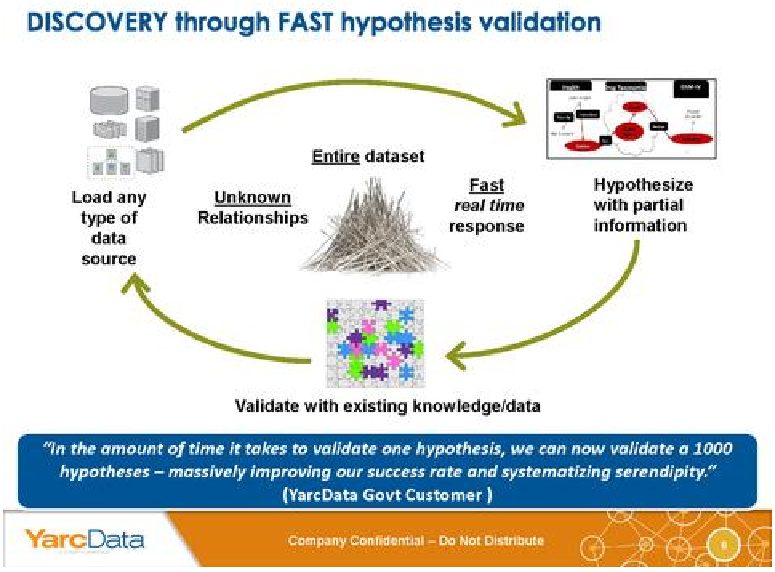
Objectivity

Objectivity brings together its flagship Objectivity/DB and InfiniteGraph solutions to address the data and systems requirements of web-scale environments. “Objectivity supports computing across vast distributed networks or embedded stand-alone devices that simply must not fail, enables persistent object management, virtually instantaneous traversal of complex, many-to-many relationships and graphs." InfiniteGraph is supported by a scale-out, distributed architecture as is Objectivity/DB, which is an object management-oriented DB. Objectivity believes its "unique" distributed approach to graph technology is unmatched in the industry, combining InfiniteGraph's strengths of "persisting and traversing complex relationships requiring multiple hops, across vast and distributed data stores." Oracle is a partner, and clients include U.S. Armed Services.


The Distributed Metadata Engine
Multiple, Concurrent Application Connections
Find, Persist and Traverse Complex Relationships, Orders of Magnitude Faster
Fuse Massive Amounts of Structured and Unstructured Data
Applications
Metadata Administration
Real-Time Applications
Analysis and Collaboration
Objectivity/db10
Distributed Metadata Engine
Fusion of Multiple Data Sources
Objectivity DB Data
Graph Data
Video
Email
Images
Docs
Audio
Text
NOSQL
RDBMS
XML

YarcData

YarcData focuses on in-memory discovery analytics as opposed to just search. A wholly owned subsidiary of supercomputer manufacturer Cray Inc., YarcData turnkey appliances help solve complex Big Data problems suitable for graph DBs. Its purpose-built Urika appliance has 512 TB of shared memory along with 8,000 processors that offer a performance boost of 2 to 4 orders of magnitude over traditional RDBMSs. Urika is particularly well suited for sifting through massive amounts of unstructured or rich text data sets as its triple store database architecture – similar to the Semantic Web – is ideal for uncovering hidden relationships within constantly changing and varied data sources. Use cases include personalized and evidence-based medicine, fraud detection, cyber security, financial risk management, and baseball analytics.


DISCOVERY through FAST hypothesis validation
Load any type of data source
Unknown Relationships
Entire dataset
Fast real time response
Hypothesize with partial information
Validate with existing knowledge/data
YarcData
Company Confidential – Do Not Distribute

"In the amount of time it takes to validate one hypothesis, we can now validate a 1000 hypotheses – massively improving our success rate and systematizing serendipity."
(YarcData Govt Customer)

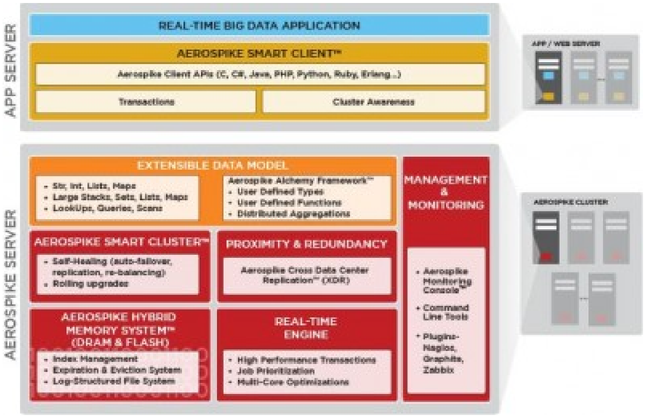
IN-MEMORY NoSQL DATABASES

In-memory databases are typically faster than disk storage-optimized databases because they rely primarily on processing data within a computer's main memory (DRAM). This obviates the need to swap data in and out of memory from hard disk storage devices (HDD) and eliminates seek time when querying data. A major game-changer for database architectures has been the introduction of non-volatile storage devices (no moving parts) such as Flash or SSD drives. In-memory DBs can also have attributes associated with other NoSQL categories e.g., Aerospike is also a distributed, highly available NoSQL DB.

Flash drives can be 100x faster than traditional HDD spinning disk drives and 10x (or more) smaller with the same capacity, allowing for increasingly larger data sets to be stored and managed in-memory – or in very close proximity to it. Flash storage is also much more affordable than DRAM. Due to their speed and compact form factor, multi-terabyte Flash drives are now being installed on computer PCIe (peripheral component interconnect express) and DIMM (dual in-line memory module) cards as well. Real-time, or near real-time updates are possible with in-memory solutions that are able to handle a mixture of live and archived data based on use cases such as online transactions.

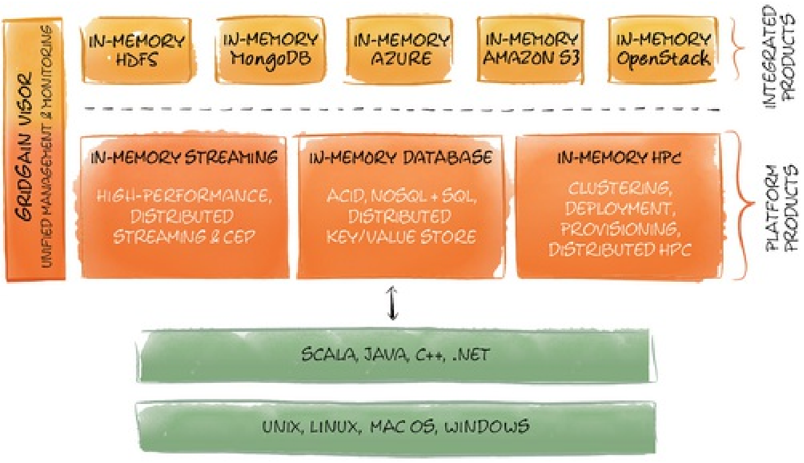
Aerospike

Aerospike is an in-memory and a distributed NoSQL DB used for hyper-scale low latency applications that need 100% availability. Aerospike takes a hybrid approach by blending computer memory (DRAM) and Flash – indexes are stored in DRAM and data can be stored in DRAM or Flash memory accessed via a native file system. "Aerospike 3 with queries, user defined functions and aggregations greatly simplifies deployment by compressing the database stack, processing all data within one database layer and eliminating the need for caching and queuing technologies." Aerospike believes simplicity is the way to scale: clusters can be smaller and fewer components need to be kept in sync, therefore, applications are easier to manage while maintaining availability as new nodes are added or servers fail. Aerospike is deployed as a user profile store by real-time bidding firms and other platforms in advertising, ecommerce, mobile and gaming. Aerospike also follows the ACID-compliant transaction standard.


APP SERVER
REAL-TIME BIG DATA APPLICATION
AEROSPIKE SMART CLIENT™
Aerospike Client APIs (C, C#, Java, PHP, Python, Ruby, Erlang...)
Transactions
Cluster Awareness
APP / WEB SERVER
AEROSPIKE SERVER
EXTENSIBLE DATA MODEL
- Str, Int, Lists, Maps
- Large Stacks, Sets, Lists, Maps
- LookUps, Queries, Scans
Aerospike Alchemy Framework™
- User Defined Types
- User Defined Functions
- Distributed Aggregations
MANAGEMENT & MONITORING
AEROSPIKE SMART CLUSTER™
- Self-Healing (auto-failover, replication, re-balancing)
- Rolling upgrades
PROXIMITY & REDUNDANCY
Aerospike Cross Data Center Replication™ (XDR)
AEROSPIKE HYBRID MEMORY SYSTEM™ (DRAM & FLASH)
- Index Management
- Expiration & Eviction System
- Log-Structured File System
REAL-TIME ENGINE
- High Performance Transactions
- Job Prioritization
- Multi-Core Optimizations
- Aerospike Monitoring Console™
- Command Line Tools
- Plugins- Nagios, Graphite, Zabbix
AEROSPIKE CLUSTER

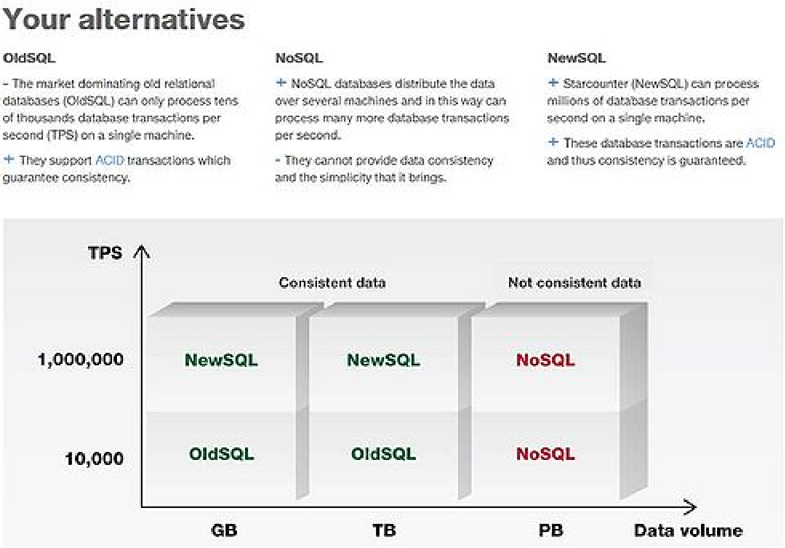
GridGain

GridGain is determined to change the way companies compute. "Just putting data online in-memory is not transformational enough" says GridGain VP of Product Management Jon Webster. "Making Flash look like memory is key. Send the compute to the data. Then memory and data are local to the processing. With GridGain, data movement is minimized." In business for just two years, GridGain's in-memory DB platform has been in development since 2005, attracting a number of household names as clients along with $10 million in a recent funding round. GridGain offers two classes of products: a high performance (HPC) in-memory computational model for risk analysis utilizing historical and streaming data, and accelerators for enabling a new class of in-memory products to enhance the performance of popular open-source solutions, including MongoDB and HDFS.



Starcounter

Starcounter is a "scale-in," NewSQL, in-memory DB capable of 3 million web requests on a single server. With ACID compliance, Starcounter 2.0 performs up to 300k writes per second utilizing a multicore server and 100k ACID write TPS on one core and scales reading transactions linearly with 500k TPS per core – making it the "world's fastest consistent" DB. Starcounter has also delivered a new solution dubbed VMDBMS, which is an integration between the application run-time virtual machine and the DBMS. "This makes our solution substantially faster than other in-memory, high-performance databases because data resides all the time in RAM and is not copied back and forth between the database and the application. A native object .NET API completely removes the object-relational mapping (ORM) and reduces the lines of code needed to implement an application." Integrated into Starcounter is a web service supporting JSON/Rest, "enabling performance all the way from the core database out to the end user clients."



Your alternatives

OldSQL

- The market dominating old relational databases (OldSQL) can only process tens of thousands database transactions per second (TPS) on a single machine.

+ They support ACID transactions which guarantee consistency.

NoSQL

+ NoSQL databases distribute the data over several machines and in this way can process many more database transactions per second.

- They cannot provide data consistency and the simplicity that it brings.

NewSQL

+ Starcounter (NewSQL) can process millions of database transactions per second on a single machine.

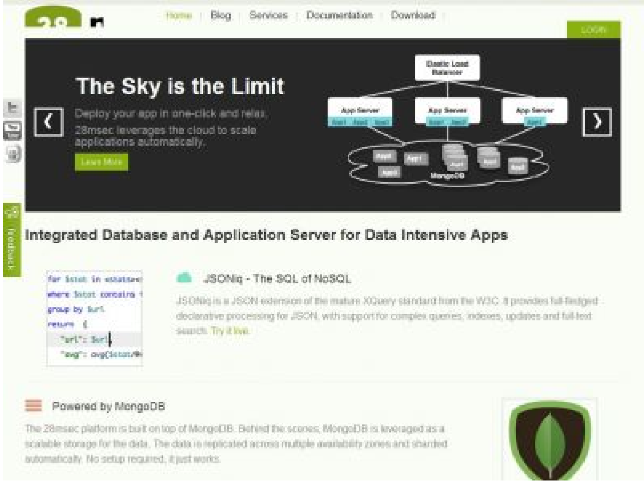
+ These database transactions are ACID and thus consistency is guaranteed.

NoSQL DATABASE SOLUTIONS and SERVICES

While all of the 21 companies profiled in this report provide database related solutions and some level of enterprise services, this group is characterized by its diversity of services provided from startups who have developed NoSQL integration tools enabling existing DB solutions, to established services firms specializing in DB implementations and consulting services, to Oracle which has dominated the RDBMS landscape for two decades. Oracle could make a good argument for being included in at least 4 categories as they offer a variety of database options including relational, open-source MySQL (acquired in the Sun deal), a version of open-source Berkeley DB and a variety of business intelligence, DB query tools and high performance storage solutions. Since Oracle defies classification, by default they land here.

28msec

28msec delivers "information agility via an Information Processing Platform that quickly extracts data out of any source and transforms that data into a valuable commodity – actionable information." Sold as a service or software, 28.io is a query "layer" designed to consolidate data silos leveraging JSON, XML, relational, object or flat-file protocols and data formats. CTO Matthias Brantner calls the platform "an accelerant for NoSQL. 28.io doesn't store data; it connects databases. Customers get a 360° view of their data when they write one query that goes to all databases – and Facebook and Twitter if required." When dealing with large datasets, single queries are parallelized and automatically balanced across multiple servers. Founders include senior Oracle architects and the author of XQuery and JSONiq. A key partnership is with the creator of XBRL financial reporting.



The Sky is the Limit

Deploy your app in one-click and relax. 28msec leverages the cloud to scale applications automatically.

Integrated Database and Application Server for Data Intensive Apps

```
for $stat in $statistics
where $stat contains (
group by $url
return {
  "url": $url,
  "avg": avg($stat/@
```

JSONiq - The SQL of NoSQL

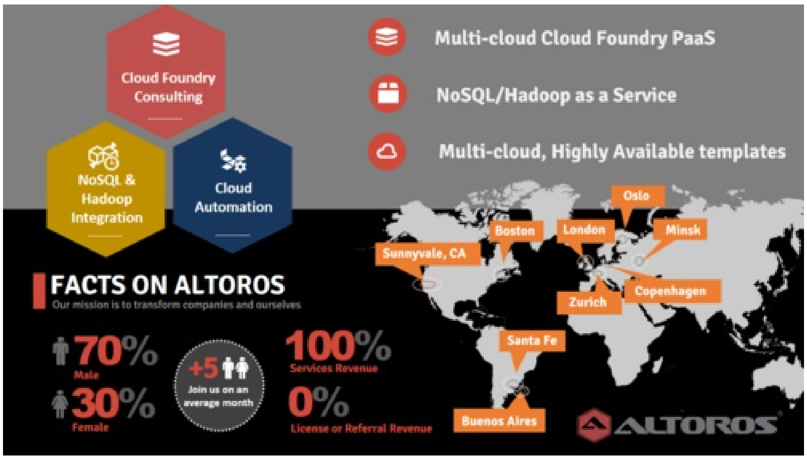
JSONiq is a JSON extension of the mature XQuery standard from the W3C. It provides full-fledged declarative processing for JSON, with support for complex queries, indexes, updates and full-text search. Try it live.

Powered by MongoDB

The 28msec platform is built on top of MongoDB. Behind the scenes, MongoDB is leveraged as a scalable storage for the data. The data is replicated across multiple availability zones and sharded automatically. No setup required, it just works.

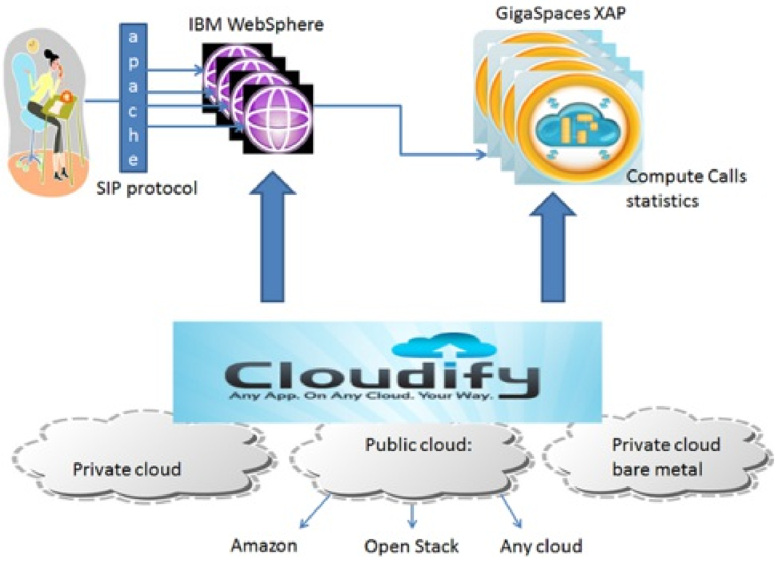
Altoros

Altoros is a vendor-independent services organization focused on Hadoop/NoSQL/Cloud Foundry PaaS enablement. Altoros configures open-source cloud platforms such as Cloud Foundry, query optimization tools, OS/hypervisors, NoSQL/Hadoop clusters on AWS, OpenStack and vShere clouds. Altoros has completed more than 25 performance benchmarks of various Hadoop, NoSQL and NewSQL solutions to support organizations of any size seeking intelligence and deployment advice. CEO Renat Khasanshyn believes that platform as a service (PaaS) solutions, such as Cloud Foundry, make infrastructure as a service (IaaS) a commodity: “While IaaS brings value, PaaS could bring 2x to 3x as much, while removing lock-in into IaaS platforms.” The Altoros team of 300 engineers and consultants is split between the U.S. and Europe and Latin America. Partners include Pivotal, Cloudera and Hortonworks.


Cloud Foundry Consulting
NoSQL & Hadoop Integration
Cloud Automation
Multi-cloud Cloud Foundry PaaS
NoSQL/Hadoop as a Service
Multi-cloud, Highly Available templates
Oslo
Boston
London
Minsk
Sunnyvale, CA
Zurich
Copenhagen
Santa Fe
Buenos Aires
FACTS ON ALTOROS
Our mission is to transform companies and ourselves
70%
Male
30%
Female
+5
Join us on an average month
100%
Services Revenue
0%
License or Referral Revenue
ALTOROS

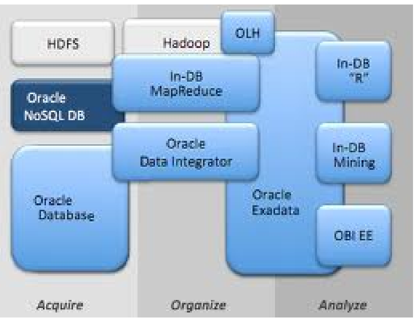
GigaSpaces

GigaSpaces is “the pioneer of a new generation of application virtualization platforms and a leading provider of end-to-end scaling solutions for distributed, mission-critical application environments and cloud enabling technologies.” GigaSpaces’ complementary solutions include XAP Elastic Application Platform, an in-memory data grid, and Cloudify, an open-source PaaS solution developed by GigaSpaces that quickly moves organizations to the cloud with little or no application code changes necessary. According to CTO Nati Shalom, “Cloudify provides the equivalent of Amazon OpenWorks on OpenStack without the vendor lock-in.” OpenStack is the most widely used open-source software for building private and public clouds. GigaSpaces offers a silo-free architecture along with deployment services to support rapid adoption of cloud-based, high-performance applications.



Oracle

Oracle is the market share leader for RDBMS and open-source SQL-centric databases. Oracle's foray into the NoSQL space is based on the Berkeley DB open-source distribution. "Berkeley DB provides a collection of well-proven, building-block technologies that can be configured to address any application need, from the hand-held device to the datacenter, from local solutions or worldwide distributions, from KBs to PBs." Director of Product Management David Segleau described "the first NoSQL appliance," which includes up to 300 TB of disk and a starter rack with 6 dual-core servers with redundant Infiniband switches, and which offers the Cloudera distribution, including Apache Hadoop to acquire and organize data. Segleau states, "We continue to enhance our NoSQL key-value store (JSON or Graphic) with enterprise-class features, including auto-failover, sharding, query load balancing, smart topology and data distribution."



Qubole

Qubole is a "Big Data as a Service" solution. Founded by the creators of Apache Hive and former managers of Facebook's data infrastructure team, Qubole is used by some of the largest brands in social media, online advertising, gaming and other data-intensive enterprise organizations and also completed an initial funding round of $7 million earlier this year. Its flagship 100% cloud-based solution, Qubole Data Service (QDS), "provides a fast, auto-scaling Hadoop service built for the cloud, with built-in data connectors and a graphical user-interface for Hive, Pig, Oozie and Sqoop – all integrated in an easy-to-use and easy-to-operate web service." Qubole's turnkey solution has a flexible, pay-as-you-go model that provides end-users with the ability to scale up and down as needed without the need of a technical team or a large capital expenditure.

[

The Stages of Data Warehousing

Steps to Implement Data Warehouse

The best way to address the business risk associated with a Datawarehouse implementation is to employ a three-prong strategy as below

1. **Enterprise strategy**: Here we identify technical including current architecture and tools. We also identify facts, dimensions, and attributes. Data mapping and transformation is also passed.
2. **Phased delivery**: Datawarehouse implementation should be phased based on subject areas. Related business entities like booking and billing should be first implemented and then integrated with each other.
3. **Iterative Prototyping**: Rather than a big bang approach to implementation, the Datawarehouse should be developed and tested iteratively.

STEP	TASKS	DELIVERABLES
1	Need to define project scope	Scope Definition
2	Need to determine business needs	Logical Data Model
3	Define Operational Datastore requirements	Operational Data Store Model
4	Acquire or develop Extraction tools	Extract tools and Software
5	Define Data Warehouse Data requirements	Transition Data Model
6	Document missing data	To Do Project List
7	Maps Operational Data Store to Data Warehouse	D/W Data Integration Map
8	Develop Data Warehouse Database design	D/W Database Design
9	Extract Data from Operational Data Store	Integrated D/W Data Extracts

STEP	TASKS	DELIVERABLES
10	Load Data Warehouse	Initial Data Load
11	Maintain Data Warehouse	On-going Data Access and Subsequent Loads

DW Jargon Busting

1. **Business Function** Something an enterprise does, or needs to do, in order to achieve its objectives.
2. **Business Meta data** The information whereby users can understand and access the data warehouse. It focuses on what data is in the warehouse, how it was transformed, the source, and the timeliness of the data.
3. **Business Process** The complete response that a business makes to an event. A business process entails the execution of a sequence of one or more process steps. It has a clearly defined deliverable or outcome. A Business Process is defined by the business event that triggers the process, the inputs and outputs, all the operational steps required to produce the output, the sequential relationship between the process steps, the business decisions that are part of the event response, and the flow of material and/or information between process steps.
4. **Central Repository** Location of a collection of documentation, customizations, modifications, or enhancements designed to alleviate the recreation of successfully completed work.
5. **Data Acquisition** The process of extracting, transforming, and transporting data from the source systems and external data sources to the data warehouse database objects.
6. **Database** A collection of data, usually in the form of tables or files, under the control of a database management.
7. **Data Extraction** The process of pulling data from operational and external data sources in order to prepare the source data for the data warehouse environment.
8. **Data Integrity** The quality of the data residing in the database objects. The measurement which users consider when analyzing the value and reliability of the data.
9. **Data Transformation** The process of redefining data based on some predefined rules. The values are redefined based on a specific formula or technique.
10. **Data Warehouse** An enterprise structured repository of subject-oriented, time-variant, historical data used for information retrieval and decision support. The data warehouse stores atomic and summary data. The data warehouse is the source data stored in the data marts.
11. **Data Warehousing:**The process of designing, building, and maintaining a data warehouse system.
12. **Data Warehouse Integration** The process on reconciling each data warehouse increment with the strategic data warehouse architecture.
13. **Data Warehouse Method (DWM)** A structured method for full life-cycle custom development data warehouse projects.
14. **Deliverable** A tangible, measurable output of a task.
15. **Enterprise** A group of departments, divisions, or companies which make up an entire business.
16. **Entity** A thing of significance, whether real or imagined, about which information eeds to be known or held. It is implemented in a database as one or more tables.
17. **Extraction,** Transformation and Loading (ETL) Tool: Software that is used to extract data from a data source like a operational system or data warehouse, modify the data and then load it into a data mart, data warehouse or multi-dimensional data cube.
18. **Implementation** The installation of an increment of the data warehouse solution that is complete, tested, operational, and ready. An implementation includes all necessary software, hardware, documentation, and all required data.
19. **Iterative Development** The application of a cyclic, evolutionary approach to the development of requirements definition, design, or construction using prototyping and iterative build techniques.
20. **Meta data** Also known as data about data is the information about the contents and uses of the data warehouse. Meta data is created by several components of the data warehouse and provides a business and technical view of the data warehouse solution.
21. **On-Line Analytical Processing (OLAP)** On-line retrieval and analysis of data to reveal business trends and statistics not directly visible in the data directly retrieved from a data warehouse. Also known as multidimensional analysis.
22. **Relational Online Analytical Processing (ROLAP):** OLAP software that employs a relational strategy to organize and store the data in its database

23. **Repository** A mechanism for storing any information about the definition of a system at any point in its life-cycle. Repository services would typically be provided for extensibility, recovery, integrity, naming standards, and a wide variety of other management function.
24. **Structured Query Language (SQL)** The ANSI internationally accepted standard for relational database systems, covering not only query but also data definition, manipulation, security, and some aspects of referential and entity integrity.
25. **Target Database** The data warehouse database object that is to store the source data once it is extracted, transformed and transported.

Chapter 3

Heading placeholder

Contents placeholder

More contents

Chapter 4

Heading placeholder

Contents placeholder

More contents

Chapter 5

Heading placeholder

Contents placeholder

More contents