ComputerWeekly.com | TechTarget

E-guide

CW+ Content

Jargon Buster Guide to

Database Management

In this e-guide:

Database management roles have long been a rich career seam for IT professionals. However, there has been, in recent years, a "Cambrian explosion" in the previously steady but sleepy world of databases.

The relational database model, breaking the traditional dependency between data storage and applications, held sway in enterprise IT for decades, from the time of its serious commercialisation by Oracle in the mid-7os. Also hegemonic has been the language implementation of the relational model, SQL – Structured Query Language; which is of the same mid-70s vintage.

But, in recent years, with the rise of the big data technologies of the Hadoop stack and the NoSQL databases, the data store and database scene has become more complex.

And when it comes to the Oracle database, its creator announced it to be "autonomous" at Oracle Open World in San

Francisco October 2017. What does this mean for database administrators (DBAs)? Does it mean the opening up of new roles of higher intellectual interest and value? Or does it mean redundancy and early retirement?

As always with IT, the technology is only one-third of the story – the others being process and, crucially, people. What are the operational and organisational design problems to be solved in this more heterogenous database management landscape?

This Jargon Buster e-guide to contemporary database management is an aid to stepping back and thinking afresh about an area of enterprise IT that has many decades of heritage. Or, as many sometimes (arguably rudely) put it, legacy.

Brian McKenna, business applications editor

Definitions

Relational database

Margaret Rouse, guest contributor

A relational database is a set of formally described tables from which data can be accessed or reassembled in many different ways without having to reorganize the database tables. The standard user and application programming interface (API) of a relational database is the Structured Query Language (SQL). SQL statements are used both for interactive queries for information from a relational database and for gathering data for reports.

What's in a relational database model?

The relational database was invented in 1970 by E. F. Codd, then a young programmer at IBM. In his paper, "A Relational Model of Data for Large Shared Data Banks," Codd proposed shifting from storing data in hierarchical or navigational structures to organizing data in tables containing rows and columns.

Each table, which is sometimes called a *relation*, in a relational database contains one or more data categories in columns, also called *attributes*. Each row, also called a *record* or *tuple*, contains a unique instance of data, or *key*, for the categories defined by the columns. Each table has a unique primary key, which identifies the information in a table. The relationship between tables can then be set via the use of foreign keys -- a field in a table that links to the primary key of another table.

For example, a typical business order entry database would include a table that described a customer with columns for name, address, phone number and so forth. Another table would describe an order: product, customer, date, sales price and so forth. A user of a relational database can then obtain a view of the database to fit their needs. For example, a branch office manager might like a view or report on all customers that bought products after a certain date. A financial services manager in the same company could, from the same tables, obtain a report on accounts that need to be paid.

When creating a relational database, you can define the domain of possible values in a data column and further constraints that may apply to that data value. For example, a domain of possible customers could allow up to 10 possible customer names but be constrained in one table to allowing only three of these customer names to be specifiable. Two constraints relate to data integrity and the primary and foreign keys:

- **Entity integrity** ensures that the primary key in a table is unique and that the value is not set to null.
- **Referential integrity** requires that every value in a foreign key column will be found in the primary key of the table from which it originated

Types of databases

There are a number of database categories, from basic flat files that aren't relational to NoSQL to newer graph databases that are considered even more relational than standard relational databases.

A flat file database consists of a single table of data that has no interrelation -- typically text files. This type of file enables users to specify data attributes, such as columns and data types.

Standard relational databases enable users to manage predefined data relationships across multiple databases. Popular relational databases include Microsoft SQL Server, Oracle Database, MySQL and IBM DB2. Cloud-based relational databases, or database as a service (DBaaS), are also widely used because they enable companies to outsource database maintenance, patching and infrastructure support requirements. Cloud relational databases include Amazon Relational Database Service (RDS), Google Cloud SQL, IBM DB2 on Cloud, Microsoft Azure SQL Database and Oracle Database Cloud Service.

A NoSQL database is an alternative to relational databases that's especially useful for working with large sets of distributed data. These databases can support a variety of data models, including key-value, document, columnar and graph formats.

A graph database expands beyond traditional column- and row-based relational data models; this NoSQL database uses nodes and edges that represent connections between data relationships and can discover new relationships between the data. Graph databases are more sophisticated than relational databases, and thus, their uses include fraud detection or web recommendation engines.

Advantages of relational databases

The main advantages of relational databases are that they enable users to easily categorize and store data that can later be queried and filtered to extract specific information for reports. Relational databases are also easy to extend and aren't reliant on physical organization. After the original database creation, a new data category can be added without all existing applications being modified.

Other relational database advantages include:

- **Accurate:** Data is stored just once, which eliminates data deduplication.
- **Flexible:** Complex queries are easy for users to carry out.
- **Collaborative:** Multiple users can access the same database.
- **Trusted:** Relational database models are mature and well-understood.

- **Secure:** Data in tables within relational database management systems (RDBMSes) can be limited to allow access by only particular users.

Next Article

In this e-guide

Database Management System (DBMS)

Margaret Rouse, guest contributor

A database management system (DBMS) is system software for creating and managing databases. The DBMS provides users and programmers with a systematic way to create, retrieve, update and manage data.

A DBMS makes it possible for end users to create, read, update and delete data in a database. The DBMS essentially serves as an interface between the database and end users or application programs, ensuring that data is consistently organized and remains easily accessible.

The DBMS manages three important things: the data, the database engine that allows data to be accessed, locked and modified -- and the database schema, which defines the database's logical structure. These three foundational elements help provide concurrency, security, data integrity and uniform administration procedures. Typical database administration tasks supported by the DBMS include change management, performance monitoring/tuning and backup and recovery. Many database management systems are also responsible for automated rollbacks, restarts and recovery as well as the logging and auditing of activity.

The DBMS is perhaps most useful for providing a centralized view of data that can be accessed by multiple users, from multiple locations, in a controlled manner. A DBMS can limit what data the end user sees, as well as how that end user can view the data, providing many views of a single database schema. End users and software programs are free from having to understand where the data is physically located or on what type of storage media it resides because the DBMS handles all requests.

The DBMS can offer both logical and physical data independence. That means it can protect users and applications from needing to know where data is stored or having to be concerned about changes to the physical structure of data (storage and hardware). As long as programs use the application programming interface (API) for the database that is provided by the DBMS, developers won't have to modify programs just because changes have been made to the database.

With relational DBMSs (RDBMSs), this API is SQL, a standard programming language for defining, protecting and accessing data in a RDBMS.

Popular types of DBMSes

Popular database models and their management systems include:

Relational database management system (RDMS) - adaptable to most use cases, but RDBMS Tier-1 products can be quite expensive.

NoSQL DBMS - well-suited for loosely defined data structures that may evolve over time.

In-memory database management system (IMDBMS) - provides faster response times and better performance.

Columnar database management system (CDBMS) - well-suited for data warehouses that have a large number of similar data items.

Cloud-based data management system - the cloud service provider is responsible for providing and maintaining the DBMS.

Advantages of a DBMS

Using a DBMS to store and manage data comes with advantages, but also overhead. One of the biggest advantages of using a DBMS is that it lets end users and application programmers access and use the same data while managing data integrity. Data is better protected and maintained when it can be shared using a DBMS instead of creating new iterations of the same data stored in new files for every new application. The DBMS provides a central store of data that can be accessed by multiple users in a controlled manner.

Central storage and management of data within the DBMS provides:

- Data abstraction and independence
- Data security

- A locking mechanism for concurrent access
- An efficient handler to balance the needs of multiple applications using the same data
- The ability to swiftly recover from crashes and errors, including restartability and recoverability
- Robust data integrity capabilities
- Logging and auditing of activity
- Simple access using a standard application programming interface (API)
- Uniform administration procedures for data

Another advantage of a DBMS is that it can be used to impose a logical, structured organization on the data. A DBMS delivers economy of scale for processing large amounts of data because it is optimized for such operations. A DBMS can also provide many views of a single database schema. A view defines what data the user sees and how that user sees the data. The DBMS provides a level of abstraction between the conceptual schema that defines the logical structure of the database and the physical schema that describes the files, indexes and other physical mechanisms used by the database. When a DBMS is used, systems can be modified much more easily when business requirements change. New categories of data can be added to the database without disrupting the existing system and applications can be insulated from how data is structured and stored.

Of course, a DBMS must perform additional work to provide these advantages, thereby bringing with it the overhead. A DBMS will use more memory and CPU than a simple file storage system. And, of course, different types of DBMSes will require different types and levels of system resources.

RDBMS (relational database management system)

Margaret Rouse, guest contributor

A relational database management system (RDBMS) is a collection of programs and capabilities that enable IT teams and others to create, update, administer and otherwise interact with a relational database. Most commercial RDBMSes use Structured Query Language (SQL) to access the database, although SQL was invented after the initial development of the relational model and is not necessary for its use.

RDBMS vs. DBMS

In general, databases store sets of data that can be queried for use in other applications. A database management system supports the development, administration and use of database platforms.

An RDBMS is a type of DBMS with a row-based table structure that connects related data elements and includes functions that maintain the security, accuracy, integrity and consistency of the data.

Functions of relational database management systems

Elements of the relational database management system that overarch the basic relational database are so intrinsic to operations that it is hard to dissociate the two in practice.

The most basic RDBMS functions are related to create, read, update and delete operations, collectively known as CRUD. They form the foundation of a well-organized system that promotes consistent treatment of data.

The RDBMS typically provides data dictionaries and metadata collections useful in data handling. These programmatically support well-defined data structures and relationships. Data storage management is a common capability of the RDBMS, and this has come to be defined by data objects that range from binary large object (blob) strings to stored procedures. Data objects like this extend the scope of basic relational database operations and can be handled in a variety of ways in different RDBMSes.

The most common means of data access for the RDBMS is via SQL. Its main language components comprise data manipulation language (DML) and data definition language (DDL) statements. Extensions are available for development efforts that pair SQL use with common programming languages, such as COBOL (Common Business-Oriented Language), Java and .NET.

RDBMSes use complex algorithms that support multiple concurrent user access to the database, while maintaining data integrity. Security management, which enforces policy-based access, is yet another overlay service that the RDBMS provides for the basic database as it is used in enterprise settings.

RDBMSes support the work of database administrators (DBAs) who must manage and monitor database activity. Utilities help automate data loading and database backup. RDBMSes manage log files that track system performance based on selected operational parameters. This enables measurement of database usage, capacity and performance, particularly query performance. RDBMSes provide graphical interfaces that help DBAs visualize database activity.

While not limited solely to the RDBMS, ACID compliance is an attribute of relational technology that has proved important in enterprise computing. Standing for *atomicity*, *consistency*, *isolation* and *durability*, these capabilities have particularly suited RDBMSes for handling business transactions.

Relational database management systems are central to key applications, such as banking ledgers, travel reservation systems and online retailing. As RDBMSes have matured, they have achieved increasingly higher levels of query optimization, and they have become key parts of reporting, analytics and data warehousing applications for businesses as well. RDBMSes are intrinsic to operations of a variety of enterprise applications and are at the center of most master data management (MDM) systems.

RDBMS product history

Many vying relational database management systems arose as news spread in the early 1970s of the relational data model. This and related methods were originally theorized by IBM researcher E.F. Codd, who proposed a database schema, or logical organization, that was not directly associated with physical organization, as was common at the time.

Codd's work was based around a concept of data normalization, which saved file space on storage disk drives at a time when such machinery could be prohibitively expensive for businesses.

File systems and database management systems preceded what could be called the RDBMS era. Such systems ran primarily on mainframe computers. While RDBMSes also ran on mainframes -- IBM's DB2 being a pointed example -- much of their ascendance in the enterprise was in UNIX midrange computer deployments. The RDBMS was a linchpin in the distributed architecture of client/server computing, which connected pools of stand-alone personal computers to file and database servers.

Numerous RDBMSes arose along with the use of client/server computing. Among the competitors were Oracle, Ingres, Informix, Sybase, Unify, Progress and others. Over time, three RDBMSes came to dominate in commercial implementations. Oracle, IBM's DB2 and Microsoft's SQL Server, which was based on a design originally licensed from Sybase, found considerable favor throughout the client/server computing era, despite repeated challenges by competing technologies.

As the 20th century drew to an end, lower-cost, open source versions of RDBMSes began to find use, particularly in web applications. Such systems included MySQL and PostgreSQL.

Eventually, as distributed computing took greater hold and as cloud architecture became more prominently employed, RDBMSes met competition in the form of NoSQL systems. Such systems were often especially designed for massive distribution and high scalability in the cloud, sometimes forgoing SQL-style full consistency for so-called *eventual consistency* of data. But, even in the most diverse and complex cloud systems, the need for some guaranteed data consistency requires RDBMSes to appear in some way, shape or form. Moreover, versions of RDBMSes have been significantly restructured for cloud parallelization and replication.

Next Article

NoSQL (Not Only SQL database)

Margaret Rouse, WhatIs.com

NoSQL is an approach to database design that can accomodate a wide variety of data models, including key-value, document, columnar and graph formats. NoSQL, which stand for "not only SQL," is an alternative to traditional relational databases in which data is placed in tables and data schema is carefully designed before the database is built. NoSQL databases are especially useful for working with large sets of distributed data.

NoSQL vs. RDBMS

The NoSQL term can be applied to some databases that predated the relational database management system, but it more commonly refers to the databases

In this e-guide

built in the early 2000s for the purpose of large-scale database clustering in cloud and web applications. In these applications, requirements for performance and scalability outweighed the need for the immediate, rigid data consistency that the RDBMS provided to transactional enterprise applications.

Notably, the NoSQL systems were not required to follow an established relational schema. Large-scale web organizations such as Google and Amazon used NoSQL databases to focus on narrow operational goals and employ relational databases as adjuncts where high-grade data consistency is necessary.

Early NoSQL databases for web and cloud applications tended to focus on very specific characteristics of data management. The ability to process very large volumes of data and quickly distribute that data across computing clusters were desirable traits in web and cloud design. Developers who implemented cloud and web systems also looked to create flexible data schema -- or no schema at all -- to better enable fast changes to applications that were continually updated.

Key-value stores

Key-value stores, or key-value databases, implement a simple data model that pairs a unique key with an associated value. Because this model is simple, it can lead to the development of key-value databases, which are extremely performant and highly scalable for session management and caching in web applications. Implementations differ in the way they are oriented to work with

RAM, solid-state drives or disk drives. Examples include Aerospike, Berkeley DB, MemchacheDB, Redis and Riak.

Document databases

Document databases, also called document stores, store semi-structured data and descriptions of that data in document format. They allow developers to create and update programs without needing to reference master schema. Use of document databases has increased along with use of JavaScript and the JavaScript Object Notation (JSON), a data interchange format that has gained wide currency among web application developers, although XML and other data formats can be used as well. Document databases are used for content management and mobile application data handling. Couchbase Server, CouchDB, DocumentDB, MarkLogic and MongoDB are examples of document databases.

Wide-column stores

Wide-column stores organize data tables as columns instead of as rows. Wide-column stores can be found both in SQL and NoSQL databases. Wide-column stores can query large data volumes faster than conventional relational databases. A wide-column data store can be used for recommendation engines, catalogs, fraud detection and other types of data processing. Google BigTable, Cassandra and HBase are examples of wide-column stores.

Graph stores

Graph data stores organize data as nodes, which are like records in a relational database, and edges, which represent connections between nodes. Because the graph system stores the relationship between nodes, it can support richer representations of data relationships. Also, unlike relational models reliant on strict schemas, the graph data model can evolve over time and use. Graph databases are applied in systems that must map relationships, such as reservation systems or customer relationship management. Examples of graph databases include AllegroGraph, IBM Graph, Neo4j and Titan.

Evolution of NoSQL

Berkeley DB was an influential system in the early evolution of NoSQL database usage. Developed at the University of California, Berkeley, beginning in the 1990s, Berkeley DB was widely described as an embedded database that closely supported specific applications' storage needs. This open source software provided a simple key-value store. Berkeley DB was commercially released by Sleepycat Software in 1999. The company was later acquired by Oracle in 2006. Oracle has continued to support open source Berkeley DB.

Other NoSQL databases that have gained prominence include cloud-hosted NoSQL databases such as Amazon DynamoDB, Google BigTable, as well as Apache Cassandra and MongoDB.

In this e-guide

The basic NoSQL database classifications are only guides. Over time, vendors have mixed and matched elements from different NoSQL database family trees to achieve more generally useful systems. That evolution is seen, for example, in MarkLogic, which has added a graph store and other elements to its original document databases. Couchbase Server supports both key-value and document approaches. Cassandra has combined key-value elements with a wide-column store and a graph database. Sometimes NoSQL elements are mixed with SQL elements, creating a variety of databases that are referred to as multimodel databases.

Next Article

Database Administrator (DBA)

Margaret Rouse, guest contributor

DBA is also an abbreviation for doing business as - a term sometimes used in business and legal writing. dBA is an abbreviation for A-weighted decibels.

A database administrator (DBA) directs or performs all activities related to maintaining a successful database environment. Responsibilities include designing, implementing, and maintaining the database system; establishing policies and procedures pertaining to the management, security, maintenance, and use of the database management system; and training employees in database management and use. A DBA is expected to stay abreast of emerging technologies and new design approaches. Typically, a DBA has either a degree in Computer Science and some on-the-job training with a particular database product or more extensive experience with a range of database products. A DBA is usually expected to have experience with one or more of the major database management products, such as Structured Query Language, SAP, and Oracle-based database management software.

Next Article

Oracle Autonomous Database

Margaret Rouse, guest contributor

Oracle Autonomous Database is a cloud-based technology designed to automate many of the routine tasks required to manage Oracle databases, which Oracle says can free up database administrators (DBAs) to do higher-level and more strategic work. Introduced in 2017, the technology combines the company's Oracle Database 18c software with a set of automated administration services that use machine learning algorithms. Oracle 18c itself isn't autonomous -- the automation capabilities are provided by what Oracle has added on top of it to create Autonomous Database. The combination is offered as a cloud service called Oracle Autonomous Database Cloud, which Oracle describes as "self-driving, self-securing and self-repairing."

Features of Oracle Autonomous Database

Because of its machine learning functionality, Oracle Autonomous Database is able to assimilate the information that it needs to take care of itself. For example, the autonomous software provisions databases on its own, finding, allocating and configuring all of the necessary hardware and software for users.

Oracle Autonomous Database also doesn't require manual tuning to optimize performance; the technology tunes itself, including automatic creation of database indexes to help improve application performance. It also automatically applies database updates and security patches, backs up databases and encrypts data to protect information against unauthorized access.

The system patches itself on a regular quarterly schedule, although users can override this feature and reschedule the automatic patches if desired. Oracle Autonomous Database can also apply out-of-cycle security updates when necessary -- for example, if Oracle releases an emergency patch to address a zero-day exploit. Additionally, the technology uses Oracle's Database Vault tool to prevent Oracle DBAs from seeing user data and the company's data masking feature to identify and conceal sensitive data.

Oracle Autonomous Database can scale itself up as needed; it also monitors capacity limits and bottlenecks in key system components in an effort to avoid performance problems. Updates are applied in a rolling fashion across a clustered system's compute nodes so applications can continue to run during the process, and Autonomous Database automatically repairs itself in the event of a system failure, according to Oracle, which guarantees 99.995% uptime on the cloud service.

The technology gathers statistics as new data is uploaded, and regularly runs tests to ensure that all changes and upgrades are safe. It scans for issues across all layers of the technology stack using diagnostic tools such as ORAchk, EXAchk, OSWatcher and Procwatcher. If an error occurs, Autonomous Database collects relevant diagnostic data, analyzes logs to establish a timeline and works backward to solve the problem. For example, it can back out data errors made by users.

Benefits of using Autonomous Database

Oracle Autonomous Database is likely to change the way that Oracle DBAs function in organizations that adopt the technology. Because many of the more mundane tasks that DBAs now handle will be automated, Oracle says they'll be able to focus on things like data security, data lifecycle management, data architecture and data modeling. Additionally, DBAs could gain more time to work on new projects and help both development teams and end users to take better advantage of Oracle databases.

From an organizational standpoint, using Autonomous Database could reduce the need for human labor on Oracle data management teams, although Oracle says it expects the autonomous software to alter DBA jobs instead of eliminating them outright in most cases. The technology could also minimize data loss and human error in Oracle databases, while cutting back substantially on both planned and unplanned downtime.

Oracle Autonomous Database runs on the company's Exadata hardware platform and can be used either in the Oracle Cloud or via Cloud at Customer, a service that deploys systems based on Oracle's cloud technologies in on-premises data centers.

Oracle is developing multiple product offerings as part of the Oracle Autonomous Database Cloud service. The first was a data warehouse implementation that supports business intelligence and analytics uses; called Oracle Autonomous Data Warehouse Cloud, it was released to a group of early users in late 2017 and became generally available in March 2018. Other versions will follow for transaction processing workloads and NoSQL databases, according to Oracle.

Next Article

In this e-guide

Stories

OpenWorld 2017: Ellison declares autonomous database 'revolutionary'

Brian McKenna, business applications editor

In an ebullient conference keynote that shows Oracle still has Amazon Web Services (AWS) chiefly in its sights, chairman and CTO Larry Ellison declared the firm's 18c version of its database the "biggest thing we've done in decades".

He told attendees of Oracle OpenWorld 2017 in San Francisco that the database, which he says "runs and tunes itself" deserves the soubriquet "revolutionary", adding that he does not use the "Silicon Valley" term lightly.

In a speech in which he had, as he did at last year's show, much sport at Amazon's expense, he said Amazon Elastic Compute Cloud (EC2) was "anything but elastic", and that the rival supplier spent in the region of $60m with Oracle in 2016.

As he concluded the first of his two keynotes at the event, he also took a sideswipe at business applications supplier rival SAP. "They acquired three major online services years ago now, and they all run on the Oracle database, not on SAP Hana. SuccessFactors still runs on Oracle. SAP is one of the biggest Oracle users on the planet," he said.

Ellison linked the announcement of the new version of the database – dubbed Oracle Autonomous Database Cloud, and said to be the world's first autonomous database – with the salience of cyber security use cases.

With the world's first 100% self-driving autonomous database, and new automated cyber defense applications that detect and remediate attacks in real time, Ellison added: "Data theft is the biggest cyber security threat. The safest place for data is autonomous Oracle database."

Ellison said this is because patching by humans – such as what failed to occur in the recent Equifax data loss case – is a structural vulnerability. But, he added, database automation will free up database professionals to do more important things, and will not eliminate database administrators altogether.

He also said the machine learning underlying the new database to be "as revolutionary as the internet itself", adding that "distinguishing between normal and abnormal patterns in data is the critical application of machine learning".

The main pitch to attendees, buttressed by an extended comparison of how Oracle runs against Amazon, was that his company's technology runs "five to eight times faster than Amazon", and that "labour costs are almost gone".

"We will write in contracts that your [database] bill will be half in the Oracle cloud compared with [data warehouse service] Amazon RedShift," said Ellison.

In a statement, Oracle said: "With total automation based on machine learning, Oracle Autonomous Database Cloud eliminates the human labour required to manage a database by enabling a database to automatically upgrade, patch and tune itself while running.

"With no more scope for human error or requirements for human performance testing, Oracle is able to minimise costly planned and unplanned downtime to less than 30 minutes a year and guarantee that organisations can cut their costs in half compared to Amazon."

Next Article

Oracle sees Autonomous Database changing DBA roles for the good

Lindsay Moore, associate managing editor

In this e-guide

As Oracle launches its Autonomous Database Cloud service, database administrators may be left to wonder if the new offering will affect their jobs. The specter of an automated system eliminating the need for people is a common fear, but it needn't be in this case -- in Oracle's view, at least.

In a recent webinar on Oracle Autonomous Database Cloud, Maria Colgan, a master product manager for databases at the company, assured database administrators (DBAs) that their jobs won't disappear as a result of the increased automation offered by the cloud service. In fact, "there's actually a shortage of really skilled database experts in our industry today," she said.

Colgan told listeners that the autonomous database administration functionality will indeed affect the role of DBAs, but in ways that she characterized as good for both them and their organizations.

According to Colgan, a typical DBA's duties can be broken down into two categories: tactical operations and business tasks. The former involves the more mundane aspects of a DBA's workday, such as performance tuning, database backups, database optimization, and configuration and provisioning of new systems.

More time for higher level work

If the tactical tasks are reduced -- or even eliminated -- DBAs will have more time to spend on higher level tasks, like data security, data lifecycle management, data architecture and data modeling, Colgan said. She added that they could also make more room in their schedules to work on new projects and to help both the development teams and business users.

For example, with the newfound free time created by an Oracle Autonomous Database Cloud deployment, DBAs could play a bigger role in application development by showing developers the features and functions in the database and how to best exploit them, and then help to optimize data access and enhance the user experience, Colgan said.

As the job requirements evolve, Colgan thinks the DBA will become more of a data engineer who must have a strong understanding of the data available in databases and how it is organized and stored in order to help developers and business users get the most out of the information. Ultimately, a good DBA will need to be a jack-of-all-trades, adept at an increasing variety of tasks, she said.

Learning about machine learning

Working with Oracle Autonomous Database Cloud will also give DBAs an opportunity to learn new skills, such as machine learning, according to Colgan. Oracle touts the cloud service's use of machine learning algorithms to automate performance tuning and other database administration tasks. DBAs can also take advantage of built-in algorithms to, for example, track and analyze user behavior in database systems, Colgan said in the webinar, which was hosted by two Oracle user groups: the Independent Oracle Users Group and the Quest International Users Group.

Oracle obviously has a vested interest in painting the autonomous database technology's potential impact on DBAs in a positive light. But some IT analysts had a similar take on how the duties of DBAs are evolving in interviews at Oracle OpenWorld 2017 in San Francisco, which is where the company announced Oracle Autonomous Database Cloud.

For example, at the conference, Forrester Research analyst Noel Yuhanna said that he sees DBA jobs shifting "toward more data-driven initiatives with more emphasis on security and governance" and an increased focus "on business value as opposed to technology."

Next Article

Oracle’s autonomous database could leave DBAs unemployed

Stuart Kennedy, guest contributor

Oracle CEO Mark Hurd threw the long-term future of hundreds of thousands of database administrators (DBAs) into question while talking up the benefits of the company’s new autonomous database.

The autonomous self-patching, self-healing database, the first version of which is 18c, is a part of a long-term play to help draw the company’s customers into Oracle’s piece of the cloud – which is increasingly packing itself with cloud-based applications and services.

Hurd said it could take almost a year to get on-premise databases patched, whereas patching was instant with the autonomous version. “If everyone had the autonomous database, that would change to instantaneous,” he said.

So where does that leave Oracle DBAs around the world? Possibly in the unemployment queue, at least according to Hurd.

“There are hundreds of thousands of DBAs managing Oracle databases. If all of that moved to the autonomous database, the number would change to zero,” Hurd said at an Oracle media event in Redwood Shores, California.

In this e-guide

That could be appealing for companies with large rosters of DBAs from a cost-cutting perspective, but that day is most likely many years away.

It is early days for the 18c version, which became available in March 2018, and most Oracle customers are still kicking its tyres.

Three customers on a panel said they were evaluating 18c with a view of using it in the future. They were Michael Sherwood, IT director for the city of Las Vegas; Glenn Coles, Yamaha US CIO; and Lynden Tennison, CIO of Union Pacific Corporation. Hertz and Accenture are also likely early 18c users.

Meanwhile, Pat Sullivan, Accenture's North America Oracle business group lead, said at the event that his firm has 20,000 DBAs and their future looked reasonably rosy with many set to become more specialised database experts – if the basic database maintenance role went away with the autonomous version.

Hurd said the performance boost from 18c was on a level with the company's high-end Exadata Database Machine, used by around 5% of Oracle's on-premise customers.

"They [Exadata customers] get 20 times better performance than our traditional on-premise customers. Imagine everybody getting Exadata-plus performance. Extreme performance, totally patched, totally optimised," he said.

Hurd was also very bullish on Oracle's applications business, which includes revenues from on-premise support, on-premise licensing and software as a service (SaaS).

"I made a prediction in the middle of last year that [the applications business] would grow double digits…and that will happen for us during the year," he said.

The company is also throwing in top-level platinum level support at no cost for anyone using Oracle's Fusion SaaS applications. The support package includes 24/7 rapid response technical support, proactive technical monitoring, implementation guidance and improved on-demand education resources.

On the autonomous platform-as-a-service front, which Oracle is increasingly targeting as a future cash cow, the company announced the availability of three new services that have baked in artificial intelligence (AI) and machine learning algorithms. These are the Oracle Autonomous Analytics Cloud, Oracle Autonomous Integration Cloud, and Oracle Autonomous Visual Builder Cloud.

Next Article

DBA skills must evolve, morph to cope with big data technologies

Jack Vaughan, senior news writer

New technologies are shaking up the status quo in data processing. Despite the rush of schema-less NoSQL and Hadoop platforms and associated tools, the change may be moderate when it comes to database administration. But some new skills will likely come to the fore for database administrators (DBAs) as systems based on those technologies get deployed in companies. And the days when the DBA was the sole "keeper of the database schema" seem to be waning.

"There clearly is a shift going on, but it's not as extreme from the DBA point of view," said Joe Caserta, president of Caserta Concepts LLC, a New York-based consulting and training services company that focuses on data warehousing and big data deployments. In traditional enterprise data shops, teams of DBAs are still busy keeping well-entrenched relational databases and data warehouses built around familiar SQL running smoothly.

Developers have gained added influence on data design, however, and Caserta said the changes will push some DBAs to obtain added skills.

Things change, things stay the same

For example, the fact that developers can initiate Hadoop and NoSQL projects with little or no upfront schema represents a change in enterprise development methods. Even so, companies are still likely to create reference data models after the fact. And while such modeling may be done by enterprise architects or data architects, some DBAs also play a modeling role -- so they'll need to learn how NoSQL systems work, Caserta said.

"They'll be using different tools and different modeling strategies," he said. "We still need someone to come up with the models. We also need someone who knows how to administer these new databases."

Handling the latter task likely will require new training: A DBA who is certified on the Oracle database is not going to be able to automatically set up and design, say, a Cassandra NoSQL database. "The methods the typical DBA needs to know have to be relearned," Caserta said, adding that the ability to manage Hadoop clusters will also become a crucial skill for DBAs. In general, "they'll need to learn how to store data when they don't have a schema."

What price agility?

Craig Mullins, president and principal consultant of Mullins Consulting Inc. in Sugarland, Texas, said the NoSQL movement has precedents that will ease the learning process for some DBAs. "It's not as new as some people think," he said, noting, for example, the similarity between the mainframe-oriented VSAM file technology and newfangled key-value data stores.

There are some fundamental differences, though. The agility offered by NoSQL software comes at a price as data integrity becomes far more challenging to achieve. But for now, full integrity is taking a back seat to data flexibility in many companies' Web-based applications. For the DBA, the big challenge is to adjust as design and development styles shift.

"There are some DBAs who are more adaptable than others. We've already seen that play out in the last 20 years," said Mullins, who has more than 30 years of data management experience and is the author of *Database Administration: The Complete Guide to DBA Practices and Procedures*, first published in 2002 and updated in a second edition in 2012.

Mullins pointed out that, in many organizations, DBAs are already expected to work with several relational database systems. DBAs with those kinds of skills "should investigate the NoSQL options so they can be in the forefront of guiding the organization wisely when and if they need the NoSQL offerings," he said.

Looking forward, Mullins added, data schema definition may be minimized, but "the availability of systems and the understanding of how data is spread across nodes will become even more important."

Salad days for DBAs?

In fact, Sue Geuens, president of DAMA International, an association of data management professionals, thinks new data architectures provide opportunities for DBAs to expand their usefulness in organizations. "DBAs have been seen as the techno-geeks who sit in corners and nurse your database servers through all their ups and downs. And I think that DBAs are getting somewhat tired of that label," said Geuens, who is based in Johannesburg, South Africa, and works as head of data services at EPI-USE Systems Ltd., an SAP-focused software and services provider.

Geuens said we likely will see different types of DBAs in the future, with some being content to stay in a traditional technical and administrative role, while others will look at learning the new technologies and tools for managing big data. "We will see a slew of new job titles for DBAs, and we'll also see specialization even deeper than the ones we currently have around the specific relational database," she said.

As Geuens emphasized, adding new skills is nothing new to the DBA job, which already comprises a full menu of capabilities. Pertinent skills include modeling, performance management and basic administration, with specializations for each database brand adding to the job's complexity.

Such expertise can mean significant pay. TechTarget's 2014 IT Salary and Careers Survey, for example, showed database administrators among recent gainers with average total compensation of $115,630. That's up 22% year-over-year.

High salary or not, working with new technologies such as NoSQL databases does require you adjust your way of thinking, said Greg Novikov, a database specialist at New York-based insurer MetLife who spoke at the MongoDB Days 2014 event held in Boston in October. "But that's the reason we're getting paid the big bucks," he added -- with a wink to the audience.

Next Article

How to manage data analytics programmes and teams

Cath Everett, guest contributor

Although data analytics technology has been around in one shape or form for years, it seems that organisations are still having problems in obtaining the full value from their often rather pricey initiatives.

The *The age of analytics: competing in a data-driven world* report, released by the McKinsey Global Institute in December 2016, revealed that while a few – mainly digital native – organisations were using their data and analytics technology effectively, most were a long way from doing so.

The most successful initiatives were found among retailers and firms offering location-based services. But McKinsey indicated that companies in manufacturing, the public sector and healthcare were obtaining less than 30% of their projects' potential value.

Beyond greenfield sites that have been built with data in mind from the ground up, Jason Foster, founder and director of data and analytics consultancy Cynozure, believes there are three main types of organisations that are struggling to getting it right.

These include companies in which senior leaders have said they are keen to benefit from data analytics but where the business is struggling to deliver.

There are also firms that have been investing and working in this area for a long time but, because they are still not gaining value, are becoming disheartened. Finally, there are the organisations that are a mixed bag in terms of maturity.

So although their e-commerce and digital marketing departments may gain a lot of value from their data and analytics activities, more "traditional" areas such as logistics do not.

The biggest obstacles to success are not so much technical as organisational, according to a study entitled *Plotting the data journey in the boardroom: the state of data analytics 2017*, conducted by software supplier MHR Analytics.

It found that while just more than three-quarters of the 300 UK-based C-level executives questioned planned to undertake a data analytics or big data project over the next 12 months, the most significant barrier to doing so was finding and training appropriate staff (42%).

Another key issue was developing a coherent business intelligence (BI) strategy for the entire company (29%), followed by how best to manage BI initiatives that are being undertaken by individual department heads and using BI to generate actionable business insights (28% respectively).

Overcoming business challenges

While Cynozure's Foster agrees that finding people with good quality skills and experience is not easy and generally requires investment in training, he also points out that "at the component level, the skills are out there, but the issue is that people try to find a few individuals to solve all their problems".

"Data analytics is a team sport and you're unlikely to find all the necessary skills in one person," says Foster. "You need to have a good mix of people with overlapping skills that complement each other."

Such a team will likely be composed both of internal staff who have been given relevant training and people who have been hired in from outside, perhaps through graduate programmes. Key roles, on the other hand, will include software engineers who can build data pipeline frameworks to correlate data from a range of sources.

Also vital are data engineers able to model data in such a way as to make it accessible to business users, and data analysts and scientists. They analyse the data, uncover insights, present them to business users in a pertinent way and then collaborate to turn those insights into action.

Last but not least is the chief data or digital officer (CDO), whose job it is to understand the value of the organisation's data and what opportunities it offers, while also orchestrating how it is handled and ensuring that governance is sound.

As for challenges around developing a coherent enterprise-wide BI strategy that does not end up fragmenting into individual departmental initiatives, Matt Jones, lead analytics strategist at data analytics consultancy Tessella, advises ensuring that it is jointly owned by IT in the shape of the CDO and the business.

"It's important to have an overall, joined-up strategy and to understand where you want analytics to take you," he says.

"If you don't, the danger is that you end up with a huge technical programme that takes years to complete, when really it's about having a vision and delivering wins quickly and iteratively."

The issue is that a lot of organisations simply jump in and buy technology without thinking about the business problems they want to solve or the new opportunities they would like to exploit.

"Identify your problem, work out what data you need and then think about the skills and technology you require," says Jones. "All too often, people gather lots of data in an analytics platform and then look for a problem to solve, but that'll only threaten your return on investment."

Cynozure's Foster agrees. "While building a central data warehouse should be a goal, you need to pick a suitable use case and start with that," he says. "You don't need every department and every data set from the outset – start small and grow."

Aligning around the customer

Another challenge people often struggle with, Foster points out, is developing the right "cultural mindset". This is important, he believes, as "this makes things stick – it's about how you turn your vision into an executable plan that will resonate and get buy-in."

Enabling this kind of "cultural alignment" tends to be particularly difficult in organisations that operate in silos, each with their own profit and loss accounts and IT systems. The secret, says Foster, is to focus business owners on something they all care about, which is generally the customer.

"Customers tend to be the thing that can pull people together around a data analytics strategy – it's a good way to win hearts and minds and get them to start thinking in a horizontal rather than vertical way," he says.

In this e-guide

Another good approach for introducing change is to ensure there is sponsorship at the top and buy-in at the bottom. “Demonstrate how using data can move the needle,” Foster says.

“This may come from some very small, quick work to show ‘if you do a and b, you get c’. Then you do some PR and marketing and demonstrate value.”

Having data analytics champions in different departments to spread the word is useful in this context, as is holding events such as “show and tells”, data dives or hackathons.

“You’ve got to attack these things on multiple fronts. There’s no silver bullet and no just ‘put in that new technology and bring in a new data scientist and everything will be fine’. Where you are now as an organisation will change how you execute, so it’s not easy,” says Foster.

Case study: JLL

“What makes a data analytics project successful is having defined outcomes or using the data to answer specific business questions rather than just pulling data into a system and getting lost in it,” says Eddie Wagoner, chief information officer at commercial property and investment management services provider JLL.

The firm, which is based in Chicago but has offices worldwide, including the UK, not only makes wide use of data analytics technology internally but also employs it on behalf of its customers.

"The same thing happens in a data lake as it does in a water lake – you can get in and have fun, or you can drown and die, so you've got to know what you're doing. The right people need to be involved and you have to have a specific destination in mind," says Wagoner.

This destination could be anything from becoming more productive and saving money to creating a revenue stream.

To optimise its activities here, JLL has set up a global BI and data governance team, which is based both in the US and around the world and is looked after by its own dedicated leader. Account managers also regularly join the team when working on specific challenges faced by their customers, which include Fortune 500 companies.

"It's a co-ordinated effort to ensure governance is in place and to avoid duplication. But also the best ideas often come from the business as they know their customers so well," says Wagoner.

Nonetheless, he acknowledges that finding, training and retaining skilled and experienced data analytics staff in the property sector is a key issue.

"Demand for people with expertise in data analysis and BI is far outstripping supply," he says. "The real estate industry, in particular, has not historically needed people such as data scientists and so we've been going to other industries to attract them to ours."

The organisation started on its data analytics journey in 2012 when its CEO Colin Dyer publicly announced that he wanted the company to become an industry leader in BI. "It was his idea," Wagoner says. "He saw the opportunity and laid it out as a strategic goal for the company."

Getting the CEO on board, he believes, is key. "That's when it'll be most successful as it becomes a strategic priority. If not, you're not going to get the smartest people interested and focusing on it," Wagoner says.

Case study: Student.com

Ensuring that data teams work with the business and are actively involved in supporting its day-to-day activities is crucial if they are to deliver real value, believes Ian Broadhead, head of data and analytics at Student.com.

Student.com, which was set up in 2011, is an online marketplace that helps international under- and post-graduates find residential accommodation in locations around the world. The company employs data and data analytics in most areas of its business to spot trends and monitor performance at all levels.

As a result, it holds enterprise-wide "huddles" to share company updates with its 100 employees in London and Shanghai every Friday.

But it also holds voluntary "data huddles" of up to half an hour each Tuesday, in which the data team presents its weekly findings on what is doing well or not so well and what has changed. Feedback is encouraged and senior leaders are available to answer any questions.

"It's a great way for us to share information in a concise way and to mention if we see any issues that need flagging," says Broadhead. "For example, there was a sudden drop here or a massive increase there, and so maybe we need to look at this or do that."

But he acknowledges that presenting data in a way that is simple to understand for everyone – from well-established technical staff to a receptionist that has just recently joined the company – is challenging.

"Techies often just produce charts and try to explain them in detail, but they can end up losing their audience. What you need is for your team to provide context and translate the data in a way that the business can understand, using business language. So it's about having good presentation skills too," Broadhead says.

To ensure his data team – which comprises of two data analysts, one data engineer and himself – is suitably integrated into the business, each has responsibility for different business areas and locations worldwide and takes part in regular meetings with other stakeholders.

As to why all too many enterprises fail to gain value from their data initiatives, Broadhead believes there are a number of reasons.

On the one hand, data teams have a habit of operating in silos and so are unaware of what the business needs and why. On the other, they often become so bogged down in introducing large technology deployments and creating dashboards that "they forget their objectives and what they're trying to do".

"My best advice is to make sure you don't become locked in an ivory tower. It's hard sometimes to communicate with the business, but you're part of it, not separate superstars, so make the most of it. It makes life much more interesting," Broadhead concludes.

Next Article

Microsoft boosts SQL Server machine learning services

Jack Vaughan, guest contributor

SEATTLE -- While SQL Server 2017 continues to get attention for opening up to Linux, many of Microsoft's database advances revolve around various ways the company is opening up analytics on its flagship database. Case in point: SQL Server machine learning services.

In this e-guide

Open source data frameworks and development languages increasingly have become a path to next-level data analytics and machine learning, and SQL Server support is central to this strategy.

The clues are various. Even before the 2017 release of the database, Microsoft brought Apache Spark and the R language into the mix. Earlier this year, the Python language joined R as part of a newly minted Azure Machine Learning developer kit.

The story took a new turn at PASS Summit 2017 last week, as Microsoft featured the capability for Azure Machine Learning users to bring their analytics models into SQL Server 2017 for native T-SQL runtime scoring. An essential element in machine learning, scoring is a way to measure the likely success of machine-generated predictions.

Native T-SQL scoring can process large amounts of data at an average of under 20 milliseconds per row, according to Rohan Kumar, general manager of Microsoft's database systems group, who spoke at PASS Summit. Native T-SQL scoring takes the form of a stored procedure for prediction that can be used without calling Microsoft's R runtime, as was the case with SQL Server 2016.

This capability is important because models built and trained to, for example, suggest new products to likely buyers can produce results while the buyers are actually web browsing. As SQL Server machine learning services head in this direction, their use could grow.

Machine learning models

Supporting such scoring in the Microsoft database could make machine learning analytics more a part of operations and less an experimental effort, according to Ginger Grant, advanced analytics consultant for SolidQ and a presenter at the event.

"Traditionally, what has happened is that you've had a data science group that sort of sat in the corner creating machine learning models. They then threw that 'over the wall' to developers who had to code it in another language," Grant said in an interview.

"Native T-SQL scoring allows people to modularize their work and environment, so things can be operationally implemented relatively quickly," she said.

Microsoft's new SQL Server machine learning services will help with real-time prediction, said Victoria Holt, who also took part in PASS Summit. She is an independent data analytics and platforms architect, as well as a trainer at SQL Relay.

"It is great to be able to leverage machine learning computation in-database," she said.

This year's inclusion of Python in the Microsoft Machine Learning workbench is also a step forward, Holt said. But it will take time for such new technologies to spread.

Holt noted that the "addition of Python extends the use of deep learning frameworks in the product. The retrained cognitive models will speed up consumption. But there is significant user training and upgrading that will need to happen before these models are adopted."

Beyond T-SQL stored procedures

Microsoft analytics advances discussed at PASS Summit were not limited to T-SQL. The company previewed scale-out features for Azure Analysis Services to improve response time for large query workloads on the cloud.

The company also moved to simplify data preparation for analytics in the cloud by releasing a public preview of Azure Data Factory that includes the ability to run SQL Server Integration Services in ADF.

Growing Microsoft SQL Server 2017 support for Python and R is significant, according to Jen Stirrup, founder of the U.K.-based Data Relish consultancy and PASS Summit board member.

Python is something of a portal to a crop of machine learning services entering the open source sphere almost daily. In Stirrup's view, deeper support for advanced analytics is the next step for big data, and Microsoft is tuned to that notion.

"The company understands that customers really want to do something with the data," she said.

"The data is such a key thing. It underpins your applications. Today, that means you have to reach out to software and languages that are not necessarily part of Microsoft's .NET," Stirrup continued. "Microsoft's moves are all about being more welcoming to open source communities."

Getting more CW+ exclusive content

As a CW+ member, you have access to TechTarget’s entire portfolio of 140+ websites. CW+ access directs you to previously unavailable “platinum members-only resources” that are guaranteed to save you the time and effort of having to track such premium content down on your own, ultimately helping you to solve your toughest IT challenges more effectively—and faster—than ever before.

Take full advantage of your membership by visiting www.computerweekly.com/eproducts

Images; stock.adobe.com

© 2018 TechTarget. No part of this publication may be transmitted or reproduced in any form or by any means without written permission from the publisher.