

Nonrelational databases

A **nonrelational database**, which is often referred to as a **NoSQL database**, is a database that does not use a table structure to store data.

NoSQL databases are purposely built for specific data models. They have flexible schemas, or structures, for building modern applications.

Primary key		Attributes				
Partition key: customerid	Sort key: sk	createDate	updateDate	email	fullName	userPreferences
123	CUSTK123	20200325T09:1641.123	20200325T09:1641.123	shirley@exampl.net	Shirley Rodriguez	{}
	https://aws.amazon.com	20200325T09:1642.123	20200325T09:1642.123	Cloud	AWS	Amazon Web Services
	https://console.aws.amazon.com	20200325T09:1643.123	20200325T09:1643.123	Cloud	AWS Console	Web console
		20200325T09:1644.123	20200325T09:1644.123	Cloud	AWS Docs	Documentation
321	CUSTK321	20200325T09:1644.123	20200325T09:1644.123	zhang@example.net	Zhang Wei	{}
	https://aws.amazon.com	20200325T09:1645.123	20200325T09:1645.123	Tools	AWS	Amazon Web Services
	https://docs.aws.amazon.com	20200325T09:1646.123	20200325T09:1646.123	Docs	AWS Docs	Documentation
		20200325T09:1647.123	20200325T09:1647.123	Cloud	AWS	Amazon Web Services

11

aws re/start

NoSQL databases use various data models for accessing and managing data. These types of databases are optimized specifically for applications that require large data volumes, low latency, and flexible data models. These requirements are achieved by relaxing some of the restrictions around data consistency that are used by other databases.

Relational (SQL) versus nonrelational (NoSQL) databases

Relational (SQL)

Structured query language

Rigid schema

Run transactional or analytical queries

Used primarily in relational databases

Easily update, delete, or add to existing databases

Uses vertical scaling, which means changing the resources (CPU, RAM), inside the computer or virtual machine (VM) to fit the needs of the database

Nonrelational (NoSQL)

Not only SQL

Not a rigid schema

Uses different types of data models, such as graph, document, key-value

Alternative to traditional relational DB

Used for large data stores in cloud and web applications

Uses horizontal scaling, which means adding more computers, servers, and so on

This slide shows a comparison between relational (SQL) and nonrelational (NoSQL) databases.

Database types and use cases

Database types	Use cases
Relational	Traditional applications, enterprise resource planning (ERP), customer relationship management (CRM), ecommerce
Nonrelational (key-value)	High-traffic web applications, ecommerce systems, gaming applications
Nonrelational (in-memory)	Caching, session management, gaming leaderboards, geospatial applications
Nonrelational (document)	High-scale industrial applications for equipment maintenance, fleet management, and route optimization

13



This table describes some database types and use cases.



Database management system (DBMS)

The diagram consists of a large light green rectangle with a dark green gradient. On the left side, there is a small dark blue square. To the right of this square, the text 'Database management system (DBMS)' is written in a dark green font. A large, diagonal watermark reading 'DO NOT COPY' and 'bufetekaye.22@gmail.com' is overlaid across the entire diagram area.

How do you manage stored data?

Database management system (DBMS) is software that provides database functionality.

Database management system (DBMS)

Database management system (DBMS)

Database management system (DBMS) is software that provides database functionality.

Two variations are:

Single user DBMS applications,
such as Microsoft Access

Multiple user DBMS applications,
such as Oracle Database, Microsoft
SQL Server, MySQL, and IBM Db2

15



How do you manage stored data?

You manage stored data with a database management system (DBMS).

A database management system (DBMS) is software or **database-as-a-service (DBaaS)**. It is used mainly for creating databases; inserting data into a database; and storing, retrieving, updating, or deleting data in the database.

A DBMS provides a tool to create databases. A company can design and model a database in the DBMS and run any command to manage the data. It gives users an interface between the data in a database and the software itself.

Database management systems (DBMS), continued

Locations



On-premises (data center) – Data is stored on computers and networks that the organization owns. These computers and networks are located in the organization's own data center.



In the cloud (virtualized data center) – Data is located outside the organization, such as in remote data centers that are owned by cloud providers such as Amazon Web Services (AWS).

16



With traditional database management, a company can use a DBMS to create and establish databases in either an on-premises data center or a virtual environment. With a traditional DBMS, cost and infrastructure can be an issue.

Because DBaaS is in the cloud, it is a virtualized data center. It resolves many issues of a traditional DBMS.

Database-as-a-service (DBaaS)



A few key points about cloud-based databases

- **Database servers are hosted by third-party providers**
 - Hosted in third-party data centers and accessed over the internet (the *cloud*) instead of being hosted on local networks
- **Reduction in cost**
 - Reduces cost of installing and maintaining the servers
- **Fully managed**
 - For example, with some AWS databases, you do not need to manage database management tasks such as server provisioning, patching, setup, configuration, backups, or recovery
- **Faster**
 - Enables you to use services (like AWS), which offer large amounts of storage and processing power in their data centers

17



Database servers are hosted in third-party data centers. They are accessed over the internet (the *cloud*) instead of being hosted on local networks.

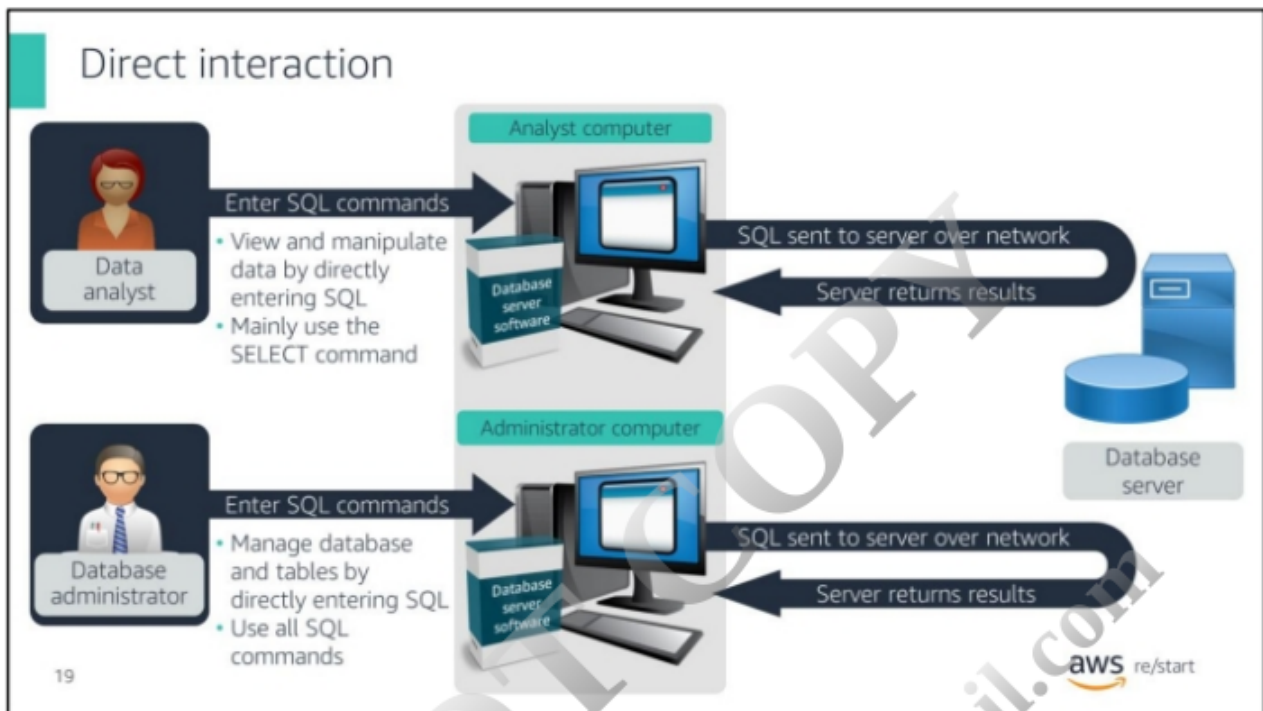
Reduction in cost: DBaaS reduces the cost of installing and maintaining the servers yourself.

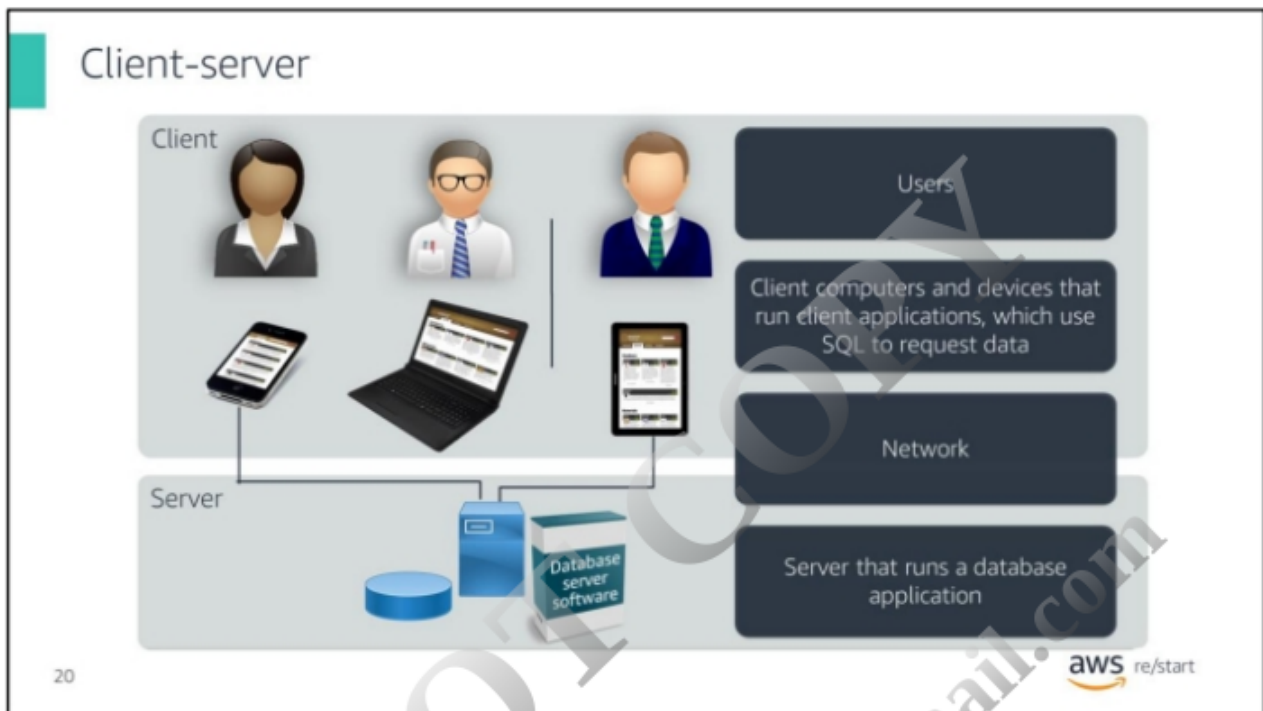
Faster:

- A DBaaS can make database access faster and more accessible to clients that are located anywhere in the world by placing the databases closer to users.
- It enables you to use services (like AWS), which offer large amounts of storage and processing power in their data centers.
- It enables you to process large datasets that might be impractical to process by using local servers.
- A DBaaS can be installed in a cloud-based virtual machine (VM) or as a service from a cloud provider.

Examples of data interaction (DI)

The next few slides offer a review of different ways to interact with data from databases.





In the client-server model, applications are created to provide an instrument for interaction between the users and the interface of a DBMS.