Engineer

SQL Server

Microsoft's relational database

What it is

SQL Server, which many teams write as MSSQL or MS SQL, is a relational database management system: a server program that keeps data in tables, enforces the rules between them and answers the applications that talk to it in Transact-SQL, or T-SQL, Microsoft's dialect of SQL. A client connects to an instance, which is one running copy of the engine, and does its work inside one of the databases that instance hosts. On disk, each of those databases is at least two files, one for data and one for its transaction log.

The core component is the Database Engine, the service that stores, processes and secures data. It installs on Windows or Linux, in a Linux container or on a virtual machine. Linux support began with SQL Server 2017 (14.x), using the same Database Engine as on Windows, and Red Hat Enterprise Linux and Ubuntu are supported for every release since. The same engine also sits underneath several of Microsoft's hosted services, among them Azure SQL Database, Azure SQL Managed Instance and SQL database in Microsoft Fabric.

Releases are named by year and numbered internally: 2017 is 14.x, 2019 is 15.x, 2022 is 16.x, and SQL Server 2025, whose support began on 18 November 2025, is 17.x. Around the engine sit further services. Integration Services (SSIS) builds packages for extract, transform and load work aimed at data warehousing, and SQL Server Agent runs jobs on a schedule. Starting with SQL Server 2025, on-premises reporting is consolidated under Power BI Report Server, and Data Quality Services and Master Data Services leave the product; 2022 and earlier releases keep them.

The editions draw the commercial lines. For SQL Server 2022 they are Enterprise, Standard, Web, Express (free), Developer and Evaluation; the last two carry the full Enterprise feature set, Developer licensed for development and test systems

only and Evaluation for 180 days. SQL Server 2025 redraws the list: Web edition ends with 2022, Developer splits into Enterprise Developer and Standard Developer, and Express grows.

Use the concept in a decision

Put “What is SQL Server?” to work in a real decision. We can help relate the concept to the system you are building or buying. A structure that matches how the business actually works, so the questions you need answered later are still possible to ask.

Explore our services Get in touch →

Test the assumptions behind the term

Use the explanation and linked work to frame questions for your project. We can examine the assumptions with you before a tool is chosen. Designed and ran 6 ETL/ELT pipelines — 6 ETL pipelines across three database engines

Designed and ran 6 ETL/ELT pipelines Get in touch →

Why it matters

A version and an edition together decide how large a server may get, which features it has and how long Microsoft keeps fixing it.

Size and features come from the edition. Express in SQL Server 2022 uses the lesser of one socket or four cores, at most 1,410 MB of memory for its buffer pool, and holds at most 10 GB in each database. In SQL Server 2025 the Express cap rises to 50 GB, with the same compute and memory limits. Standard in 2022 uses the lesser of four sockets or 24 cores and 128 GB of buffer pool; in 2025 those become 32 cores and 256 GB. For Enterprise, the edition table lists the operating system maximum for compute and buffer-pool memory under core-based licensing. The legacy Enterprise Server+CAL model, unavailable for new agreements, limits each instance to 20 cores. SQL Server Agent comes with Enterprise, Standard and, in 2022, Web; Express leaves it out, so a free instance that needs nightly backups needs a scheduler from outside the product.

The fix schedule comes from the version. Under Microsoft’s Fixed Lifecycle Policy each release gets at least ten years of support: five of mainstream support, with functional, performance, scalability and security updates, and five of extended support, with security updates alone. For SQL Server 2019, mainstream support ran from November 2019 to the end of February 2025, and extended support continues to January 2030. Microsoft’s update list shows CU32, from February 2025, as the latest cumulative update for 2019, with security releases published on top of it since. SQL Server 2022 stays in mainstream support until January 2028 and in extended support until January 2033; SQL Server 2025 moves to extended support in January 2031 and leaves it in January 2036.

The product also reaches people who never run it in production, because data travels in its formats. A `.bak` file is a SQL Server backup, and reading one means restoring it into an instance, as one of the projects below did.

What it looks like in practice

For a mid-sized tech company, a DevOps role ran MS SQL Server 2019 alongside PostgreSQL 14 on ten database servers built on Ubuntu 20.04 LTS VPS instances. Backups used `pg_dump` for PostgreSQL and SQL Server Agent jobs for MS SQL, with retention policies and offsite storage. Prometheus and Grafana tracked the servers, and the result was 99.9% uptime across all ten.

Elsewhere, SQL Server was the format the data arrived in. Data for internal dashboards and client-facing Python APIs came from a remote warehouse behind an IPsec VPN, as compressed archives holding Parquet, CSV and `.bak` files. The pipeline restored each backup into a local MSSQL Server instance with `RESTORE DATABASE` and validated schema integrity. Staging loads used `bcp`, `psql` and SSIS depending on the source format, and T-SQL and PL/pgSQL procedures cleaned and enriched the rows before they went on to BigQuery. With the `.bak` restores scripted along with the rest, the six pipelines did in under 20 minutes what had taken 3+ hours by hand, and the data synced daily instead of weekly.

A geographical map application, by contrast, needed to leave. It sat on a legacy MSSQL database, and the team moved it to PostgreSQL with the PostGIS extension for native geospatial indexing. More than 50 million spatial records went through ETL into EPSG:4326, and GiST and SP-GiST indexes went onto the geometry and geography columns. Off SQL Server, the average spatial query took under 250 milliseconds instead of 2.5 seconds, and backend CPU load fell by 65%.

Where it goes wrong

Each of these starts at a boundary: between versions, between machines, or between a backup and its first use.

A backup travels forward in version, and only forward. A backup restores onto the version of SQL Server that wrote it or onto a newer one. A database restored from an earlier version is upgraded automatically on the way in, and that upgrade

is one-way. A `.bak` file written by a 2022 server therefore needs a 2022 or 2025 instance, and a 2019 instance turns it away. Check which version wrote a file before building the instance that reads it.

The restore asks for folders the new server lacks. A backup records where its data and log files lived on the source server, and each SQL Server version uses its own default path. On a different machine, list the logical file names first and then place each one with `MOVE`:

```
RESTORE FILELISTONLY
FROM DISK = '/srv/incoming/sales.bak';

RESTORE DATABASE Sales
FROM DISK = '/srv/incoming/sales.bak'
WITH MOVE 'Sales_Data' TO '/srv/mssql/data/Sales.mdf',
     MOVE 'Sales_Log' TO '/srv/mssql/data/Sales_log.ldf';
```

Backups pile up and nobody reads one back. Microsoft's own page on the subject puts it bluntly: "You don't have a restore strategy until you test your backups." Restore a copy onto a test system on a schedule and run `DBCC CHECKDB` against it. Backup checksums detect problems with the backup media itself.

A bulk-copy file arrives without its table. Files written by `bcp` carry the rows and nothing about their shape: no schema and no format information. When the source table has changed or gone, importing them again needs either a format file or a table with an identical definition. Microsoft recommends native format (`-n`) between two SQL Server instances and character format (`-c`) when the data crosses into another system.

The free edition fills up. Express suits learning and small applications, and 10 GB per database in 2022, or 50 GB in 2025, is a small allowance for anything that keeps history. Plan the move to a licensed edition, and its budget, before the database reaches the cap.

An old release keeps running on the security-only track. A server on SQL Server 2019 still works, and until January 2030 it receives security updates alone. The instances in the first project above ran 2019 between 2022 and 2024, inside mainstream support. Because backups restore forward, the way off such a release is a restore onto a newer instance, then a test of everything that talks to it.

Further reading

- What is SQL Server? — Microsoft's overview of the engine, where it can be installed and the services around it.
- SQL Server 2019 lifecycle — the start, mainstream and extended support dates for the release the practice section ran.
- Editions and supported features of SQL Server 2025 — compute, memory and database-size limits per edition, with notes on what changed since 2022.
- What is SQL Server on Linux? — the release Linux support started with and the distributions each release runs on.
- RESTORE statements (Transact-SQL) — the full syntax, `MOVE` and `FILELISTONLY`, and the rules for restoring across versions.
- bcp utility — bulk copy in and out of the Database Engine, data formats and format files.

Glossary

Choose for your actual constraints

Bring a representative dataset, the questions it must answer and the current bottleneck. We can agree how to measure quality, throughput and recovery.

See what it costs Get in touch →

Make the terminology practical

Ask how "What is SQL Server?" applies to your organisation. A concrete example from your current system is enough to start the discussion.

Explore our services

welcome@engineer.company +45 50 10 61 84 Copenhagen, Denmark

Take this page with you

Contact card

Carefully Crafted With Love by D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – present · Engineer ApS · All rights reserved.

Printed from https://engineer.company/glossary/sql-server/