Scale out horizontally

Autoscaling

FAQ

Resources

**Solutions** 

Sign In

Azure

Solution overview

Scaling up vs. scaling out An intro to database scalability in cloud computing.

Scale up vertically

talk about scalability

growing amount of work, when we talk about whether to scale up vs. scale out, we are frequently referring to databases and data—and lots of it.

Data, data everywhere—what we talk about when we

Scalability in <u>cloud computing</u> is the ability to quickly and easily increase

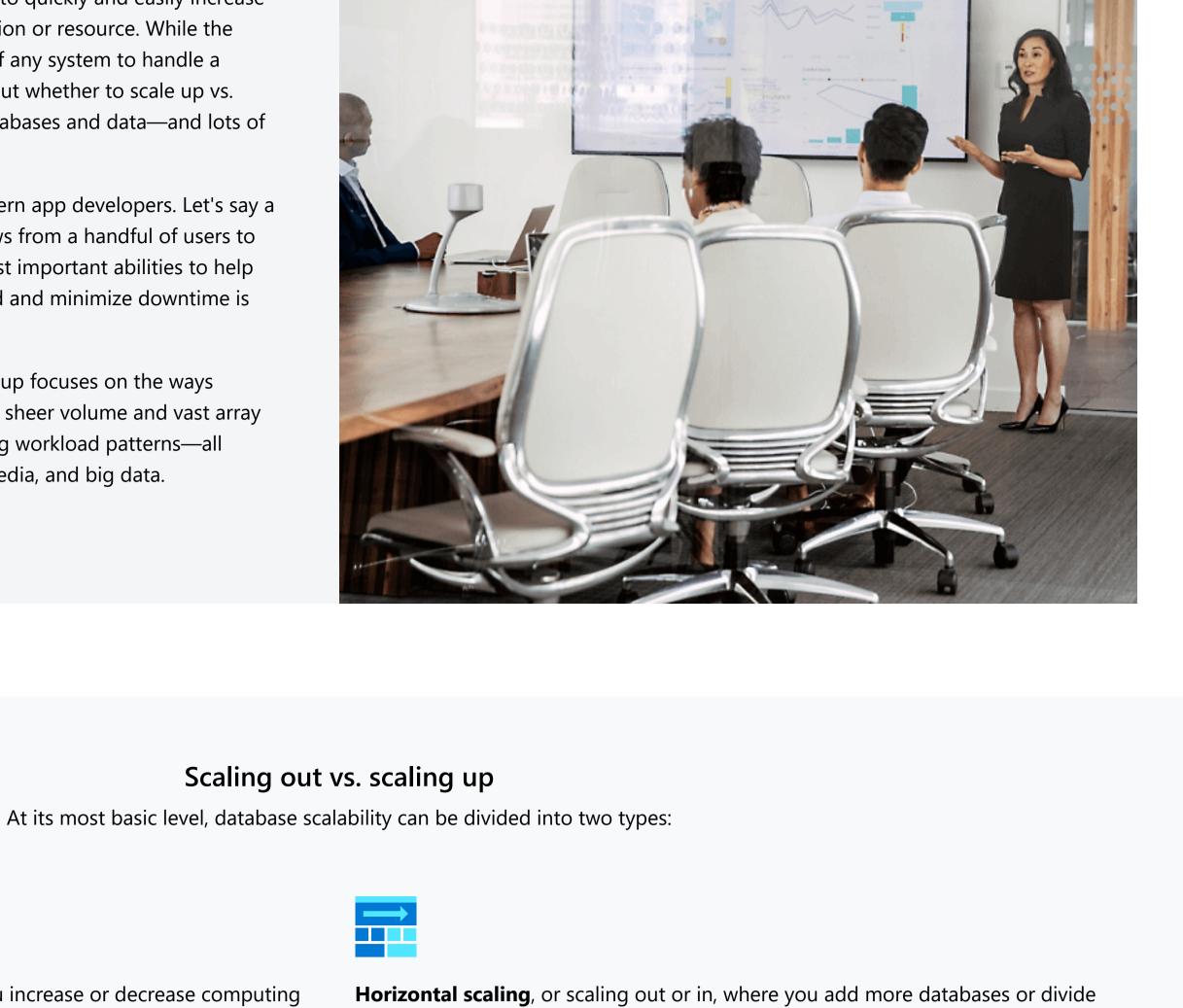
or decrease the size or power of an IT solution or resource. While the

term scalability can refer to the capability of any system to handle a

Database scalability is top of mind for modern app developers. Let's say a new app takes off—and demand for it grows from a handful of users to millions of users worldwide. One of the most important abilities to help the app developers keep pace with demand and minimize downtime is the ability to scale efficiently. This conversation on scaling out vs. scaling up focuses on the ways scalability helps us to adapt and handle the sheer volume and vast array

generated from the cloud, mobile, social media, and big data. Learn more about databases >

of data, changing data volumes, and shifting workload patterns—all



## power or databases as needed—either by changing performance levels or by using your large database into smaller nodes, using a data partitioning approach called elastic database pools to automatically adjust to your workload demands. sharding, which can be managed faster and more easily across servers.

Scaling up vertically

You need to quickly react to fix performance

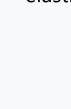
Scaling out horizontally

rapidly changing data.

database restores in only minutes.

optimization.

issues that can't be solved with classic database



You see that your workloads are hitting some

performance limit such as CPU or I/O limits.

Vertical scaling, or scaling up or down, where you increase or decrease computing

Learn more about database sharding

Vertical scaling is used when you need to react quickly to fix a performance issue that you can't resolve using classic database optimization techniques—such as query changes or indexing. Scaling up is useful to handle spikes in your workloads where the current performance level cannot satisfy all the demands. Scaling up lets you add more resources to easily handle peak workloads. Then, when the resources are not needed anymore, scaling down lets you go back to the original state and save on cloud costs.

requirements.

Some types of database technologies, most notably nonrelational or NoSQL

up or down are beginning to offer exciting options to match the scalability

storage to grow as needed, and enable nearly instantaneous backups and fast

databases, are developed with unique capabilities to scale out data horizontally

by <u>Database sharding</u>—enabling them to handle large, unrelated, indeterminate, or

And, some relational (SQL) database services that originally offered services to scale

advantages of nonrelational databases. Hyperscale services like Microsoft Azure SQL

<u>Database Hyperscale</u> and <u>Azure Database for PostgreSQL Hyperscale</u> enable users to

rapidly scale storage up to 100 TB, provide flexible, cloud-native architecture allowing

a single database.

You've maxed out your performance

requirements, even in the highest performance

tiers of your service, or if your data cannot fit into

You need a solution that allows you to change

service tiers to adapt to changing latency

Scale up when:

enough resources for their workloads, even operating on the highest performance levels. With horizontal scaling, data is split into several databases, or shards, across servers, and each shard can be scaled up or down independently. How does partitioning data improve scalability? When you scale up a single database by adding resources such as virtual machines (VMs), it will eventually reach a physical

hardware limit. Because data partitions are each hosted on a separate server, if you

divide data across multiple shards, you can scale out a system almost limitlessly.

App developers start to consider scaling out or horizontal scaling when they can't get

Scale out when: You have geo-distributed applications where You have a global sharding scenario—such as

every app should access part of the data in the

associated to that region without affecting other

region. Each app will access only the shard

Autoscaling is the process of automatically and

performance requirements of a system. As the volume

of work grows, apps may need additional resources to

maintain the necessary performance levels or meet

additional resources are no longer needed, you can

save on cloud spend by having an automatic service in

Autoscaling takes advantage of the elasticity of cloud-

hosted environments. It eases management overhead

constantly make decisions about adding or removing

by reducing the need for system operators to

resources or checking the system's performance.

growing demand. If demand slows down and the

place to de-allocate unused resources.

dynamically matching resources to meet the

shards.

load balancing—where you have a large number

of geo-distributed clients that insert data in their

**Autoscaling** 

While there are two main ways that apps can scale—

automate vertical scaling, because scaling up often

requires making the system temporarily unavailable

Autoscaling is more common when scaling horizontally

resources are provisioned. If demand drops, resources

can be shut down seamlessly without downtime and

Many providers of cloud-based systems, such as

vertically or horizontally— it's less common to

because scaling out or in means just adding or

removing instances of a resource and your app

continues running without interruption as new

while it is being redeployed.

own dedicated shards.

Microsoft Azure, support automatic horizontal scaling.

• Learn more about scalability with Azure SQL **Database** 

• Learn more about fast NoSQL database

autoscaling with Azure Cosmos DB

Expand all | Collapse all

de-allocated.

Resources

Frequently asked questions

Learn more cloud computing terms >

What are NoSQL databases?

> What are databases?

> What is PostgreSQL

> What is caching?

> What is a platform as a service (PaaS)?

> What is database sharding?

Create a Hyperscale (Citus) server group in the Azure portal

Alternative methods to scale your

Quickstarts and learning modules

Explore Azure database and analytics

**Dynamically scale database resources** 

with minimal downtime with Azure SQL

services

database

Discover, assess, and migrate on-prem apps, infrastructure, and data with Azure <u>Migrate</u>

Accelerate, guide, and automate your

Start your cloud journey at the Azure

**Database migration** 

database migration

migration center

Related products and services

**Explore cloud scalability with Azure** 

developers.

**Azure SQL** 

development.

Family of SQL cloud databases providing flexible

options for app migration, modernization, and

Discover a comprehensive approach to scaling up vs. scaling out—one

that fits your own scenario across on-premises, multicloud, and edge

environments. The Azure family of database services offers a choice of

fully managed relational, NoSQL, and in-memory databases, spanning

proprietary and open-source engines, to fit the needs of modern app

Save time and money with automated infrastructure management—

Find the right database products for your needs with Azure >

including automation solutions for scalability, availability, and security.

**Azure Cosmos DB** 

scale.

cloud.

developers.

**Azure SQL Managed Instance** 

Fast NoSQL database with open APIs for any

**Azure Cache for Redis** 

low-latency data caching.

**SQL Server on Virtual Machines** 

Migrate SQL Server workloads to the cloud at

**Azure PostgreSQL** 

PostgreSQL.

the lowest TCO.

Fully managed, intelligent, and scalable

**Azure Database for MySQL** Fully managed, scalable MySQL database.

**Azure SQL Database** 

Managed intelligent SQL in the cloud.

**Azure Maria DB** 

Managed MariaDB database service for app

Managed, always up-to-date SQL instance in the

Accelerate applications with high-throughput,

Ready when you are—let's set up your free account

Start free

Scale without limits with managed databases

Focus on building apps and make your job simpler with your databases managed by Microsoft

in

**Products and pricing** 

Free Azure services

Cloud economics

Optimise your costs

Flexible purchase options

Products

Pricing

Solutions

Support

Solution architectures

Azure demo and live Q&A

Azure.

**Get started** 

Solutions and support

**Partners** Azure Marketplace Find a partner Join ISV Success

Resources for accelerating growth

Resources Blog

Training and certifications Documentation Developer resources Students

Analyst reports, white papers and e-

**Events and webinars** 

books

**Videos** 

What is AI? What is laaS? What is SaaS? What is PaaS? What is DevOps?

**Q** Chat with sales

Cloud computing

What is cloud computing?

What is cloud migration?

What is a hybrid cloud?

Change language English (India)

Get the Azure mobile app

**Explore Azure** 

What is Azure?

Global infrastructure

Datacentre regions

Customer enablement

Trust your cloud

Customer stories

Get started

Consumer Health Privacy Diversity and Inclusion Accessibility Privacy & Cookies Data Protection Notice Trademarks Terms of use **Your Privacy Choices** Privacy Data Management Contact us Feedback Sitemap © Microsoft 2024