



**#ASLI ENGINEERING**

# An Introduction to Sharding and Partitioning



**BY**

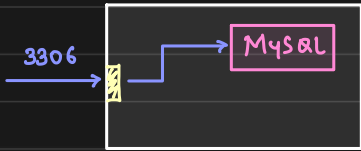
**ARPIT BHAYANI**

# Sharding and Partitioning

**Sharding**: method of distributing data across multiple machines


**Partitioning**: splitting a subset of data within the same instance

How a database is scaled?



Virtual Server  
Ec2 machine

A database server is just a database process (mysqld, mongod) running on an Ec2 machine.

And we represent this as 

You put your database in production, serving real traffic



You are getting more users, that your DB is unable to manage

you scale up your DB ... give it more CPU, RAM and DISK



Bulkier server + Read Replica

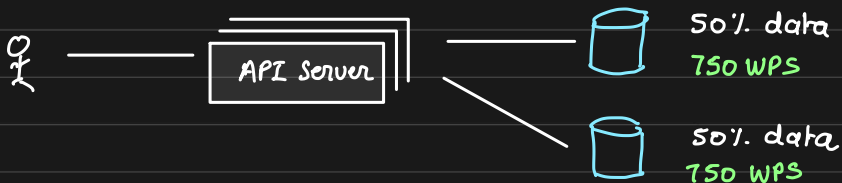
Your product went viral and your bulky database is unable to handle the load, so you scale up again



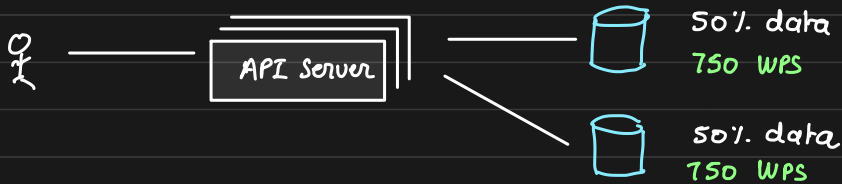
But, after a certain stage you know you would not be able to scale "up" your DB because **Vertical scaling has limit**

So, you will have to resort to **Horizontal Scaling**

Say, one DB server was handling **1000 WPS** and we cannot scale up beyond that but we are getting **1500 WPS**, we scale horizontally and split the data



By adding one more database server, we reduced the load to 750 WPS on each node and thus handled **higher throughput**

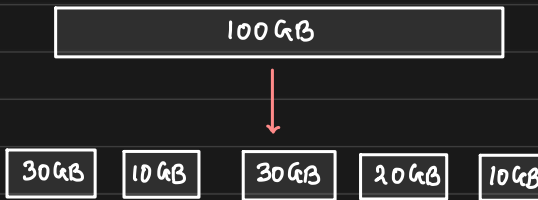


Each database server is thus a **shard**  
and we say that the data is **partitioned**

Overall, a database is sharded while the data is partitioned

↗  
Oversimplification, most people use the  
terms interchangeably

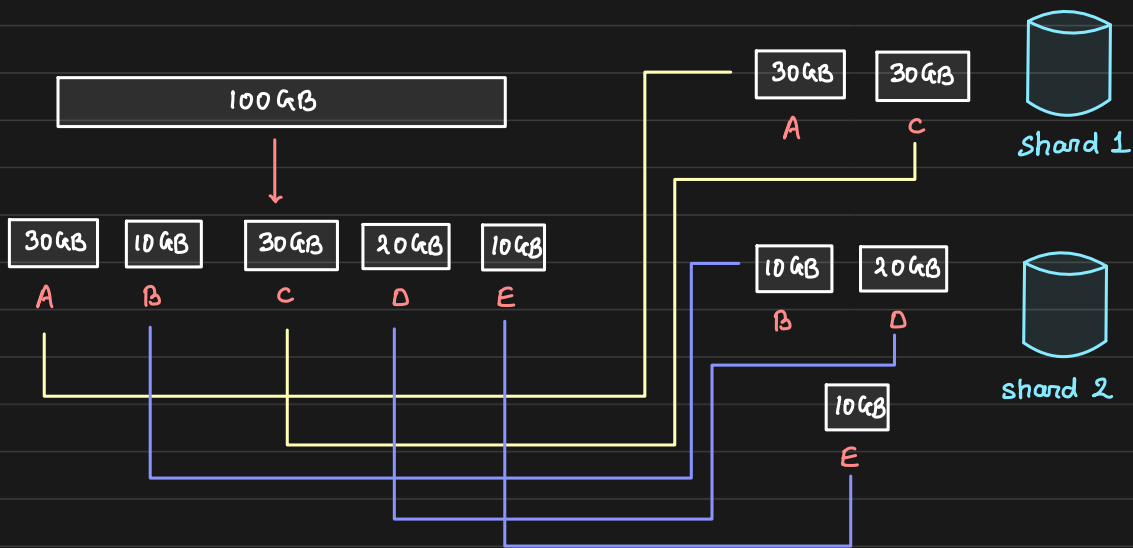
↑  
Split across



You partitioned the 100GB of  
total data into 5 mutually  
exclusive partitions.

Each of these partitions can either live on  
one database server or a couple of  
them can share one server.

And this depends on the **# shards** you have

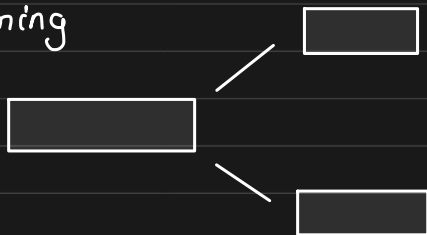


5 partitions of our 100GB dataset is distributed across 2 shards

How to partition the data?





There are two categories of partitioning

1. Horizontal Partitioning
2. Vertical Partitioning



When we "split" the 100GB data, we could have used either of the ways but deciding which one to pick depends on load, usecase, and access pattern.

# Sharding and Partitioning

		Partitioning	
		No	YES
sharding	No		
	YES	 Read Replica	

## Advantages of sharding

- Handle large Reads and Writes
- Increase overall storage capacity
- Higher availability

## Disadvantages of sharding

- operationally complex
- cross-shard queries expensive