

[System Design Architectural patterns]

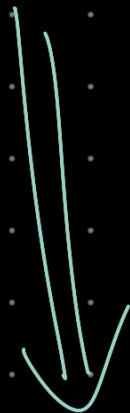
① Client-Server

↳ Consider this as monolith.



② Microservice

③ Serverless



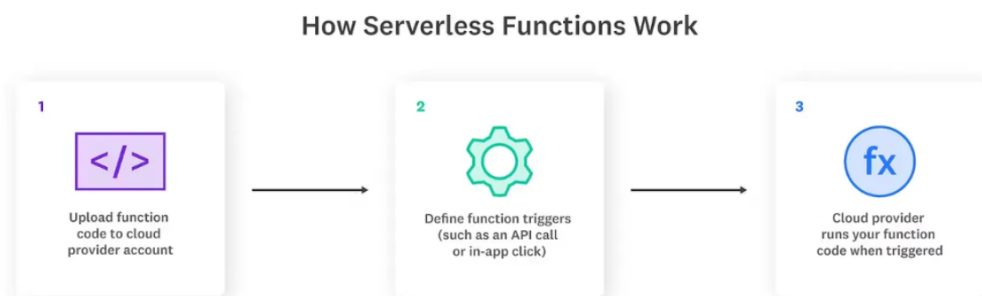
Serverless architecture, despite its name, does not mean that servers are no longer involved. Instead, it refers to a model where the cloud provider dynamically manages the allocation of machine resources, and developers deploy their code in the form of functions or services without worrying about the underlying infrastructure.

In a serverless setup:

- **Developers write functions:** Code is deployed as small, discrete units of functionality, typically in the form of event-driven functions.
- **Cloud providers manage servers:** The cloud provider (e.g., AWS, Azure, Google Cloud) automatically provisions, scales, and manages the infrastructure needed to execute the code.
- **Billing is based on execution:** Users are charged only for the compute resources they consume, typically measured by the number of executions, duration of the execution, and the amount of memory used.

How Does Serverless Architecture Work?

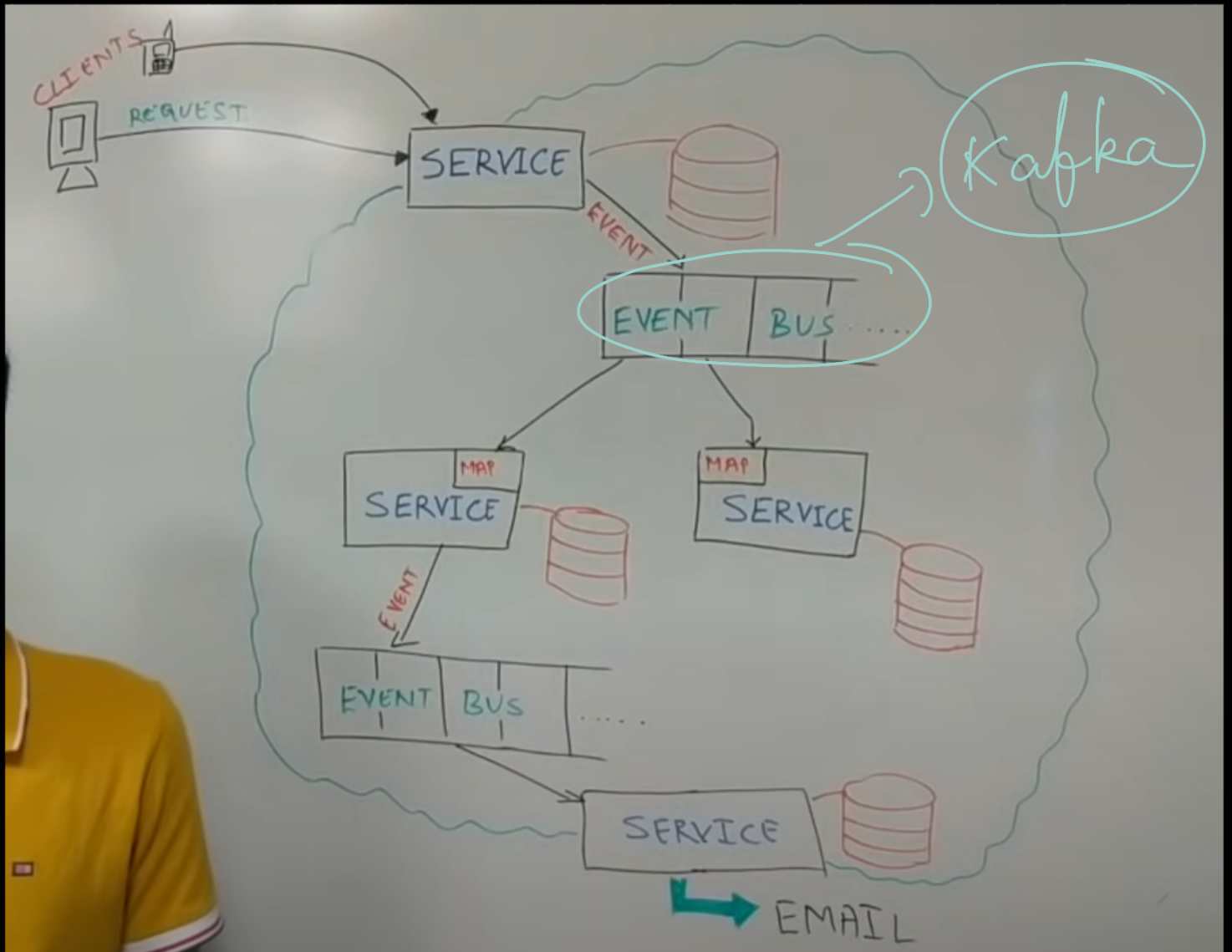
Serverless architecture is centered around the concept of "Functions as a Service" (FaaS).



Source: Datadog

Source: <https://www.datadoghq.com/knowledge-center/serverless-architecture/>

④ Event driven Architecture



⑤ Peer to peer

↳ Decentralized

↳ Eg → Torrent, Blockchain, VPN,
CON