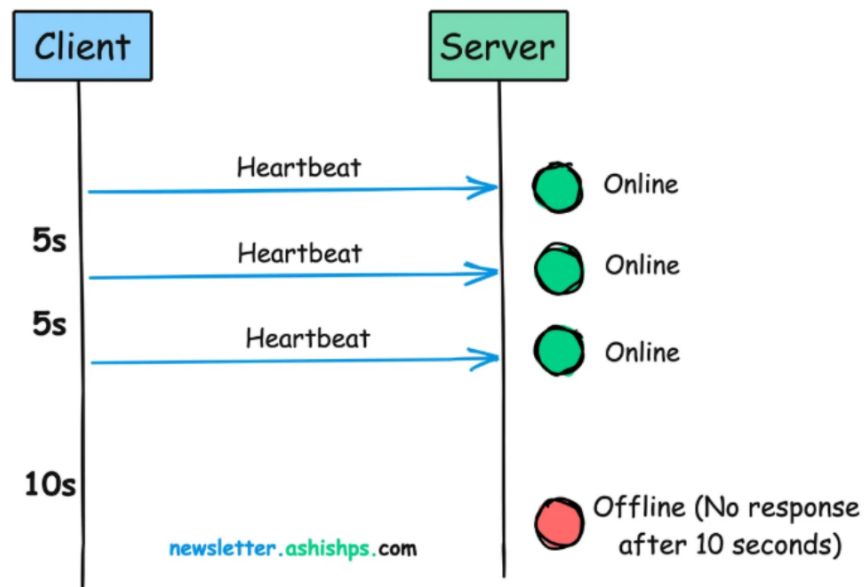


[Heartbeats]

But, how do we know if a particular service is alive and working as expected?

This is where **heartbeats** come into play.



↳ E.g. → /healthz route in our server

What exactly is a Heartbeat?

In distributed systems, a heartbeat is a periodic message sent from one component to another to monitor each other's health and status.

Its primary purpose is to signal, "Hey, I'm still here and working!"

This signal is usually a small packet of data transmitted at regular intervals, typically ranging from seconds to minutes, depending on the system's requirements.



Why Do We Need Heartbeats?

Without a heartbeat mechanism, it's hard to quickly detect failures in a distributed system, leading to:

- Delayed fault detection and recovery
- Increased downtime and errors
- Decreased overall system reliability

Heartbeats can help with:

- Monitoring: Heartbeat messages help in monitoring the health and status of different parts of a distributed system.
- Detecting Failures: Heartbeats enable a system to identify when a component becomes unresponsive. If a node misses several expected heartbeats, it's a sign that something might be wrong.
- Triggering Recovery Actions: Heartbeats allow the system to take corrective actions. This could mean moving tasks to a healthy node, restarting a failed component, or letting a system administrator know that they need to step in.
- Load Balancing: By monitoring the heartbeats of different nodes, a load balancer can distribute tasks more effectively across the network based on the responsiveness and health of each node.

Heartbeats in Action: Real-World Examples

- **Database Replication:** Primary and replica databases often exchange heartbeats to ensure data is synchronized and to trigger failover if the primary becomes unresponsive.
- **Kubernetes:** In the Kubernetes container orchestration platform, each node sends regular heartbeats to the control plane to indicate its availability. The control plane uses these heartbeats to track the health of nodes and make scheduling decisions accordingly.
- **Elasticsearch:** In an Elasticsearch cluster, nodes exchange heartbeats to form a gossip network. This network enables nodes to discover each other, share cluster state information, and detect node failures.

Heartbeats are the invisible pulses that keep distributed systems alive and well-coordinated.