HeartBeats: How Distributed Systems Stay

#8 System Design - Heartbeats

ASHISH PRATAP SINGH
APR 20, 2024

♡ 85 💬 7 🔁 3

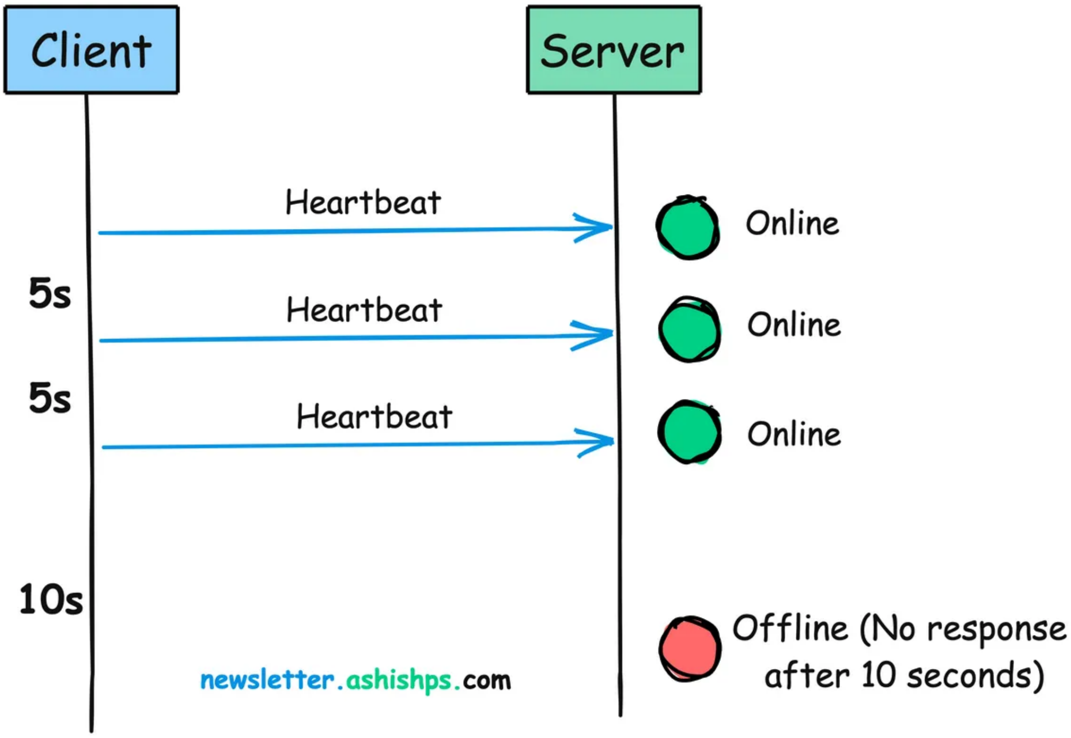
In a distributed system, **things fail.**

Hardware malfunctions, software crashes, or network connections drop.

Whether you're watching your favorite show online, making an online purc checking your bank balance, you're relying on a complex network of interco services.

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

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



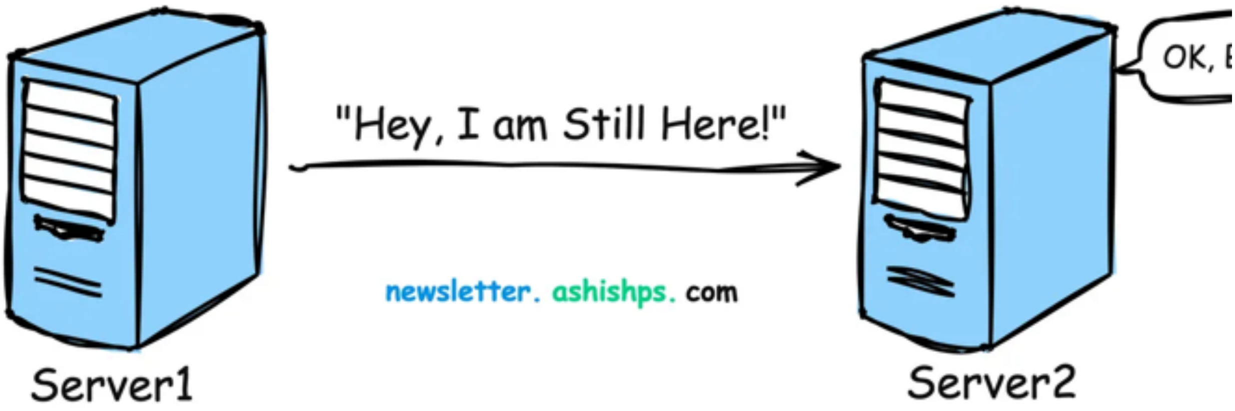
In this article, we'll learn about what heartbeats are, why they're important, work, and real-world examples where they're used.

What exactly is a Heartbeat?

> In distributed systems, a heartbeat is a periodic message sent from one c 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, 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 dist 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 sta different parts of a distributed system.
- **Detecting Failures:** Heartbeats enable a system to identify when a com becomes unresponsive. If a node misses several expected heartbeats, it' that something might be wrong.
- **Triggering Recovery Actions:** Heartbeats allow the system to take corr actions. This could mean moving tasks to a healthy node, restarting a fa component, or letting a system administrator know that they need to st
- **Load Balancing**: By monitoring the heartbeats of different nodes, a loa can distribute tasks more effectively across the network based on the

responsiveness and health of each node.

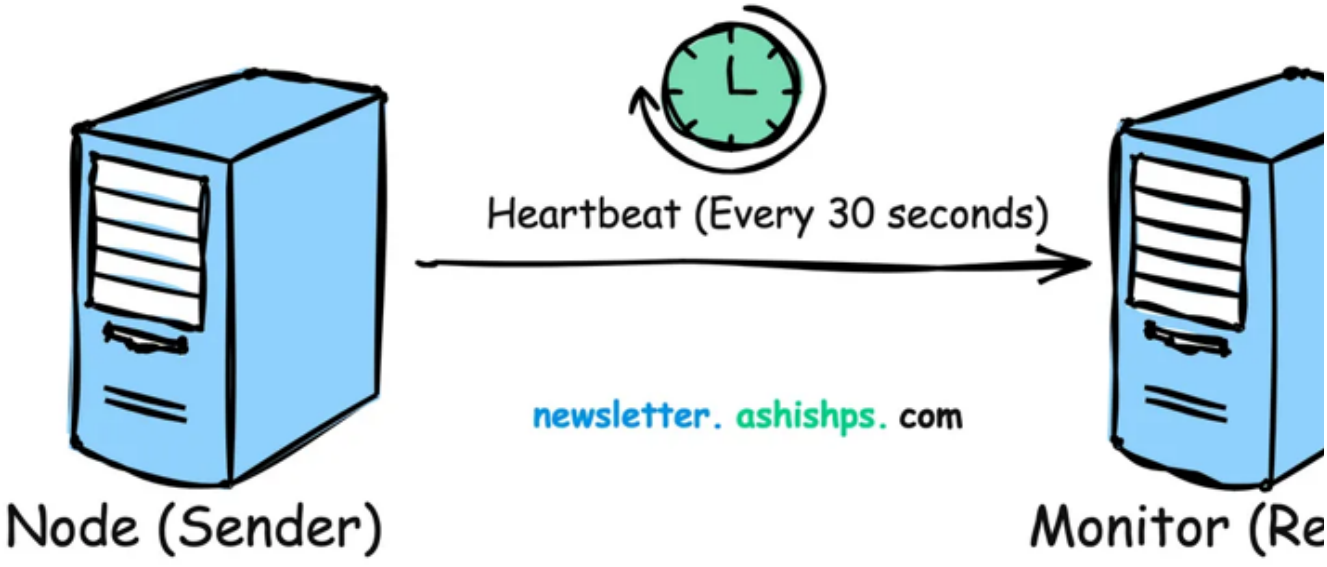
How Do Heartbeats Work?

The heartbeat mechanism involves two primary components:

1. **Heartbeat sender** (**Node**): This is the node that sends periodic heartbea
2. **Heartbeat receiver** (**Monitor**): This component receives and monitors t heartbeat signals.

Here's a simplified overview of the process:

1. The **node** sends a heartbeat signal to the **monitor** at regular intervals (e seconds).
2. The monitor receives the heartbeat signal and updates the node's statu or "**available**".
3. If the monitor doesn't receive a heartbeat signal within the expected ti marks the node as "**unavailable**" or "**failed**".
4. The system can then take appropriate actions, such as redirecting traffi initiating failover procedures, or alerting administrators.



While conceptually simple, heartbeat implementation has a few nuances:

- **Frequency:** How often should heartbeats be sent? There needs to be a b they're sent too often, they'll use up too much network resources. If the too infrequently, it might take longer to detect problems.
- **Timeout:** How long should a node wait before it considers another nod This depends on expected network latency and application needs. If it's it might mistake a live node for a dead one, and if it's too slow, it might longer to recover from problems.
- **Payload:** Heartbeats usually just contain a little bit of information like a timestamp or sequence number. But, they can also carry additional data much load a node is currently handling, health metrics, or version infor

Types of Heartbeats

There are two primary types of heartbeats:

1. **Push heartbeats**: Nodes actively send heartbeat signals to the monitor.
2. **Pull heartbeats**: The monitor periodically queries nodes for their status

Challenges and Considerations

While heartbeats are a fundamental part of maintaining system integrity, th without challenges:

- **Network Congestion**: If not managed correctly, the constant flow of he signals can contribute to network congestion.
- **False Positives**: Poorly configured heartbeat intervals might lead to fals in failure detection, where a slow but functioning component is incorre identified as a failed one.
- **Resource Usage**: Continuous monitoring requires computational resou must be optimized to prevent undue strain on the system.

- **Split-Brain Scenarios:** In some rare cases, a network failure can partiti system, and both sides might declare the other dead. This requires mor sophisticated failure-handling mechanisms.

Heartbeats in Action: Real-World Example

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

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

So, the next time you encounter a distributed system, take a moment to app silent guardians – the heartbeats – that work tirelessly to keep the system's steady and strong.

Thank you for reading!

If you found it valuable, hit a like ❤️ and consider subscribing for more suc every week.

If you have any questions or suggestions, leave a comment.

This post is public so feel free to share it.

P.S. If you’re enjoying this newsletter and want to get even more value, cons
becoming a paid subscriber.

As a paid subscriber, you'll unlock all **premium articles** and gain full access
premium courses on algomaster.io.

There are group discounts, gift options, and referral bonuses available.

Checkout my Youtube channel for more in-depth content.

Follow me on LinkedIn, X and Medium to stay updated.

Checkout my GitHub repositories for free interview preparation resources.

I hope you have a lovely day!

See you soon,
Ashish

Did you enjoy this post?

Share it with your network to spread the word

Share now

---

85 Likes · 3 Restacks

← Previous

Discussion about this post

Comments Restacks

Write a comment...

Priya 20 Apr 2024

Liked by Ashish Pratap Singh

One may find this interesting - https://en.m.wikipedia.org/wiki/Heartbleed

LIKE (2) REPLY

1 reply by Ashish Pratap Singh

Ashwani Yadav 21 Apr 2024

Liked by Ashish Pratap Singh

thanks a lot Ashish!. great post

LIKE (1) REPLY

1 reply by Ashish Pratap Singh

5 more comments...