HYPERTEST

How Microservices Architecture is Different from Monolithic Architecture?

How Microservices Architecture is Different from Monolithic Architecture?

Aspect	Microservices Architecture	Monolithic Architecture
Structure	Composed of multiple, small, independent services.	Composed of a single, unified, and often large codebase.
Scalability	Highly scalable as each service can be scaled independently.	Scaling often requires scaling the entire application, which can be less efficient.
Deployment	Services can be deployed independently, allowing for faster and more frequent updates.	Requires redeploying the entire application for updates, which can be slower and riskier.
Technological Flexibility	Different services can use different technologies best suited for their requirements.	Generally uses a uniform technology stack across the entire application.
Development and Maintenance	Easier to develop and maintain due to smaller and more focused codebases for each service.	Can be challenging to develop and maintain due to the large size and complexity of the codebase.
Resilience	Failure in one service does not affect the entire application, leading to higher overall resilience.	A failure in any part of the application can bring down the whole system.
Team Structure	Aligns well with small, independent teams, each responsible for different services.	Typically managed by larger, more centralized teams.
Inter-Service Communication	Requires careful management of inter-service communication, which can be complex.	Internal modules/functions communicate more easily within a single application environment.
Examples	Companies like Netflix, Amazon, and Spotify use microservices for their flexibility and scalability.	Traditional enterprise applications, early-stage startups, or small-scale applications often use this model.

HYPERTEST

95 Third Street
2nd Floor, 94103 San Francisco,
California, USA

Follow us on