

The Service Mesh



Resilience

The property of a material that enables it to *resume its original shape* or position after being bent, stretched, or compressed.

Stress

unpredictable load
flaky hardware
buggy software

Resilience

load shedding
graceful failure handling
provisioning & scaling



Dedicated hardware, big iron,
overprovisioning, buy two of 'em



2000



“Cloud Native”



2017

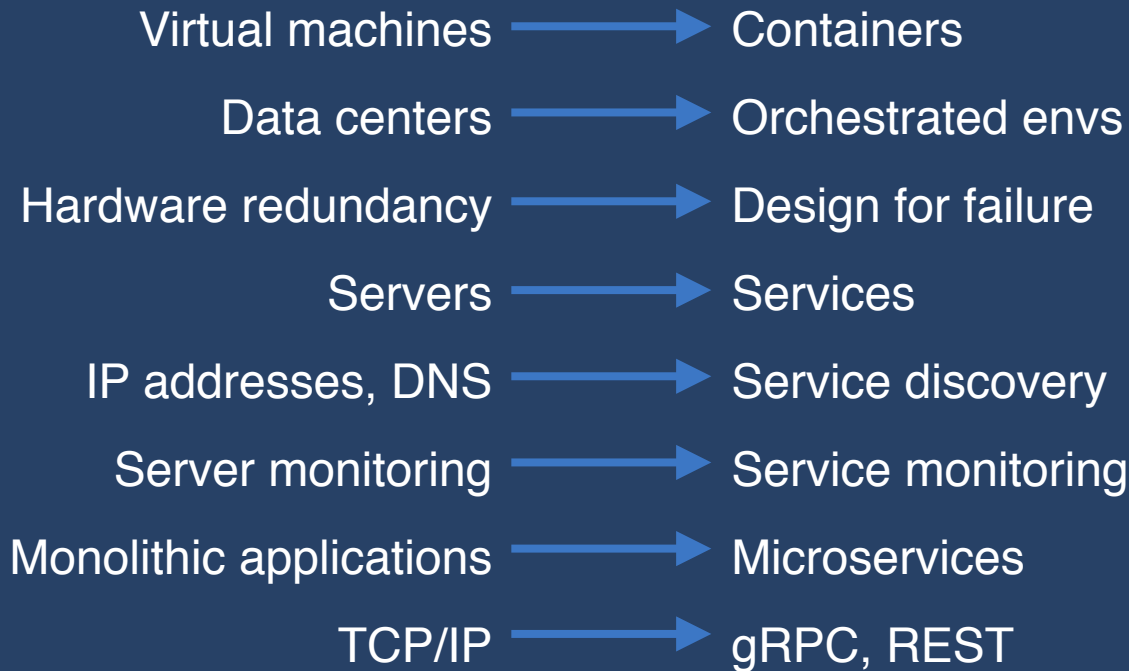


Containers

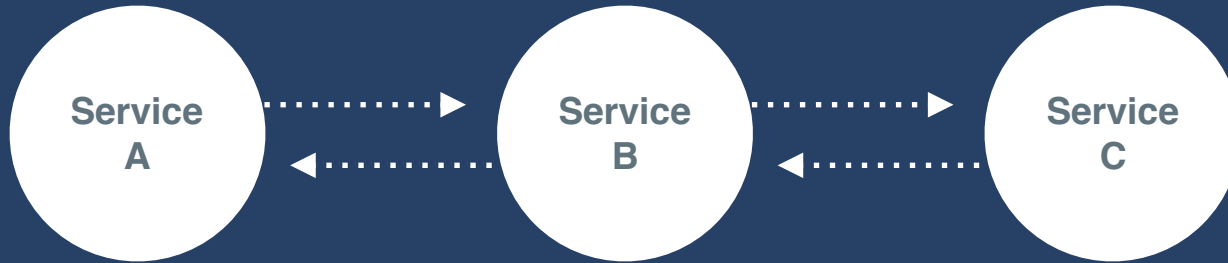
Orchestration

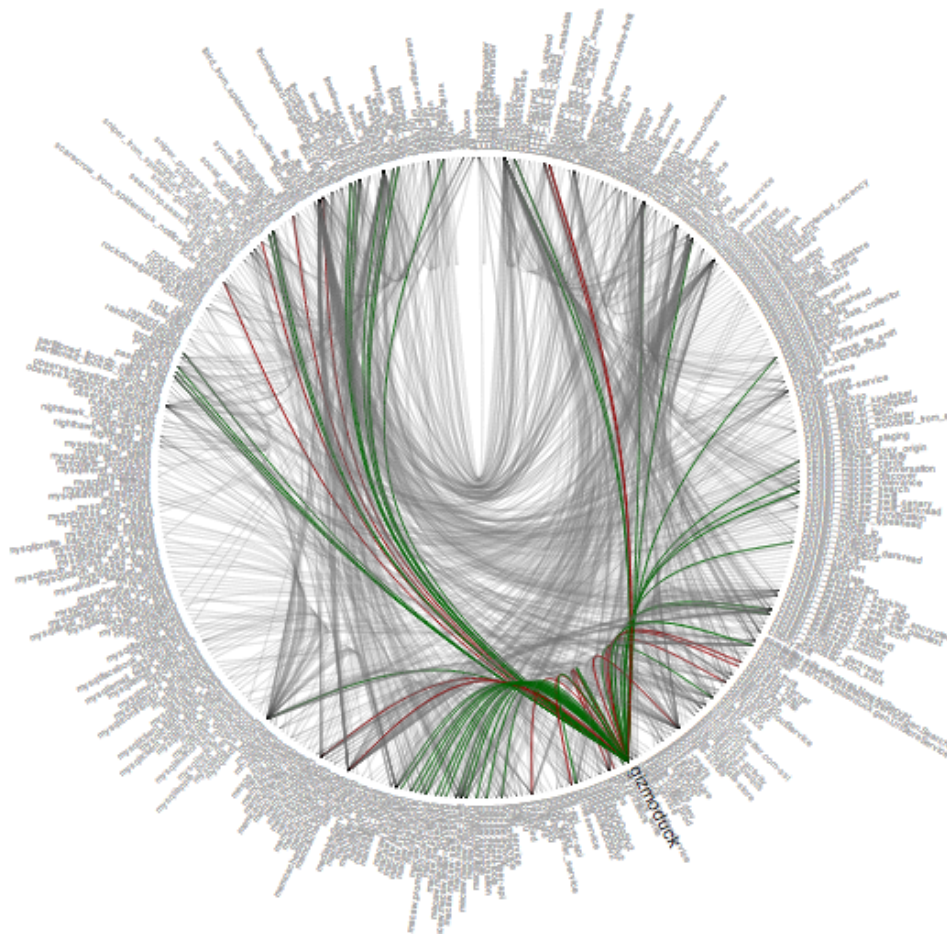
Microservices

The Fundamental Abstractions Have Changed



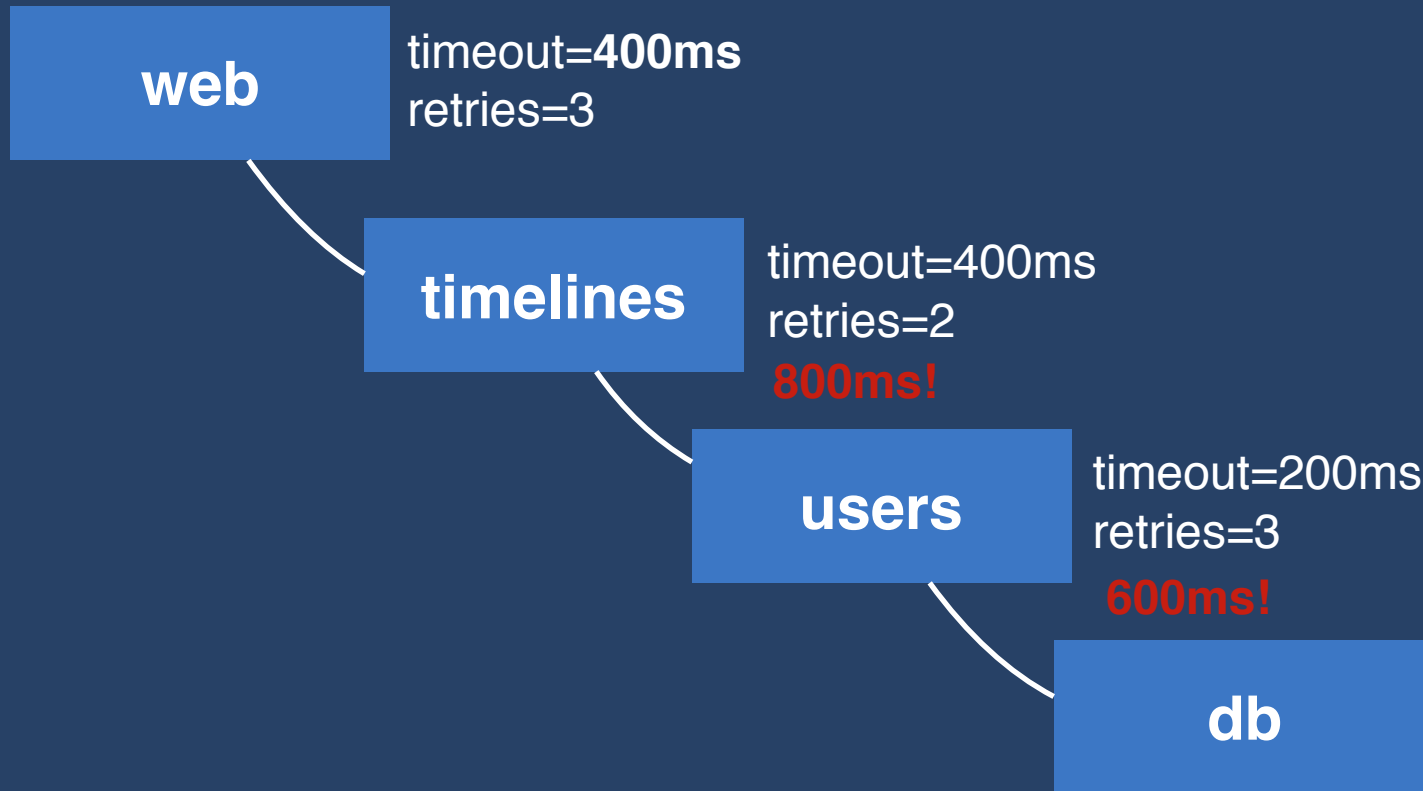
Runtime Communication





Twitter circa 2013

Example: Timeouts & Retries



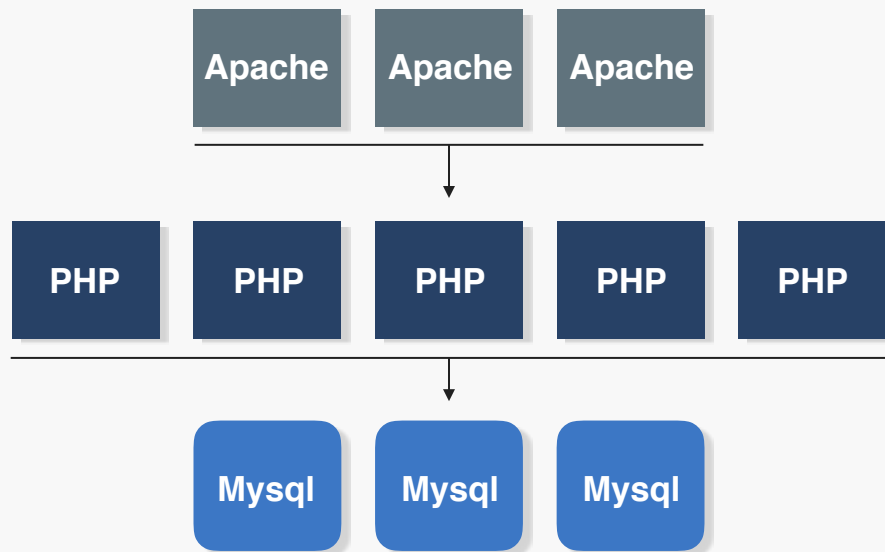


What's missing?

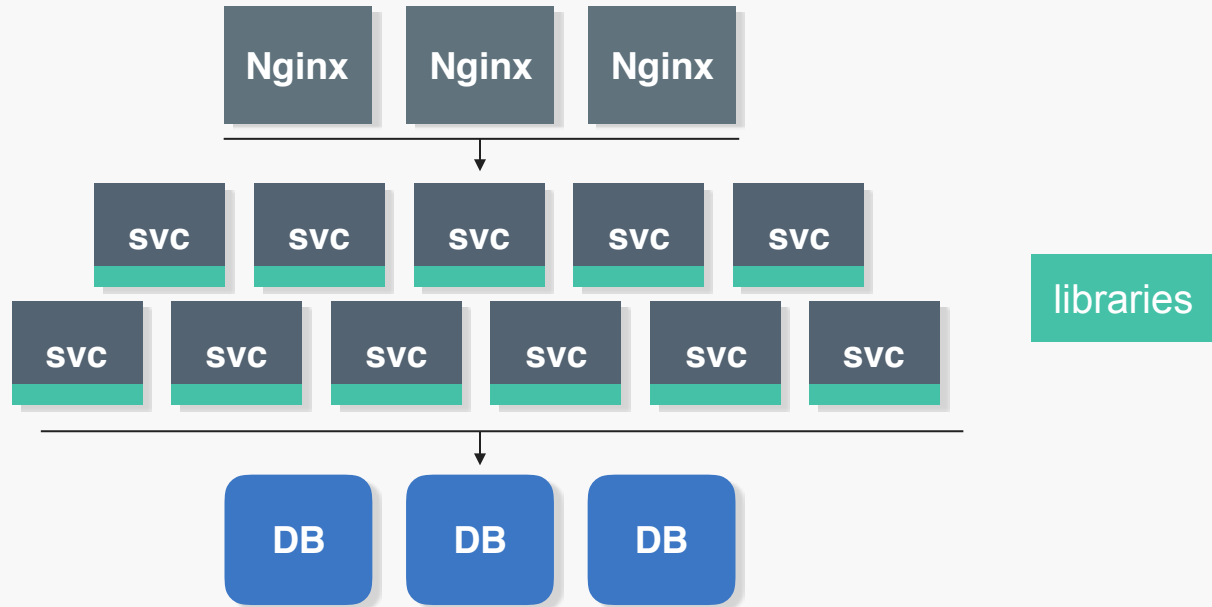
The service mesh is
a dedicated layer for managing
service-to-service communication

Managed. Monitored. Controlled.

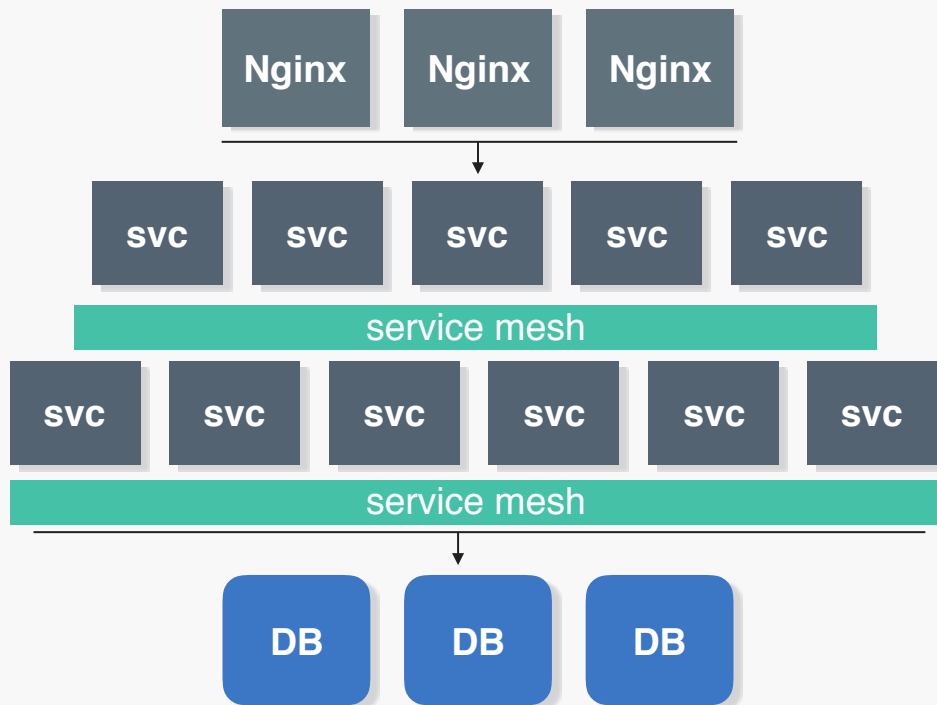
LAMP



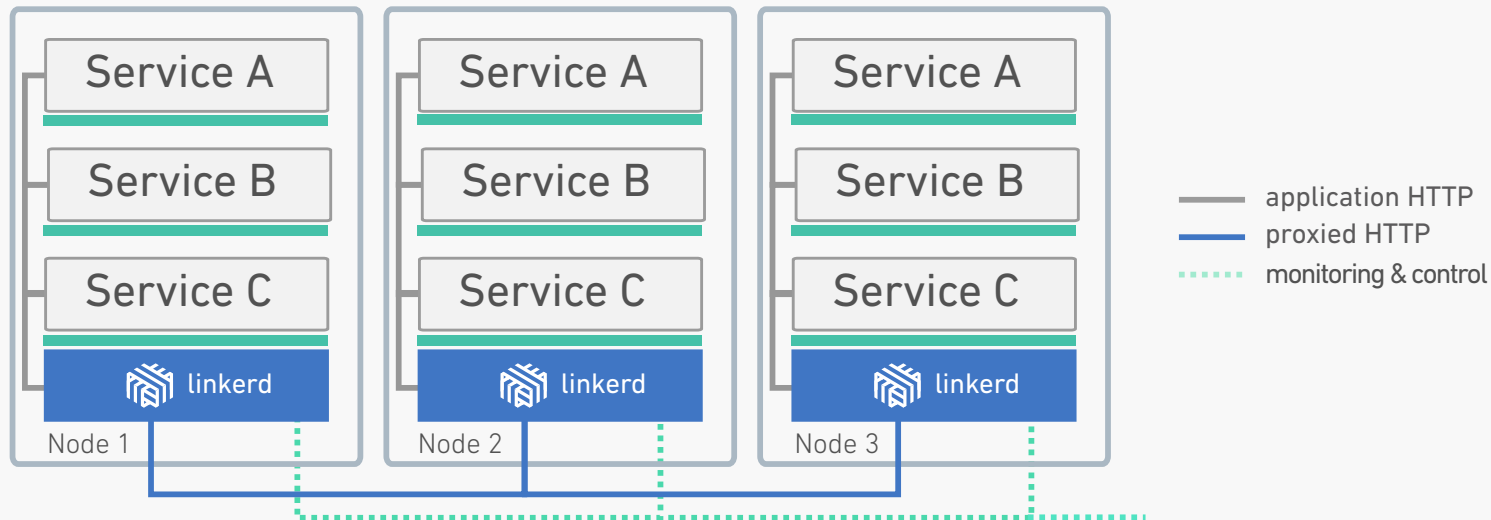
Fat clients



The service mesh



The Linkerd service mesh



Visibility Reliability Security Governance

If you're building a cloud native application,
you need a **service mesh**.





linkerd

linkerd.io



FAST, LIGHTWEIGHT, AND PERFORMANT



ANY LANGUAGE, ANY ENVIRONMENT



LATENCY-AWARE LOAD-BALANCING



RUNTIME TRAFFIC ROUTING



DROP-IN SERVICE DISCOVERY



PRODUCTION-TESTED AND PROVEN AT SCALE

Linkerd Roadmap

- Istio integration
- SPIFFE
- Dark traffic
- OpenTracing support
- Pure TCP (in beta)
- Speed, performance, memory safety! (Rust!)

- ✓ Load balancing, retries, circuit breaking, service discovery, deadlines...
- ✓ TLS (incl. mutual auth)
- ✓ gRPC, HTTP/2, HTTP, Thrift...
- ✓ Prometheus, Zipkin
- ✓ Kubernetes, Consul, ZooKeeper, Mesos, ...