Path to Resilient and Observable Microservices

Peter Jausovec

@pjausovec

Safe Harbor Statement

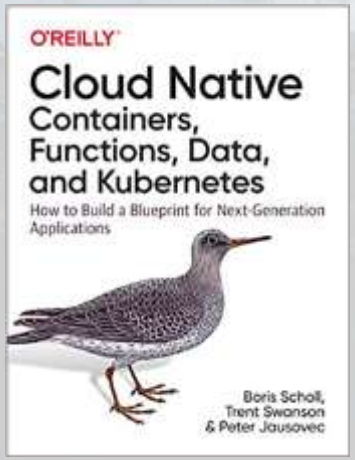
The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

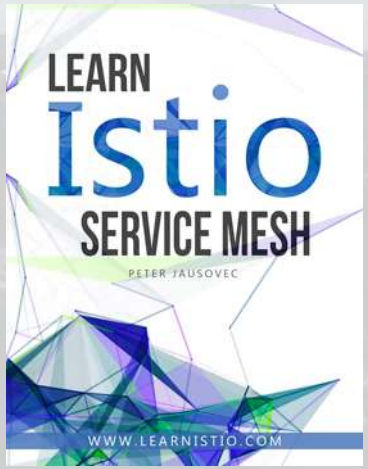
About me

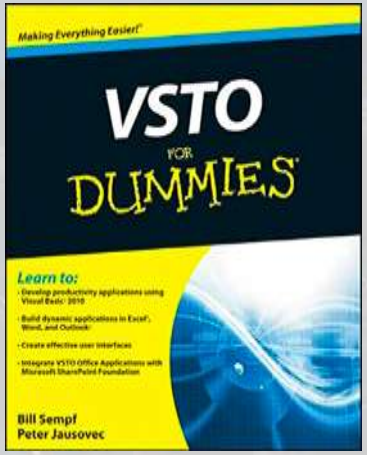
Peter Jausovec

- Consulting Member of Technical Staff at Oracle (Fn)
- Previously: Microsoft (Visual Studio, SharePoint, Azure, …)
- Wrote a couple of books

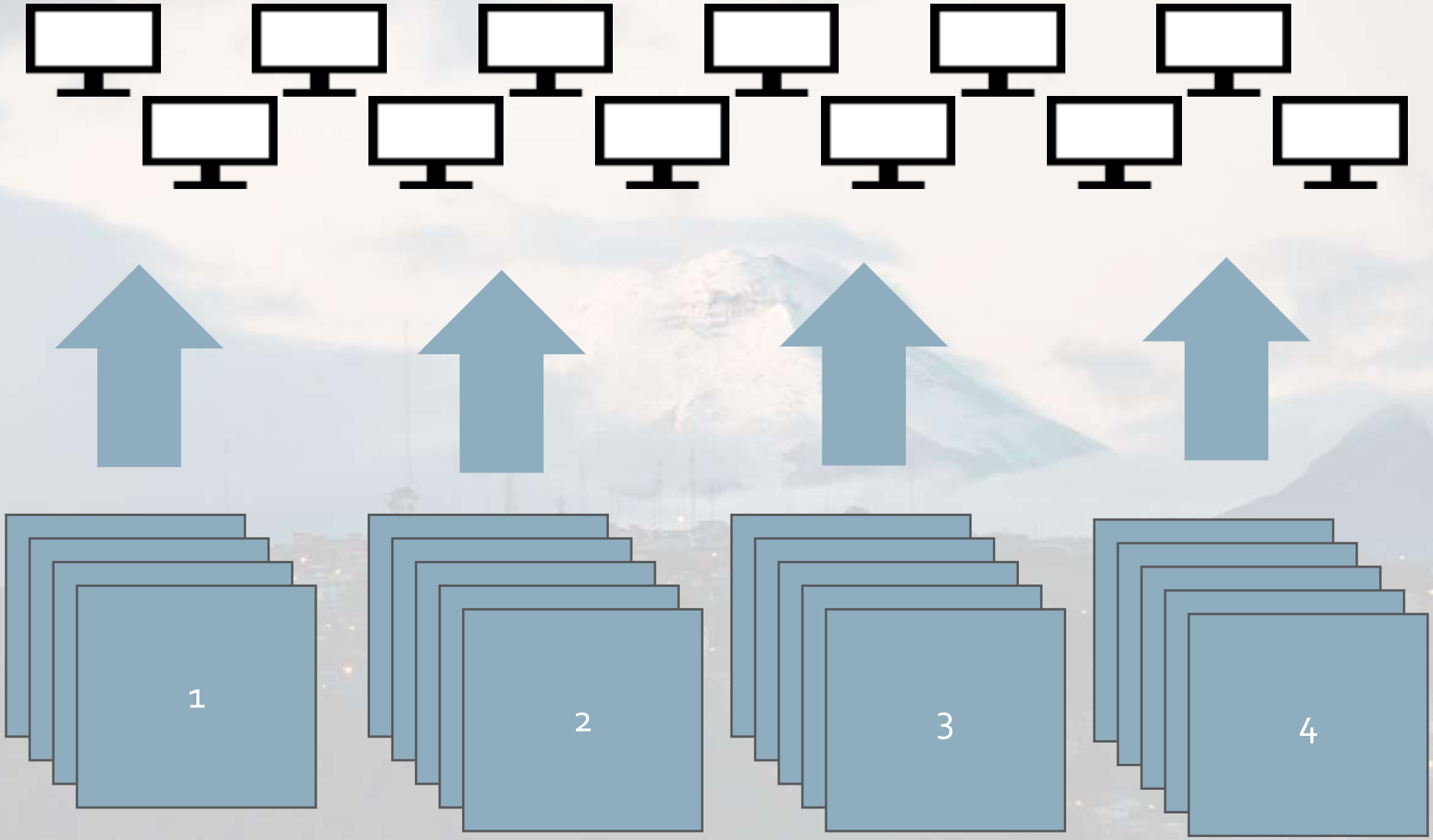
@pjausovec







Microservices



Ability to **provide and maintain** an **acceptable level** of service in the face of faults and challenges to normal operation

Ability to **recover from failures** and **continue to function**

After failure occurs,
return the service to a
fully functioning state

High Availability

- Healthy
- No (significant) downtime
- Responsive
- Meeting SLAs

Disaster Recovery

HA design can't handle the impact of faults

- Data backup
- Archiving

Path to Resiliency

Understand the requirements

Define service availability

Design for resiliency

Strategies for detection & recovery

Testing

Monitoring

Resiliency Strategies

Load balancing

Timeouts and retries

Circuit breakers and bulkhead pattern

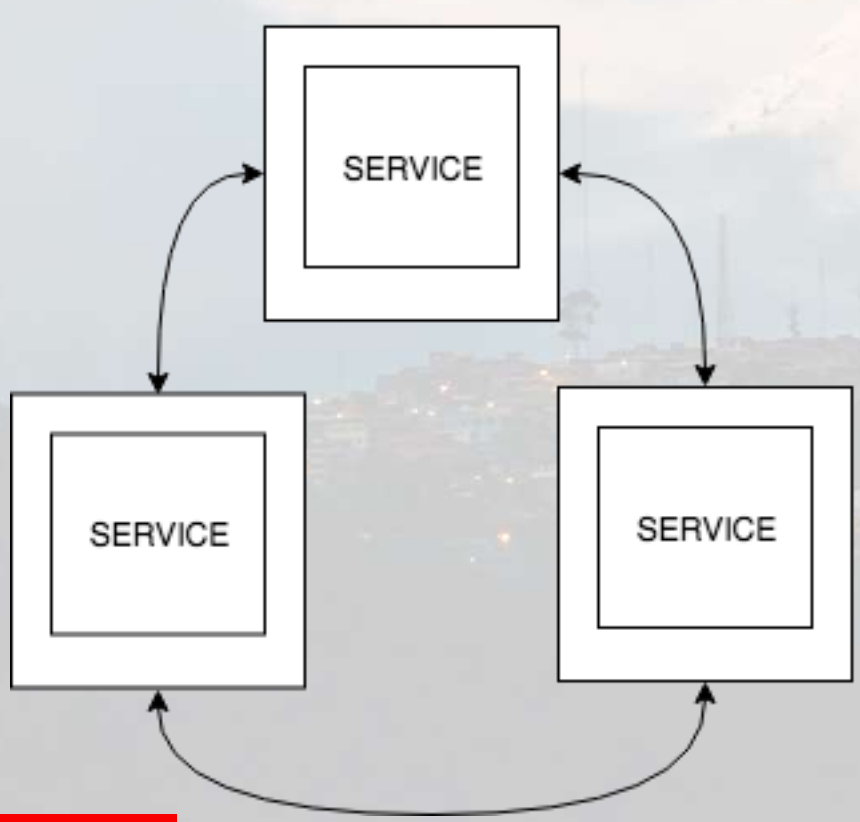
Data replication

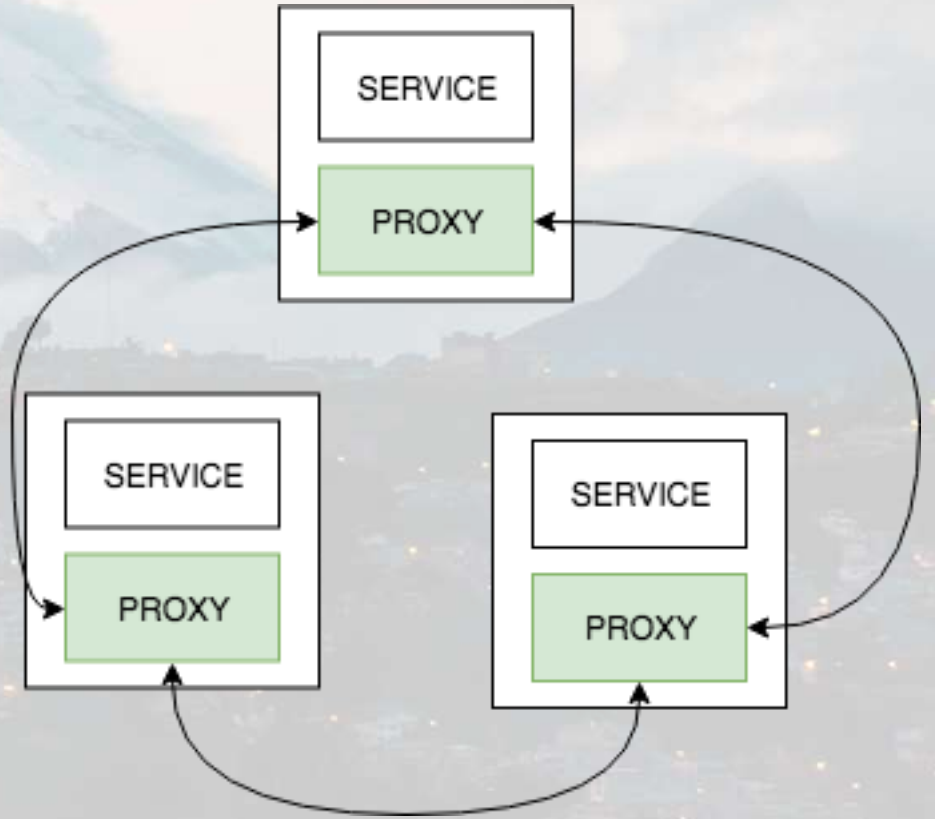
Graceful degradation

Rate limiting

How to do it?

A dedicated infrastructure layer for managing service-to-service communication to make it manageable, visible and controlled.





Resiliency Strategies

Load balancing

Timeouts and retries

Circuit breakers and bulkhead pattern

Data replication

Graceful degradation

Rate limiting

Testing for Resiliency

Test

Measure

Analyze (fix the issues)

Testing for Resiliency

SERVICE MESH

- Inject failures:
  - Delays
  - Faults

Observability is the act of measuring, collecting and analyzing metrics, traces, logs, events, … from services

instrument → collect & store → analyze, visualize and alert

Observability

Use correlation IDs

Structure logging

No private information!

Unique metric names

Categorize log entries

Use common format

Observability

SERVICE MESH

Grafana

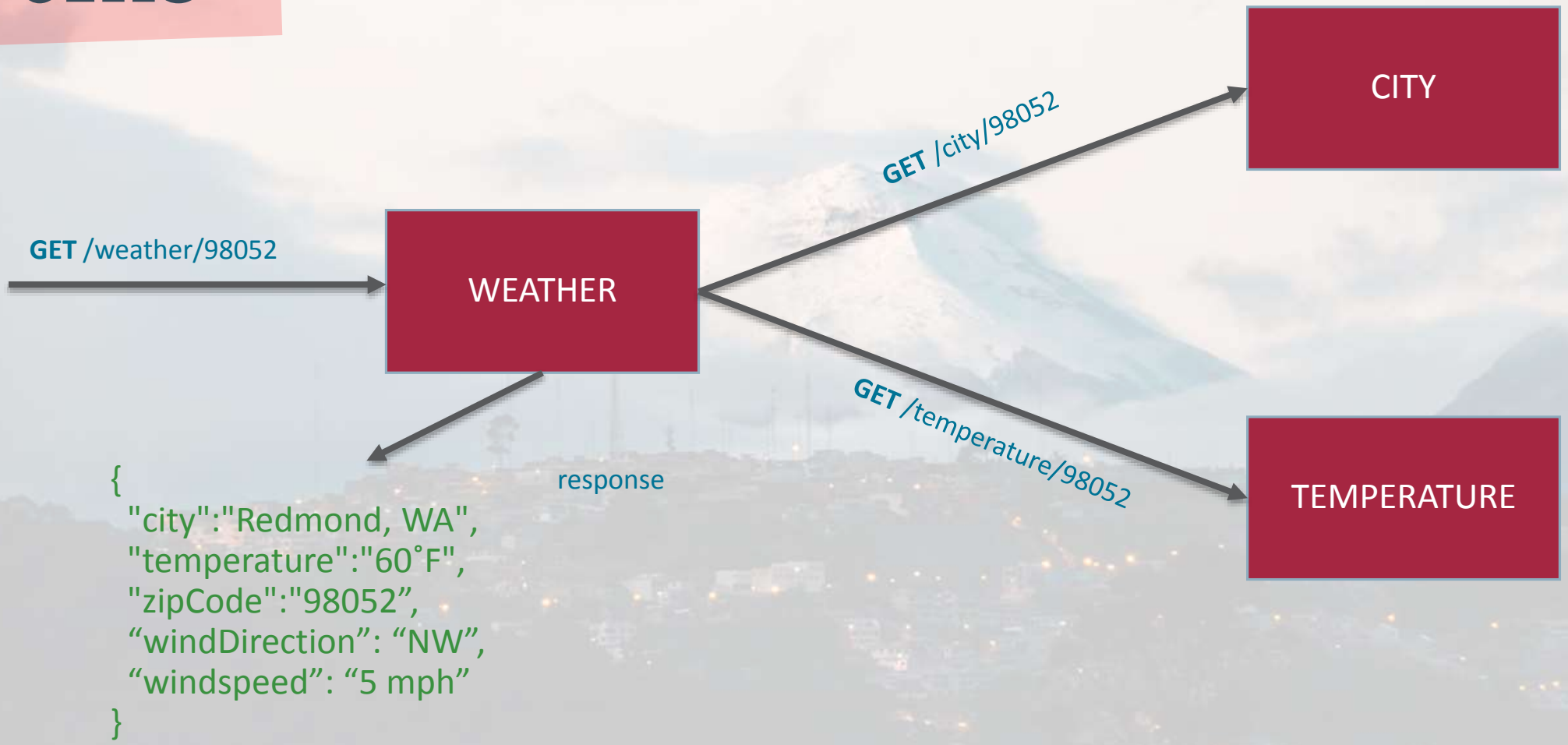
Jaeger

Kiali

ELK (Elasticsearch + Fluentd + Kibana)

PagerDuty

Demo



Thank you!

@pjausovec

https://peterj.dev