Well-Architected Reliability:

Best practices you can use today

Danny Wright (he/him)

Principal Technical Account Manager
AWS Enterprise Support

Continuous Resilience Roadmap

High availability

Disaster recovery

Continuous resilience

Best Practices

Well-Architected Framework

Resilience Modeling

Resilience Hub

Chaos Engineering

Recovery Planning

Disaster Recovery

Incident Analysis

Correction Of Errors

Agenda

Well-Architected Reliability

- Design principles
- Best practices

Key takeaways and helpful resources

Answer *your* questions

“We needed to build systems that embrace failure as a natural occurrence.”

Werner Vogels
CTO, Amazon.com

Definitions

Reliability

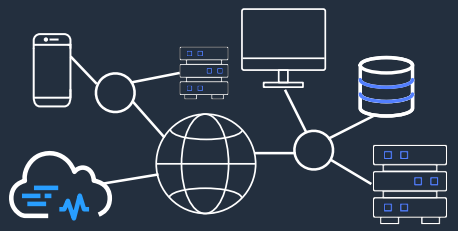
Ability of a workload to perform its required function correctly and consistently…

Resilience

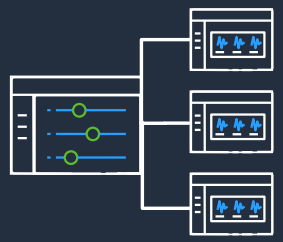
Ability of a workload to recover from infrastructure or service disruptions…

– Reliability Pillar, AWS Well-Architected Framework

High Reliability is Now the New Normal



Complex ecosystems



Incremental releases

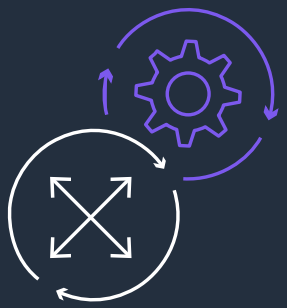
Expanding channels

Multi-disciplinary teams

Specialized services

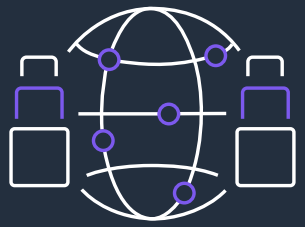
Always on

Six Pillars of AWS Well-Architected Framework



Operational excellence

Security



Reliability

Performance efficiency

Cost optimization

Sustainability

Well-Architected is a Set of Best Practices

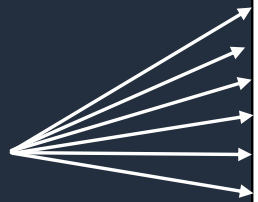
Design principles

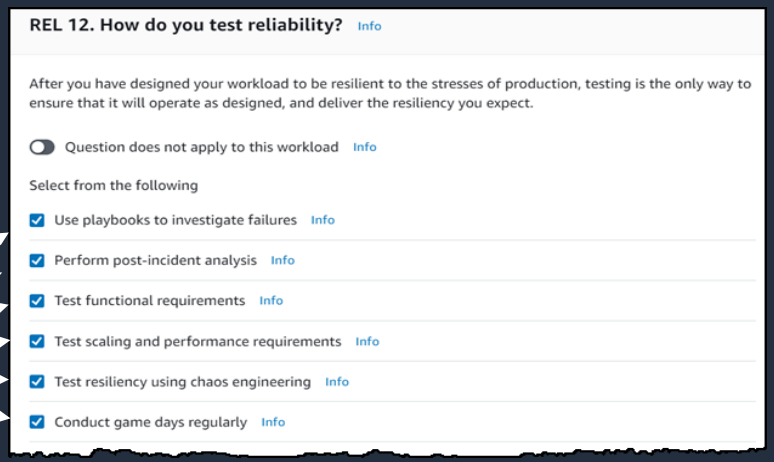
Questions

Best Practices

Whitepaper

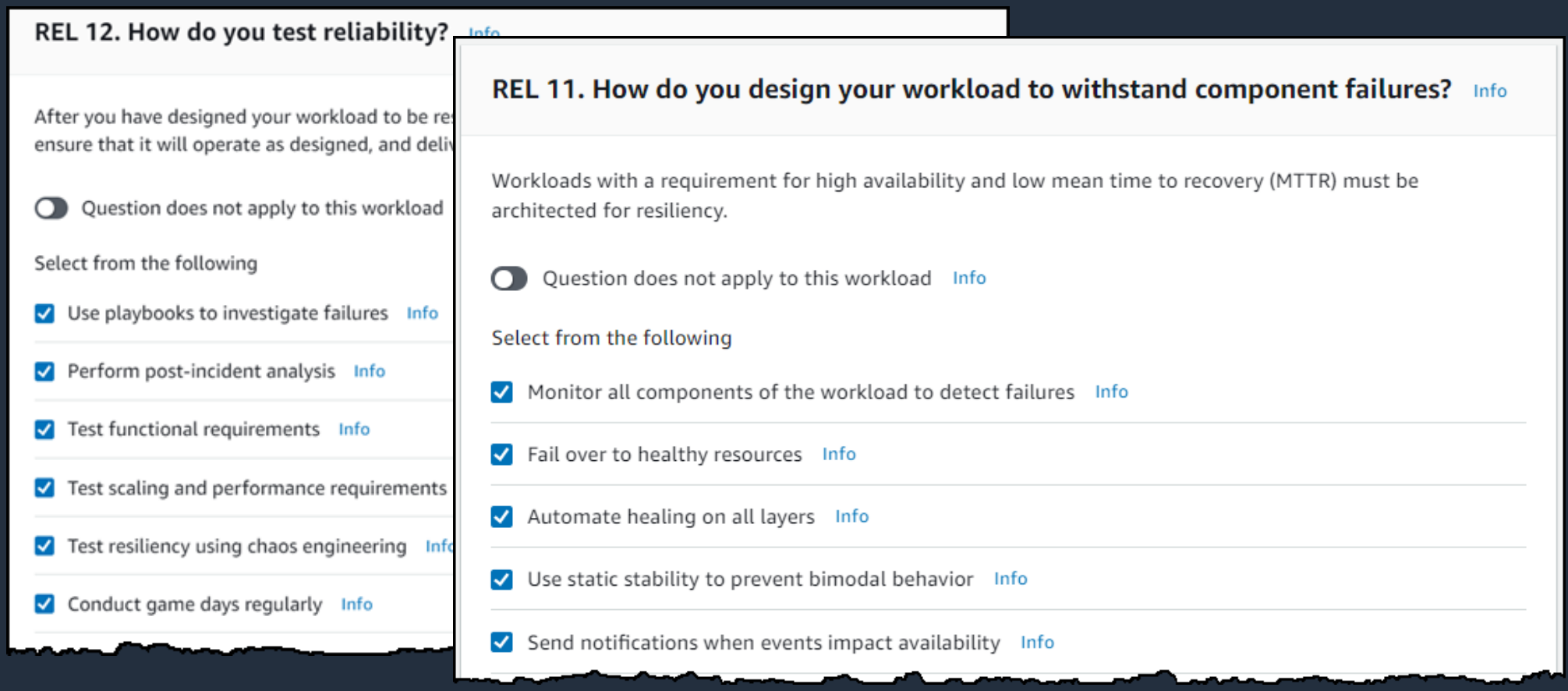
Labs





Well-Architected Tool

AWS Well-Architected Tool

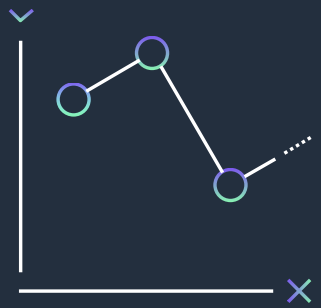


Best Practices: Well-Architected Reliability

Four Areas | 13 “Questions”

Foundations

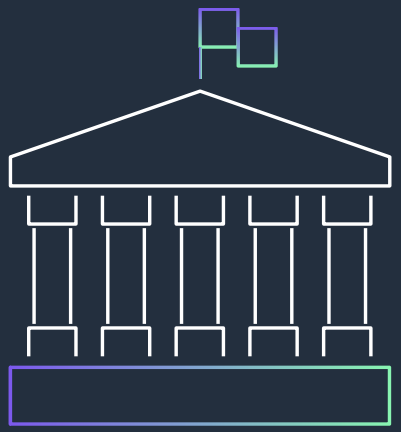
Workload architecture



Change management

Failure management

Well-Architected Reliability Pillar



Foundations

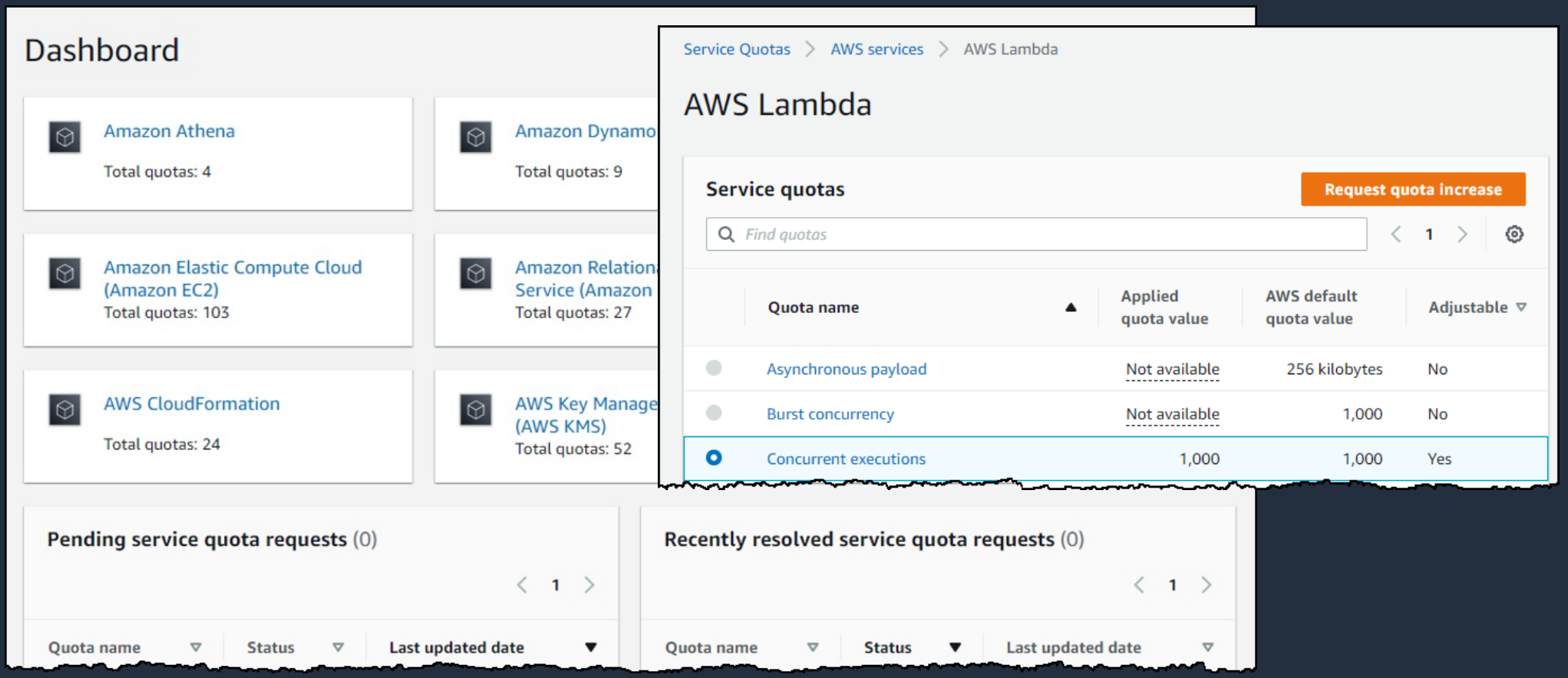
REL 1 **Service quotas and limits**

- Know your AWS resource quotas
- Raise quotas where necessary
- Work within hard limits

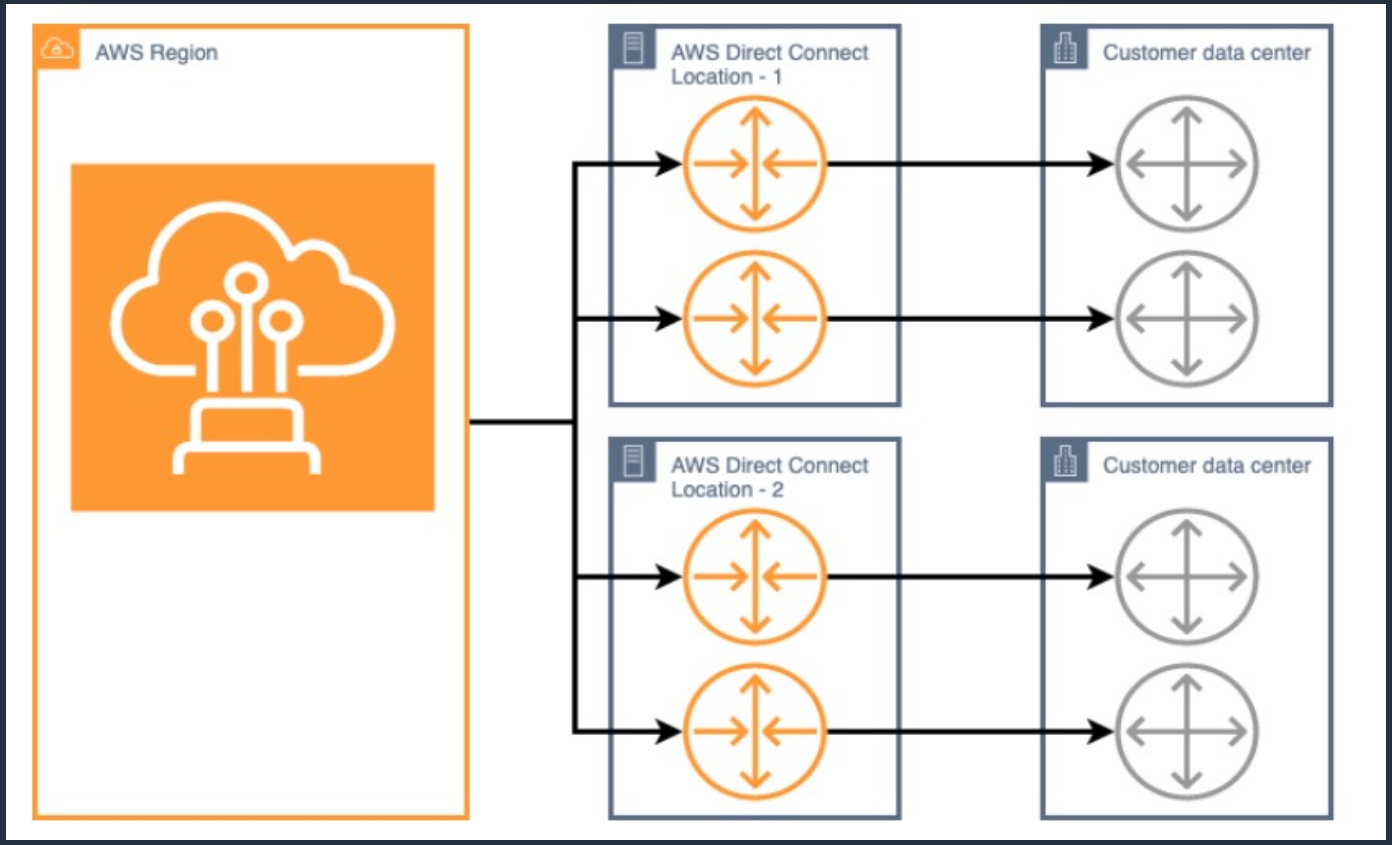
REL 2 **Network topology**

- Highly available endpoints
- Redundant connections
- Size and layout of network

Manage Service Quotas



Provision Redundant Connectivity



Well-Architected Reliability Pillar

Workload architecture

REL 3 **Service architecture**

- Business domains
- Right-sized, decoupled services

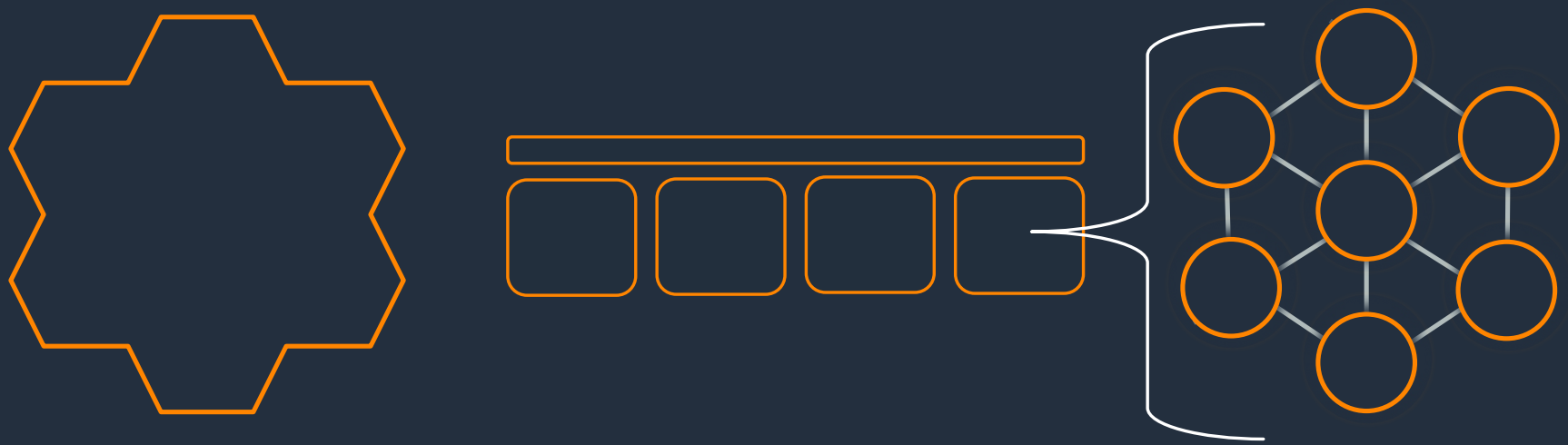
REL 4 **Prevent failures**

- Loosely coupled dependencies
- Idempotent, consistent

REL 5 **Mitigate failures**

- Graceful degradation
- Throttle, fail fast, limit retries

Choose How to Segment Your Workload



Monolithic application

- Does everything
- Shared release pipeline
- Rigid scaling
- High impact of change
- Hard to adopt new technologies

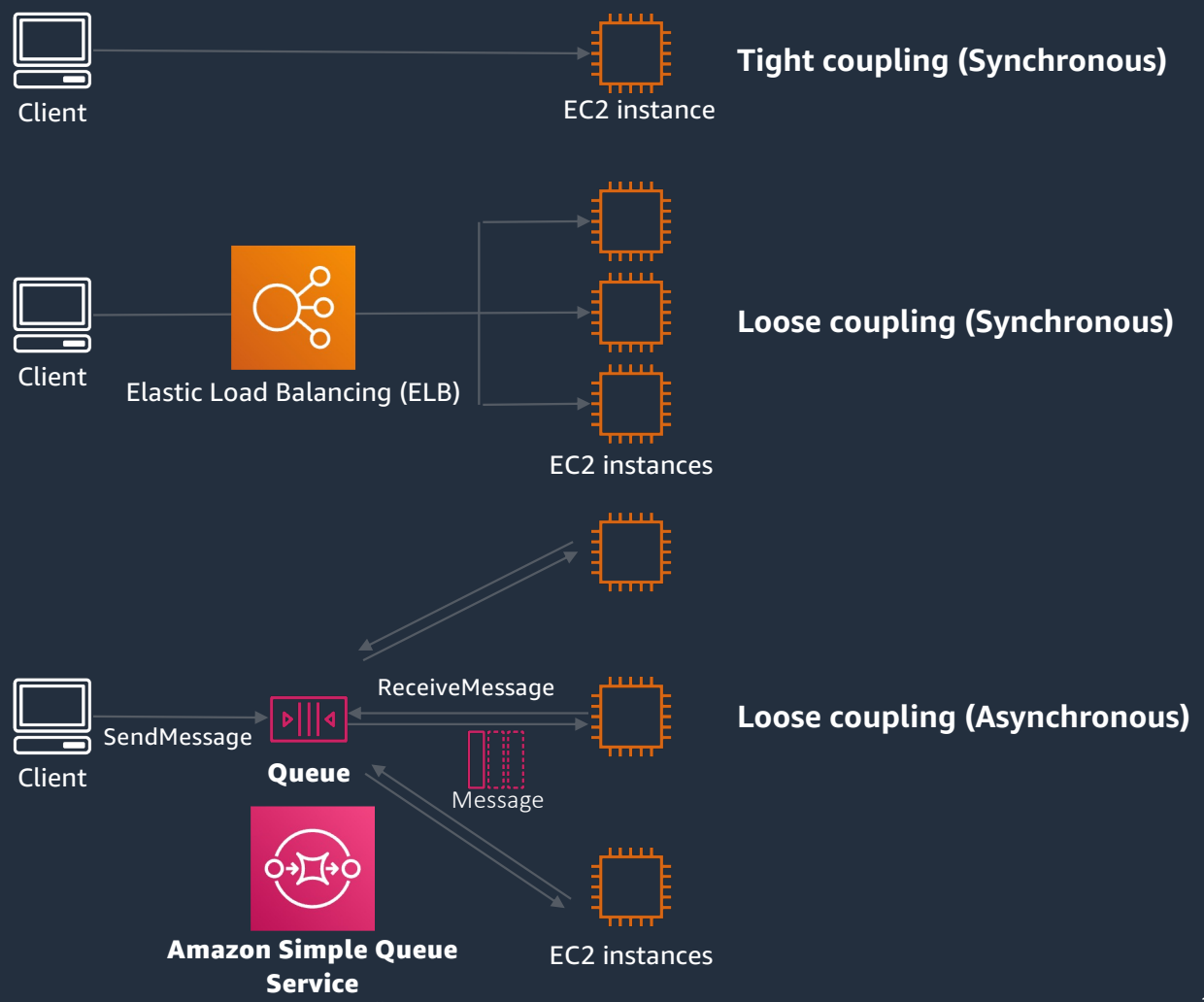
Service oriented

- Does some things
- Services surfaced via comms protocol
- Some coupled services

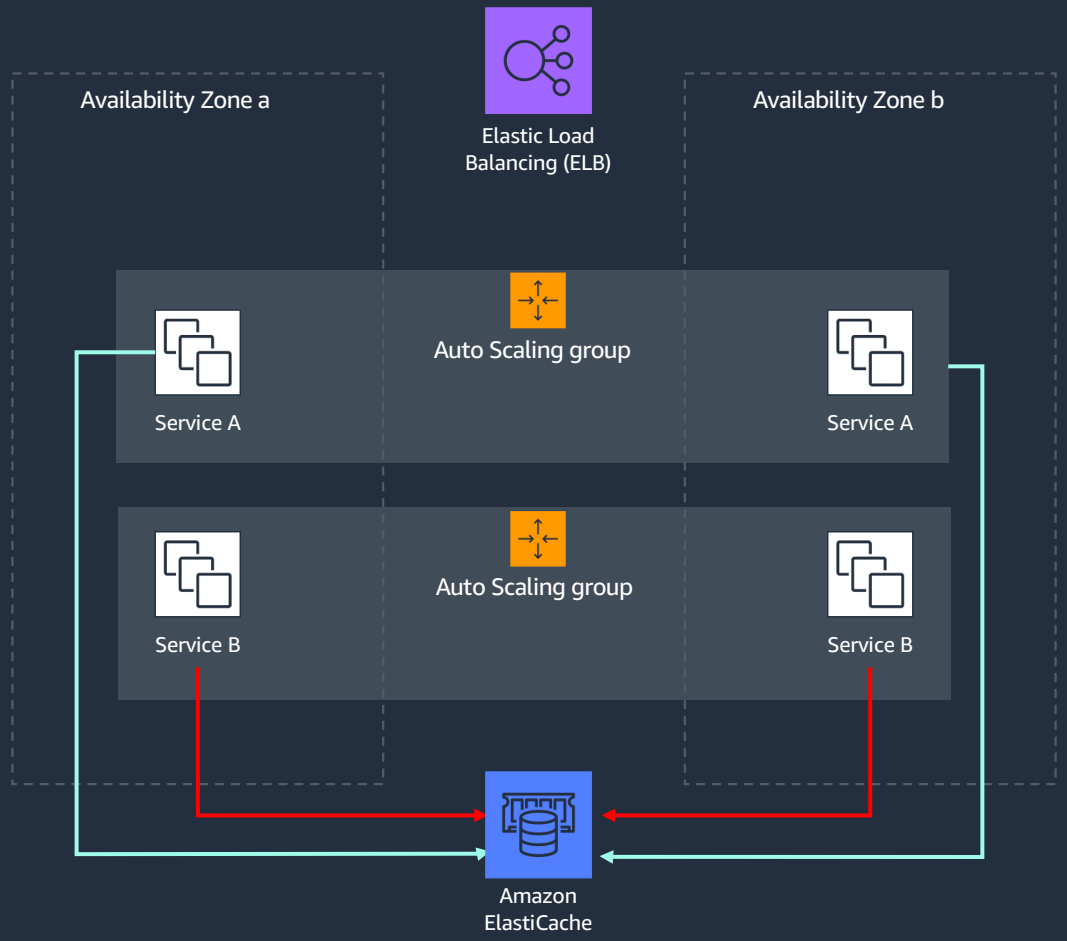
Microservices

- Does one thing
- Independent deployments
- Independent scaling
- Small impact of change
- Choice of technology

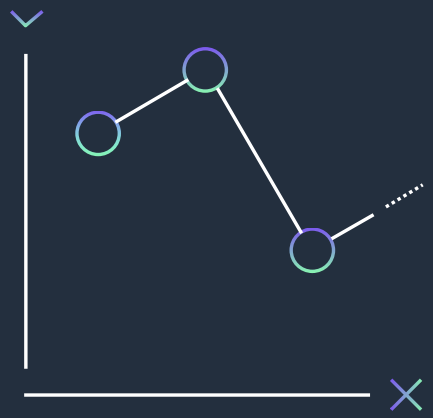
Implement Loosely Coupled Dependencies



Make Services Stateless Where Possible



Well-Architected Reliability Pillar



Change management

REL 6 **Monitor workload resources**

- You can’t manage what you can’t see
- Implement observability

REL 7 **Adapt to changes in demand**

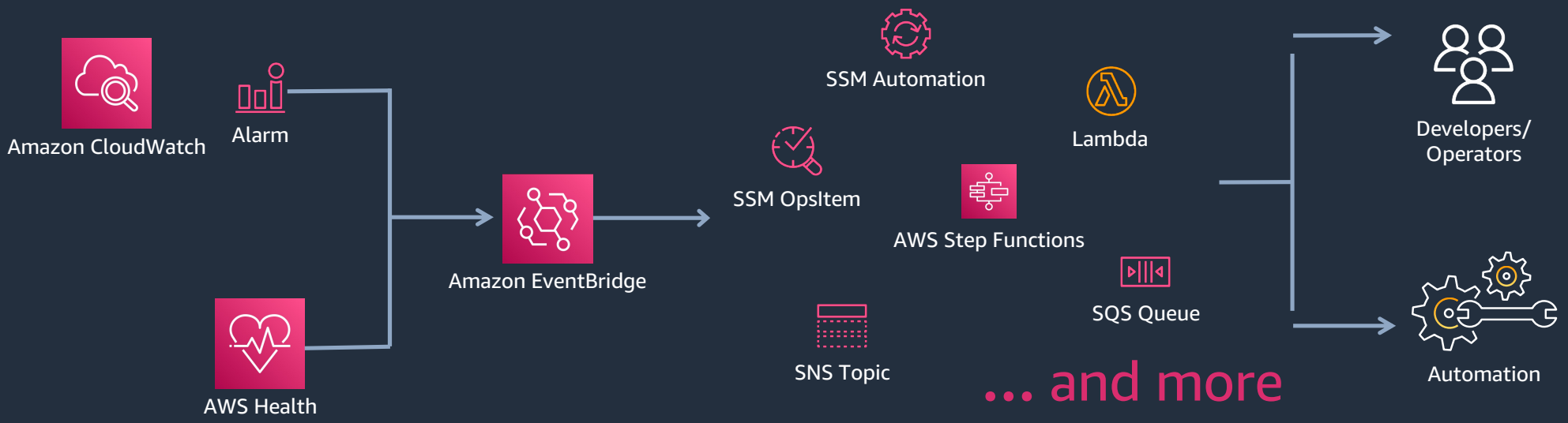
- Scale up and down as needed
- Use automation

REL 8 **Implement change**

- Deployments can be dangerous
- Manage and reduce negative impacts

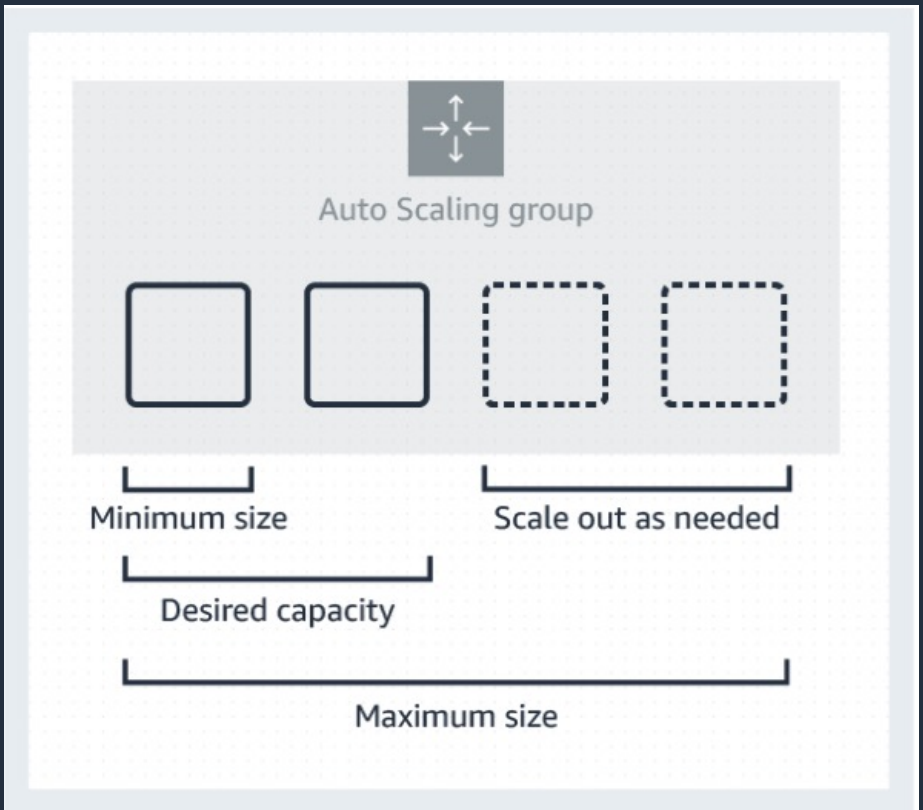
Real-time Processing and Alarming

- Send notifications
- Automate responses




```
{
  "source": ["aws.health"],
  "detail-type": ["AWS Health Event"],
  "detail": {
    "service": ["S3"],
    "eventTypeCategory": ["issue"],
    "eventTypeCode": ["AWS_S3_INCREASED_GET_API_ERROR_RATES", "AWS_S3_INCREASED_PUT_API_ERROR_RATES"]
  }
}
```

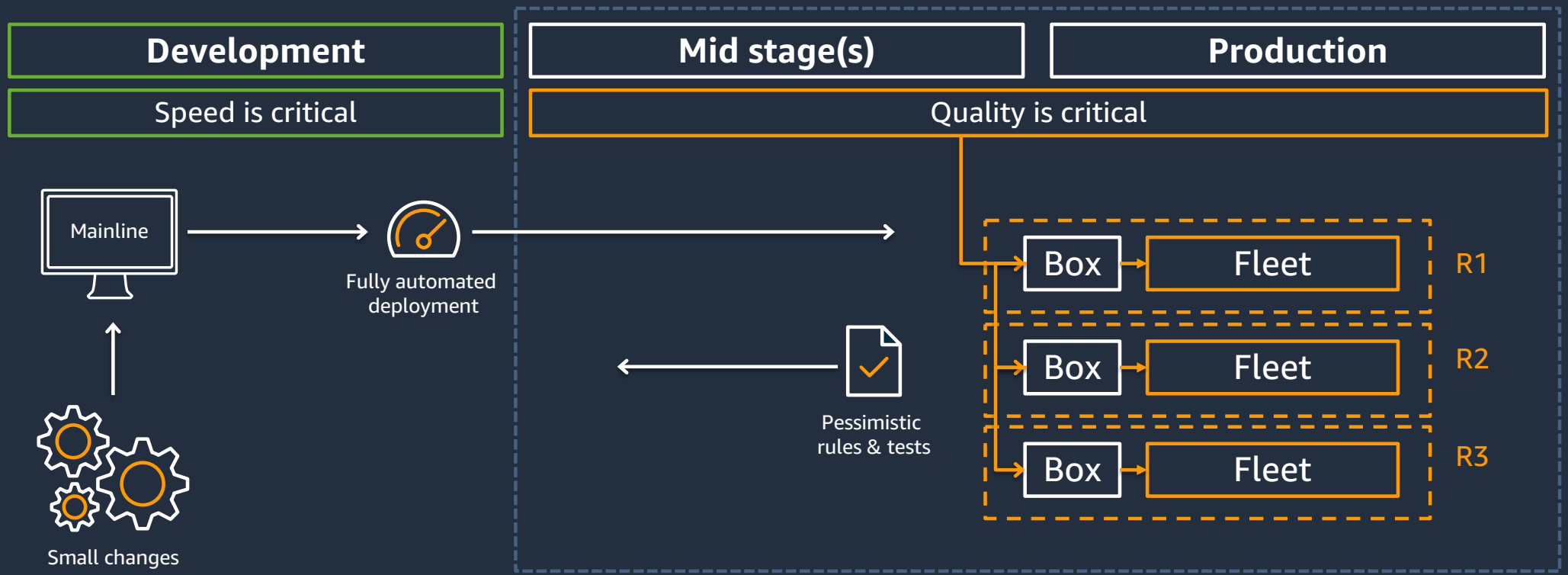
Use Automation When Obtaining or Scaling Resources



- EC2 instances
- Aurora read-replicas
- Aurora storage
- DynamoDB throughput
- ECS tasks
- Lambda concurrent instances

Deploy Changes with Automation

Also…Integrate testing as part of your deployment



Well-Architected Reliability Pillar

Failure management

REL 9 Back up data

- Identify data to back up
- Automate and secure back up

REL 10 Fault isolation

- Use multiple sites
- AWS Region, Availability Zone, etc

REL 11 Withstand component failures

- Automate recovery
- Notify when failures occur

Deploy the Workload to Multiple Locations

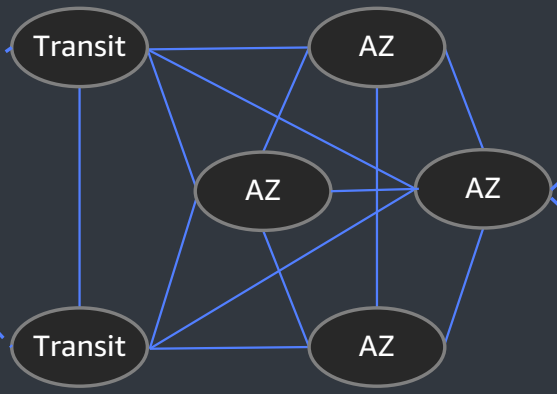
AWS REGIONS AND AVAILABILITY ZONES (AZS)

31 AWS Regions worldwide

AWS Regions

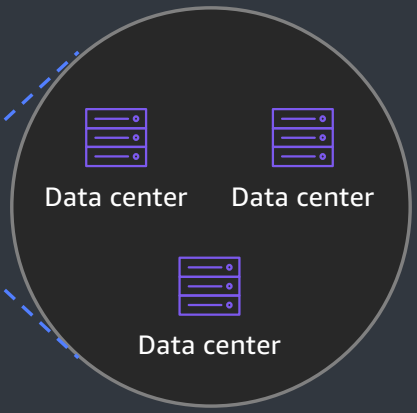
Announced Regions

Each AWS Region has multiple AZs



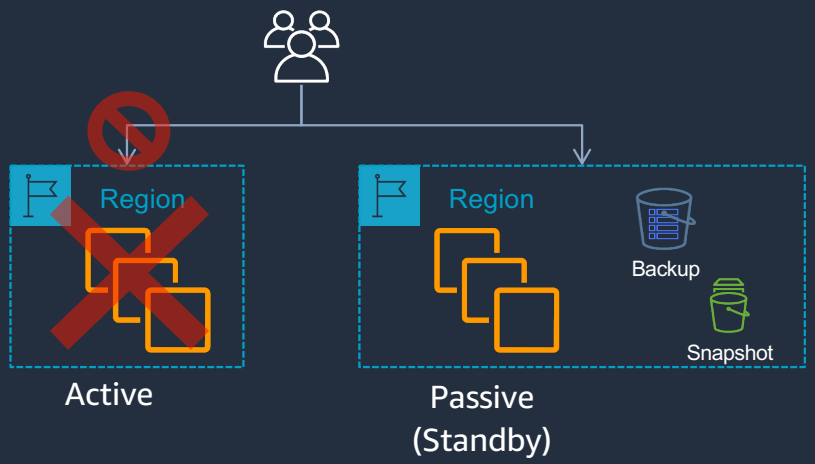
A Region is a physical location in the world

Each AZ includes one or more discrete data centers



Data centers, each with redundant power, networking, and connectivity, housed in separate facilities

Rely on the Data Plane and Not the Control Plane During Recovery



If control plane fails….

Amazon Route 53

What will work	What may not work
DNS resolution and health checks	Updates to routing policies

Route 53 Application recovery controller

Changes to routing controls	CRUDL routing controls

Well-Architected Reliability Pillar

Failure management

REL 12 Test reliability

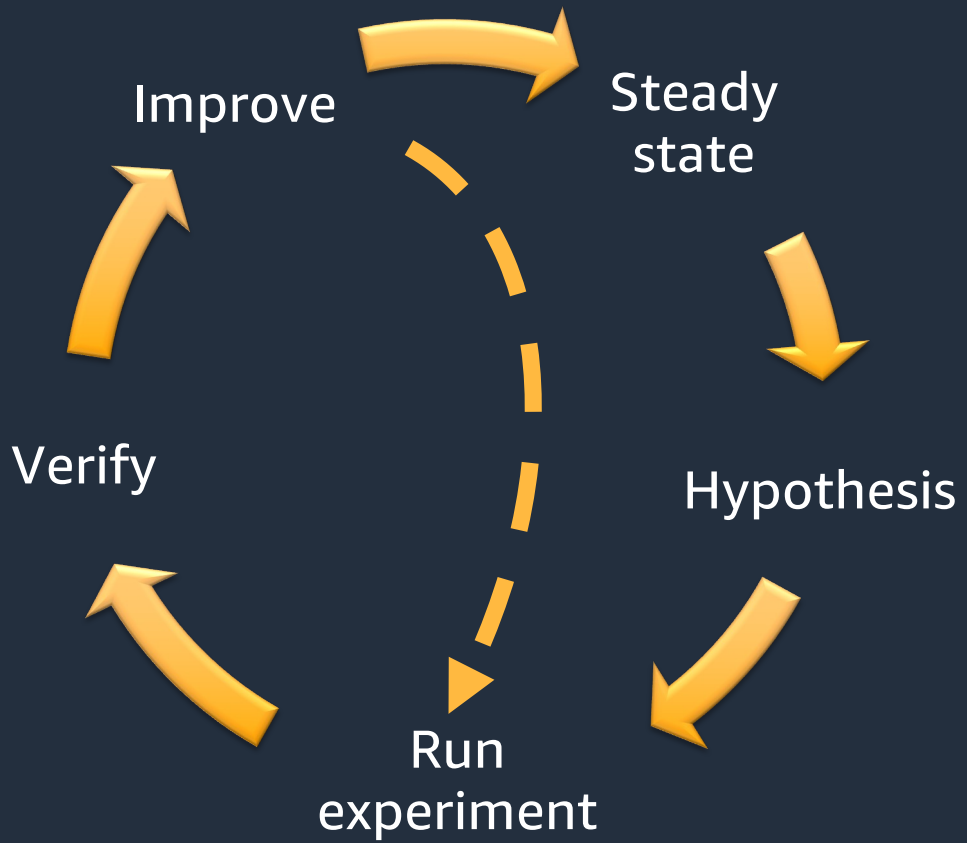
- Simulate failure modes
- Test scaling, performance, processes

REL 13 Disaster recovery

- Set objectives
- Implement and test DR strategy

Test Resilience Using Chaos Engineering

A SCIENTIFIC METHOD



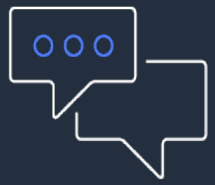
Conduct Game Days Regularly

SIMULATE FAILURE OR EVENT TO TEST SYSTEMS RESILIENCE, PROCESSES, AND TEAM RESPONSES

People

Cross-discipline team

Processes



Briefing

Overview

Roles

Planning

Preparation

Hypothesis

Execution

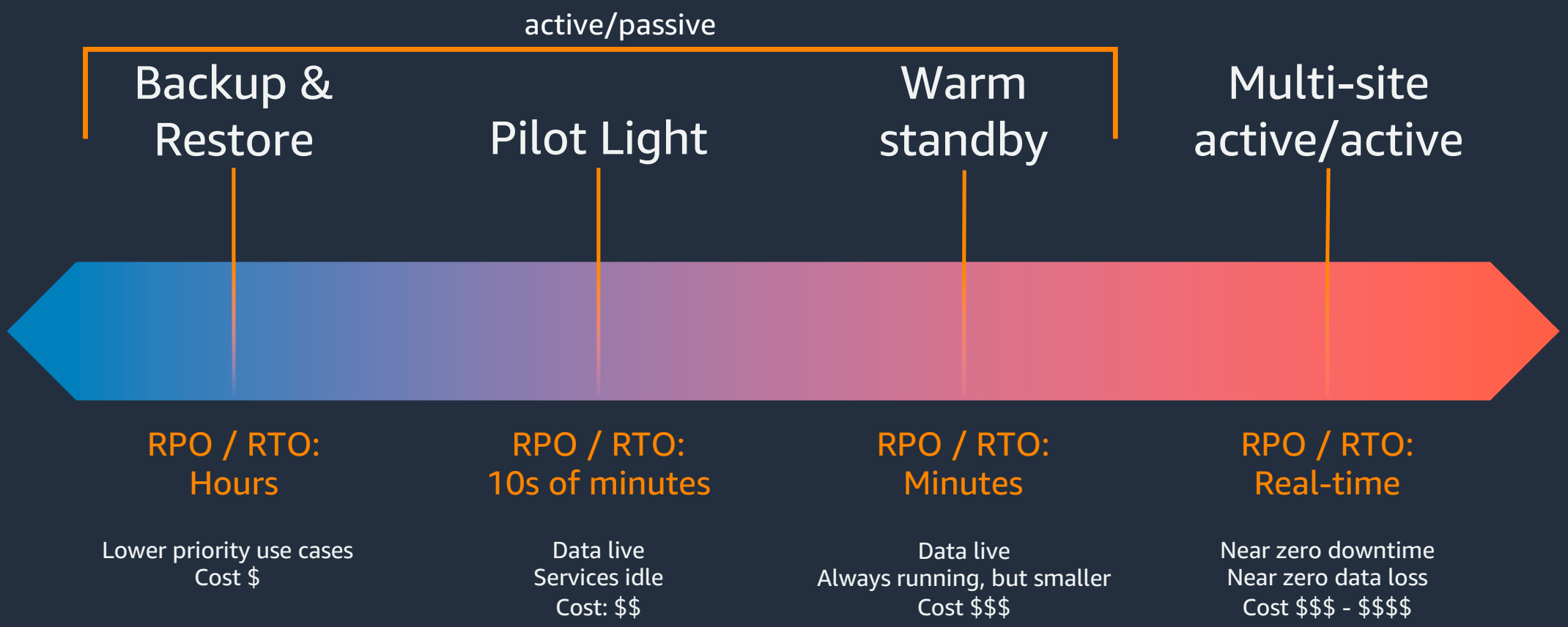
Run experiment

Analysis

Verify

Improve

Strategies for Disaster Recovery



Key Takeaways

- Reliability is a continuous improvement process
- Your system is not reliable and resilient unless you test that
- Prepare for failure – evaluate risks
- Call to action: get started with the best practices mentioned!!!

Resources

AWS **Well-Architected**

Whitepaper: Reliability Pillar: AWS Well-Architected Framework
bit.ly/reliability-pillar

Well-Architected hands-on labs
wellarchitectedlabs.com/reliability

Well-Architected Tool
aws.com/well-architected-tool/

AWS Resilience Hub
aws.amazon.com/resilience-hub/
go.aws/3FF5UA0 (Blog post)

Q & A

Danny Wright
Principal Technical Account Manager
AWS Enterprise Support
dwriamz@amazon.com

Please complete the session survey.

Well-Architected Reliability: Best Practices You Can Use Today