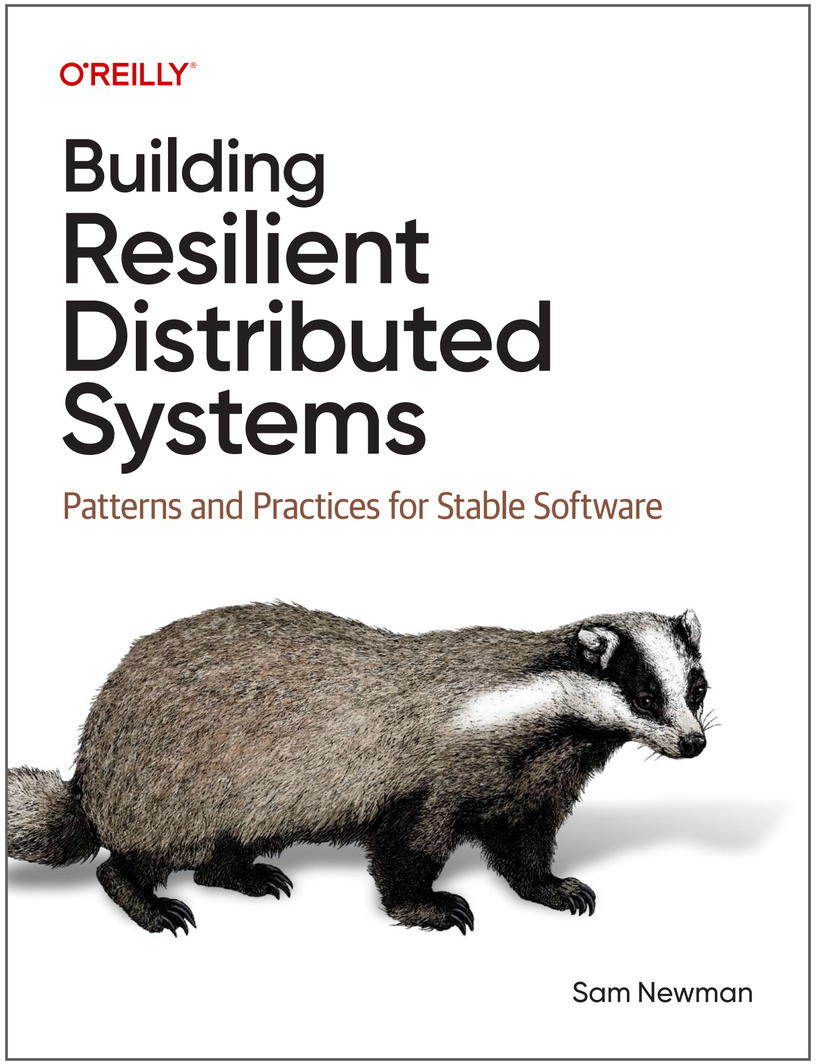
PROGRESSIVE COLLAPSE

Sam Newman

https://flickr.com/photos/testlab/21496317363/

NEW BOOK OUT 2026



https://samnewman.io/

https://www.theguardian.com/society/from-the-archive-blog/gallery/2018/may/16/ronan-point-tower-collapse-may-1968

https://www.theguardian.com/society/from-the-archive-blog/gallery/2018/may/16/ronan-point-tower-collapse-may-1968

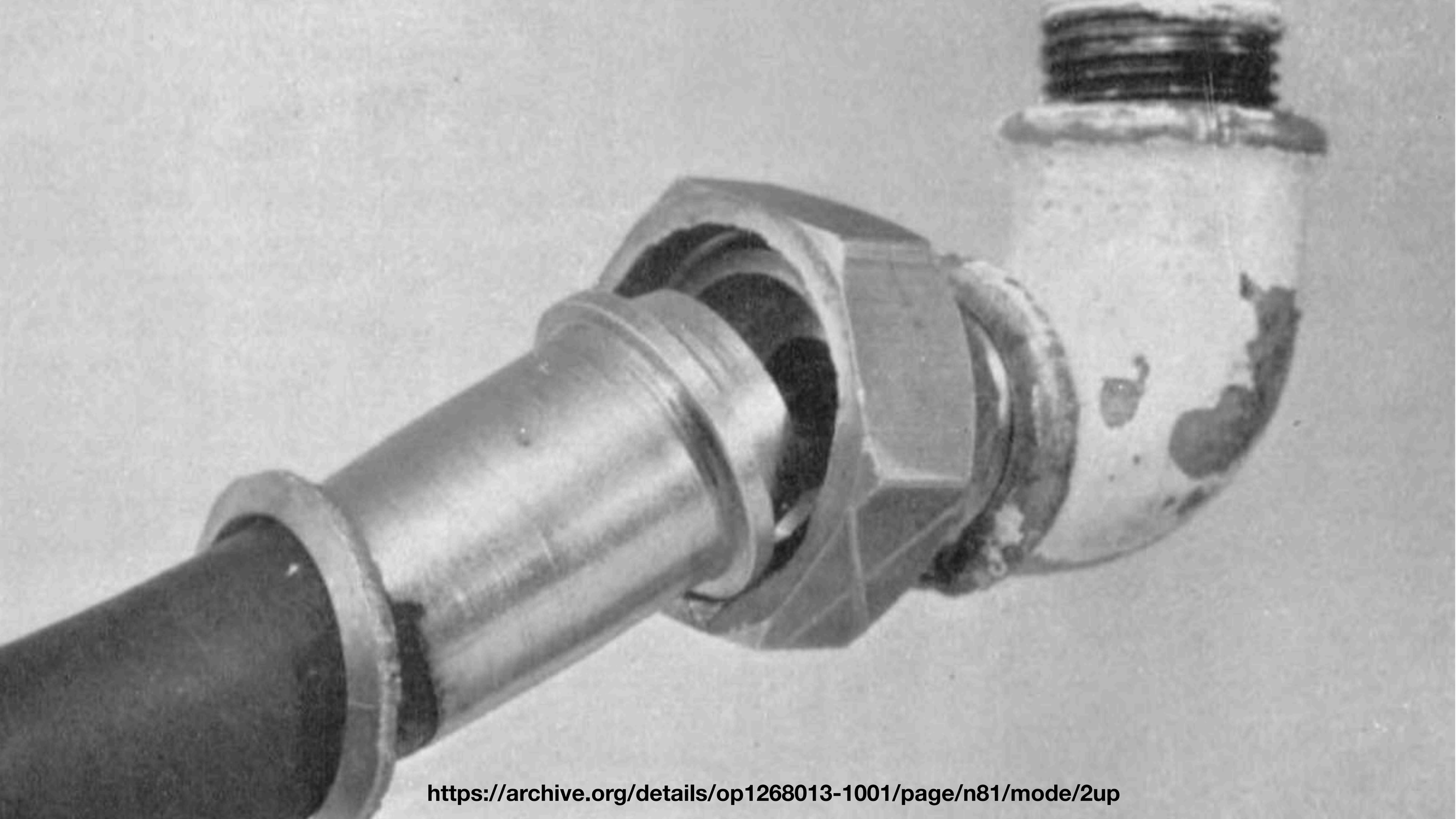
Progressive Collapse

Progressive collapse in digital systems

Techniques to mitigate a progressive collapse

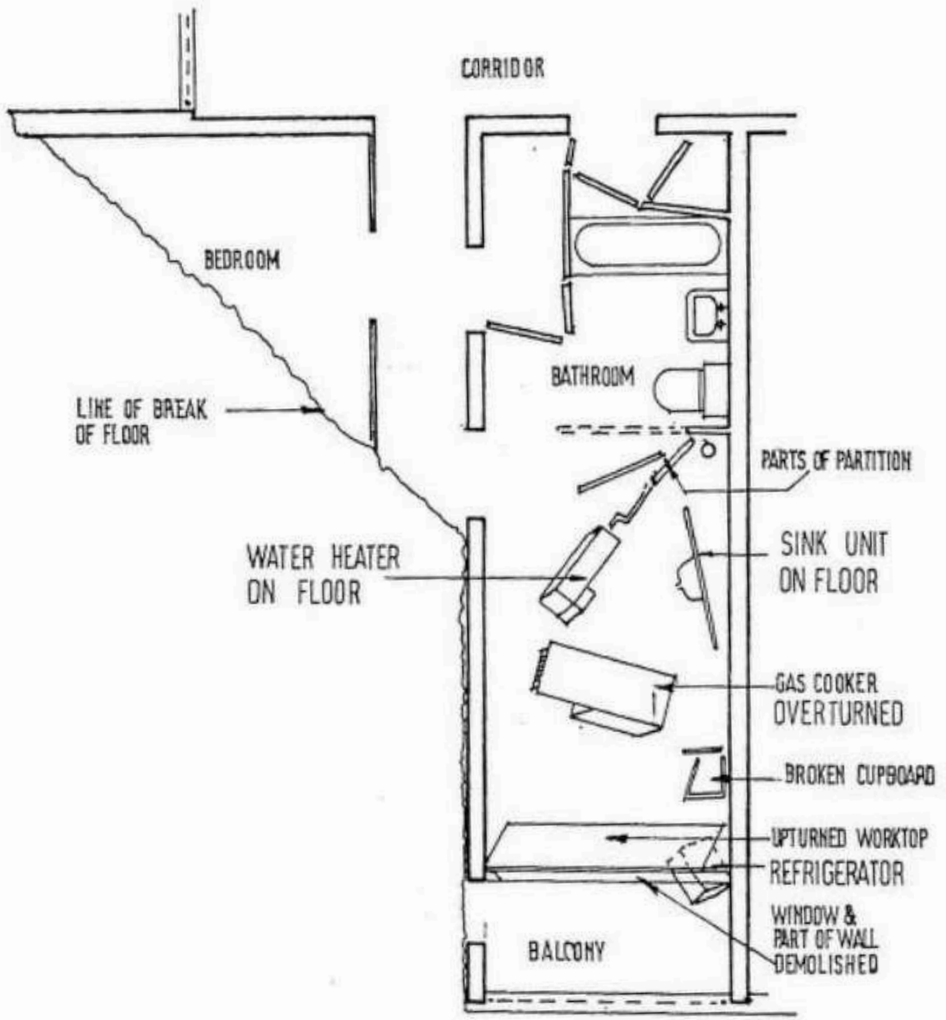
What happened?

https://www.flickr.com/photos/comedynose/54043067246/



https://archive.org/details/op1268013-1001/page/n81/mode/2up

Plate No 7 *Miss Hodge's gas cooker showing the flexible hose*



Plan (c) *The extent of the damage to Flat 90*

https://archive.org/details/op1268013-1001/page/n79/mode/2up

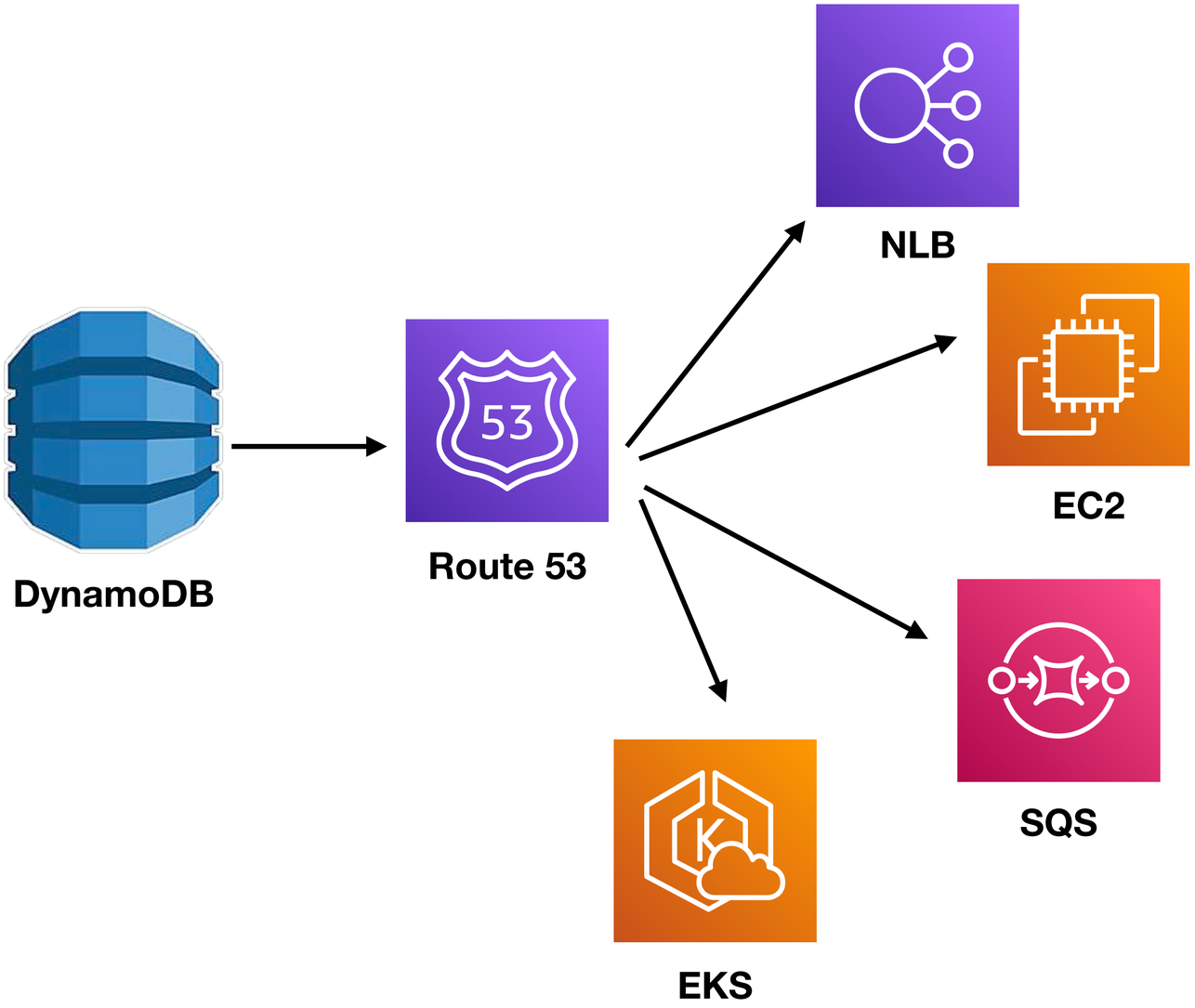

Ivy's Flat

Progressive Collapse

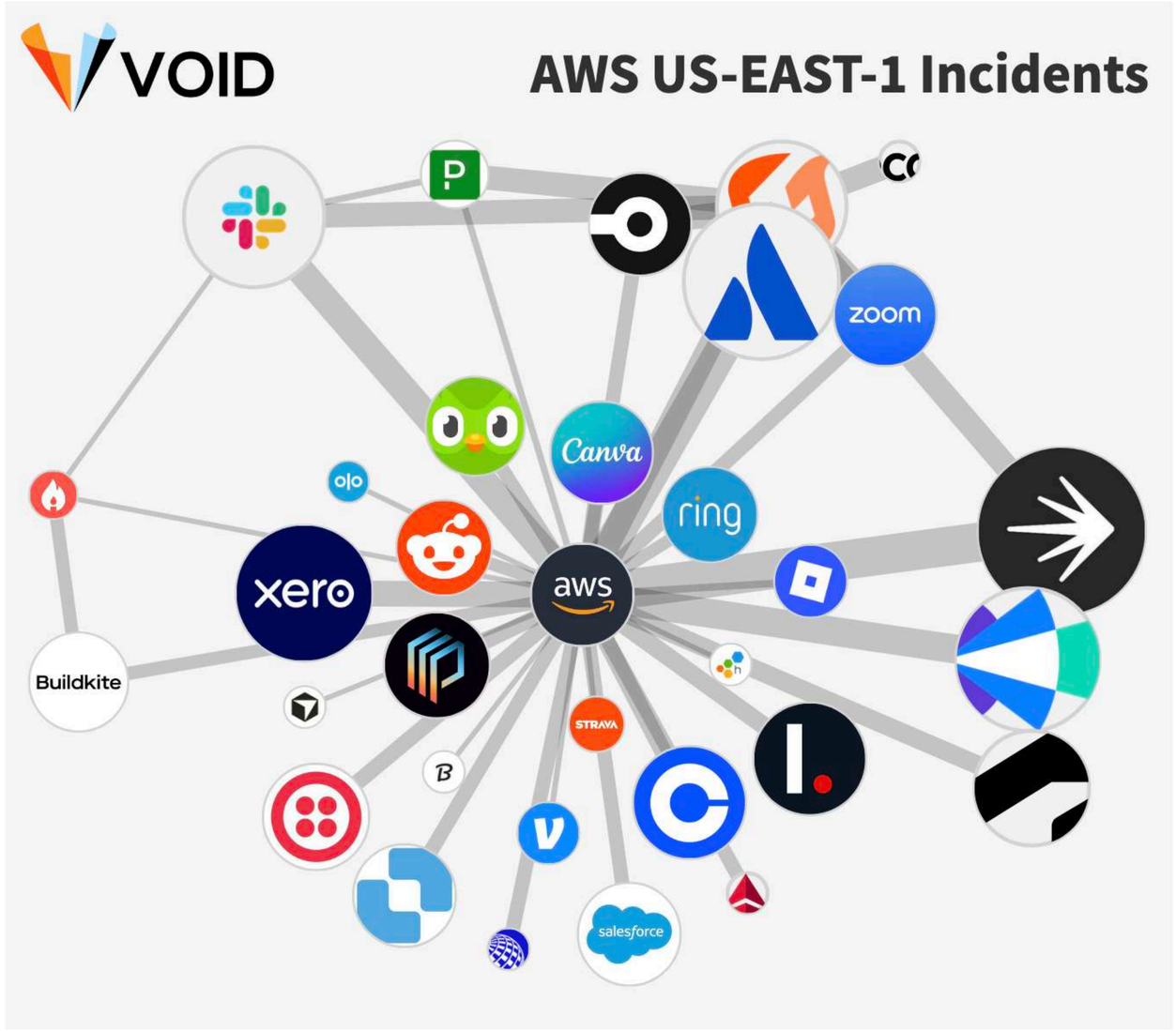
A small failure results in a significant collapse in the wider system

https://youtu.be/y97rBdSYbkg?si=ZiwBurQaHEhFf-Di

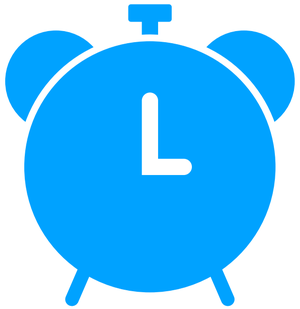
US-EAST-1 Outage, October 2025



And many more…

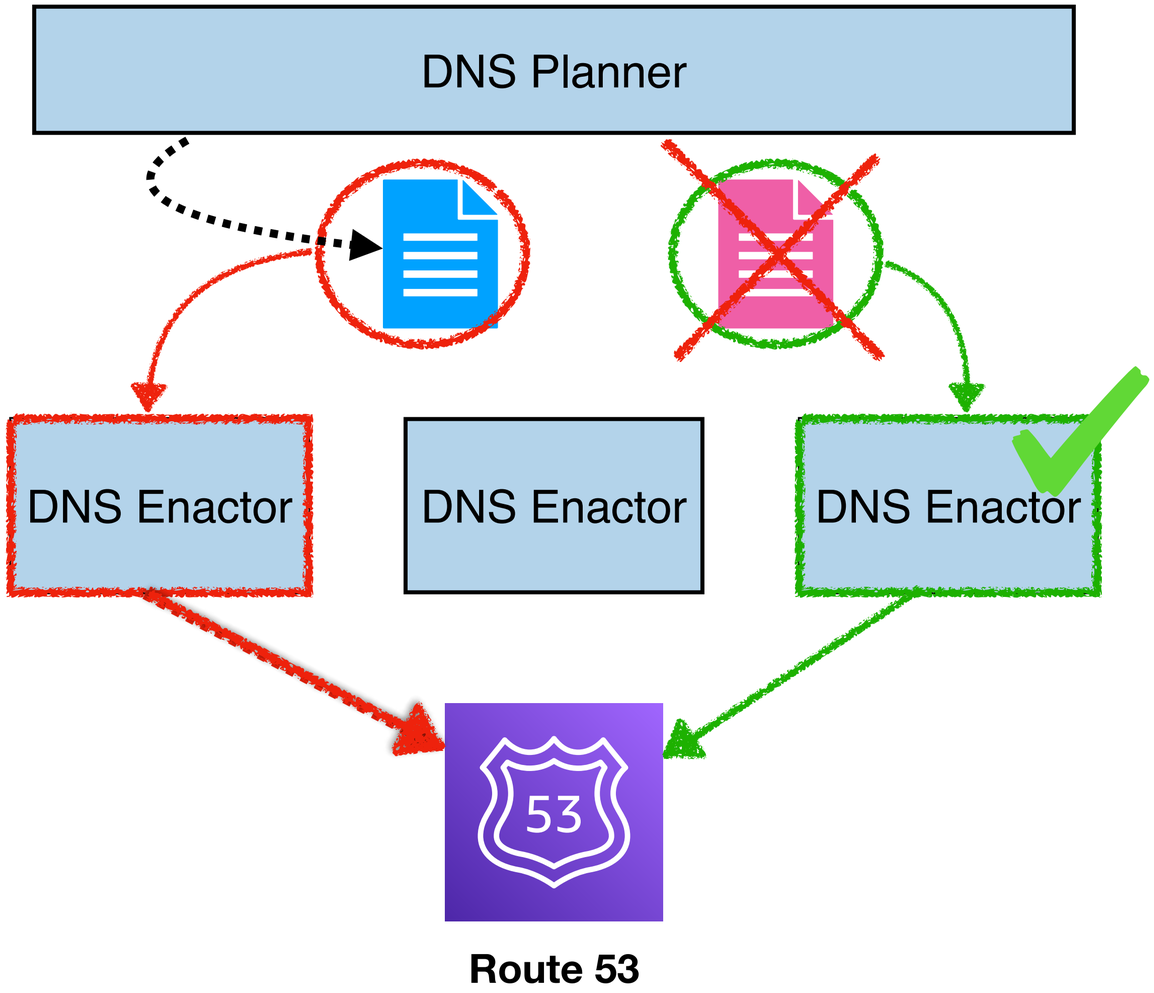


https://www.thevoid.community/aws-2025-outage-graph



Took much longer than normal

New plans started arriving



A second enactor picked up a newer plan

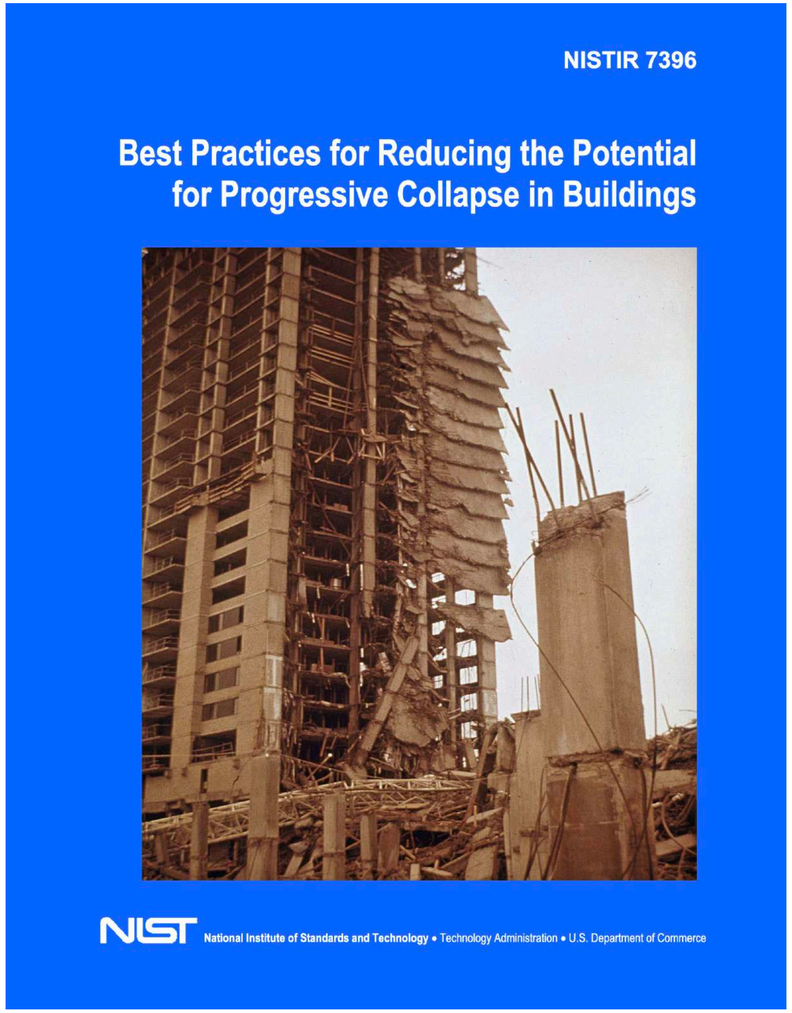
And completed quickly

The original enactor completed, then overwrote the new plans

https://aws.amazon.com/message/101925/

How to mitigate a progressive collapse?

NIST 7396



https://nvlpubs.nist.gov/nistpubs/Legacy/IR/nistir7396.pdf

1. Reduce Hazards

2. Strengthen Components

3. Reduce Interconnection

1. Reduce Hazards

Stop allowing gas in high-rise buildings

Improve ventilation

Disabled the automated DNS management

Improved testing

2. Strengthen Components


https://flickr.com/photos/58827557@N06/50604349197/

Re-examine the use of the large panel system

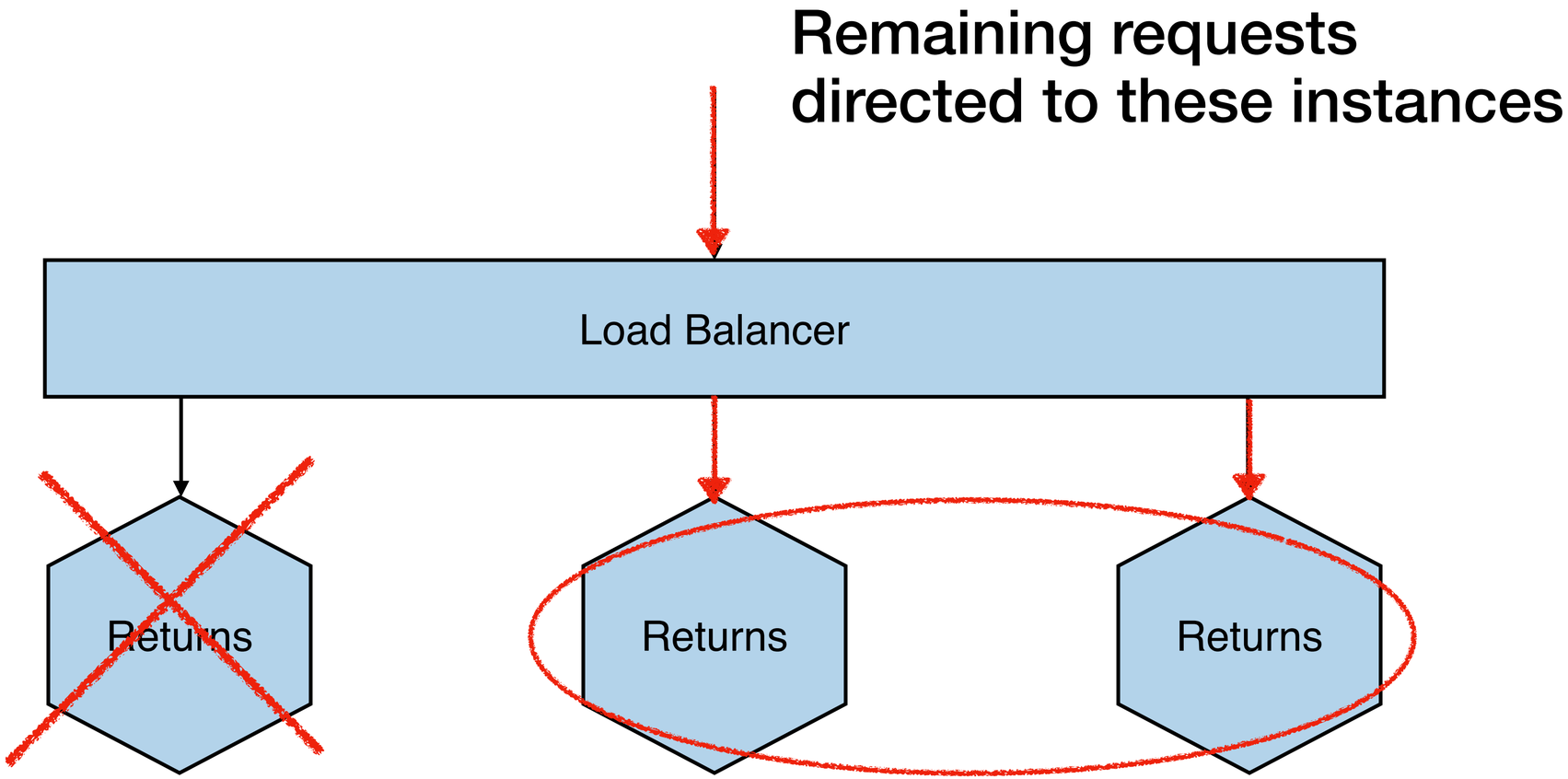
Improved quality control & building regulations

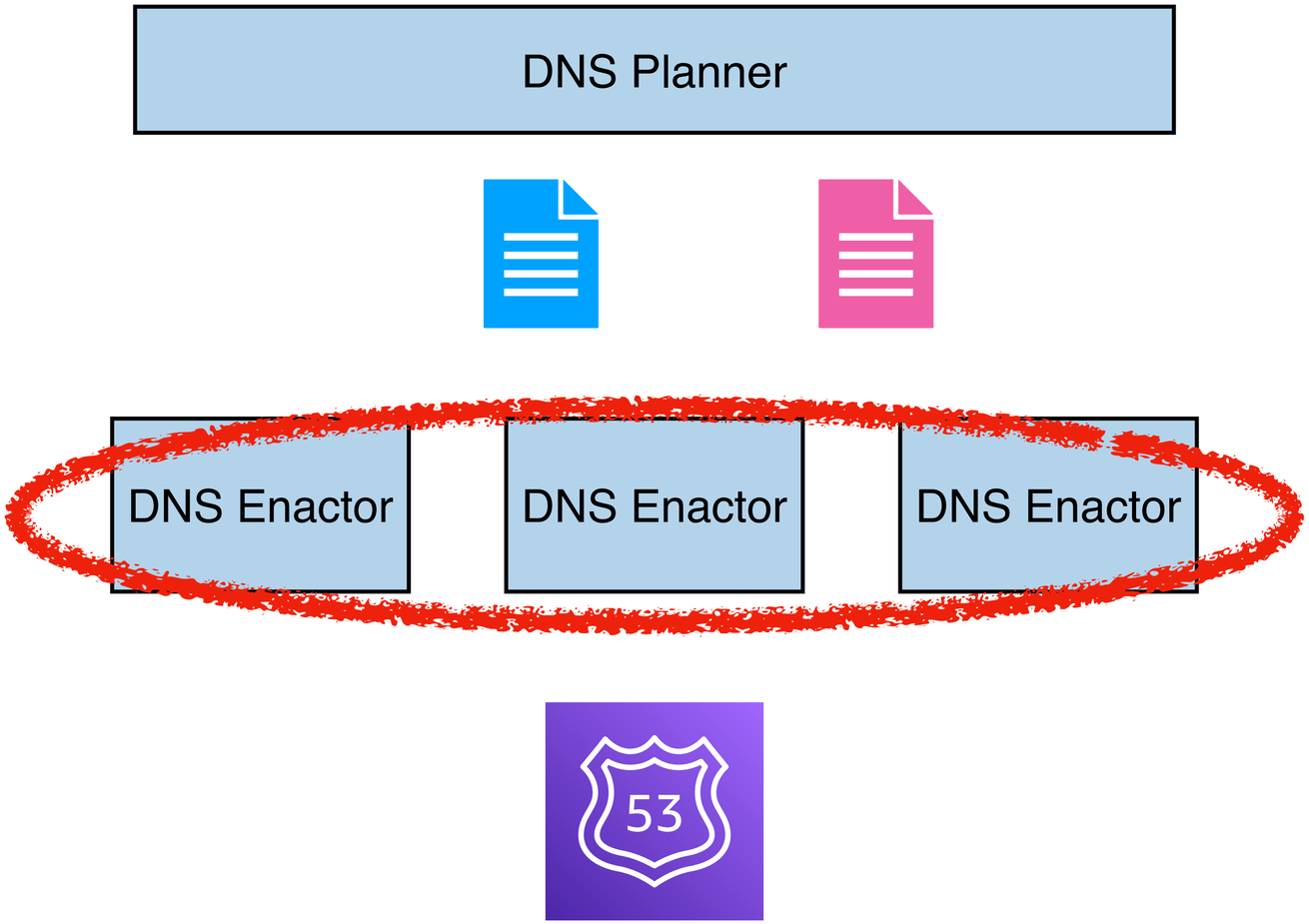
How do we strengthen a service?

Redundancy

https://flickr.com/photos/jorel314/4182427316/

REDUNDANT SERVICE INSTANCES

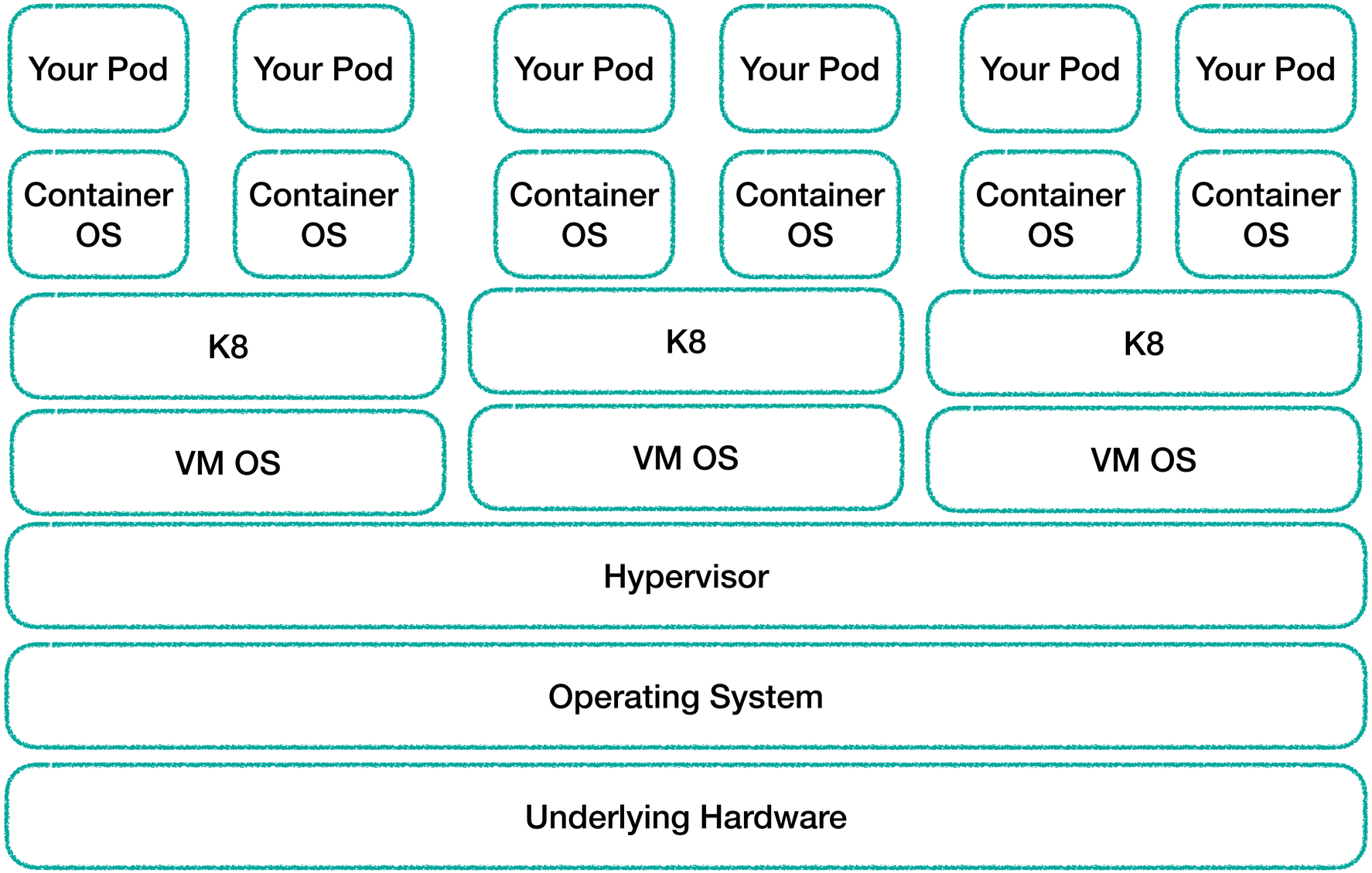




Route 53

https://aws.amazon.com/message/101925/

Increasing a system's complexity to handle more failure scenarios can introduce more sources of failure


Your Pod
Your Pod
Your Pod
Your Pod
Your Pod
Your Pod
Container OS
Container OS
Container OS
Container OS
Container OS
Container OS
K8
K8
K8
VM OS
VM OS
VM OS
Hypervisor
Operating System
Underlying Hardware

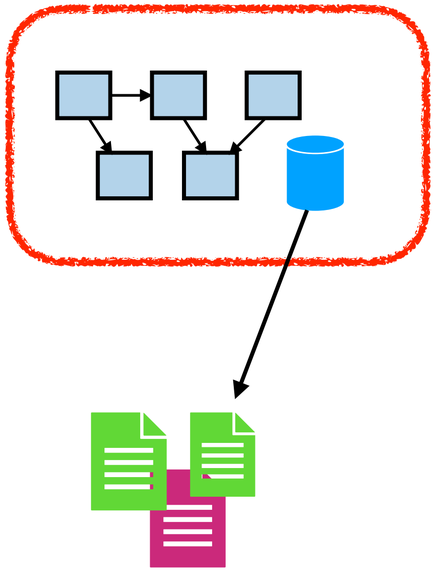
Work to fix the race condition

For some users - considering multi-region

https://docs.aws.amazon.com/whitepapers/latest/aws-fault-isolation-boundaries/regions.html

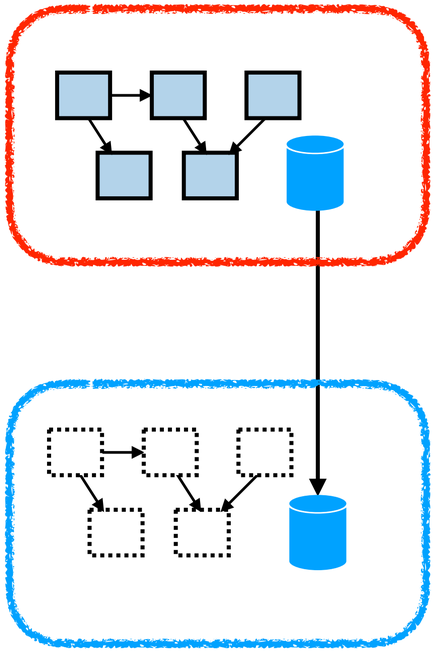
MULTI-SITE OPTIONS

Backup & Restore



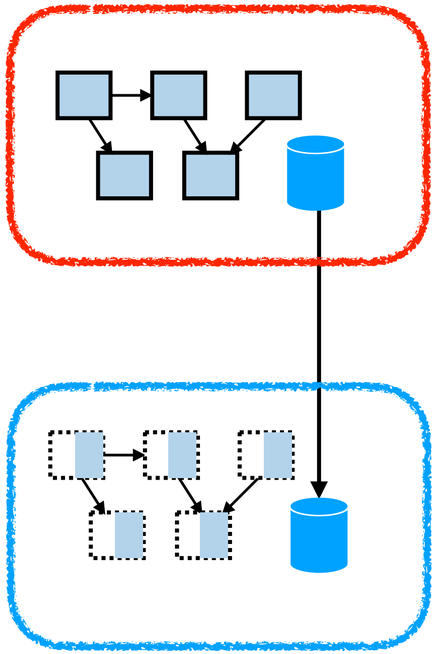
Hours or longer to recover

Pilot Light



Minutes to hours to recover

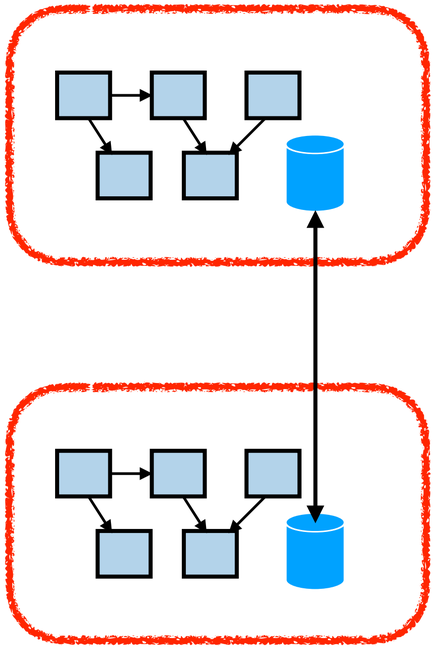
Warm Failover



Some load handled almost immediately

More continual “testing” possible

Multi-site

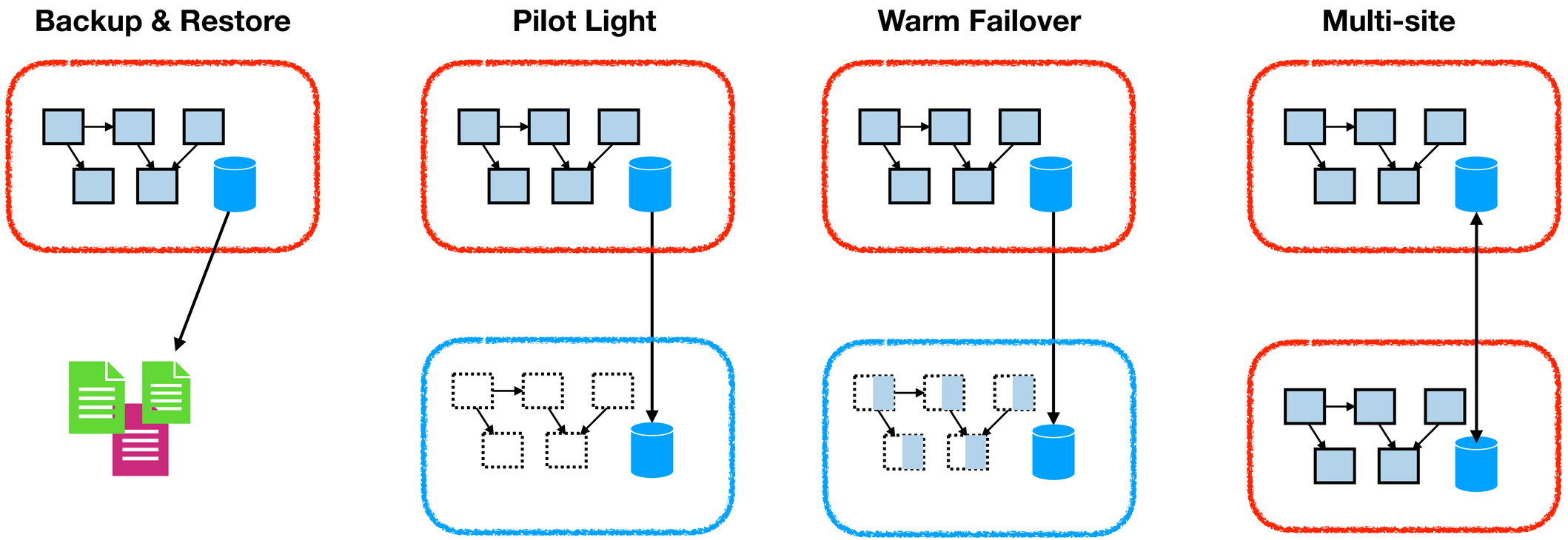


Active-active

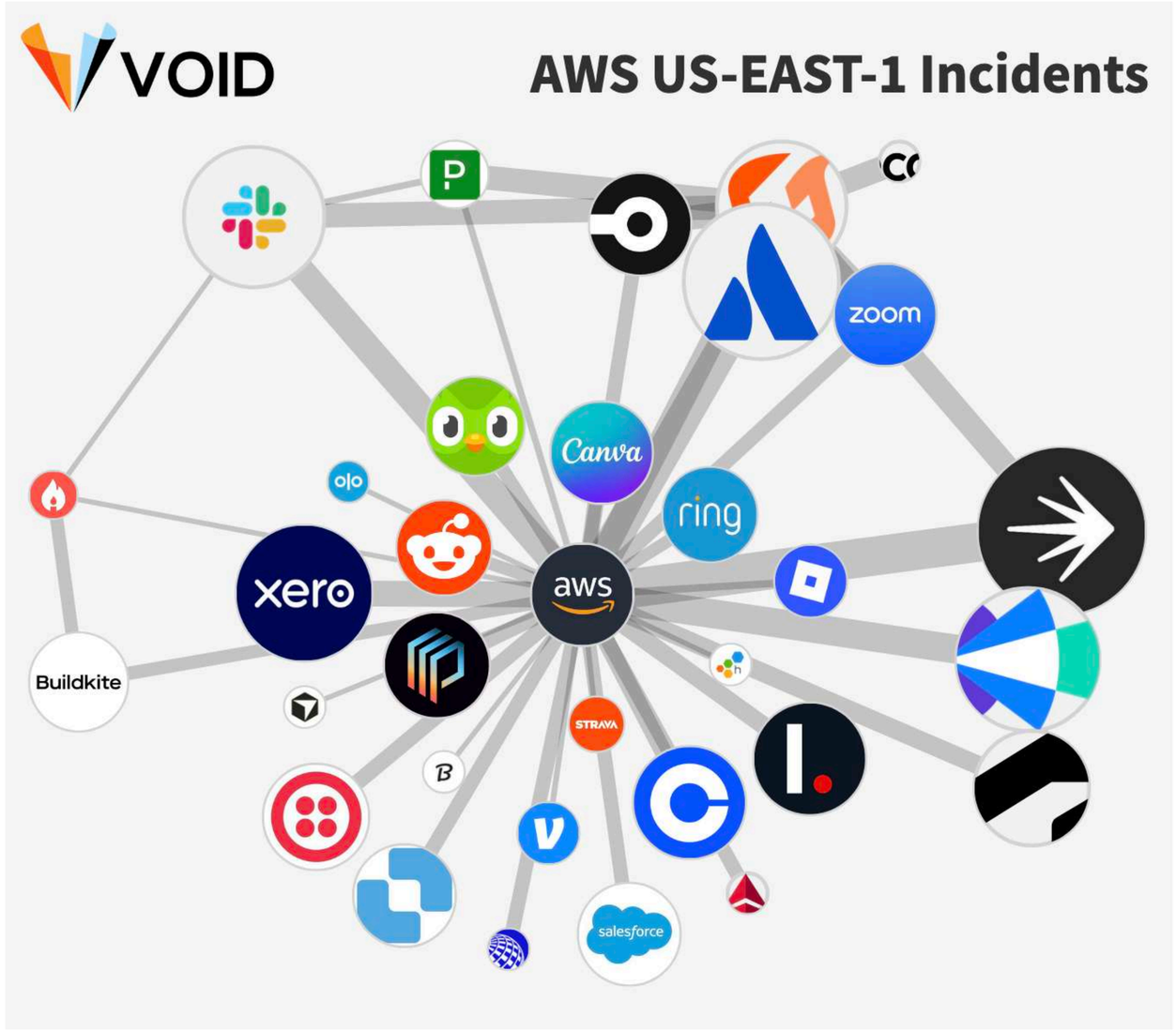
Complexity in terms of two-way synchronisation of data

Anyone telling you that two way data synchronisation is “simple” or “easy” is trying to sell you something

MULTI-REGION OPTIONS




THIS NOTE IS LEGAL TENDER
FOR ALL DEBTS, PUBLIC AND PRIVATE
Treasurer of the United States.
100
FEDERAL RESERVE
HC 38553744 C3
https://flickr.com/photos/pictures-of-money/17123251389/



https://www.thevoid.community/aws-2025-outage-graph

AWS Architecture Blog

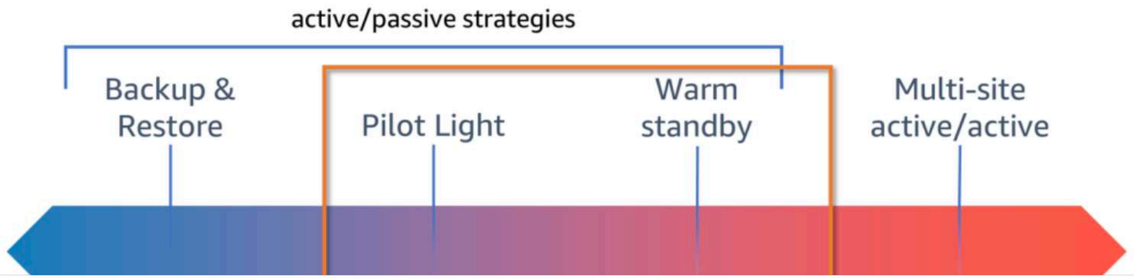
Disaster Recovery (DR) Architecture on AWS, Part III: Pilot Light and Warm Standby

by Seth Eliot | on 14 MAY 2021 | in Amazon CloudWatch, Amazon DynamoDB, Amazon RDS, Amazon Route 53, Amazon VPC, AWS CLI, AWS Global Accelerator, AWS Lambda, AWS Management Console, Elastic Load Balancing | Permalink | Share

In this blog post, you will learn about two more active/passive strategies that enable your workload to recover from disaster events such as natural disasters, technical failures, or human actions. Previously, I introduced you to four strategies for disaster recovery (DR) on AWS. Then we explored the backup and restore strategy. Now let's learn about the pilot light and warm standby strategies.

DR strategies: Pilot light or warm standby

When selecting your DR strategy, you must weigh the benefits of lower RTO (recovery time objective) and RPO (recovery point objective) vs the costs of implementing and operating a strategy. The pilot light and warm standby strategies both offer a good balance of benefits and cost, as shown in Figure 1.



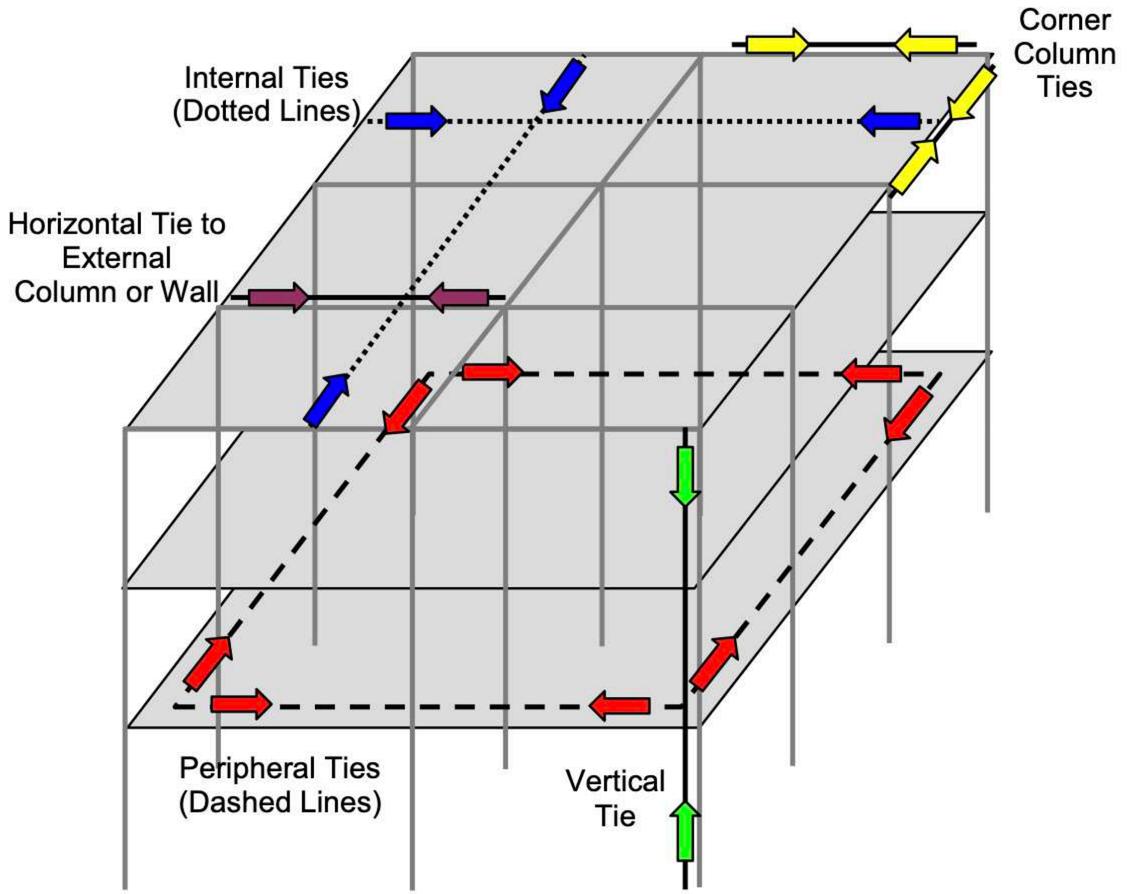
https://aws.amazon.com/blogs/architecture/disaster-recovery-dr-architecture-on-aws-part-iii-pilot-light-and-warm-standby/

We'll look at multi-vendor cloud shortly

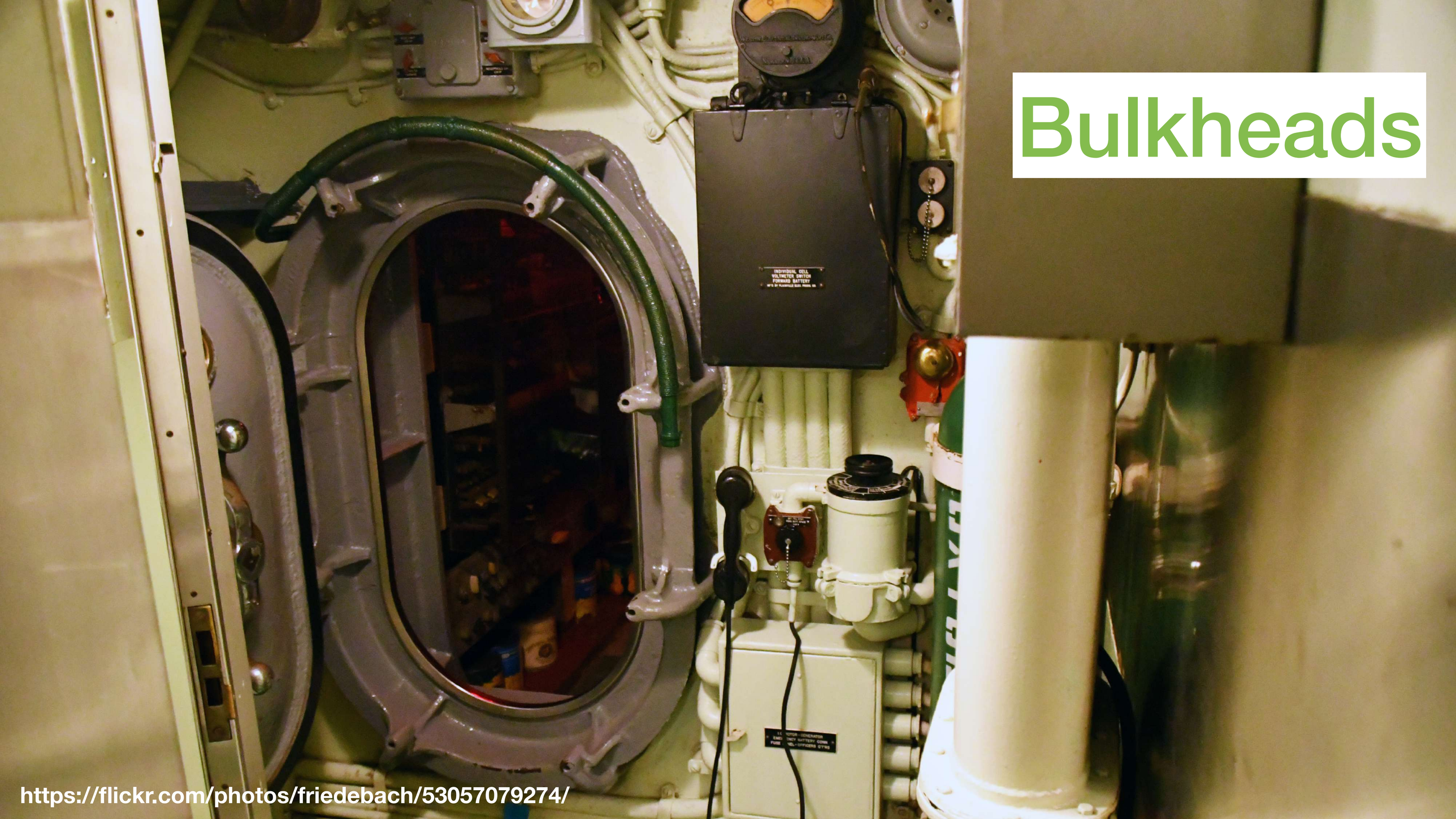
3. Reduce Interconnection

https://flickr.com/photos/testlab/21496317363/

Alternative load bearing paths

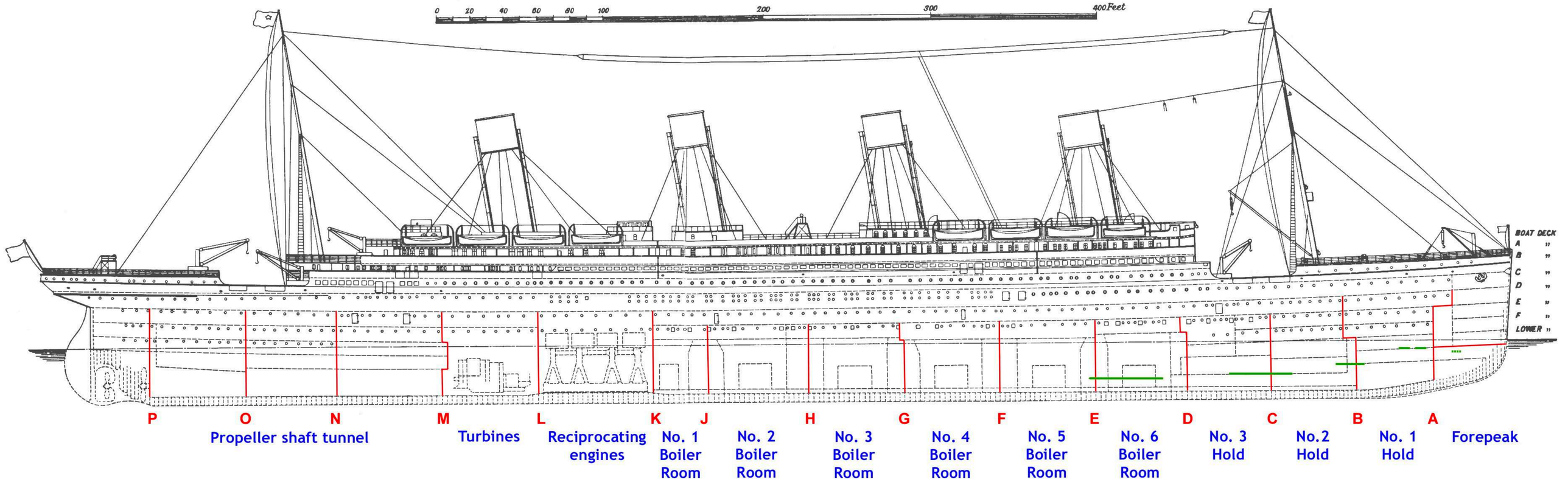


https://nvlpubs.nist.gov/nistpubs/Legacy/IR/nistir7396.pdf



Bulkheads

https://flickr.com/photos/friedebach/53057079274/



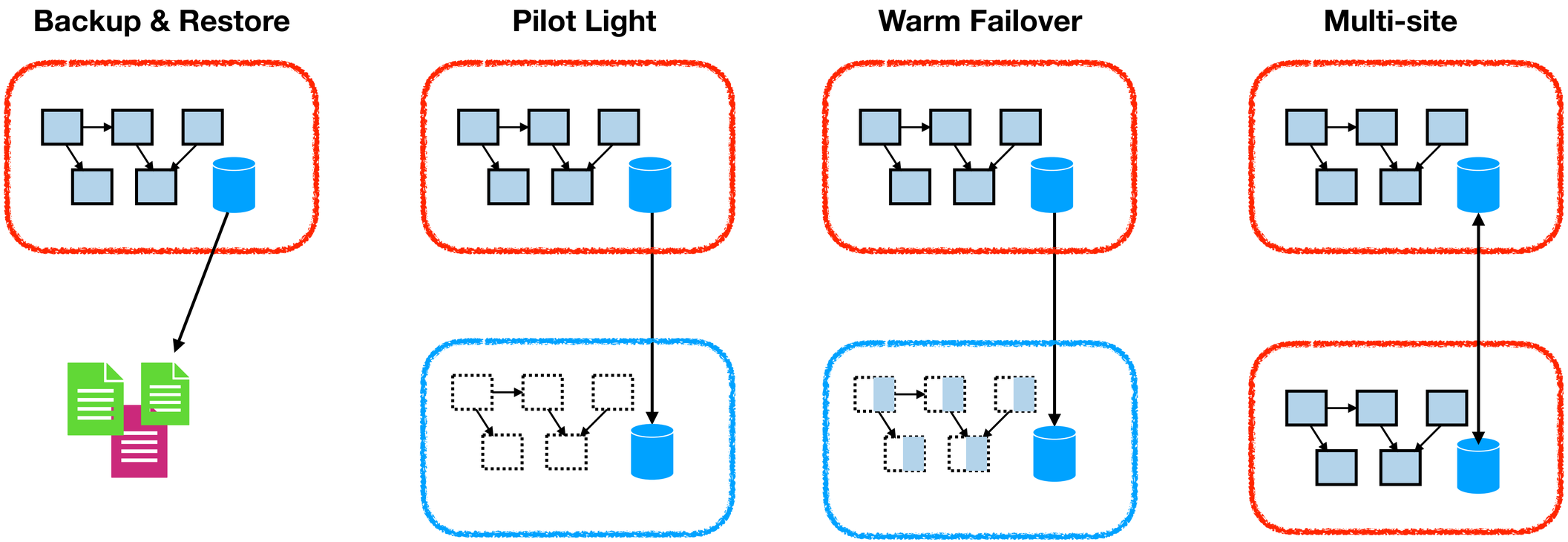
https://twitter.com/marinamaral2/status/985850360047783936

Smart-device vendors looking at ensuring devices can be used without needing the cloud

Why not multicoloud?

You need skills and expertise in not just one cloud, but two (or more)

MULTI-SITE OPTIONS



This is hedging against the risk of a whole cloud failure

Multi-site setups can increase interconnection

Processing payments in Monzo Stand-in

This blog post was accurate when we published it – head to monzo.com or your Monzo app for the most up to date information.

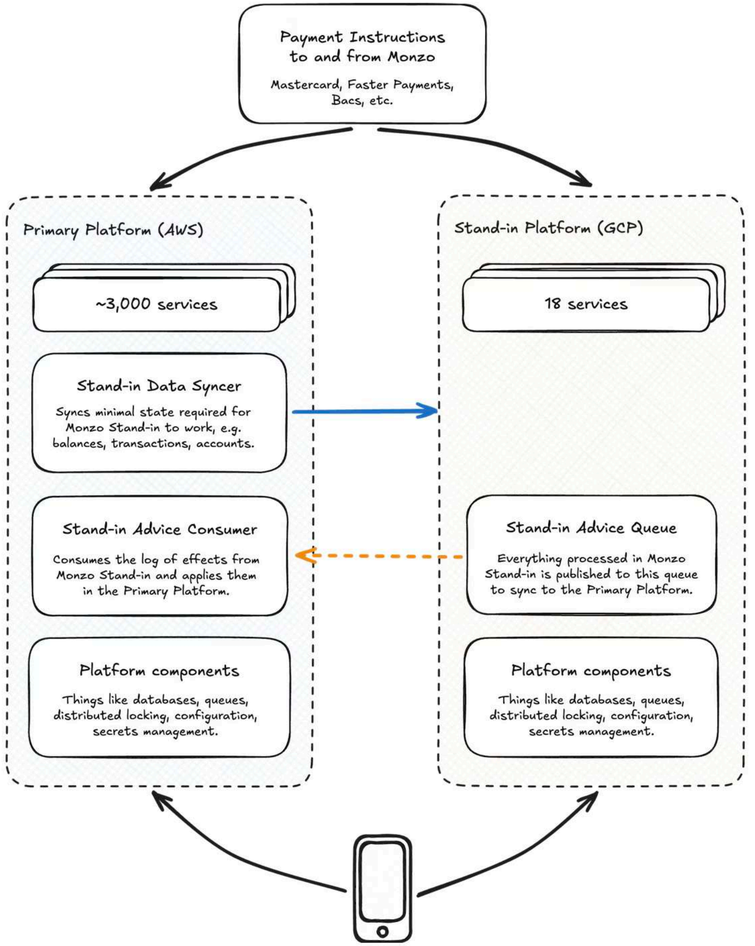
Monzo Stand-in is our backup bank that we can use in rare platform outages. An important function of a bank is our ability to process customer payments. We send and receive these customer payments through different *payment schemes* including:

- Card payments via Mastercard.
- Bank transfers via Faster Payments.
- Direct debits and direct credits via Bacs.

Monzo Stand-in is able to process these types of payments. This means that in the event of a major outage of the Primary Platform, customers can continue to use their card and send or receive bank transfers as normal.

How we receive payments in Monzo Stand-in

To understand how we receive payments in Monzo Stand-in, let's first cover how we receive them in the Primary Platform.



https://monzo.com/blog/tolerating-full-cloud-outages-with-monzo-stand-in

Progressive Collapse

A small failure results in a significant collapse in the wider system

1. Reduce Hazards

2. Strengthen Components

3. Reduce Interconnection

Summary of the Amazon DynamoDB Service Disruption in the Northern Virginia (US-EAST-1) Region

We wanted to provide you with some additional information about the service disruption that occurred in the N. Virginia (us-east-1) Region on October 19 and 20, 2025. While the event started at 11:48 PM PDT on October 19 and ended at 2:20 PM PDT on October 20, there were three distinct periods of impact to customer applications. First, between 11:48 PM on October 19 and 2:40 AM on October 20, Amazon DynamoDB experienced increased API error rates in the N. Virginia (us-east-1) Region. Second, between 5:30 AM and 2:09 PM on October 20, Network Load Balancer (NLB) experienced increased connection errors for some load balancers in the N. Virginia (us-east-1) Region. This was caused by health check failures in the NLB fleet, which resulted in increased connection errors on some NLBs. Third, between 2:25 AM and 10:36 AM on October 20, new EC2 instance launches failed and, while instance launches began to succeed from 10:37 AM, some newly launched instances experienced connectivity issues which were resolved by 1:50 PM.

DynamoDB

Between 11:48 PM PDT on October 19 and 2:40 AM PDT on October 20, customers experienced increased Amazon DynamoDB API error rates in the N. Virginia (us-east-1) Region. During this period, customers and other AWS services with dependencies on DynamoDB were unable to establish new connections to the service. The incident was triggered by a latent defect within the service's automated DNS management system that caused endpoint resolution failures for DynamoDB.

Many of the largest AWS services rely extensively on DNS to provide seamless scale, fault isolation and recovery, low latency, and locality. Services like DynamoDB maintain hundreds of thousands of DNS records to operate a very large heterogeneous fleet of load balancers in each Region. Automation is crucial to ensuring that these DNS records are updated frequently to add additional capacity as it becomes available, to correctly handle hardware failures, and to efficiently distribute traffic to optimize customers' experience. This automation has been designed for resilience, allowing the service to recover from a wide variety of operational issues. In addition to providing a public regional endpoint, this automation maintains additional DNS endpoints for several dynamic DynamoDB variants including a FIPS compliant endpoint, an IPv6 endpoint, and account-specific endpoints. The root cause of this issue was a latent race condition in the DynamoDB DNS management system that resulted in an incorrect empty DNS record for the service's regional endpoint (**dynamodb.us-east-1.amazonaws.com**) that the automation failed to repair. To explain this event, we need to share some details about the DynamoDB DNS management architecture. The system is split across two independent components for availability reasons. The first component, the DNS Planner, monitors the health and capacity of the load balancers and periodically creates a new DNS plan for each of the service's endpoints consisting of a set of load balancers and weights. We produce a single regional DNS plan, as this greatly simplifies capacity management and failure mitigation when capacity is shared

https://aws.amazon.com/message/101925/

NISTI

Best Practices for Reducing the Pot
for Progressive Collapse in Buil

NIST National Institute of Standards and Technology • Technology Administration • U.S. Department of Commerce

Summary of the Amazon DynamoDB Service Disruption in the Northern Virginia (US-EAST-1) Region

We wanted to provide you with some additional information about the service disruption that occurred in the N. Virginia (us-east-1) Region on October 19 and 20, 2025. While the event started at 11:48 PM PDT on October 19 and ended at 2:20 PM PDT on October 20, there were three distinct periods of impact to customer applications. First, between 11:48 PM on October 19 and 2:40 AM on October 20, Amazon DynamoDB experienced increased API error rates in the N. Virginia (us-east-1) Region. Second, between 5:30 AM and 2:09 PM on October 20, Network Load Balancer (NLB) experienced increased connection errors for some load balancers in the N. Virginia (us-east-1) Region. This was caused by health check failures in the NLB fleet, which resulted in increased connection errors on some NLBs. Third, between 2:25 AM and 10:36 AM on October 20, new EC2 instance launches failed and, while instance launches began to succeed from 10:37 AM, some newly launched instances experienced connectivity issues which were resolved by 1:50 PM.

DynamoDB

Between 11:48 PM PDT on October 19 and 2:40 AM PDT on October 20, customers experienced increased Amazon DynamoDB API error rates in the N. Virginia (us-east-1) Region. During this period, customers and other AWS services with dependencies on DynamoDB were unable to establish new connections to the service. The incident was triggered by a latent defect within the service's automated DNS management system that caused endpoint resolution failures for DynamoDB.

Many of the largest AWS services rely extensively on DNS to provide seamless scale, fault isolatio
locality. Services like DynamoDB maintain hundreds of thousands of DNS records to operate a ve
balancers in each Region. Automation is crucial to ensuring that these DNS records are updated
as it becomes available, to correctly handle hardware failures, and to efficiently distribute traffic
This automation has been designed for resilience, allowing the service to recover from a wide va
addition to providing a public regional endpoint, this automation maintains additional DNS endp
DynamoDB variants including a FIPS compliant endpoint, an IPv6 endpoint, and account-specific
issue was a latent race condition in the DynamoDB DNS management system that resulted in an
service's regional endpoint (**dynamodb.us-east-1.amazonaws.com**) that the automation failed
need to share some details about the DynamoDB DNS management architecture. The system is s
components for availability reasons. The first component, the DNS Planner, monitors the health
and periodically creates a new DNS plan for each of the service's endpoints consisting of a set of
produce a single regional DNS plan, as this greatly simplifies capacity management and failure r

Processing payments in Monzo Stand-in

This blog post was accurate when we published it – head to monzo.com or your Monzo app for the most up to date information.

Monzo Stand-in is our backup bank that we can use in rare platform outages. An important function of a bank is our ability to process customer payments. We send and receive these customer payments through different *payment schemes* including:

- Card payments via Mastercard.
- Bank transfers via Faster Payments.
- Direct debits and direct credits via Bacs.

Monzo Stand-in is able to process these types of payments. This means that in the event of a major outage of the Primary Platform, customers can continue to use their card and send or receive bank transfers as normal.

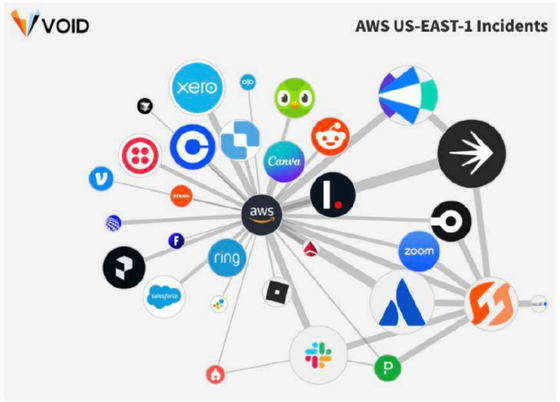
How we receive payments in Monzo Stand-in

To understand how we receive payments in Monzo Stand-in, let's first cover how we receive them in the Primary Platform.

Help Update the AWS Outage Network Graph

We're building an interactive network graph that shows which companies were impacted by the AWS outage, and also some of the interdependencies between services like Slack, Zoom, CircleCI, Launch Darkly and more. We're looking for the community's input to help us add companies that were impacted by this outage.

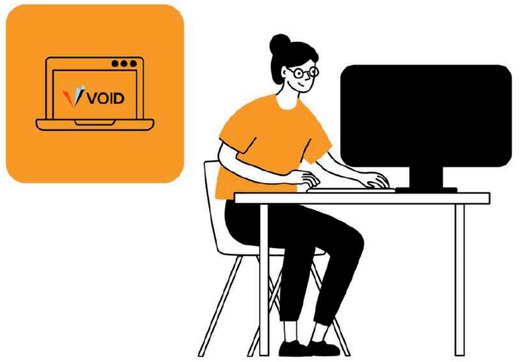
Learn more



Announcing the VOID Incident Management Survey

We're conducting the first ever industry benchmark survey on incident management: what organizations do to prepare for, manage, and analyze incidents. We plan to use this research to advocate for better incident practices in the industry, and to plan future research to help improve the experience of anyone who deals with software incidents as a part of their job.

Take the survey



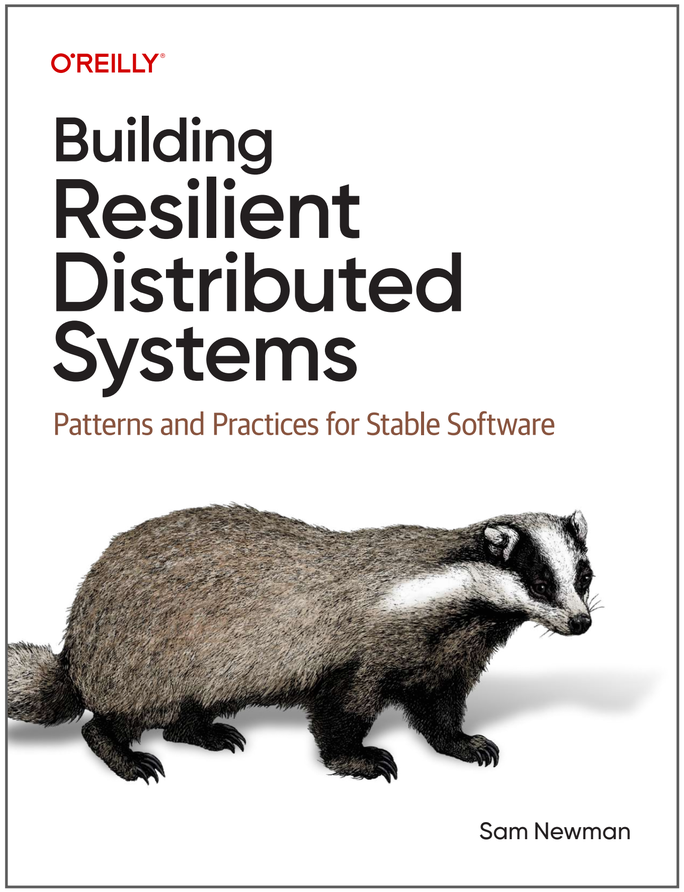
https://www.thevoid.community/

Create every opportunity
to learn from a failure

And share what
you learned

https://youtu.be/llvD2agDBnM?si=nKqdg5FlWdMs7Pdu

NEW BOOK IN EARLY ACCESS!



Slides

https://samnewman.io/books/building-resilient-distributed-systems/