
AWS
re:Invent
NOV. 28 – DEC. 2, 2022 | LAS VEGAS, NV

NET402

Dive deep on AWS networking infrastructure

Stephen Callaghan (he/him)
Sr. Principal Engineer
AWS

JR Rivers (he/him)
Sr. Principal Engineer
AWS

Agenda

Networking tenets

Ideal vs. real

Evolution of infrastructure networking

Hardware innovation

Software innovation

AWS networking

Infrastructure networking	Amazon EC2 networking	Edge networking
Routers/switches	Virtual private cloud (VPC)	Amazon Route 53
Copper/optical cables	Elastic network interface	AWS Global Accelerator
Data centers	AWS Hyperplane	Amazon CloudFront
Inter-Region backbone	Elastic Fabric Adapter (EFA)	AWS Direct Connect
Internet peering/transit	Placement groups	AWS Cloud WAN

Tenets

Tenets

Secure

Tenets

Secure

Available

Tenets

Secure

Available

Scalable

Tenets

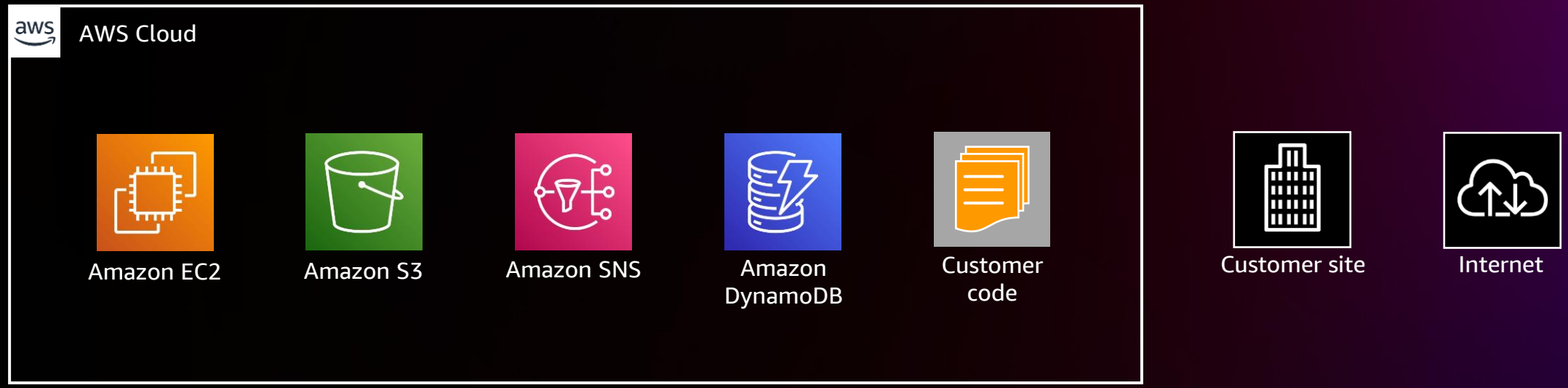
Secure

Available

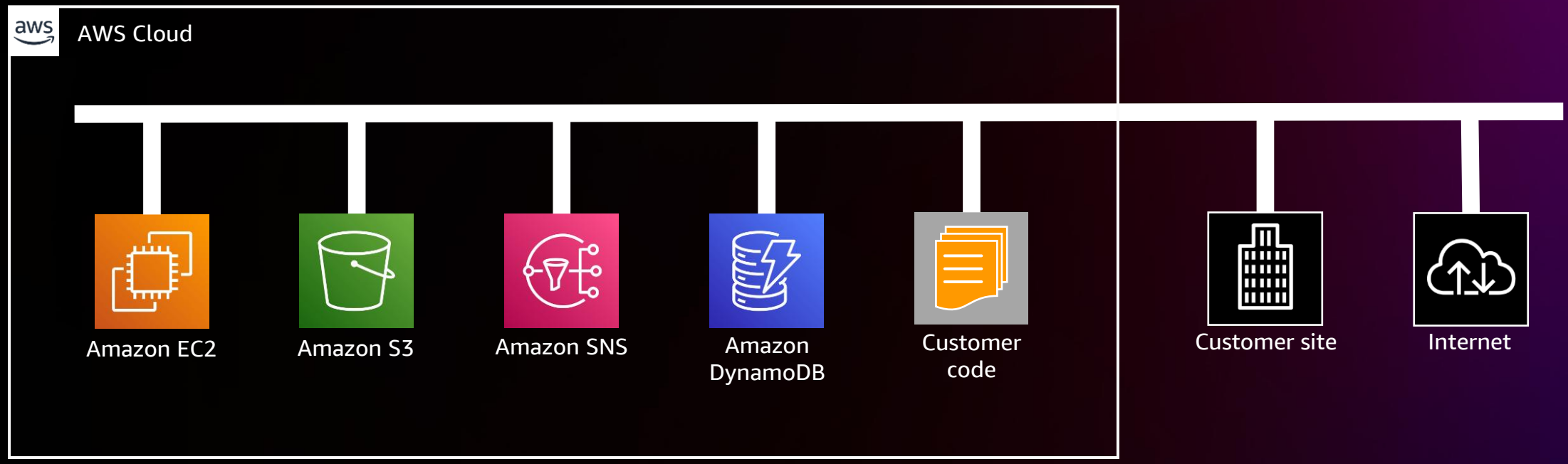
Scalable

Performant

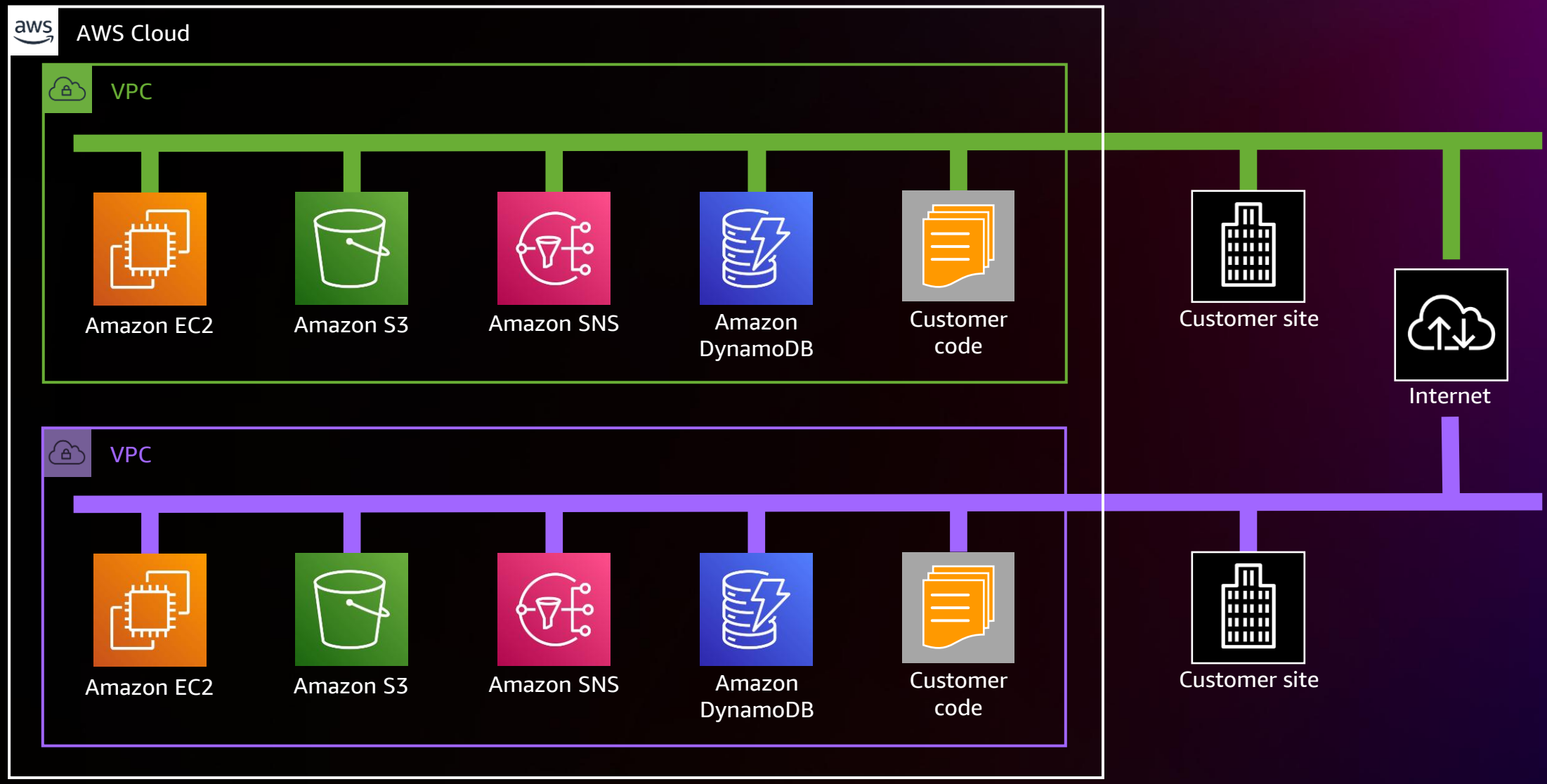
Ideal vs. real



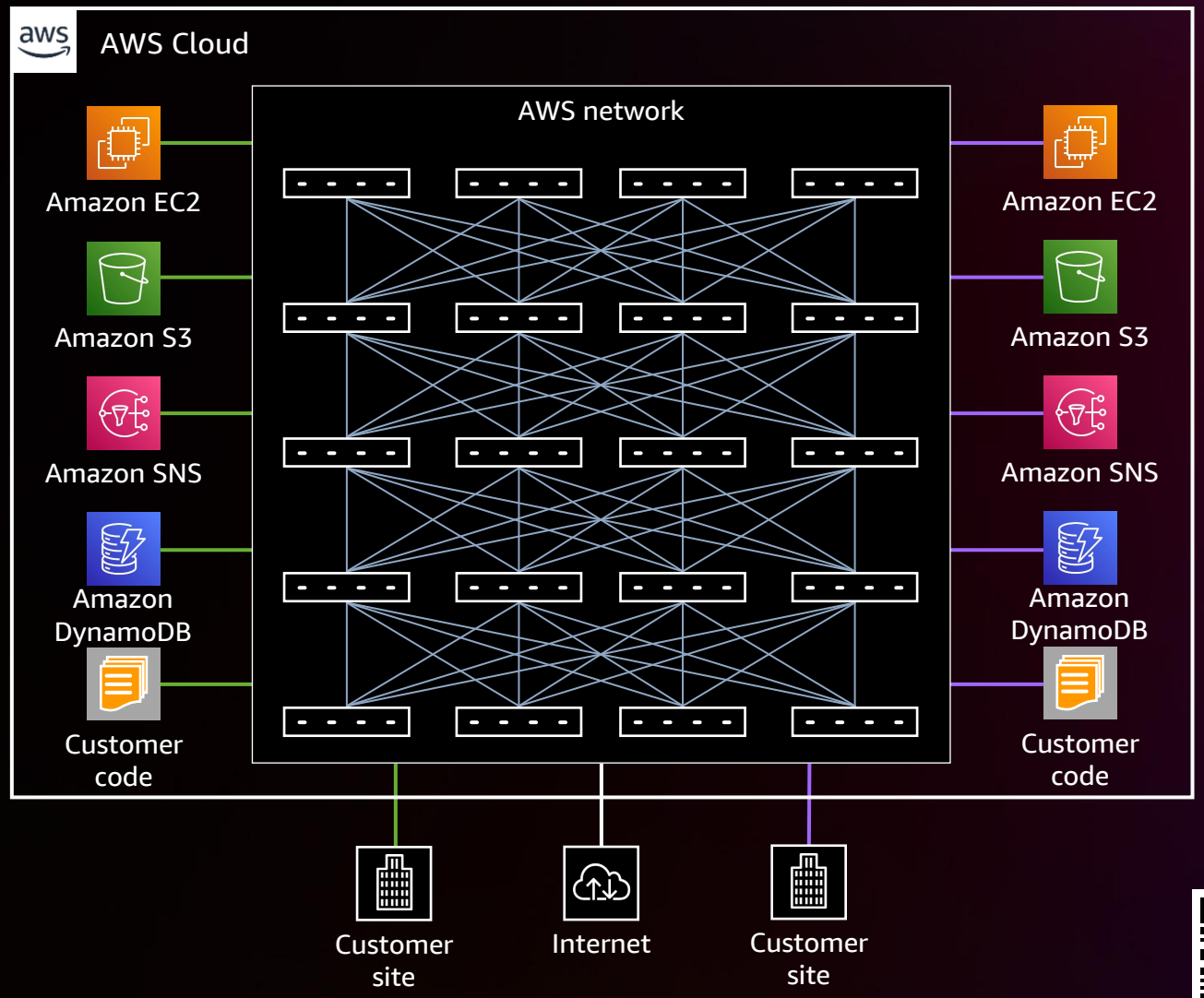
Ideal vs. real



Ideal vs. real



Ideal vs. real



Another Day, Another Billion Packets

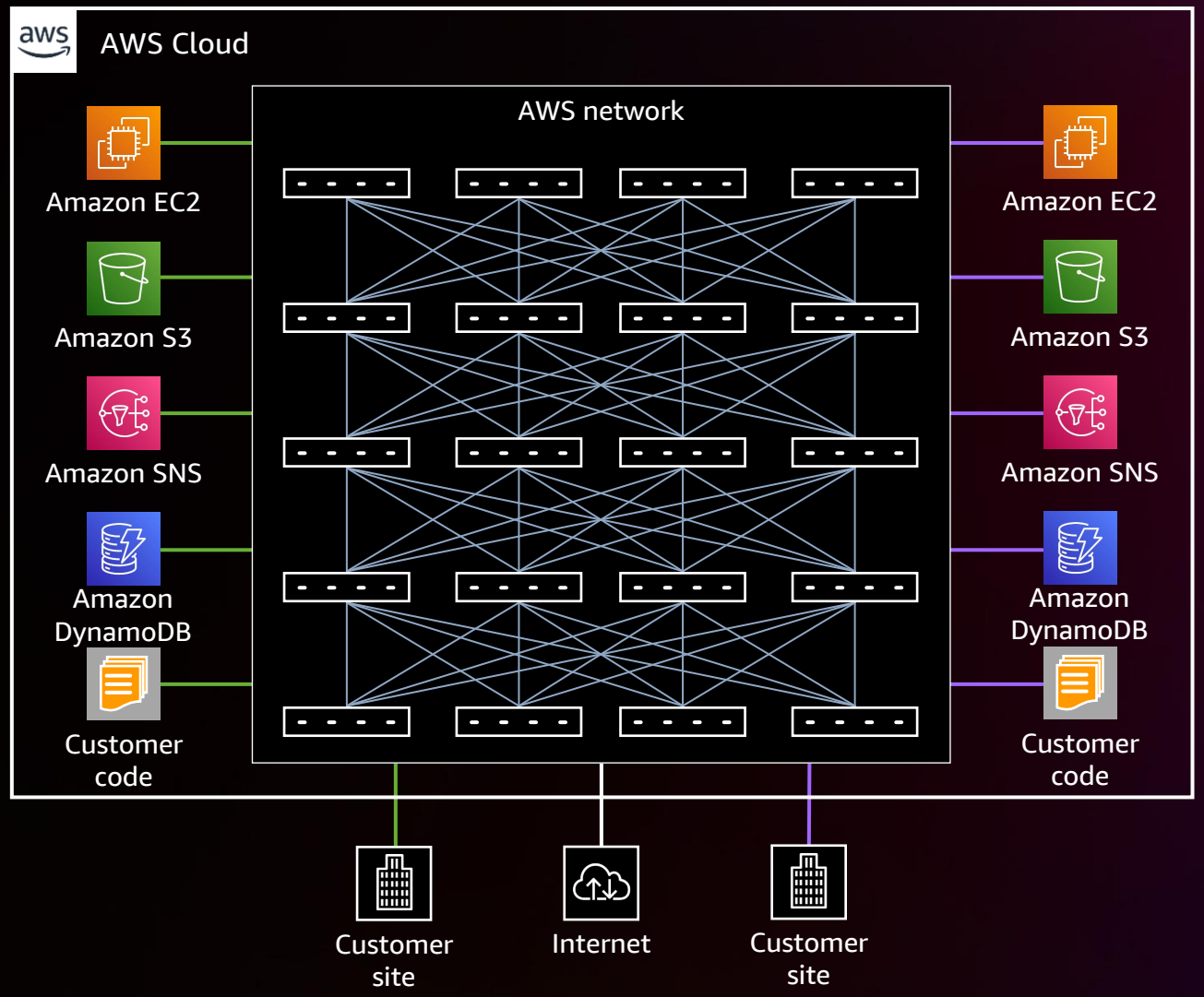
Ideal vs. real

Configuration updates

Software upgrades

Tooling enhancements

Network scaling



Device failures

Link cuts

Software faults

Evolution of infrastructure networking

Owning a large network

James Hamilton: “Datacenter Networks are in my Way”

https://perspectives.mvdirona.com/2010/10/datacenter-networks-are-in-my-way/

Owning a large network

James Hamilton: “Datacenter Networks are in my Way”

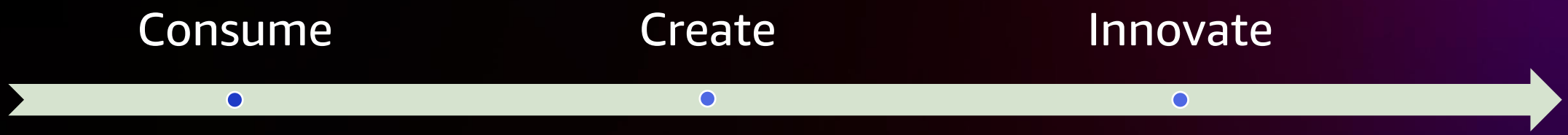
https://perspectives.mvdirona.com/2010/10/datacenter-networks-are-in-my-way/

AWS blog post, 2010-07-13

“New Amazon EC2 Instance Type – The Cluster Compute Instance”

Within this network you can create one or more placement groups of type “cluster” and then launch Cluster Compute Instances within each group. Instances within each placement group of this type benefit from non-blocking bandwidth and low latency node to node communication.

Phases of evolution



Consume

Industry hardware and software

Basic automation

Pushed beyond design intentions

Large chassis backplane/midplane

Core concepts into create

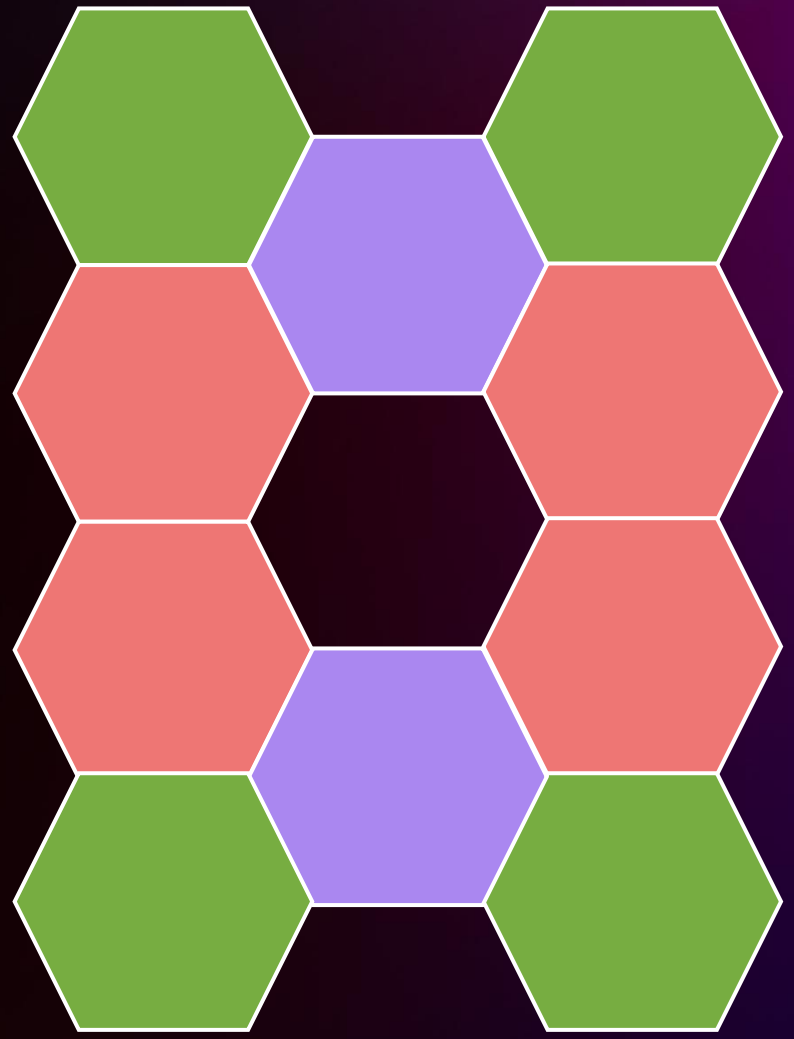
Embrace Moore’s law

Own our destiny

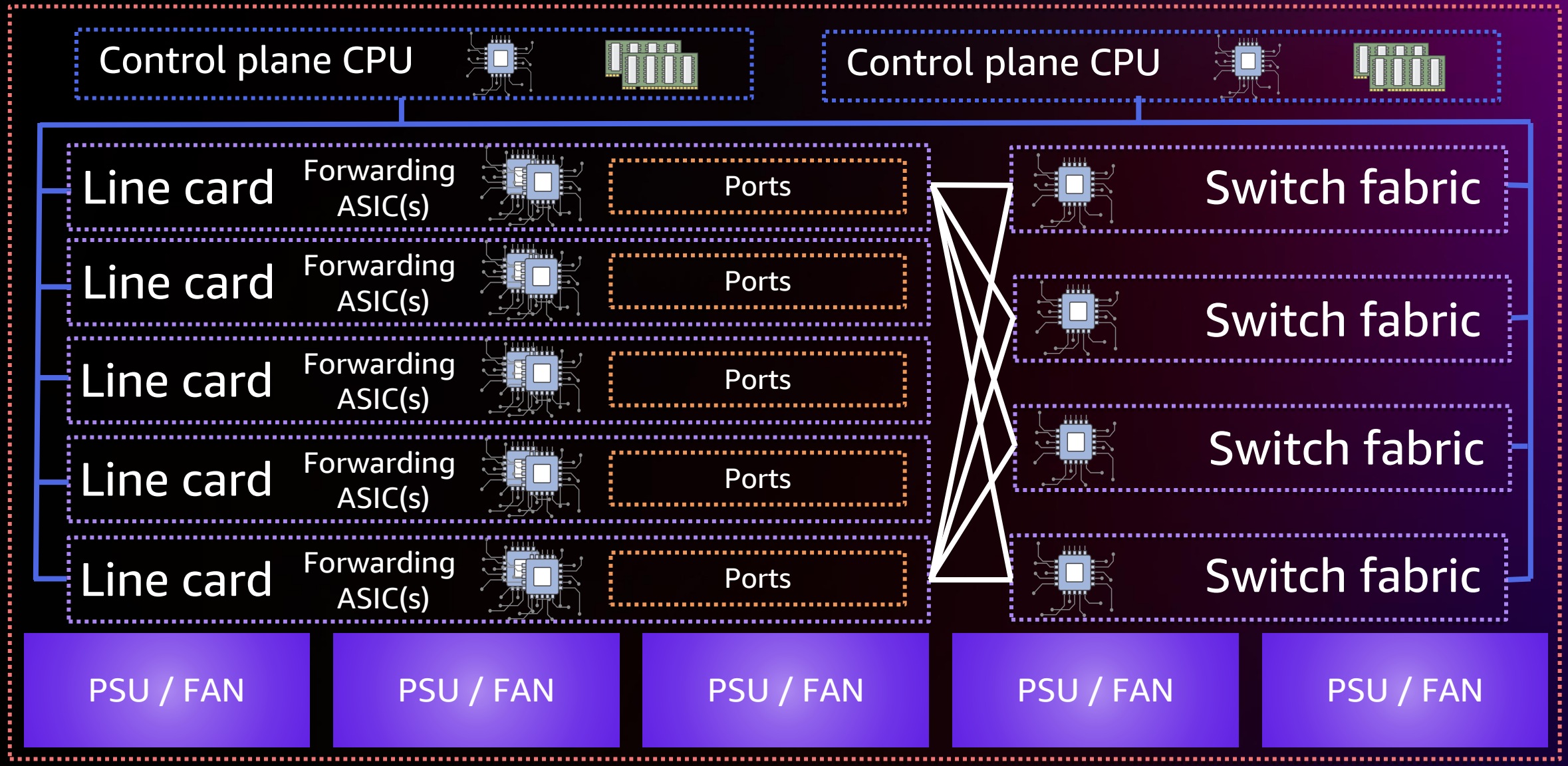
Use repeatable design patterns

Limit effect boundaries

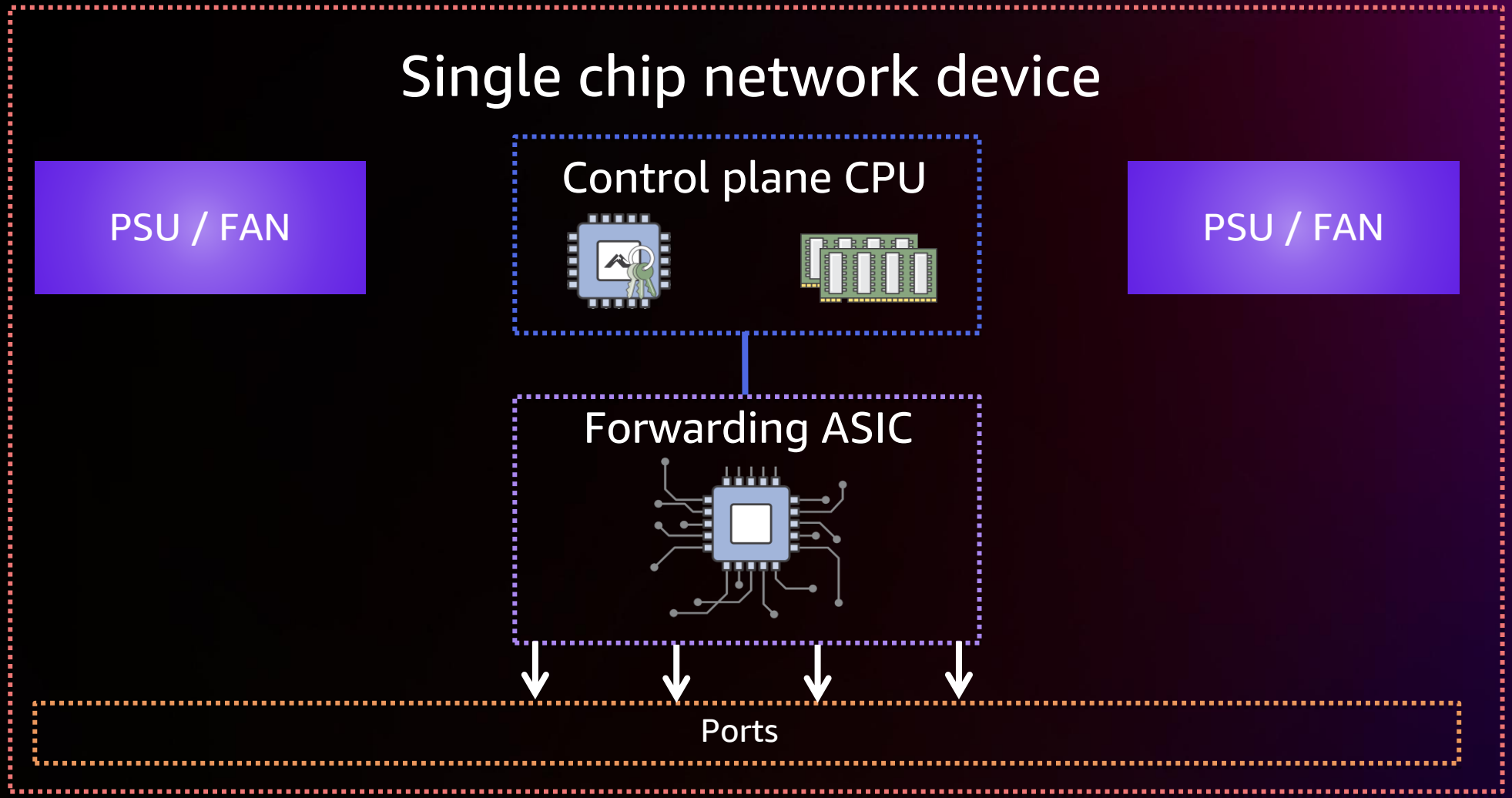
Constantly iterate and evolve



Chassis platforms



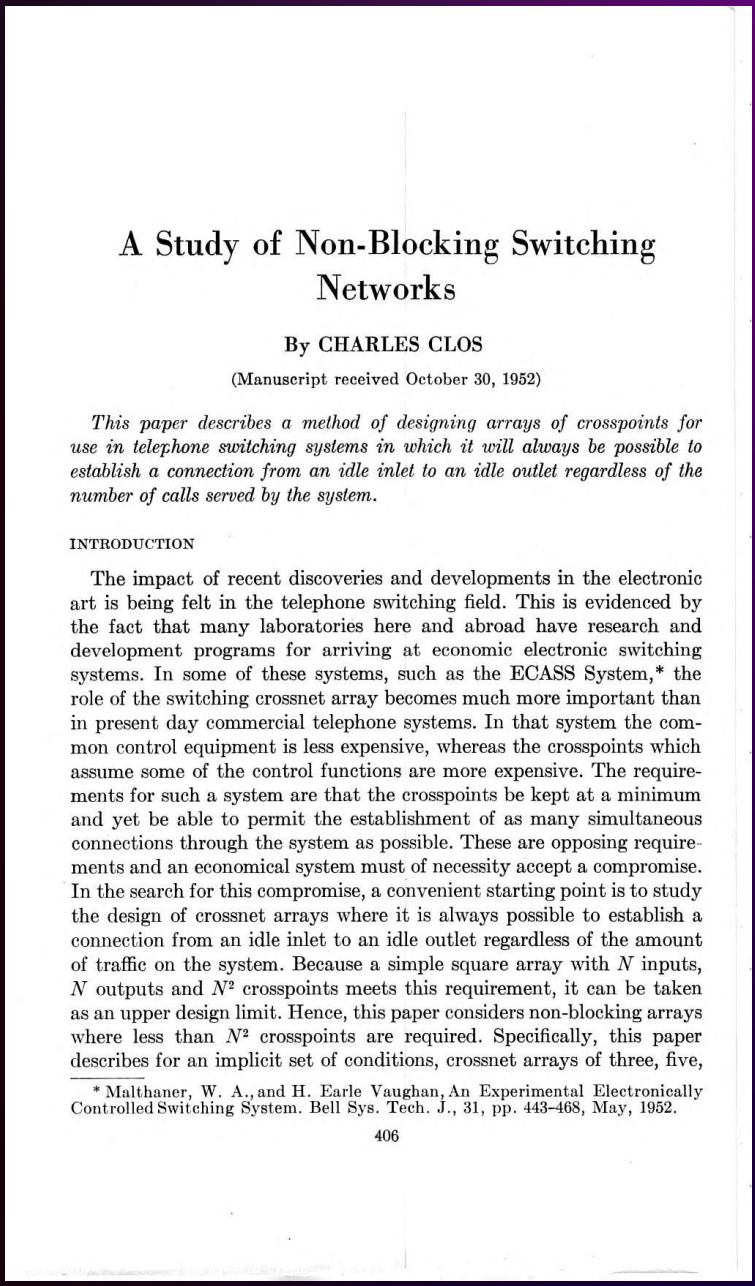
Single chip–based platforms



Create

TOPOLOGY AND HARDWARE

Clos fabric



A Study of Non-Blocking Switching Networks

By CHARLES CLOS

(Manuscript received October 30, 1952)

This paper describes a method of designing arrays of crosspoints for use in telephone switching systems in which it will always be possible to establish a connection from an idle inlet to an idle outlet regardless of the number of calls served by the system.

INTRODUCTION

The impact of recent discoveries and developments in the electronic art is being felt in the telephone switching field. This is evidenced by the fact that many laboratories here and abroad have research and development programs for arriving at economic electronic switching systems. In some of these systems, such as the ECASS System,* the role of the switching crossnet array becomes much more important than in present day commercial telephone systems. In that system the common control equipment is less expensive, whereas the crosspoints which assume some of the control functions are more expensive. The requirements for such a system are that the crosspoints be kept at a minimum and yet be able to permit the establishment of as many simultaneous connections through the system as possible. These are opposing requirements and an economical system must of necessity accept a compromise. In the search for this compromise, a convenient starting point is to study the design of crossnet arrays where it is always possible to establish a connection from an idle inlet to an idle outlet regardless of the amount of traffic on the system. Because a simple square array with N inputs, N outputs and N^2 crosspoints meets this requirement, it can be taken as an upper design limit. Hence, this paper considers non-blocking arrays where less than N^2 crosspoints are required. Specifically, this paper describes for an implicit set of conditions, crossnet arrays of three, five,

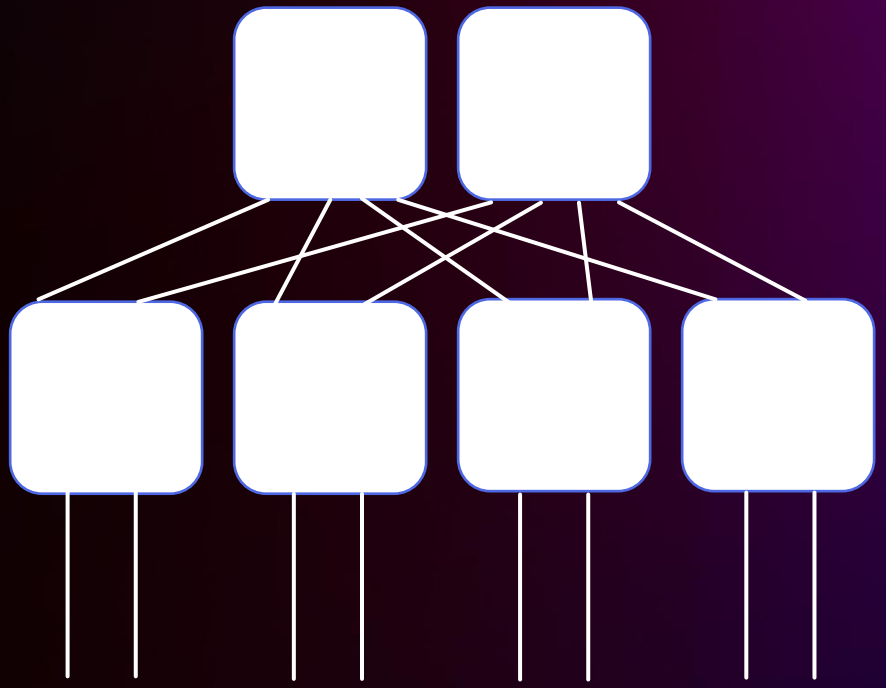
* Malthaner, W. A., and H. Earle Vaughan, An Experimental Electronically Controlled Switching System. Bell Sys. Tech. J., 31, pp. 443–468, May, 1952.

406

Create

TOPOLOGY AND HARDWARE

Clos fabric

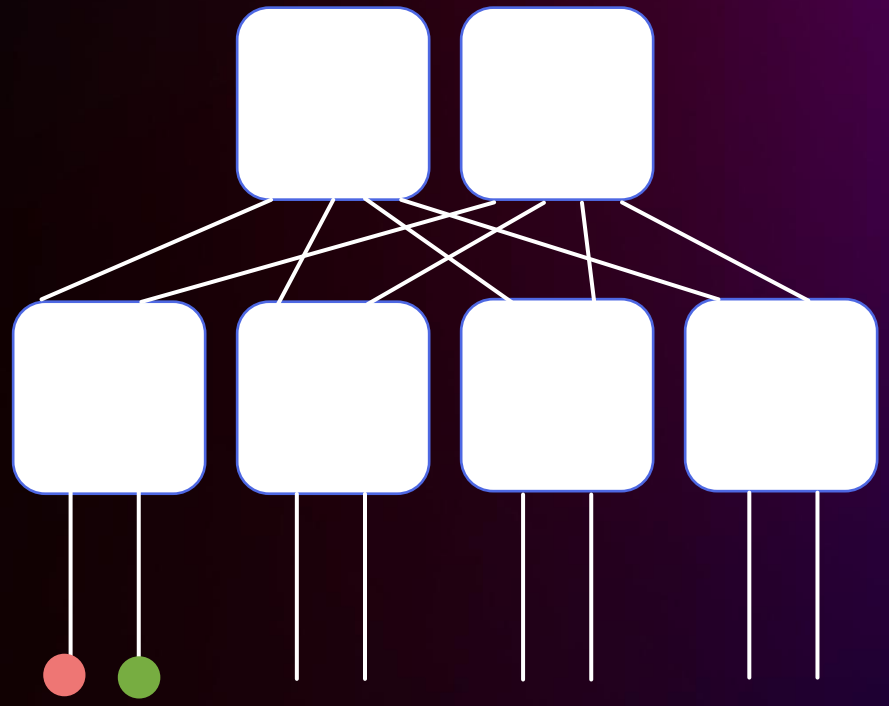


6 x 4-port switches = 8-port switch

Create

TOPOLOGY AND HARDWARE

Clos fabric



6 x 4-port switches = 8-port switch

Create

TOPOLOGY AND HARDWARE

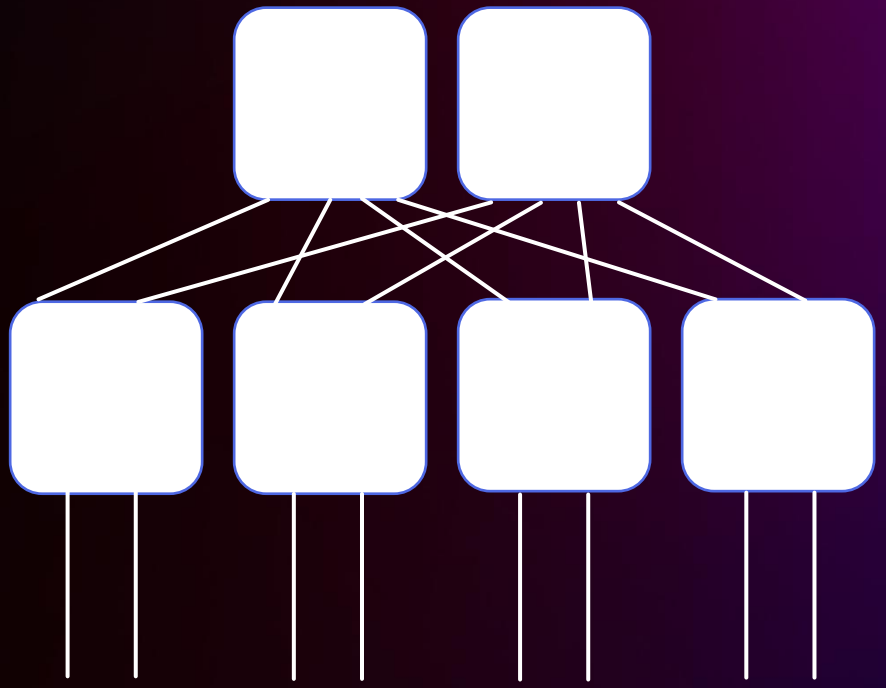
Clos fabric

32 to 5,120 devices

Merchant silicon

Internal software

Optimized network design



6 x 4-port switches = 8-port switch

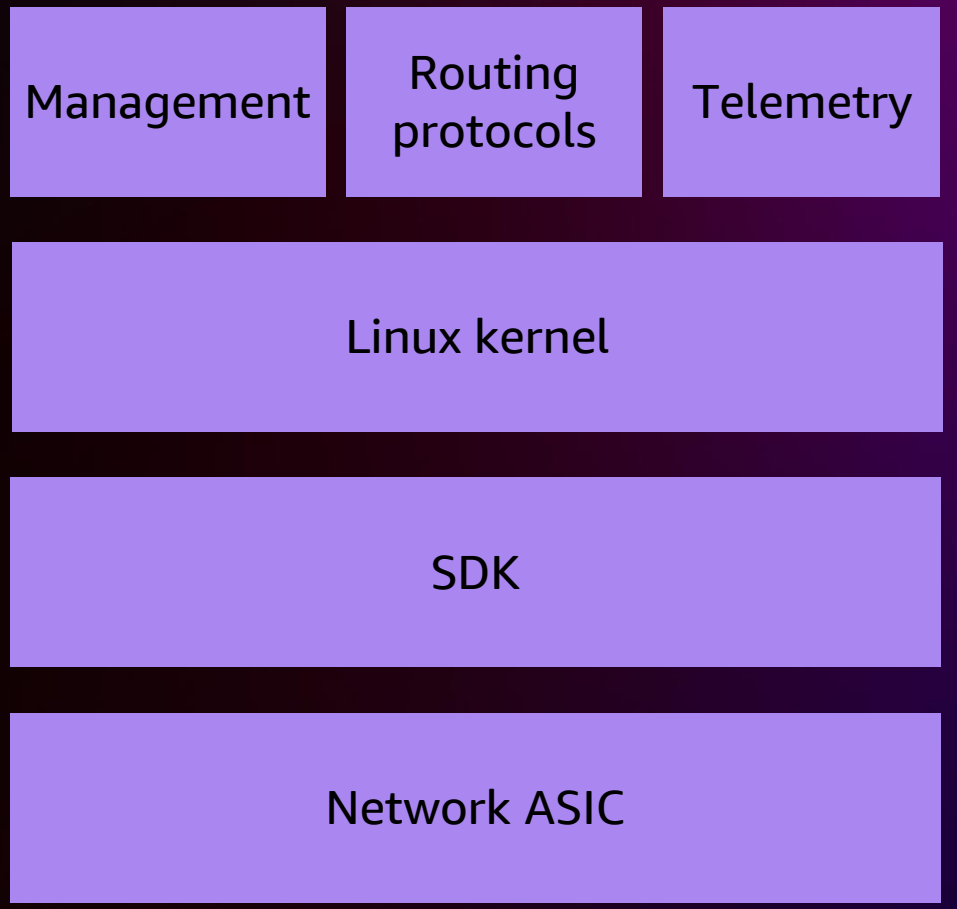
Create

NETWORK OPERATING SYSTEM

Linux-based

Multi-sourced manufacturing

Multi-ASIC



Create

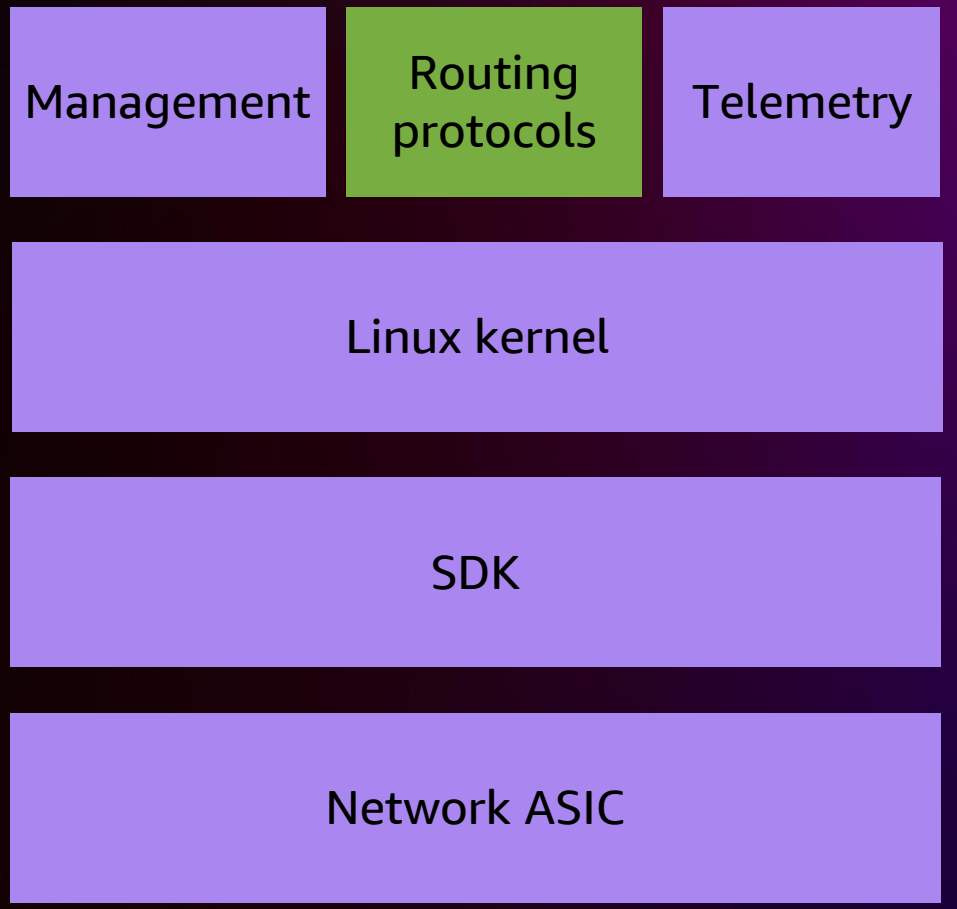
NETWORK OPERATING SYSTEM

Linux-based

Multi-sourced manufacturing

Multi-ASIC

OSPF/BGP ++



Create

Config generation

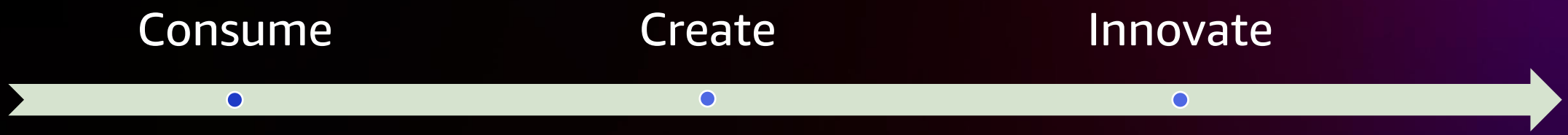
Deployment coordination

Active telemetry

Auto-remediation

NOC-less

Phases of evolution



Innovate

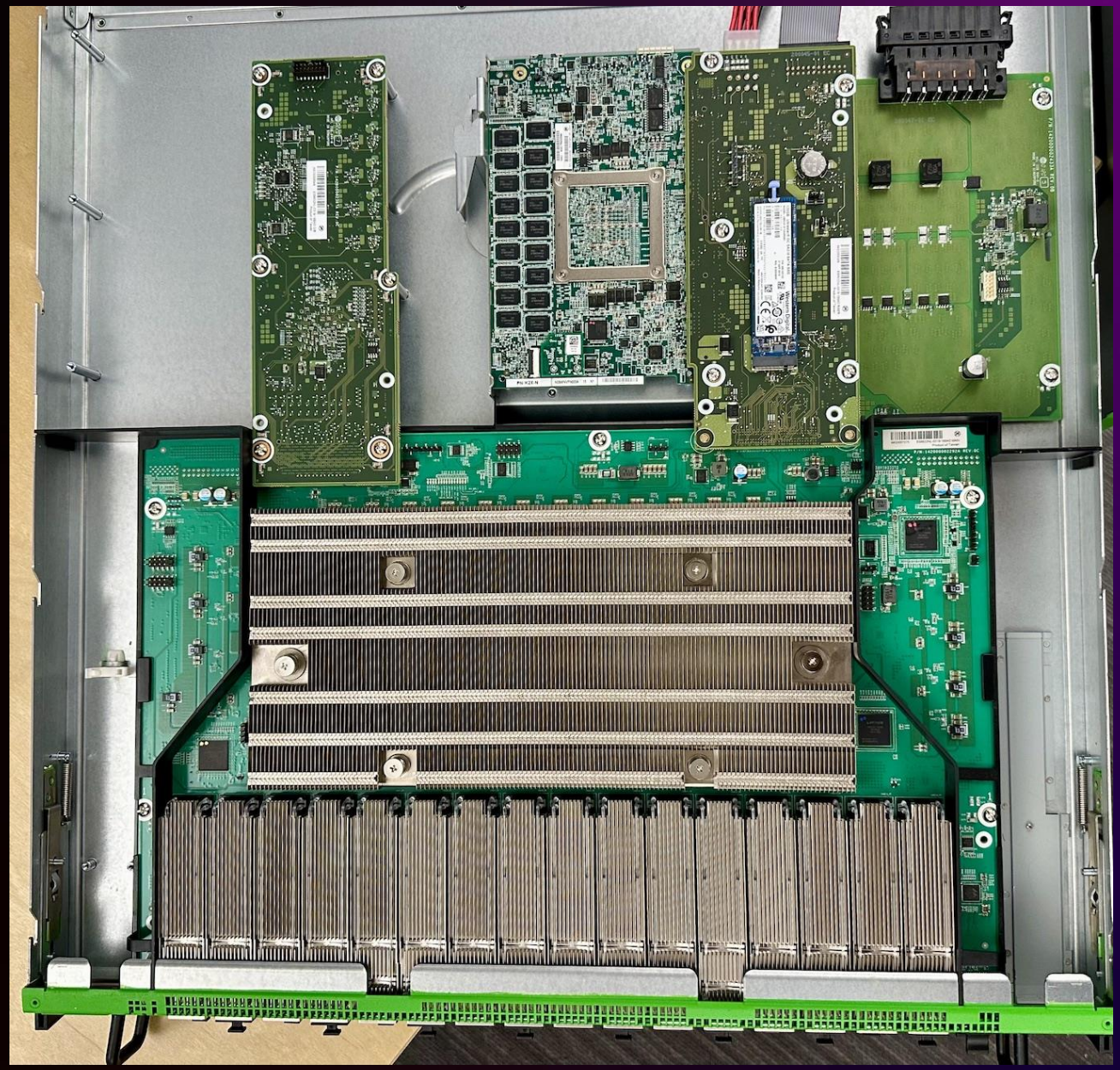
PRESENT DAY

Freedom to examine trade-offs

Custom hardware

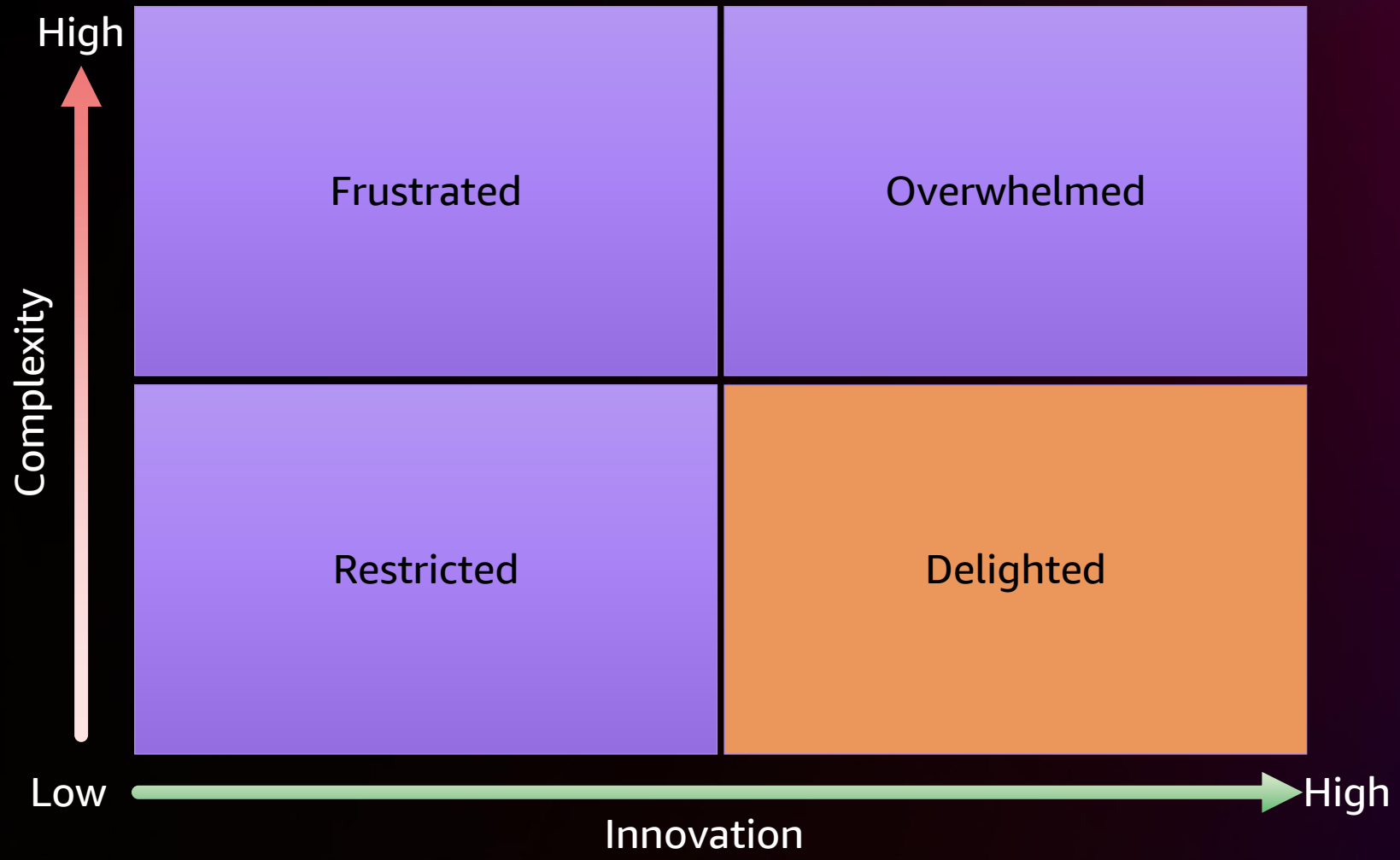
Multi-domain applications

Focus on the benefit



Innovate

CHOICE + SCALE = COMPLEXITY



Innovate

Simplicity scales

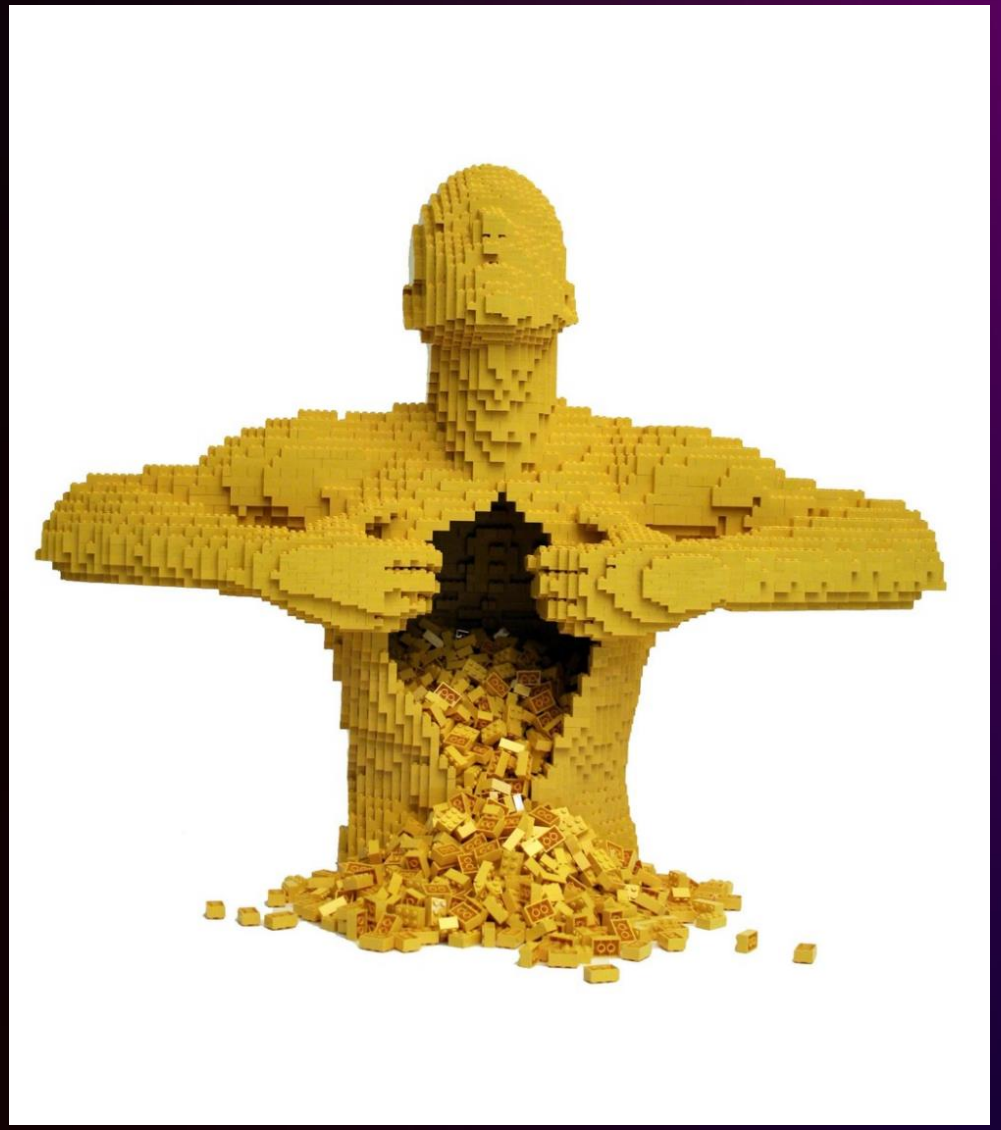
Complexity creates technical debt. A complex network is one that creates decisions during deployment, introduces uncertainty in operations and has corner cases in design. Simple systems are more robust. We think hard before adding features, functions, options or variations.

Hardware deep dive

How we do it

"A complex system that works is invariably found to have evolved from a simple system that worked. The inverse proposition also appears to be true: A complex system designed from scratch never works and cannot be made to work. You have to start over, beginning with a working simple system."

John Gall, *General Systemantics: An essay on how systems work, and especially how they fail*, 1975



Source: Nathan Sawaya brickartist.com

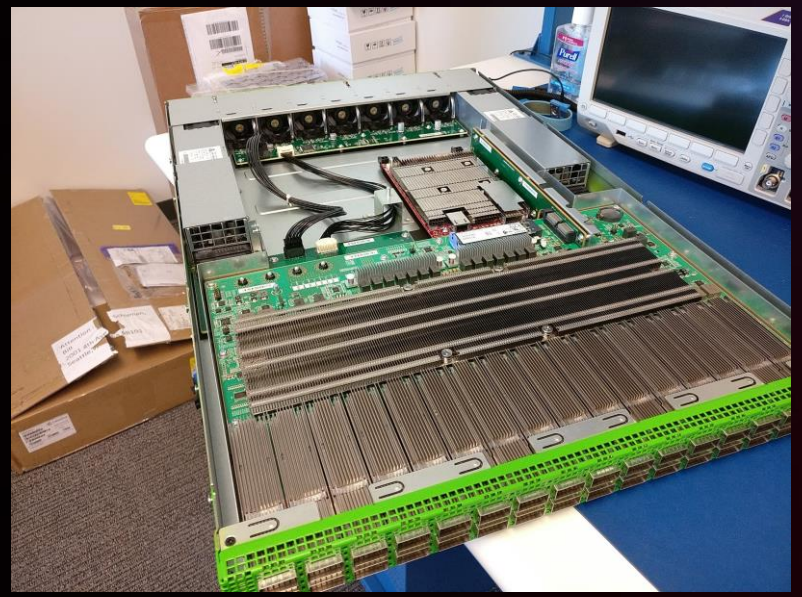
How we do it

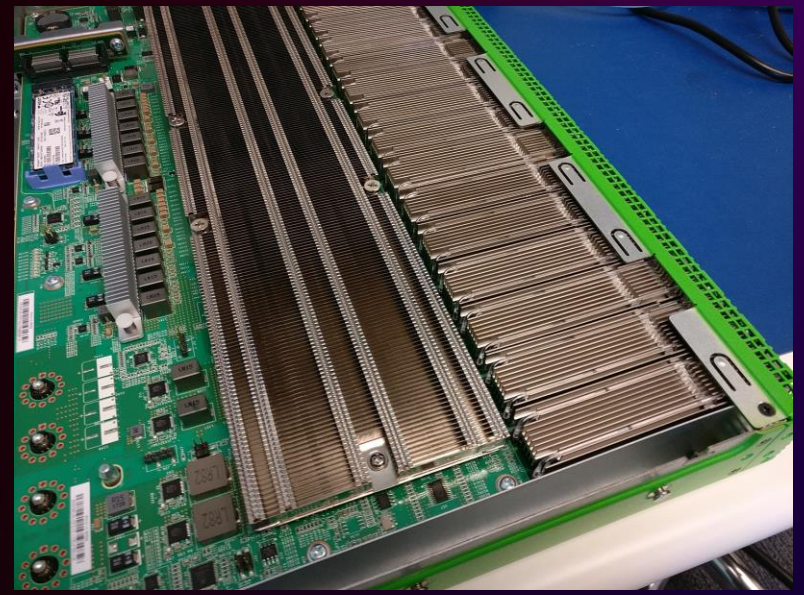
NOT REALLY. . .

Image source: CERN

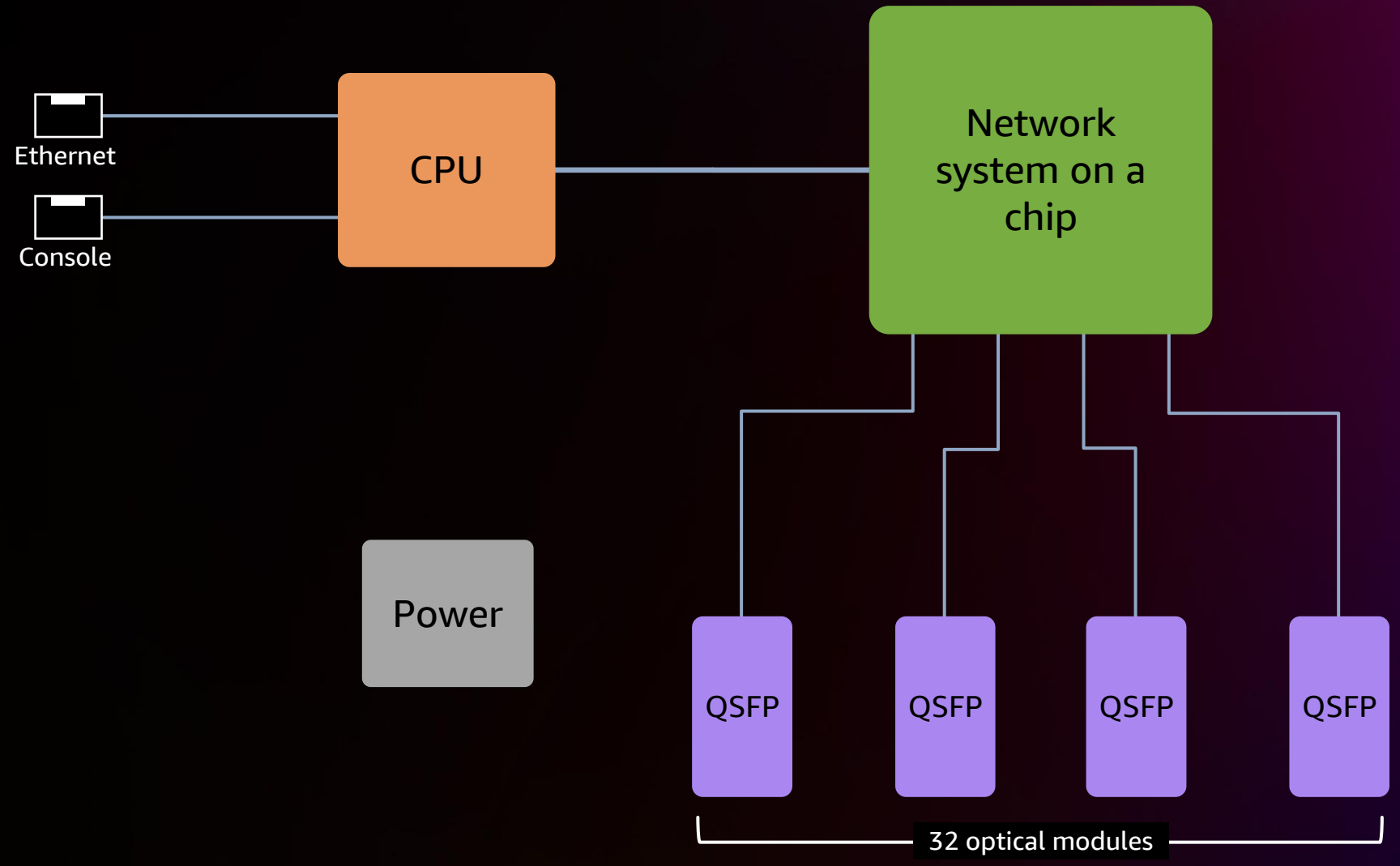
How we do it

WE'VE COME A LONG WAY

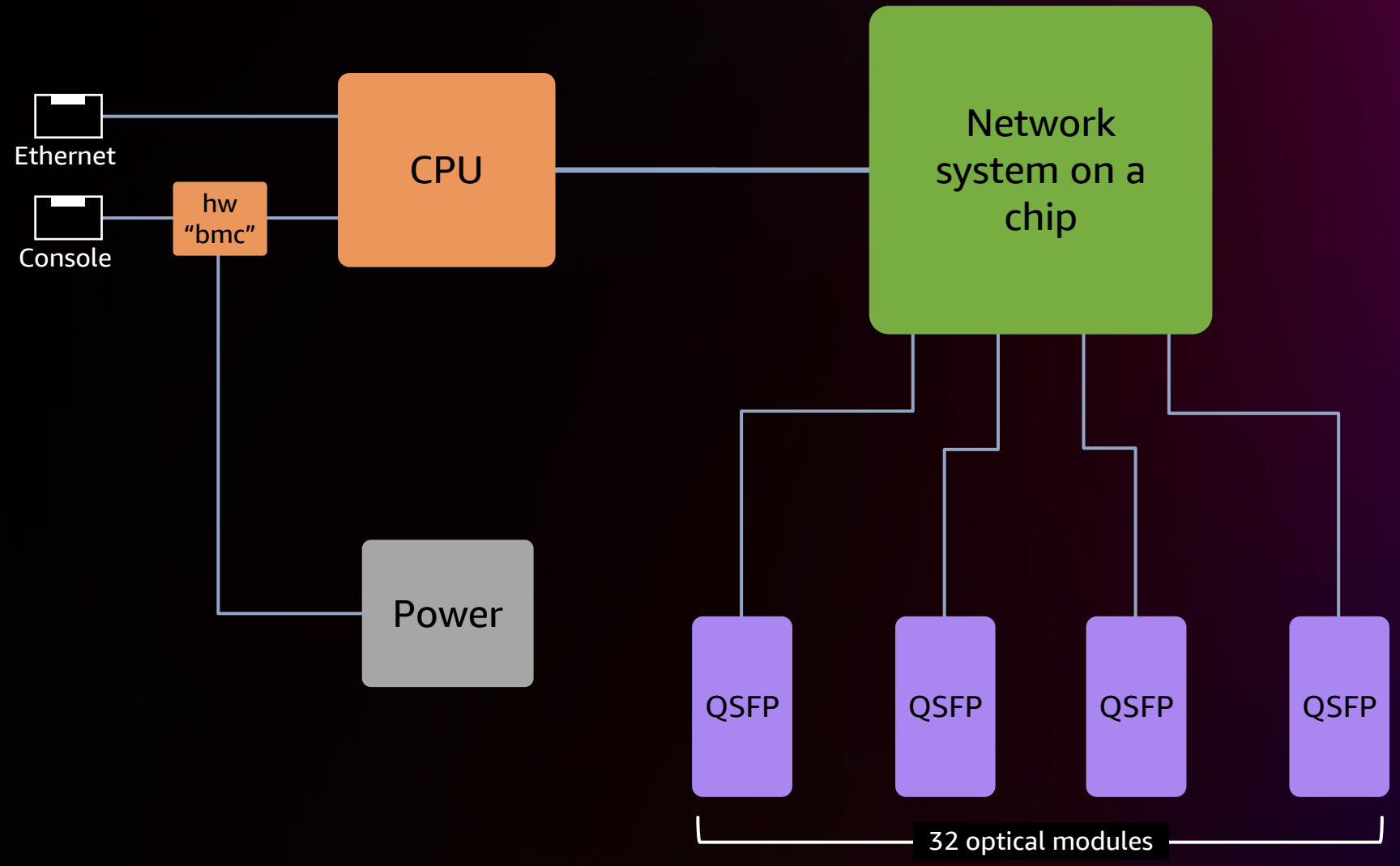




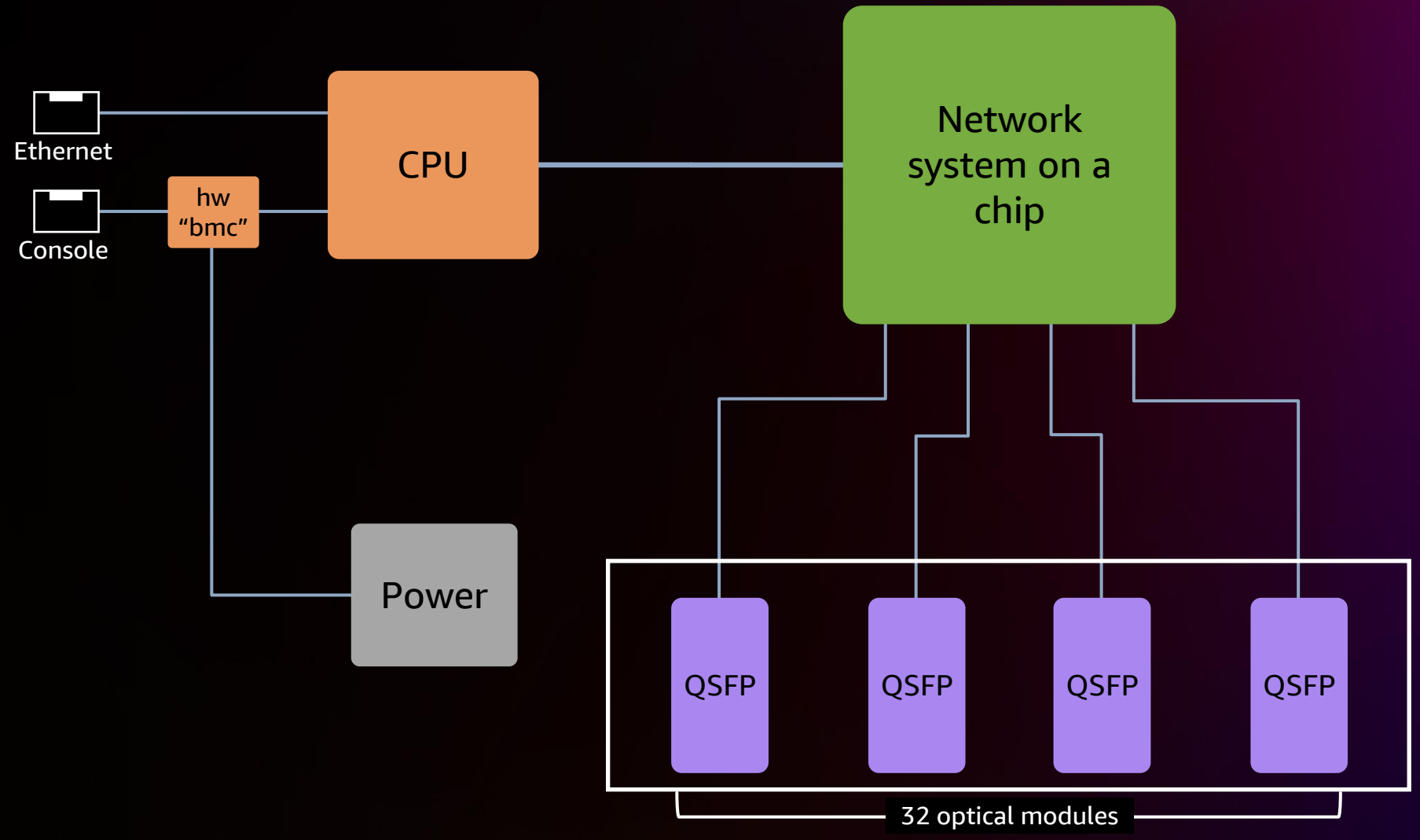
How we do it



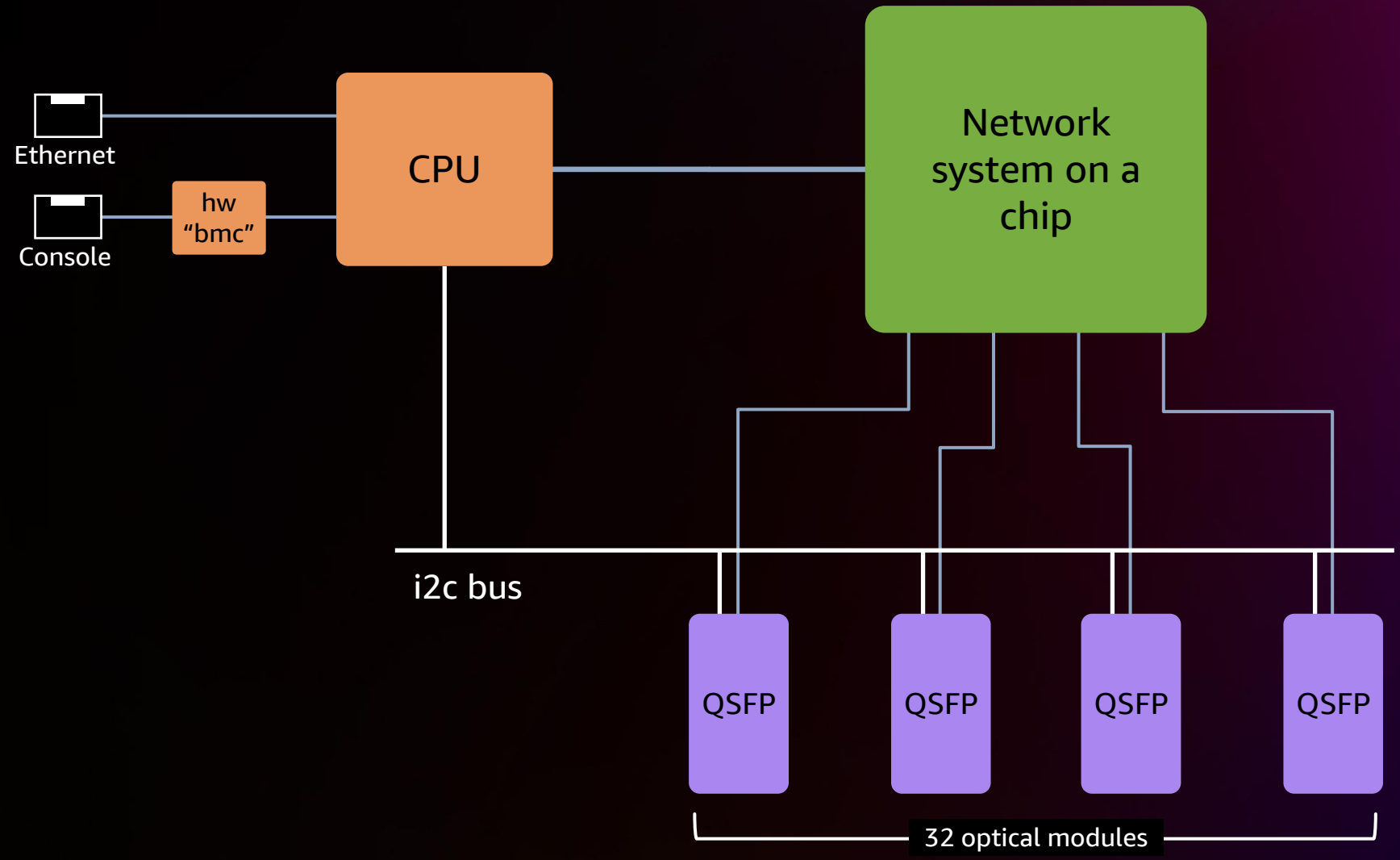
How we do it



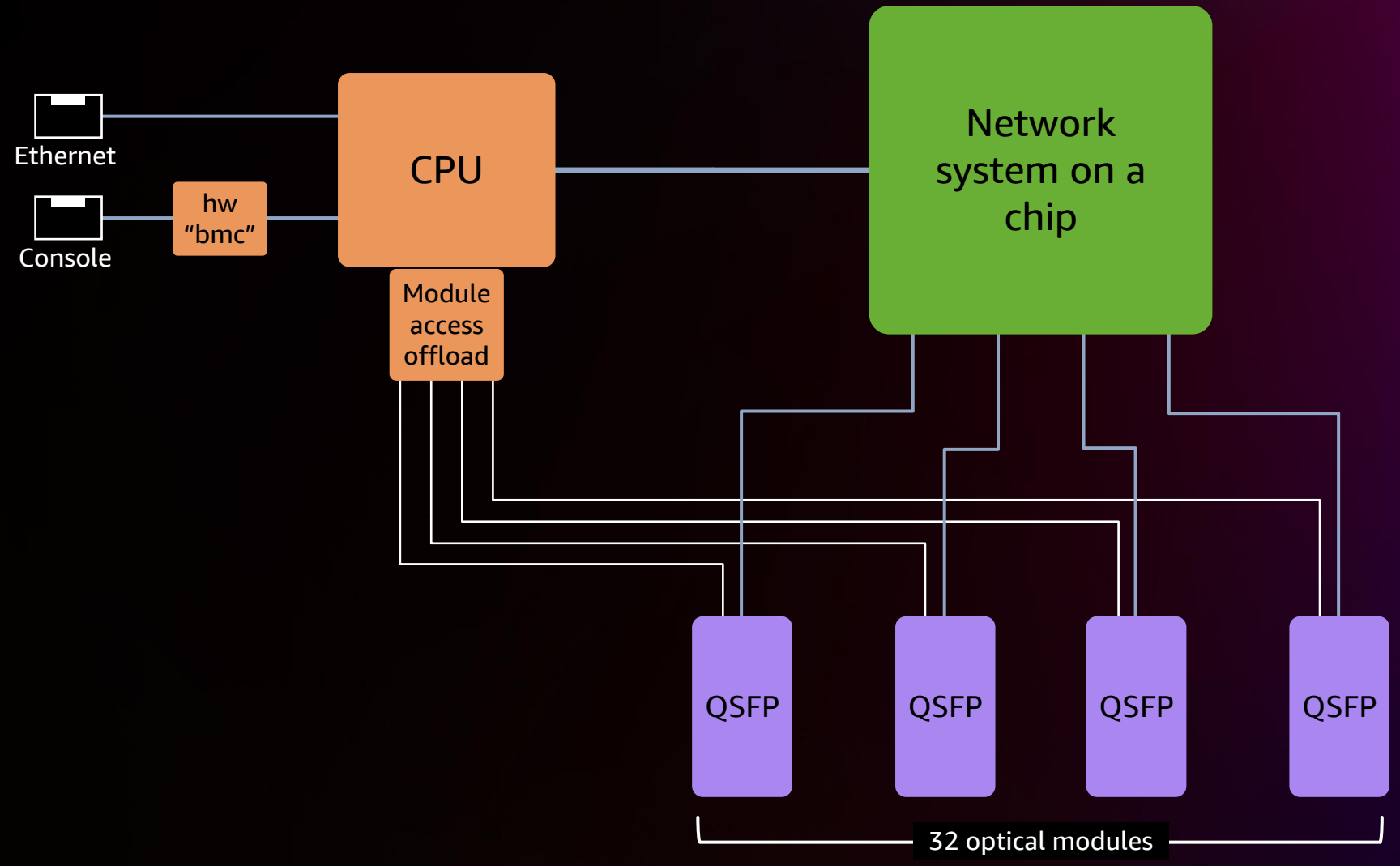
How we do it



How we do it

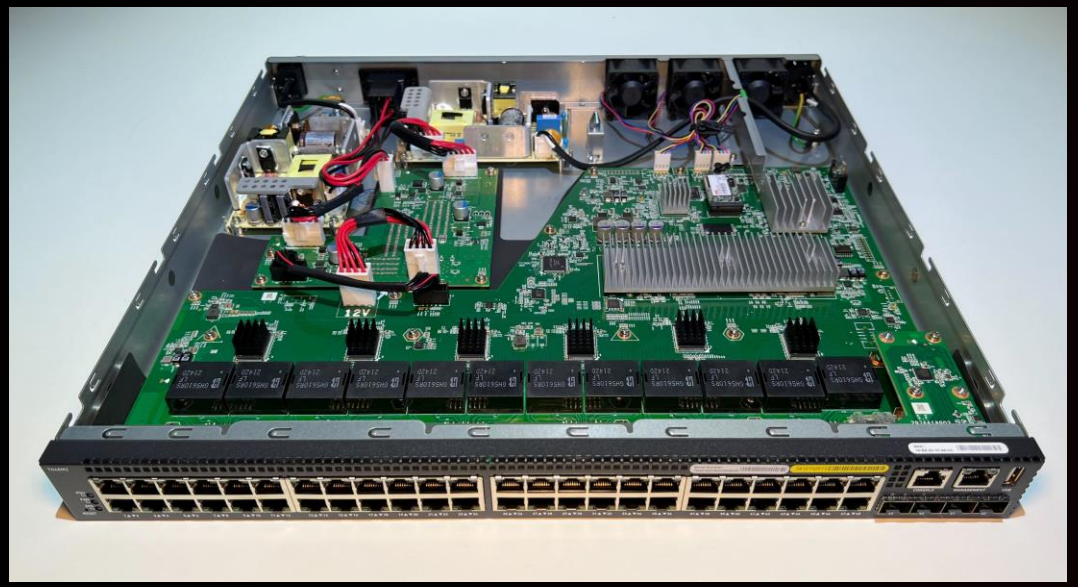


How we do it



How we do it

MANAGEMENT NETWORKS



Out-of-band switch

Console server

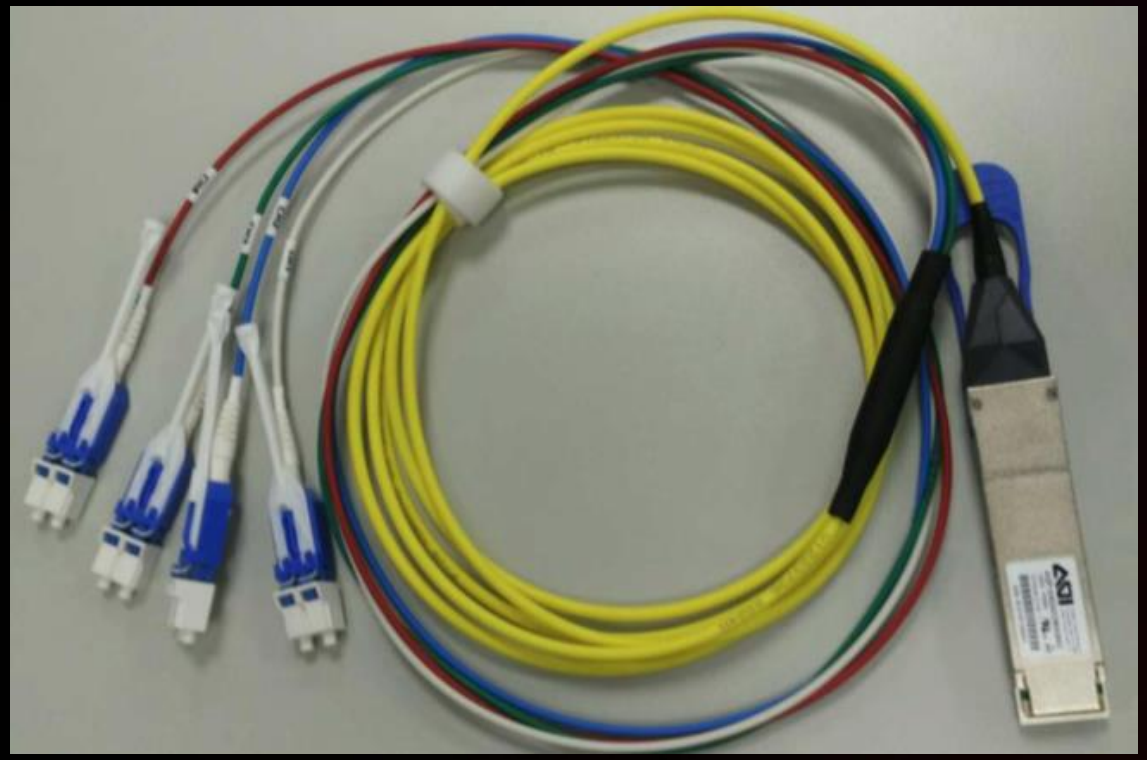
How we do it – In rack

Direct-attach copper (DAC) cabling

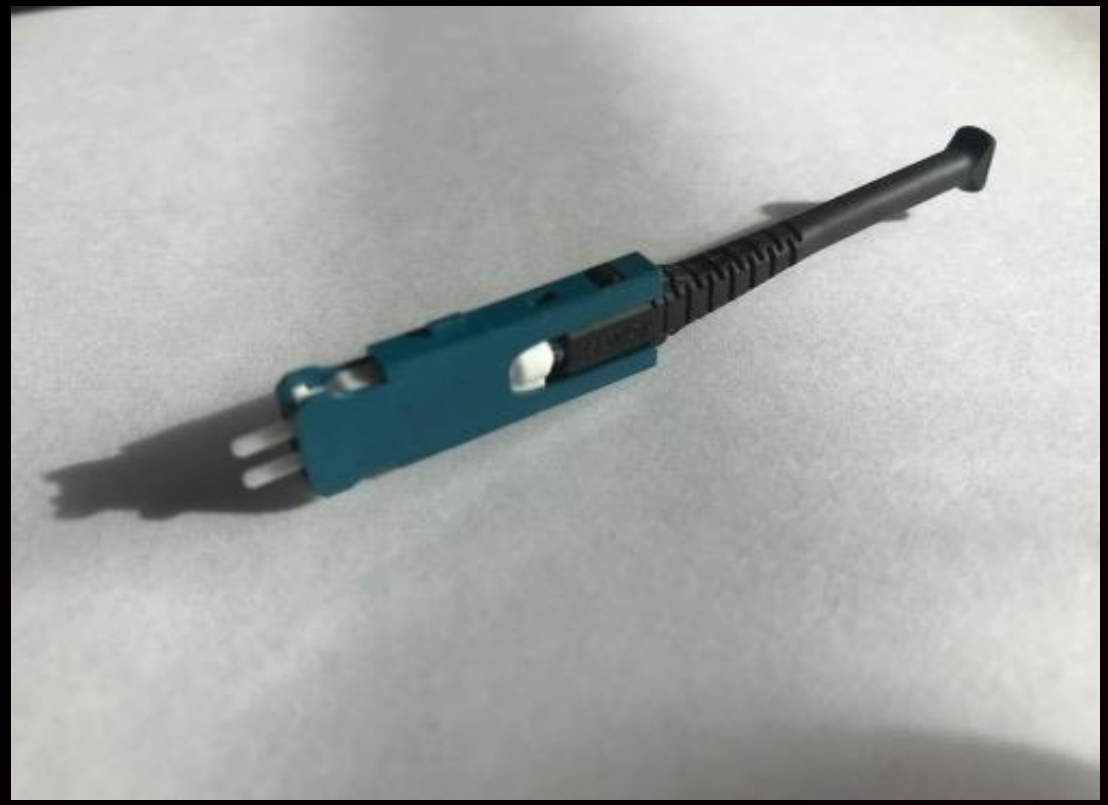
- 100G 6.7mm OD at 2.5m
- 400G 11mm OD at 2.5m

Active DAC with retimers

How we do it – Short reach



How we do it – SN connector



How we do it

MEDIUM HAUL

Data center interconnect (DCI)

OIF 400G ZR

400G – ZR+ 400km

Integrated routing, DWDM, encryption

Software

Metal boxes and a lot of cables

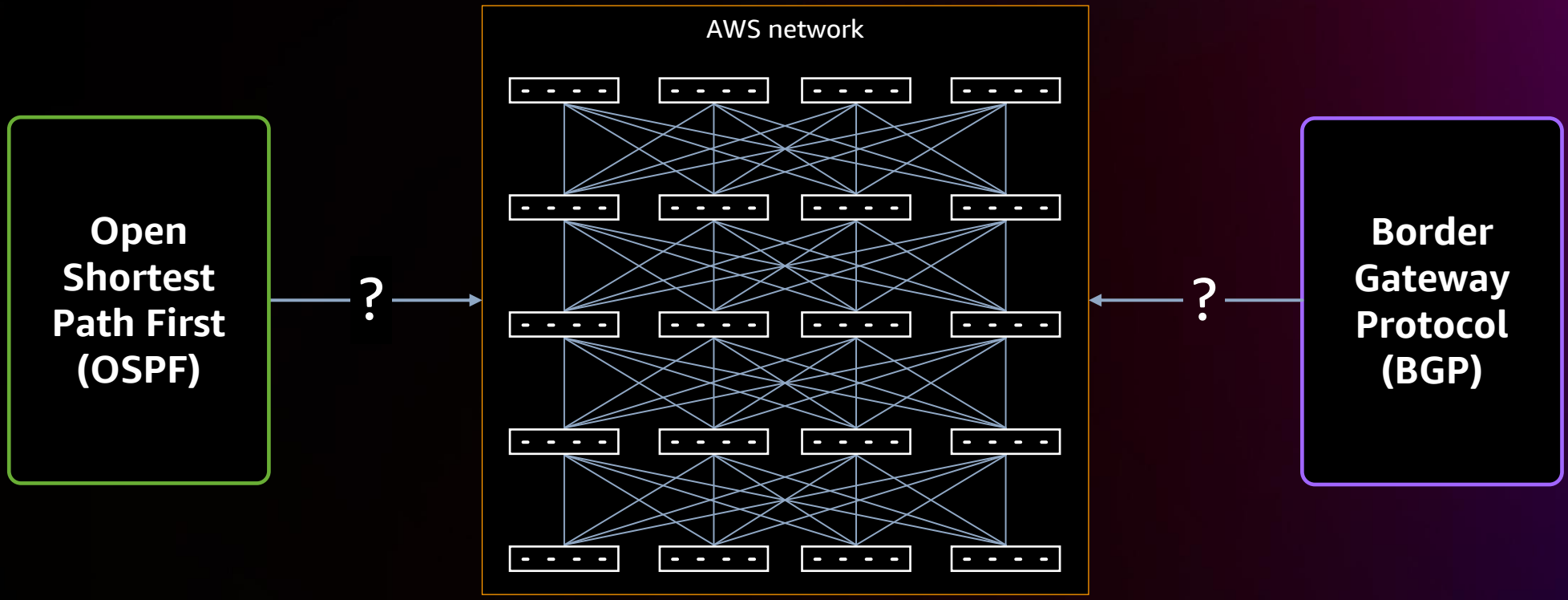
Small number of rack variations

Rack and cable switches for burn-in

Collect inventory and compare with bill of materials

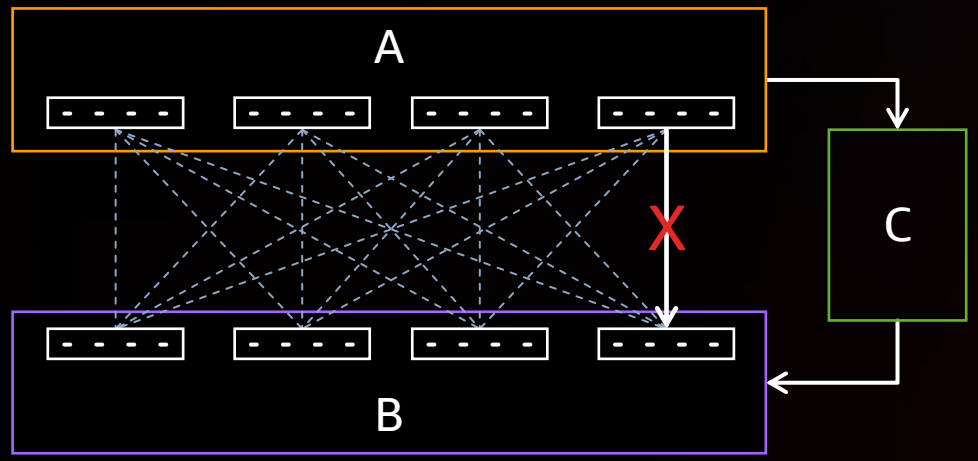
Reprogram with AWS controlled binaries

Which way do I go?

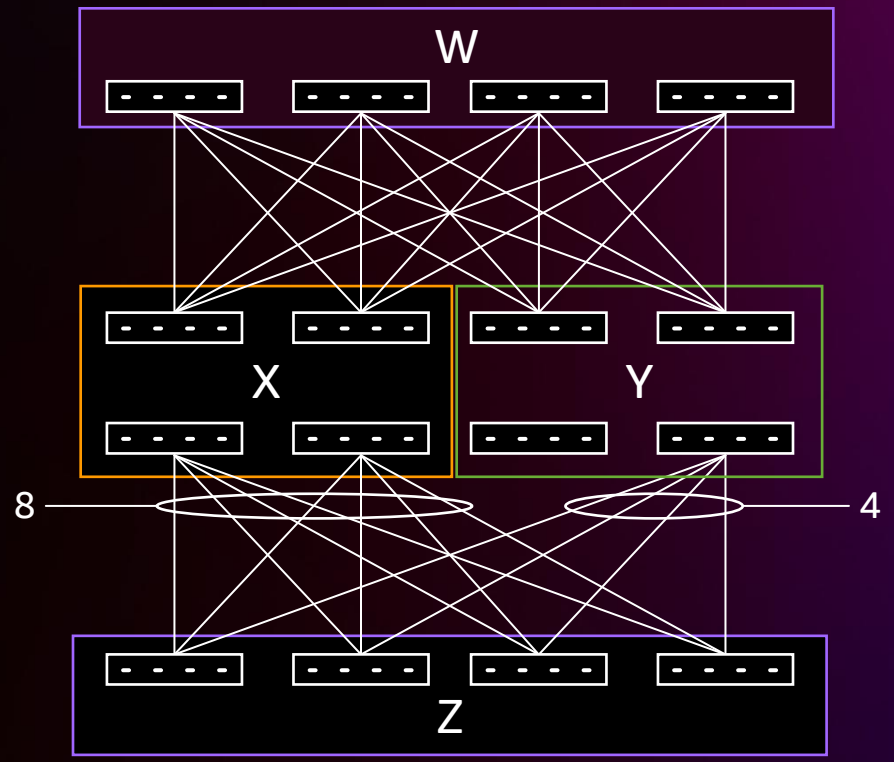


Which way do I go?

ISSUES WITH OFF-THE-SHELF PROTOCOLS



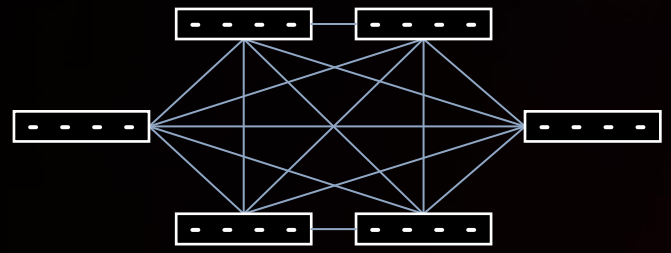
Last link standing



Cross-domain imbalance

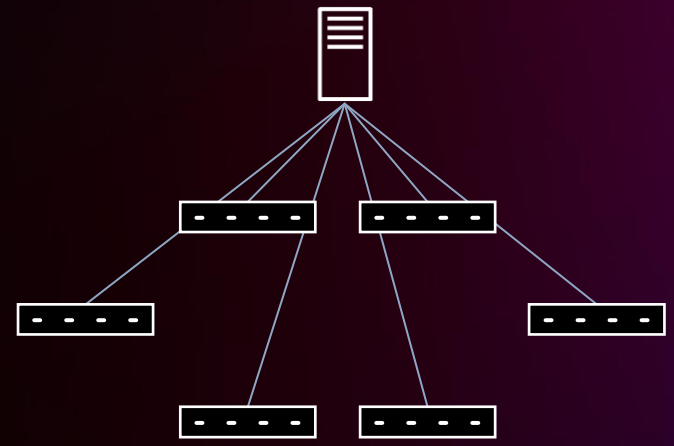
Which way do I go?

DISTRIBUTED VERSUS CENTRAL



Statically stable Low scope of impact

Distributed (classical)

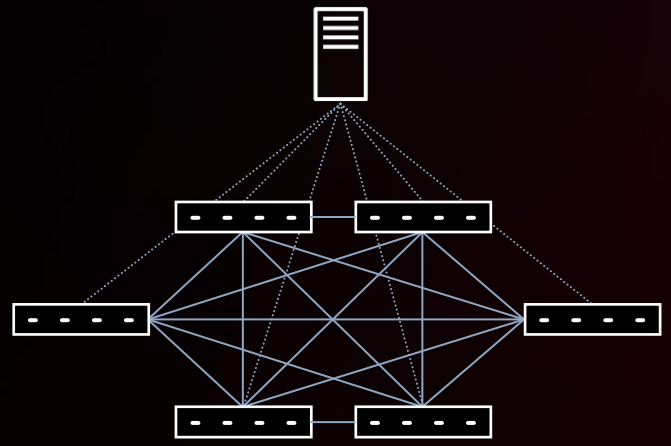


High visibility Deterministic

Centralized (SDN)

Which way do I go?

BEST OF BOTH WORLDS



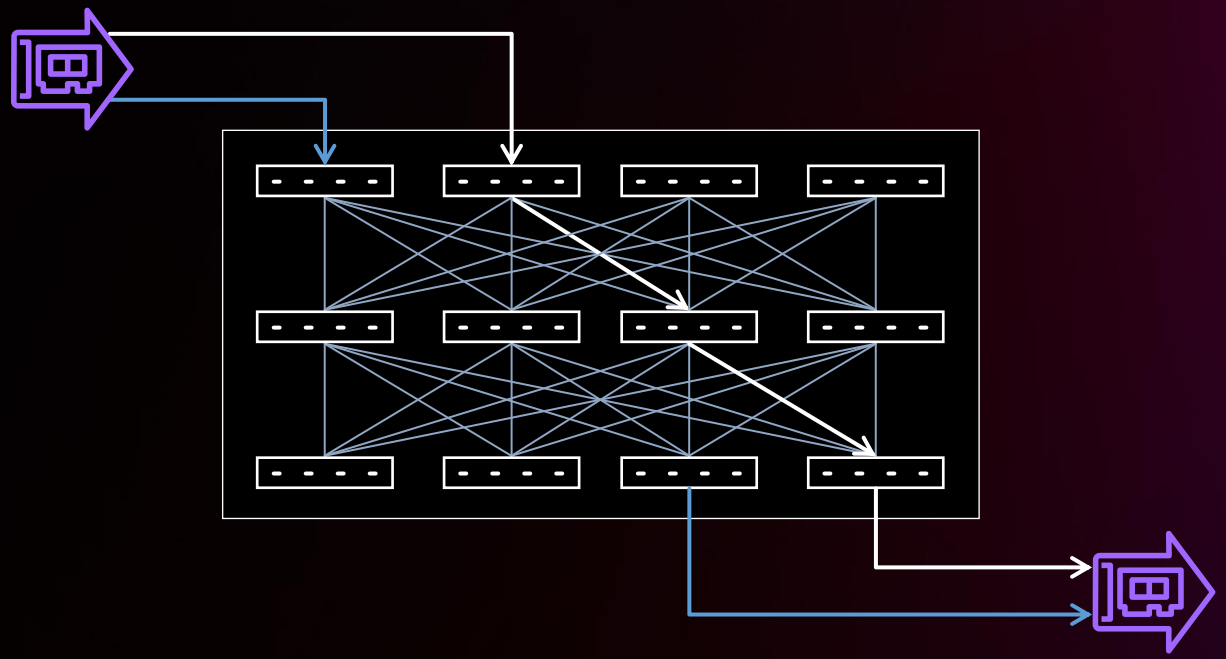
Statically stable Deterministic

Highly visible Low scope of impact

Hybrid

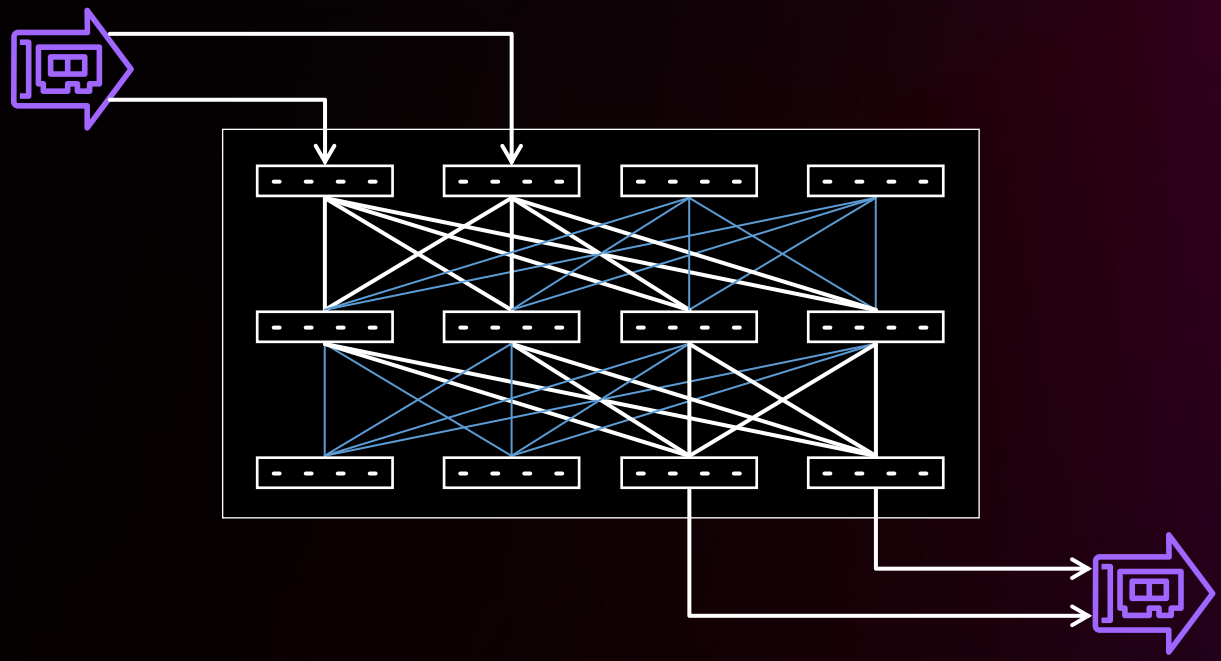
So many paths!

TRADITIONAL TCP BEHAVIOR



So many paths!

ELASTIC FABRIC ADAPTER (EFA) AND SCALABLE RELIABLE DATAGRAM



Dave Brown's Keynote
Session: NET211-L

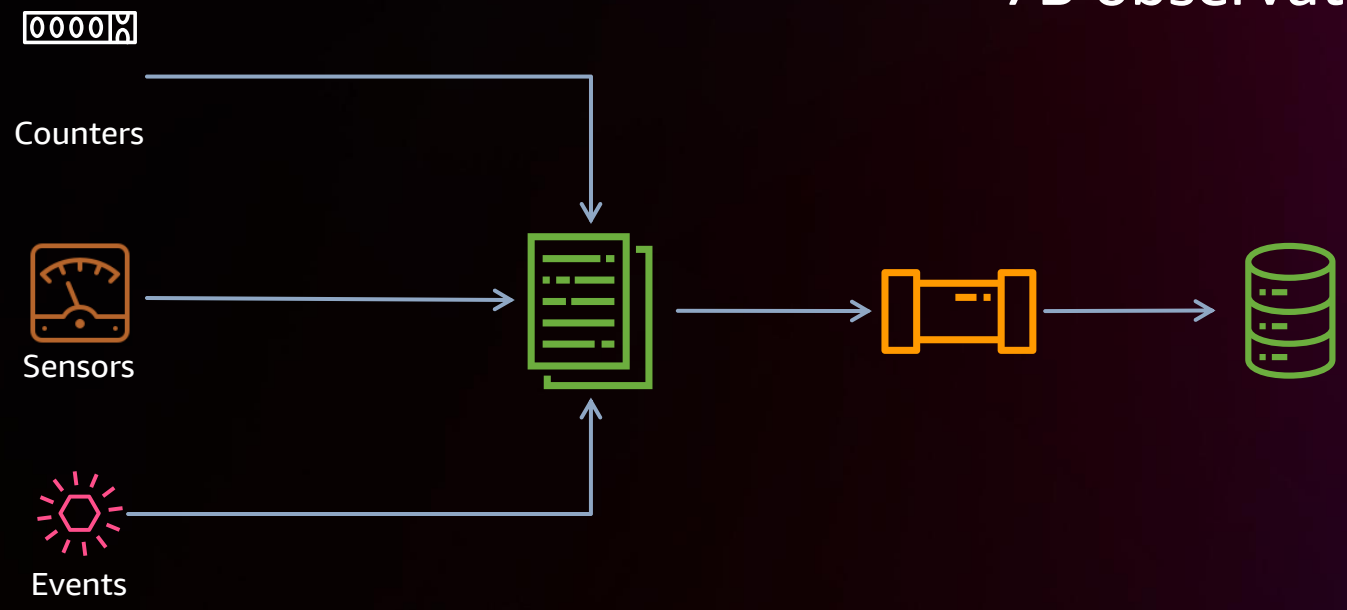
Monday Night Live with Peter DeSantis – 2018

Scaling HPC Applications on EC2 – 2018

Doctor, why does it hurt?

PASSIVE MONITORING

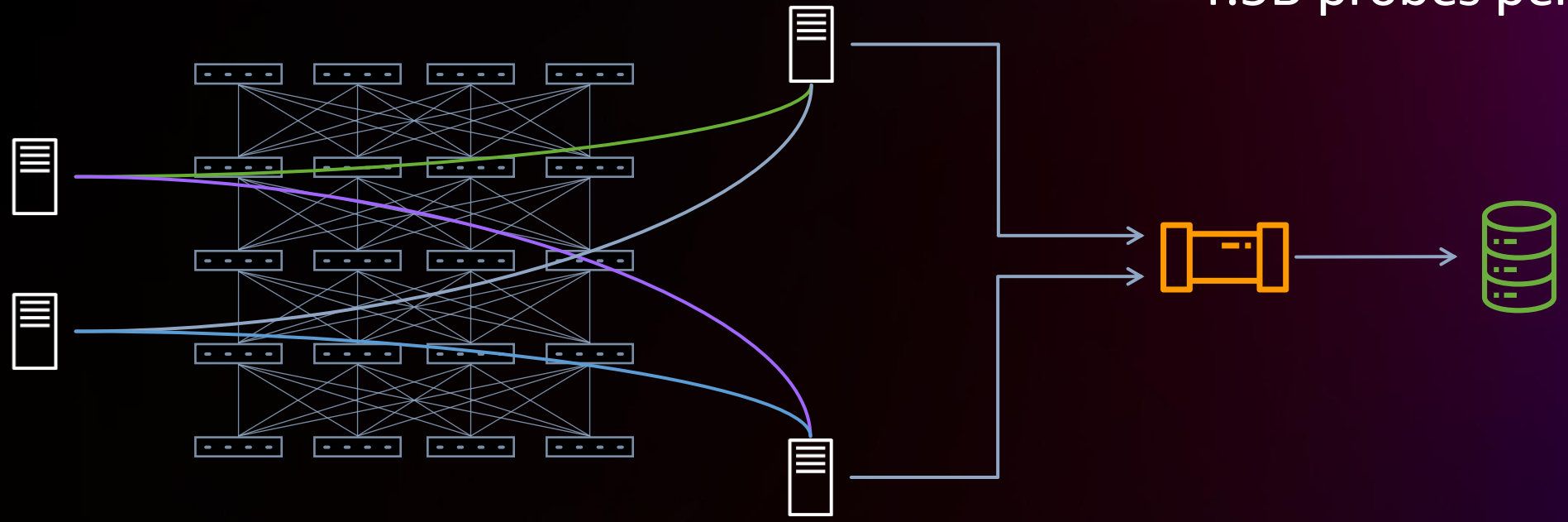
>7B observations per minute



Doctor, why does it hurt?

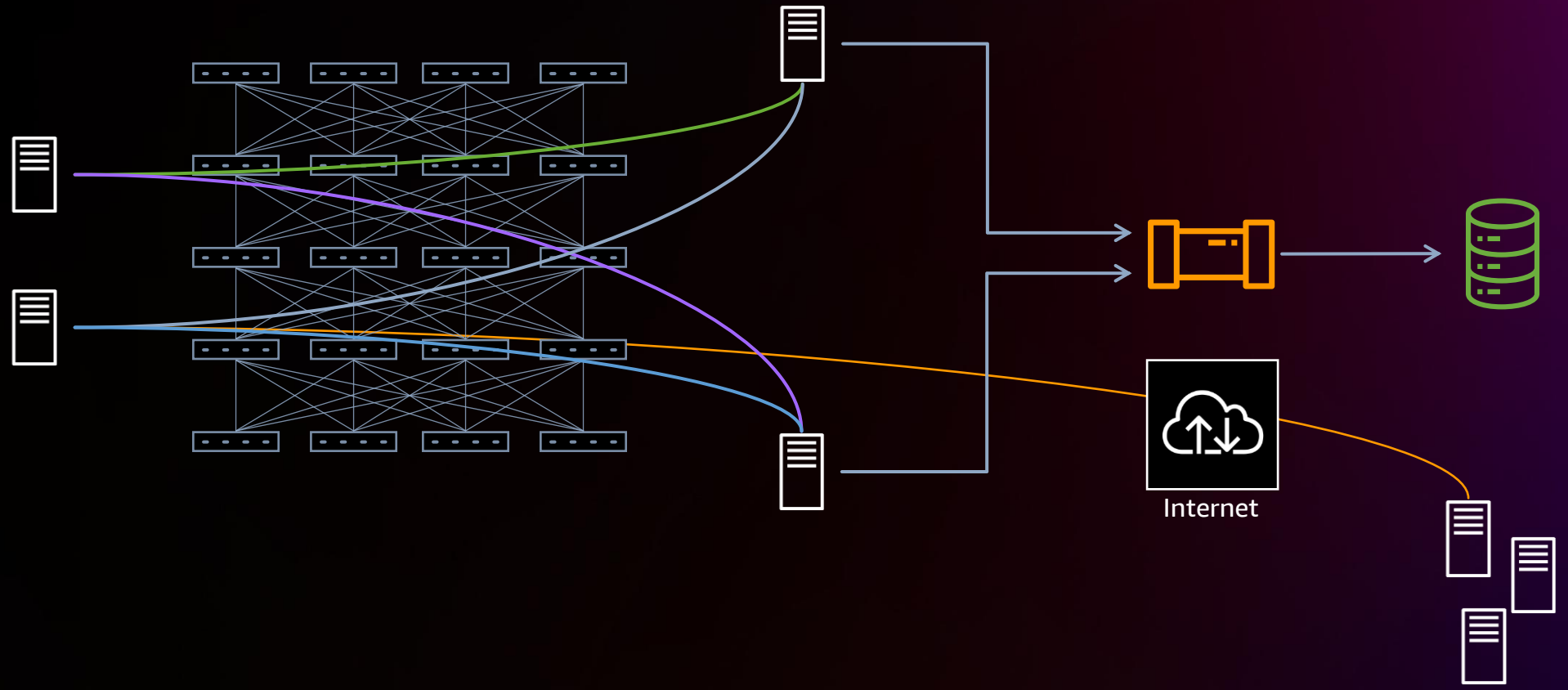
ACTIVE MONITORING

>1.5B probes per minute



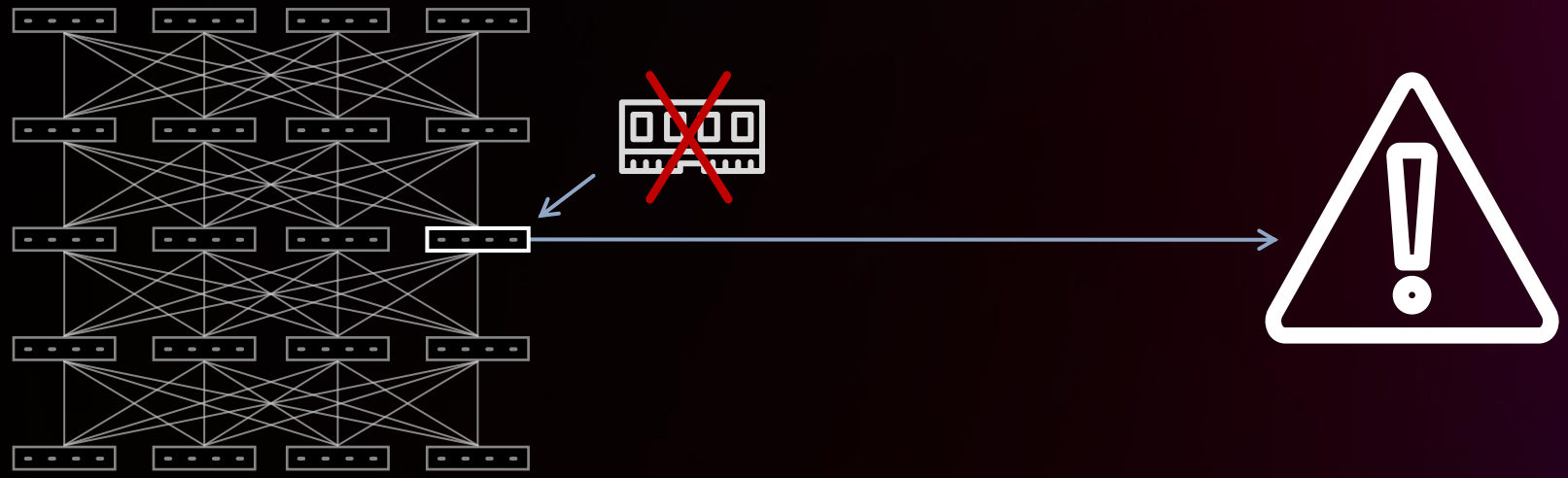
Doctor, why does it hurt?

ACTIVE MONITORING



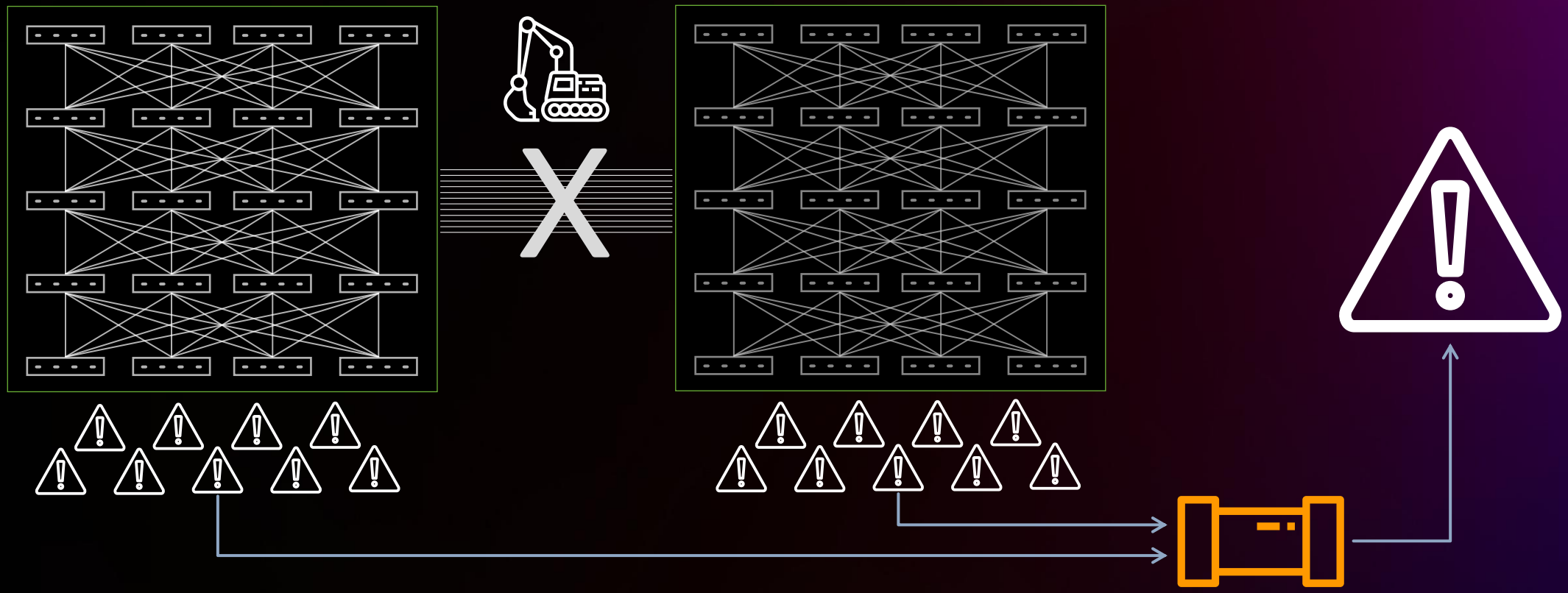
Doctor, why does it hurt?

CLEAR SIGNAL



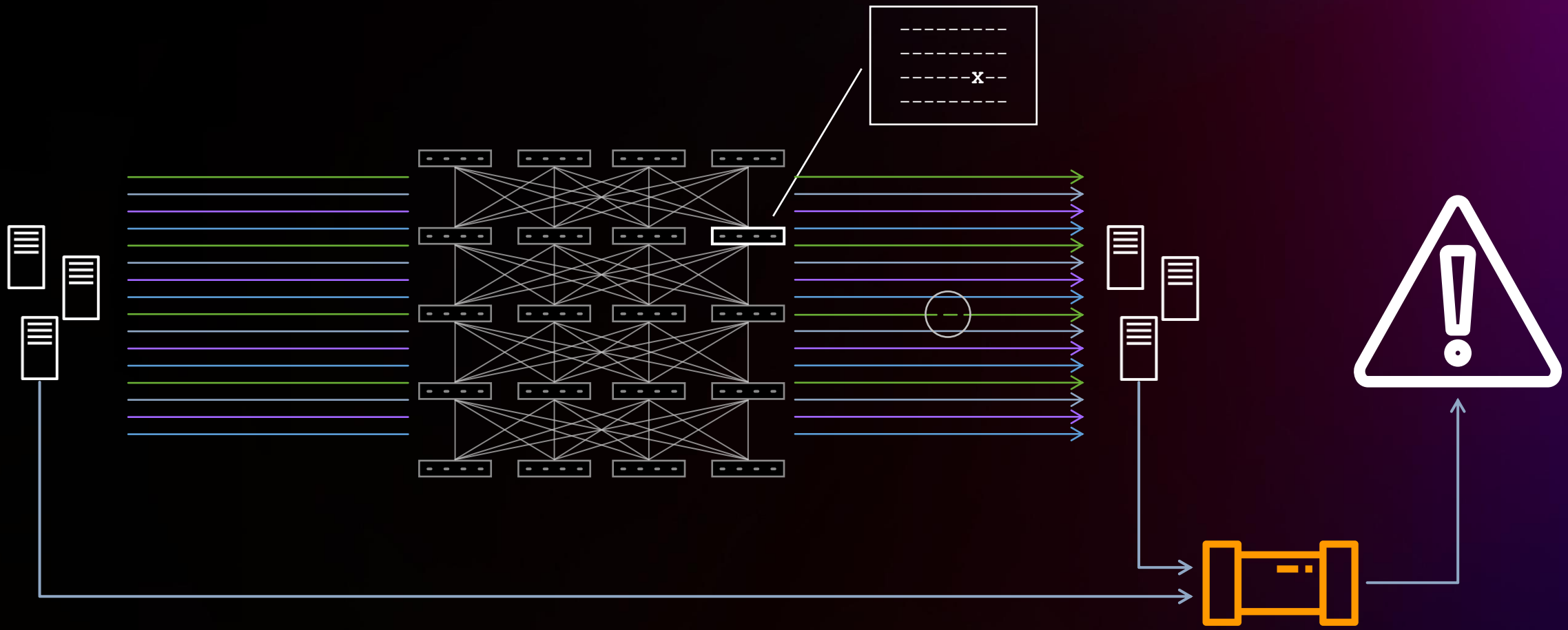
Doctor, why does it hurt?

CORRELATION



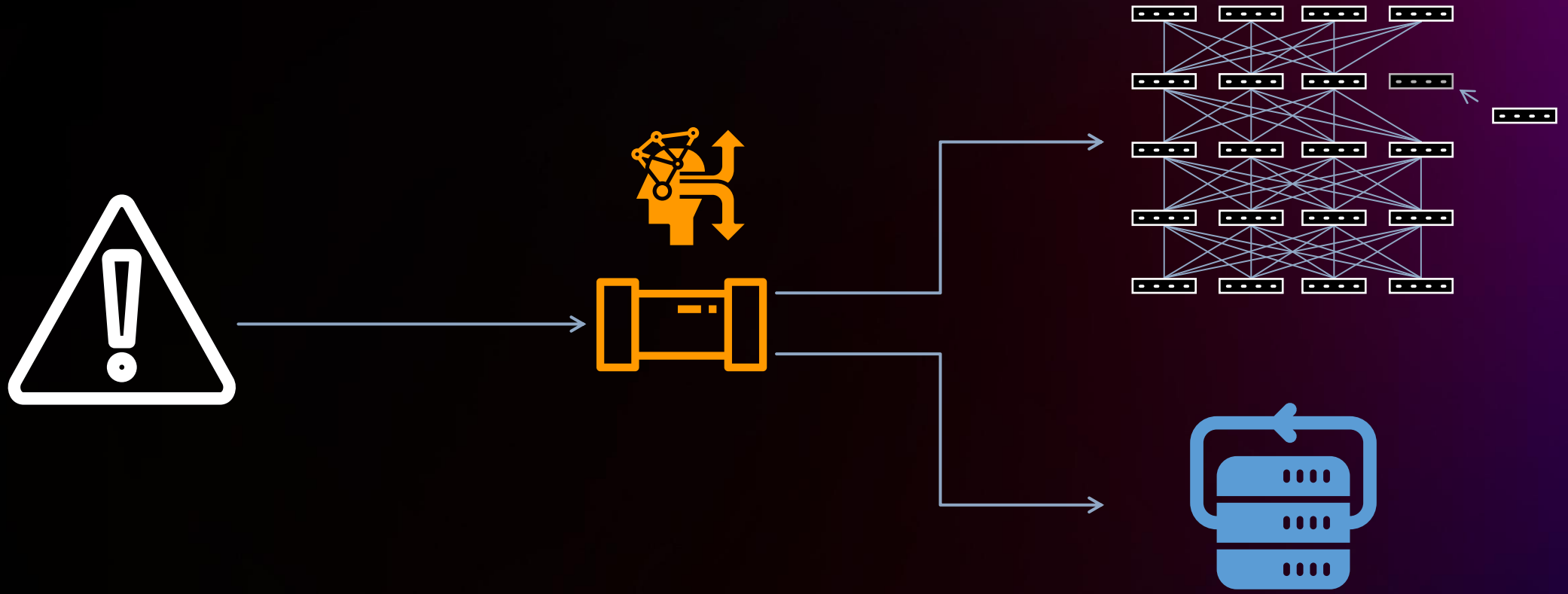
Doctor, why does it hurt?

TRIANGULATION



Ahhh . . . That’s better

AUTO-REMEDIATION

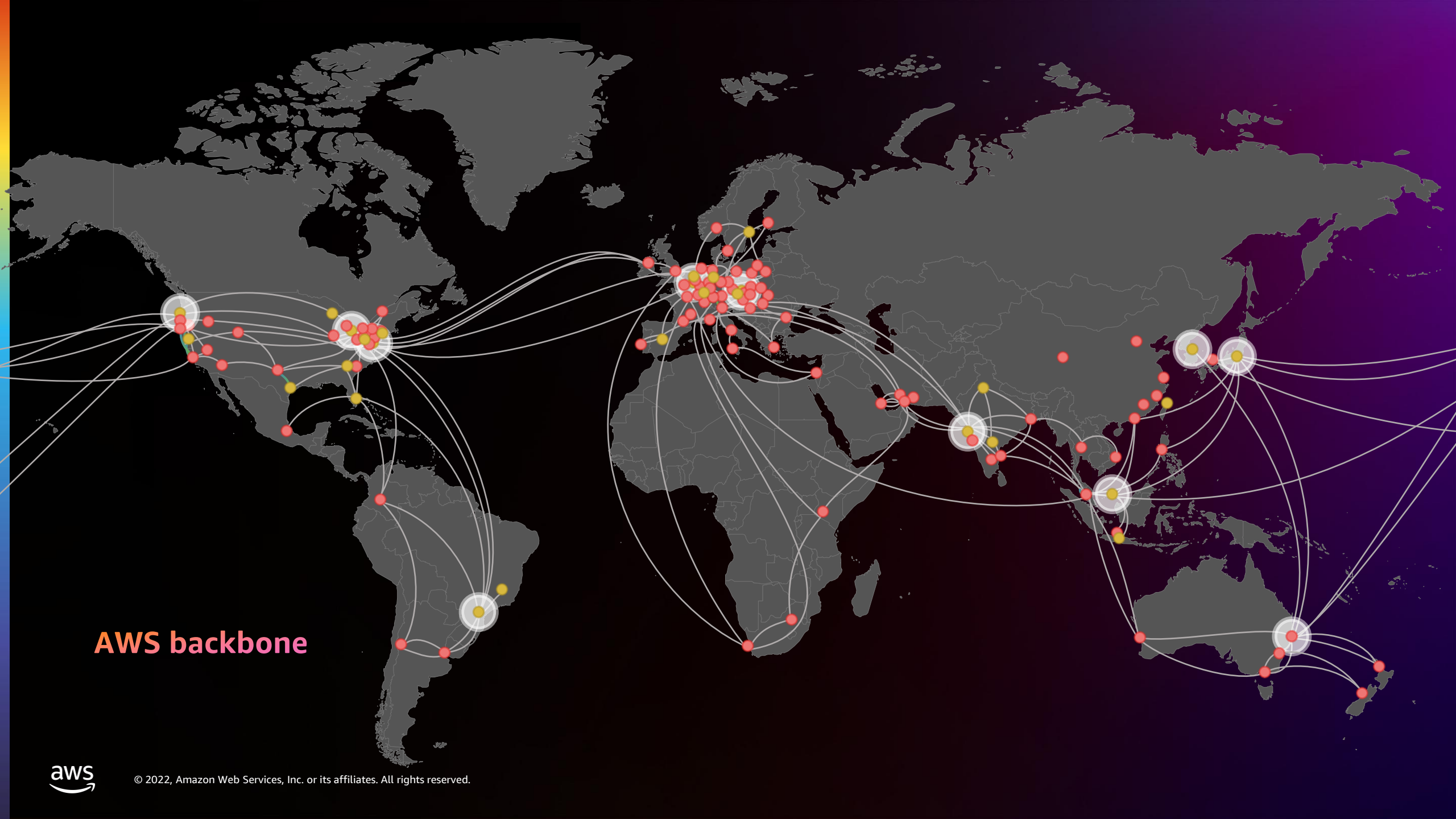


Layered control

Local for speed

Central for optimization

Hierarchical abstractions



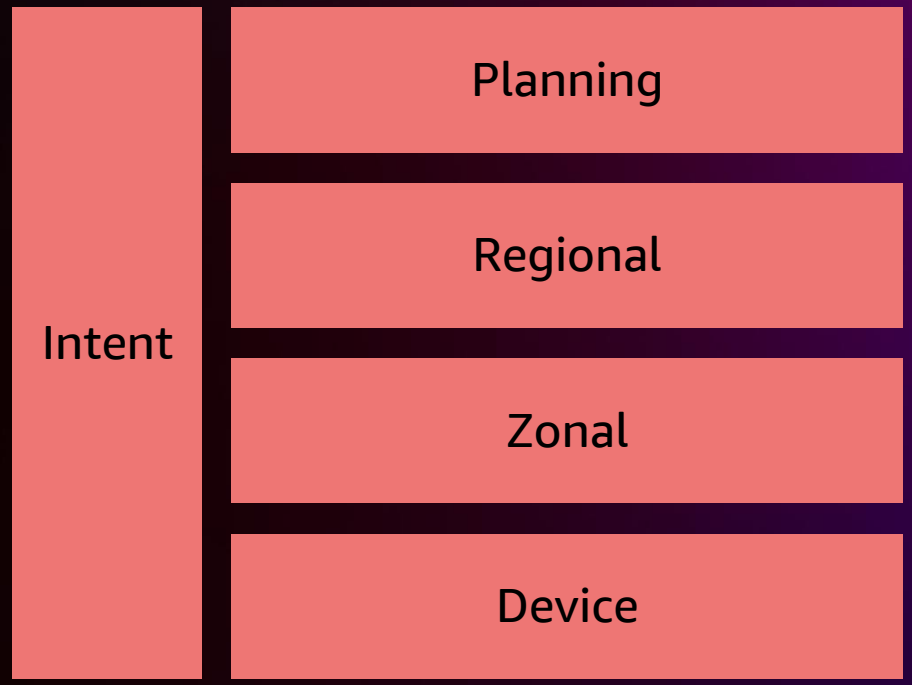
Future: Intentful management

Expected behaviors

Hierarchical

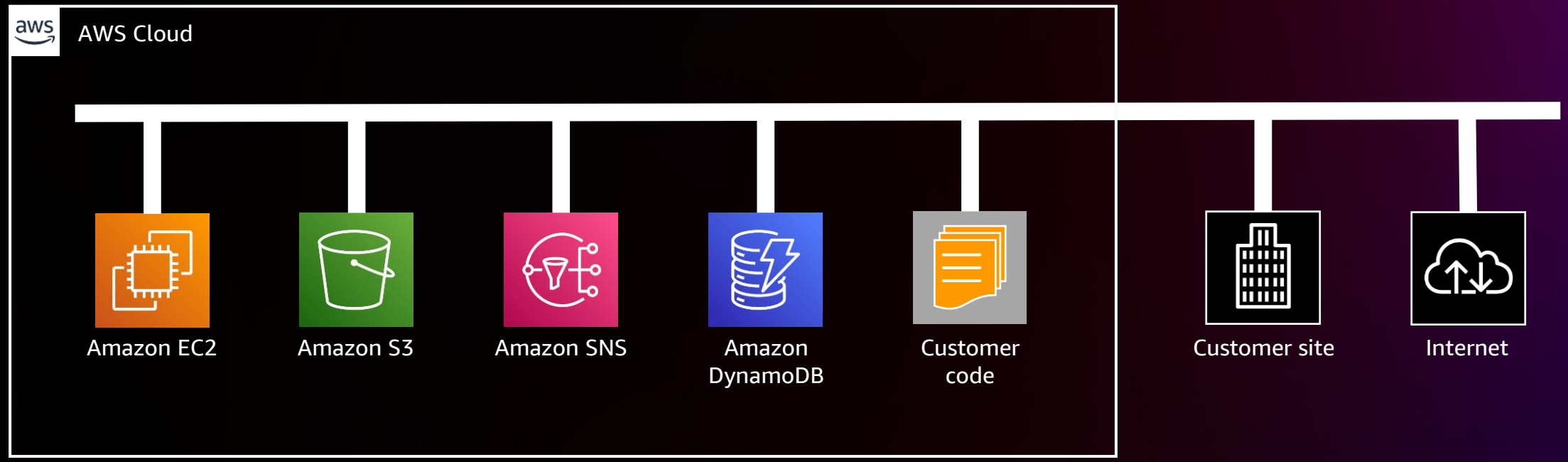
Multi-domain

Closed loop



Thank you!

Secure Available Scalable Performant



Thank you!

Stephen Callaghan
stephcal@amazon.com

JR Rivers
jrizzle@amazon.com

Please complete the session survey in the **mobile app**