Cloud Repatriation

- Next Year's Buzzword

- Nathan Moore
- Principal Architect, Inertia Labs

Our Story So Far…

- A Brief History of Data Centers
  - ENIAC in 1945
    - University of Pennsylvania, required a dedicated building
  - Computer Rooms
    - 1960's dedicated rooms in existing buildings to house transistorized computers
  - Network Closet
    - 1980's Rise of client-server architecture, server housed in network closet and attached to office network
  - Data Center
    - 1990's Internet connectivity and dedicated servers in centralized facility, back to a shared, dedicated building
  - Cloud
    - 2000's Virtualized services delivered from shared physical servers
  - Hyperscalers
    - 2010's Massive build-out of physical infrastructure to support expanded Cloud offerings
  - Infrastructure-as-a-Service
    - 2020's The entire physical stack is now programmatically available with multiple providers at every level

Growth has been driven by the quest to find economies of scale and economies of scope in Compute

https://blog.enconnex.com/data-center-history-and-evolution

- A Brief History of Cloud
  - Named in 1994
  - VMWare virtualization in 1999
  - AWS EC2 in 2006
  - OpenStack in 2010
  - Terraform in 2014
  - Kubernetes in 2014
  - *-as-a-Service
    - API-driven services to be interconnected via common protocols

  Layers of abstraction mean physical infrastructure is well hidden

https://en.wikipedia.org/wiki/History_of_cloud_computing

How did we get here?

- Cloud has been dominant for over a decade
  - Cloud-only apps and services have been built and run and grown successfully for years
  - There are plenty of companies with no expertise in anything behind the Cloud provider's API
  - Infrastructure-as-a-Service has become "Service," and Infrastructure is a low class, dirty word Software does not speak

There are lots of organizations which have either lost or never had expertise in systems and networks

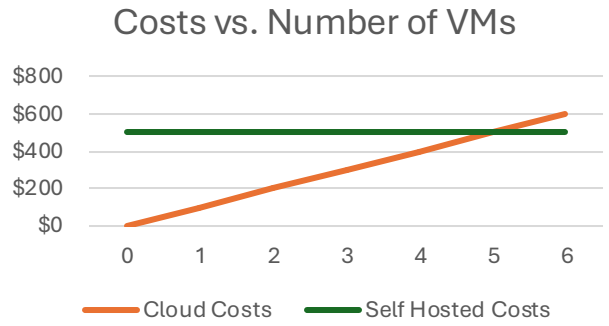
So what's the problem?

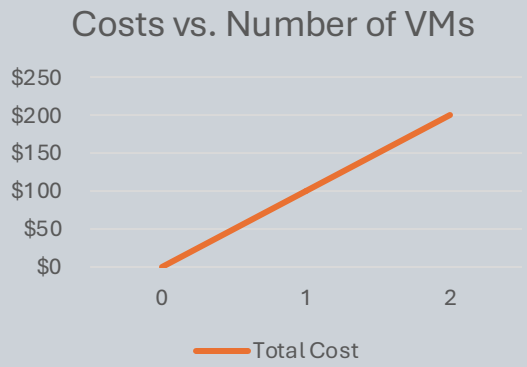
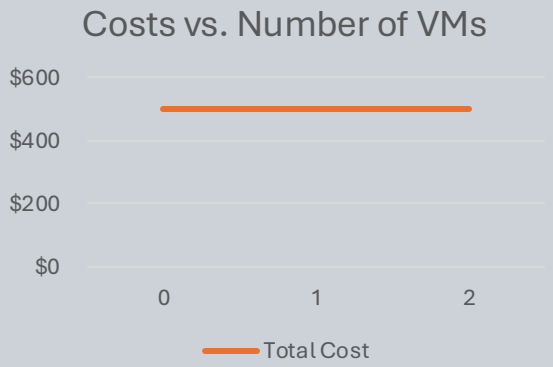
- Cloud-native companies are discovering problems with
  - Cost
  - Control
  - Performance
  - Compliance
  - Scope and Size of Service

Cloud-native means Cloud-dependent

Cost

- Variable vs. Fixed costs



Cloud	Self hosted
Costs are entirely variable, and increase linearly with usage. A second VM doubles the cost of a single VM. 	Costs are fixed up to a given server. The added cost of running a second VM on top of an existing server is roughly $0. 

Control

GCP auto-deleted the entire account of a $125Bn pension fund

- https://www.theguardian.com/australia-news/article/2024/may/09/unisuper-google-cloud-issue-account-access
  - "UniSuper has approximately $125bn in funds under management."
- https://cloud.google.com/blog/products/infrastructure/details-of-google-cloud-gcve-incident
  - "... there was an inadvertent misconfiguration of the GCVE service by Google operators due to leaving a parameter blank. This had the unintended and then unknown consequence of defaulting the customer's GCVE Private Cloud to a fixed term, with automatic deletion at the end of that period."

Bad things happen- who should bear the risk? At small scale, outsourcing risk is acceptable because the cost of a handful of angry customers is relatively small. At large scale, a formal SLA will bound risk to the company offering the service.

Performance

- VM contention
- Network contention
- API access contention
  - Rate limits
- Data Gravity
  - Egress is expensive, Ingress is free

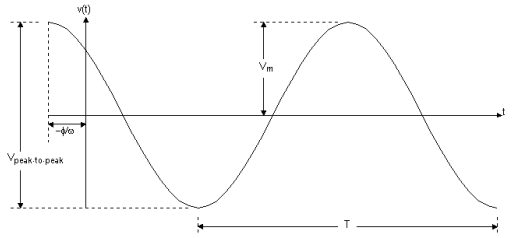
Compliance

- PCI
- CCPA
- GDPR

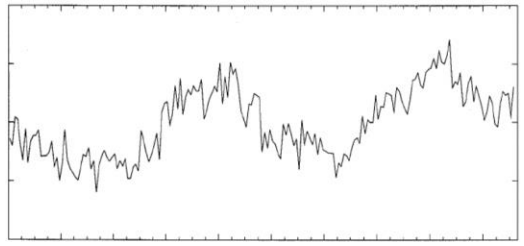
Even if your cloud vendor claims compliance, can you enforce it?

Size and Scope of Service

- Optimizing for base load



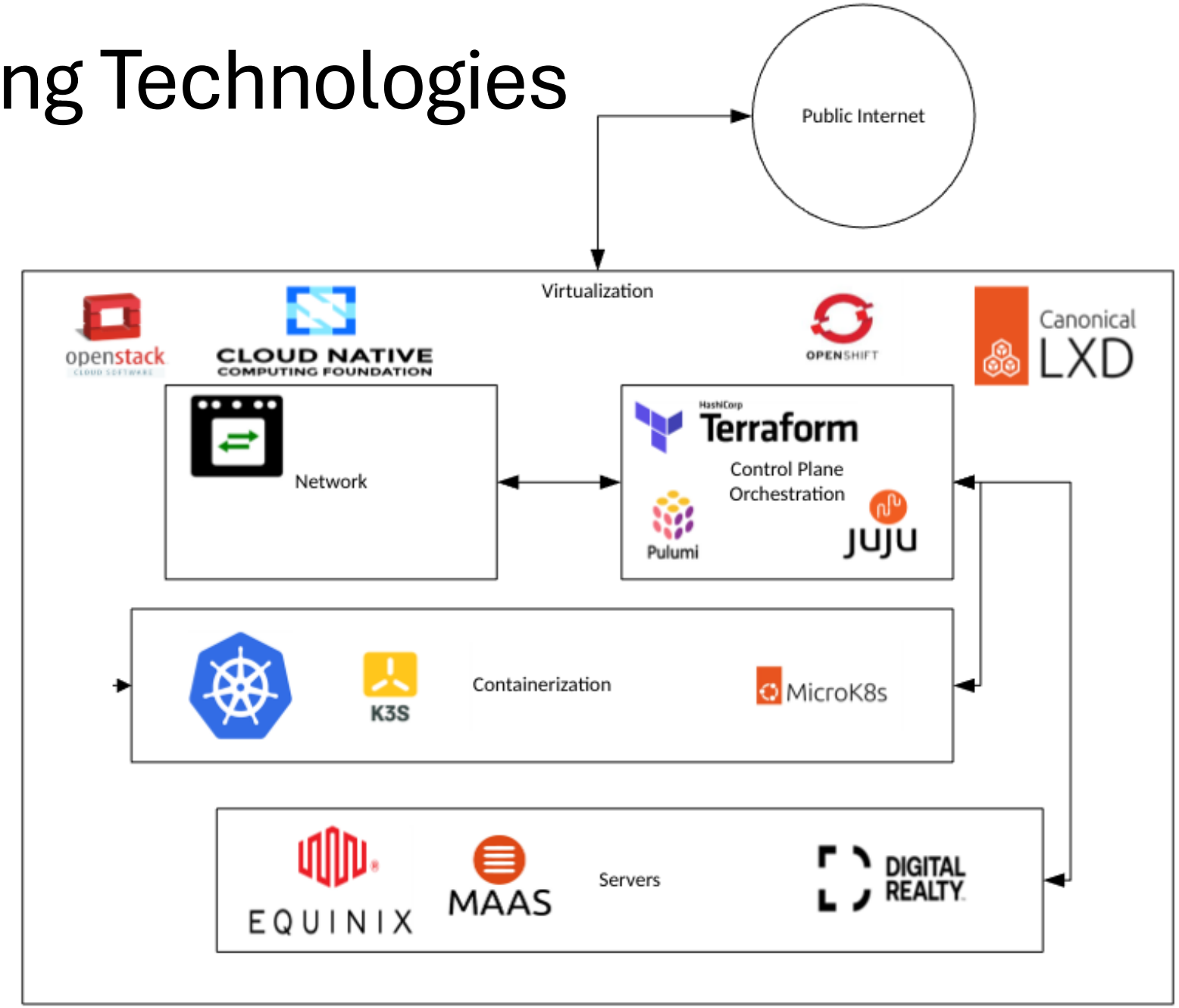
- Optimizing for individual projects



Repatriate Back from Cloud to Hybrid

- Negative externalities of being Cloud-only are becoming more and more evident
- Repatriation does not mean no more cloud, instead it means evolving into an intelligent Hybrid
  - Cloud where it makes sense
    - New projects, dev work, scaling risk, business continuity
  - Self hosting where it makes sense
    - Known large scale, large expected traffic, older projects where scope is known, batch processing
  - Anything in-between
    - Shop specific custom knowledge

Enabling Technologies



Build the Stack You Want

- If you like your cloud, you can keep it
  - Multiple projects which replicate a professional cloud
- Pick-and-choose the parts which work for you, and ignore what doesn't
- Customize based on your shop's requirements
- Optimize to deliver an ideal production environment

Vendors for Every Part of the Stack

- Getting the risk-reward balance right
  - What you want to do for yourself
    - e.g. Control your own Cloud
    - Control your own network circuits and bandwidth
  - What you want to outsource
    - Utilize a third party Metal-as-a-Service
      - Third party is responsible for physical hardware only

Optimization Strategies

- Identify workloads
  - Production
  - Staging
  - Development
- Identify gates and boundaries between workloads
- Identify inter-dependencies between workloads
- Identify failure modes
- Identify critical junctures for Observability

Optimization Strategies

- Self-host Production and leave everything else in the Cloud
  - If Production is responsible for the bulk of costs, self-hosting only Production minimizes costs with the least effort and least disruption
  - If Production growth forecasts are stable, capacity planning is reasonable and fine-tuning requirements results in minimizing otherwise-needed overcapacity

Optimization Strategies

- Workloads with inter-dependencies
  - Issues with race conditions or mutexes
    - Establish bulk-heads, timeouts, workarounds as appropriate
    - Utilize observability, monitor and measure known points of contention

Optimization Strategies

- Determine allowable failure modes
  - Business Continuity plan
  - Service Failover
  - Reliability and Consistency metrics

Optimization Strategies

- Observability
  - Self-hosting means Self-error-correcting
    - Identify errors
    - Quantify effects of errors
    - Trace errors through Workload's stack to root cause

- Cloud native means vulnerable to Cloud risks and Cloud costs
  - Hybrid helps alleviate effects of risks
  - Self-hosting even part of otherwise Cloud-based services reduces costs at scale
  - Bring in-house critical business needs and risks, and outsource everything else

Questions?