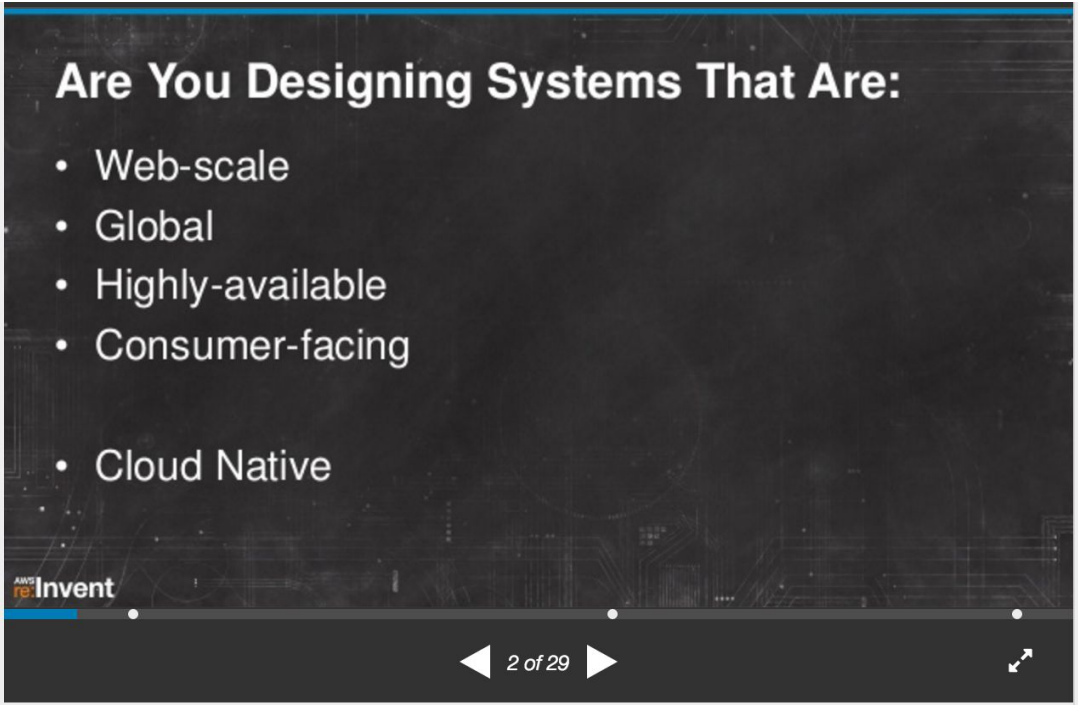
What is Cloud Native and why should I care?

Alexis Richardson
CNCF TOC Chair & CEO Weaveworks
23 Feb 2017

Open Source Cloud Computing *for Applications*

We curate & promote a trusted tool kit for modern architectures

Netflix pioneered the concept of cloud native as a practical tool

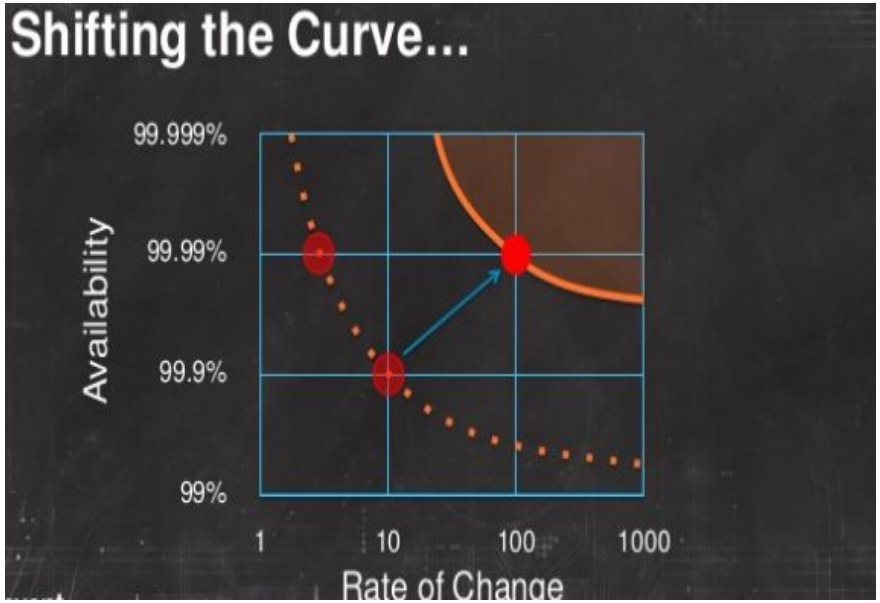


Netflix Development Patterns for Scale, Performance & Availability (DMG206) | AWS re:Invent 2013

1,590 views

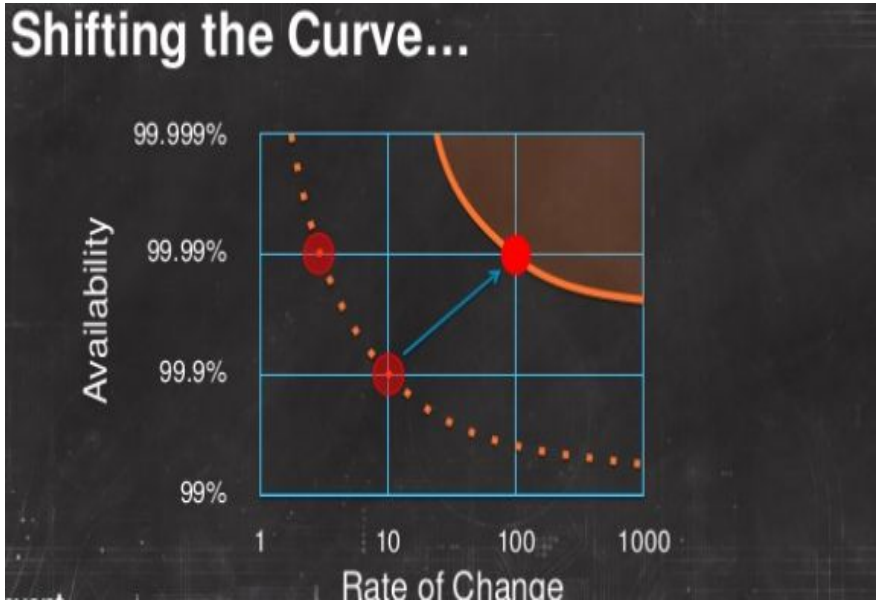
https://www.slideshare.net/AmazonWebServices/dmg206

Netflix wanted SPEED and ACCESS at SCALE



https://www.slideshare.net/AmazonWebServices/dmg206

Netflix wanted SPEED and ACCESS at SCALE



https://www.slideshare.net/AmazonWebServices/dmg206

Must Read!

Cloud native powers a whole industry today

So can anyone be like Netflix?

Maybe we can dream :-)

The *Need for Speed.* This is real.

Figure 1

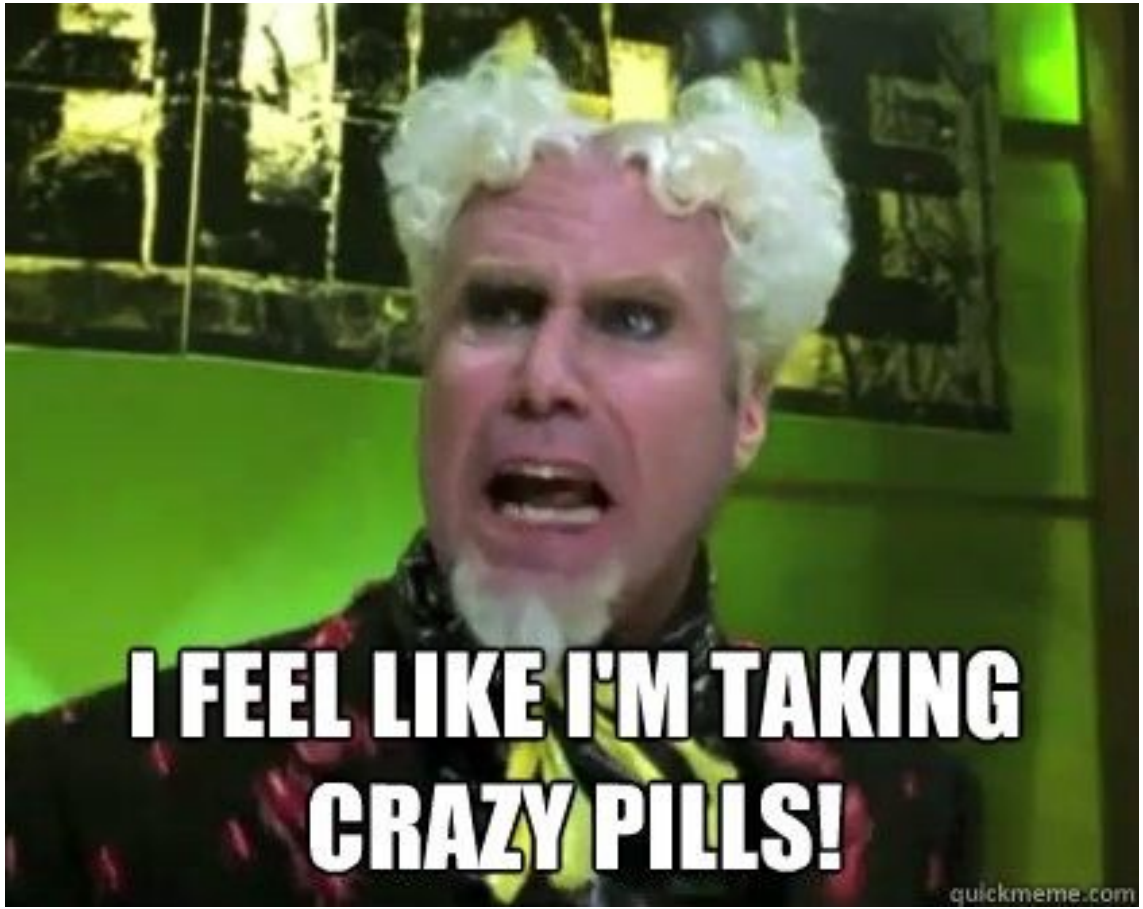
Comparison of IT performance metrics between high[1] and low performers

	2015 *(Super High vs. Low)*	2014 *(High vs. Low)*
Deployment Frequency	**30x**	**30x**
Deployment Lead Time	**200x**	**200x**
Mean Time to Recover (MTTR)	**168x**	**48x**
Change Success Rate	**60x**	**3x**

Puppet Labs state of devops 2015

So what is it about Cloud Native that helps me go fast

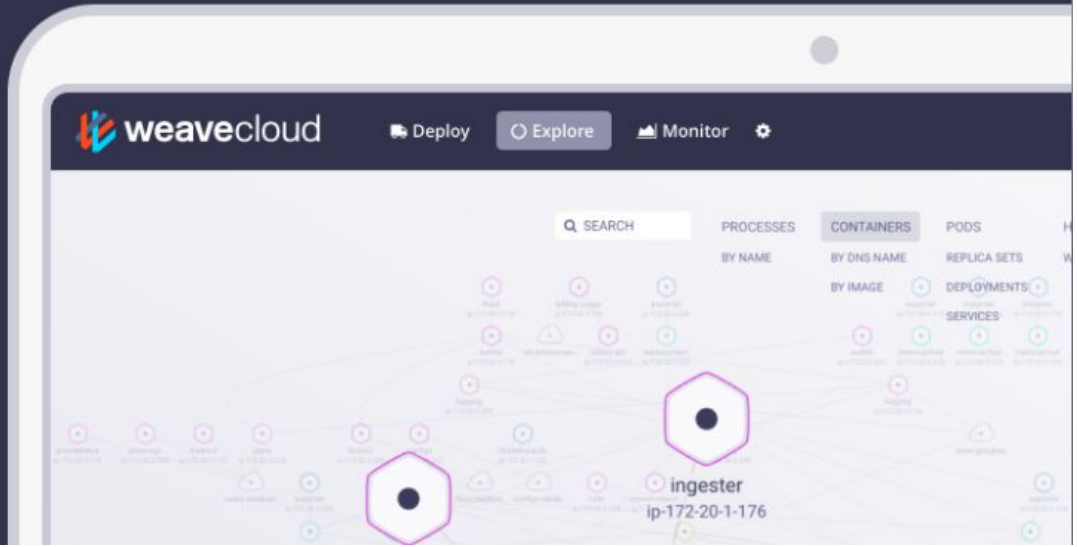
To learn more @Weaveworks we experimented on our own product



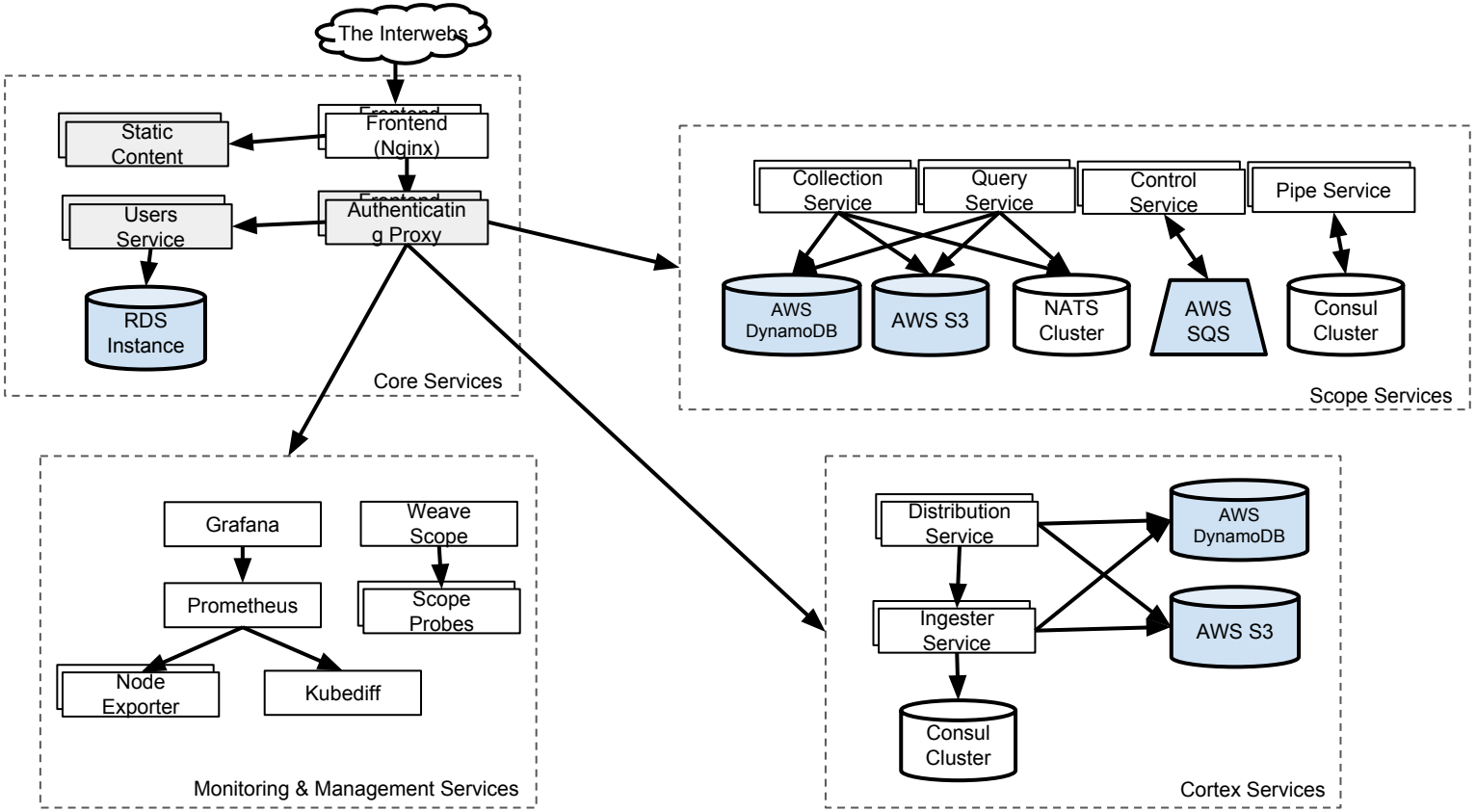
Simplifying delivery for **cloud-native** development

Built on open source software, Weave Cloud is software-as-a-service that allows DevOps teams to connect, monitor and deploy containers and microservices.

WATCH DEMO



Our requirements don’t fit one pattern, eg “*12 factor*”



Weave Cloud business requirements

24-7-365, Global, Multi-tenant, Secure…. etc.

Team focus 100% on rapid app development; not VM admin & plumbing.

We can scale components up/down in line with use/cost

Don’t spend money on wiring (Prometheus ‘just works’ with Docker, Kubernetes..)

We can run the Weave Cloud app anywhere (open source & not only Amazon)

Our solution learnings: what mattered most to us?

1) Automation. Lots of automation. End to end. Automate all the things.
→ **CI/CD! Orchestration! Observability!**

2) You need to focus on the app not the infrastructure, e.g. using standard packaging that Just Works consistently anywhere. → **Containers!**

3) You need to understand and apply new cloud native patterns and tools for monitoring, logging, uptime management & more → **Microservices & beyond!**

The ABCDE of Automation

App is developed & tested locally

Built automatically using CI of our choice

Container image pushed automatically

Deployed automatically using Weave Cloud deploy service...

...to an **Execution Environment** of your choice

Source control

Configuration as code

Lesson Learnt: *Cloud Native needs good tools*

Open source

Run anywhere

Software you can trust, managed by credible teams & processes

Easy to monitor and control

Interoperates with other tools and common conventions

Lesson Learnt: *the infrastructure has to be boring*

To focus on your app, the infrastructure has to be boring.

Use containers.

Use PaaS/CaaS or any container platform you like.

Watch out for the 1% failure problem

Lesson Learnt: *We need good PATTERNS*

Microservices (and Microliths)

Cattle not Pets

Observability and Control baked in

Traffic Patterns - Blue/Green, Canary, smart routing & load balancing…

…

Cloud Native is **Patterns**


#ohtechta

pause

Open Source Cloud Computing *for Applications*

We curate & promote a trusted tool kit for modern architectures

Cloud Native is **Patterns**

CNCF is & **Tools**
& Education

Patterns for what?

→ avoiding the pain by learning from others :-)

Availability

Automation

Acceleration

Anywhere!

Patterns for what?

→ avoiding the pain by learning from others :-)

Availability	Microservices & Netflix for everyone
Automation	Deployment & Management
Acceleration	CI/CD & “the ABCDE” of automation
Anywhere!	Containers are portable

Patterns need **Software**

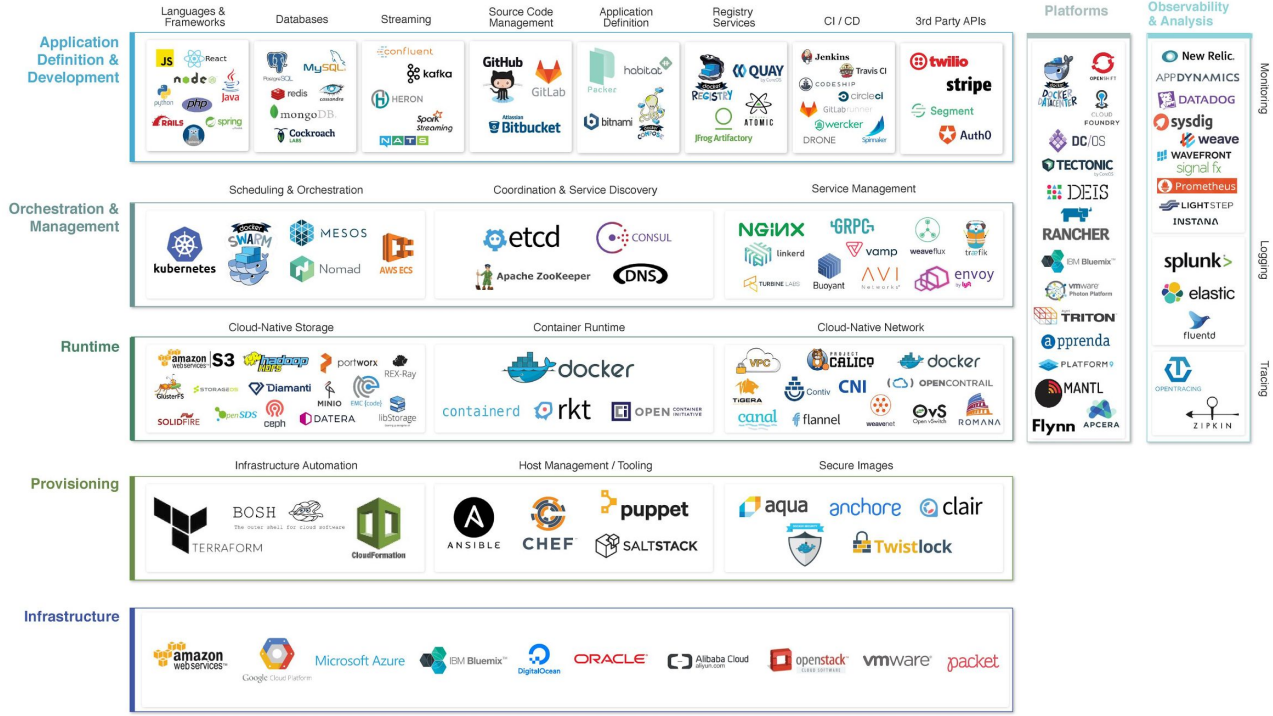
Tools you can trust

High quality, high velocity projects in CNCF today

- Kubernetes - container orchestration
- Prometheus - monitoring & analysis
- Fluentd - log forwarding
- OpenTracing - tracing interop
- Linkerd - traffic management
- Voting on: gRPC, CoreDNS
- More to come...

Cloud Native Landscape v0.9.3

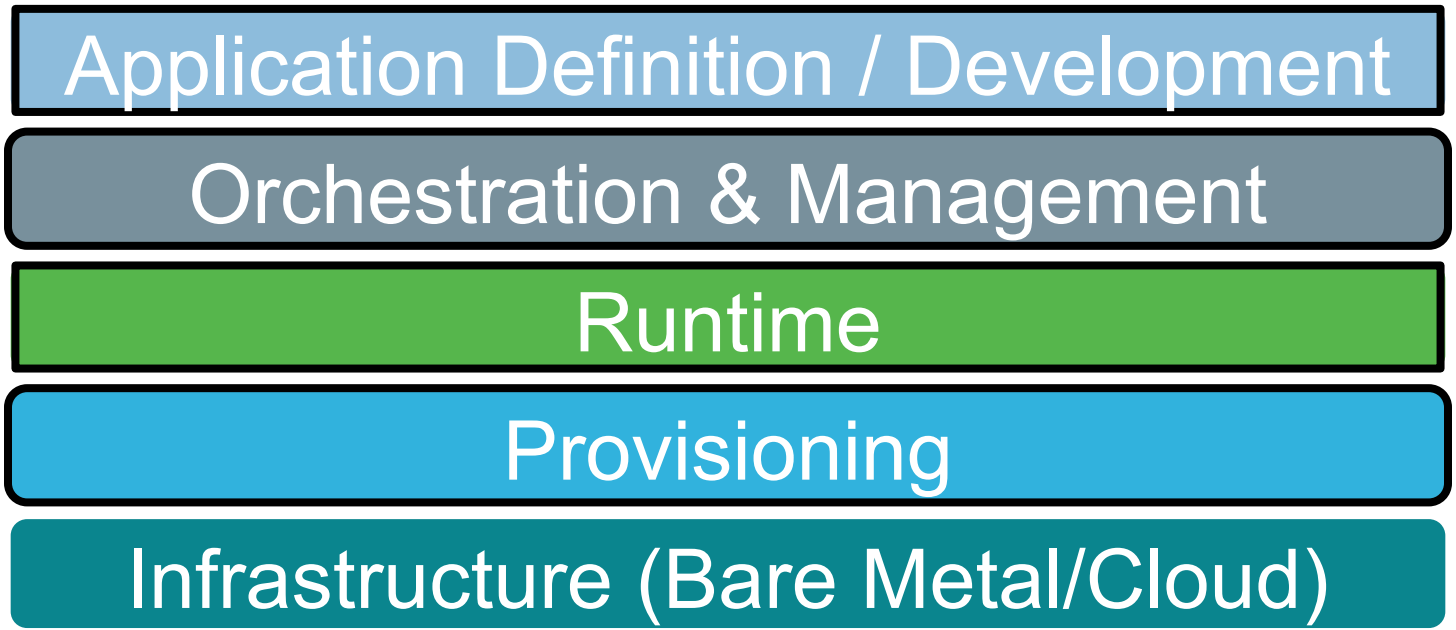
Redpoint | Amplify Partners | Cloud Native Computing Foundation



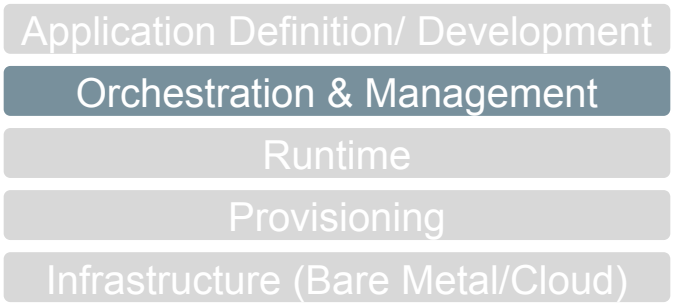
http://github.com/cncf/landscape

@dankohn1 @lennypruss @sraney

Cloud Native Reference Architecture



Example: Management

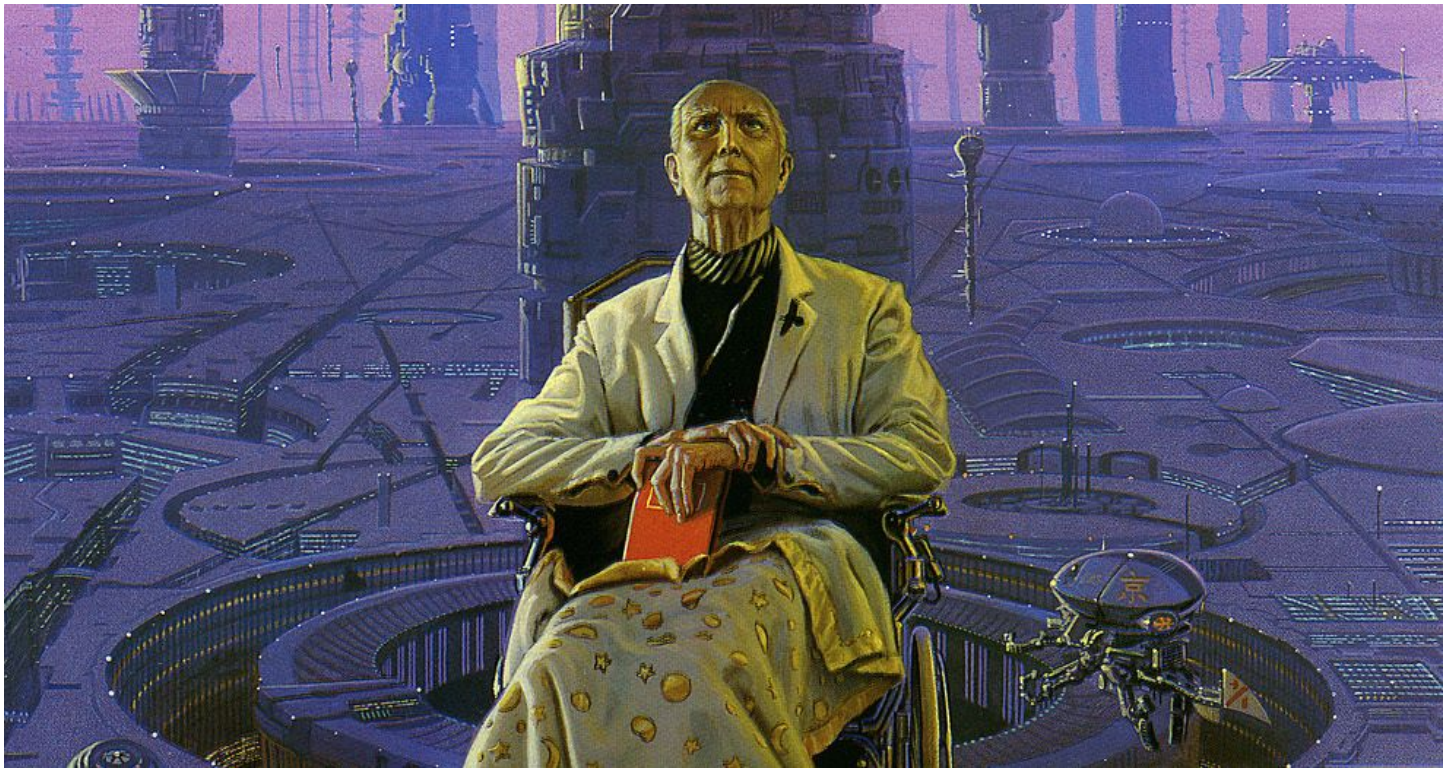


- Observability
  - View / Filter / Replay
  - Monitoring / Trace / Stream / Log
  - Business Intelligence
- Orchestration
- Coordination
  - Configuration
  - Discovery
  - DNS
- Service Management
  - Routing / Proxy / Load Balancer
  - Policy / Placement / Traffic Management

Assume we have the software … why do we need a **Foundation?**

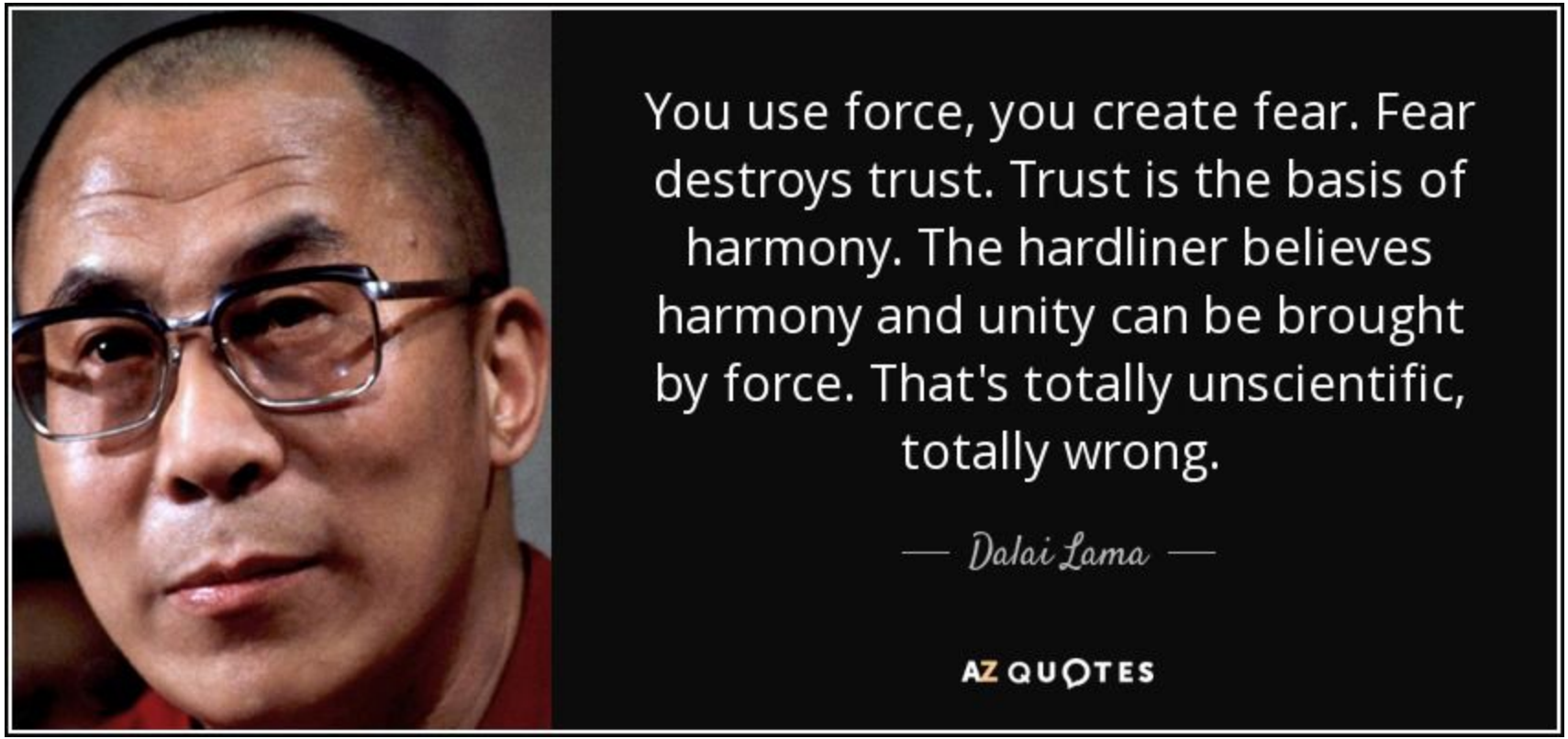
What even is a
Foundation?

..a benevolent technocracy run by civilised robots?



..a federation of collaborating powers?

..a confluence of karmic forces?



The Linux Foundation

Safeguards Linux for the long term

Provides a nexus for collaboration and trust

Is an ubiquitous open source brand

Good for customers & the community!

The LF is great because it makes it easy to bring together collaborators for projects like a common tool kit for cloud native applications → enter the CNCF

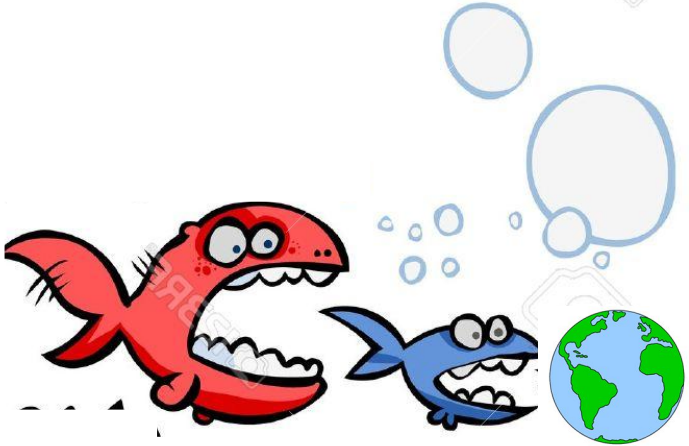
Let’s do it together

Common Open Source

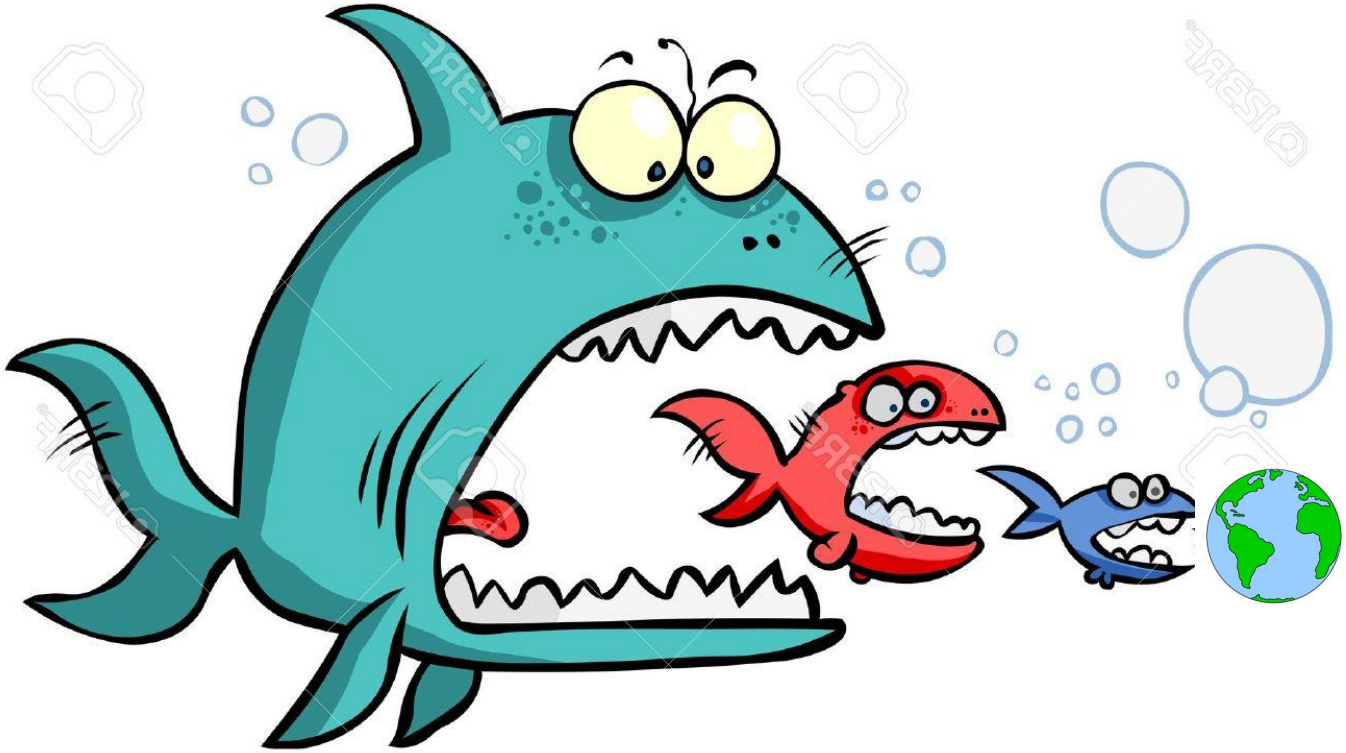
is not proprietary

Software is eating the world

Open Source is eating Software



Cloud is eating Open Source



Without a commons, we risk Cloud Lock In

Foundations curate open source to stop lock in

First it was the big software vendors

Then: big web companies like eBay, AirBnb

Now - big “traditional” companies too

→ Everyone wants to use open source, in a well managed commons, for the “boring infra” bits

Foundations provide Education

Easy - Fast - No Confusion

Guidance and clarity on “Cloud Native”
A badge of trust, quality & interoperability
A common set of tools, APIs & examples

Shared through a modern, trusted commons

Foundations Support Open Source Projects

Infrastructure - Promotion - Interoperability

Infrastructure support with compute and CI

Promotional support by exposing projects to a wider audience

Making sure that projects play nicely with each other .. and more.

Standards

Standards are…

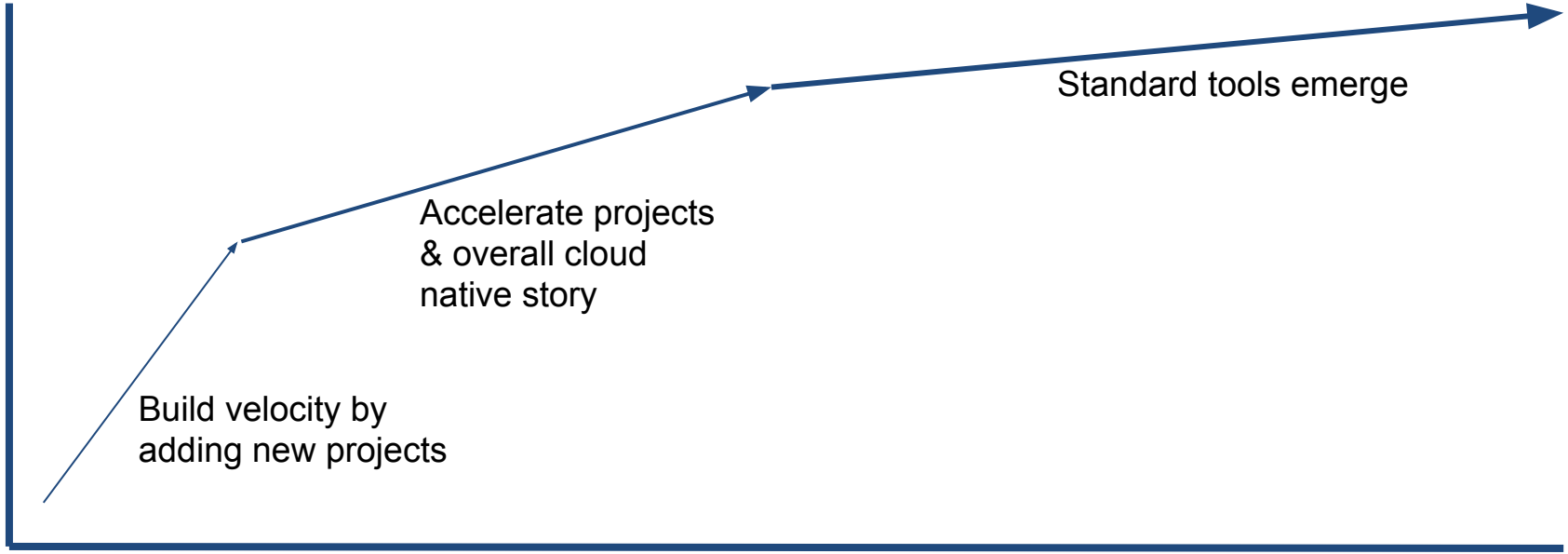
An algorithm for identifying areas of disagreement and maximising them.

Standards are slow. And emerge slowly.

CNCF does not need standards, we need ease of interop and “glue” code

We like conventions that arise from real world use by the community & will strive to identify these and promote them as needed

3 phase plan



https://www.cncf.io/blog/2016/11/08/cloud-native-software-can-trust

Bob Wise, Samsung: “An Ode To Boring”

“I call on the CNCF to formally foster a common community container implementation project backed by the Kubernetes, Mesos, and Cloud Foundry communities. We need a transparent, community-driven implementation ... to become the default container implementation for a wide number of open source orchestration systems”

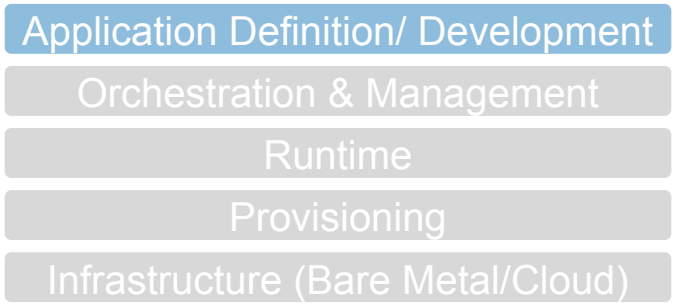
https://medium.com/@bob_48171/an-ode-to-boring-creating-open-and-stable-container-world-4a7a39971443#.2w2edyeir

What is Cloud Native and why should I care?

Alexis Richardson
CNCF TOC Chair & CEO Weaveworks
22 Feb 2017

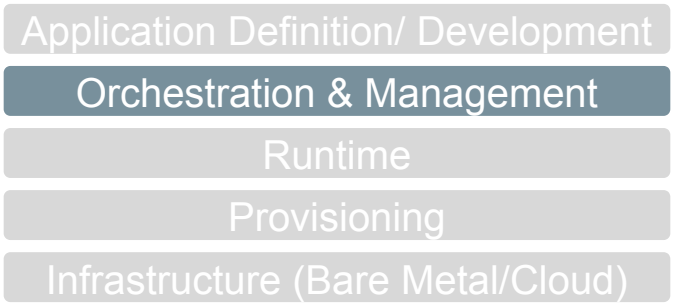
Appendix

Application Definition/ Deployment Layer



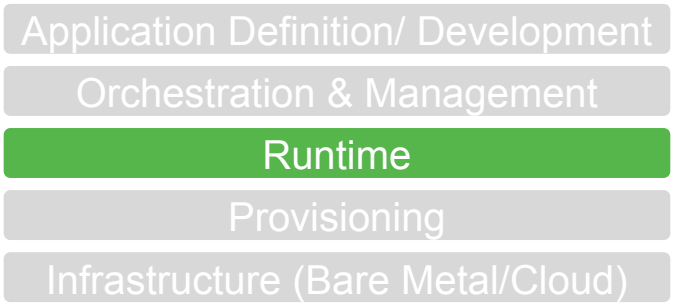
- Application Composition
- Application Delivery
- Application Development Frameworks
- Application Operational Tooling
- CI/CD
- Image Registry / Repository
- Governance and ops model

Orchestration & Management Layer



- Observability
  - View / Filter / Replay
  - Monitoring / Trace / Stream / Log
  - Business Intelligence
- Orchestration
- Coordination
  - Configuration
  - Discovery
  - DNS
- Service Management
  - Routing / Proxy / Load Balancer
  - Policy / Placement / Traffic Management

Runtime Layer

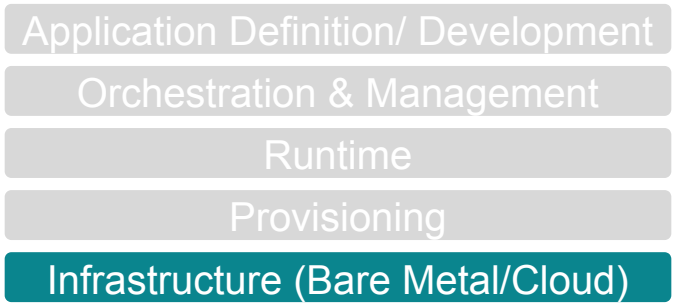


- Resource Management
  - Container Scheduling
  - Container Deployment
- Cloud Native – Network
  - Network Segmentation
  - SDN & APIs (eg CNI, libnetwork)
- Cloud Native – Data
  - Data Management
  - Databases & APIs
- Overall Container Service
  - (Some) PaaS/Platform Services

Provisioning Layer

- OS Management
- Secure Images
- Host level Devops Deployment Tooling & Provisioning

Infrastructure (Bare Metal/Cloud) Layer



- Out of scope for CNCF projects as we do not define infrastructure vendors or cloud solutions but part of reference architecture
- Potentially in the future we will provide “certification”