Joyent®

Node.js & ContainerPilot

Wyatt Preul // jsgeek.com/cp-webinar/

Benefits of Containers

- consistent environments, immutable
- operations that developers can do, speeding up delivery of software
- OS level virtualization, more performant than VM

Docker pitfall - PID 1

- bring your own init (BYOI)
- container inits exist: tini, dumb-init, my_init

Docker pitfall - lifecycle

- need setup and teardown hooks in container
- perform initialization before starting
- perform cleanup (finish writes) before container is killed

Microservice pitfall - load balancer

- subdomains setup for environment (qa, stg, prod)… mistakes will happen, not uncommon for a prod service to point to a QA service, oops
- with lots of microservices and hosts, misconfiguration is likely more common
- increased latency between services

Microservice pitfall - health

- indicate issue with service, or at least an issue between the load balancer and the service - can be unreliable source of truth
- sometimes perform full checks, db connection, memory usage, exposed as public endpoint (/health) … can DoS a service

TRITON
ContainerPilot

Addresses previous issues + FOSS

ContainerPilot

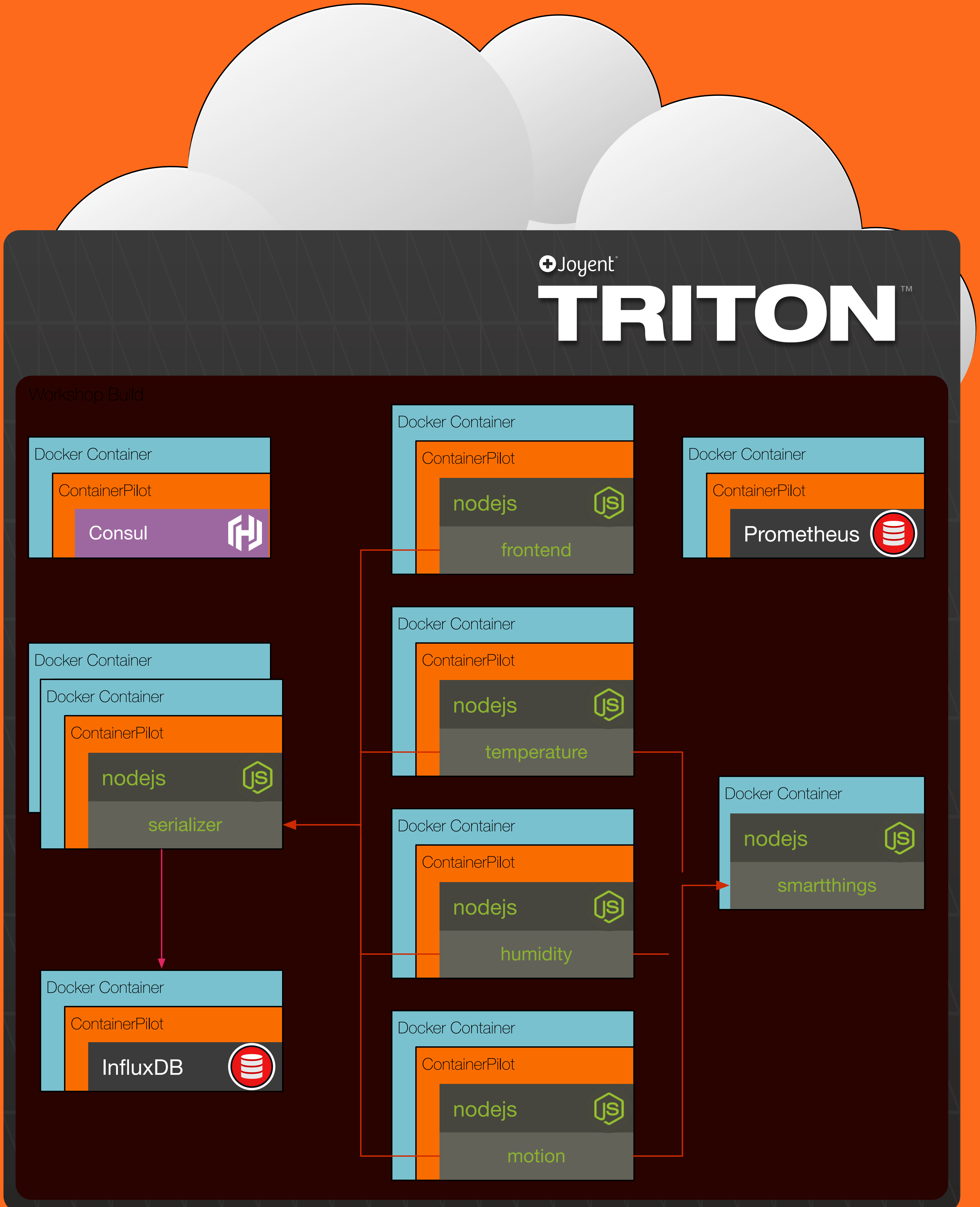
- tool to automate a container’s service discovery, life cycle management, and configuration portable, works anywhere docker does
- capabilities:
    - health checks
    - handles startup and shutdown of services
    - runs as pid 1 in the container
    - watches a service catalog for changes in related services
    - consul, etcd, zookeeper, etc.
    - automatically reconfigures service upon state change
- open-source, free: github.com/joyent/containerpilot

nodejs-example

github.com/autopilotpattern/nodejs-example

Node.js modules

- hapi - web API framework
- Seneca - microservices framework
- Piloted - ContainerPilot integration, relies on consul
- Wreck - simple module for making performant HTTP requests


Joyent
TRITON
Workshop Build
Docker Container
ContainerPilot
Consul
Docker Container
ContainerPilot
nodejs
frontend
Docker Container
ContainerPilot
Prometheus
Docker Container
Docker Container
ContainerPilot
nodejs
serializer
Docker Container
ContainerPilot
nodejs
temperature
Docker Container
nodejs
smartthings
Docker Container
ContainerPilot
nodejs
humidity
Docker Container
ContainerPilot
InfluxDB
Docker Container
ContainerPilot
nodejs
motion

Code & Demo

```
$ git clone https://github.com/autopilotpattern/nodejs-example.git


$ cd nodejs-example


$ EDITOR .
```

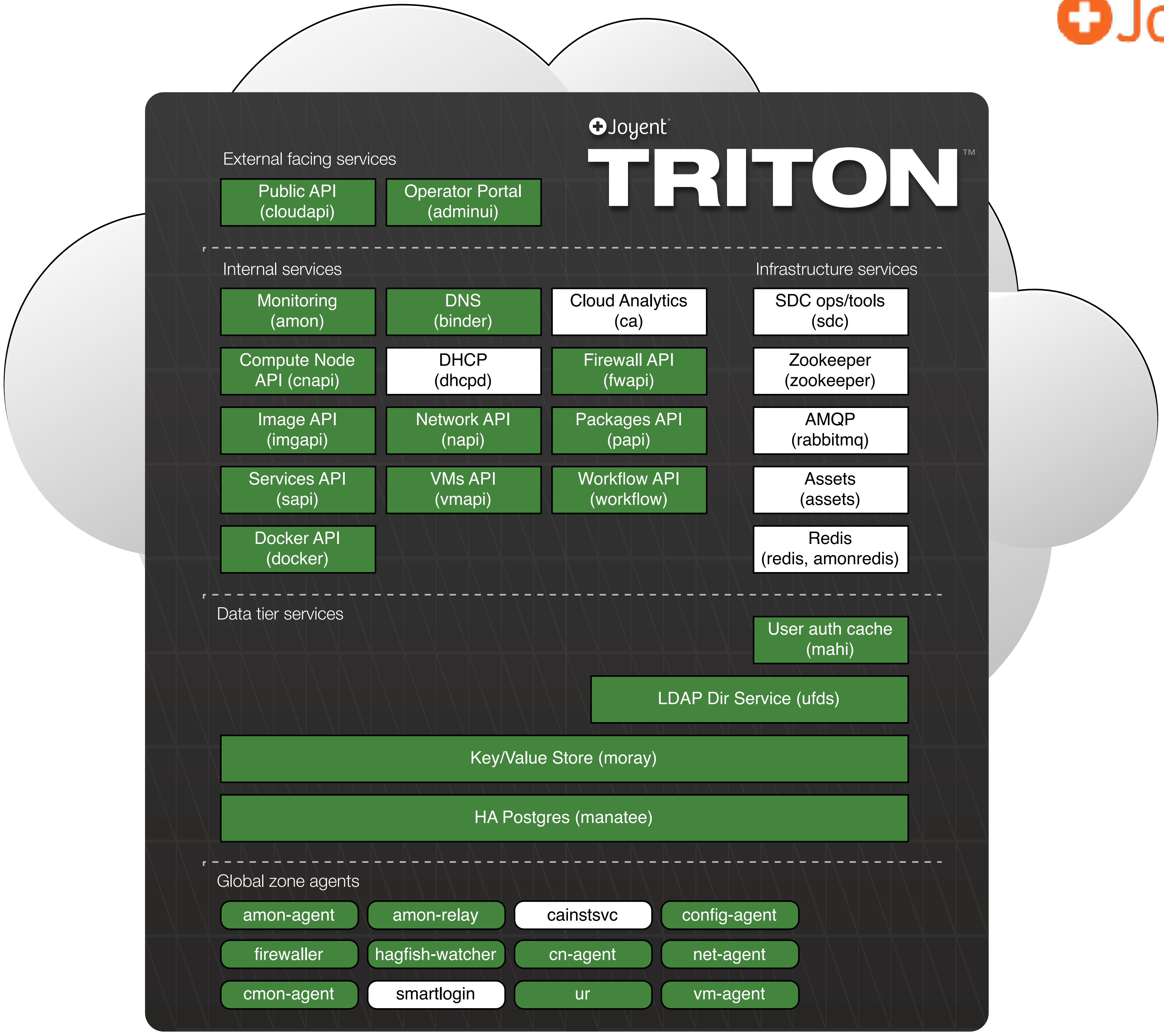
Recap

- Use ContainerPilot with Node.js docker containers (piloted module)
- Use consul for discovery (autopilotpattern/consul)
- Make microservices independently deployable and fault tolerant

Deploying to prod

Triton Provides

- Containers as a Service
  - Docker - The data center is the docker host
- Software for Public and Private deployment
- High Performance
- High Security
- Open Source!


Joyent
TRITON™
External facing services
Public API (cloudapi)
Operator Portal (adminui)
Internal services
Monitoring (amon)
DNS (binder)
Cloud Analytics (ca)
Compute Node API (cnapi)
DHCP (dhcpd)
Firewall API (fwapi)
Image API (imgapi)
Network API (napi)
Packages API (papi)
Services API (sapi)
VMs API (vmapi)
Workflow API (workflow)
Docker API (docker)
Infrastructure services
SDC ops/tools (sdc)
Zookeeper (zookeeper)
AMQP (rabbitmq)
Assets (assets)
Redis (redis, amonredis)
Data tier services
User auth cache (mahi)
LDAP Dir Service (ufds)
Key/Value Store (moray)
HA Postgres (manatee)
Global zone agents
amon-agent
amon-relay
cainstsvc
config-agent
firewaller
hagfish-watcher
cn-agent
net-agent
cmon-agent
smartlogin
ur
vm-agent

Docker on Triton

- Docker Containers = Triton Instances
- No difference other than how the are managed
  - Docker - via Docker API (docker run etc)
  - Triton Instances - via CloudAPI (triton create)
- Based on LX instances
- Native networking
  - Each container get's it's own IP address(es)
  - No port mapping as such. Firewall rules used to open "mapped" ports
  - Container name service, A Records for groups of services (e.g. consul.srvc.us-sw-1.cns.joyent.com)

Docker on Triton - Demo

```
$ eval $(triton env)

$ docker-compose up -d

$ open http://$(triton ip nodejsexample_frontend_1)

$ docker logs -f nodejsexample_frontend_1
```

Joyent®

Questions?

Links @ jsgeek.com/cp-webinar