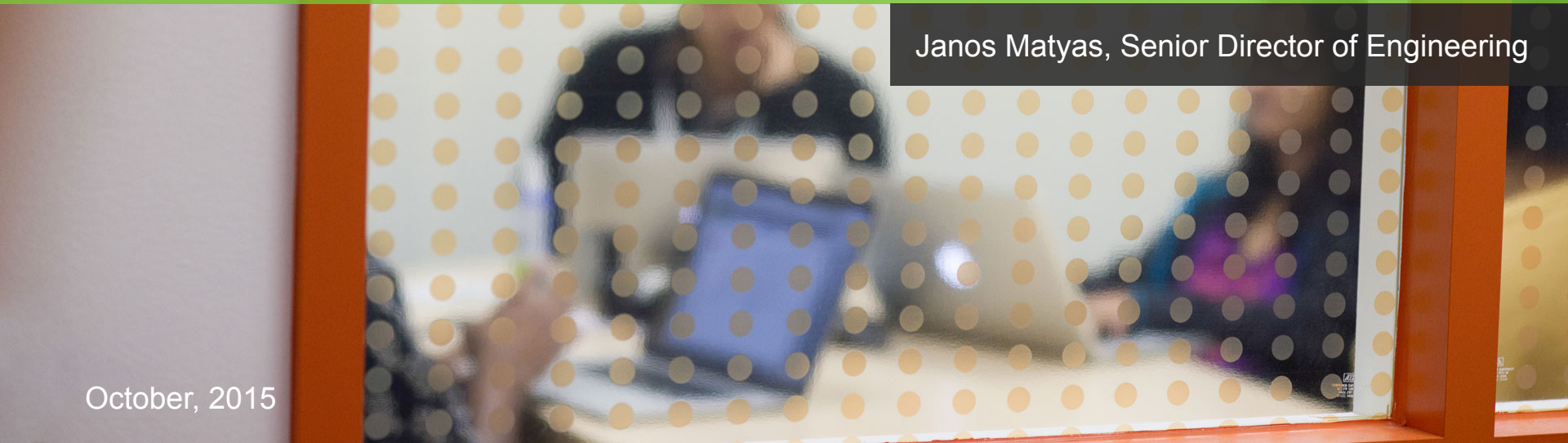
Hortonworks

One click Hadoop clusters - anywhere



Janos Matyas, Senior Director of Engineering

October, 2015

Overview

- Introduction
- Goals and motivations
- Technology stack
- How it works
- Results/achievements/future plans
- Demo and Q&A

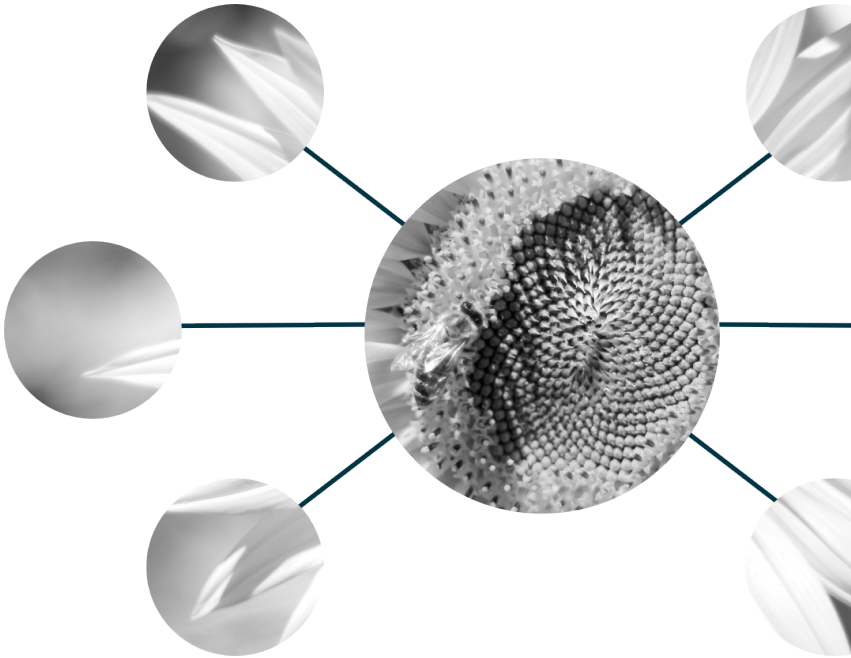
Hortonworks

Goals and motivations

- Full Hadoop stack provisioning – everywhere
- Automate and unify the process
- Zero-configuration approach
- Same process through a cluster lifecycle (Dev, QA, UAT, Prod)
- Provide tooling - UI, REST API and CLI/shell
- Secure and multi-tenant
- SLA policy based autoscaling

Technology stack

- Docker
- Swarm
- Consul
- Apache Ambari



Hortonworks

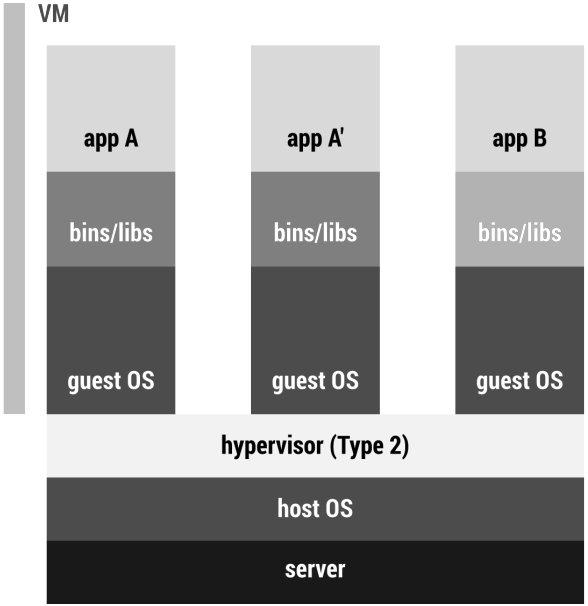
Docker

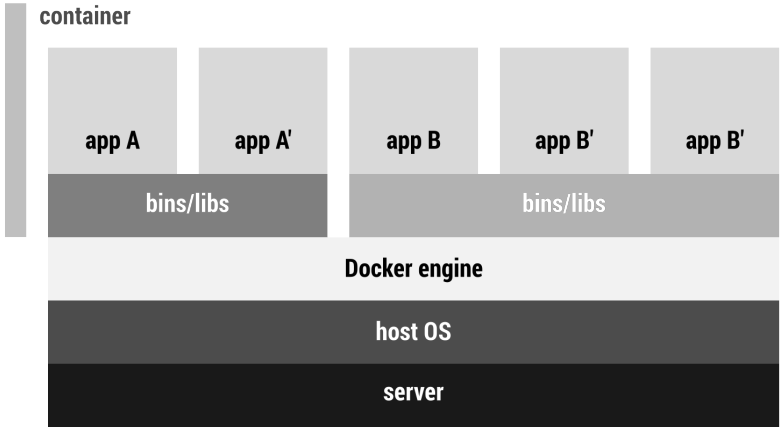
- Container based virtualization
- Lightweight and portable
- Build once, run anywhere
- Ease of packaging applications
- Automated and scripted
- Isolated

Hortonworks

Docker – How it works

- Containers are isolated, but share OS and bins/ libraries
- No need to emulate hardware





Hortonworks

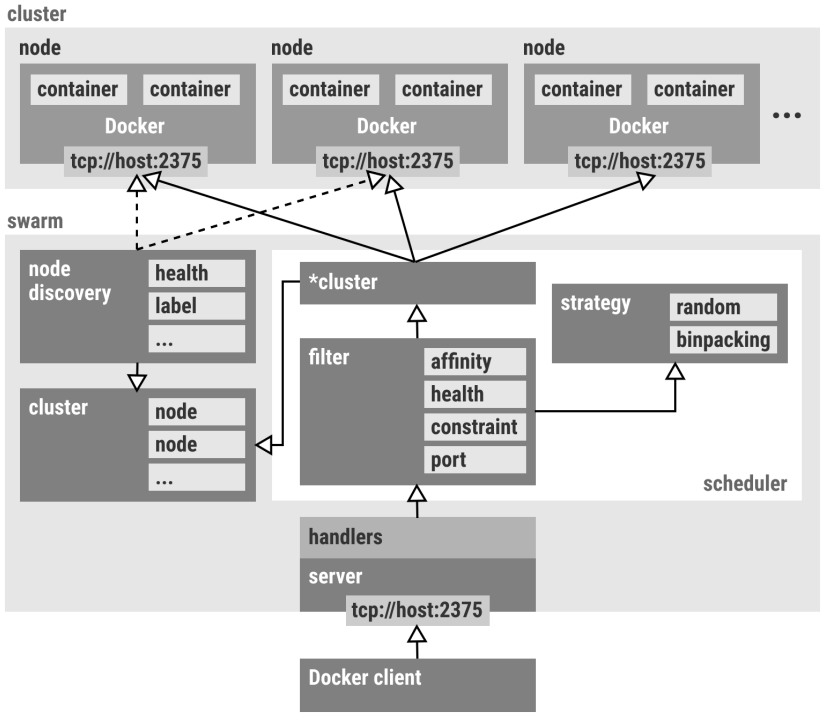
Swarm

- Native clustering for Docker
- Distributed container orchestration
- Same API as Docker

Hortonworks

Swarm – How it works

- Swarm managers/agents
- Discovery services
- Advanced scheduling



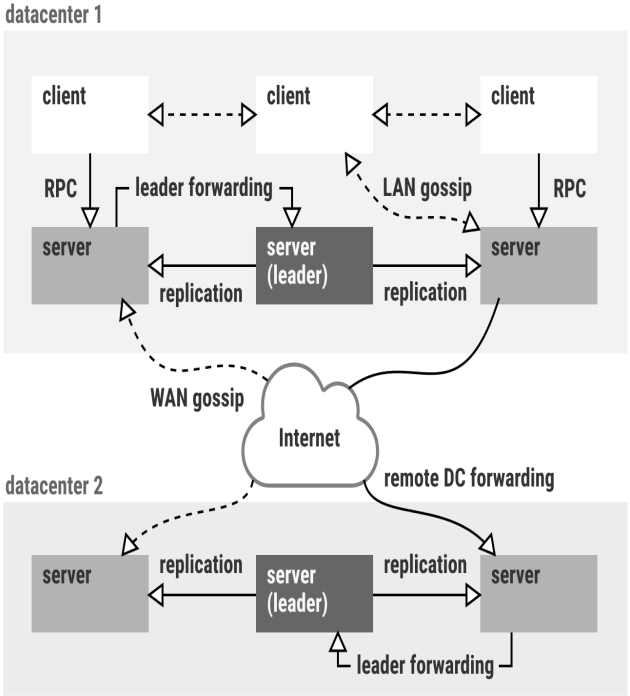
Consul

- Service discovery/registry
- Health checking
- Key/Value store
- DNS
- Multi datacenter aware

Hortonworks

Consul – How it works

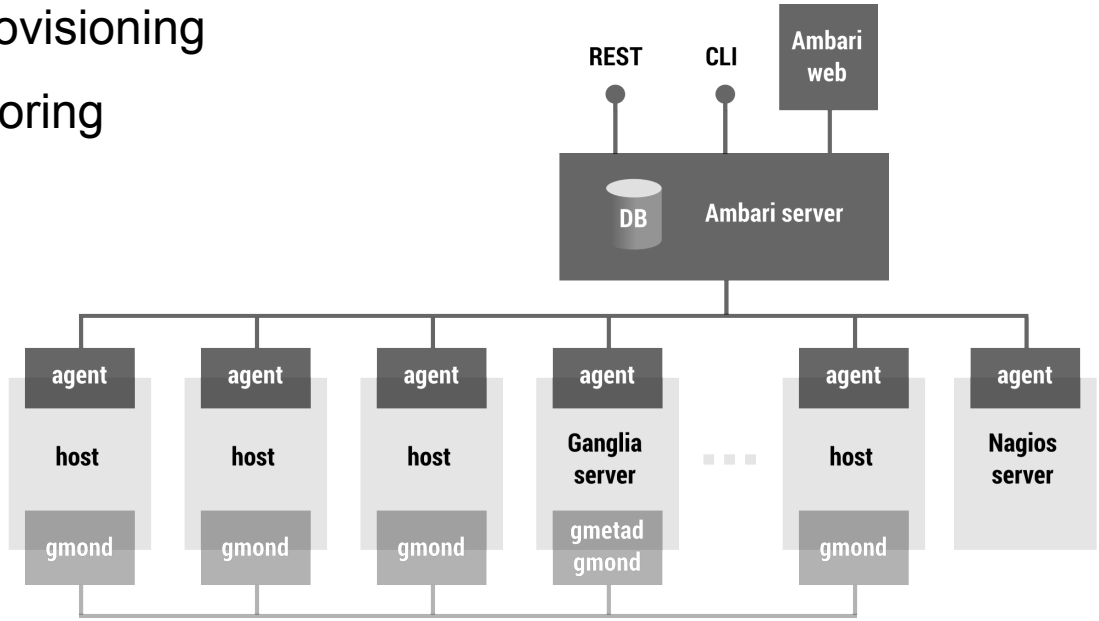
- Consul servers/agents
- Consistency through a quorum (RAFT)
- Scalability due to gossip based protocol (SWIM)
- Decentralized and fault tolerant
- Highly available
- Consistency over availability (CP)
- Multiple interfaces - HTTP and DNS
- Support for watches



Hortonworks

Apache Ambari

- Easy Hadoop cluster provisioning
- Management and monitoring
- Key feature - Blueprints
- REST API, CLI shell
- Extensible
  - Stacks
  - Services
  - Views



Hortonworks

Apache Ambari – How it works

- Ambari server/agents
- Define a blueprint (blueprint.json)
- Define a host mapping (hostmapping.json)
- Post the cluster create

Hortonworks

Cloudbreak

Cloudbreak is a cloud-agnostic Hadoop as a Service API. Abstracts the provisioning and ease management and monitoring of on-demand clusters.

Cloudbreak is a powerful left surf that breaks over a coral reef, a mile off southwest the island of Tavarua, Fiji.

Hortonworks

Cloudbreak

- Benefits
  - Zero configuration
  - Elastic
  - Secure
  - Infrastructure agnostic
  - Heterogenous clusters
  - Auto-scaling
- Main REST resources
- /template – specify an instance group infrastructure
- /stack – creates an infrastructure based on a template
- /blueprint – describes a Hadoop cluster
- /cluster – creates a Hadoop cluster

Hortonworks

Cloudbreak – How it works

- Start VMs - with a running Docker daemon
- Cloudbreak Bootstrap
  - Start Consul Cluster
  - Start Swarm Cluster (Consul for discovery)
- Start Ambari servers/agents - Swarm API
- Ambari services registered in Consul (Registrator)
- Post Blueprint

Hortonworks

Cloudbreak - Features

- Extensible – easy to implement Service Provider Interface
- Cloudbreak “recipes”
  - Automate host configuration
  - Pre/post Ambari lifecycle hooks
  - Services reconfiguration
  - Automate/execute custom actions
- Side – effects
  - Ambari CLI/shell and Groovy based client
  - Cloud Foundry’s UAA Dockerized
  - Munchausen – bootstrap Swarm with Consul
  - Dockerized full Hadoop stack (Apache Hadoop 60K+, Ambari 12K+, Spark 10K+ downloads)

Cloudbreak - Hadoop as a Service API

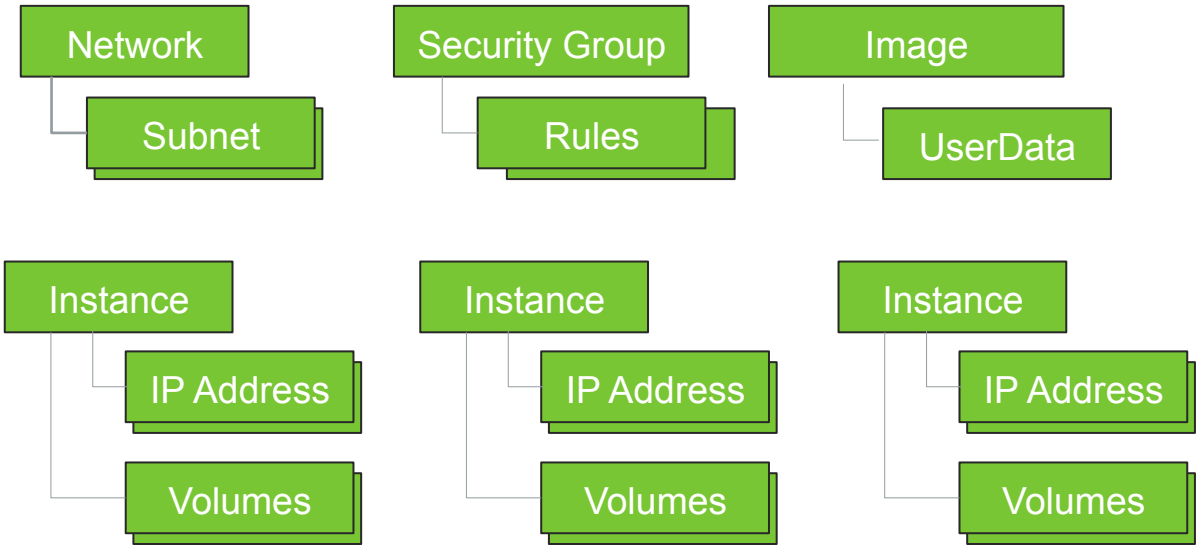
- Public tech preview
  - Microsoft Azure
  - Amazon AWS
  - Google Cloud Platform
  - OpenStack
- Private tech preview – R&D
  - Bare metal
  - Rackspace Managed Cloud
  - HP Helion Public Cloud

***integration SPI is available**

Hortonworks

Cloudbreak – SPI

- Cloud providers have very different API, though model is very similar
- Non – invasive implementation
- One interface to implement - *CloudPlatformConnector*



Hortonworks

Periscope

Hortonworks

Periscope

- Benefits
  - Zero configuration
  - Metric and time based alarms
  - SLA policy based autoscaling
  - Secure
  - Hostgroup specific
- Main REST resources
- /clusters – specify a cluster to be monitored
- /alerts– time and metric based
- /policies – specify an SLA policy for a cluster based on an alarm
- /applications – specify an SLA policy for an application (under development)

Hortonworks

Periscope – How it works

- Configures/monitors alarms in Ambari
- Setup alarm, cooldown periods
- Manages cluster sizes
- Allow to associate SLA scaling policies to alarms
- Orchestrates Cloudbreak to up/downscale the cluster

Hortonworks

Demo and Q&A