High Availability

Keeping Applications Running

Cloud Foundry's Four Levels of High Availability

Pivotal

Overview

- After completing this lesson, you should understand:
  - The Four Levels of High Availability
  - What is BOSH and how it contributes to HA
  - Availability Zones
  - How failed application instances can be automatically restarted

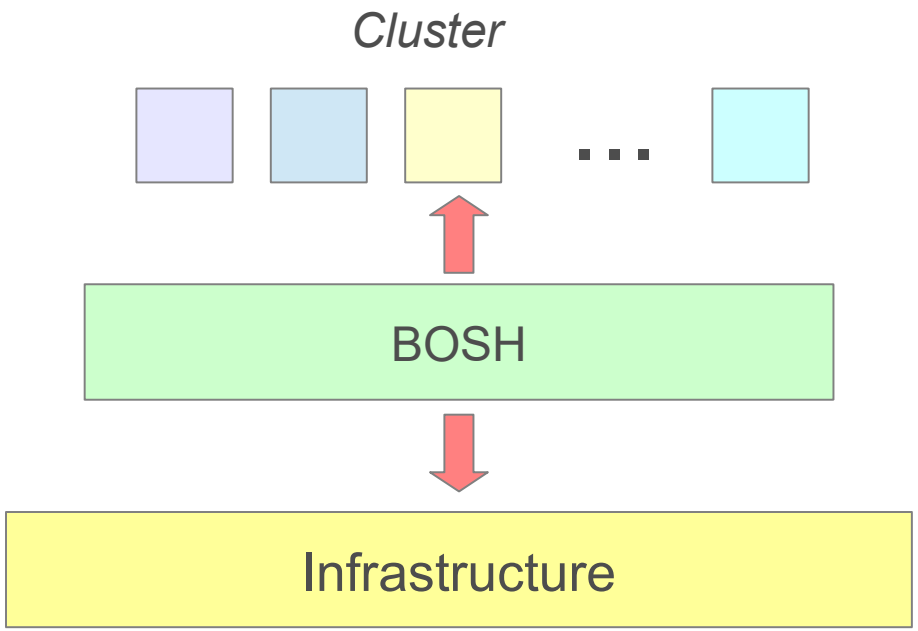
Pivotal

Roadmap

- **BOSH Overview**
- The Four Levels of High Availability
  - BOSH and Health Monitoring
  - Availability Zones
  - Handling Application Instance failure (Converger)
- Appendix: The pre-DIEGO Health Manager

What is BOSH?

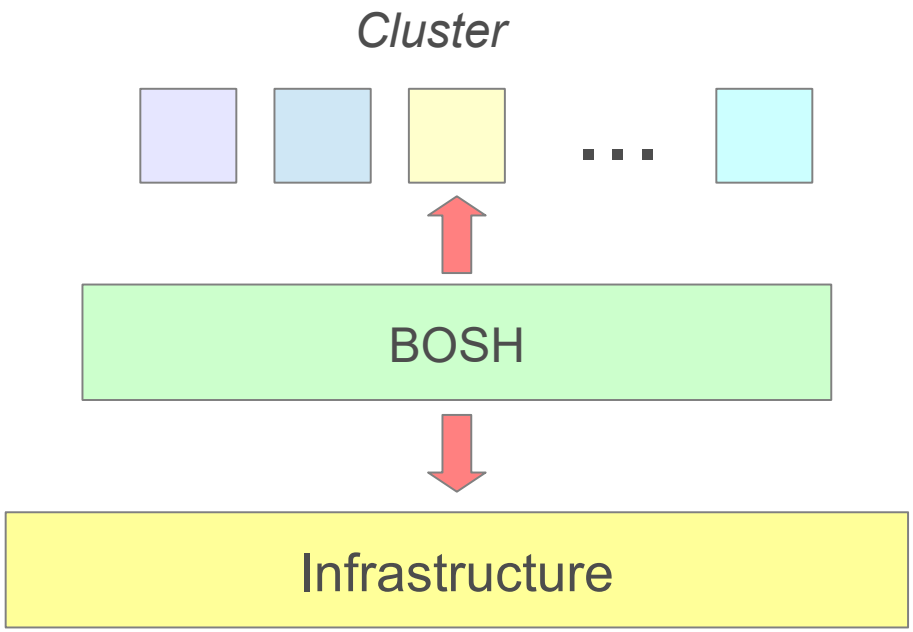
- Open source software
- Deploys & manages the lifecycle of any software
- Designed for distributed systems & clusters
  - *Like Cloud Foundry!*
- BOSH can run on any infrastructure
  - For example: vSphere, Amazon Web Services, a Vagrant VM …



Pivotal

Two Perspectives of BOSH

- BOSH is *middleware*
  - Connects the infrastructure to the cluster
- Looking “up”, it is deploying and managing the lifecycle of a cluster
  - Create/recreate VMs, start/restart processes, enable continuous delivery
- Looking “down”, it is using and abstracting the infrastructure
  - Cluster doesn't care what infrastructure it uses

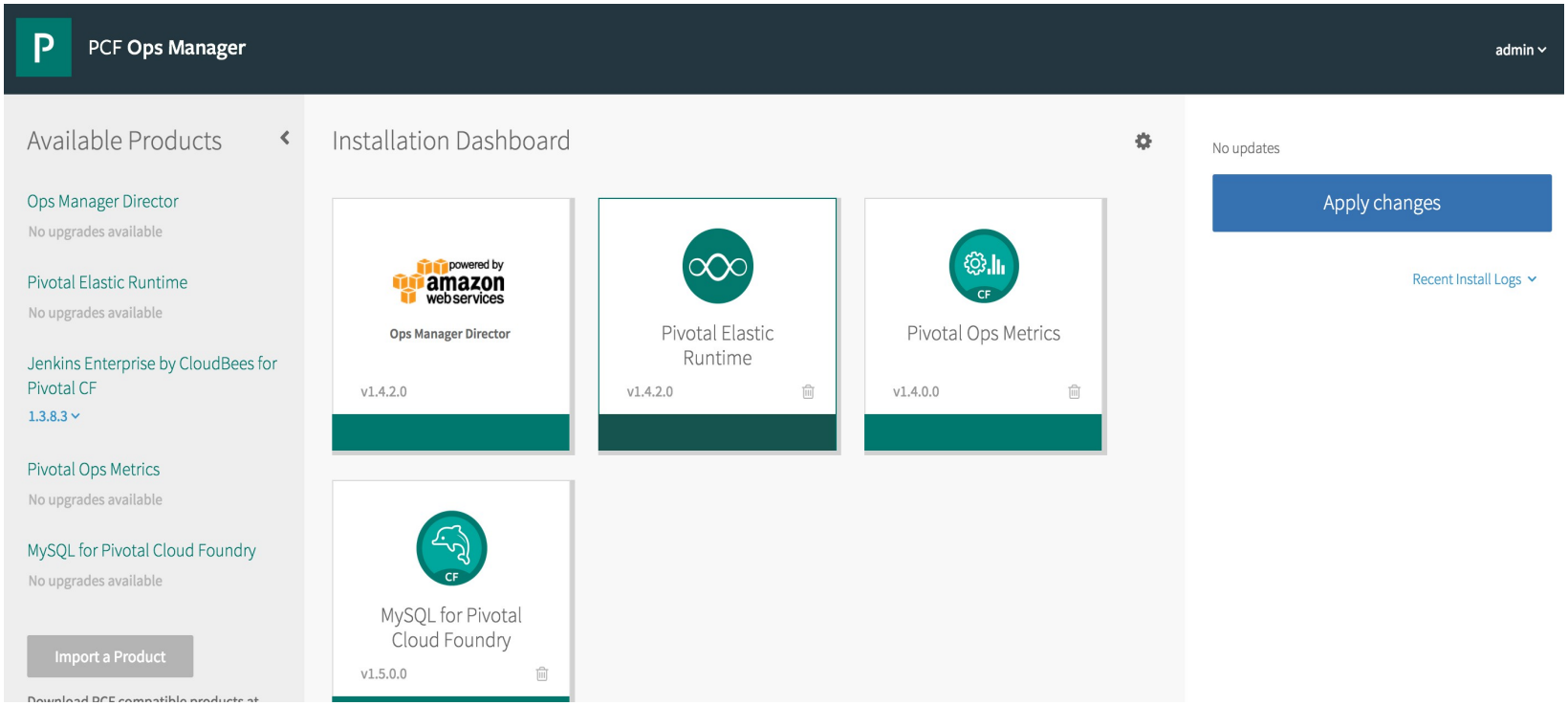


Installing Cloud Foundry

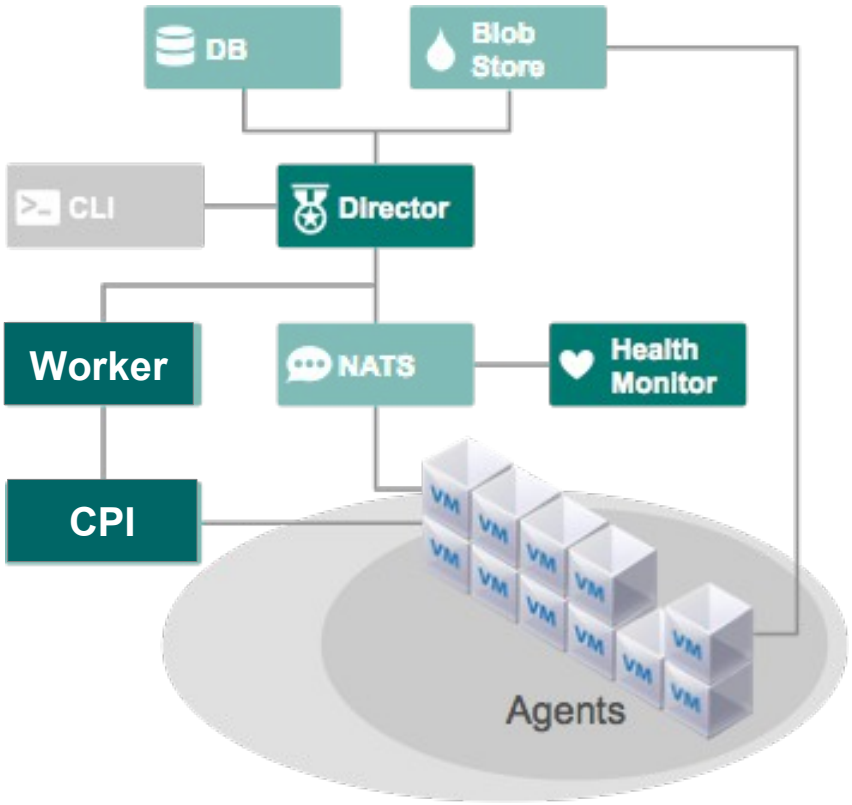
- Actually requires installing BOSH
  - BOSH then deploys everything else
- BOSH deploys "*Releases*"
  - One or more template VMs or "stem-cells"
  - The software to run on the VMs it creates
  - A manifest to drive the whole process
    - Defining a BOSH manifest by hand is complex
- The Elastic Runtime and Services are Bosh Releases
  - Deployed and maintained by BOSH
- Cloud Foundry *Operators* typically use BOSH

Ops Manager and BOSH

- Ops Manager is a user interface on top of BOSH
- When you install or make changes in Ops Manager, BOSH-level changes are being made
  - You will see BOSH-related messages in the logs



BOSH Components



Director

- Co-ordinates BOSH (like CF cloud-controller)
- Maintains own databases

Health Monitor

- Receives heartbeats from deployed VMs

Worker

- Used to run or modify VMs

Cloud Provider Interface

- Interface to underlying IaaS

Pivotal

Roadmap

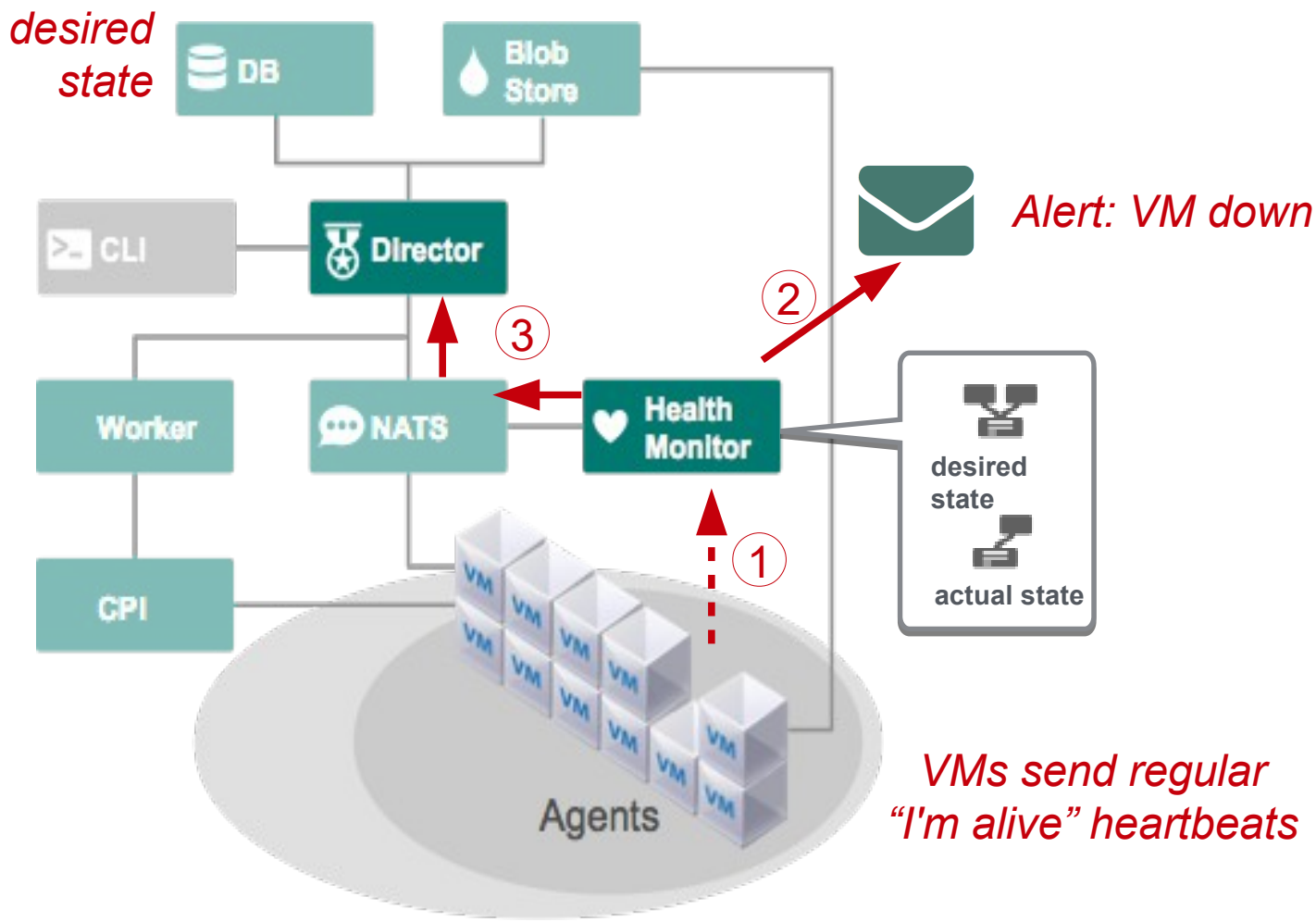
- BOSH Overview
- **The Four Levels of High Availability**
  - **BOSH and Health Monitoring**
  - Availability Zones
  - Handling Application Instance failure (Converger)
- Appendix: The pre-DIEGO Health Manager

High Availability

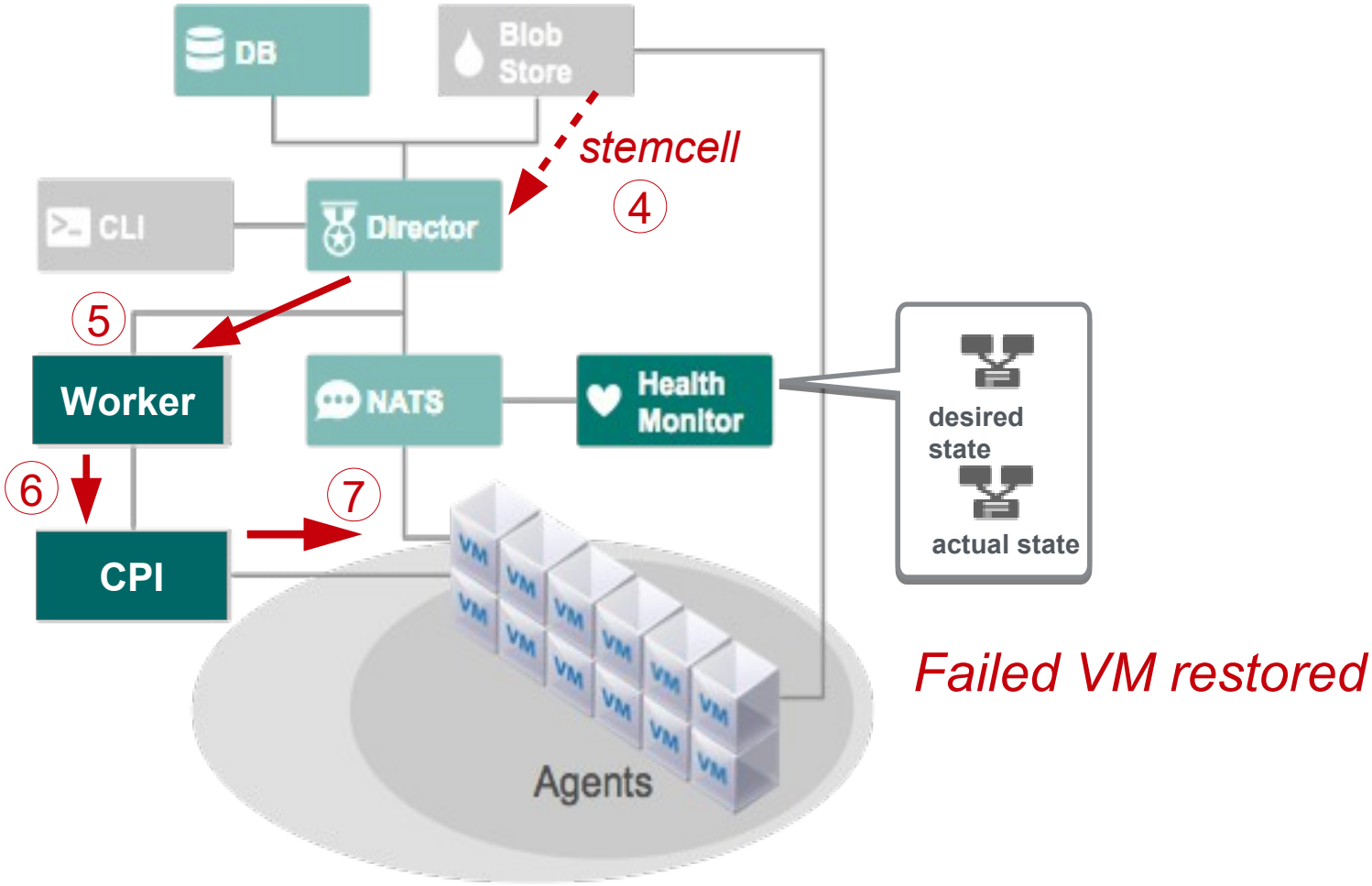
- BOSH monitors the state of the VMs it creates and the processes running on those VMs
  - Dedicated Health Monitor process
- Health Monitor compares the desired state of the system with the actual state
  - Based on periodic heartbeats from VMs
  - If they differ it arranges for failed VMs or processes to be restarted

Pivotal

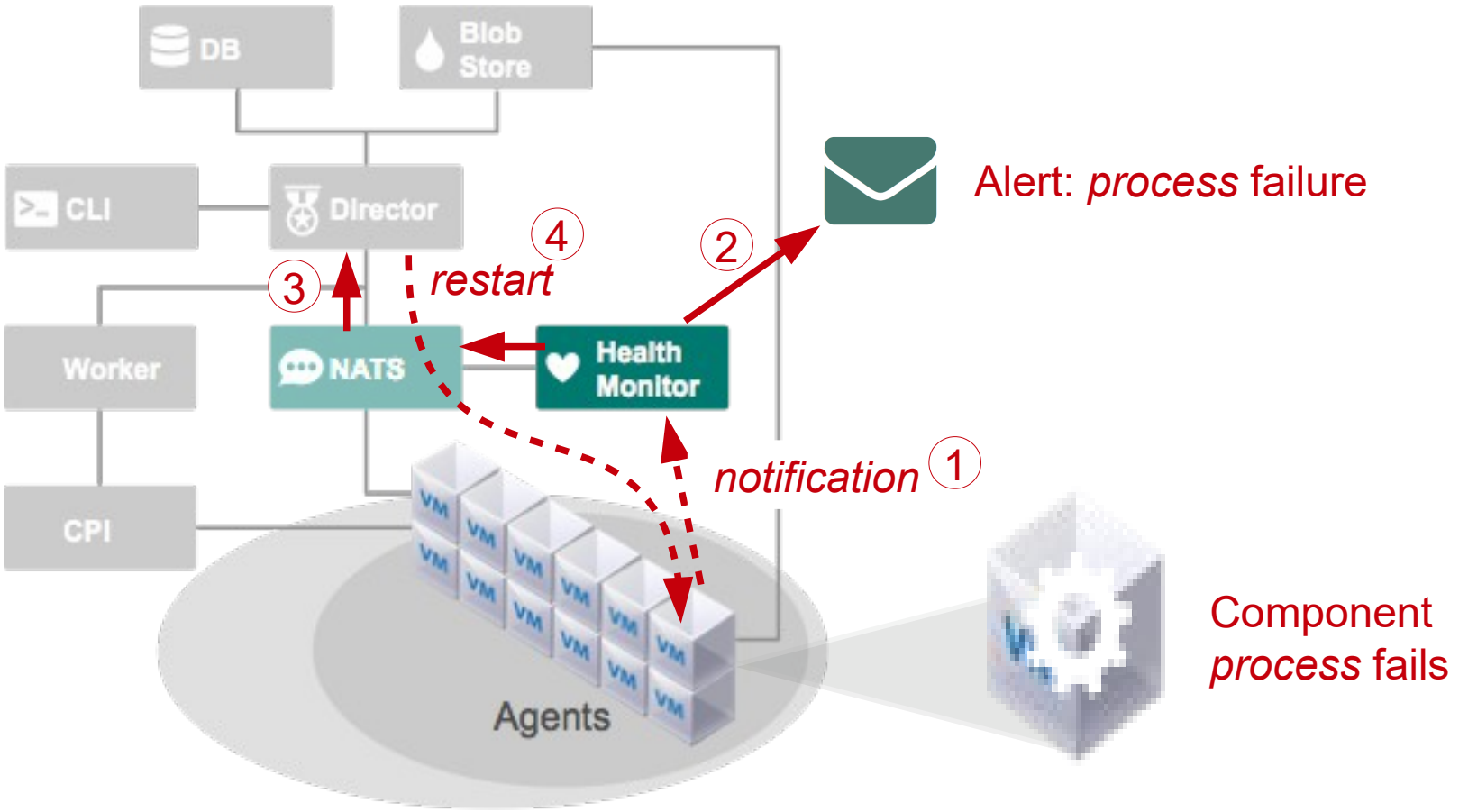
Scenario 1: Component VM Failure



Scenario 1: Component VM Failure Fixed



Scenario 2: Component *Process* Failure

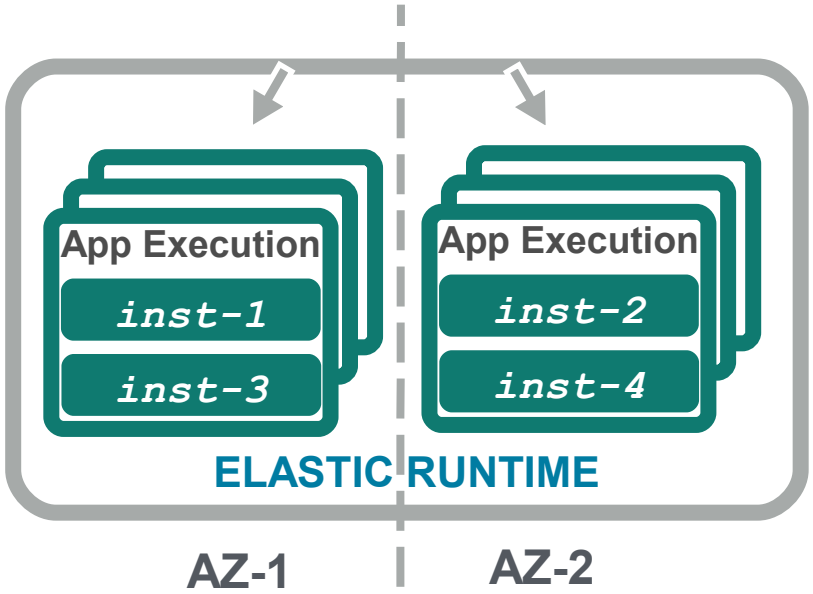


Roadmap

- BOSH Overview
- **The Four Levels of High Availability**
  - BOSH and Health Monitoring
  - **Availability Zones**
  - Handling Application Instance failure (Converger)
- Appendix: The pre-DIEGO Health Manager

Application High Availability

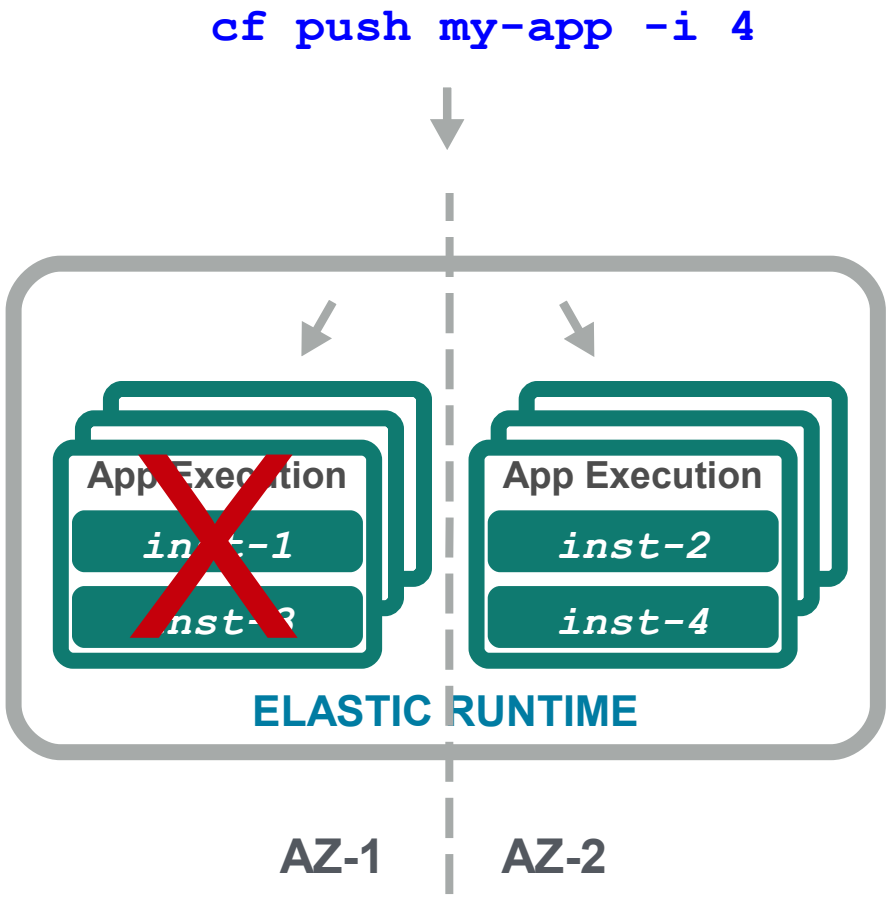
- Cloud Foundry will scale application instances across Availability Zones
- Zones correspond to independent infrastructure segments
  - Such as different racks, or even different data centers
- Provide physical isolation & redundancy
  - Feature of the IaaS layer
  - Defined to CF during installation



Pivotal

Scenario: Availability Zone Failure

- If one zone fails, the application instances in the other zone pick up the load
  - ***No Outage***
- *Recommendation:* push at least **twice** as many instances as zones



Roadmap

- BOSH Overview
- **The Four Levels of High Availability**
  - BOSH and Health Monitoring
  - Availability Zones
  - **Handling Application Instance failure (Converger)**
- Appendix: The pre-DIEGO Health Manager

Pivotal

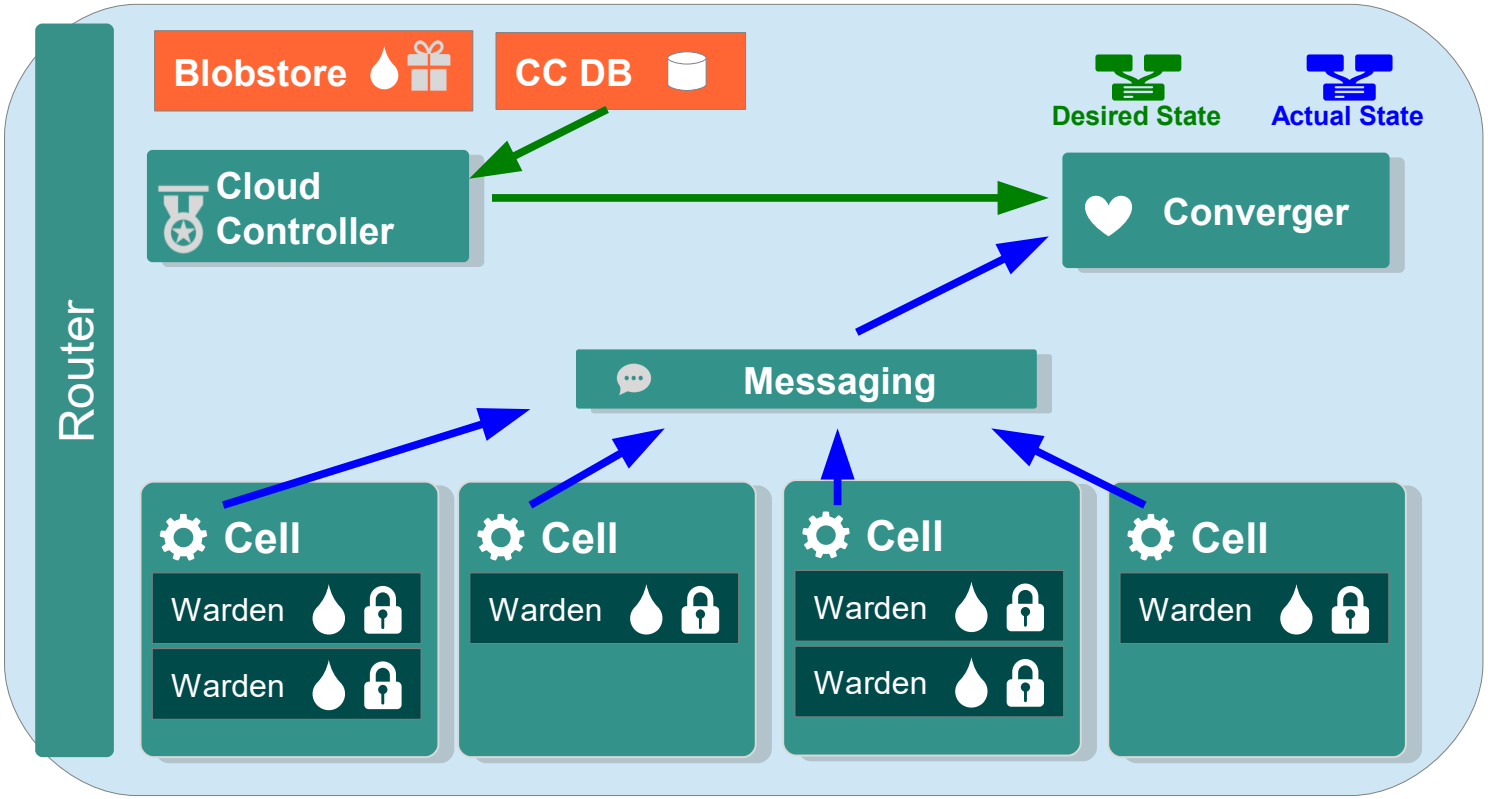
Converger (DIEGO)

- Converger compares desired with actual state
  - State may change because
    - A new application is deployed
    - An application is scaled up or down
    - An instance fails
  - Any discrepancies cause an instance start/stop request to be placed on the Bulletin Board
  - Cloud Controller is *not* involved in restart
- Instance failure is *not* a special case
  - The auctioneer requests a Cell to use
  - The instance is started in a container on that Cell

Pivotal

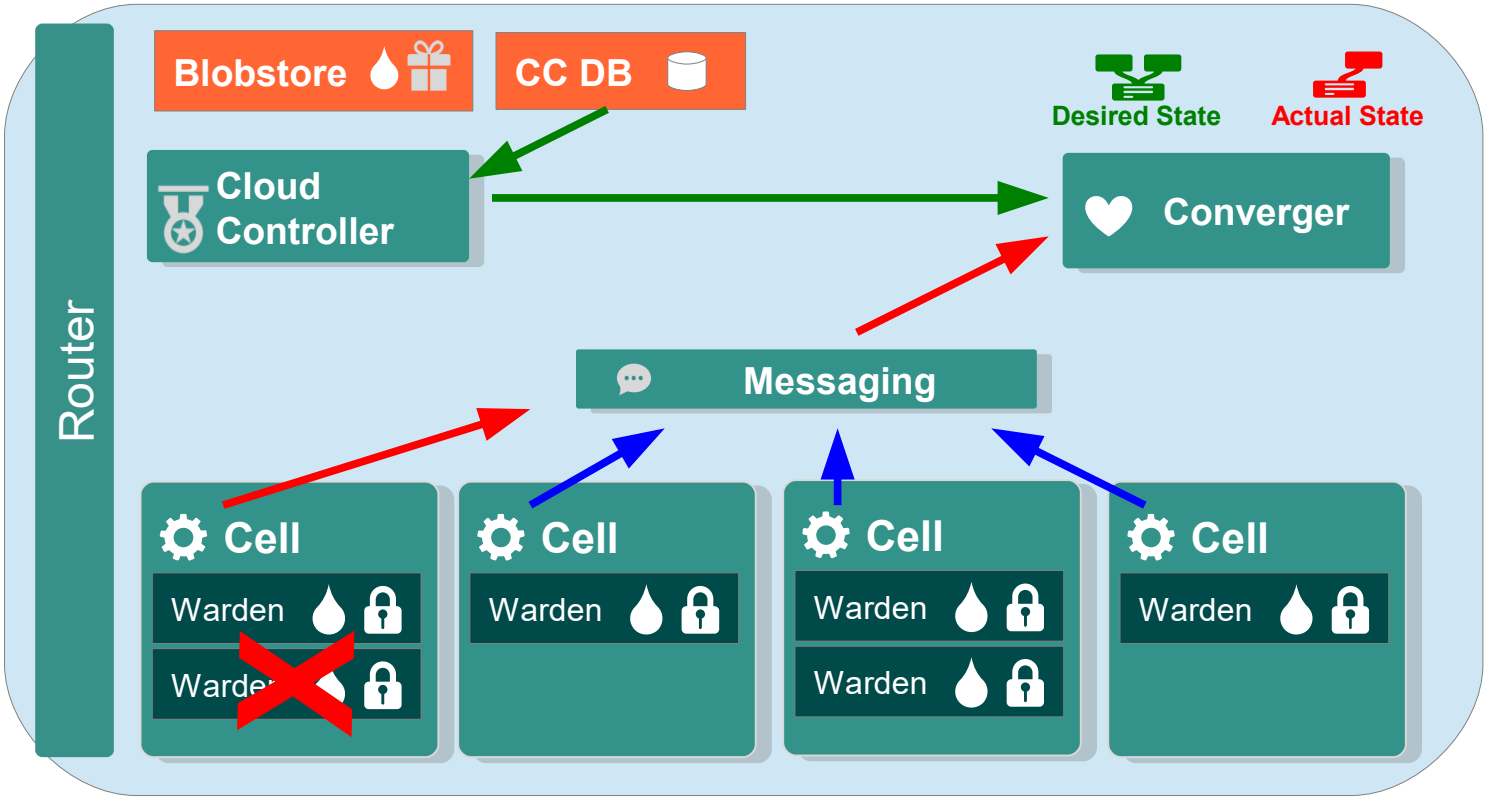
Health Management

1. Applications run within containers on Cells
2. Cells send heartbeat messages, messages sent to Converger
3. Converger obtains “desired state” from Cloud Controller (from CC DB)
4. Does “desired state” = “actual state”? Yes



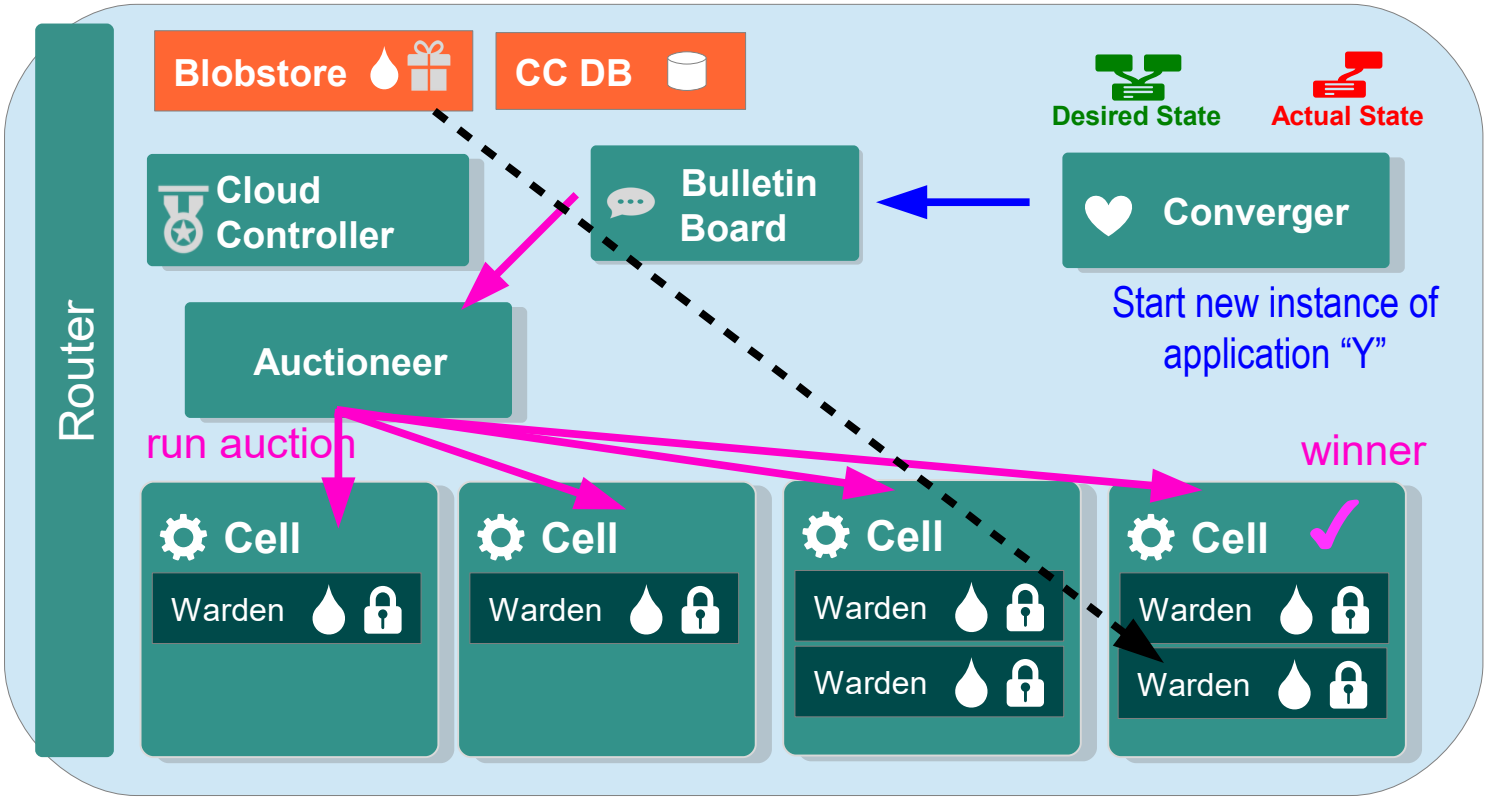
Health Management – Detecting Failures

1. What if an application crashes **X**?
2. Cells send heartbeat messages, messages sent to Converger
3. Converger obtains “desired state” from Cloud Controller
4. “Desired state” = “actual state”? No



Health Management - Replacing an *Application*

1. Converger puts *new instance* request on Bulletin Board
2. Auctioneer accepts request and conducts and auction
3. Auctioneer tells winning Cell to run Droplet in a new container



Summary

- After completing this lesson, you should have learned about:
  - BOSH and its internal architecture
  - How BOSH provides high-availability
  - Availability Zones
  - How the Cloud Foundry enables application restart

Pivotal

Roadmap

- BOSH Overview
- The Four Levels of High Availability
  - BOSH and Health Monitoring
  - Availability Zones
  - Handling Application Instance failure
- **Appendix: Health Manager**

Health Manager (DEA)

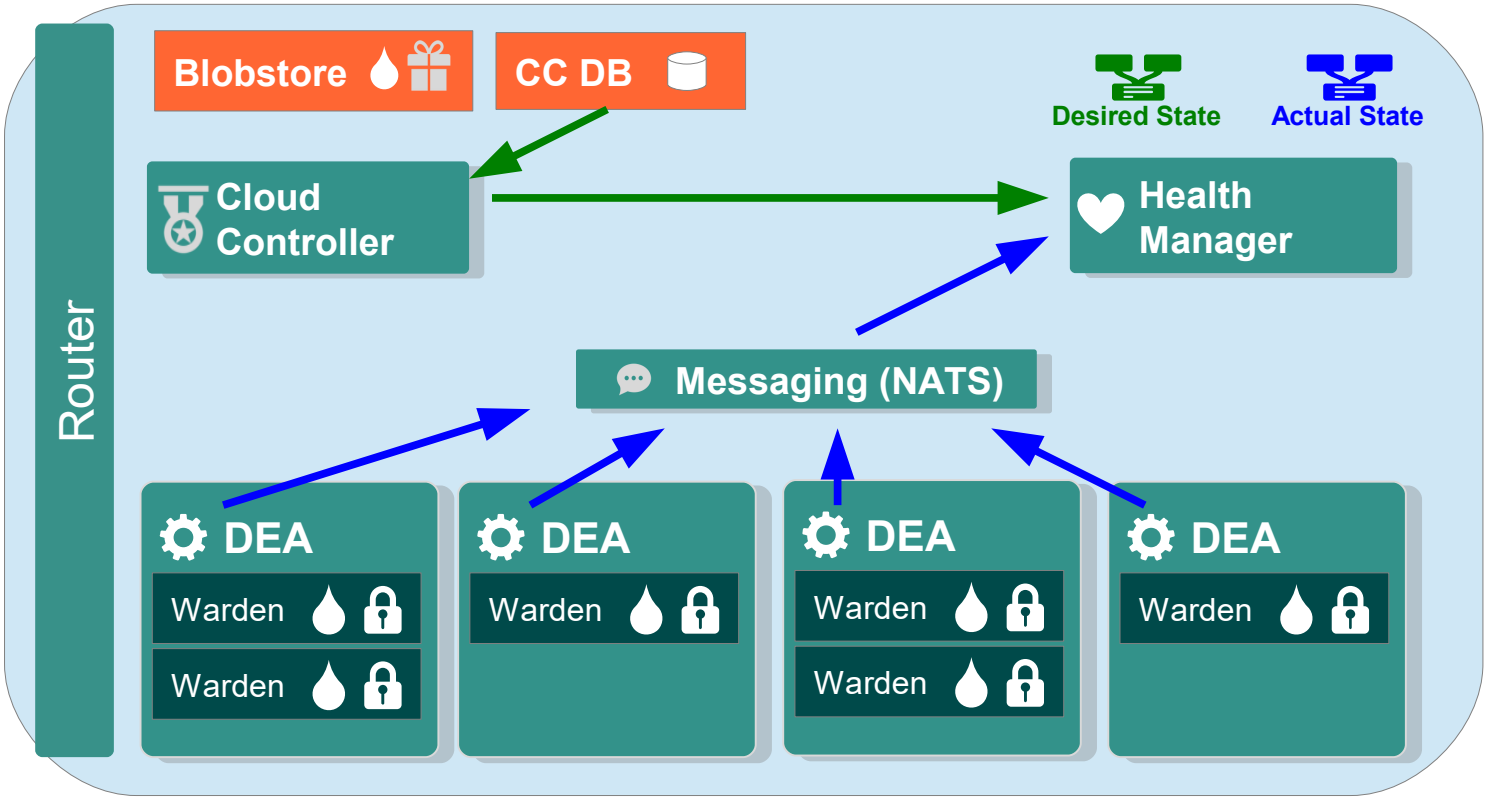
- Component of previous (DEA) Elastic Runtime system
  - Monitors applications to determine version, current state and number of instances
    - Actual state of an application is based on heartbeats from DEAs running the application
  - Determines applications' desired state, version, and instances and compare to actual state
    - Desired state based upon the Cloud Controller database
  - Directs Cloud Controller to take action to correct any discrepancies in the state of applications

Process is very similar to how the *Health Monitor* maintains BOSH

Pivotal

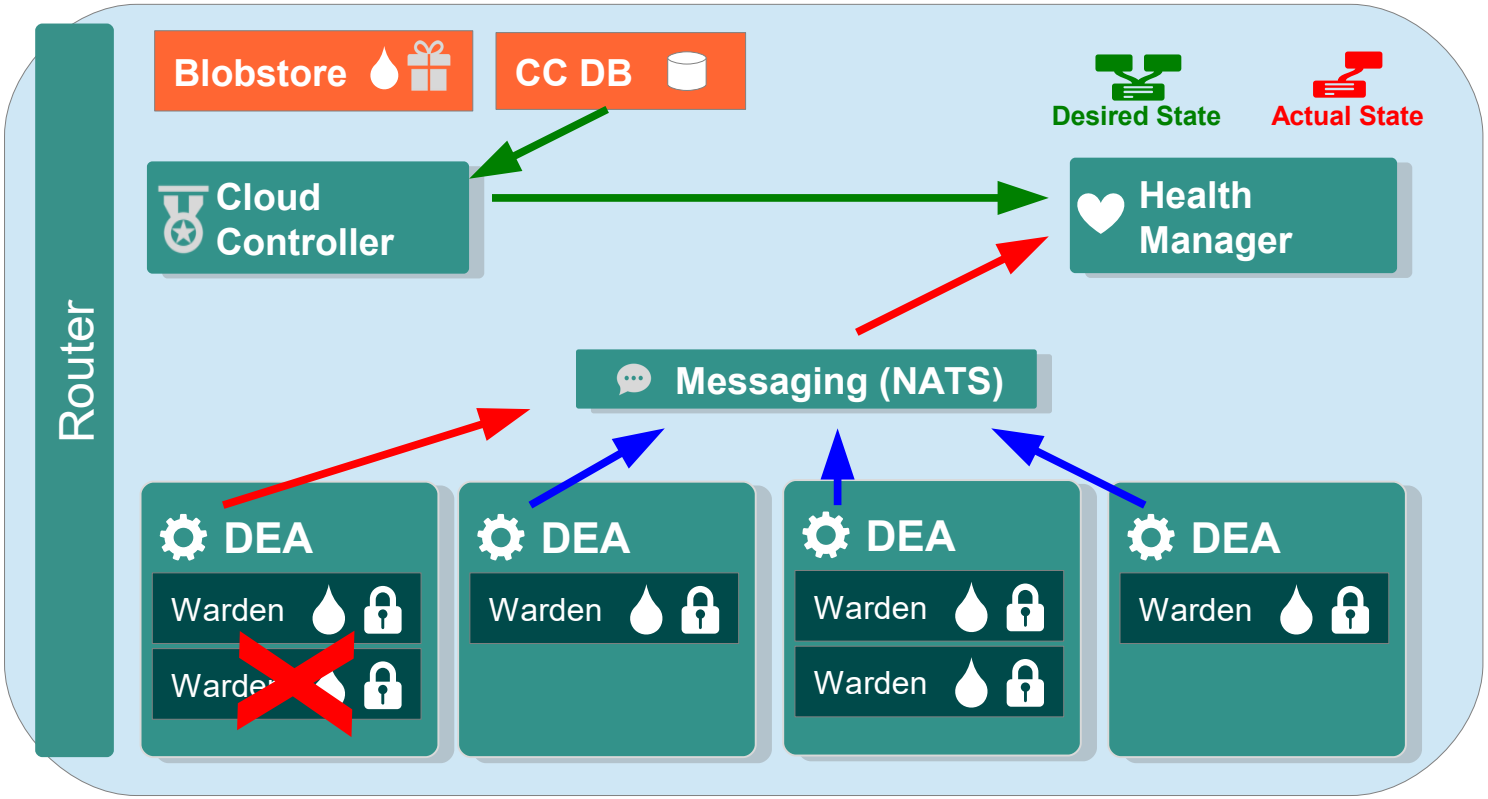
Health Management – DEAs

1. Applications run within containers on DEAs
2. DEAs send heartbeat messages, messages sent to Health Manager
3. Health Manager obtains “desired state” from Cloud Controller
4. Does “desired state” = “actual state”? Yes



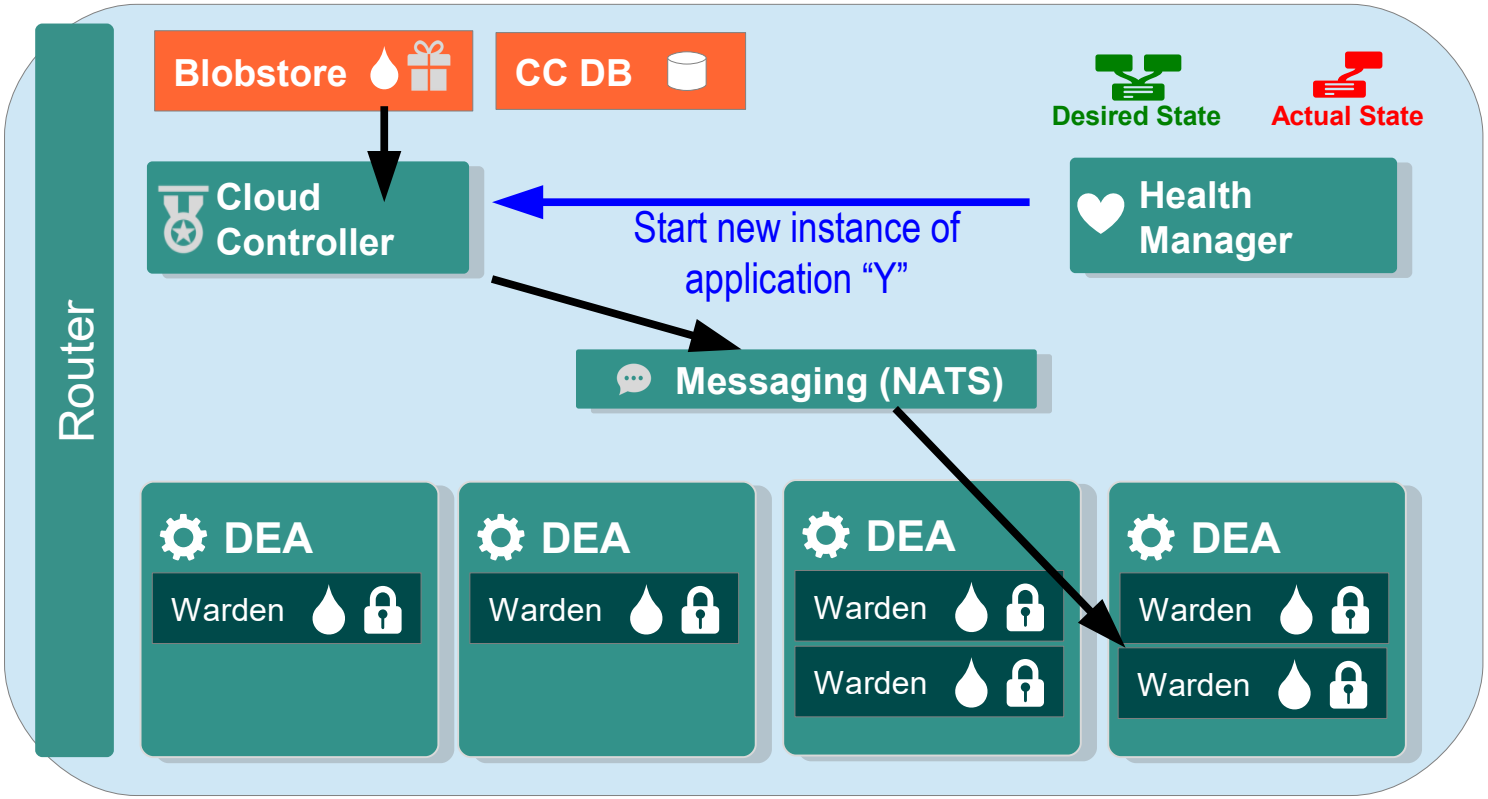
Health Management – Detecting Failures

1. What if an application crashes **X**?
2. DEAs send heartbeat messages, messages sent to Health Manager
3. Health Manager obtains “desired state” from Cloud Controller
4. “Desired state” = “actual state”? No



Health Management - Replacing an *Application*

1. Health Manager instructs Cloud Controller
2. Cloud Controller clones Droplet into container on DEA



Pivotal