Pivotal

BOSH Fundamentals

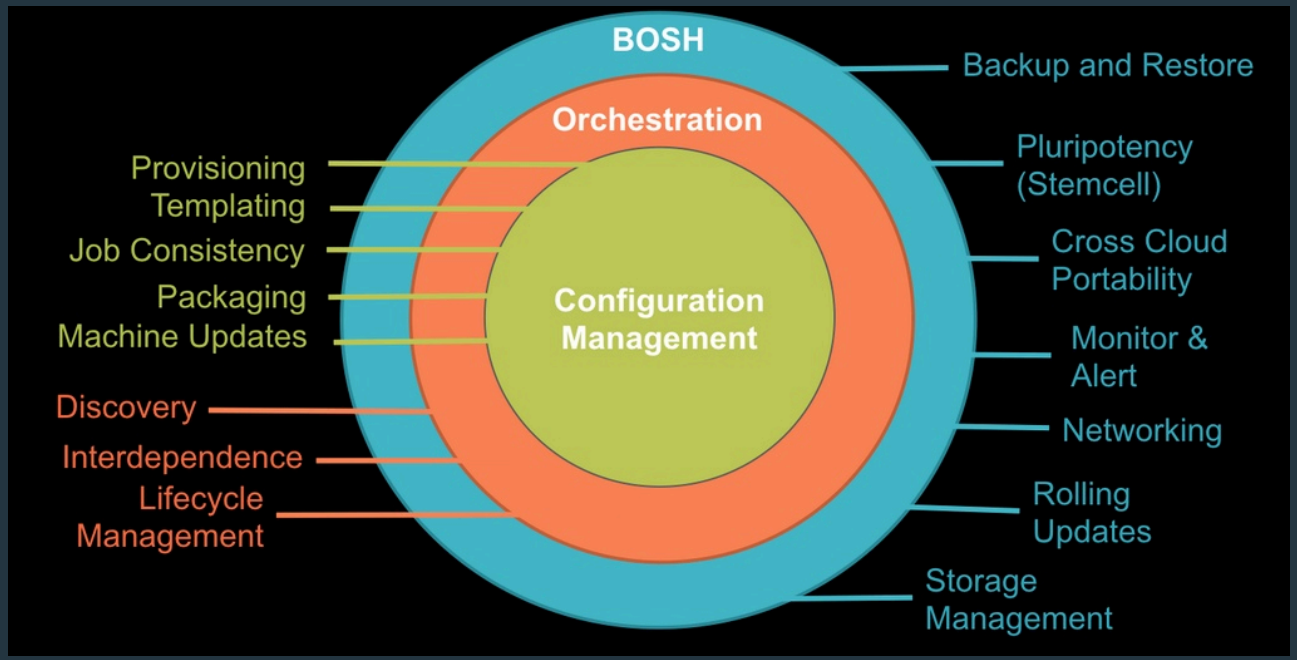
Vivian Fialho – Platform Architect
vfialho@pivotal.io

Agenda

- BOSH Overview
- BOSH Architecture
- BOSH Deployments and Releases
- BOSH HA and Platform Updates

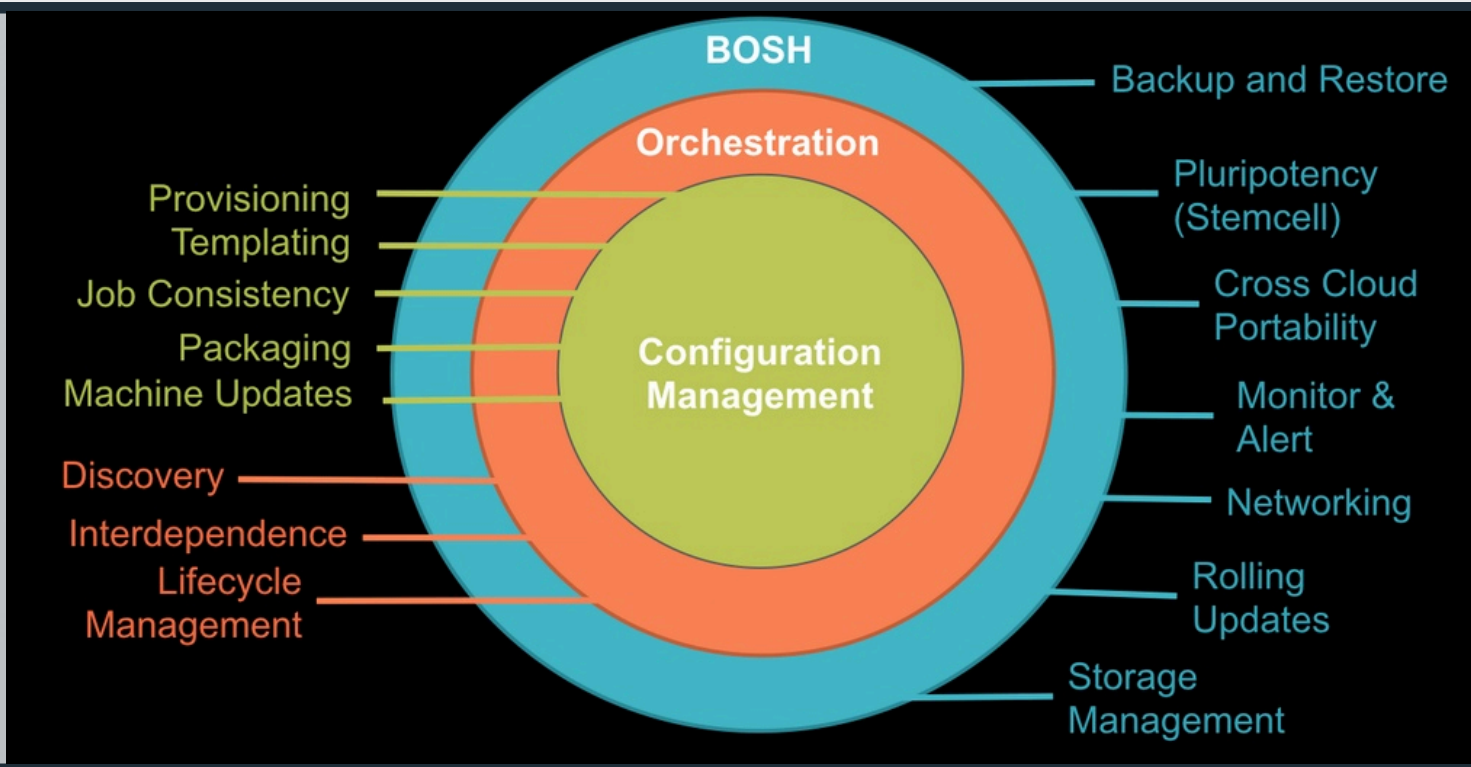
Pivotal

“ BOSH is an open-source tool chain for release engineering, deployment and lifecycle management of large-scale distributed services ”



Why BOSH?

- Provision services, not machines
- Enables continuous delivery
- Cloud-agnostic view of Platform Ops
- Holistic Toolchain for “rule them all"
- Eliminate bespoke automation on top of config management



Pivotal

CM & Orchestration tools

Pivotal

What is missing?

- These tools were designed to ensure servers are in a desired configuration, not to rapidly deploy software.
- Coordinating actions across different systems is left up to the user, requiring scripts or other tools.
- Manage services ...not servers or VMs

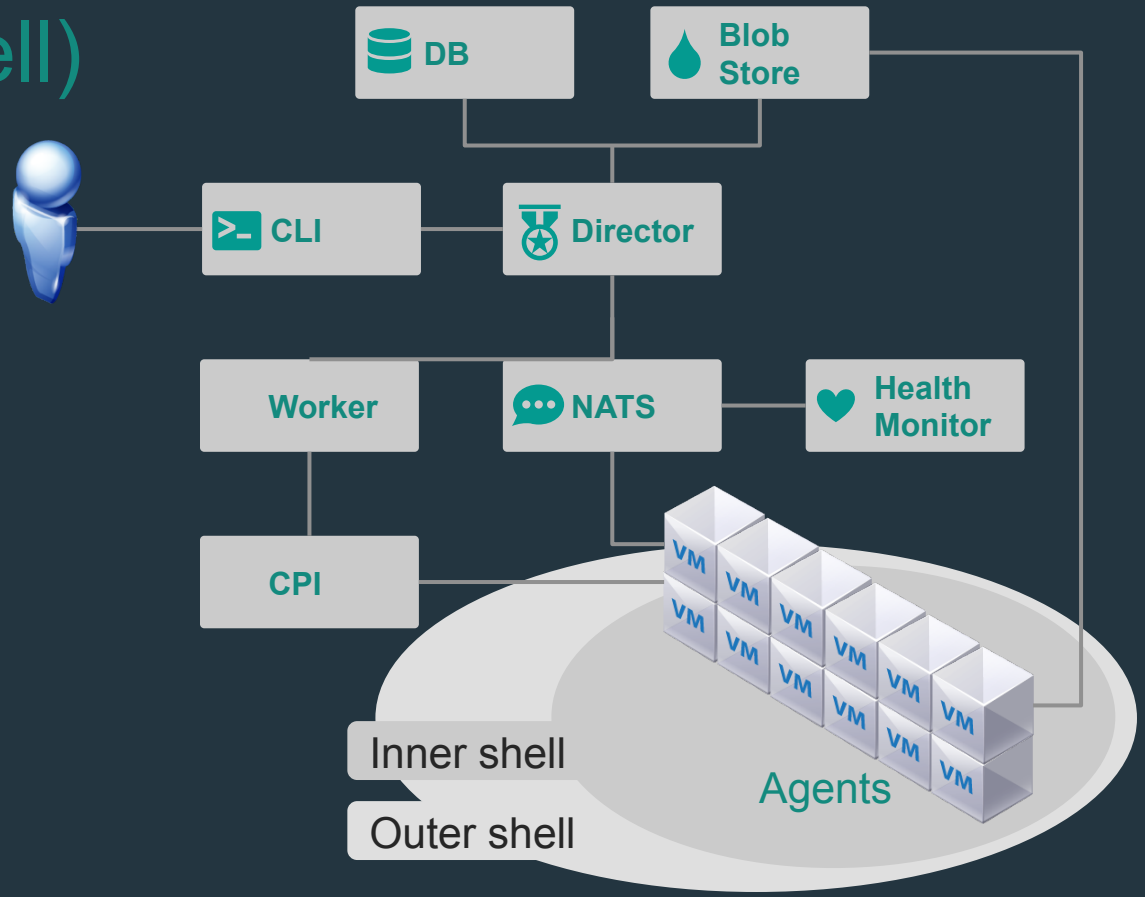
Pivotal

BOSH (Outer Shell) Logical View

Deploys and manages large scale distributed systems. **BOSH** provides the means to go from deployment (i.e., Chef/Puppet) to VM creation and management (i.e., cloud CPI). It includes interfaces for vSphere, vCloud, AWS and OpenStack. Additional CPI can be written for alternative IaaS providers.

Key Elements:

- CLI
- Director
- Blobstore
- Workers
- Message Bus
- Health Monitor
- IaaS CPI
- Agents



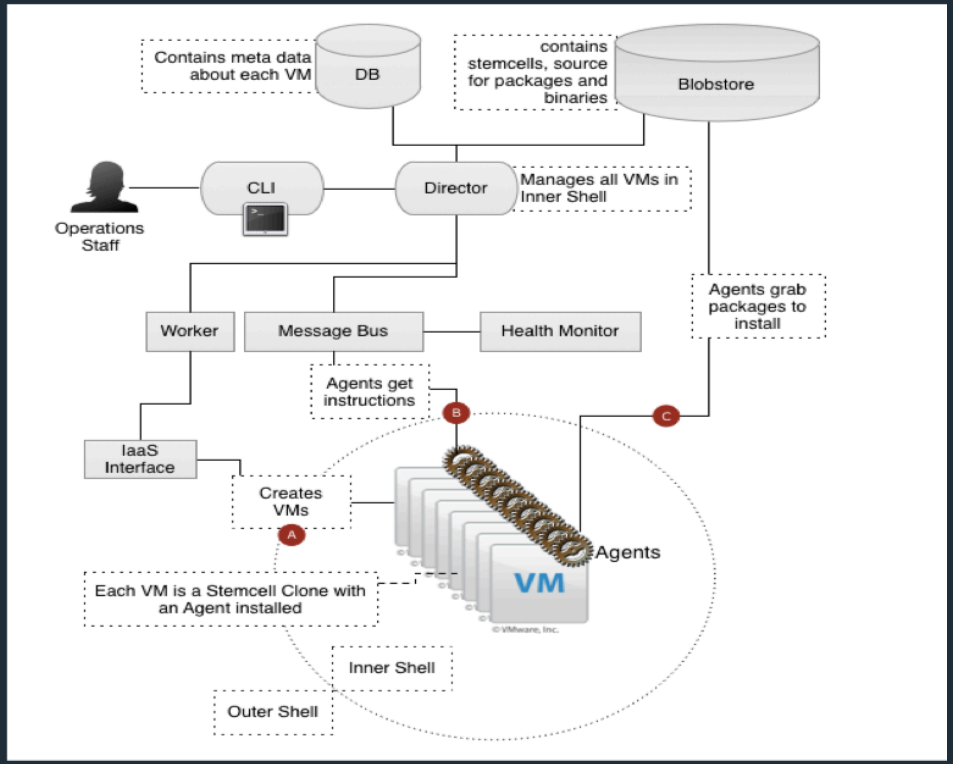
The BOSH Architecture

Very similar to CF architecture itself

Director as analogy to Cloud Controller

Different CPIs exist per IaaS implementation

Workers responsible for executing tasks as dictated by Director



BOSH: Cloud Provider Interface

Stemcell

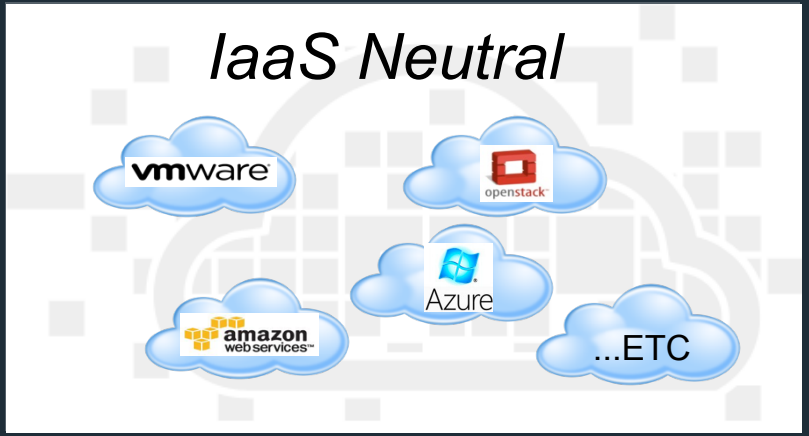
create_stemcell(image, cloud_properties)
delete_stemcell(stemcell_id)

VM

create_vm(agent_id, stemcell_id, resource_pool,
networks, disk_locality, env)
delete_vm(vm_id)
reboot_vm(vm_id)
configure_networks(vm_id, networks)

Disk

create_disk(size, vm_locality)
delete_disk(disk_id)
attach_disk(vm_id, disk_id)
detach_disk(vm_id, disk_id)

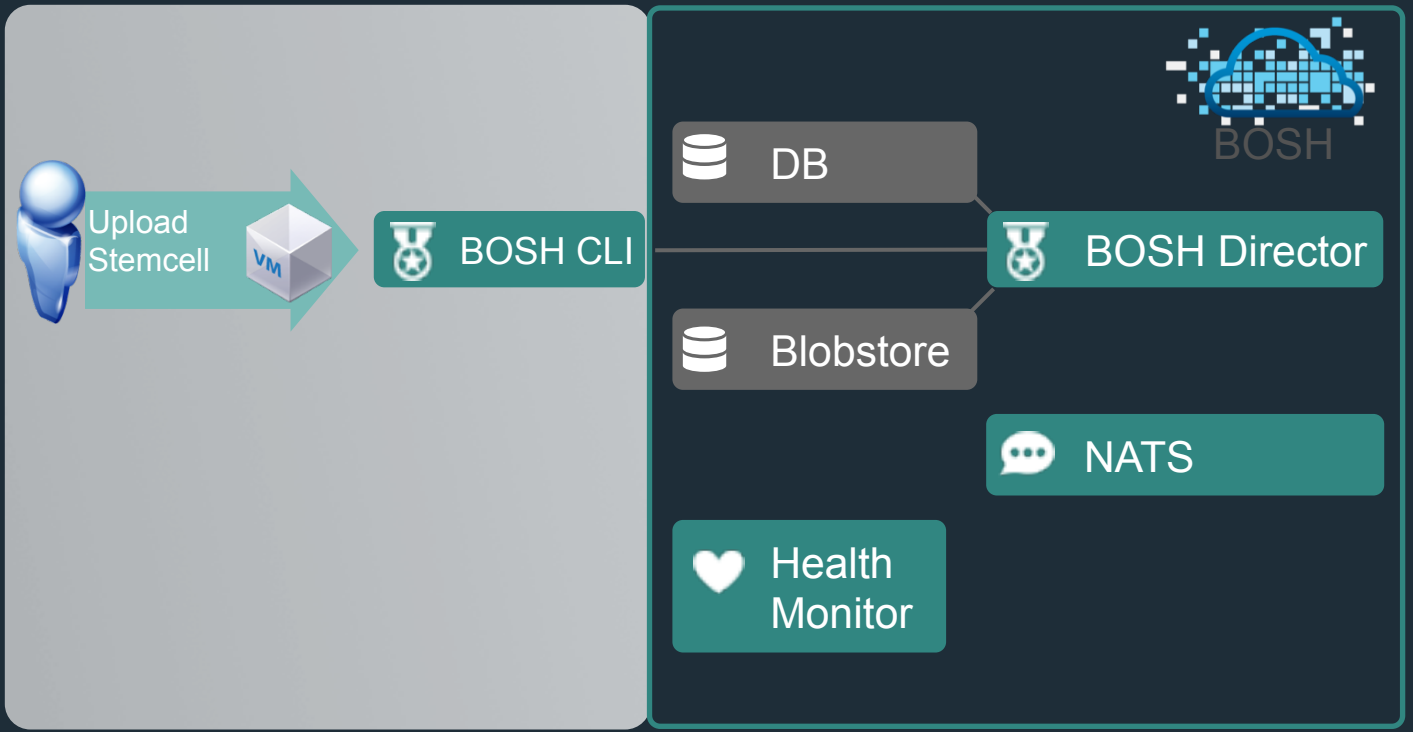


Pivotal

3 Components of a BOSH Deployment

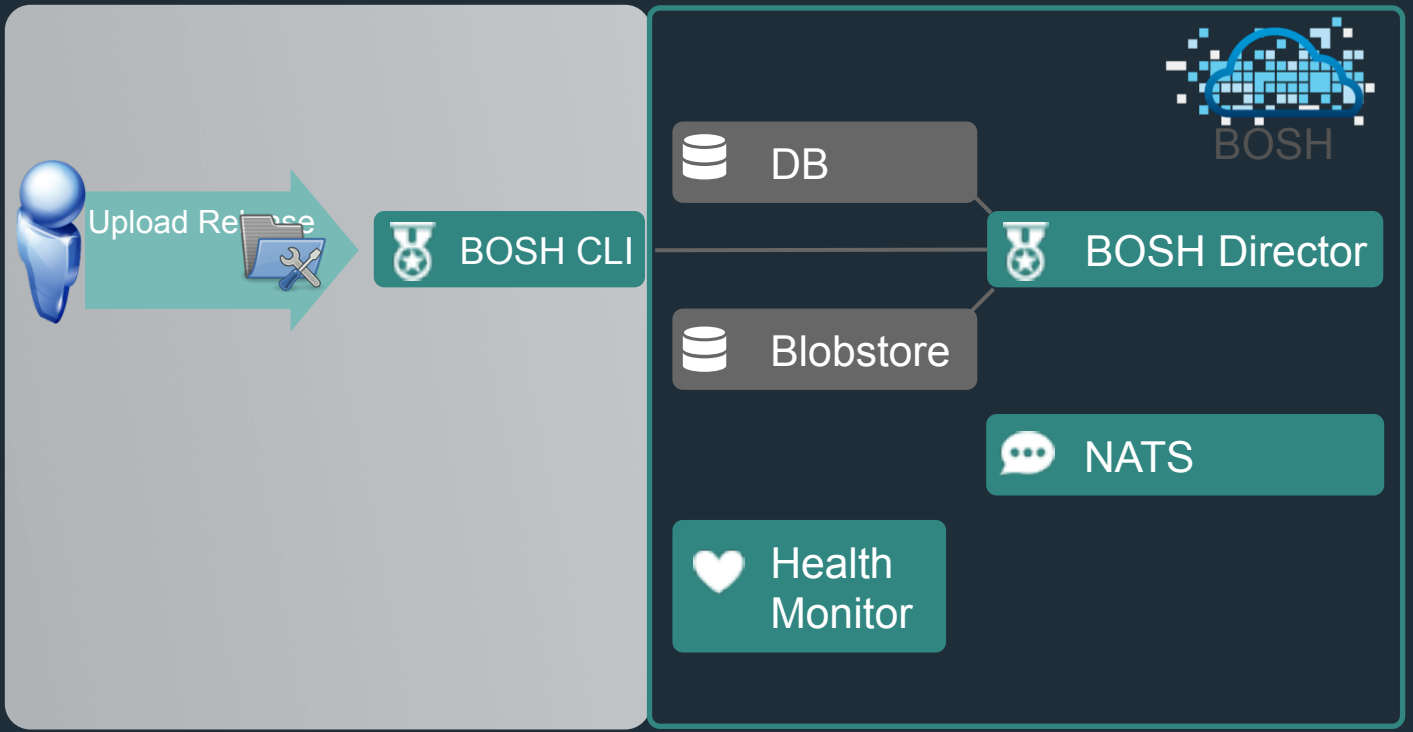
Pivotal

BOSH deployment



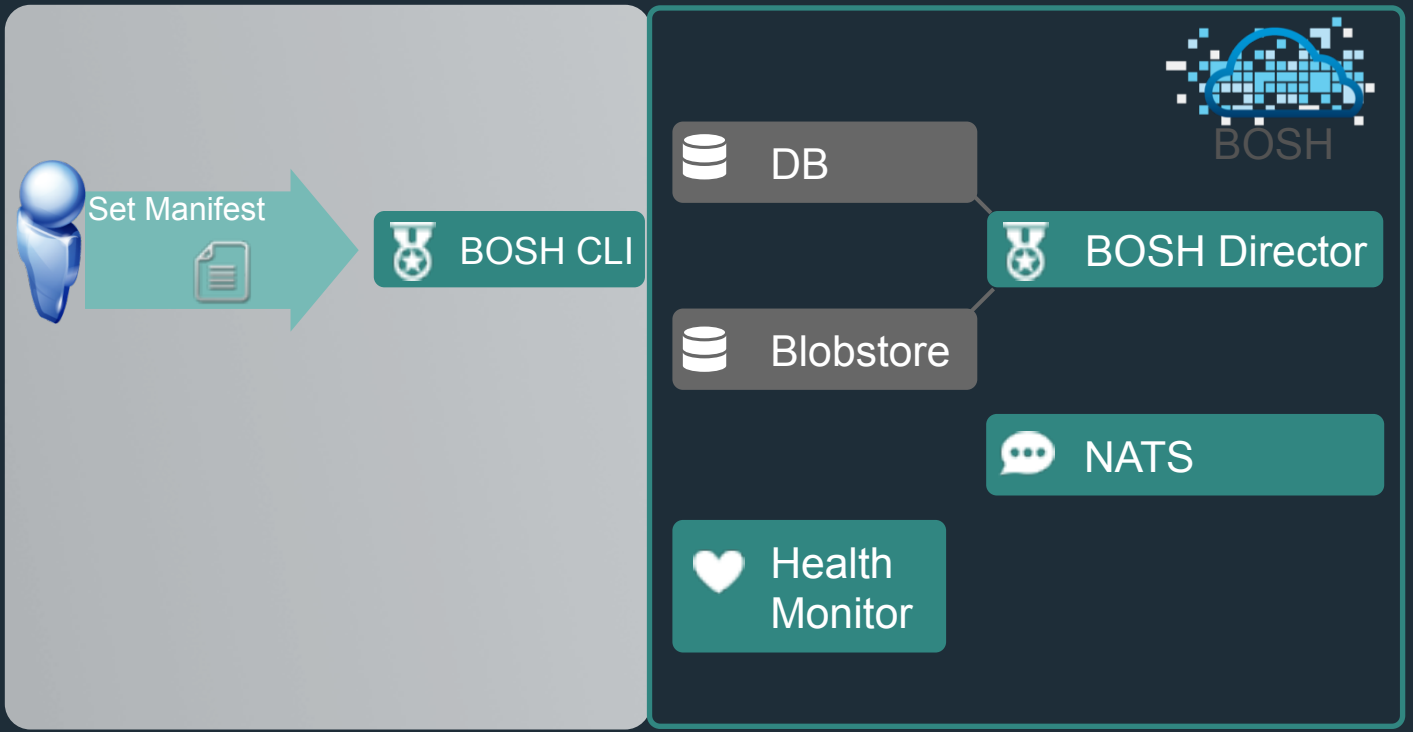
Pivotal

BOSH deployment



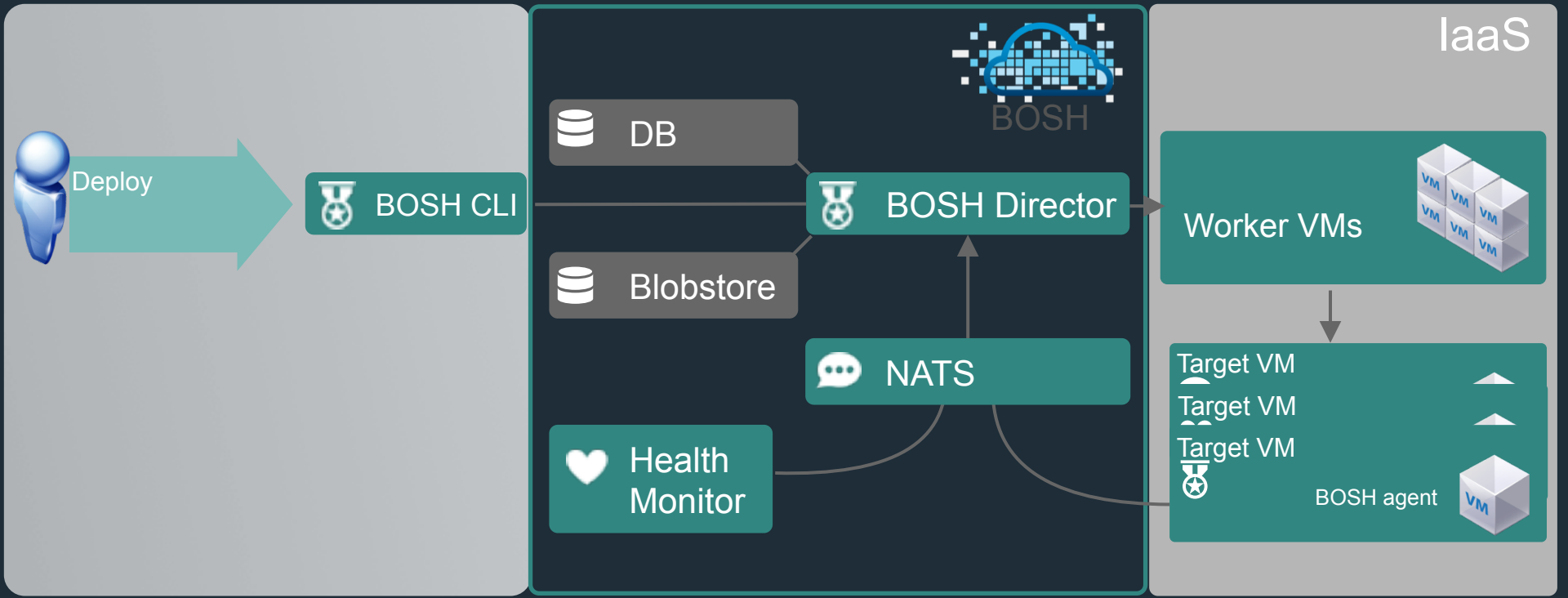
Pivotal

BOSH deployment



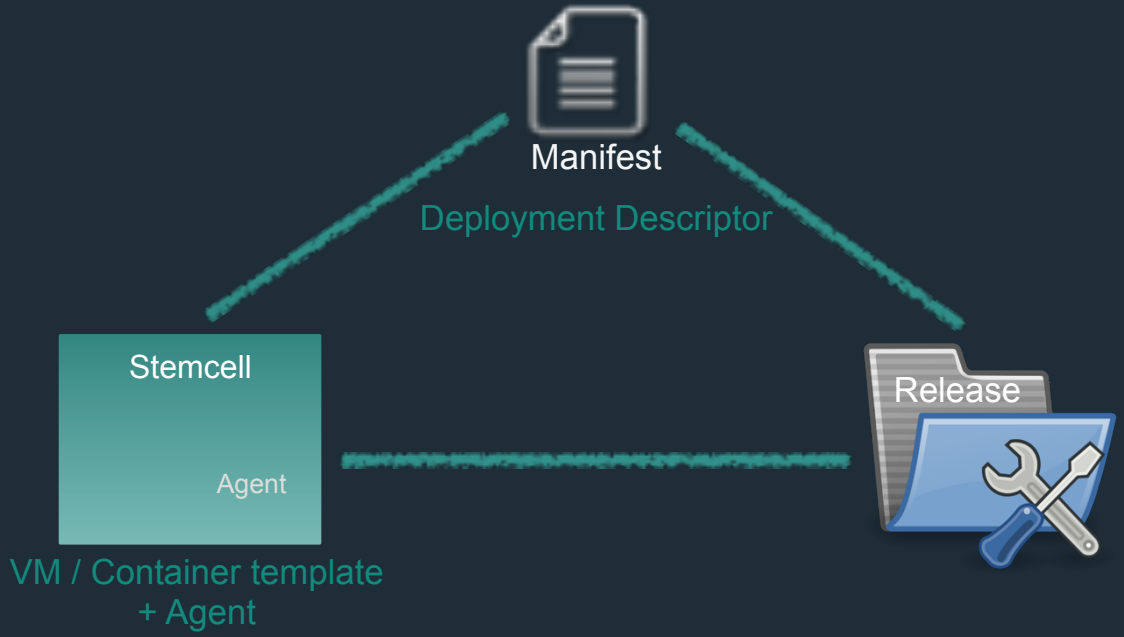
Pivotal

BOSH deployment



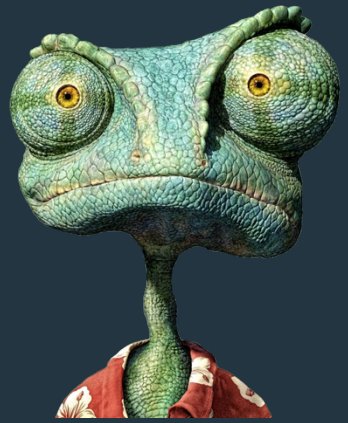
Pivotal

3 Components of a BOSH Deployment



Pivotal

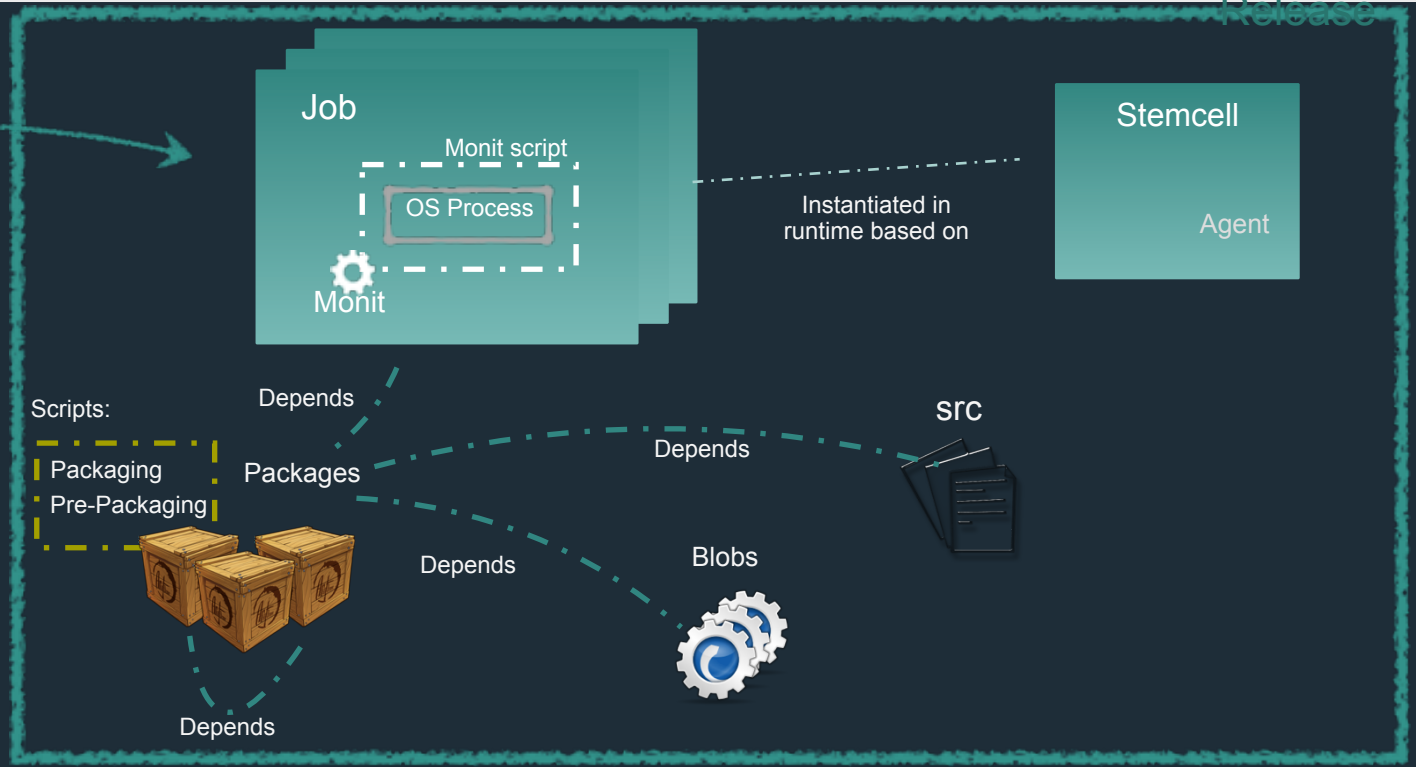
… so what exactly is a BOSH release?



Anatomy of BOSH releases

```
jobs:
- instances: 1
  name: postgres
  networks:
  - name: default-network
    static_ips:
    - 10.244.0.2
  persistent_disk: 2024
  release: bosh_intro
  resource_pool: default-pool
  template: postgres
  properties:
    port: 5637
```

Manifest



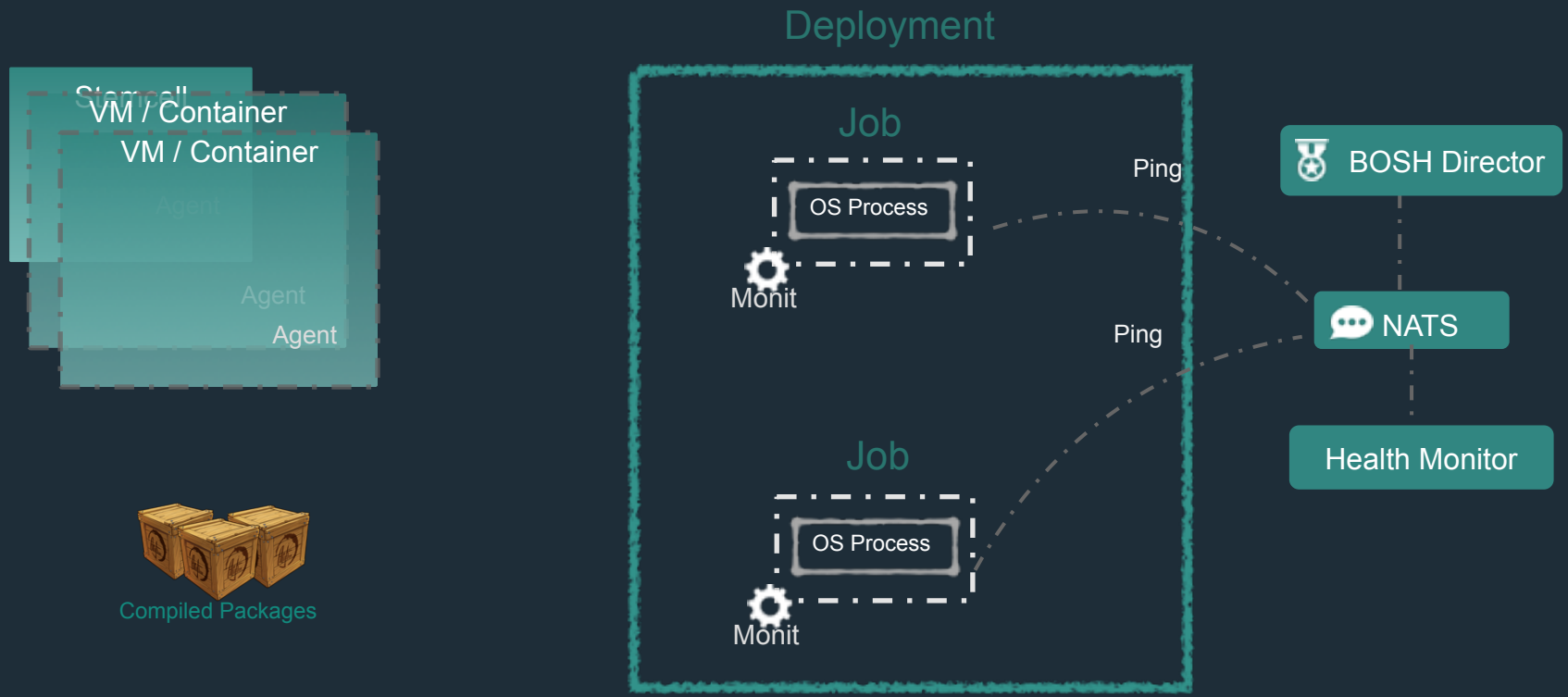
Example of Jobs:

- DEA (CF)
- CloudController (CF)
- Service Broker (MySQL & Rabbit)
- Locator (possible GemFire release)

http://docs.cloudfoundry.org/bosh/create-release.html

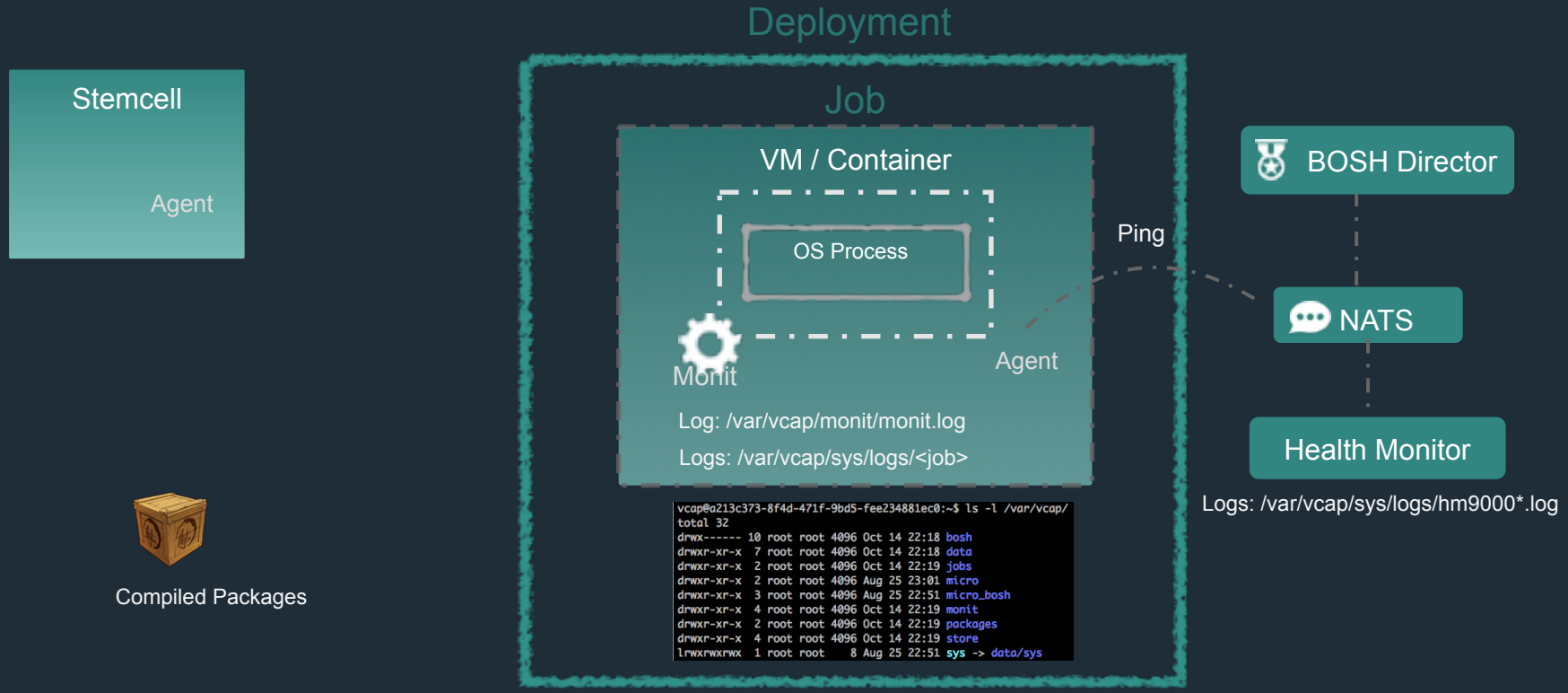
Pivotal

Job Creation



Pivotal

Logs



Pivotal


BOSH AND HIGH AVAILABILITY

4 Layers of Built-in High Availability

Application Instance

Platform Processes

Platform VMs

Availability Zones

Pivotal

4 Layers of Built-in High Availability

Application Instance

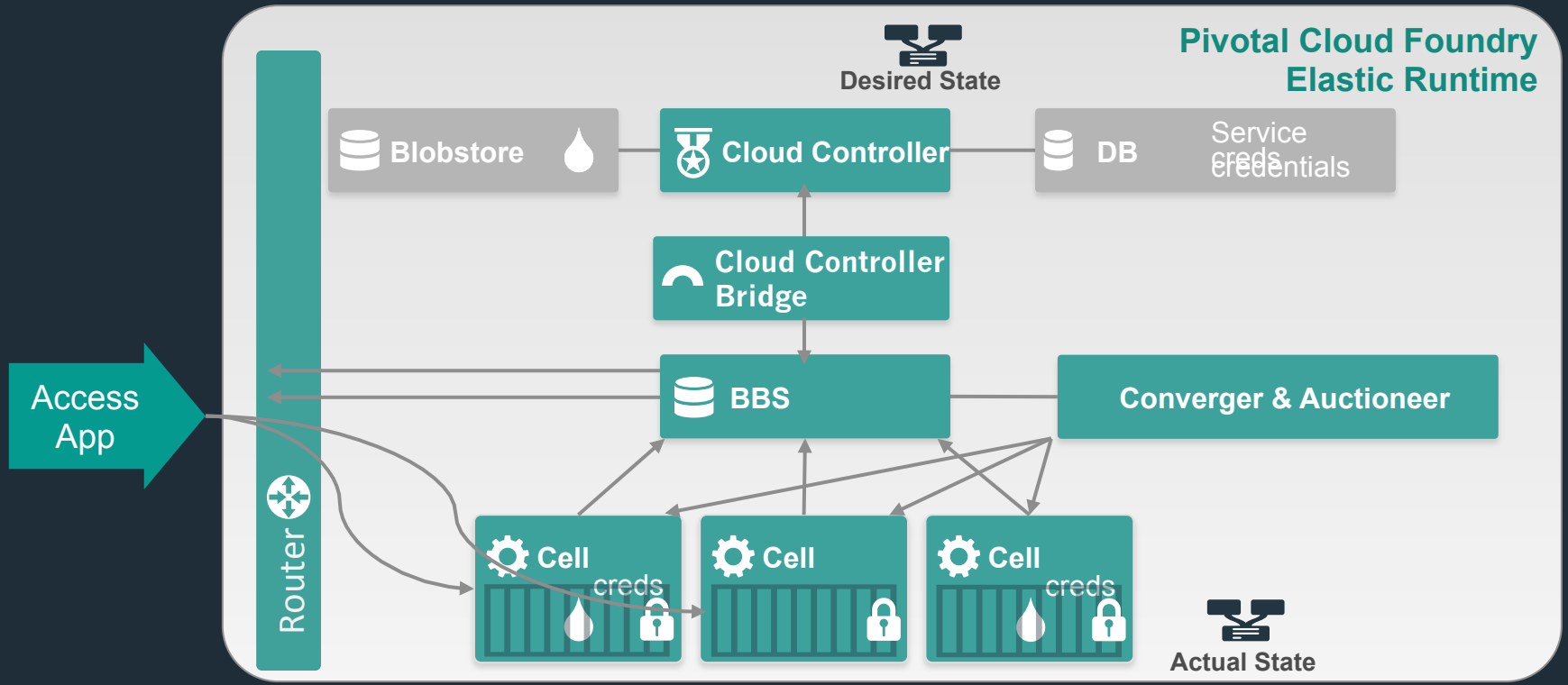
Platform Processes

Platform VMs

Availability Zones

Pivotal

Application Instance HA



Pivotal

4 Layers of Built-in High Availability

Application Instance

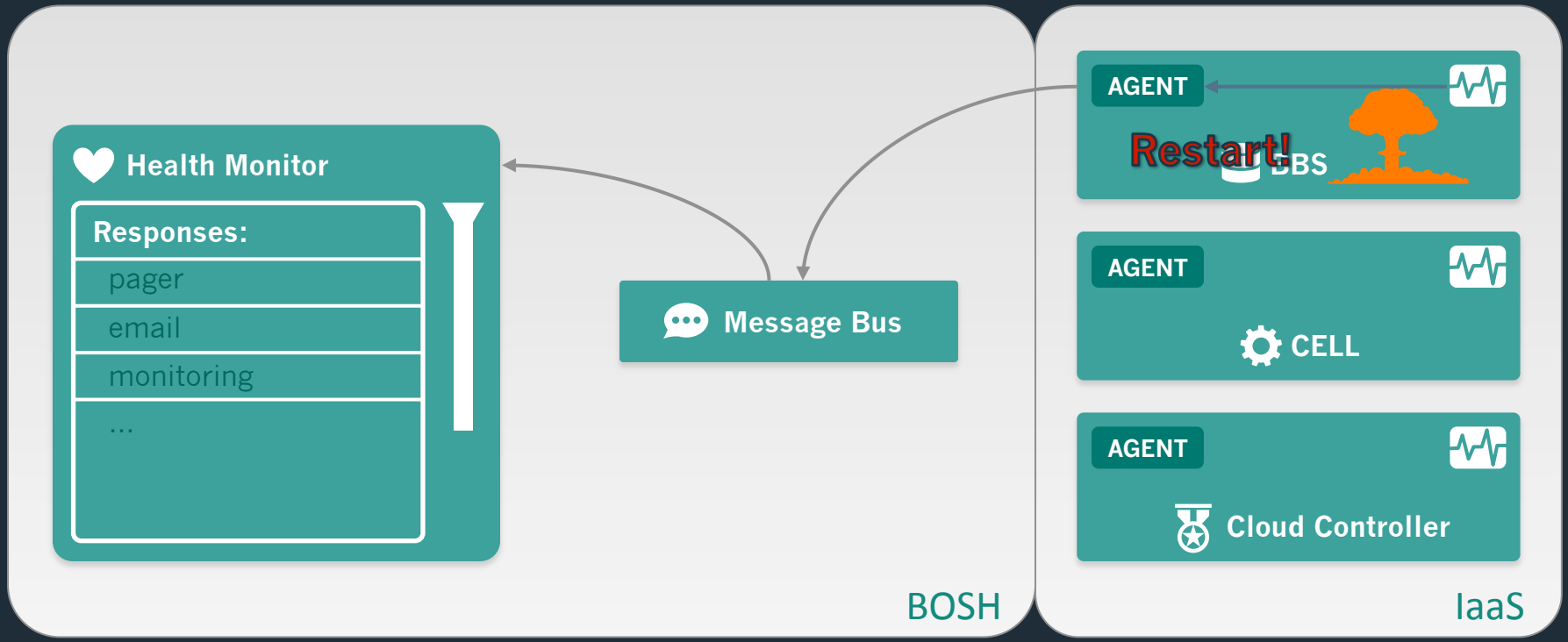
Platform Processes

Platform VMs

Availability Zones

Pivotal

Platform Process HA



Pivotal

4 Layers of Built-in High Availability

Application Instance

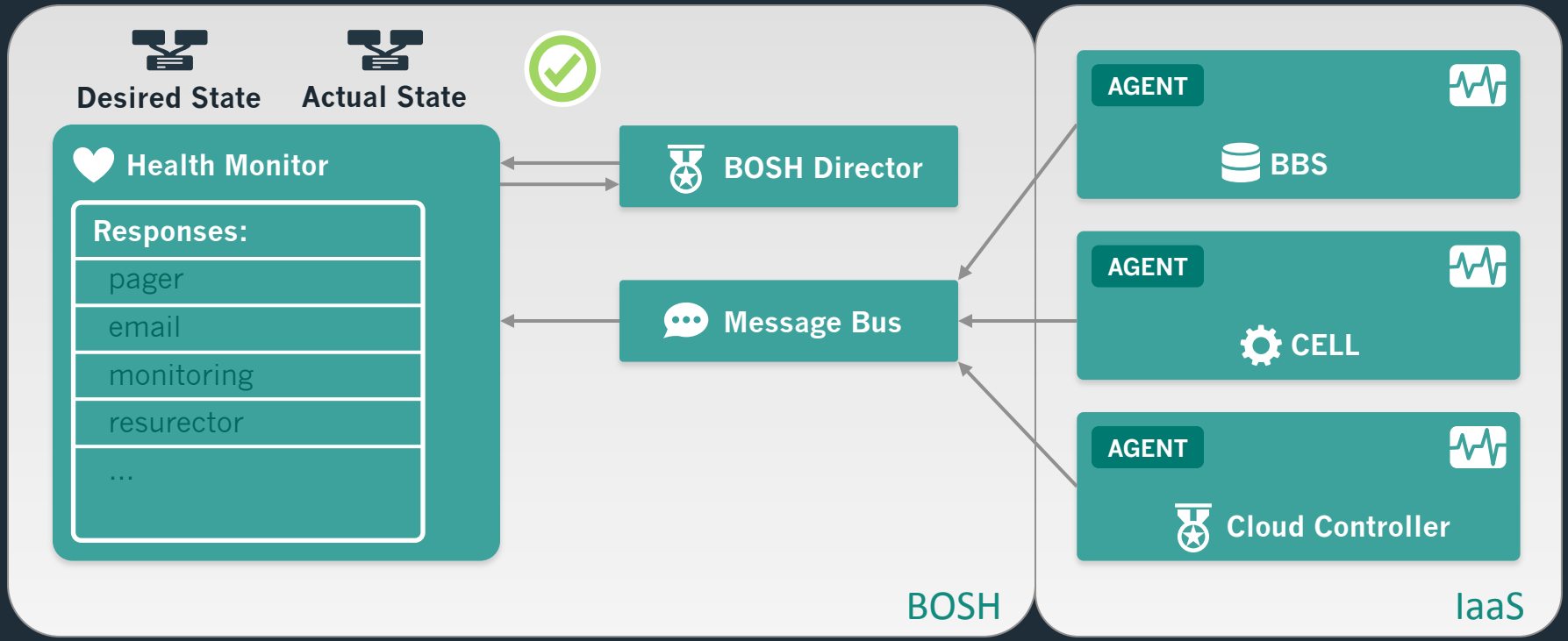
Platform Processes

Platform VMs

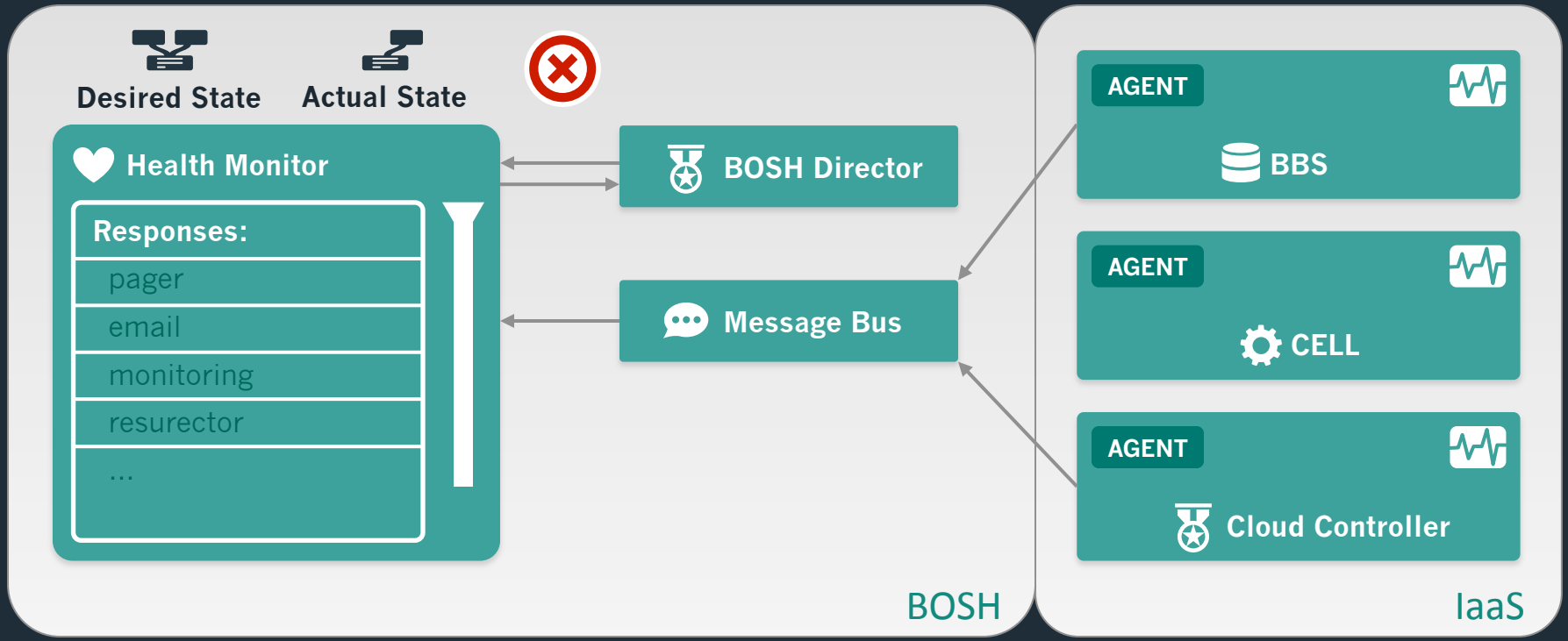
Availability Zones

Pivotal

Platform VM HA

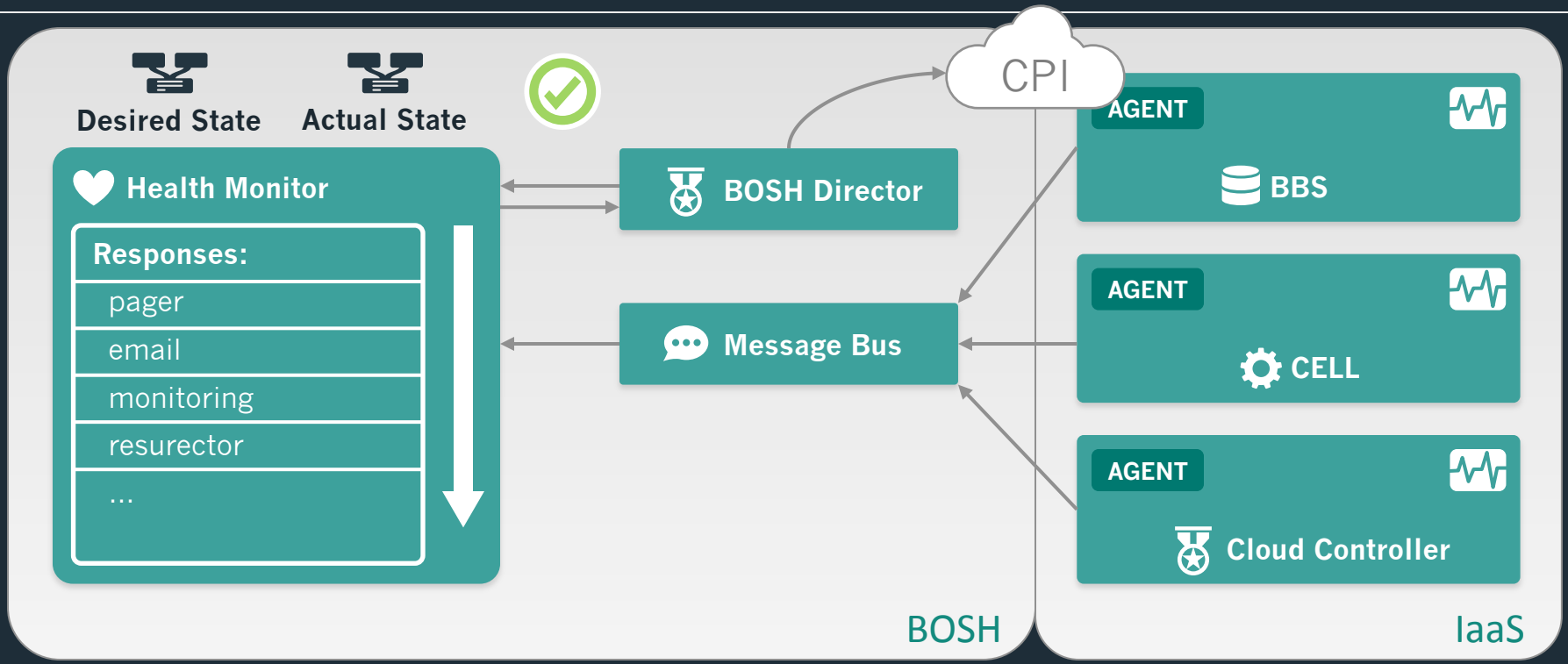


Platform VM HA



Pivotal

Platform VM HA



4 Layers of built-in High Availability

Application Instance

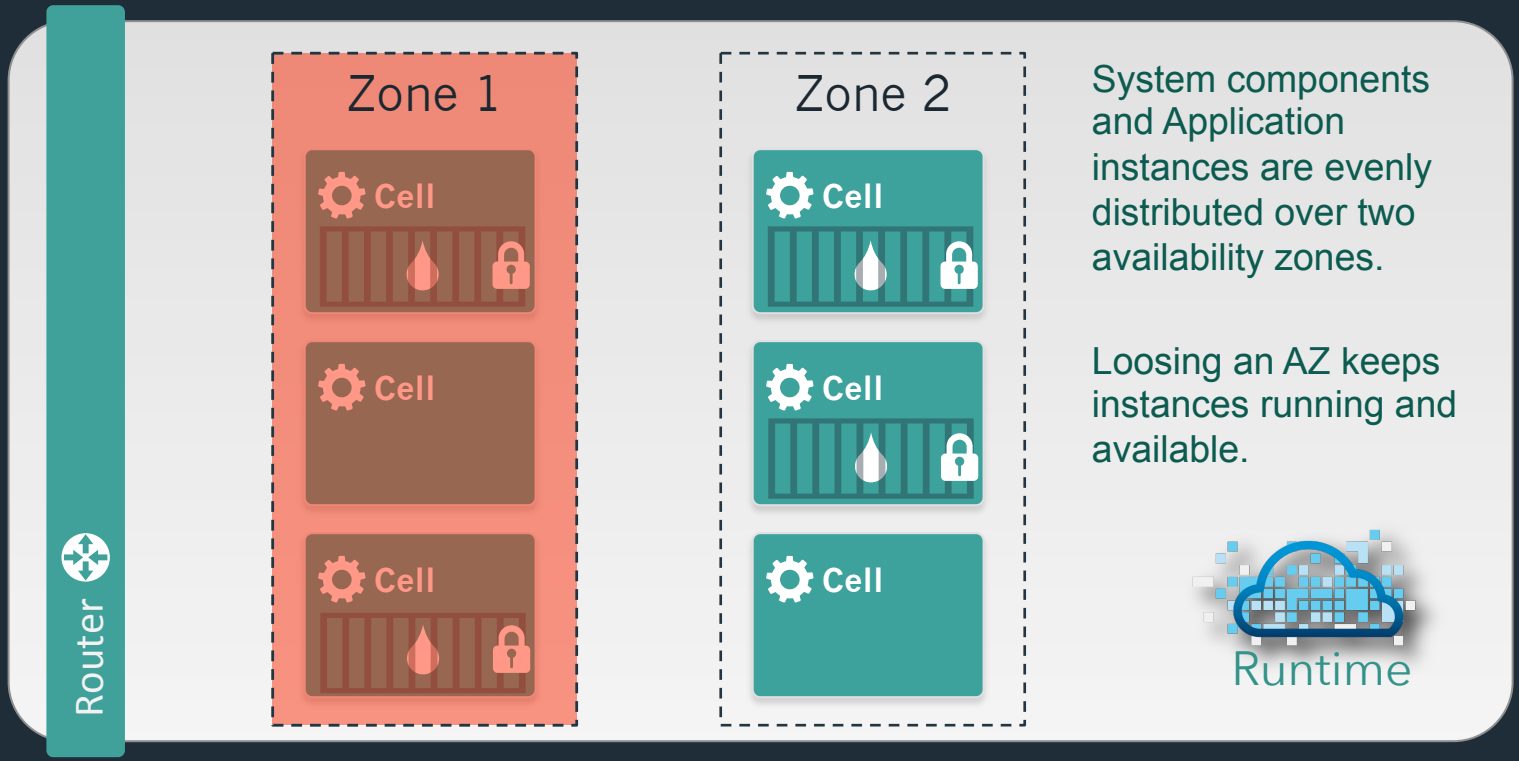
Platform Processes

Platform VMs

Availability Zones

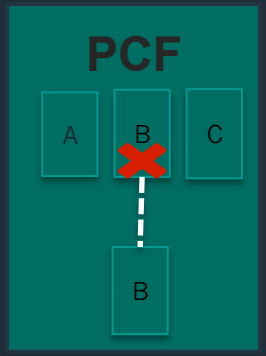
Pivotal

Availability Zones



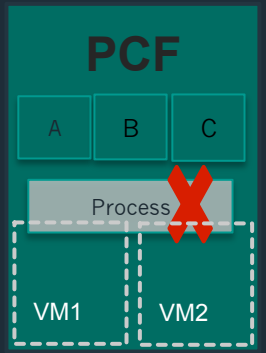
Pivotal

How Pivotal CF enables four layers of HA



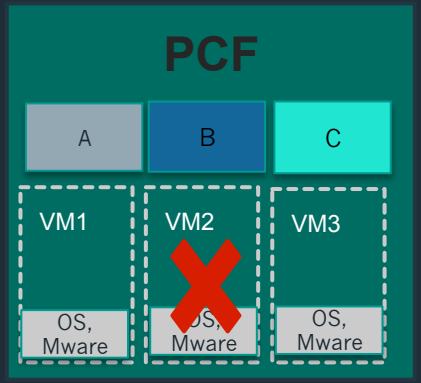
If an **app fails,** PCF reboots app in a new container.

App Fail



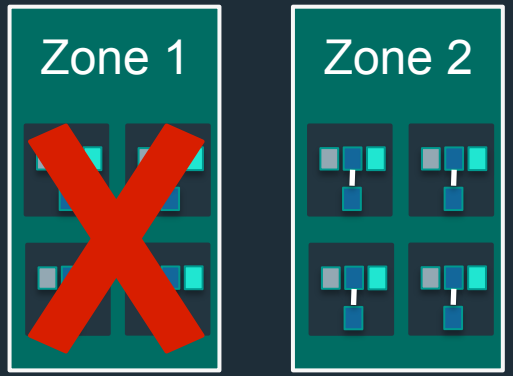
If a **process fails,** PCF restarts the process

Process Fail



If an **OS or network failure occurs** PCF kills the VM and reboots the host in a new Virtual machine.

VM Fail



If **a datacenter rack fails**, PCF ensures applications stay running in multiple availability zones

Rack Fail

Pivotal


BOSH-BASED PLATFORM UPDATES

Canary Deployments

```
update:
  canaries: 1
  canary_watch_time: 30000-300000
  max_errors: 2
  max_in_flight: 1
  update_watch_time: 30000-300000
```

Manifest

Pivotal

How Do Canary Deployments Work?

```
33 update:
34   canaries: 1
35   canary_watch_time: 30000-300000
36   max_errors: 2
37   max_in_flight: 1
38   update_watch_time: 30000-300000
```

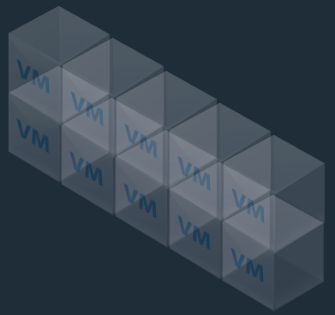
- `canaries` [Integer, required] Number of canary instances. Canary instances are being updated before other instances and any update error for canary instance means the deployment should stop. This prevents a buggy package or job from taking over all job instances, as only canaries will be affected by a problematic code. After canaries are done, other instances of this job will be updated in parallel (respecting `max_in_flight` setting).
- `canary_watch_time` [Range, Integer] How long to wait for canary update to declare job healthy or unhealthy. If Integer is given, director will sleep for that many seconds and check if job is healthy. If Range `lo..hi` is given it will wait for `lo` ms, see if job is healthy, and if it's not it will sleep some more, all up until `hi` ms have passed. If job is still unhealthy it will give up.
- `update_watch_time` [Range Integer]: Semantically no different from `canary_watch_time`, used for regular (non-canary) updates.
- `max_in_flight` [Integer, required] Maximum number of non-canary instance updates that can happen in parallel.

Pivotal

How Do Canary Deployments Work?

```
33 update:
34   canaries: 1
35   canary_watch_time: 30000-300000
36   max_errors: 2
37   max_in_flight: 1
38   update_watch_time: 30000-300000
```

No downtime, atomic rolling update



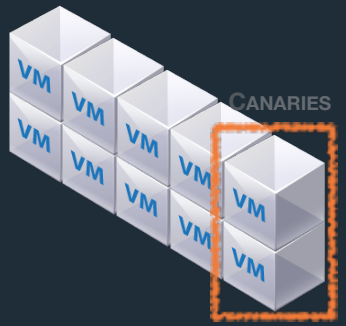
Pivotal

How Do Canary Deployments Work?

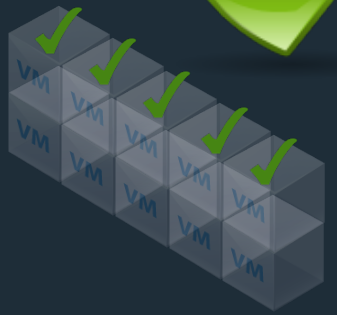
EXAMPLE:

OF CANARIES: 2

MAX IN FLIGHT: 2



v1.0



v1.1

Pivotal

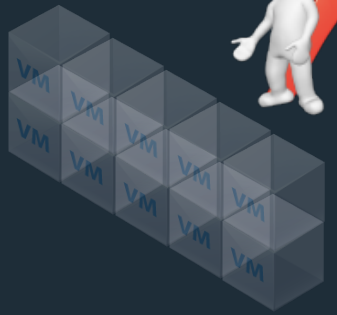
How Do Canary Deployments Work?

EXAMPLE:

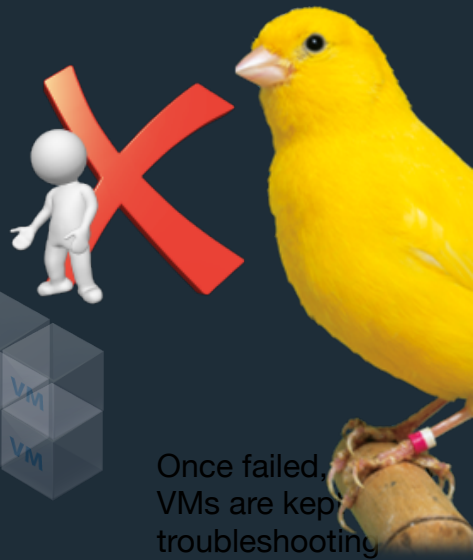
OF CANARIES: 2

MAX IN FLIGHT: 2

v1.1



v1.2



Once failed, VMs are kept for troubleshooting purposes.

Pivotal

How Do Canary Deployments Work?

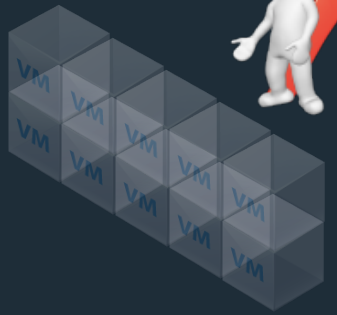
EXAMPLE:

OF CANARIES: 2

MAX IN FLIGHT: 2

v1.1

Once failed, Canary VMs are kept for troubleshooting purposes.

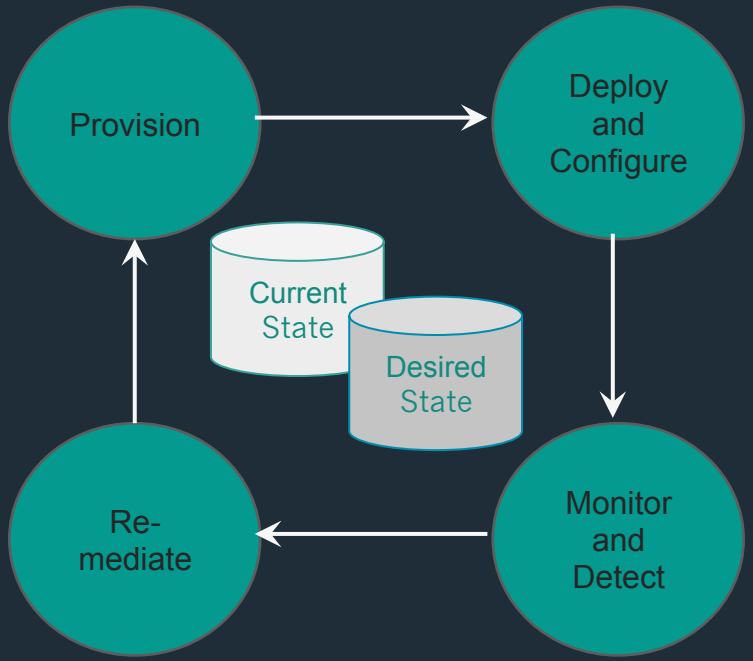


v1.2

Pivotal

BOSH – Purpose-built for “Day 2 Operations”

Consistent, Reliable, Scalable, Secure



- Checks against “desired state to return consistency
- No ad hoc automation burden
- Manage services, not servers
- 4 layers of Self Healing

Pivotal

Pivotal

Troubleshooting a BOSH deployment

BOSH tasks

bosh tasks recent

```
➜ ~ bosh tasks recent
Acting as user 'admin' on 'Bosh Lite Director'

+----+-------+----------------------+-------+-----------------+-----------------------------------------------------------------------------------------+
| #  | State | Timestamp            | User  | Description     | Result                                                                                  |
+----+-------+----------------------+-------+-----------------+-----------------------------------------------------------------------------------------+
| 10 | done  | 2016-01-10 20:16:16 UTC | admin | create deployment | /deployments/redis                                                                    |
| 8  | done  | 2016-01-10 20:04:40 UTC | admin | create deployment | /deployments/redis                                                                    |
| 7  | error | 2016-01-10 20:02:55 UTC | admin | create deployment | `redis-master/0' asked for a static IP 10.244.0.1 but it's already reserved/in...     |
| 6  | error | 2016-01-10 20:02:39 UTC | admin | create deployment | Required property `networks' was not specified in object...                           |
| 5  | error | 2016-01-10 20:01:59 UTC | admin | create deployment | Property `name' (value nil) did not match the required type `String'                  |
| 4  | error | 2016-01-10 20:01:29 UTC | admin | create deployment | Required property `update' was not specified in object ({"name"=>"redis",...         |
| 2  | done  | 2016-01-10 03:26:34 UTC | admin | create release    | Created release `redis/9.1'                                                           |
| 1  | done  | 2016-01-10 03:08:17 UTC | admin | create stemcell   | /stemcells/bosh-warden-boshlite-ubuntu-trusty-go_agent/3147                           |
+----+-------+----------------------+-------+-----------------+-----------------------------------------------------------------------------------------+
```

Pivotal

BOSH tasks

bosh tasks recent –no-filter

```
➜ ~ bosh tasks recent --no-filter
Acting as user 'admin' on 'Bosh Lite Director'

+----+-------+----------------------+----------+-----------------------------------------------------------------------+-----------------------------------------------------------------------------------------------+
| #  | State | Timestamp            | User     | Description                                                           | Result                                                                                        |
+----+-------+----------------------+----------+-----------------------------------------------------------------------+-----------------------------------------------------------------------------------------------+
| 21 | done  | 2016-01-10 23:56:26 UTC | admin    | scan cloud                                                            | scan complete                                                                                 |
| 20 | done  | 2016-01-10 23:50:25 UTC | admin    | retrieve vm-stats                                                     |                                                                                               |
| 19 | done  | 2016-01-10 23:49:25 UTC | admin    | retrieve vm-stats                                                     |                                                                                               |
| 18 | done  | 2016-01-10 23:48:59 UTC | admin    | retrieve vm-stats                                                     |                                                                                               |
| 17 | done  | 2016-01-10 22:36:55 UTC | admin    | scan and fix                                                          | scan and fix complete                                                                         |
| 16 | done  | 2016-01-10 22:36:54 UTC | admin    | scan and fix                                                          | scan and fix complete                                                                         |
| 15 | done  | 2016-01-10 22:36:54 UTC | admin    | scan and fix                                                          | scan and fix complete                                                                         |
| 14 | done  | 2016-01-10 20:31:35 UTC | admin    | ssh: setup:{"job"=>"redis-master", "indexes"=>["0"], "ids"=>["0"]}    |                                                                                               |
| 13 | done  | 2016-01-10 20:28:16 UTC | admin    | retrieve vm-stats                                                     |                                                                                               |
| 12 | done  | 2016-01-10 20:28:06 UTC | admin    | retrieve vm-stats                                                     |                                                                                               |
| 11 | done  | 2016-01-10 20:27:31 UTC | admin    | retrieve vm-stats                                                     |                                                                                               |
| 10 | done  | 2016-01-10 20:16:16 UTC | admin    | create deployment                                                     | /deployments/redis                                                                            |
| 8  | done  | 2016-01-10 20:04:40 UTC | admin    | create deployment                                                     | /deployments/redis                                                                            |
| 9  | error | 2016-01-10 20:04:39 UTC | admin    | ssh: setup:{"job"=>"redis", "indexes"=>[], "ids"=>[]}                 | No instances matched {:deployment_id=>1, :job=>"redis"}                                       |
| 7  | error | 2016-01-10 20:02:55 UTC | admin    | create deployment                                                     | `redis-master/0' asked for a static IP 10.244.0.1 but it's already reserved/in...             |
| 6  | error | 2016-01-10 20:02:39 UTC | admin    | create deployment                                                     | Required property `networks' was not specified in object...                                   |
| 5  | error | 2016-01-10 20:01:59 UTC | admin    | create deployment                                                     | Property `name' (value nil) did not match the required type `String'                          |
| 4  | error | 2016-01-10 20:01:29 UTC | admin    | create deployment                                                     | Required property `update' was not specified in object ({"name"=>"redis",...                  |
| 3  | done  | 2016-01-10 07:00:02 UTC | scheduler | scheduled SnapshotDeployments                                         | Enqueued snapshot tasks []                                                                    |
| 2  | done  | 2016-01-10 03:26:34 UTC | admin    | create release                                                        | Created release `redis/9.1'                                                                   |
| 1  | done  | 2016-01-10 03:08:17 UTC | admin    | create stemcell                                                       | /stemcells/bosh-warden-boshlite-ubuntu-trusty-go_agent/3147                                   |
+----+-------+----------------------+----------+-----------------------------------------------------------------------+-----------------------------------------------------------------------------------------------+
```

Pivotal

Debug a BOSH task

bosh task 10 --debug

```
➜ ~  bosh task 10 --debug
Acting as user 'admin' on 'Bosh Lite Director'

Director task 10
I, [2016-01-10T20:15:40.491686 #1759] [0x3f9453c45334]  INFO -- TaskHelper: Director Version: 1.3074.0
I, [2016-01-10T20:15:40.491731 #1759] [0x3f9453c45334]  INFO -- TaskHelper: Enqueuing task: 10
I, [2016-01-10 20:15:40 #22555] []  INFO -- DirectorJobRunner: Looking for task with task id 10
D, [2016-01-10 20:15:40 #22555] [] DEBUG -- DirectorJobRunner: (0.000491s) SELECT NULL
D, [2016-01-10 20:15:40 #22555] [] DEBUG -- DirectorJobRunner: (0.000183s) SELECT * FROM "tasks" WHERE "id" = 10
I, [2016-01-10 20:15:40 #22555] []  INFO -- DirectorJobRunner: Starting task: 10
I, [2016-01-10 20:15:40 #22555] [task:10]  INFO -- DirectorJobRunner: Creating job
D, [2016-01-10 20:15:40 #22555] [task:10] DEBUG -- DirectorJobRunner: (0.000134s) SELECT NULL
D, [2016-01-10 20:15:40 #22555] [task:10] DEBUG -- DirectorJobRunner: (0.000133s) SELECT * FROM "tasks" WHERE "id" = 10
I, [2016-01-10 20:15:40 #22555] [task:10]  INFO -- DirectorJobRunner: Performing task: 10
D, [2016-01-10 20:15:40 #22555] [task:10] DEBUG -- DirectorJobRunner: (0.000284s) SELECT NULL
D, [2016-01-10 20:15:40 #22555] [task:10] DEBUG -- DirectorJobRunner: (0.000060s) BEGIN
D, [2016-01-10 20:15:40 #22555] [task:10] DEBUG -- DirectorJobRunner: (0.000195s) UPDATE "tasks" SET "state" = 'processing', "times
 '/var/vcap/store/director/tasks/10', "checkpoint_time" = '2016-01-10 20:15:40.803556+0000', "type" = 'update_deployment', "usernam
D, [2016-01-10 20:15:40 #22555] [task:10] DEBUG -- DirectorJobRunner: (0.000457s) COMMIT
I, [2016-01-10 20:15:40 #22555] [task:10]  INFO -- DirectorJobRunner: Reading deployment manifest
D, [2016-01-10 20:15:40 #22555] [task:10] DEBUG -- DirectorJobRunner: Manifest:
---
name: redis
director_uuid: 9bca5701-66c0-43ee-a4d0-64813b4b5ece
releases:
- name: redis
  version: '9.1'
compilation:
  workers: 6
  network: default
  reuse_compilation_vms: true
  cloud_properties: {}
```

Pivotal

List VMs

bosh vms

```
➜  redis-boshrelease git:(master) ✗ bosh vms
Acting as user 'admin' on 'Bosh Lite Director'
Deployment `redis'

Director task 11

Task 11 done

+---------------+---------+---------+-----------+
| VM            | State   | VM Type | IPs       |
+---------------+---------+---------+-----------+
| redis-master/0 | running | default | 10.244.0.2 |
| redis-slave/0  | running | default | 10.244.0.3 |
| redis-slave/1  | running | default | 10.244.0.4 |
+---------------+---------+---------+-----------+

VMs total: 3
```

- running: indicates that all release job's processes are successfully running at that moment
- failing: indicates one of the release job's processes is not successfully running (could be failing to start, or exiting after some time, etc.)
- unresponsive: the Director did not receive any response from the Agent

Pivotal

Monitor VM's state and vitals

bosh vms --vitals

```
➜ redis-boshrelease git:(master) ✗ bosh vms --vitals
Acting as user 'admin' on 'Bosh Lite Director'
Deployment `redis'

Director task 13

Task 13 done

+---------------+---------+---------+-----------+----------------------+------+------+------+-------------+------------+-----------+-----------+-----------+
| VM            | State   | VM Type | IPs       | Load                 | CPU  | CPU  | CPU  | Memory Usage | Swap Usage | System    | Ephemeral | Persistent |
|               |         |         |           | (avg01, avg05, avg15) | User | Sys  | Wait |              |            | Disk Usage | Disk Usage | Disk Usage |
+---------------+---------+---------+-----------+----------------------+------+------+------+-------------+------------+-----------+-----------+-----------+
| redis-master/0 | running | default | 10.244.0.2 | 0.08, 0.07, 0.07     | 0.3% | 0.2% | 0.0% | 15% (920.4M) | 0% (0B)    | 2%        | 7%        | 2%        |
| redis-slave/0  | running | default | 10.244.0.3 | 0.08, 0.07, 0.07     | 0.3% | 0.2% | 0.0% | 15% (920.2M) | 0% (0B)    | 2%        | 7%        | 2%        |
| redis-slave/1  | running | default | 10.244.0.4 | 0.08, 0.07, 0.07     | 0.3% | 0.2% | 0.0% | 15% (920.2M) | 0% (0B)    | 2%        | 7%        | 2%        |
+---------------+---------+---------+-----------+----------------------+------+------+------+-------------+------------+-----------+-----------+-----------+

VMs total: 3
```

Pivotal

SSH into a VM

bosh ssh job/index

```
➜ redis-boshrelease git:(master) ✗ bosh ssh redis-master/0
Acting as user 'admin' on deployment 'redis' on 'Bosh Lite Director'
Target deployment is `redis'

Setting up ssh artifacts

Director task 14

Task 14 done
Starting interactive shell on job redis-master/0
Ubuntu 14.04.3 LTS \n \l

Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-29-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

bosh_ibezv8b77@5af1b21f-7a5a-4188-9cdd-40472db53397:~$ sudo su -
```

The Agent on each VM sends an alert when someone/something tries to log into the system via SSH. Successful and failed attempts are recorded.

Some IaaS set a “secret” root password. By default is “c1oudcw0” (SHHH!!!)

Pivotal

Setting a custom root password

```
resource_pools:
  - name: default
    network: default
    stemcell:
      name: bosh-warden-boshlite-ubuntu-trusty-go_agent
      version: latest
    cloud_properties: {}
    env:
      bosh:
        password: "$6$092f07e912df1395$guwt9rAPzwW3q/jaIq5vwu9"
```

Hashed password - Generate SHA hash using “mkpasswd -m sha-512”

Pivotal

Process health-check

"monit summary"
"monit (status|start|stop)"

https://mmonit.com/monit/

```
root@5af1b21f-7a5a-4188-9cdd-40472db53397:~# monit status
The Monit daemon 5.2.4 uptime: 43m

Process 'redis'
  status                            running
  monitoring status                 monitored
  pid                               2772
  parent pid                        1
  uptime                            43m
  children                          0
  memory kilobytes                  10892
  memory kilobytes total            10892
  memory percent                    0.1%
  memory percent total              0.1%
  cpu percent                       0.0%
  cpu percent total                 0.0%
  data collected                    Sun Jan 10 20:48:19 2016

System 'system_5af1b21f-7a5a-4188-9cdd-40472db53397'
  status                            running
  monitoring status                 monitored
  load average                      [0.06] [0.08] [0.06]
  cpu                               0.0%us 0.1%sy 0.0%wa
  memory usage                      934744 kB [15.2%]
  swap usage                        4 kB [0.0%]
  data collected                    Sun Jan 10 20:48:19 2016
```

Pivotal

Process recovery in action

If a process dies, the BOSH agent will restart it again automatically

```
root@5af1b21f-7a5a-4188-9cdd-40472db53397:~# ps -aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   8308  4352 ?        S<l  20:04   0:00 initd -dropCapabilities=false -title="wshd: 9vhck6dosv8"
root        17  0.0  0.0    196    40 ?        S<   20:04   0:00 runsvdir -P /etc/service log: ..........................................
root        23  0.0  0.0    176     4 ?        S<s  20:04   0:00 runsv agent
root        24  0.0  0.0    176     4 ?        S<s  20:04   0:00 runsv rsyslog
root        25  0.0  0.0    176     4 ?        S<s  20:04   0:00 runsv ssh
root        27  0.0  0.0    192     4 ?        S<   20:04   0:00 svlogd -tt /var/vcap/bosh/log
root        28  0.0  0.2 303824 15000 ?        S<l  20:04   0:00 /var/vcap/bosh/bin/bosh-agent -P ubuntu -C /var/vcap/bosh/agent.json
syslog      31  0.0  0.0 280696  3812 ?        S<l  20:04   0:00 rsyslogd -n
root        89  0.0  0.0    176     4 ?        S<s  20:04   0:00 runsv monit
root        90  0.0  0.0    192     4 ?        S<   20:04   0:00 svlogd -tt /var/vcap/monit/svlog
root        92  0.0  0.0  91484  2740 ?        S<l  20:04   0:00 /var/vcap/bosh/bin/monit -I -c /var/vcap/bosh/etc/monitrc
root        94  0.0  0.0  61376  5420 ?        S<   20:04   0:00 /usr/sbin/sshd -D
root      2772  0.1  0.1  36744 10892 ?        S<sl 20:04   0:04 redis-server *:6379
root      2800  0.0  0.0  93024  5764 ?        S<s  20:31   0:00 sshd: bosh_ibezv8b77 [priv]
bosh_ib+  2811  0.0  0.0  93024  2984 ?        S<   20:31   0:00 sshd: bosh_ibezv8b77@pts/0
bosh_ib+  2812  0.0  0.0  19792  3772 pts/0    S<s  20:31   0:00 -bash
root      2823  0.0  0.0  46560  3460 pts/0    S<   20:31   0:00 sudo su -
root      2824  0.0  0.0  48204  3100 pts/0    S<   20:31   0:00 su -
root      2825  0.0  0.0  19784  3788 pts/0    S<   20:31   0:00 -su
root      2837  0.0  0.0  17172  2640 pts/0    R<+  20:44   0:00 ps -aux
root@5af1b21f-7a5a-4188-9cdd-40472db53397:~# kill -9 2772
```

Pivotal

VM health-check

- The Health Monitor continiously checks the presence of the deployed VMs.
- The Agent on each VM produces a heartbeat every minute and sends it to the Health Monitor.
- The Health Monitor is extended by a set of plugins. Each plugin is given an opportunity to act on each heartbeat, so in cases of failure it can notify external services or perform actions against the Director.

Pivotal

Health Monitor plugins

- Event Logger: Logs events to a file
- Resurrector: Recreates VMs that have stopped heartbeating
- Emailer: Sends configurable e-mails on events reciept
- OpenTSDB: Sends events to OpenTSDB
- Graphite: Sends events to Graphite
- PagerDuty: Sends events to PagerDuty.com using their API
- DataDog: Sends events to DataDog.com using their API
- AWS CloudWatch: Sends events to Amazon's CloudWatch using their API

Configuring Health Monitor: http://bosh.io/docs/hm-config.html

Pivotal

Resurrector Health Monitor plugin

- It's responsible for automatically recreating VMs that become inaccessible.
- It continiously cross-references VMs expected to be running against the VMs that are sending heartbeats. When resurrector does not receive heartbeats for a VM for a certain period of time, it will kick off a task on the Director (scan and fix task) to try to "resurrect" that VM.
- Under certain conditions the Resurrector will consider the system in the "meltdown" and will stop sending requests to the Director.

Pivotal

Disabling manually the Resurrector

"bosh vm resurrection job index on|off"

```
➜ ~ bosh vm resurrection redis-slave 0 off
Acting as user 'admin' on deployment 'redis' on 'Bosh Lite Director'
➜ ~ bosh vms --details
Acting as user 'admin' on 'Bosh Lite Director'
Deployment `redis'

Director task 20

Task 20 done

+---------------+---------+---------+-----------+--------------------------------------+--------------------------------------+-------------+
| VM            | State   | VM Type | IPs       | CID                                  | Agent ID                             | Resurrection |
+---------------+---------+---------+-----------+--------------------------------------+--------------------------------------+-------------+
| redis-master/0 | running | default | 10.244.0.2 | e216a2f1-b3a1-4a06-7a46-f220ab967aca | 5af1b21f-7a5a-4188-9cdd-40472db53397 | active       |
| redis-slave/0  | running | default | 10.244.0.3 | 1634f786-a79e-49e1-7e27-0bb687f138c6 | edbfaab1-0fa8-4fd1-84ce-05eab4938f85 | paused       |
| redis-slave/1  | running | default | 10.244.0.4 | e4d96bff-92f8-4e64-6588-d7b272dadaca | 5c75a317-8514-42e2-ace2-026349c96370 | active       |
+---------------+---------+---------+-----------+--------------------------------------+--------------------------------------+-------------+

VMs total: 3
```

Pivotal

Manual repair with Cloud Check

BOSH provides the Cloud Check CLI command (a.k.a cck) to repair IaaS resources used by a specific deployment. It is not commonly used while normal operations; however, it becomes essential when some IaaS operations failed and the Director cannot resolve problems without a human decision or when the Resurrector is not enabled.

- VM is missing
- VM is not responsive (unresponsive agent)
- Persistent Disk is not attached
- Persistent Disk is missing

Pivotal

Manual repair with Cloud Check

bosh cck

```
➜ ~ bosh cck
Acting as user 'admin' on deployment 'redis' on 'Bosh Lite Director'
Performing cloud check...

Director task 21
  Started scanning 3 vms
  Started scanning 3 vms > Checking VM states. Done (00:00:00)
  Started scanning 3 vms > 3 OK, 0 unresponsive, 0 missing, 0 unbound, 0 out of sync. Done (00:00:00)
     Done scanning 3 vms (00:00:00)

  Started scanning 0 persistent disks
  Started scanning 0 persistent disks > Looking for inactive disks. Done (00:00:00)
  Started scanning 0 persistent disks > 0 OK, 0 missing, 0 inactive, 0 mount-info mismatch. Done (00:00:00)
     Done scanning 0 persistent disks (00:00:00)

Task 21 done

Started         2016-01-10 23:56:26 UTC
Finished        2016-01-10 23:56:26 UTC
Duration        00:00:00

Scan is complete, checking if any problems found...
No problems found
```

Pivotal

VM Configuration Locations

- /var/vcap/bosh/log/current: current BOSH agent log

- /var/vcap/monit/monit.log: Monit activity log. Includes information about starts, stops, restarts, etc. of release job processes monitored by Monit

Pivotal

VM Configuration Locations

- /var/vcap/jobs: contains evaluated release jobs for the assigned deployment job
- /var/vcap/packages: contains enabled release packages for the assigned deployment job
- /var/vcap/data: used by the release jobs to keep *ephemeral* data
- /var/vcap/store: used by the release jobs to keep *persistent* data

Pivotal

VM Configuration Locations

- /var/vcap/sys/run: directory that is used by the release jobs to keep miscellaneous ephemeral data about currently running processes, for example, pid and lock files
- /var/vcap/sys/log: directory that is used by the release jobs to keep logs

Pivotal

Log rotation

BOSH log rotates release job logs with the Logrotate log file management utility with the following non-configurable settings:

- missingok: Skip missing log files and do not generate an error message
- rotate 7: Keep seven log files at a time
- compress: Compress old log files with gzip
- delaycompress: Postpone compression of log files until the next rotation cycle
- copytruncate: Copy log files, then truncate in place instead of creating new files
- size 50M: Rotate log files when they exceed 50 MB in size.

Pivotal

Pivotal

Open.
Agile.
Cloud-Ready.