

Pivotal.

# Anatomy of a BOSH deployment



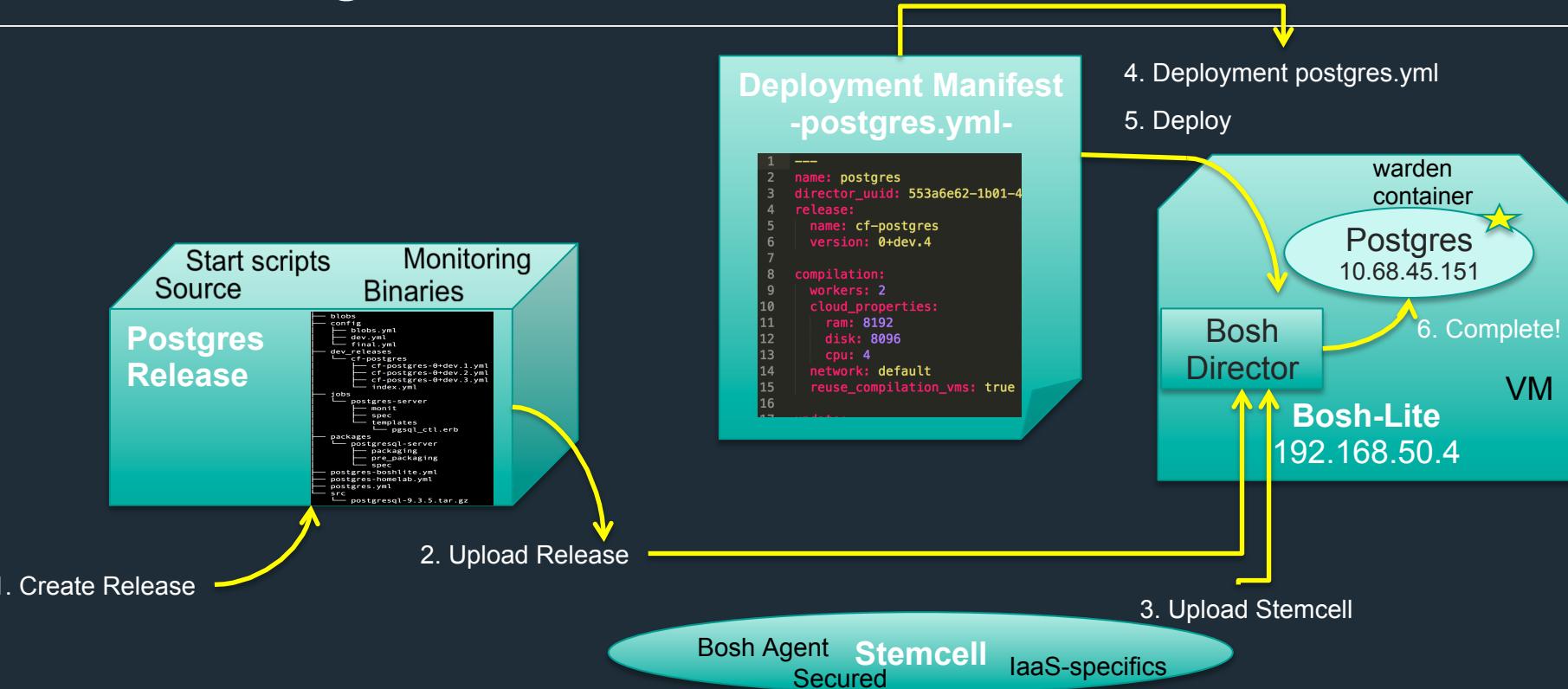
Pivotal.

# LAB 2:

## Deploying an Existing Bosh Release



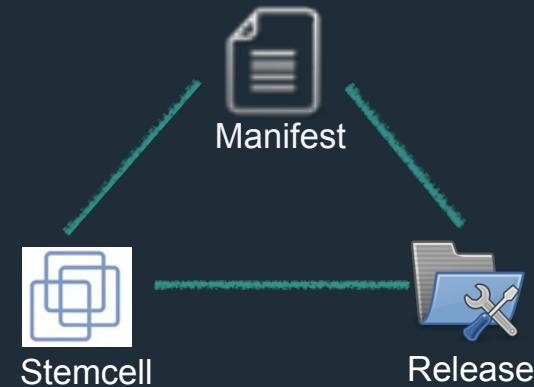
# Lab 2: Big Picture



# What is a BOSH deployment?

An encapsulation of software and configuration to be deployed to the cloud.

In BOSH terminology, an encapsulation of software **releases** and configuration declared via a **manifest file** to be deployed on VMs built from **stemcells**.



# What is a stemcell?

- A versioned Operating System image wrapped with IaaS specific packaging
- Contains a bare minimum OS skeleton with a few common utilities, a BOSH agent, and a few configuration files to securely configure the OS
- Stemcells do no contain any specific information about any software that will be installed

# Available stemcells

## LATEST STEMCELLS

### Ubuntu Trusty

AWS Xen-HVM Light 18KB	<a href="#">3173 / prev...</a>	bosh-aws-xen-hvm-ubuntu-trusty-go_agent 2016-01-07722:05:30.0002
AWS Xen 485MB	<a href="#">3173 / prev...</a>	bosh-aws-xen-ubuntu-trusty-go_agent 2016-01-07722:05:30.0002
AWS Xen Light 18KB	<a href="#">3173 / prev...</a>	bosh-aws-xen-ubuntu-trusty-go_agent 2016-01-07722:05:30.0002
Azure Hyper-V 486MB	<a href="#">3169 / prev...</a>	bosh-azure-hyperv-ubuntu-trusty-go_agent 2016-01-04719:56:49.0002
OpenStack KVM 514MB	<a href="#">3173 / prev...</a>	bosh-openstack-kvm-ubuntu-trusty-go_agent 2016-01-07722:05:30.0002
OpenStack KVM (raw) 486MB	<a href="#">3173 / prev...</a>	bosh-openstack-kvm-ubuntu-trusty-go_agent-raw 2016-01-07722:05:30.0002
vCloud ESXi 512MB	<a href="#">3173 / prev...</a>	bosh-vcloud-esxi-ubuntu-trusty-go_agent 2016-01-07722:05:30.0002
vSphere ESXi 512MB	<a href="#">3173 / prev...</a>	bosh-vsphere-esxi-ubuntu-trusty-go_agent 2016-01-07722:05:30.0002
BOSH Lite Warden 499MB	<a href="#">3147 / prev...</a>	bosh-warden-boshlite-ubuntu-trusty-go_agent 2015-12-04707:57:17.0002

### CentOS 7.x

AWS Xen 658MB	<a href="#">3169 / prev...</a>	bosh-aws-xen-centos-7-go_agent 2016-01-04719:08:01.0002
AWS Xen Light 12KB	<a href="#">3169 / prev...</a>	bosh-aws-xen-centos-7-go_agent 2016-01-04719:08:01.0002
AWS Xen-HVM Light 12KB	<a href="#">3169 / prev...</a>	bosh-aws-xen-hvm-centos-7-go_agent 2016-01-04719:08:01.0002
Azure Hyper-V 658MB	<a href="#">3169 / prev...</a>	bosh-azure-hyperv-centos-7-go_agent 2016-01-04719:58:55.0002
OpenStack KVM 699MB	<a href="#">3169 / prev...</a>	bosh-openstack-kvm-centos-7-go_agent 2016-01-04719:08:01.0002
OpenStack KVM (raw) 658MB	<a href="#">3169 / prev...</a>	bosh-openstack-kvm-centos-7-go_agent-raw 2016-01-04719:08:01.0002
vSphere ESXi 697MB	<a href="#">3169 / prev...</a>	bosh-vsphere-esxi-centos-7-go_agent 2016-01-04719:08:01.0002

<http://bosh.io/stemcells>

# What is a release?

---

- A versioned collection of source code, binary artifacts, configuration templates, and anything else required to build and deploy software in a reproducible way
- Ultimately, a release is the layer placed on top of a stemcell
- Releases should not be tied to a particular IaaS and stemcell and should not depend on internet access

# Available releases

## COMMUNITY RELEASES

[cf-platform-eng/docker-boshrelease](#)  
[cf-platform-eng/hadoop-boshrelease](#)  
[cf-platform-eng/mesos-boshrelease](#)  
[cf-platform-eng/networking-boshrelease](#)  
[cf-platform-eng/shipyard-boshrelease](#)  
[cloudfoundry-community/admin-ui-boshrelease](#)  
[cloudfoundry-community/bosh-cloudstack-cpi-release](#)  
[cloudfoundry-community/broker-registrar-boshrelease](#)  
[cloudfoundry-community/cf-haproxy-boshrelease](#)  
[cloudfoundry-community/cf-services-contrib-release](#)  
[cloudfoundry-community/cf-subway-boshrelease](#)  
[cloudfoundry-community/consul-boshrelease](#)  
[cloudfoundry-community/crate-boshrelease](#)  
[cloudfoundry-community/docker-registry-boshrelease](#)  
[cloudfoundry-community/jumpbox-boshrelease](#)  
[cloudfoundry-community/logstash-docker-boshrelease](#)  
[cloudfoundry-community/memcache-release](#)  
[cloudfoundry-community/monitor-server-boshrelease](#)

<http://bosh.io/releases>

# What is a deployment manifest?

- A YAML file that identifies one or more releases, one or more stemcells and specifies how to orchestrate and configure them for a given deployment

# Anatomy of a deployment manifest

- Deployment identification
- Releases
- Networks
- Resource Pools
- Disk Pools
- Jobs
- Compilation
- Update
- Properties

# Deployment Identification

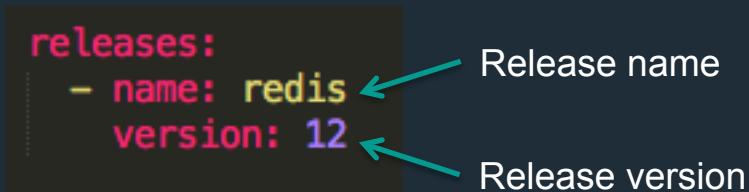
Identifies the name of the deployment and the BOSH Director to be used

```
name: my-redis-deployment          Deployment name  
director_uuid: cf8dc1fc-9c42-4ffc-96f1-fbad983a6ce6    BOSH Director UUID
```

# Releases

Identifies a collection of releases (name and version) we want to deploy

```
releases:  
  - name: redis  
    version: 12
```



Release name  
Release version

# Networks

A BOSH network is an IaaS-agnostic representation of the networking layer. There are three types of networks that BOSH supports:

- **Manual**: BOSH decides how to assign IPs
- **Dynamic**: BOSH defers IP selection to the IaaS
- **VIP**: Virtual IP address

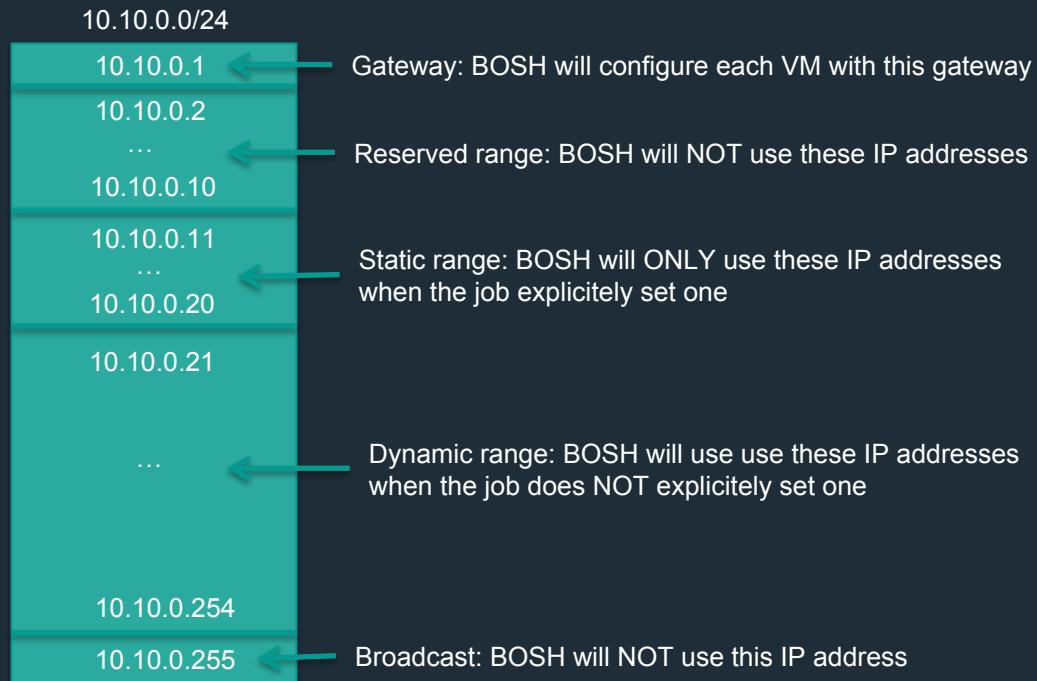
# Manual networks

Manual networking allows you to specify one or more subnets and let BOSH choose available IPs from one of the subnet ranges

```
networks:
- name: my-manual-network ← Network name
  type: manual ← Network type
  subnets:
    - range: 10.10.0.0/24 ← Network IP range
      gateway: 10.10.0.1 ← Network gateway IP address
      dns:
        - 10.10.0.2 ← DNS IP addresses for this network
    reserved:
      - 10.10.0.2-10.10.0.10 ← IPs that will not be used
    static:
      - 10.10.1.11-10.10.1.20 ← IPs that can only be used for static IP reservations
  cloud_properties:
    subnet: subnet-9be6c3f7 ← Specific IaaS Properties
```

# Manual networks

```
networks:  
- name: my-manual-network  
  type: manual  
  subnets:  
    - range: 10.10.0.0/24  
      gateway: 10.10.0.1  
      dns:  
        - 10.10.0.2  
      reserved:  
        - 10.10.0.2-10.10.0.10  
      static:  
        - 10.10.1.11-10.10.1.20  
    cloud_properties:  
      subnet: subnet-9be6c3f7
```



# Dynamic networks

Dynamic networking defers IP selection to the IaaS

```
networks:  
- name: my-dynamic-network ← Network name  
  type: dynamic ← Network type  
  dns:  
    - 10.10.0.2 ← DNS IP addresses for this network  
  cloud_properties:  
    subnet: subnet-9be6c3f7 ← Specific IaaS Properties
```

# VIP networks

VIP networking enables the association of a virtual IP address that is not backed by any particular NIC

```
networks:
```

```
- name: my-vip-network
```

```
  type: vip
```

Network name

Network type

# IP reservation types

	Manual	Dynamic	VIP
Static IP assignment	Supported	Not supported	Supported
Automatic IP assignment	Supported	Supported	Not Supported

- **Static:** IP is explicitly requested by the user in the deployment manifest
- **Automatic:** IP is selected automatically based on the network type

# Multi-networks

IaaS	Manual	Dynamic	VIP
AWS	Single per job instance	Single per job instance	Single per job instance
OpenStack	Multiple per job instance	Single per job instance	Single per job instance
vSphere	Multiple per job instance	Not supported	Not supported
vCloud	Multiple per job instance	Not supported	Not Supported

# Resource Pools

Identifies a collection of VM specifications

```
resource_pools:  
- name: redis-servers  
  network: default  
  stemcell:  
    name: bosh-aws-xen-ubuntu-trusty-go_agent  
    version: 23173  
  cloud_properties:  
    instance_type: m1.medium  
    availability_zone: us-east-1a
```

Resource Pool name  
Network to use  
Stemcell (name and version)  
Specific IaaS Properties

# Disk Pools

Identifies a collection of disk specifications

```
disk_pools:  
  - name: ssd-disk  
    disk_size: 2  
    cloud_properties:  
      type: ssd
```

Disk Pool name

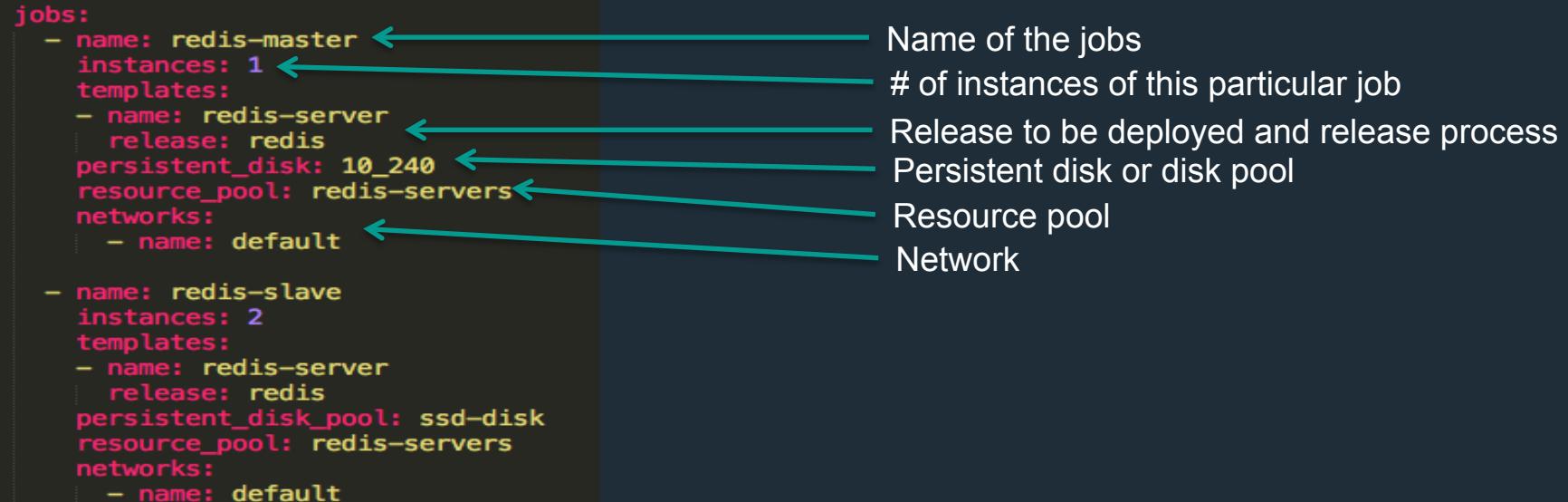
Disk size (Mb)

Specific IaaS Properties

# Jobs

A job is a composition of configuration and installed software on a particular VM

```
jobs:  
  - name: redis-master  
    instances: 1  
    templates:  
      - name: redis-server  
        release: redis  
    persistent_disk: 10_240  
    resource_pool: redis-servers  
    networks:  
      - name: default  
  
  - name: redis-slave  
    instances: 2  
    templates:  
      - name: redis-server  
        release: redis  
    persistent_disk_pool: ssd-disk  
    resource_pool: redis-servers  
    networks:  
      - name: default
```



Name of the job  
# of instances of this particular job  
Release to be deployed and release process  
Persistent disk or disk pool  
Resource pool  
Network

# Job's networks

Depending on the job's network, additional properties might be configured

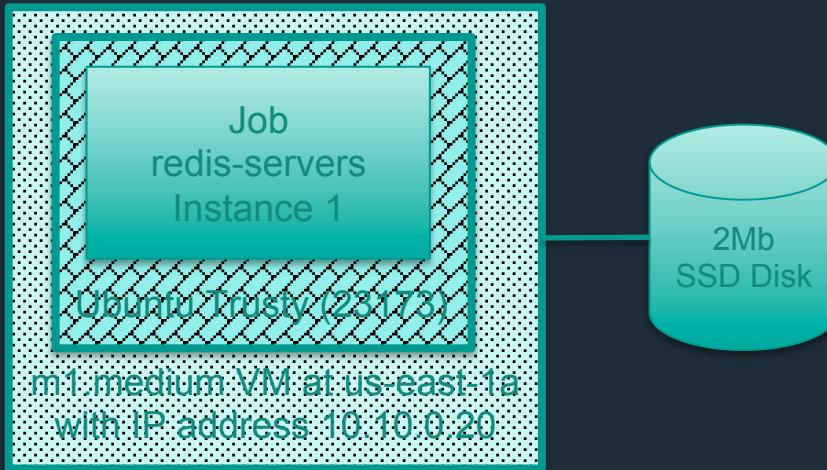
```
jobs:  
  - name: redis-master  
    instances: 1  
    templates:  
      - name: redis-server  
        release: redis  
        persistent_disk: 10_240  
        resource_pool: redis-servers  
        networks:  
          - name: static  
            static_ips:  
              - 10.0.10.20  
          - name: vip  
            static_ips:  
              - 72.52.4.119
```

Manual networks allows to specify the desired private IP address(es)

VIP networks require to specify the public IP address(es)

# Jobs

```
resource_pools:  
- name: redis-servers  
  network: default  
  stemcell:  
    name: bosch-aws-xen-ubuntu-trusty-go_agent  
    version: 23173  
  cloud_properties:  
    instance_type: m1.medium  
    availability_zone: us-east-1a  
  
disk_pools:  
- name: ssd-disk  
  disk_size: 2  
  cloud_properties:  
    type: ssd  
  
jobs:  
- name: redis-master  
  instances: 1  
  templates:  
- name: redis-server  
  release: redis  
  persistent_disk_pool: ssd-disk  
  resource_pool: redis-servers  
  networks:  
- name: default  
  static_ips:  
- 10.10.0.20
```



# Job types

- **Service** (default): runs indefinitely and restarts if it fails
- **Errand**: starts with a manual trigger and does not restart if it fails

```
- name: acceptance-tests
  templates:
    - name: acceptance-tests
      release: redis
  instances: 1
  lifecycle: errand ← Job type
  resource_pool: errands
  networks:
    - name: default
```

# Compilation

Compilation VMs are ephemeral instances where the source code from a release package is converted to a binary compatible with the stemcell

```
compilation:  
  workers: 2 ← # of compilation VMs  
  reuse_compilation_vms: true ← If VMs are re-used when compiling other packages  
  network: default ← Network  
  cloud_properties:  
    instance_type: c1.medium ← Specific IaaS Properties  
    availability_zone: us-east-1a
```

# Update

Describes how BOSH updates job instances during the deployment

```
update:  
  canaries: 2 # of canary instances  
  max_in_flight: 2 Max # of non-canary instances to update in parallel  
  canary_watch_time: 1000-30000 Wait time (ms) to check whether canary instances are healthy  
  update_watch_time: 1000-30000 Wait time (ms) to check whether instances are healthy  
  serial: false How jobs are updated (serial or parallel)
```

# Canary Instances

Canary instances are instances updated before other instances. Any update error in a canary instance causes the deployment to stop. Since only canaries instances are affected before an update stops, problematic packages or jobs are prevented from taking over all job instances.



Canaries are Great! - <https://blog.pivotal.io/pivotal-cloud-foundry/products/canaries-are-great>

# Update

Global update properties can be overridden for a particular job

```
jobs:
  - name: redis-master
    instances: 1
    templates:
      - name: redis-server
        release: redis
        persistent_disk: 10_240
        resource_pool: redis-servers
    networks:
      - name: default
    update:
      canaries: 1
      max_in_flight: 1
      canary_watch_time: 10000-60000
      update_watch_time: 10000-60000
      serial: true
```

# Properties

Describes the global properties in order to configure jobs for a specific environment

```
properties:  
  redis:  
    max_connections: 10
```

# Properties precedence

- BOSH applies the properties in the release template spec file to the job.
- If an identically named property exists in the Properties block of the deployment manifest, the value of this property overrides the previous value.
- If an identically named property exists in the Properties sub-block of the Jobs block of the deployment manifest, the value of this property overrides all previous values.

# Properties order

## Release template spec

```
---
```

```
name: redis-server
packages:
  - redis-server
properties:
  redis.max_connections: 3
    description: Maximum number of connections
  default: 1
```

1 Job properties

2 Global properties

3 Release template default

## Deployment manifest

```
jobs:
  - name: redis-master
    instances: 1
    templates:
      - name: redis-server
        release: redis
        persistent_disk: 10_240
        resource_pool: redis-servers
        networks:
          - name: default
            properties:
              redis:
                max_connections: 5
  properties:
    redis:
      max_connections: 10
```

Pivotal™

# Changing the deployment manifest

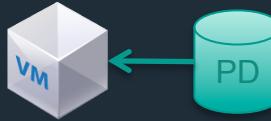
- An update of a release or stemcell version will trigger the compilation of (newer) packages and the update of the affected jobs
- An update of a network properties might trigger a VM recreation (ie: modifying the network security group)
- An update of a resource pool's properties or a job's resource pool will trigger a VM recreation
- An update of a job's network will trigger a VM recreation
- An update of a job's property will trigger the update of the job

# Persistent disk considerations

- Everything inside a VM is considered ephemeral except any attached persistent disk
- A VM recreation will keep all data from the persistent disk intact
- An update of the size and/or type of a persistent disk will trigger a migration of the data to a new persistent disk

# Persistent disks

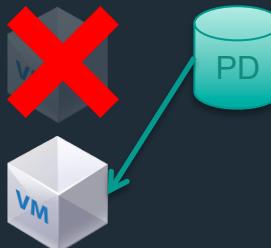
1. A PD is attached to a VM



2. If the VM is deleted, the PD remains

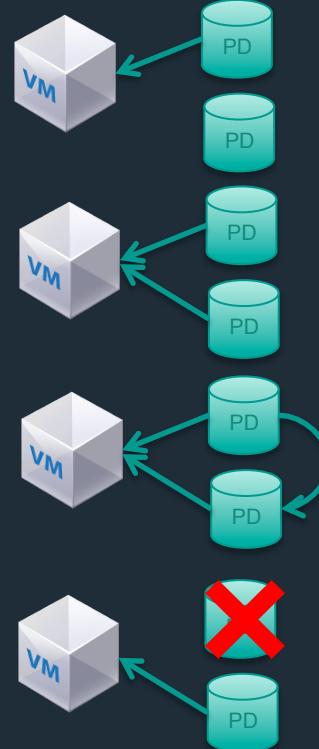


3. The old PD is reattached to the new VM



# Persistent disk migration

1. A new PD is created with the new properties
2. The new PD is attached to the VM
3. The BOSH agent migrates the data from the old PD to the new PD
4. The old PD is detached and deleted



# BOSH documentation

---

<http://bosh.io/docs>

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
+---+-----+-----+-----+-----+
```

# BOSH tasks

bosh tasks recent --no-filter

#	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"} `redis-master/0` asked for a static IP 10.244.0.1 but it's already reserved/in...
7	error	2016-01-10 20:02:55 UTC	admin	create deployment	Required property `networks` was not specified in object...
6	error	2016-01-10 20:02:39 UTC	admin	create deployment	Property `name` (value nil) did not match the required type `String` Required property `update` was not specified in object ({"name"=>"redis",...}
5	error	2016-01-10 20:01:59 UTC	admin	create deployment	Enqueued snapshot tasks □ Created release `redis/9.1'
4	error	2016-01-10 20:01:29 UTC	admin	create deployment	/stemcells/bosh-warden-boshlite-ubuntu-trusty-go_agent/3147
3	done	2016-01-10 07:00:02 UTC	scheduler	scheduled SnapshotDeployments	
2	done	2016-01-10 03:26:34 UTC	admin	create release	
1	done	2016-01-10 03:08:17 UTC	admin	create stemcell	

# 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', "username" 'root'
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: {}
  
```

# 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

# 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
```

# 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!!!)

# 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”

# 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
```

# 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<1 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<1 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<1 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<1 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<s1 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
```

# VM health-check

---

- The Health Monitor continuously 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.

# Health Monitor plugins

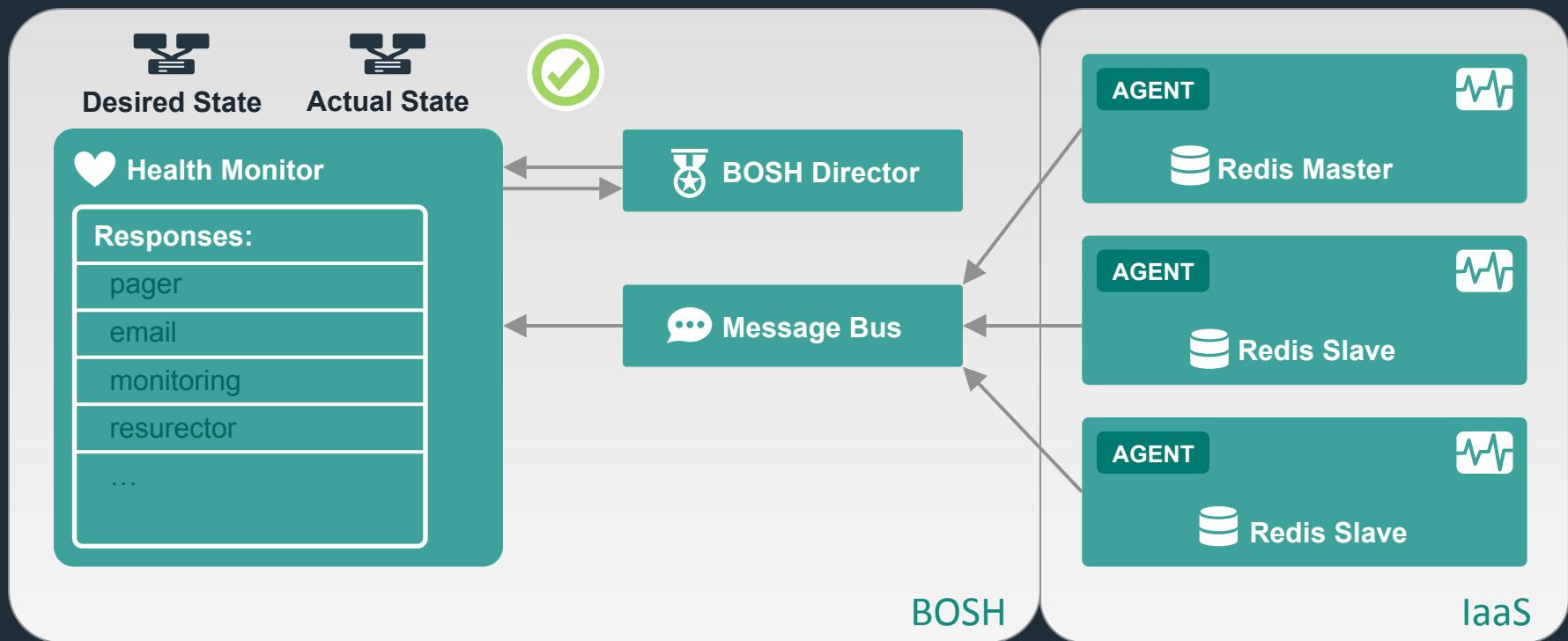
- Event Logger: Logs events to a file
- Resurrector: Recreates VMs that have stopped heartbeating
- Emailer: Sends configurable e-mails on events receipt
- OpenTSDB: Sends events to OpenTSDB
- Graphite: Sends events to Graphite
- PagerDuty: Sends events to PagerDuty.com using their API
- DataDog: Sends 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>

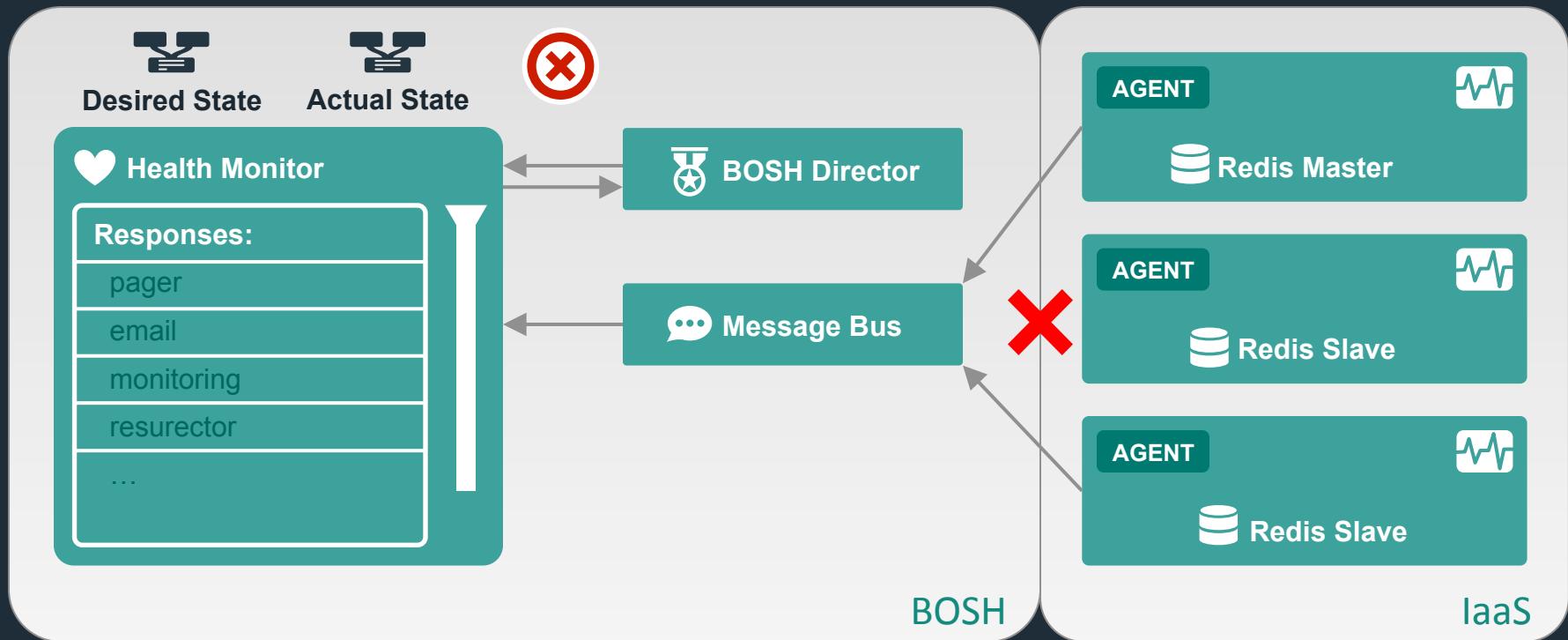
# Resurrector Health Monitor plugin

- It's responsible for automatically recreating VMs that become inaccessible.
- It continuously 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.

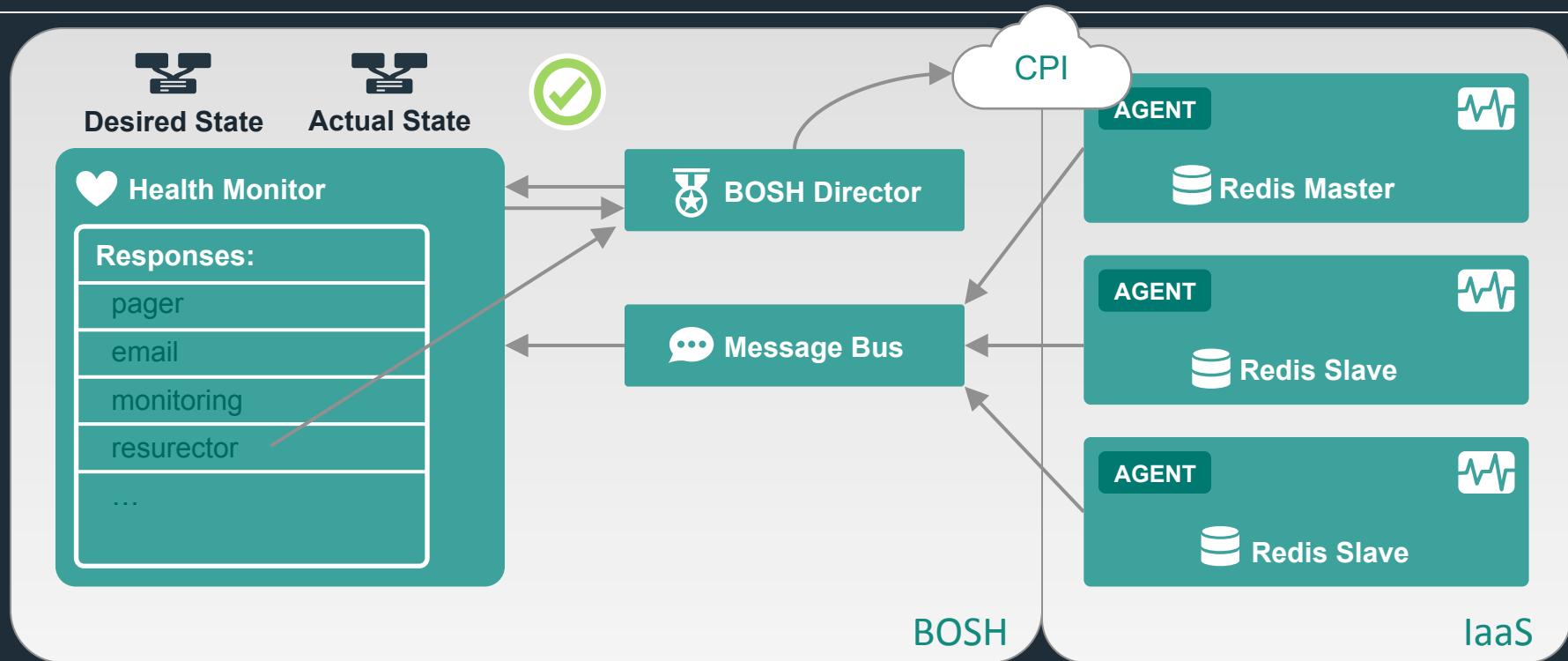
# Automatic repair with Resurrector



# Automatic repair with Resurrector



# Automatic repair with Resurrector



# Manually Disabling the Resurrector

“bosch 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-05eb4938f85	paused
redis-slave/1	running	default	10.244.0.4	e4d96bff-92f8-4e64-6588-d7b272dadaca	5c75a317-8514-42e2-ace2-026349c96370	active

VMs total: 3

# 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

# 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
```

# 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

# 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

# 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

# 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.