BOSH: The Simple Solution for Complex Systems

Kira Combs
Pivotal Cloud Foundry
@kicombs
The Old World

My Product

[Diagram with various icons and arrows connecting to "My Product"]
The Old World

Component 1

Component 2

Component X
There’s a Better Way!
BOSH

The outer shell for cloud software
What is BOSH?
Key Principles

- Identifiability
- Reproducibility
- Consistency
- Agility
How Do I Get There?
Get BOSH(ed) (The Director)

- Get an Environment (AWS, Azure, local...)
- Install the bosh CLI:
  - https://bosh.io/docs/bosh-cli.html
- Create a manifest

```
$ gem install bosh_cli --no-ri --no-rdoc
```

```
$ bosh micro deploy light-bosh-stemcell-3012-aws-xen-ubuntu-trusty-go_agent.tgz
```
Stemcell

OS Update
BOSH Release

```
bosh init release mysystem
```

```
  $ tree
  .
  ├── blobs
  │   └── blobs.yml
  ├── config
  │   └── blobs.yml
  └── packages
      └── src

5 directories, 1 file
```

Populate Me!

Normal product code
BOSH Release

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

((https://bosh.io/releases))
The BOSH Manifest

- Components separated into jobs
- Define:
  - Config
  - Properties
  - Resources
  - IP Ranges
  - Subnets
The BOSH Manifest

- name: my-system-deployment
  director_uuid: 1234abcd-5678-efab-9012-3456cdef7890

releases:
  - {name: mysystem, version: 12}

resource_pools:
  - name: system-servers
    network: default
    stemcell:
      name: bosh-aws-xen-ubuntu-trusty-go_agent
      version: 2708
    cloud_properties:
      instance_type: m1.small
      availability_zone: us-east-1c

https://bosh.io/releases

Name of Deployment
Bosh Director UUID
The Release and desired version
Resource Pool
Stemcell Version
The BOSH Manifest

Networks:
- name: default
type: manual
subnets:
  - range: 10.10.0.0/24
gateway: 10.10.0.1
static:
  - 10.10.0.16 - 10.10.0.18
reserved:
  - 10.10.0.2 - 10.10.0.15
dns: [10.10.0.6]
cloud_properties:
  subnet: subnet-d597b993

Compilation:
workers: 2
network: default
reuse_compilation_vms: true
cloud_properties:
  instance_type: c1.medium
  availability_zone: us-east-1c

update:
canaries: 1
max_in_flight: 3
canary_watch_time: 15000-30000
update_watch_time: 15000-30000
The BOSH Manifest

**Components of the System**

```yaml
jobs:
- name: my-system-component-one
  instances: 1
  templates:
    - {name: mysystem-server, release: mysystem}
    persistent_disk: 10_240
    resource_pool: mysystem-servers
  networks:
    - name: default

- name: my-system-component-two
  instances: 2
  templates:
    - {name: mysystem-server, release: mysystem}
    persistent_disk: 10_240
    resource_pool: mysystem-servers
  networks:
    - name: default

properties:
  mysystem:
    max_connections: 10
```

**Custom Global Properties**
The Learning Curve

BEST THING EVER!
THIS IS COOL...
MEH
HATE
HATE
HATE

Rate of Learning

Time
Cloud Foundry Loves BOSH
Deploying CF using BOSH

- Generate a deployment manifest
- Upload stemcell
- BOSH create, upload, and deploy

```bash
bosh upload stemcell
bosh create release
bosh upload release
bosh deploy
```
<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>acceptance-tests</td>
<td>71b87f474b5a4e2d7c6b20d52f873f9e7388</td>
<td></td>
</tr>
<tr>
<td>binary-buildpack</td>
<td>71b87f474b5a4e2d7c6b20d52f873f9e7388</td>
<td></td>
</tr>
<tr>
<td>bixtoner</td>
<td>62a0b2c0443b528f4043f9e7388</td>
<td></td>
</tr>
<tr>
<td>cz_uploader</td>
<td>62a0b2c0443b528f4043f9e7388</td>
<td></td>
</tr>
<tr>
<td>cloud_controller_plugin</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>cloud_controller_ng</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>cloud_controller_worker</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>customizer</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>cz_client</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>cz_webserver</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>daemon</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>etcd</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>etcd_controller_worker</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>etcd_server</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>gcp-host</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>google_buildpack</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>proxmox</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>proxmox_buildpack</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>proxmox_offline_buildpack</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>nets</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>nets_stream_forwarder</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>nts_mediator</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>nodejs_buildpack</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>nginx</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>nodejs_buildpack</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>postgres</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>python_buildpack</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>rails_buildpack</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>rails_test</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>sample_tests</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>shapei</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>staticfile_buildpack</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>types</td>
<td>53c5bed0daa3f48f4354e4f39e7388</td>
<td></td>
</tr>
<tr>
<td>acceptance-tests</td>
<td>71b87f474b5a4e2d7c6b20d52f873f9e7388</td>
<td></td>
</tr>
</tbody>
</table>
Not Just Cloud Foundry

MySQL

RabbitMQ

riak

Docker

Fan

redis
rabbitmq-smoke-tests-boshrelease
Smoke Tests for RabbitMQ CloudFoundry/BOSH deployments
Updated 6 hours ago

cf-rabbitmq-example-stomp-app
Ruby ★ 0 ▼ 0
Updated 6 hours ago

cf-rabbitmq-smoke-tests
Smoke Tests for RabbitMQ CloudFoundry/BOSH deployments
Updated 6 hours ago

docker-registry-boshrelease
Shell ★ 4 ▼ 3
Run your own private Docker Registry in standalone mode (without requiring the public index) on AWS, OpenStack or vSphere with BOSH
Updated 8 hours ago

bosh-softlayer-tools
Go ★ 0 ▼ 1

(https://github.com/cloudfoundry-community)
GitHub - murraju/kafka-boshrelease: A BOSH release for Apache Kafka. Contribute to kafka-boshrelease development by creating an account on GitHub.

GitHub - hybris/kafka-boshrelease
https://github.com/hybris/kafka-boshrelease

BOSH release for kafka broker - GitHub
https://github.com/viniciusccarvalho/kafka-boshrelease
BOSH release for kafka broker. Contribute to kafka-boshrelease development by creating an account on GitHub.
Useful Links and Tutorials


BOSH Documentation: [http://bosh.io/docs](http://bosh.io/docs)

BOSH CF Community: [https://github.com/cloudfoundry-community](https://github.com/cloudfoundry-community)

Deploy CF with BOSH: [https://docs.cloudfoundry.org/deploying/index.html](https://docs.cloudfoundry.org/deploying/index.html)

BOSH Lite: [https://github.com/cloudfoundry/bosh-lite](https://github.com/cloudfoundry/bosh-lite)
Questions?