

Deployments got you up all night?

Learn to sleep easy with *AWS* codedeploy



Adam Dockter
VP of Engineering
ServiceTarget

Who are we?

ServiceTarget

Small Company 4 Developers

AWS Infrastructure

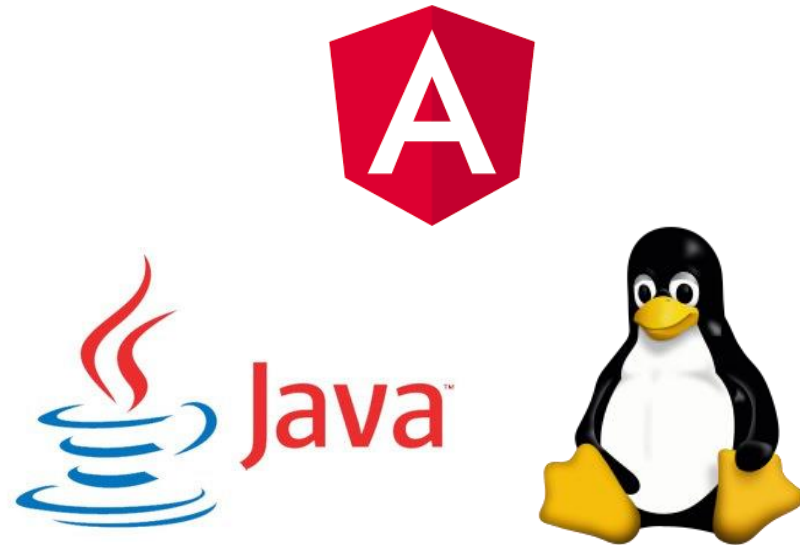
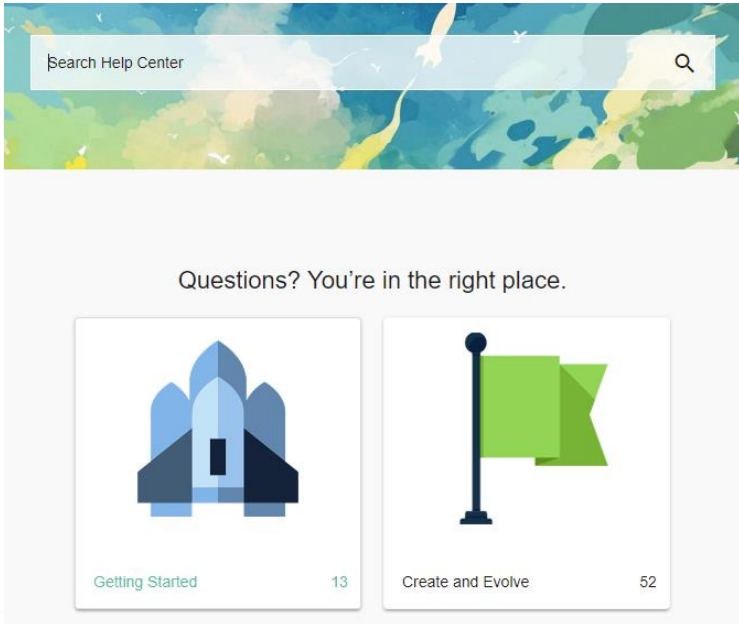
NO QA!!



**CAN'T SOMEONE ELSE
JUST DO IT?**

About our product

Self service web application deployed on AWS



Goal of this Session

Deploying To Production

Before



After



Assumptions

Basic understanding of AWS concepts



S3



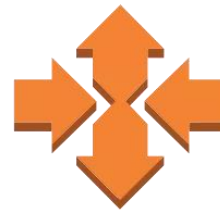
EC2



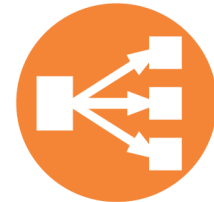
IAM



AWS CLI



Autoscaling



ELB

CodeDeploy



S3



AWS CodeDeploy



EC2

Complexity of deployments

1. Deregister from ELB
2. Stop service
3. Update data storage
4. Run utility scripts
5. Secure copy files
6. Start services
7. Register to ELB
8. Run validation

×

Environments

1. Dev
2. QA
3. Production

EC2 Instances

- at least 3

=

72

Complexity of deployments

1. Deregister from ELB
2. Stop service
3. Update data storage
4. Run utility scripts
5. Secure copy files
6. Start services
7. Register to ELB
8. Run validation



Environments

1. Dev
2. QA
3. Production

EC2 Instances
- at least 3



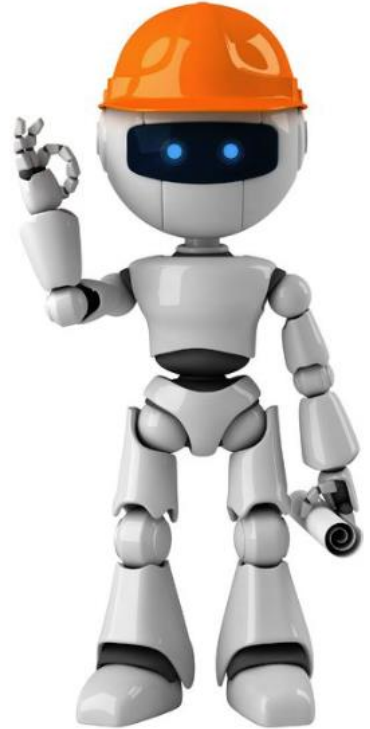
Repetitive Work
Low Value Work
Time consuming
Human Error
Bad Hours

Complexity of deployments

1. Deregister from ELB
2. Stop service
3. Update data storage
4. Run utility scripts
5. Secure copy files
6. Start services
7. Register to ELB
8. Run validation



- Environments
1. Dev
 2. QA
 3. Production
- EC2 Instances
- at least 3



Complexity of deployments

1. Deregister from ELB
2. Stop service
3. Update data storage
4. Run utility scripts
5. Secure copy files
6. Start services
7. Register to ELB
8. Run validation



Environments

1. Dev
2. QA
3. Production

EC2 Instances
- at least 3



AWS CodeDeploy

CodeDeploy



S3




AWS CodeDeploy



EC2

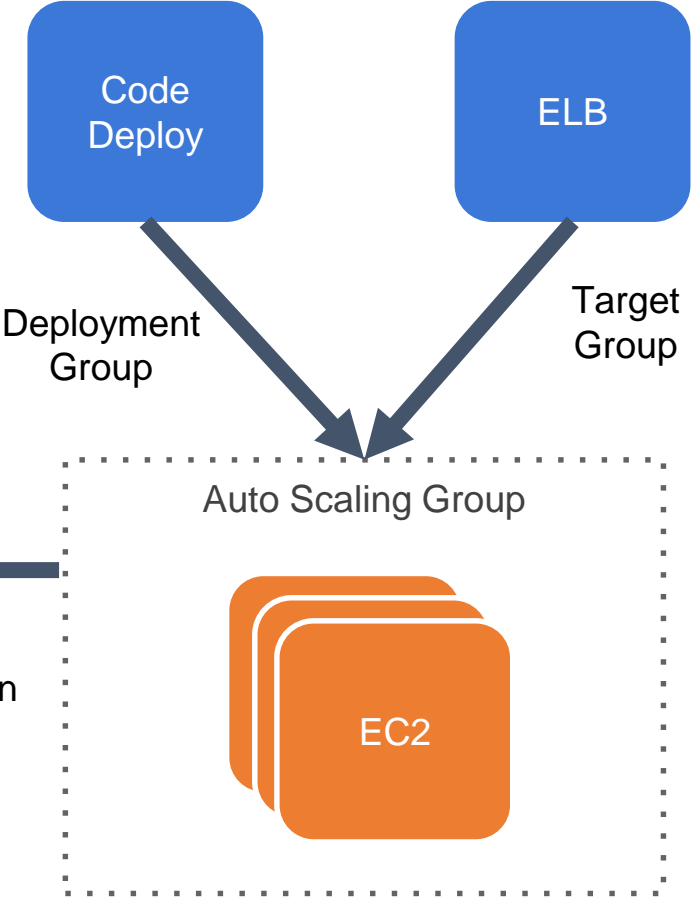
EC2

- Installed Node JS
- Installed AWS CodeDeploy Agent 



Launch Configuration

An arrow points from the Launch Configuration text to the AMI icon.



Deployment Group

one at a time



Notes

- Application changes are backward compatible
- Able to keep most services running to continue to service load needs

Deployment Group



Deployment Group

half at a time



Notes

- Backward compatibility not required
- Lose half of your service bandwidth, require low traffic time

Deployment Group



Deployment Group

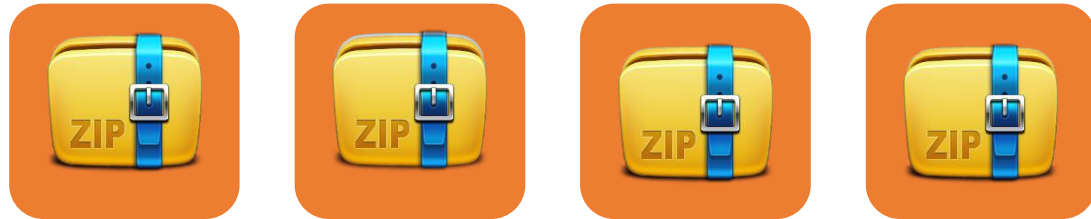
all at once



Notes

- Backward compatibility not required
- Downtime required
- Fastest

Deployment Group



Deployment Group

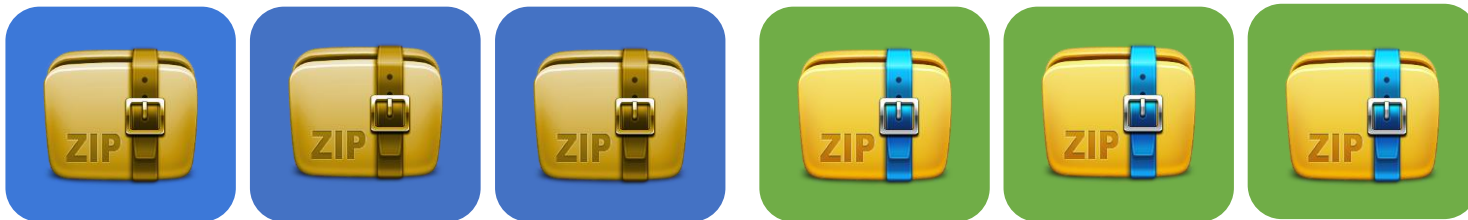
blue / green



Notes

- Backward compatibility not required, for all scenarios
- Requires autoscaling and launch configurations

Deployment Group



CodeDeploy



S3

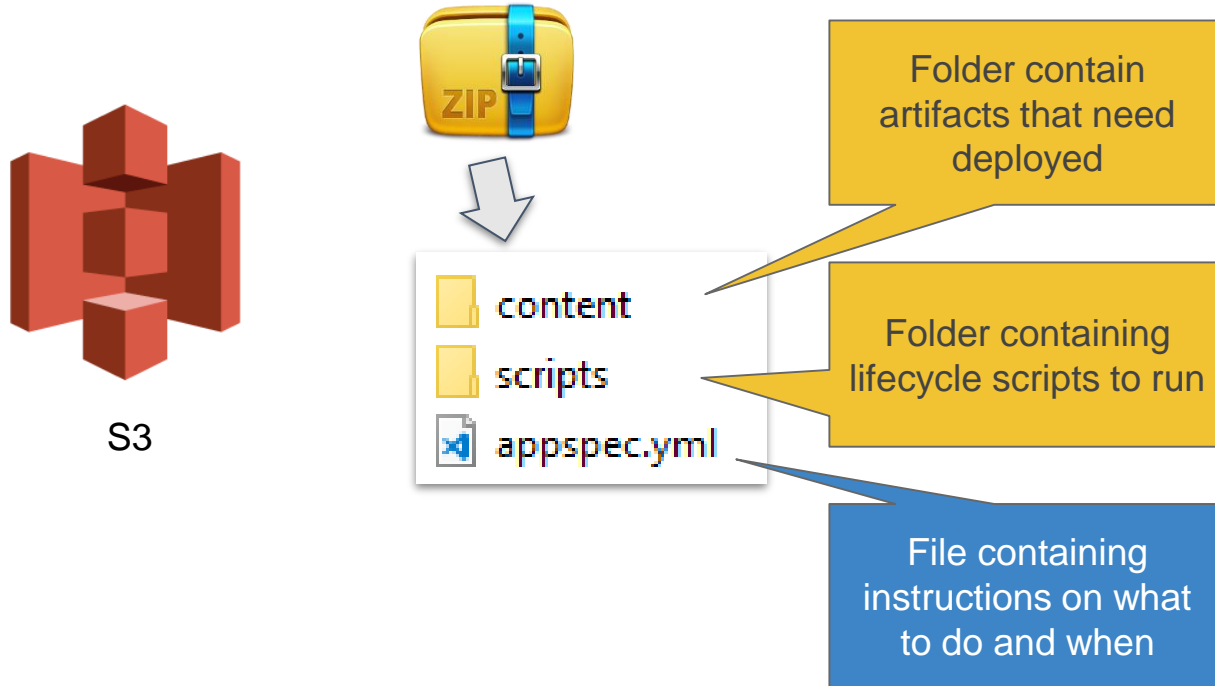


AWS CodeDeploy



EC2

Deployment Package




```
version: 1.2
```

```
os: linux
```

```
files:
```

- source: content/ServiceManager-1.2.tgz
destination: /var/app-server
- source: content/newrelic
destination: /opt/newrelic

```
hooks:
```

```
ApplicationStop:
```

- location: scripts/deregister
runas: awsDeploy
- location: scripts/stopLogstash.sh
runas: awsDeploy
- location: scripts/stopApiServer.sh
runas: awsDeploy

```
AfterInstall:
```

- location: scripts/unpackApiSnapshot.sh
runas: awsDeploy
- location: scripts/updateApiConfig.sh
runas: awsDeploy
- location: scripts/startLogstash.sh
runas: awsDeploy
- location: scripts/startApiServer.sh
runas: awsDeploy
- location: scripts/registerElb.sh
runas: awsDeploy

Deployment version
Operating System

```
version: 1.2
```

```
os: linux
```

```
files:
```

- source: content/ServiceTarget-1.2.tgz
destination: /var/app-server
- source: content/newrelic
destination: /opt/newrelic

```
hooks:
```

```
ApplicationStop:
```

- location: scripts/deregisterElb.sh
runas: awsDeploy
- location: scripts/stopLogstash.sh
runas: awsDeploy
- location: scripts/stopApiServer.sh
runas: awsDeploy

```
AfterInstall:
```

- location: scripts/unpackApiSnapshot.sh
runas: awsDeploy
- location: scripts/updateApiConfig.sh
runas: awsDeploy
- location: scripts/startLogstash.sh
runas: awsDeploy

```
ApplicationStart:
```

- location: scripts/startApiServer.sh
runas: awsDeploy
- location: scripts/registerElb.sh
runas: awsDeploy

File locations in your archive file and where they should go on the EC2 instance

```
version: 1.2
```

```
os: linux
```

```
files:
```

```
- source
```

```
destination
```

```
- source
```

```
destination
```

```
hooks:
```

```
ApplicationStop:
```

- location: scripts/deregisterElb.sh
runas: awsDeploy
- location: scripts/stopLogstash.sh
runas: awsDeploy
- location: scripts/stopApiServer.sh
runas: awsDeploy

Life cycle hooks:

1. ApplicationStop
2. BeforeInstall
3. AfterInstall
4. ApplicationStart
5. ValidateService

Scripts:

1. location
2. timeout
3. runas

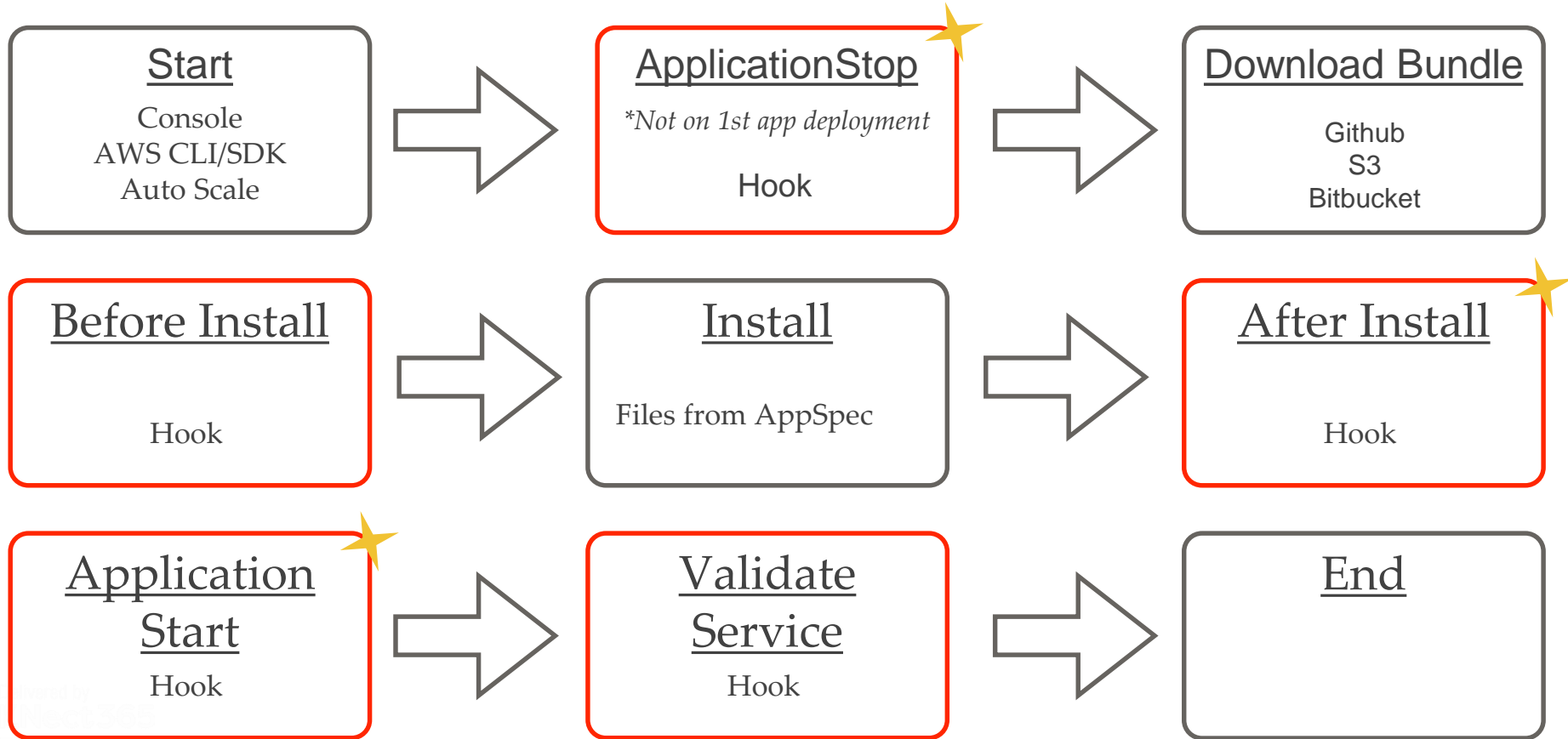
```
AfterInstall:
```

- location: scripts/unpackApiSnapshot.sh
runas: awsDeploy
- location: scripts/updateApiConfig.sh
runas: awsDeploy
- location: scripts/startLogstash.sh
runas: awsDeploy

```
ApplicationStart:
```

- location: scripts/startApiServer.sh
runas: awsDeploy
- location: scripts/registerElb.sh
runas: awsDeploy

Life cycle and hooks



DEMO TIME!



AWS CodeDeploy

Questions?

how do i go
how do i go back to the old facebook
how do i google something
how do i go offline on facebook
how do i go back to the old gmail

