QCon

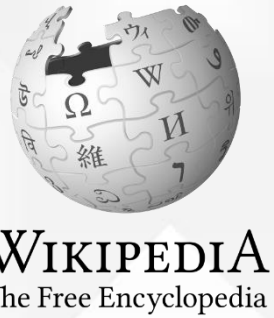
基于AWS的DevOps实践指南

代 闻

AWS解决方案架构师

wendai@amazon.com

什么是DevOps?



“

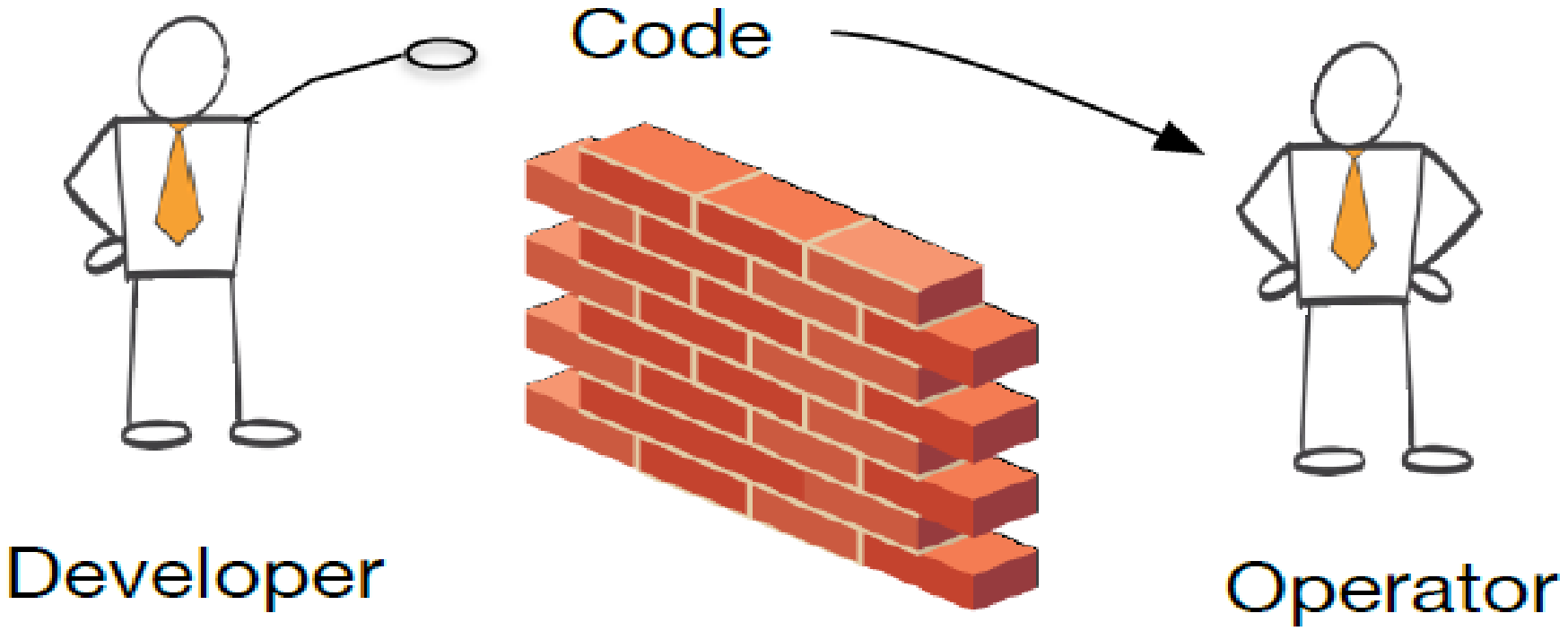
DevOps is a software development method that stresses communication, collaboration, integration, automation, and measurement of cooperation between software developers and other IT professionals …. to improve operations performance.

”

Source: http://en.wikipedia.org/wiki/DevOps

为什么需要DevOps?

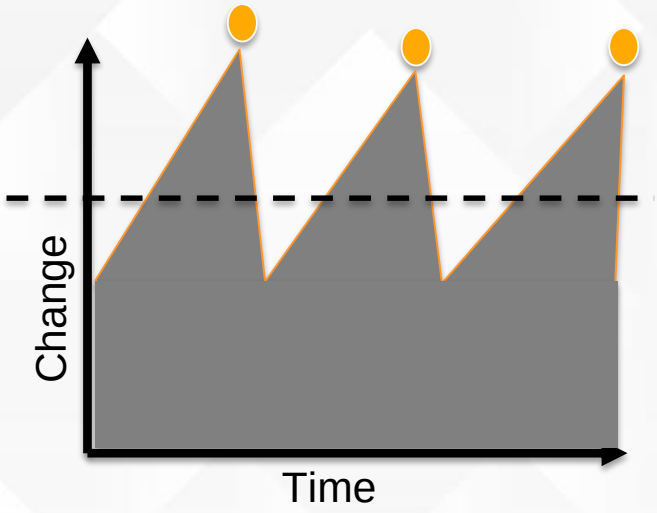
因为我们不希望事情是这样的...



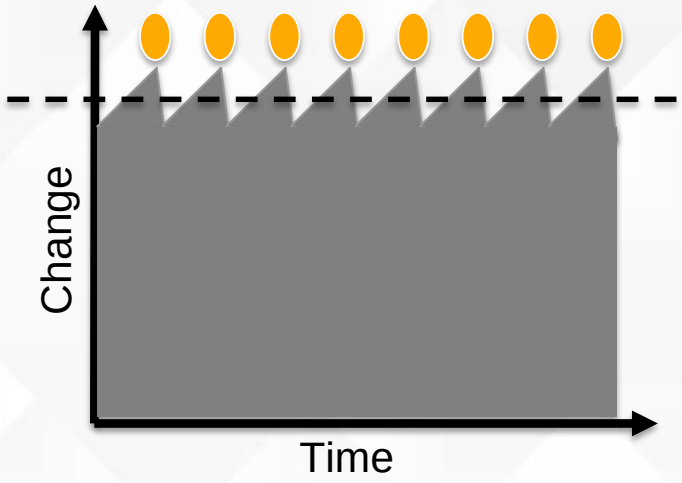
为什么需要DevOps?

我们希望...

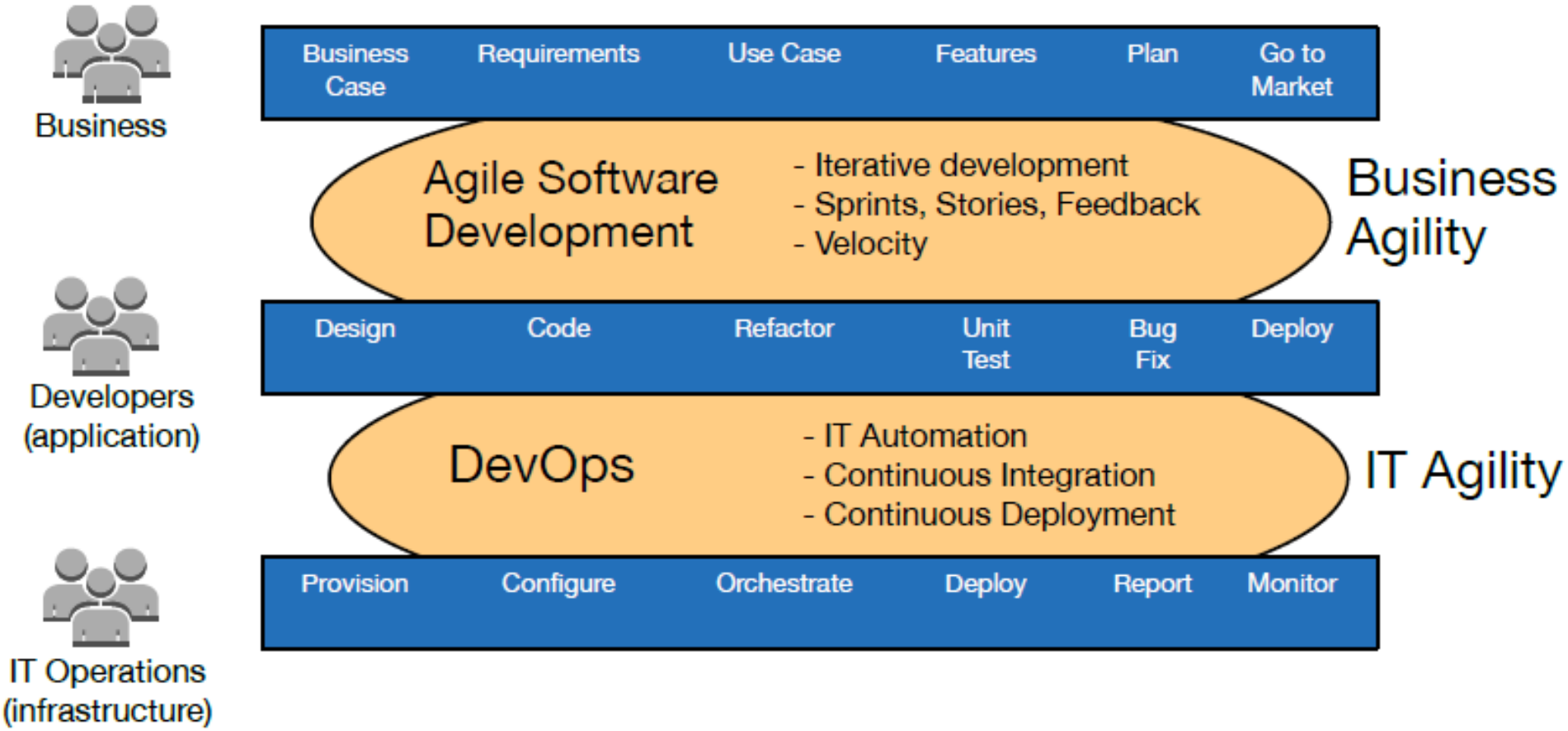
瀑布式开发，版本发布少



快速迭代，敏捷开发

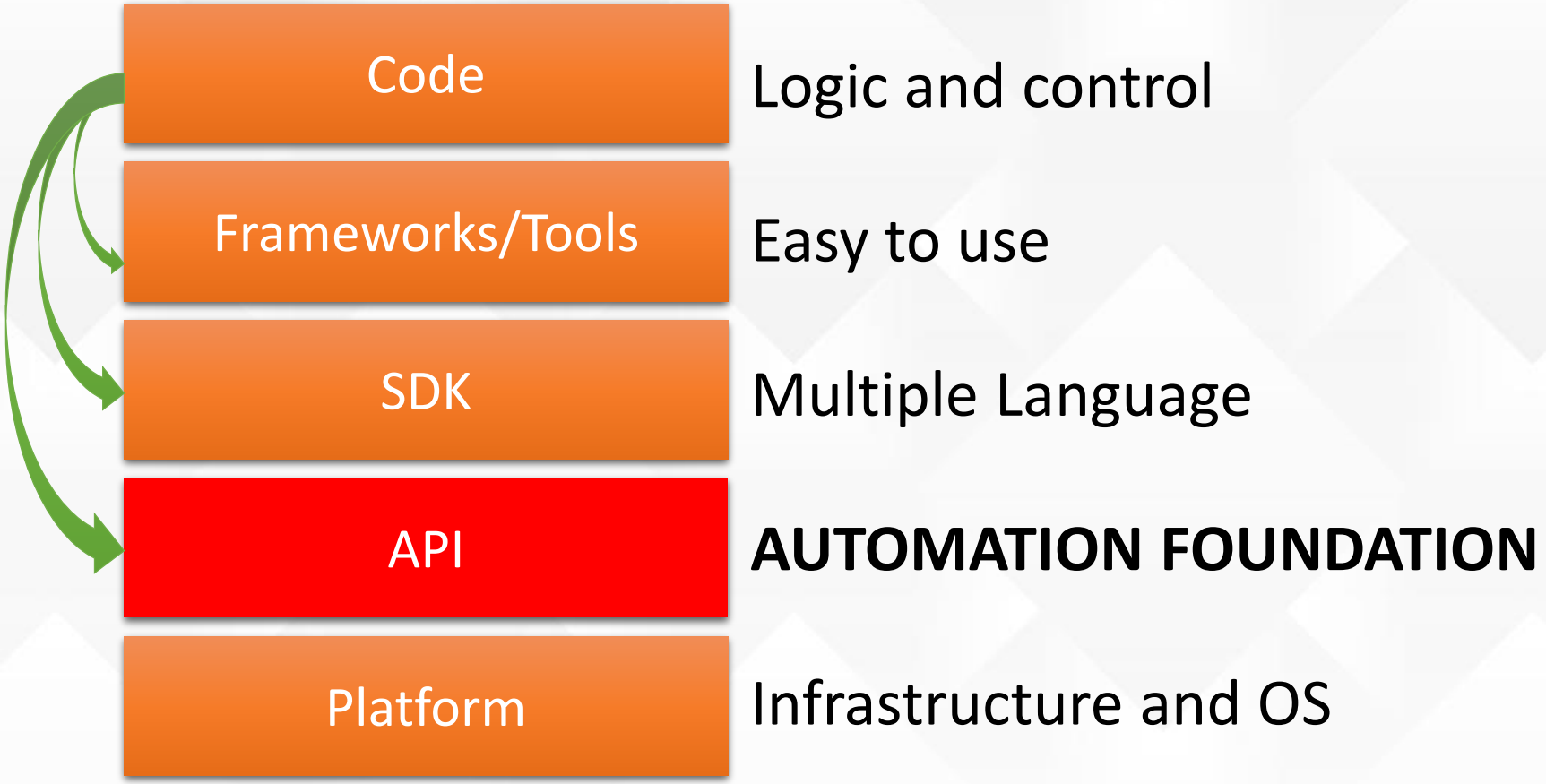


为什么需要DevOps?

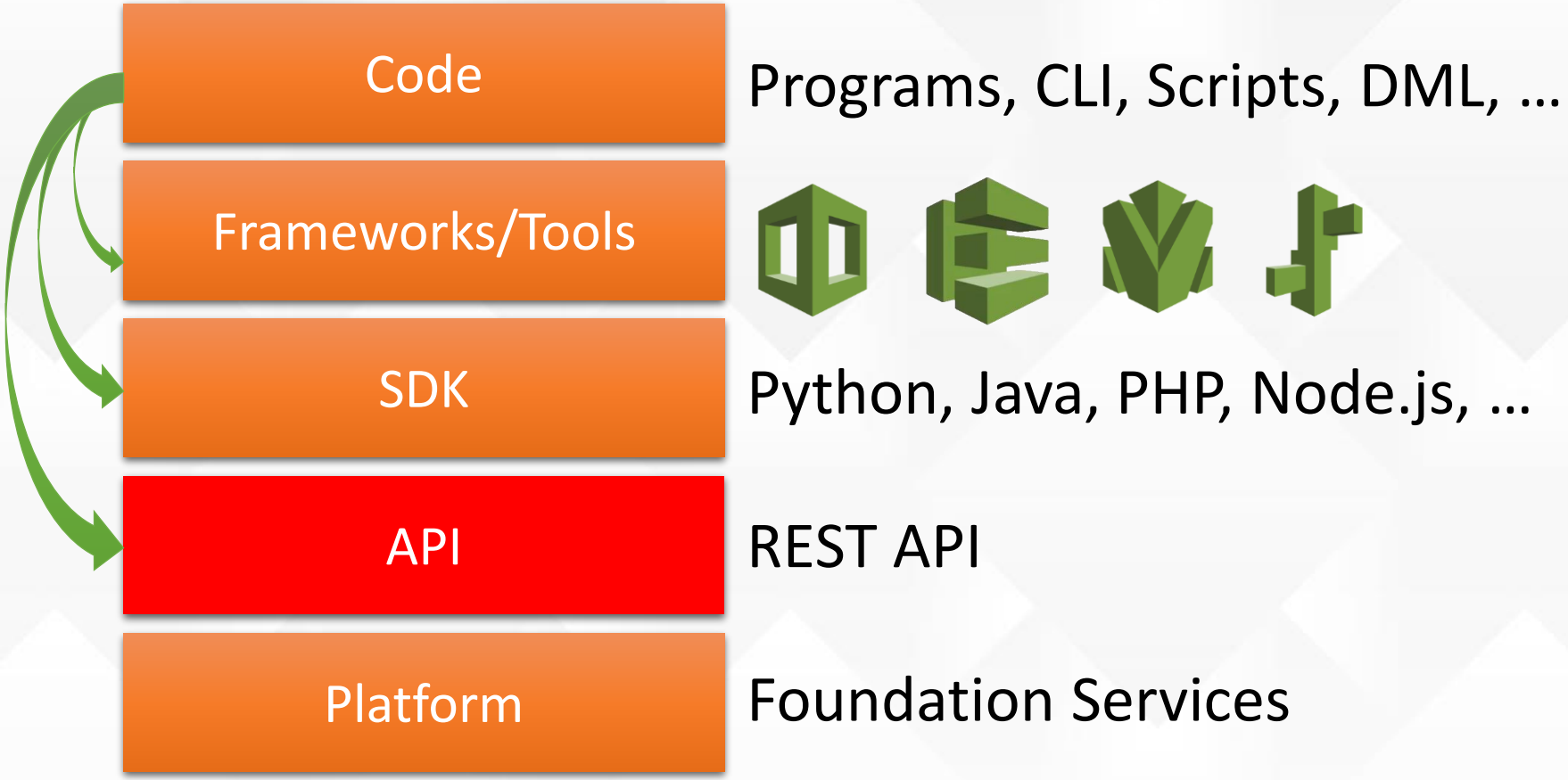


怎么实现DevOps?

从软件开发的角度看DevOps



AWS对DevOps的全面支持



基于AWS的DevOps实践要素

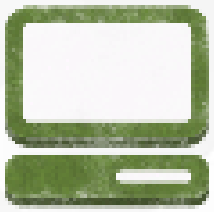
代码和命令行 CloudFormation CodeDeploy OpsWorks ElasticBeanstalk 第三方服务

API & SDK

AWS Services

代码和命令行

操作AWS服务的三种方式



Management Console

APIs and SDKs

Command Line Interface

DevOps基础

AWS Tools (SDK, CLI, IDE, etc.): http://aws.amazon.com/tools/

DEMO

Python Code – Start two EC2 instances

runec2.py

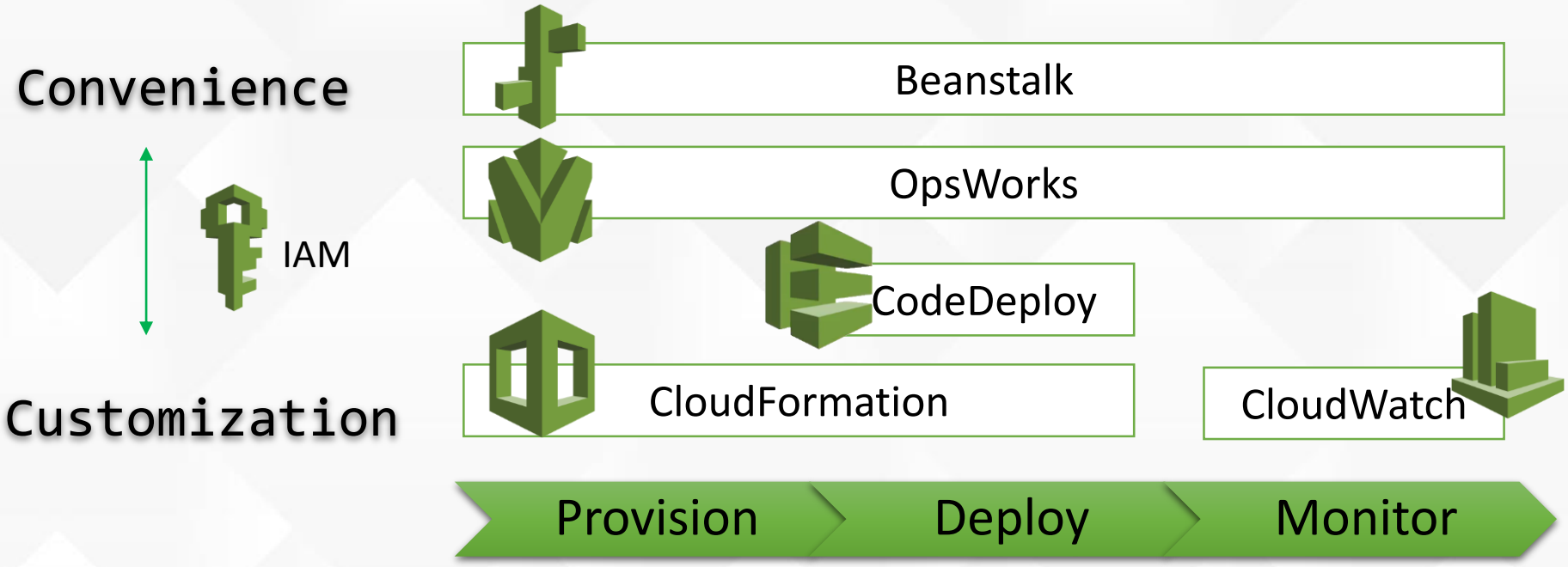
```
1  import boto.ec2
2
3  conn = boto.ec2.connect_to_region("cn-north-1")
4  conn.run_instances(
5      'ami-981d8fa1',
6      min_count=2,
7      max_count=2,
8      key_name='wendai-cn',
9      instance_type='t2.micro',
10     security_groups=['wslinux'],
11     )
12
```

CLI
- Add Tag to EC2 instances

```
aws ec2 create-tags
--resources i-ffb064c7 i-8eb561b6
--tags Key=Name,Value=QConEC2
```

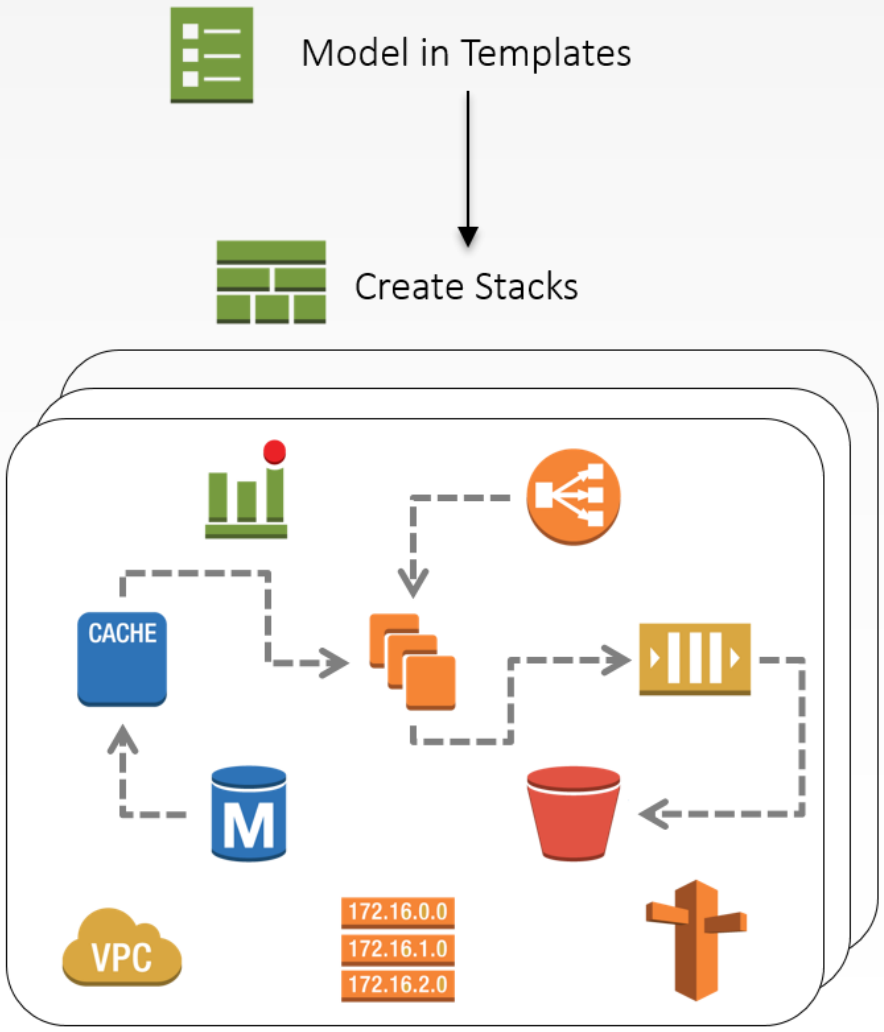
大规模基础架构的DevOps需要框架和工具

AWS DevOps服务适用场景



CloudFormation

基础平台模板化



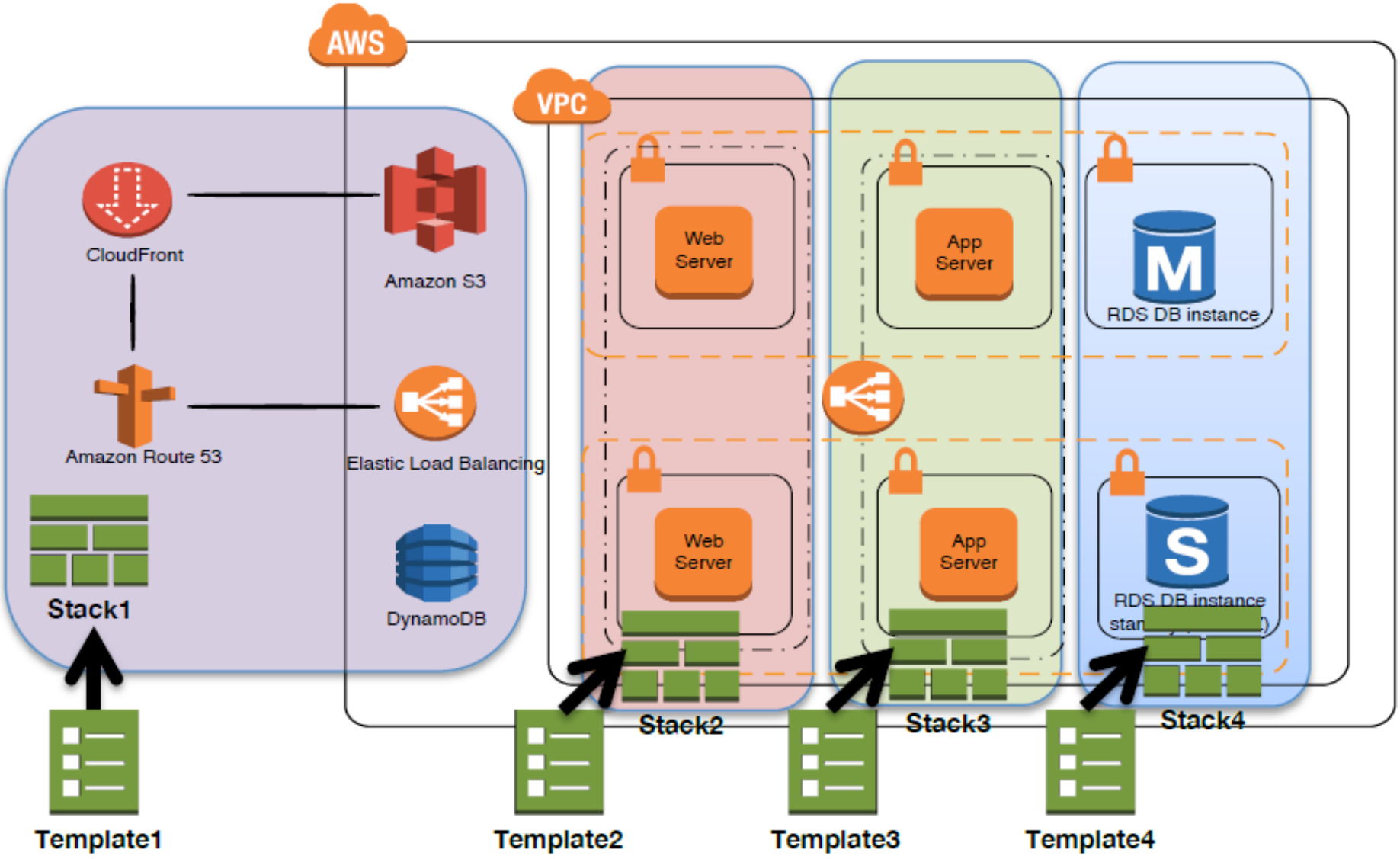
- 简化AWS服务的部署，快速部署一个Stack
- 模板化基础平台
- CloudFormation自动解决资源部署的先后和依赖关系
- 版本控制
- 第三方管理工具可以通过API集成CloudFormation

Infrastructure as Code

```
{
        "Description" : "Create an EC2 instance running the Amazon Linux 32 bit AMI.",
        "Parameters" : {
                "KeyPair" : {
                        "Description" : "The EC2 Key Pair to allow SSH access to the instance",
                        "Type" : "String"
                }
        },
        "Resources" : {
                "Ec2Instance" : {
                        "Type" : "AWS::EC2::Instance",
                        "Properties" : {
                                "KeyName" : { "Ref" : "KeyPair" },
                                "ImageId" : "ami-75g0061f",
                                "InstanceType" : "m1.medium"
                        }
                }
        },
        "Outputs" : {
                "InstanceId" : {
                        "Description" : "The InstanceId of the newly created EC2 instance",
                        "Value" : { "Ref" : "Ec2Instance" }
                }
        }
}
```

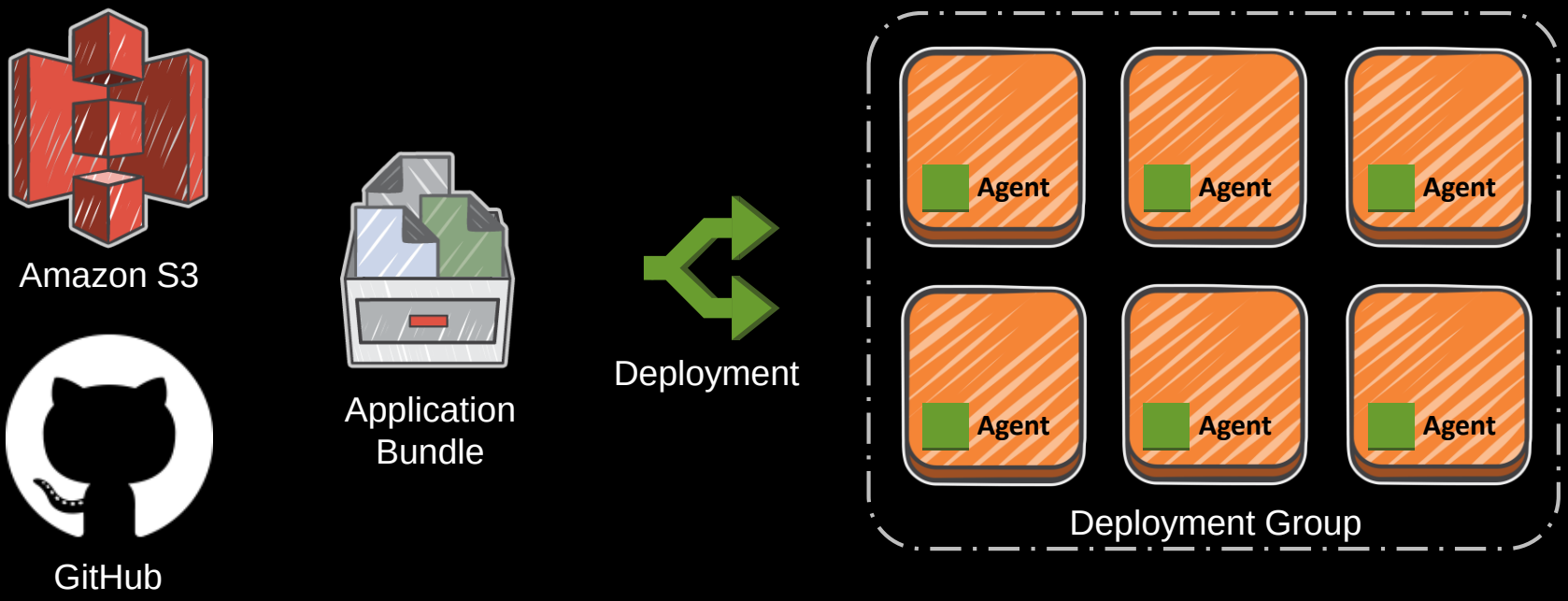
Samples: http://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/sample-templates-services-us-west-2.html

基于模板的快速部署


AWS
VPC
CloudFront
Amazon S3
Amazon Route 53
Elastic Load Balancing
DynamoDB
Stack1
Web Server
App Server
RDS DB instance
Web Server
App Server
RDS DB instance
Stack2
Stack3
Stack4
Template1
Template2
Template3
Template4

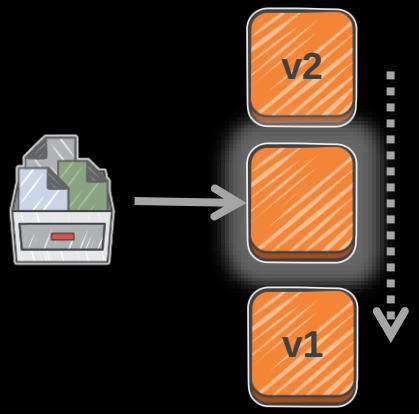
CodeDeploy

自动化应用部署

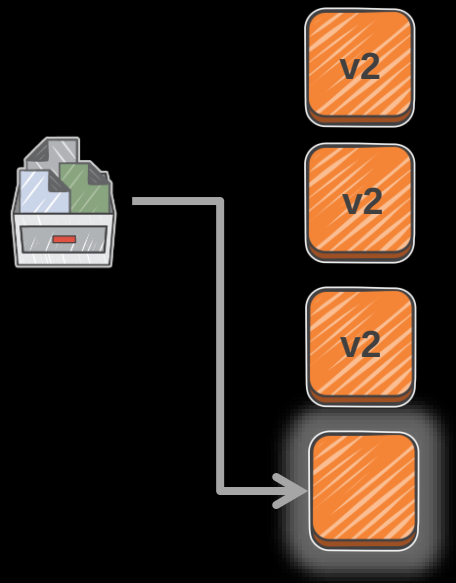

Amazon S3
GitHub
Application Bundle
Deployment
Agent
Agent
Agent
Agent
Agent
Agent
Deployment Group

Deployment Configuration

Rolling updates



Auto Scaling support



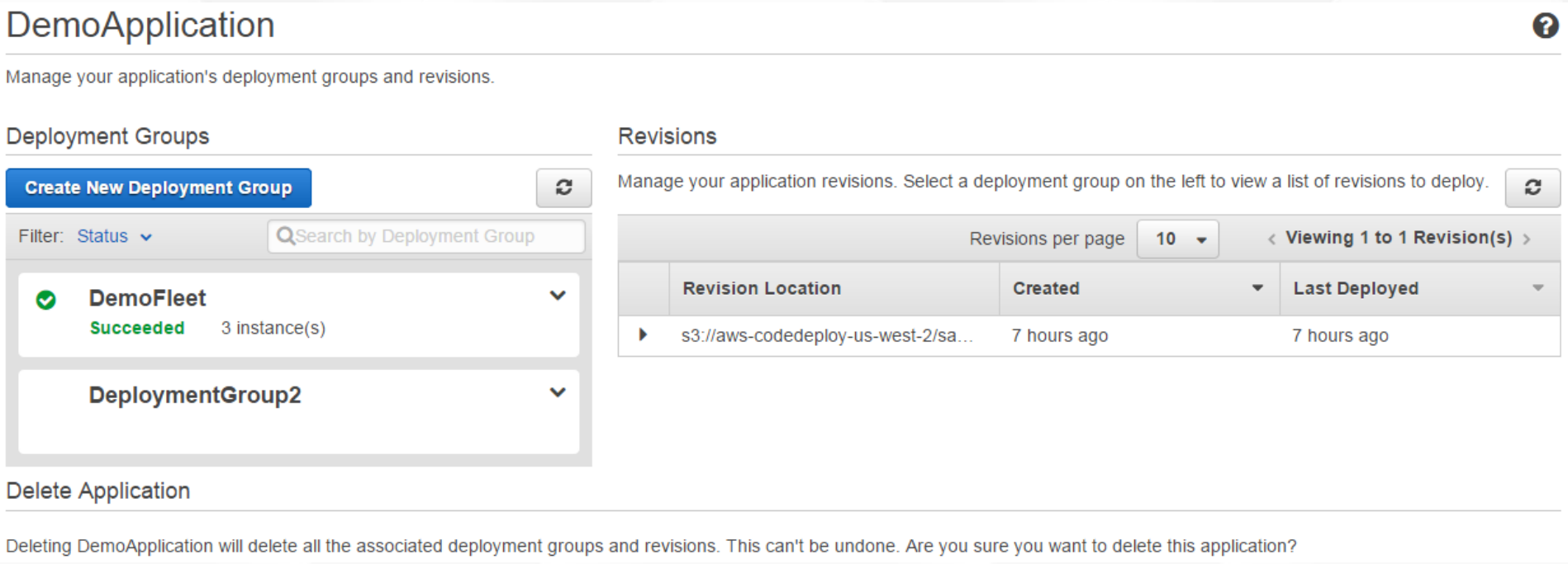
Customized Install

```
files:
  - source: /web_files/
    destination: /var/www/html/
hooks:
  BeforeInstall:
    - location: setup/install_dep.sh
  ApplicationStart:
    - location: setup/start_server.sh
    - location: setup/start_logger.sh
  ApplicationStop:
    - location: setup/stop_server.sh
    - location: setup/flush_logs.sh
```

Application Specification File

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/html/WordPress
hooks:
  BeforeInstall:
    - location: scripts/install_dependencies.sh
      timeout: 300
      runas: root
  AfterInstall:
    - location: scripts/change_permissions.sh
      timeout: 300
      runas: root
  ApplicationStart:
    - location: scripts/start_server.sh
      timeout: 300
      runas: root
  ApplicationStop:
    - location: scripts/stop_server.sh
      timeout: 300
      runas: root
```

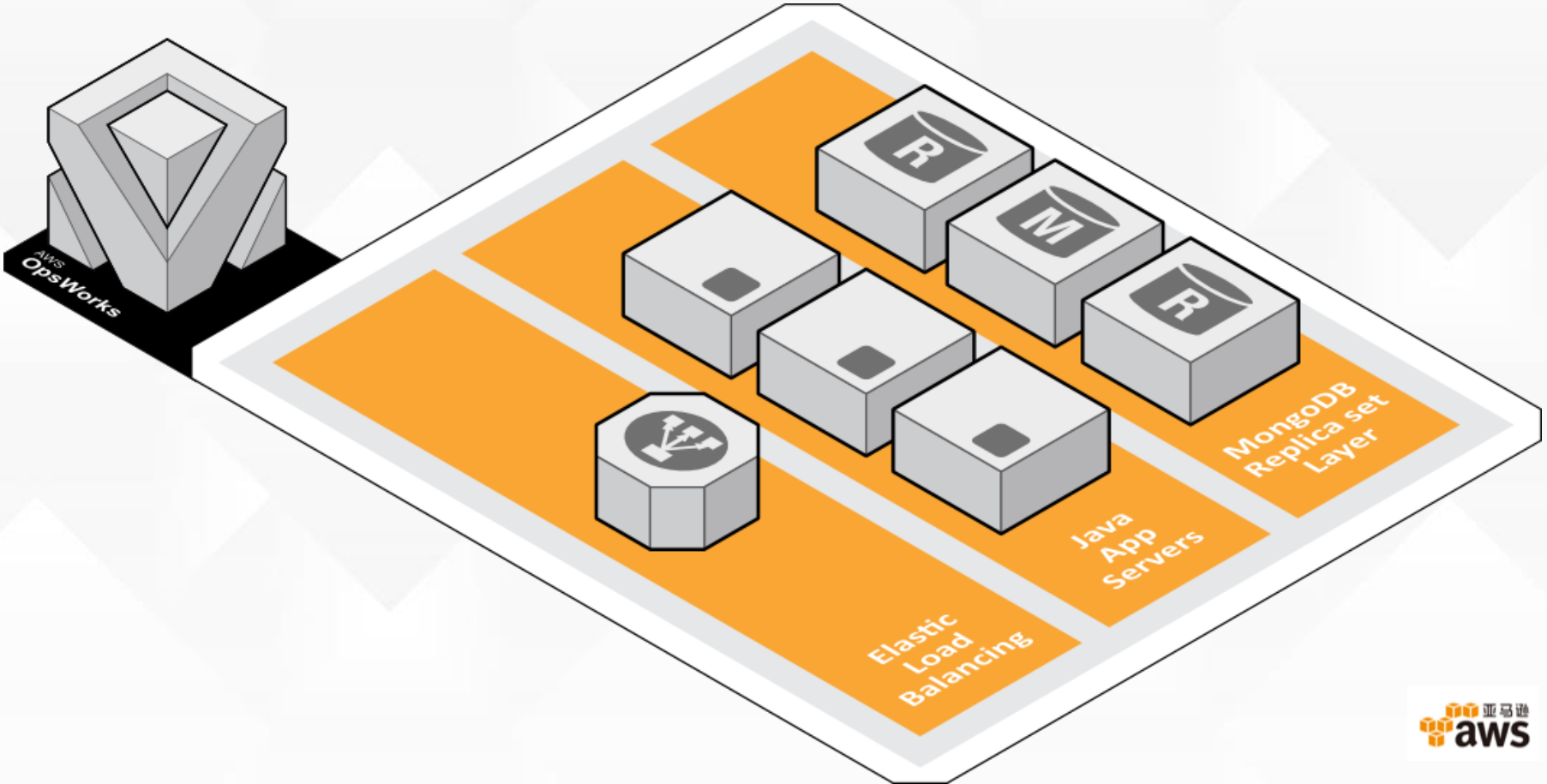
Demo Snapshot



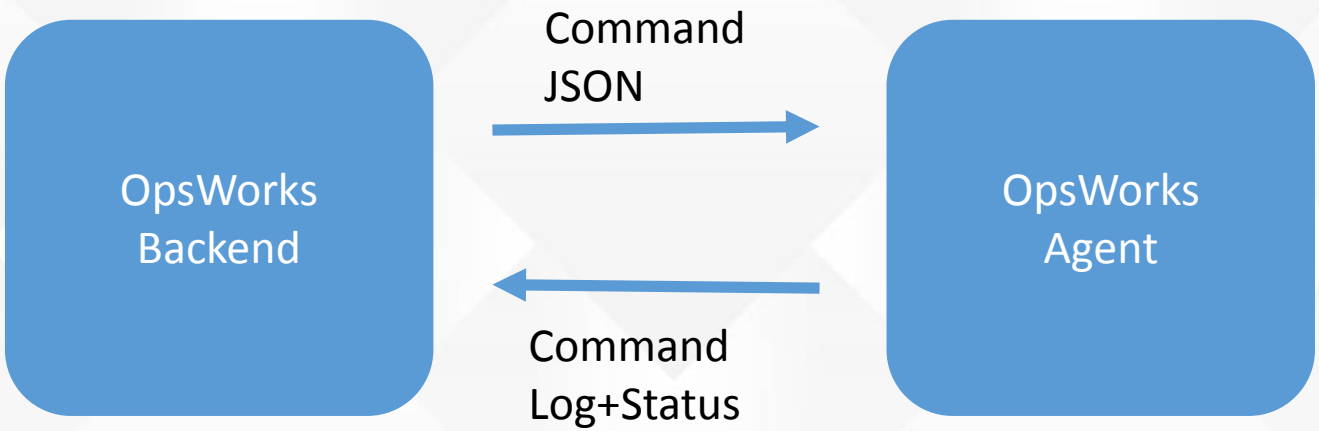
OpsWorks

OpsWork

AWS OpsWorks is a flexible application management solution with automation tools that enable you to model and control your applications and their supporting infrastructure.



OpsWorks工作原理



Amazon EC2, Amazon EBS, EIP,
Amazon VPC, Elastic Load Balancing….
Auto-Scaling, Auto-Healing….

On-instance execution via
Chef client/zero

大大简化了Chef环境的搭建

Lifecycle events

setup

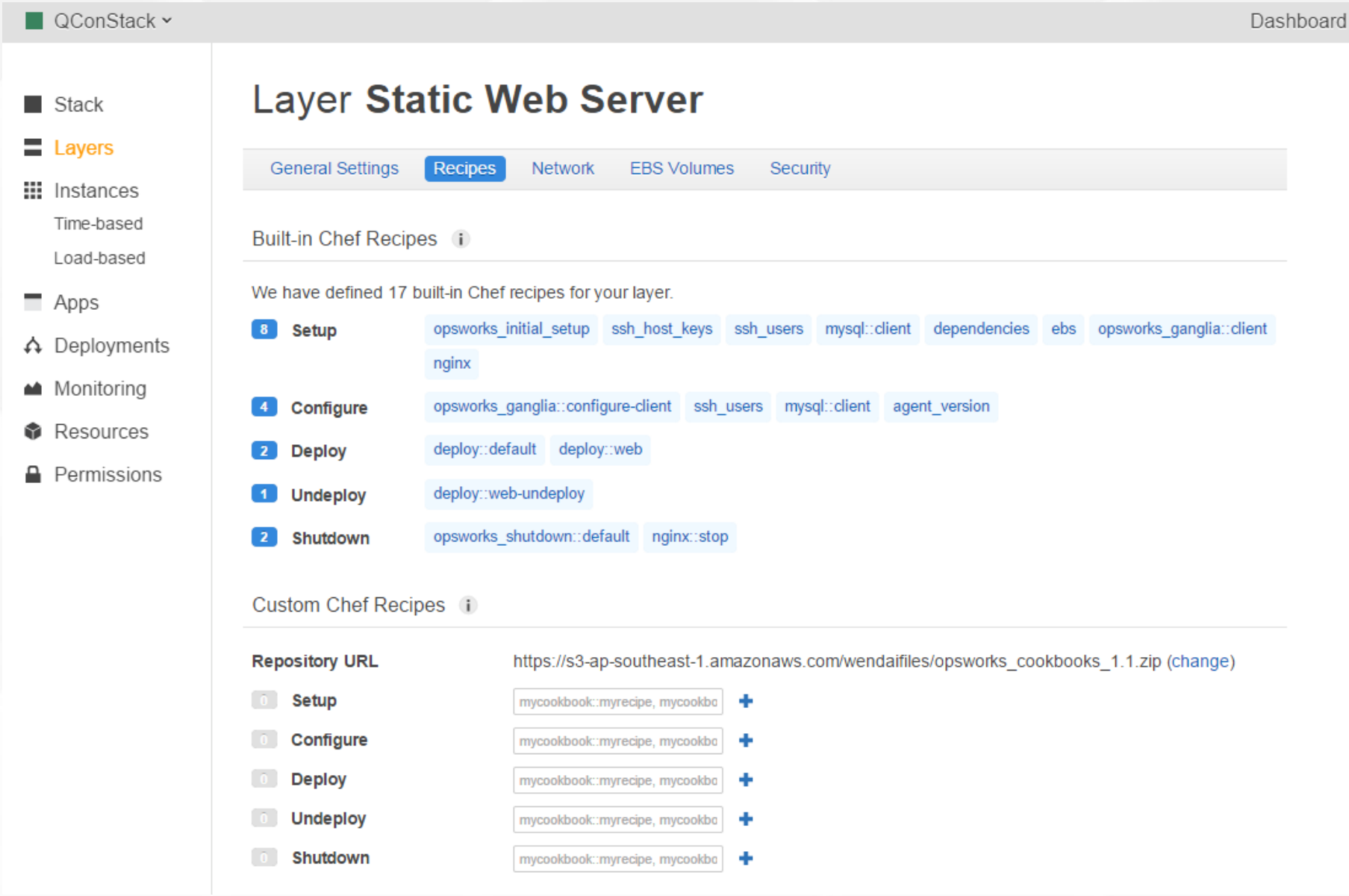
configure

deploy

undeploy

shutdown

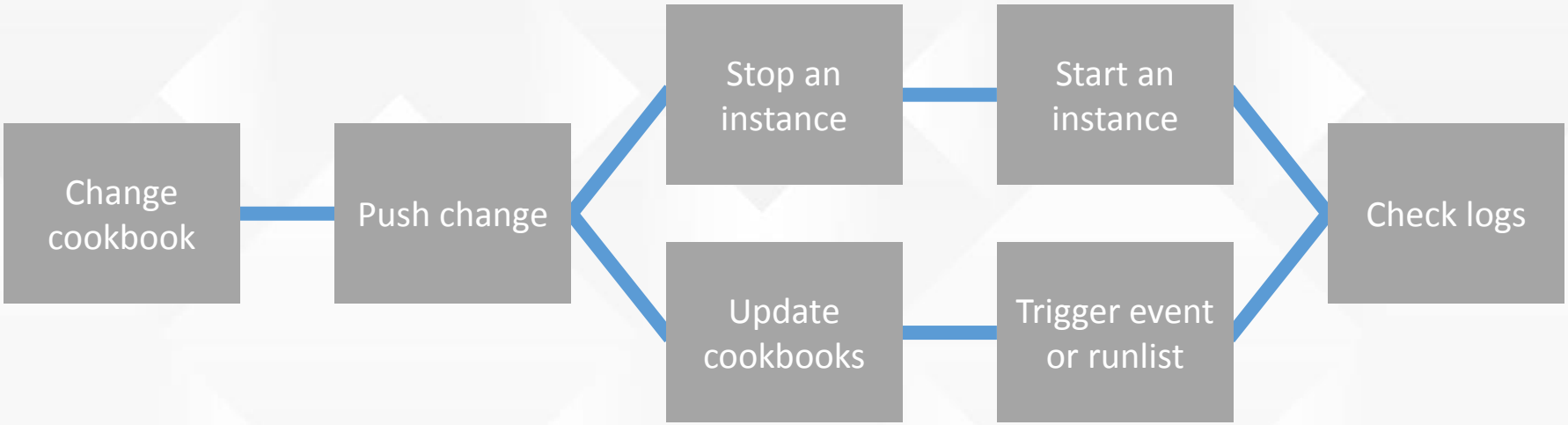
Built-in and Custom Chef Recipes



Custome Recipe Demo – 创建目录

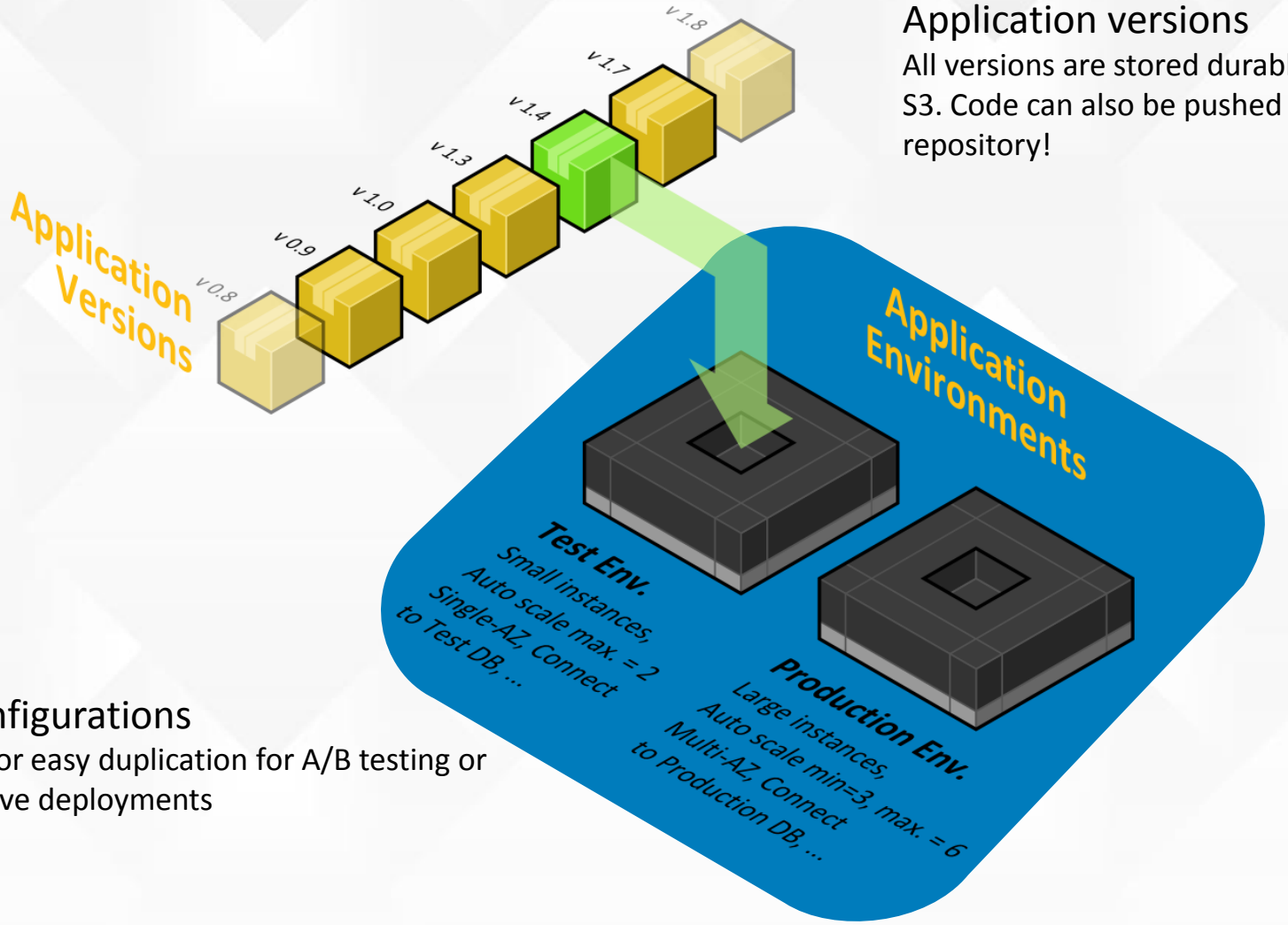
```
Chef::Log.info("******Creating a data directory.******")

data_dir = value_for_platform(
  "centos" => { "default" => "/srv/www/shared" },
  "ubuntu" => { "default" => "/srv/www/data" },
  "default" => "/srv/www/config"
)

directory data_dir do
  mode 0755
  owner 'root'
  group 'root'
  recursive true
  action :create
end
~
~
```

Cookbook development workflow



ElasticBeanstalk

Elastic Beanstalk 工作原理



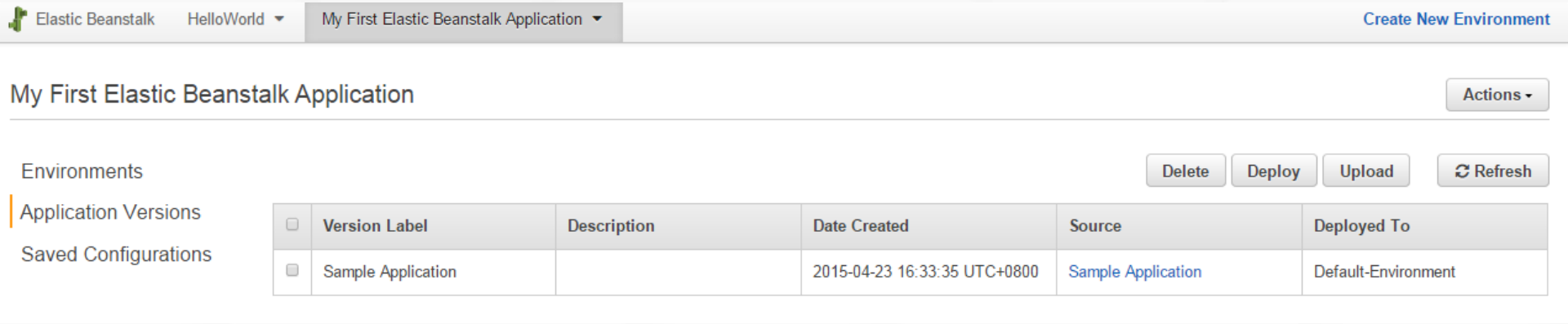
Application versions

All versions are stored durably in Amazon S3. Code can also be pushed from a Git repository!

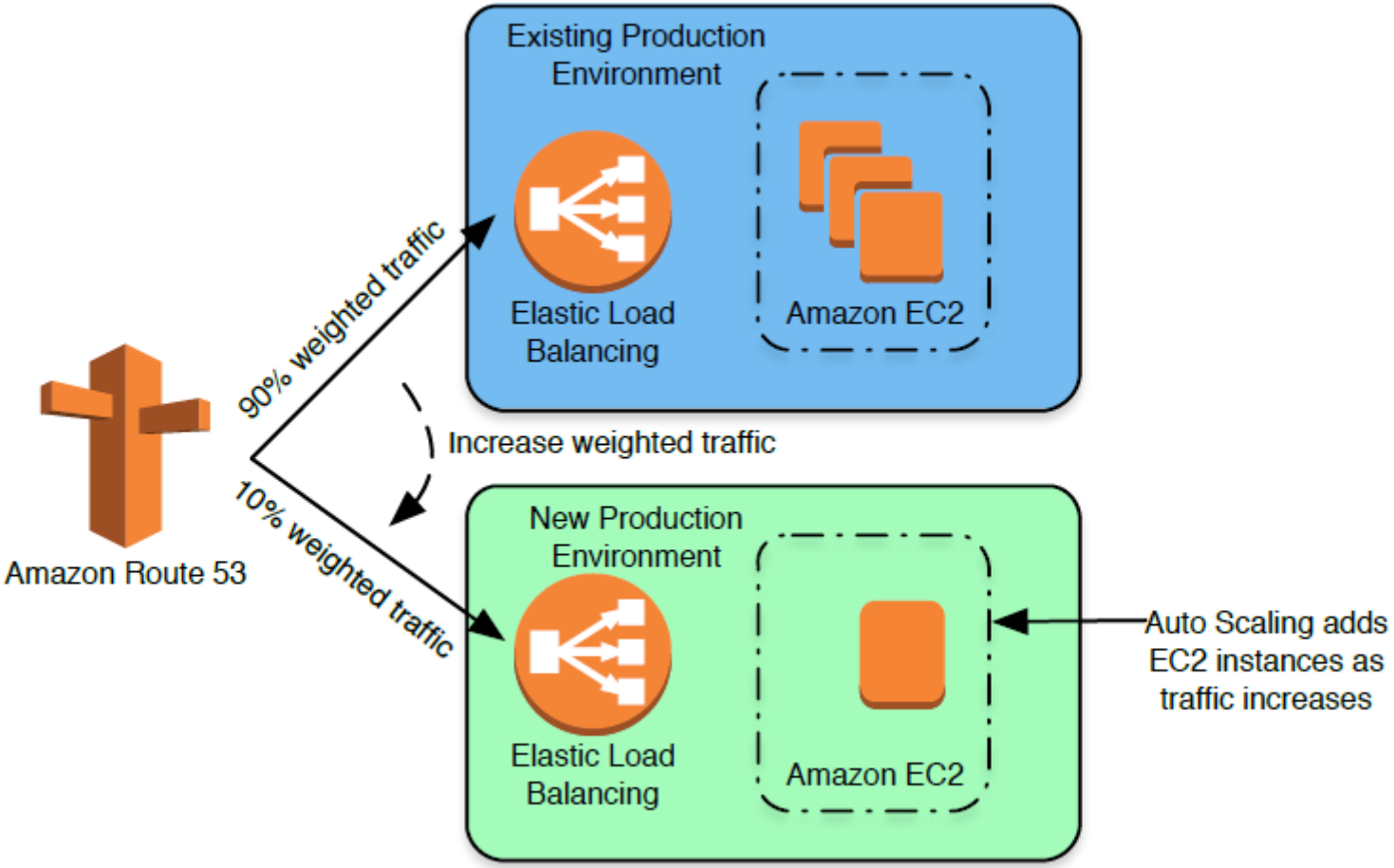
Saved configurations

Save these for easy duplication for A/B testing or non-disruptive deployments

Application



蓝绿部署


Existing Production Environment
Elastic Load Balancing
Amazon EC2
90% weighted traffic
Increase weighted traffic
10% weighted traffic
Amazon Route 53
New Production Environment
Elastic Load Balancing
Amazon EC2
Auto Scaling adds EC2 instances as traffic increases

Demo: CLI workflow

Initial app deployment:

01 Initialize your Git repository

```
$ git init .
```

02 Create your Elastic Beanstalk app

```
$ eb init
```

03 *Follow the prompts to configure the environment*

04 Add your code

```
$ git add .
```

05 Commit

```
$ git commit –m “v1.0”
```

06 Create the resources and launch the application

```
$ eb create
```

Demo: CLI workflow

Update your app:

01 Update your code

02 Push the new code

```
$ git add .
$ git commit –m “v2.0”
$ eb deploy
```

03 Monitor the deployment progress

```
$ eb status
```

贯彻始终的安全与监控

Amazon CloudWatch

Monitor resources

AWS IAM (Identity & Access Mgmt)

Manage **users, groups & permissions**

Convenience

IAM

Customization

Beanstalk

OpsWorks

CodeDeploy

CloudFormation

CloudWatch

Provision Deploy Monitor

感谢您宝贵的时间！