



AWS Summit

AWS技术峰会 2015 · 上海





深入浅出AWS基础架构运维自动化

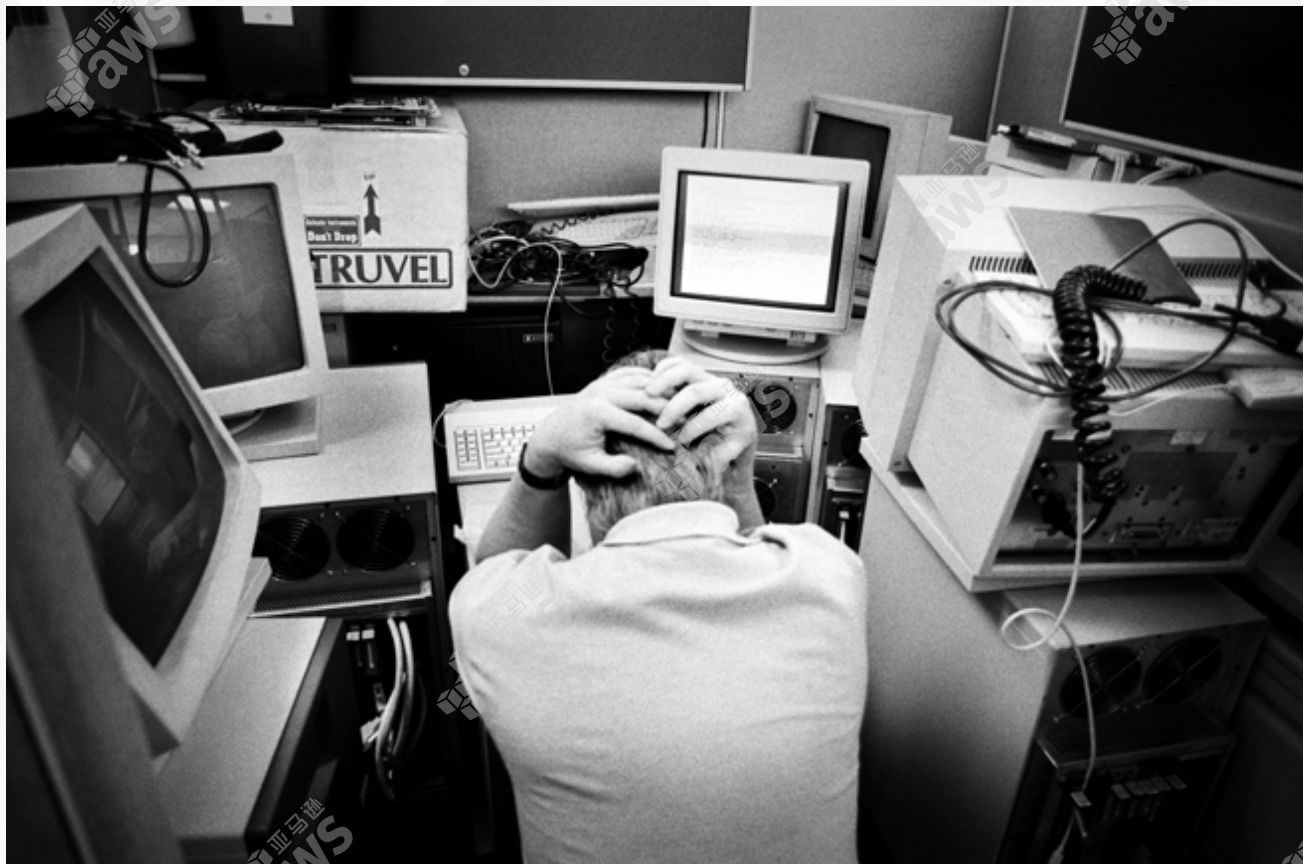
肖凌 – AWS解决方案架构师



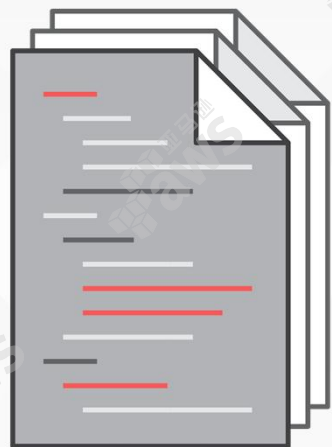
我需要架构运维自动化吗？

- 需求1 – 请在这个礼拜五之前帮我准备一套Rail网站开发环境！
- 需求2 – 请在明天下班之前把所有的CentOS打上HeartBleed安全补丁！
- 需求3 – 请在周三下午一起讨论怎么搭建新的一套测试环境供电商开发组使用！
- 需求4 – 我怎么管理数以百计的计算和存储资源？
-
- 需求 N

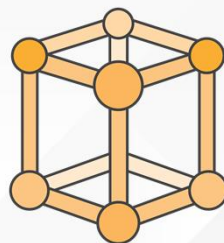
这是要上演上个世纪的悲剧吗？



AWS云的时代—基础架构软件化即Infrastructure As a Code



基础架构软件化



架构优化



管理统一



高效部署



系统安全

用软件方式定义与运行基础架构

应用程序

基础架构

用软件方式定义与运行基础架构

应用程序



源代码

基础架构



JSON

用软件方式定义与运行基础架构

应用程序



源代码



编译器

基础架构



JSON



AWS Service API

用软件方式定义与运行基础架构

应用程序



源代码



编译器



运行的应用程序

基础架构



JSON



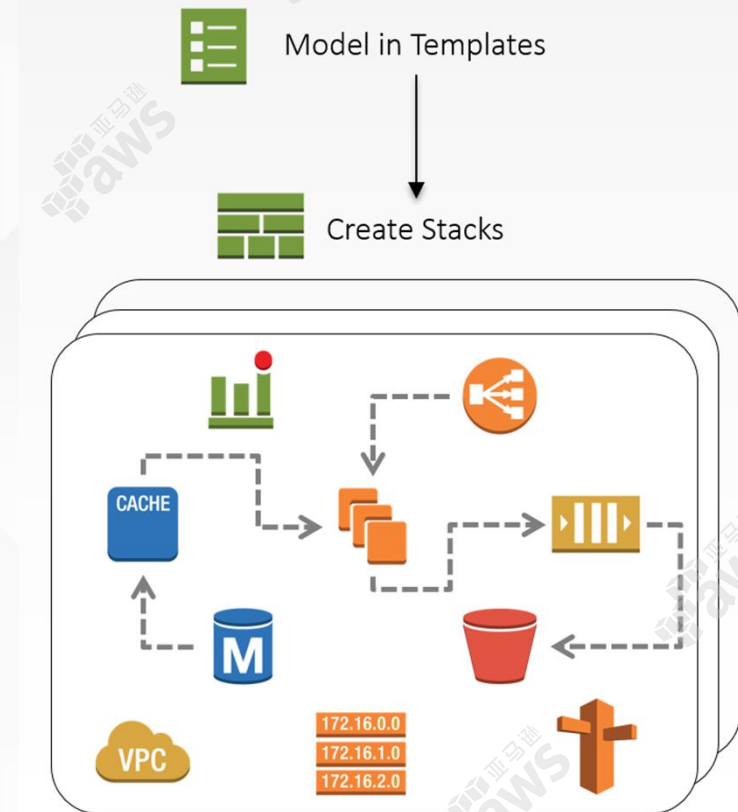
AWS Service API



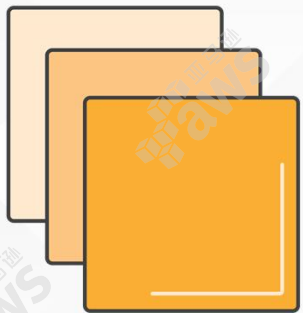
运行的基础架构

AWS架构运维利器CloudFormation

- 用JSON模板的方式部署你的架构和应用程序
- 根据脚本定义，自动化快速的做到架构复制（跨区域），修改以及软件版本控制。
- 所有的AWS资源可以在模板中方便定义，自动生成。
- 与CI/CD集成，做到DevOps。



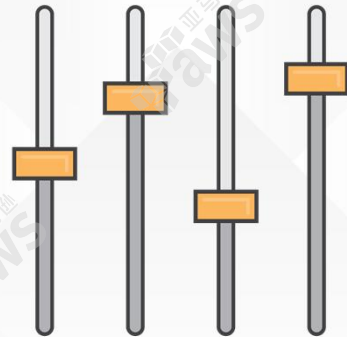
快速上手CloudFormation



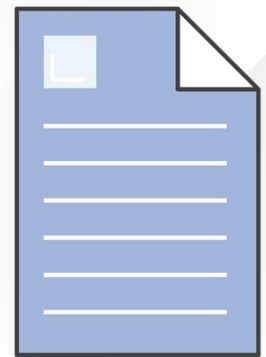
堆栈



模板



参数



策略

模板



CloudFormation



堆栈



JSON格式文件

CloudFormation 生成过程

堆栈生成

参数定义
资源创建
动态配置

堆栈生成
堆栈更新
错误发现与自动回滚

AWS所有服务支持
区域定义
定制化

实例1 - 使用模板生成EC2实例

```
{
  "Description": "Create an EC2 instance running the latest Amazon Linux AMI.",
  "Parameters": {
    "KeyPair": {
      "Description": "The EC2 Key Pair to allow SSH access to the instance",
      "Type": "String"
    }
  },
  "Resources": {
    "Ec2Instance": {
      "Properties": {
        "ImageId": "ami-9d23aeea",
        "InstanceType": "m3.medium",
        "KeyName": {
          "Ref": "KeyPair"
        }
      },
      "Type": "AWS::EC2::Instance"
    }
  },
  "Outputs": {
    "InstanceId": {
      "Description": "The InstanceId of the newly created EC2 instance",
      "Value": {
        "Ref": "Ec2Instance"
      }
    }
  },
  "AWSTemplateFormatVersion": "2010-09-09"
}
```

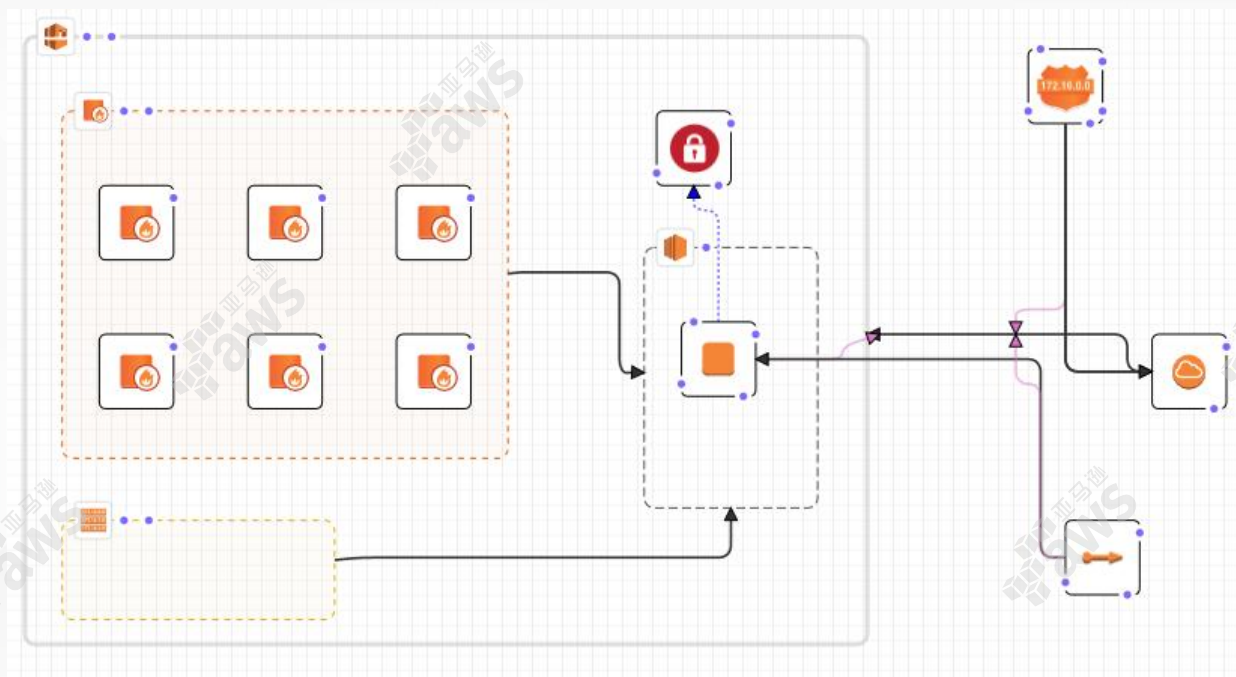
参数输入

资源定义

输出

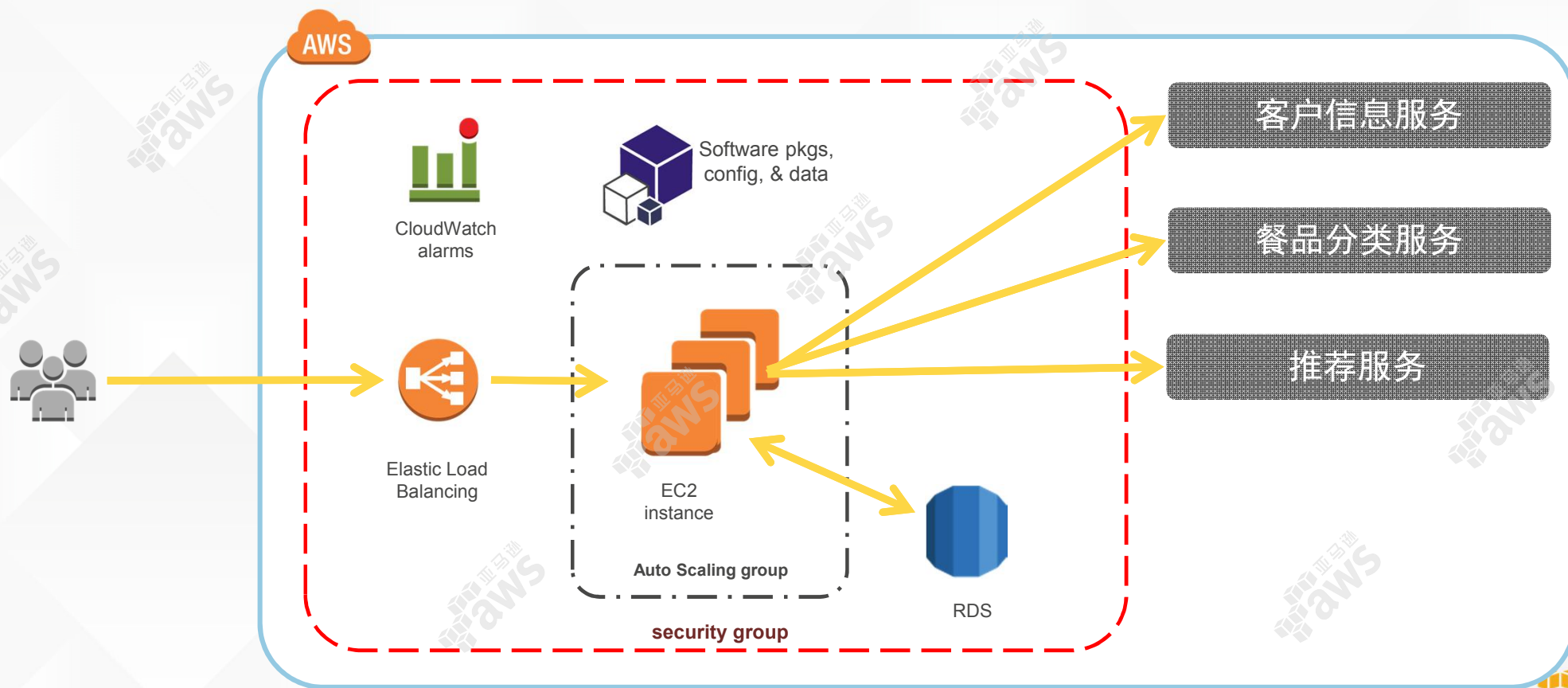
可视化设计 – CloudFormation Designer

- 可视化模板资源
- 即托即用式IDE编辑
- 定制化模板

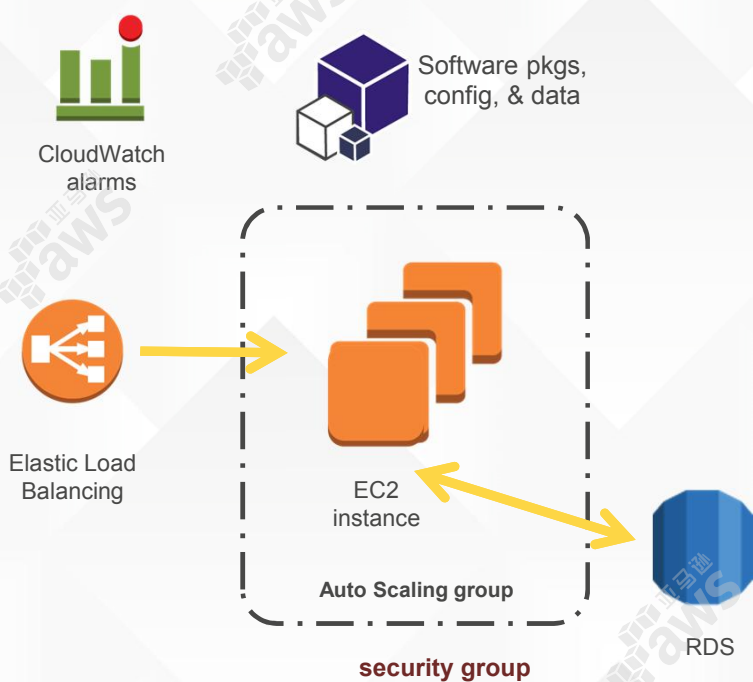


演示1 – 利用CloudFormation自动化部署WordPress网站

实例2 – 使用模板构建订餐系统



创建模板 - 资源



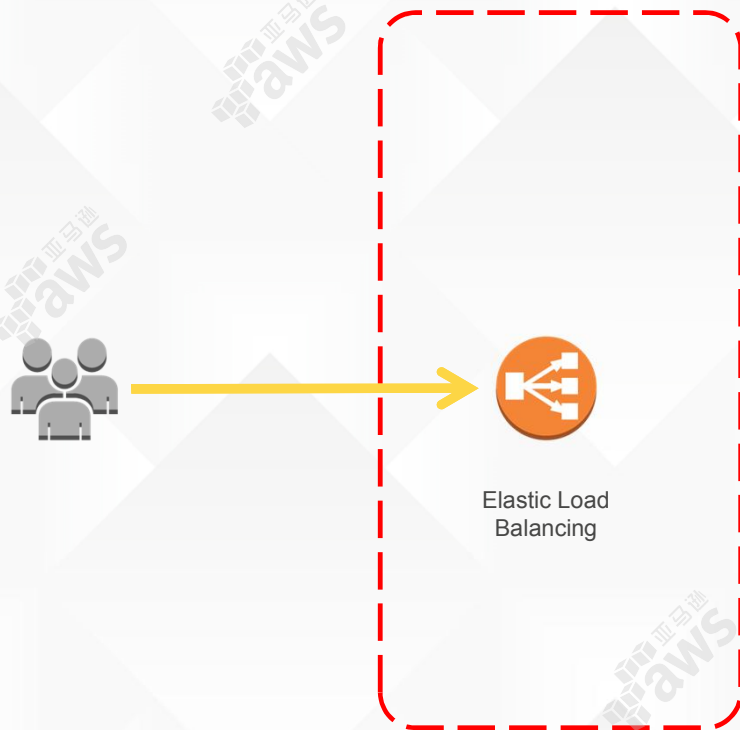
```
"Resources" : {  
  "SecurityGroup" : {},  
  "WebServerGroup" : {  
    "Type" : "AWS::AutoScaling::AutoScalingGroup",  
    "Properties" : {  
      "MinSize" : "1",  
      "MaxSize" : "3",  
      "LoadBalancerNames" : [ { "Ref" :  
"LoadBalancer" } ],  
      ...  
    }  
  },  
  "LoadBalancer" : {},  
  "CacheCluster" : {},  
  "Alarm" : {}  
},
```

创建模板 - 参数输入



```
"Parameters" : {  
  "CustomerDBServiceEndPoint" : {  
    "Description" : "URL of the Customer DB Service",  
    "Type" : "String"  
  },  
  "CustomerDBServiceKey" : {  
    "Description" : "API key for the Customer DB  
Service",  
    "Type" : "String",  
    "NoEcho" : "true"  
  },  
  "InstanceType" : {  
    "Description" : "WebServer EC2 instance type",  
    "Type" : "String",  
    "Default" : "m3.medium",  
    "AllowedValues" :  
["m3.medium","m3.large","m3.xlarge"],  
    "ConstraintDescription" : "Must be a valid  
instance type"  
  }  
}
```

创建模板 - 输出



```
"Resources" : {  
  "LoadBalancer" : {},  
  ...  
},  
"Outputs" : {  
  "WebsiteDNSName" : {  
    "Description" : "The DNS name of the website",  
    "Value" : {  
      "Fn::GetAtt" : [ "LoadBalancer", "DNSName" ]  
    }  
  }  
}
```

创建模板 – 部署和配置软件



```
"AWS::CloudFormation::Init": {  
  "webapp-config": {  
    "packages" : {}, "sources" : {}, "files" : {},  
    "groups" : {}, "users" : {},  
    "commands" : {}, "services" : {}  
  },  
  
  "chef-config" : {}  
}
```

- ✓ 声明
- ✓ 可调试
- ✓ 可更新
- ✓ 安全
- ✓ BIOT™ (Bring In Other Tools)

通过AWS API接口自动化生成堆栈

```
aws cloudformation create-stack  
  --stack-name ec2InstanceCmdLineDemo  
  --template-url https://s3-cn-north-1.amazonaws.com.cn/cf-templates-  
1fhelryvrdrbr-eu-west-1/2014174d0r-ec2Instance.template  
  --parameters ParameterKey=KeyPair,ParameterValue=ManagementKeyPair
```

AWS CloudFormation CLI命令

cancel-update-stack

create-stack

delete-stack

describe-stack-events

describe-stack-resource

describe-stack-resources

describe-stacks

get-stack-policy

get-template

list-stack-resources

list-stacks

set-stack-policy

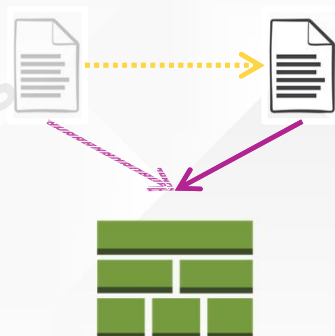
update-stack

validate-template

\$ aws cloudformation update-stack help

更新堆栈

直接更新



快速

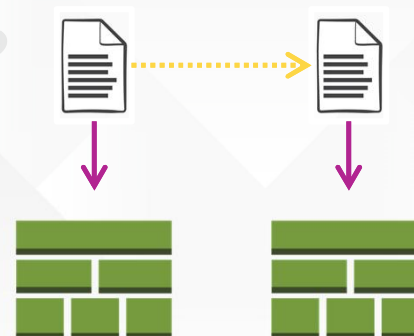
成本优化

更加简单的状态
和数据迁移

模板

堆栈

蓝绿部署



应用请求

需要切换但对生产
运行时没有影响

使用Rolling Update实现在线升级



ELB健康检查

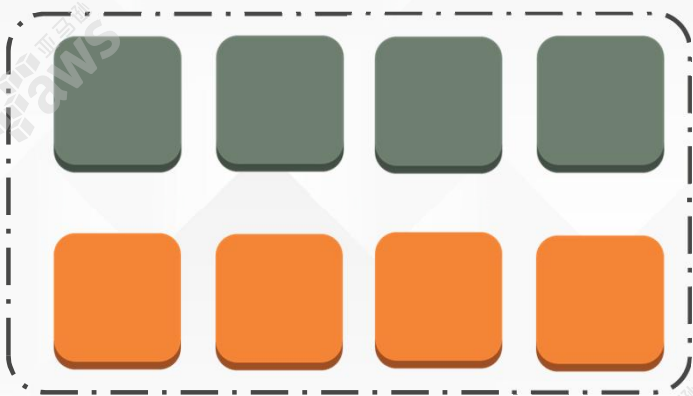
1. Update一组EC2实例，其他实例仍然执行当前进程。

2. ELB健康检查校验。

3. 完成ELB检查，继续下一组EC2更新。

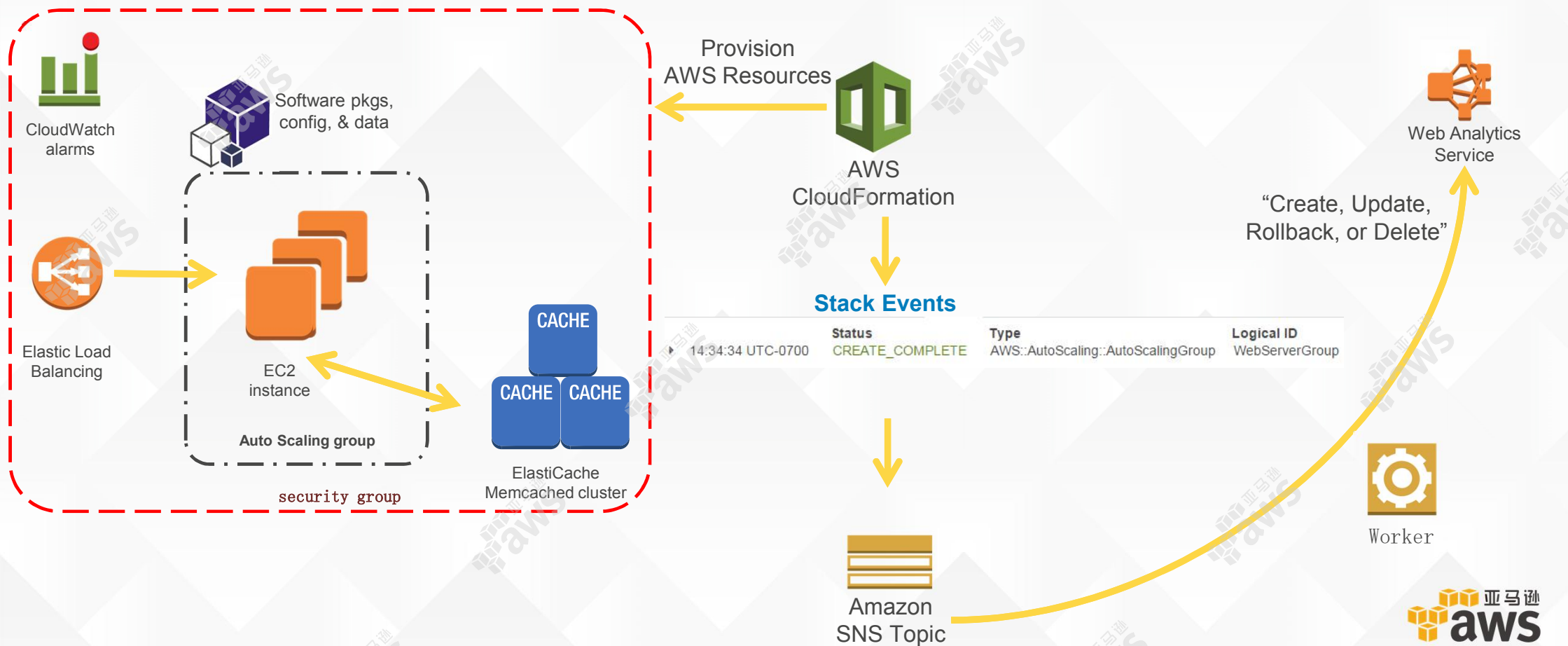
4. 如果失败，回滚（RollBack）CloudFormation。

单一模板内自动完成

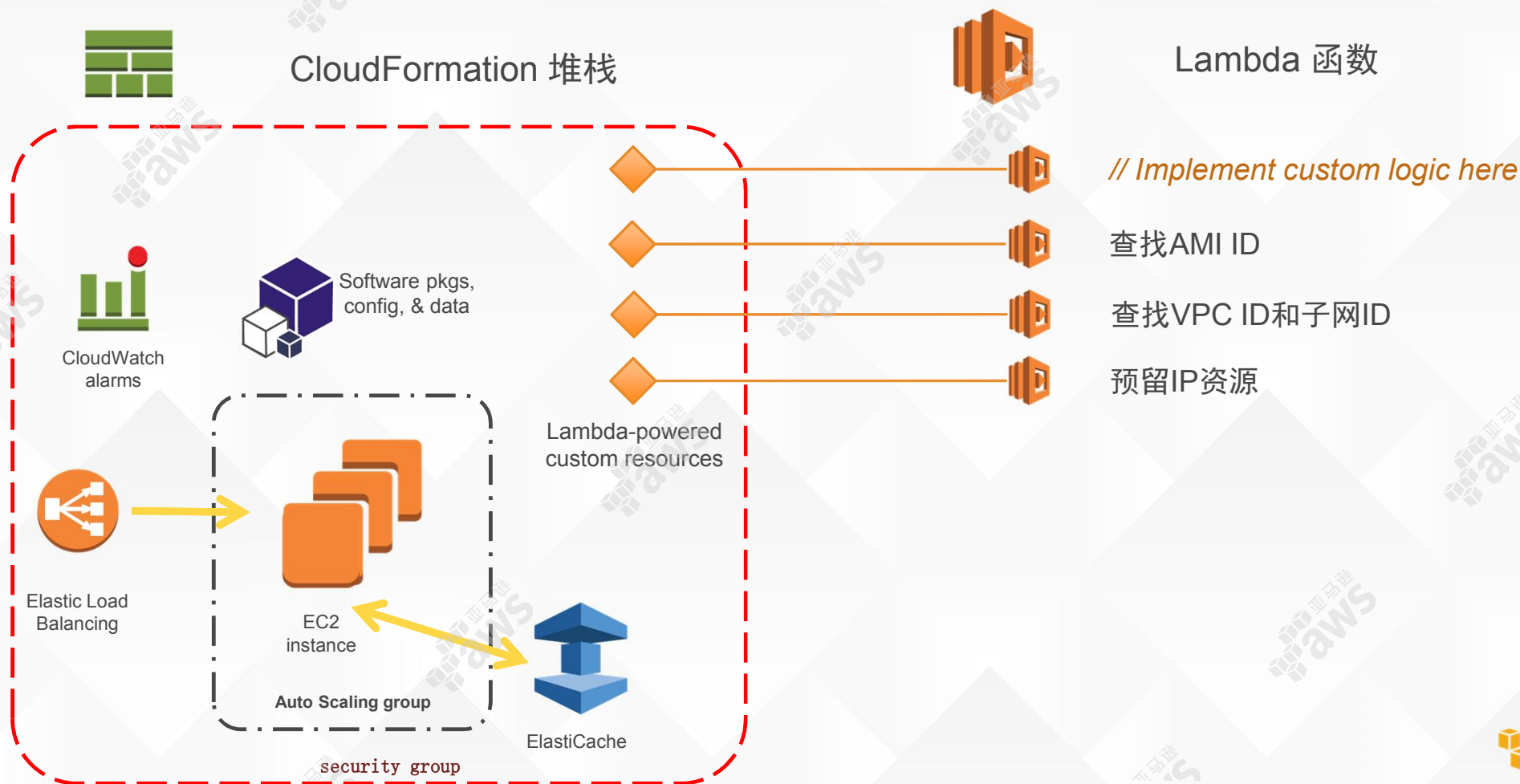


Auto Scaling group

让CloudFormation和AWS平台服务无缝整合



利用Lambda做动态定制化配置操作



使用CloudFormation和OpsWorks做集成化应用架构部署

基础架构生成

CloudFormation

- 模板化
- 复制
- 自动化

EC2

SQS, SNS, Kinesis, etc.

databases

VPC

IAM

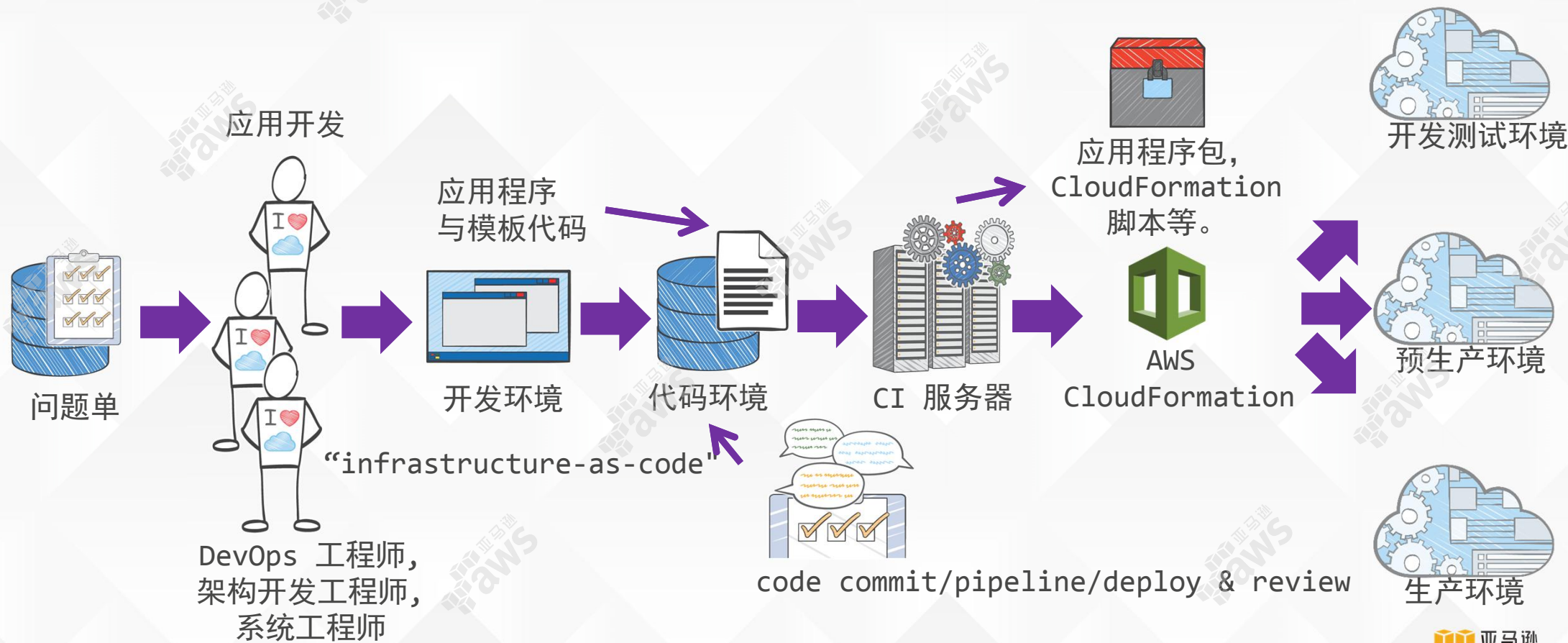
应用部署

下载, 安装软件
配置应用程序
升级重启

OpsWorks

- 应用程序生存周期管理
- 交互式应用程序管理

在CI/CD中利用CloudFormation自动化DevOps



对了， CloudFormation 功能是免费的哦！

- CloudFormation功能本身是免费的！
- 用户只需要对生成的资源付费



AWS架构运维利器大盘点

- **OpsWorks**
- **Code Commit / Code Pipeline/ CodeDeploy**
- **Boto – AWS Python SDK**
- **丰富的集成工具和SDK**

参考文档：

- AWS CloudFormation官方文档：
http://docs.aws.amazon.com/zh_cn/AWSCloudFormation/latest/UserGuide/Welcome.html
- AWS OpsWorks官方文档: https://aws.amazon.com/cn/opsworks/?nc1=h_ls
- Boto AWS Python开发工具: <https://aws.amazon.com/cn/sdk-for-python/>



Section Title



Thank You

