



2012云计算架构师峰会

Cloud Computing Architects Summit China 2012

揭示企业级IT架构转型 分享最新技术的应用落地



Cloud Foundry PaaS平台对软件开发的影响

What is PaaS?

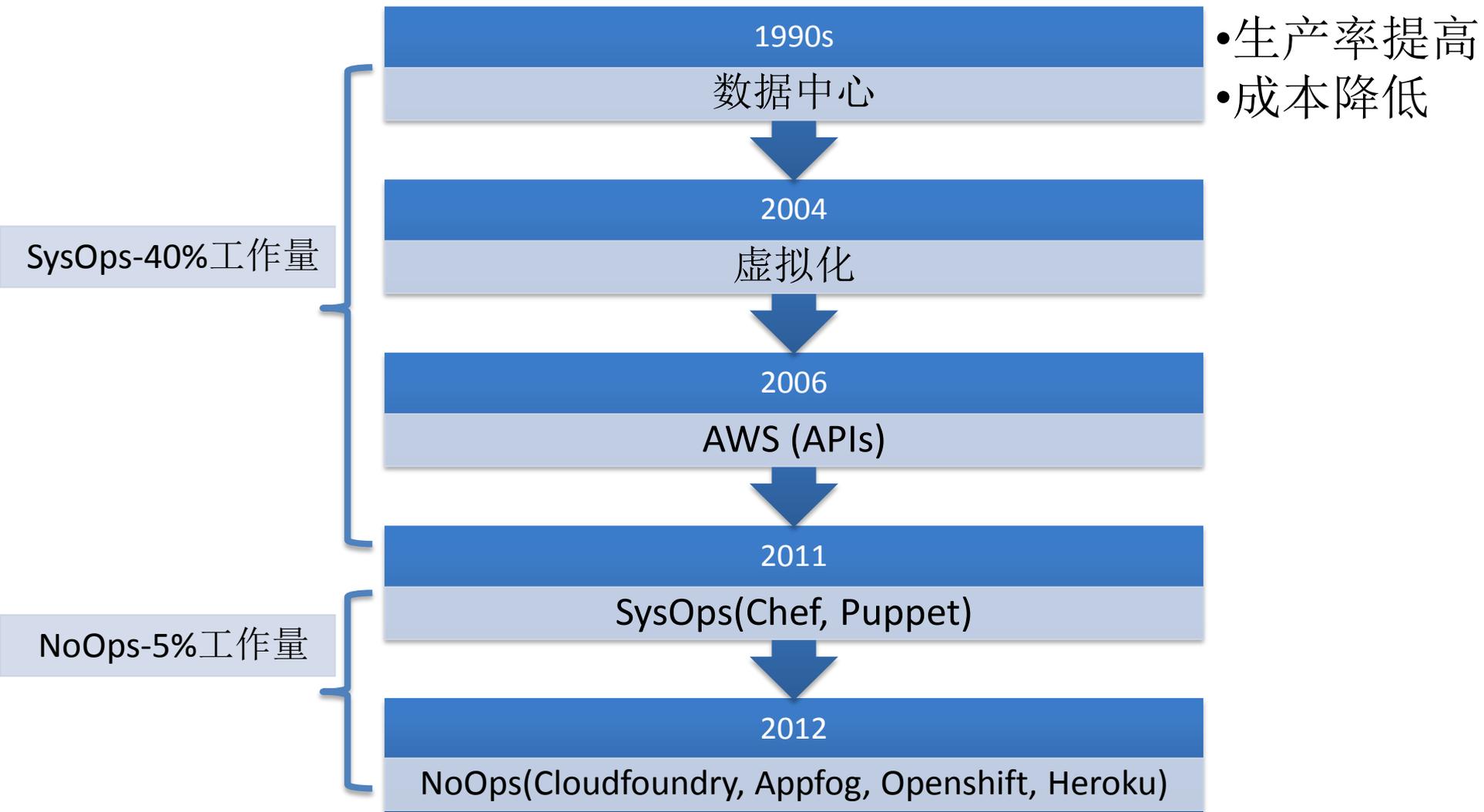
“The capability [...] to deploy onto the cloud infrastructure [...] applications created using programming languages, libraries, services, and tools supported by the provider.”

The NIST Definition of Cloud Computing

Why PaaS?

- 快速响应市场，缩短反馈周期（Time to Market）
- 开发人员专注于业务开发（NoOps）
- 水平扩展性
- 统一开发、运维环境

应用部署环境发展



云环境开发特点

- 安全
- 服务（数据库, nosql, messaging）
- 本地资源（文件系统, 端口）
- Share nothing
- 数据备份
- Service orientate
- 移植

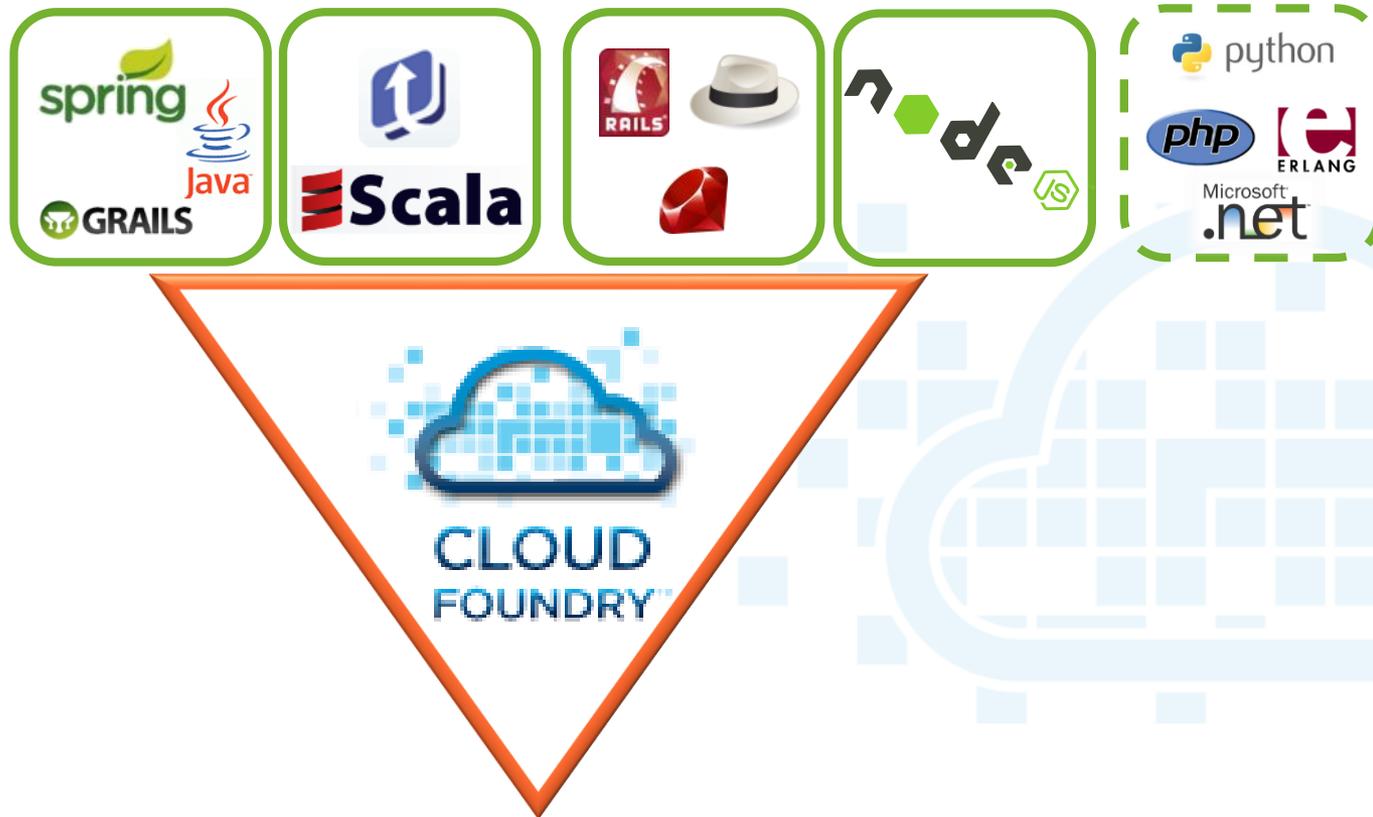
What is CloudFoundry?

CloudFoundry.com vs CloudFoundry OSS

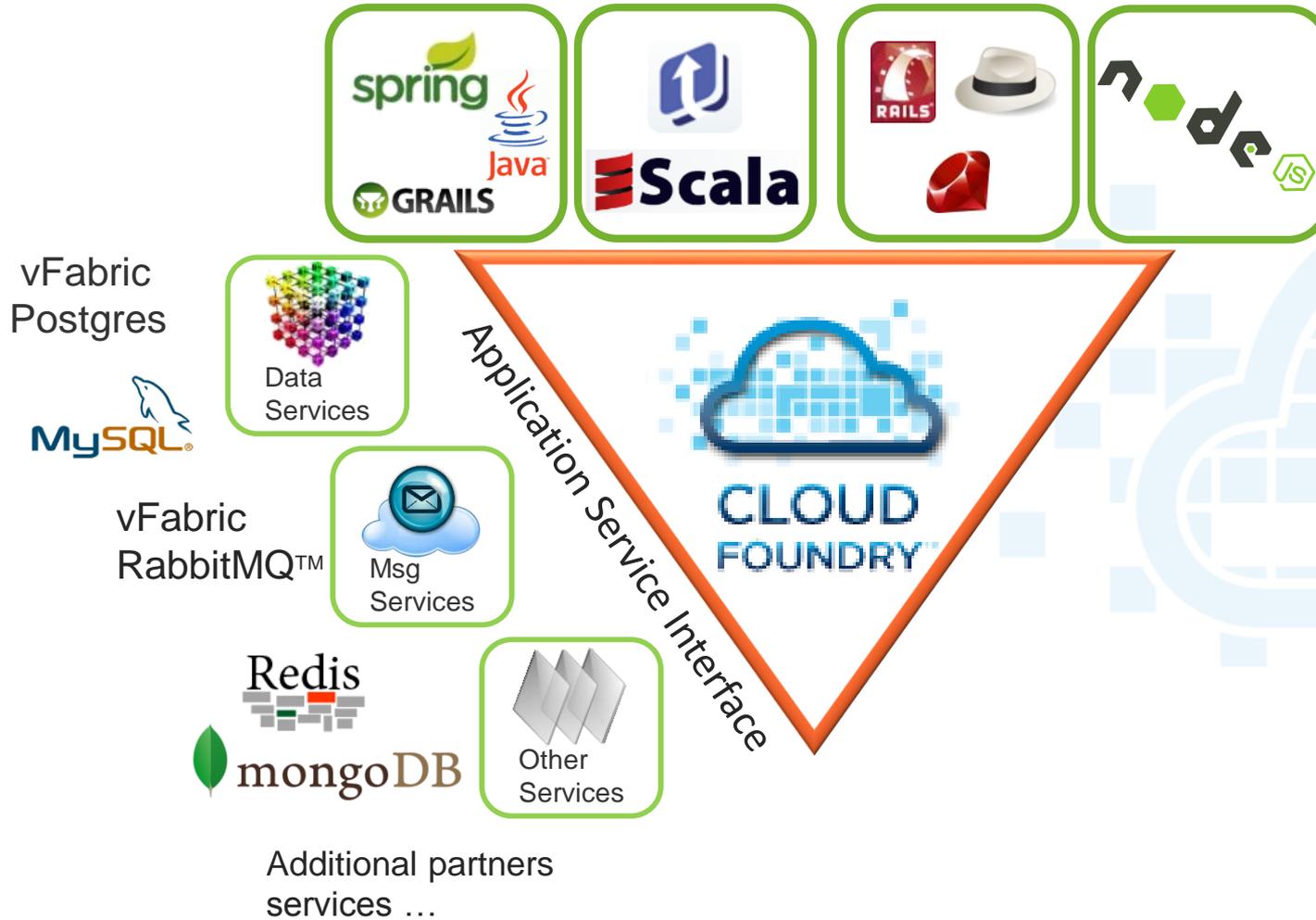
- CloudFoundry.com是Vmware公司运营的
 - 运行在vSphere上
 - 目前是公测版（年底推出正式版）
- CloudFoundry开源版是Vmware公司创立的
 - <http://github.com/cloudfoundry>
 - CloudFoundry.com, AppFog等公有云PaaS实例

Cloud Foundry 开放云应用平台 – 丰富编程框架

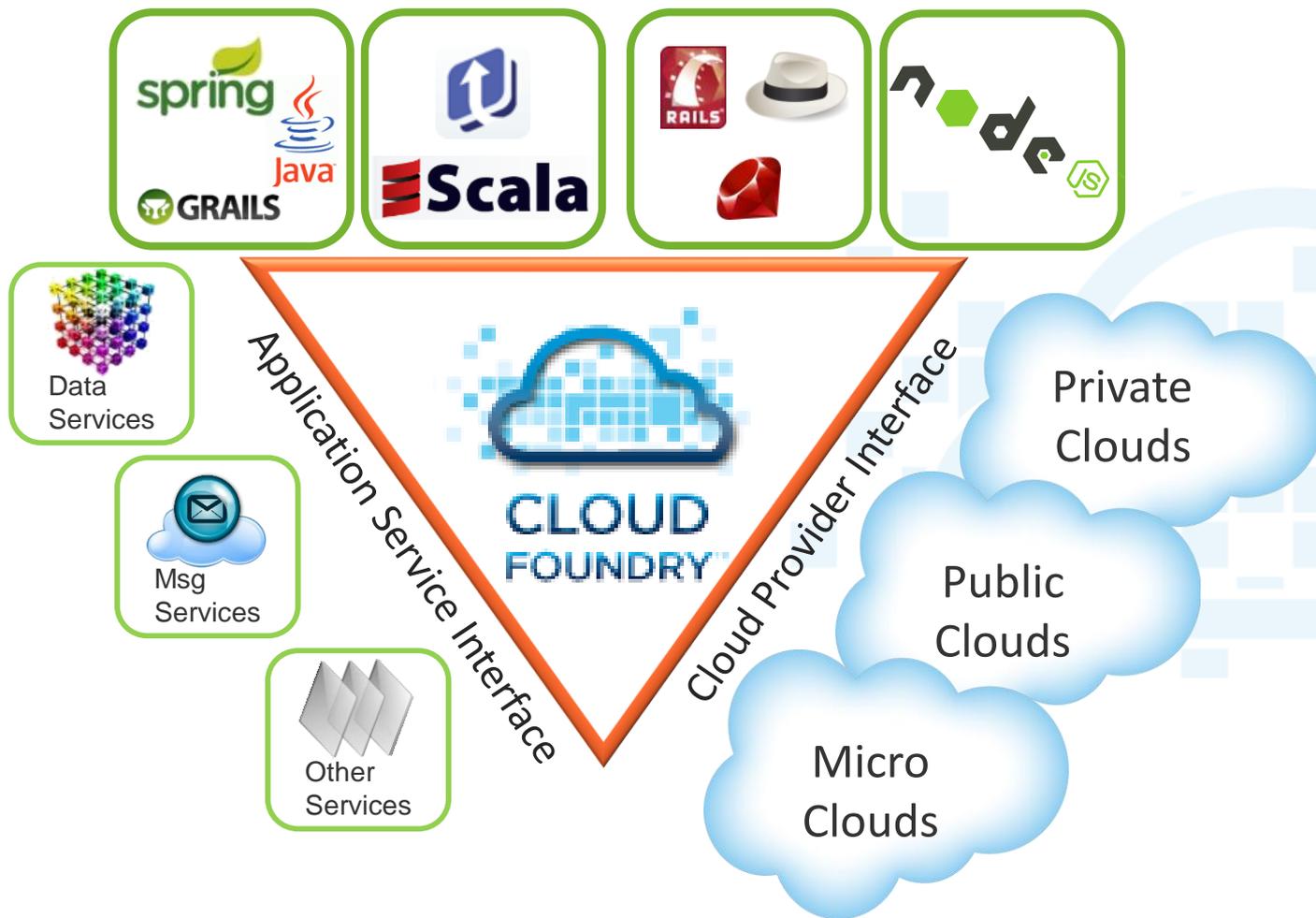
OSS community



Cloud Foundry 开放云应用平台 – 多样的服务



Cloud Foundry 开放云应用平台 – 多平台支持



避免
被锁定

Cloud Foundry的特点

■ 开源

- Github上开放源码，并鼓励社区贡献和check-in
- 目前所有核心模块都已经开源
- 开放的开发平台和流程，VMware R&D与社区开发者采用相同的代码审核和签入流程
- Apache 2开源许可协议

■ 开放

- 支持多种语言、框架、服务和底层IaaS基础平台
- 支持多种运维方式，公有，私有，CF.com

■ 自动化

- 应用和服务的自动化配置



应用平台支持: 不断增多

■ Java平台

- [Grails](#) 模仿Rails的Java平台实现
- Java_web 普通Java web程序
- [Lift](#) 基于Scale的web框架
- [Spring](#) 流行的Java框架

■ Ruby平台

- [Rack](#) 最小化的Ruby Web框架
- [Rails3](#) 一站式的Ruby Web框架
- [Sinatra](#) 极简主义的Ruby Web框架

■ Python平台

- [Django](#) 最流行的PythonWeb框架
- Wsgi Python的CGI

■ 其他平台

- Static page 静态页面
- [node.js](#) 异步Web框架
- [Erlang](#)
- [php](#)
- standalone 独立的程序

服务平台支持：不断增多

■ 主流服务

- MongoDB 最流行的Nosql数据库
- Mysql传统开源关系数据库
- Neo4j图数据库
- Postgresql Mysql的有力竞争者
- Redis极快的内存KV数据库

■ 存储

- Atmos EMC专业存储
- FileSystem 远程NFS支持
- Vblob 提供Amazon S3支持

■ 消息

- RabbitMQ 出色的Erlang队列系统

■ 大数据

- Hadoop集成

如何在私有云环境中安装Cloud Foundry

▪ dev_setup

- Ubuntu 10.04
- 适合小规模试验环境
- 每台机器需要手工安装和修改配置文件

▪ Github上 :

https://github.com/cloudfoundry/oss-docs/tree/master/vcap/single_and_multi_node_deployments_with_dev_setup

▪ 博客 :

<http://qing.weibo.com/2294942122/88ca09aa330004r8.html> 和
<http://qing.weibo.com/2294942122/88ca09aa33000975.html>

▪ BOSH

- 大规模的部署
- 集中式的配置文件
- 支持自动化扩容
- 对各类IaaS平台集成

▪ Git上的文档+源码

<https://github.com/cloudfoundry/oss-docs/blob/master/bosh/documentation/documentation.md>

▪ <https://github.com/cloudfoundry/oss-docs/tree/master/bosh>

▪ 博客文章

▪ <http://www.brianmmclain.com/> (必读)

▪ <http://drnicwilliams.com/2012/04/16/creating-a-bosh-from-scratch-on-aws/>

通过BOSH扩展Cloud Foundry对IaaS的支持



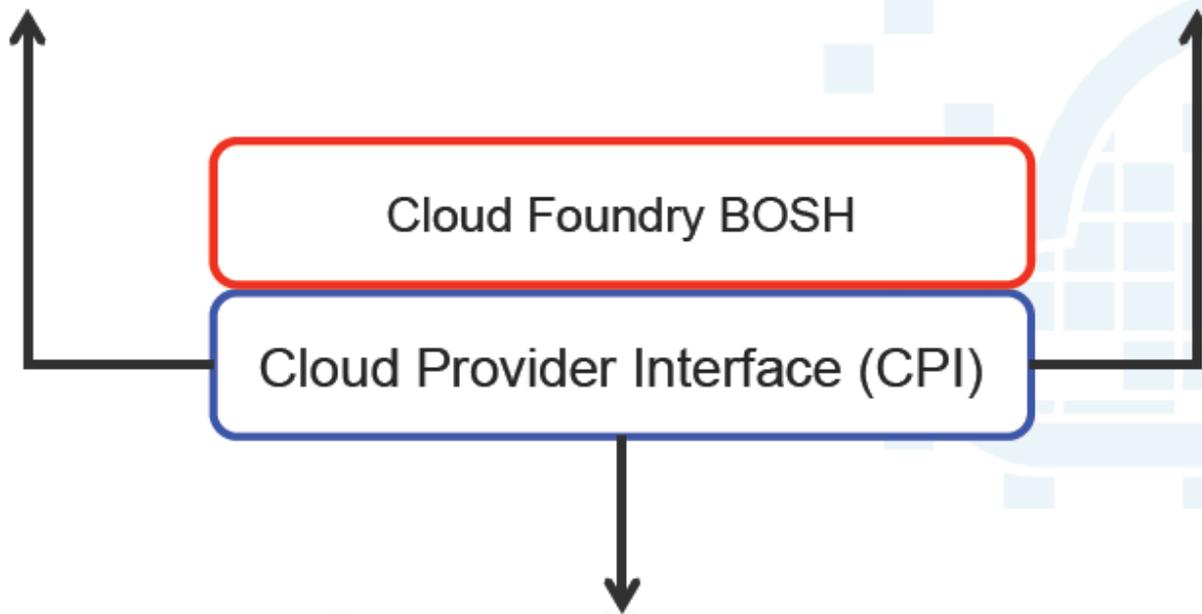
vSphere: battle tested implementation,
thousands of deployments

vCloud Director: *“work in progress”*, 2H 2012



CPI: code complete

functional status: *“work in progress”*



contribute: github.com/cloudfoundry/bosh

LINUX OF THE CLOUD

cloud foundry os layering

vmc shell:

- scriptable CLI
- thin UI layer above REST API
- STS plugin for integration into Eclipse, Cloud9 IDE, etc.

vcap core:

- REST control api, uniform syscall API across all cloud foundry instances
- highly portable application model, URL addressable apps
- service layer: uniform access to files, databases, kv stores, etc. uniform
- IaaS/HAL abstraction orchestrated by BOSH
- OSS with liberal licensing Apache2

aws

OpenStack

Azure

vSphere

google

eucalyptus

Hyper-V

cloudstack

multi-blade, multi-core, large memory x86, unlimited net, storage, dc

cloud foundry syscall

REST control API:
app management
service management
user/system management

app management:

- http url addressable apps
- simple horizontal scale out
- REST api
- vmc push; vmc update
- vmc stop; vmc start
- vmc mem; vmc map
- vmc instances
- vmc {files, logs, crashlogs}

OS
syscall

service management:

- next-gen "I/O" system
- unified discovery/binding
- REST api
- vmc services
- vmc create-service
- vmc delete-service
- vmc bind-service
- vmc unbind-service

Hardware Abstraction Layer

vmc命令

Create app, update app, control app

```
vmc push [appname] [--path] [--url] [--instances N] [--mem] [--no-start]
vmc update <appname> [--path PATH]
vmc stop <appname>
vmc start <appname>
vmc target [url]
```

Update app settings, get app information

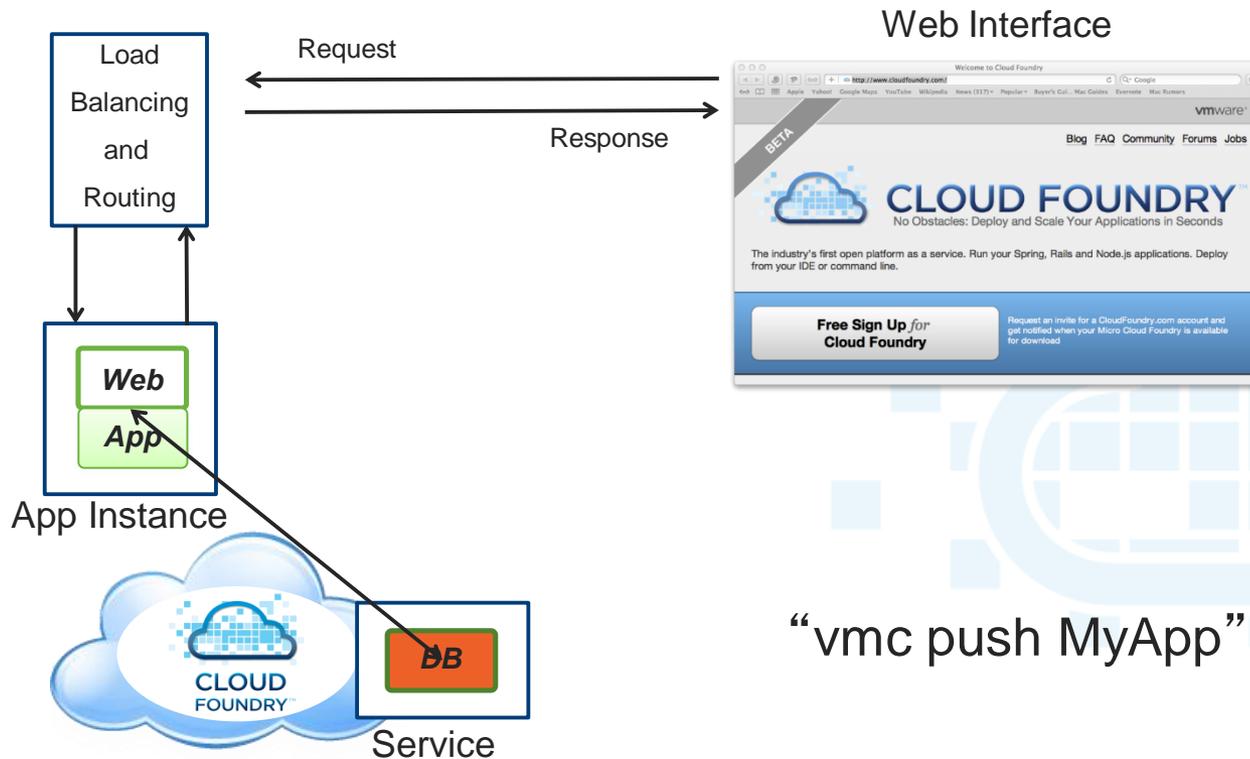
```
vmc mem <appname> [memsize]
vmc map <appname> <url>
vmc instances <appname> <num | delta>
vmc {crashes, crashlogs, logs} <appname>
vmc files <appname> [path]
```

Deal with services, users, and information

```
vmc create-service <service> [--name servicename] [--bind appname]
vmc bind-service <servicename> <appname>
vmc unbind-service <servicename> <appname>
vmc delete-service <servicename>
```

```
vmc user, vmc passwd, vmc login, vmc logout, vmc add-user
vmc services, vmc apps, vmc info
```

How Apps are Accessed on Cloud Foundry

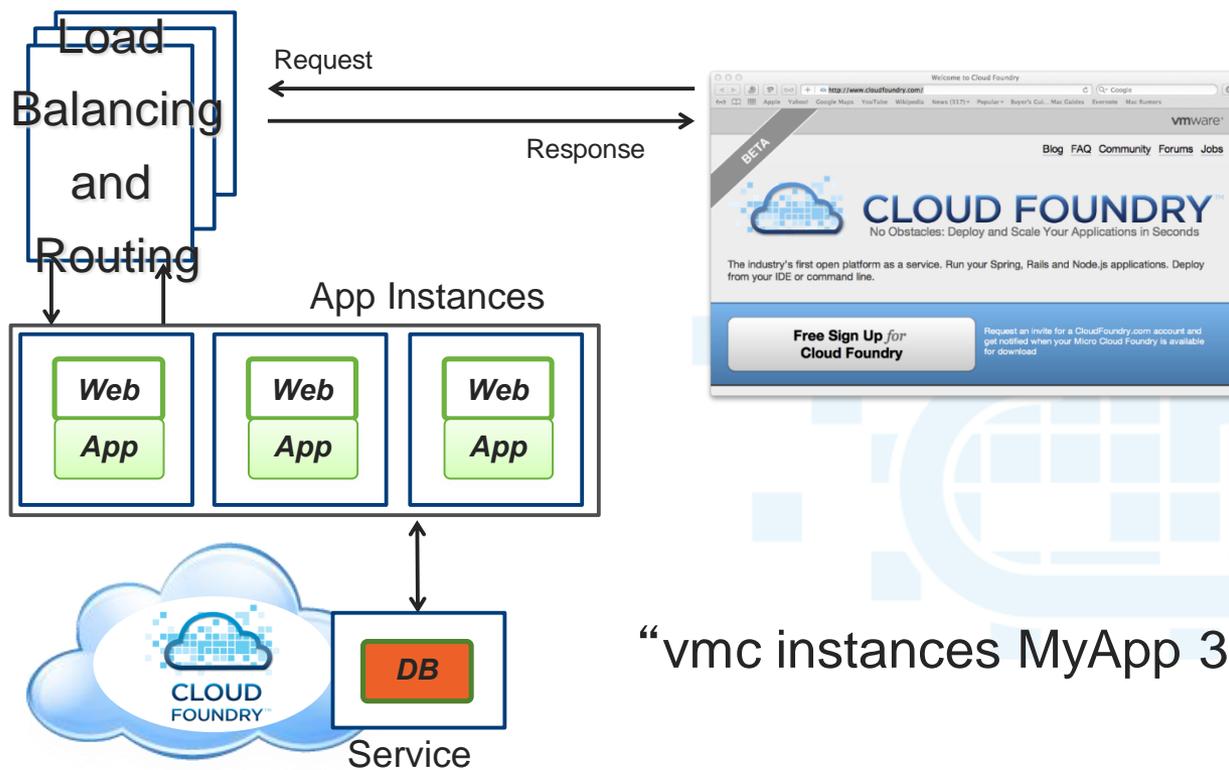


“vmc push MyApp”

Access Cloud Foundry Chinese Blog for more details

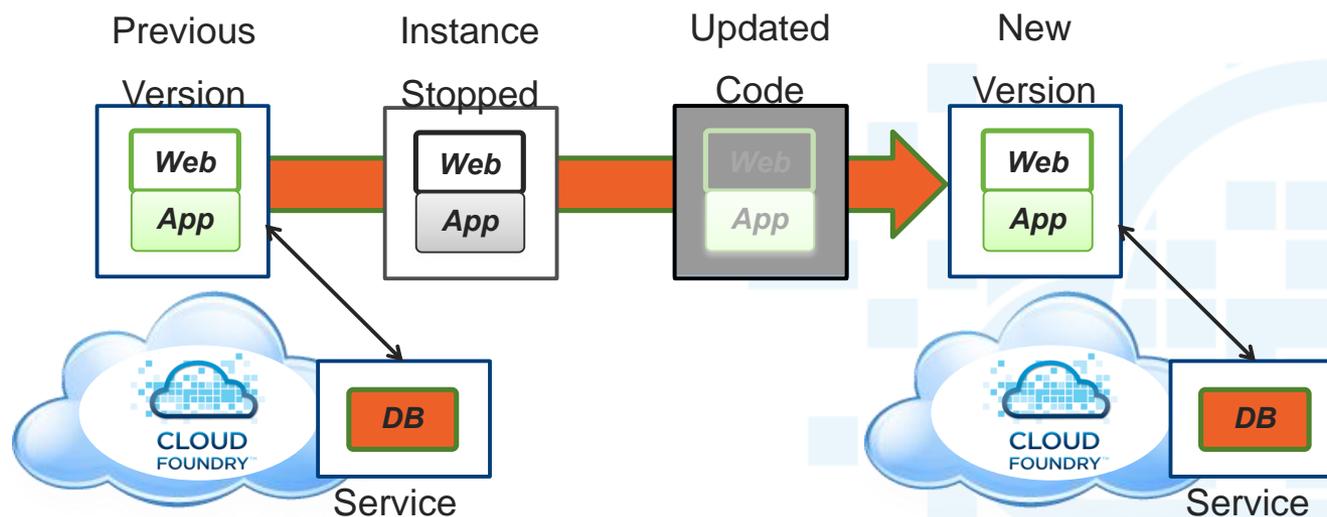
<http://cnblog.cloudfoundry.com>

How Apps are Scaled on Cloud Foundry



“vmc instances MyApp 3”

How Apps are Updated on Cloud Foundry



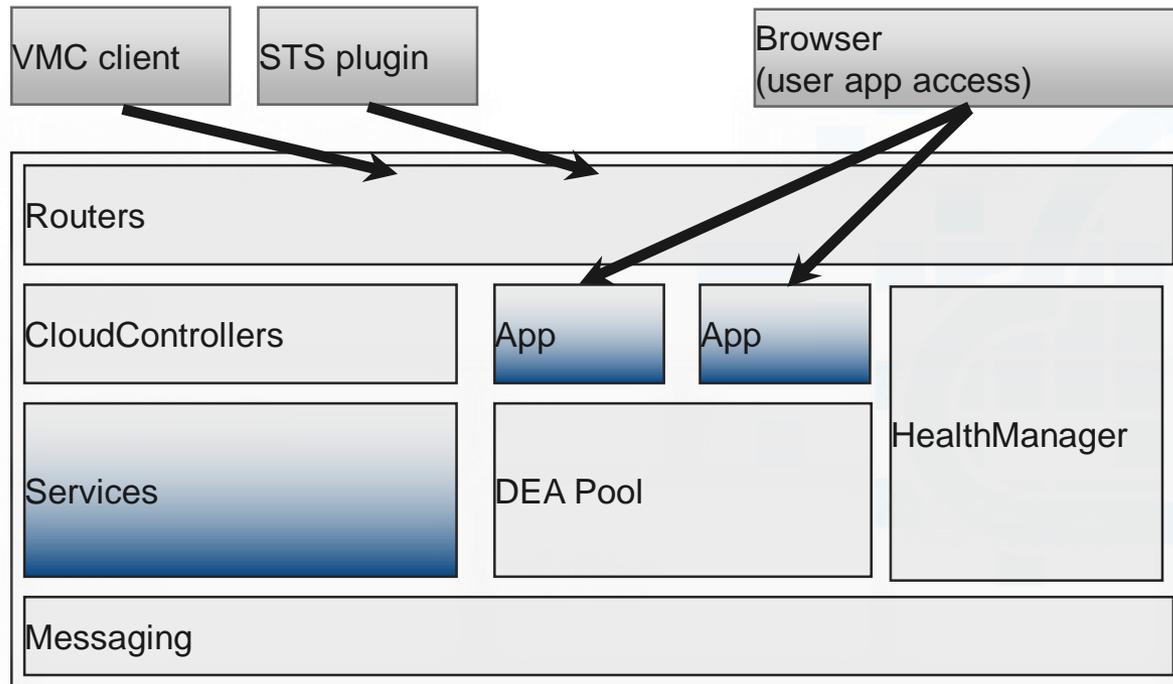
“vmc update MyApp”

CloudFoundry架构理念

- 越核心的代码应该越简洁
- 为MTTR优化，而不是MTBF
- 消息驱动，组件松耦合
- 组件间动态发现，无依赖性



Cloud Foundry Logical View



Cloud Foundry中文资源

- 网站: <http://www.cloudfoundry.cn/>
 - 博客: <http://cnblog.cloudfoundry.com/>
 - 社区: <http://cloudfoundry.csdn.net/>
 - 微博: @cloudfoundry
 - 视频: <http://u.youku.com/CloudFoundryCN>
- 

Q&A

**更多内容，PPT下载
请关注官方微博**

@CloudFoundry