



From No Ops to NoOps

--新浪微博稳定性工程团队运维实践

首先，你得发自内心的愿望伟大



我们是谁？

- DevOps @Sina
- Lamp项目托管平台
- 已托管项目500+
- 日均请求10B+

我们是谁？

- 每年新增PHP项目100+
- 3000+台前端服务器 100亿+ Hits/天
- 45000+行虚拟主机配置 30G代码
(GZip后)
- 3000+台数据库服务器 200亿+ 数据
库请求/天

运维演变进程

- 平台化阶段
- 运维工具发展阶段
- 自动化运维萌芽阶段
- 自动化运维发展阶段

平合化阶段

- 建立统一的标准与规范
- 简化操作与维护
- 统一监控与问题修复
- 抽象服务器拓扑与部署情况



Users



Users



Users



Project A



Project B



Project C



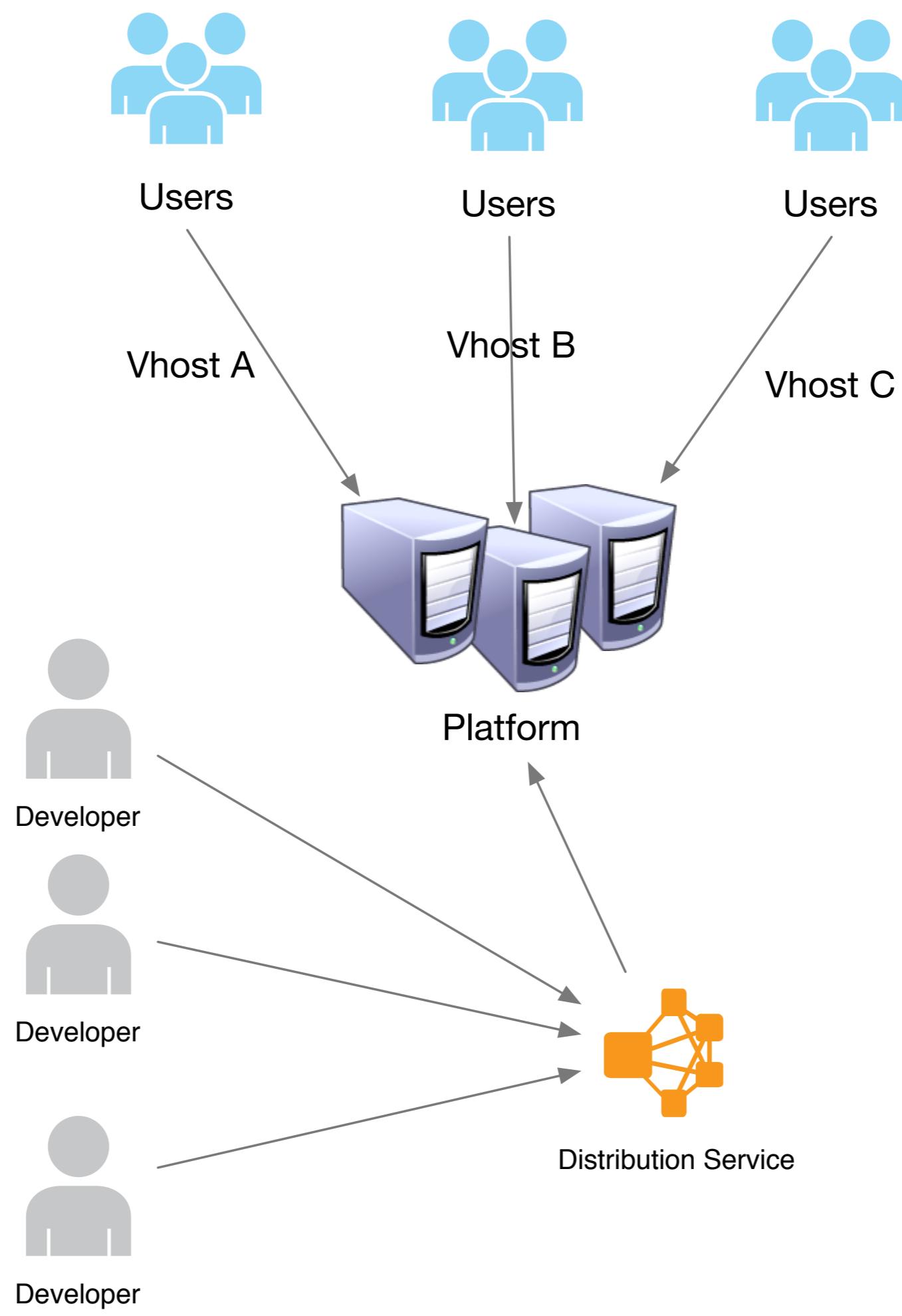
Developer



Developer

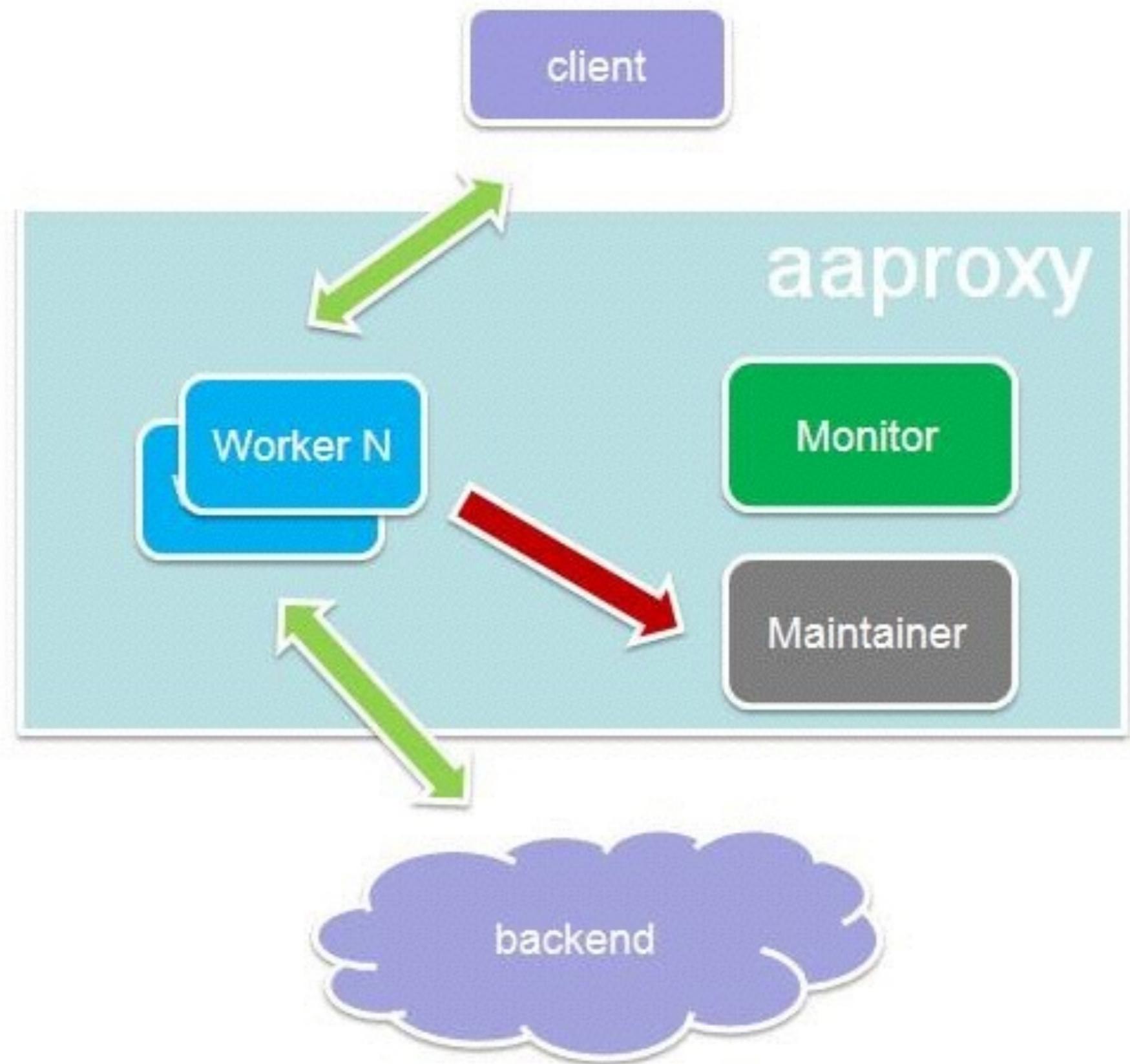


Developer



平合化阶段问题

- HTTP调用也能引发灾难
- 服务互相影响恶化小项目故障
- 曾尝试使用限制每虚拟主机的进程数来解决问题



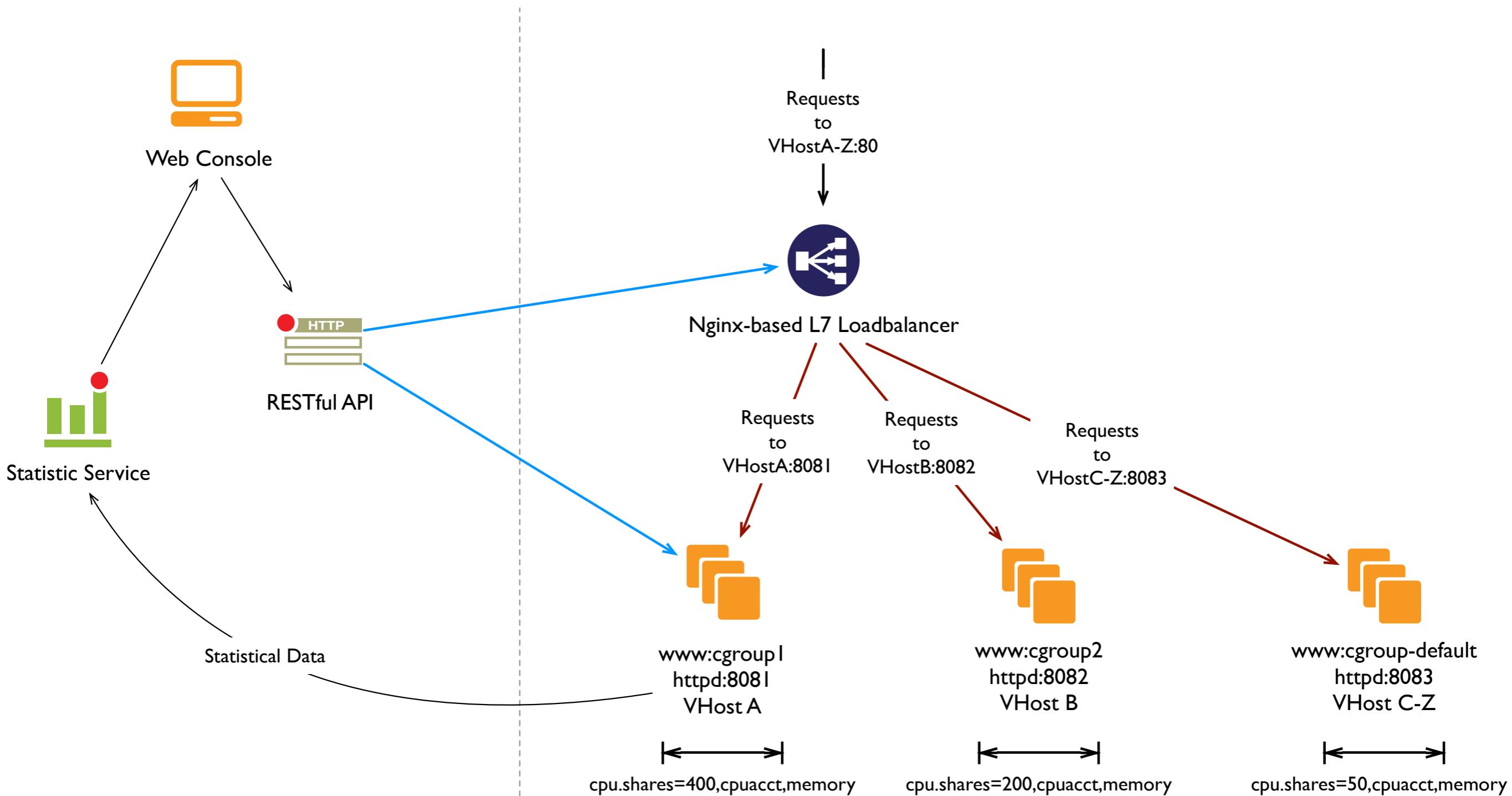
通用TCP透明代理

AAProxy

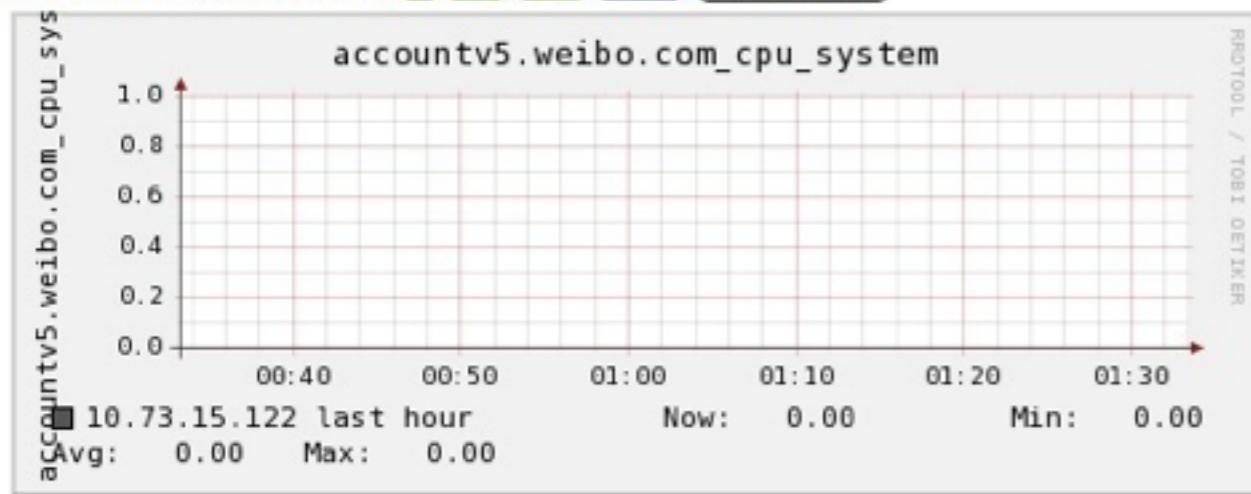
- 目前只工作于HTTP协议
- 类滑动窗口流量控制
- ToDo:
 - DNS异步解析
 - 对非HTTP协议支持
- 热点内容Cache与性能优化

应用 *CGroup*

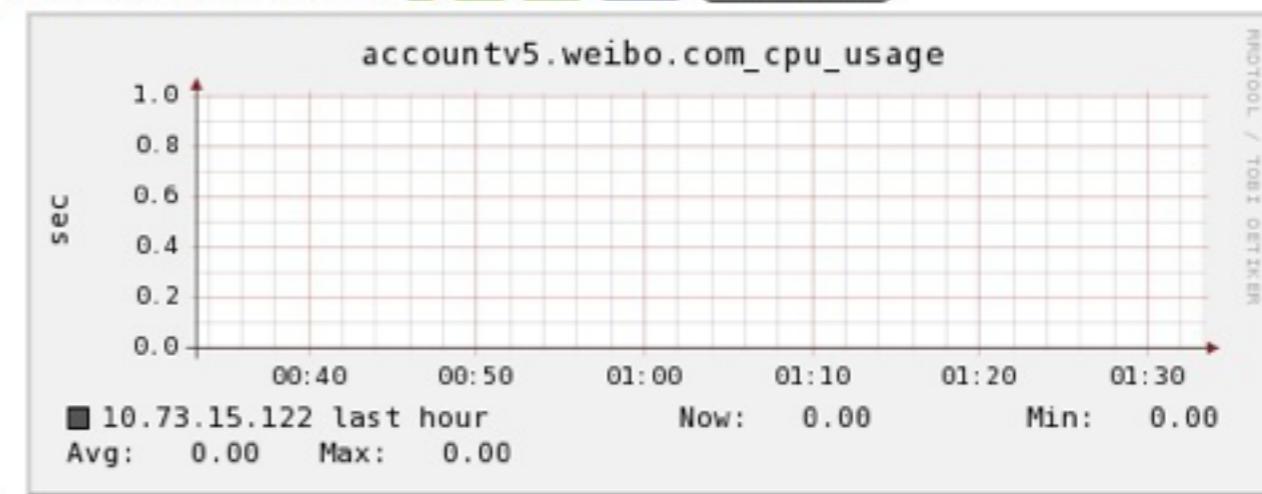
- 解决单服务器资源抢占的“明星”解决方案
- 是否需要同时引入LXC?
- 无CPU抢占时?
- 一并解决各应用资源使用统计问题



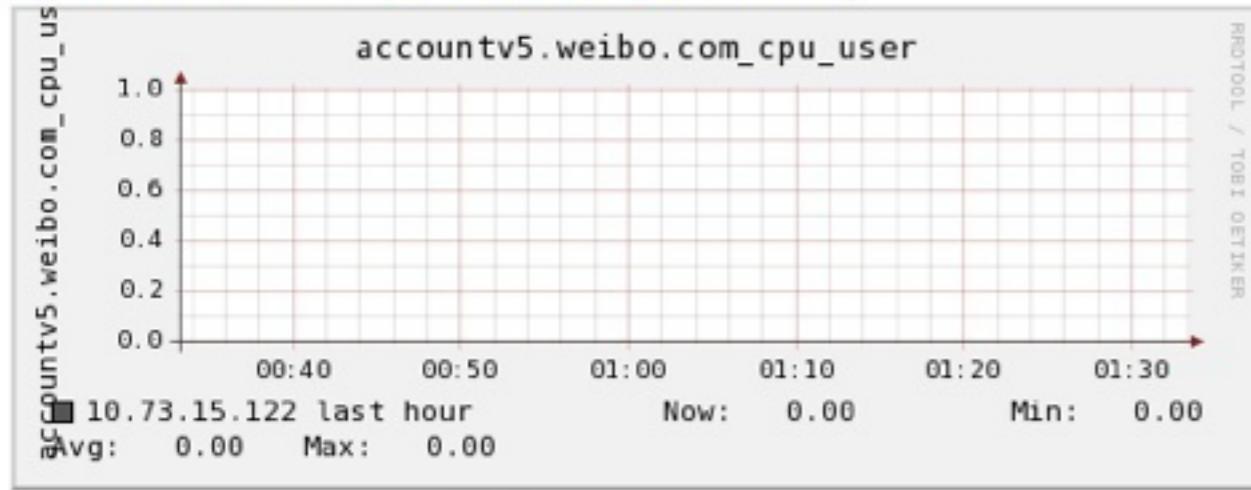
accountv5.weibo.com_cpu_system [+](#) [CSV](#) [JSON](#) [Inspect](#) [Hide/Show Events](#)



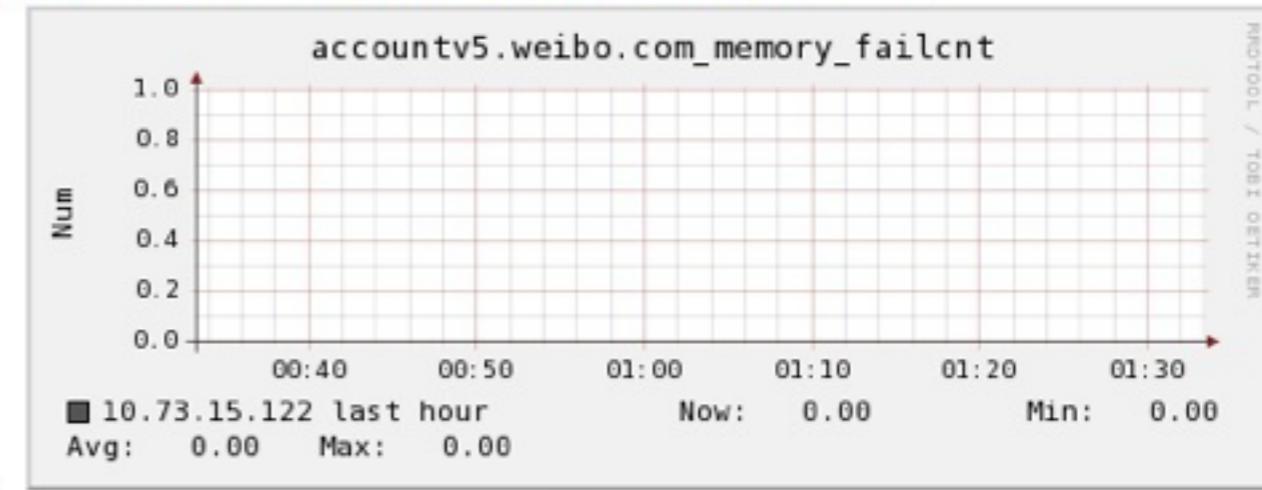
accountv5.weibo.com_cpu_usage [+](#) [CSV](#) [JSON](#) [Inspect](#) [Hide/Show Events](#)



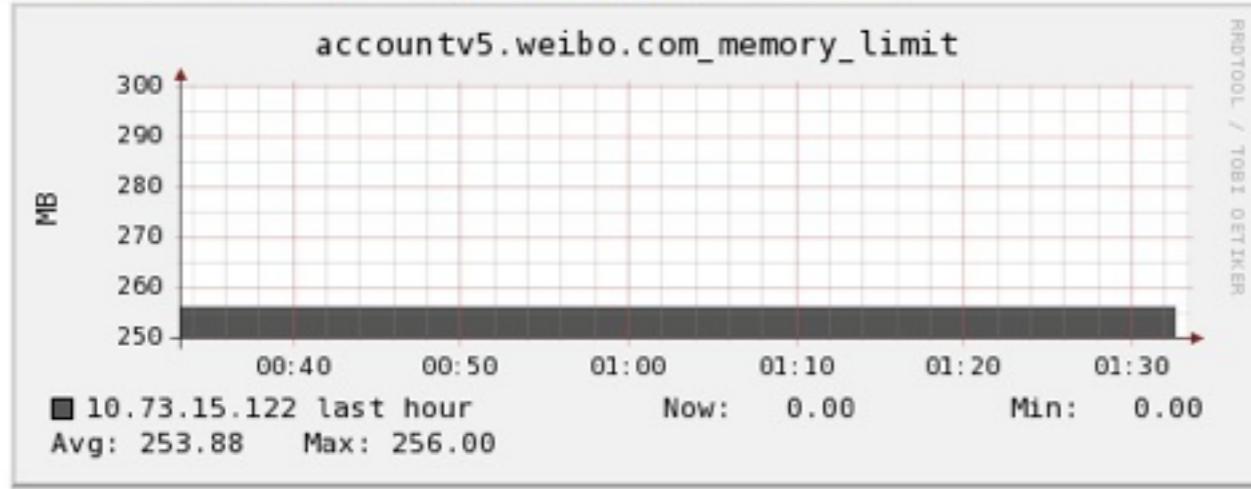
accountv5.weibo.com_cpu_user [+](#) [CSV](#) [JSON](#) [Inspect](#) [Hide/Show Events](#)



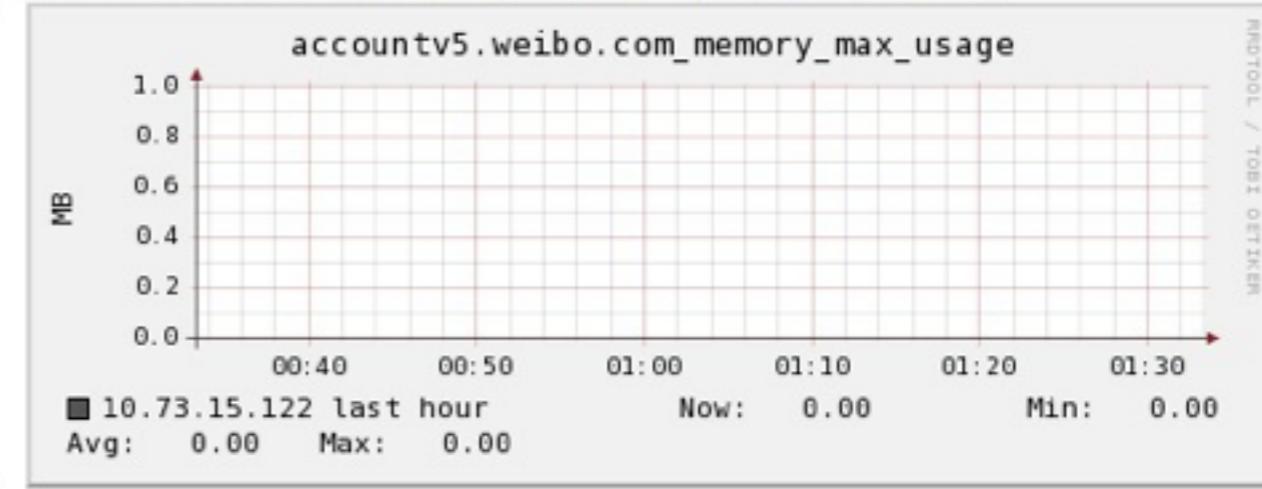
accountv5.weibo.com_memory_failcnt [+](#) [CSV](#) [JSON](#) [Inspect](#) [Hide/Show Events](#)



accountv5.weibo.com_memory_limit [+](#) [CSV](#) [JSON](#) [Inspect](#) [Hide/Show Events](#)



accountv5.weibo.com_memory_max_usage [+](#) [CSV](#) [JSON](#) [Inspect](#) [Hide/Show Events](#)



accountv5.weibo.com_memory_usage [+](#) [CSV](#) [JSON](#) [Inspect](#) [Hide/Show Events](#)

accountv5.weibo.com_tasks [+](#) [CSV](#) [JSON](#) [Inspect](#) [Hide/Show Events](#)

运维工具发展阶段

- 为服务器按照功能划分为多个角色
- 每个角色包含多个模块，每个模块维护统一的配置文件
- 根据之前经验，差异由系统维护最为合理
- CFEengine管理一切
- 差异配置怎么办？

httpd-vhost.conf_tmpl

```
...
SetEnv SINASRV_ZONE_IDC @@XD@@
SetEnv SINASRV_ZONE_ISP @@CNC@@
SetEnv SINASRV_ZONE_ID @@010201@@
```



FE Node

```
/etc/httpd-vhost.conf
```

vhost_config.pl

```
...
%idc_rule = (
    zjm => {
        SINASRV_ZONE_IDC => "ZJM",
        SINASRV_ZONE_ISP => "CNC",
        SINASRV_ZONE_ID => "010101",
    }
)
...
elsif ($line =~ m!^ \s* SetEnv \s+ SINASRV_ZONE_IDC.+!) {
    push @vhost_configfile, "SetEnv SINASRV_ZONE_IDC ". $idc_rule{$idc}{SINASRV_ZONE_ID};
    next;
} elsif ($line =~ m!^ \s* SetEnv \s+ SINASRV_ZONE_ISP.+!) {
    push @vhost_configfile, "SetEnv SINASRV_ZONE_ISP ". $idc_rule{$idc}{SINASRV_ZONE_ISP};
    next;
} elsif ($line =~ m!^ \s* SetEnv \s+ SINASRV_ZONE_ID.+!) {
    push @vhost_configfile, "SetEnv SINASRV_ZONE_ID ". $idc_rule{$idc}{SINASRV_ZONE_ID};
    next;
}
...
```

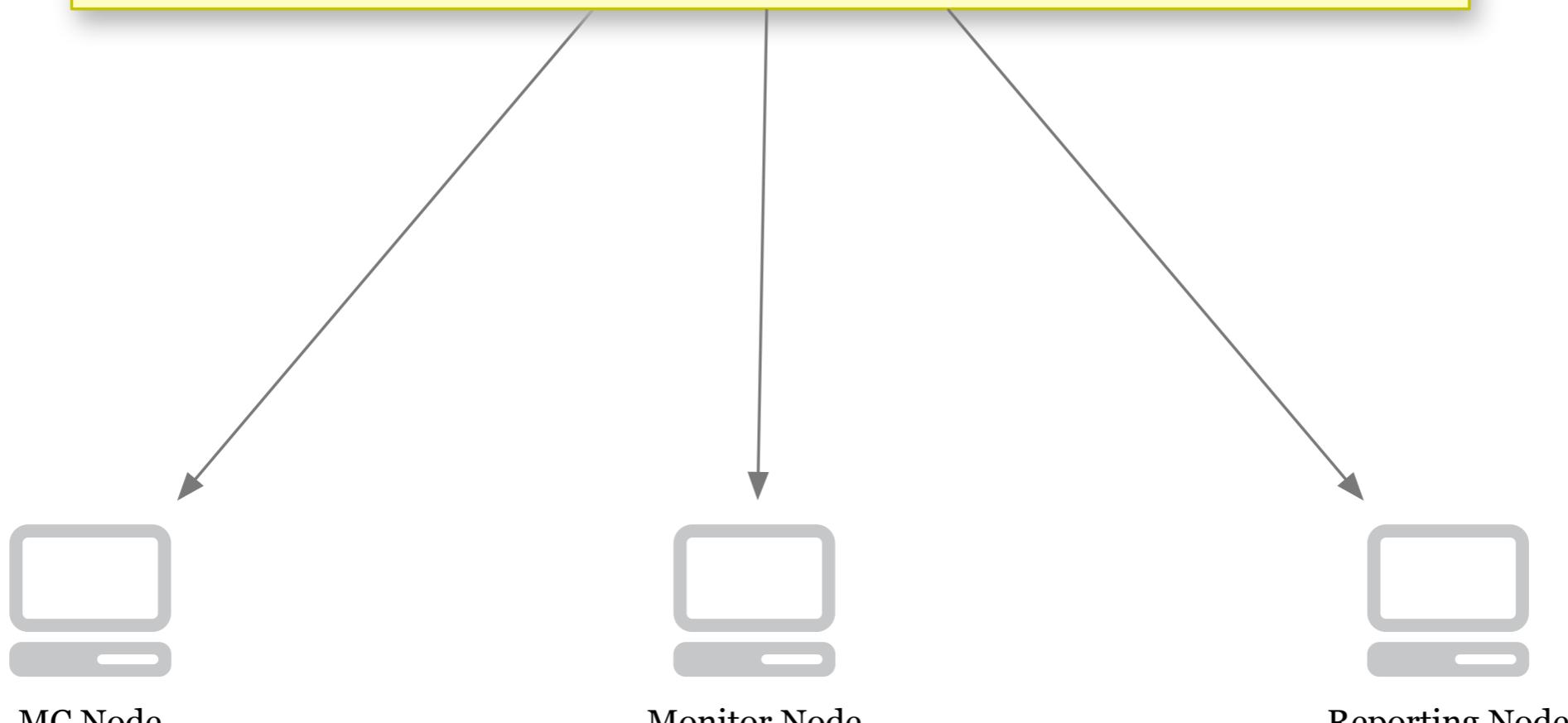
没有配置文件的服务怎么办？

- 那就创造一个配置文件吧～
- 当然要创造一个全局的～
- 索性与监控程序结合起来怎么样？
- INI VS YAML
- 就比如Memcached!

mc.conf

```
[MBlog_Userinfo]
ip=10.55.22.100:10000 10.55.22.101:10000 10.55.22.102:10000 10.55.22.103:10000
memsize=10.55.22.100:1G 10.55.22.101:1G 10.55.22.102:1G 10.55.22.103:1G
conn_limit=10.55.22.100:1W 10.55.22.101:1W 10.55.22.102:1W 10.55.22.103:1W

[MBlog_Counter]
ip=10.55.22.100:10001 10.55.22.101:10001 10.55.22.102:10001 10.55.22.103:10001
memsize=10.55.22.100:1G 10.55.22.101:1G 10.55.22.102:1G 10.55.22.103:1G
conn_limit=10.55.22.100:1W 10.55.22.101:1W 10.55.22.102:1W 10.55.22.103:1W
```



```
/etc/init.d/memcached
/etc/cron.d/check_mc
```

```
/etc/monitor.d/check_mc.py
```

```
/etc/reporting.d/get_mc_stats.py
```

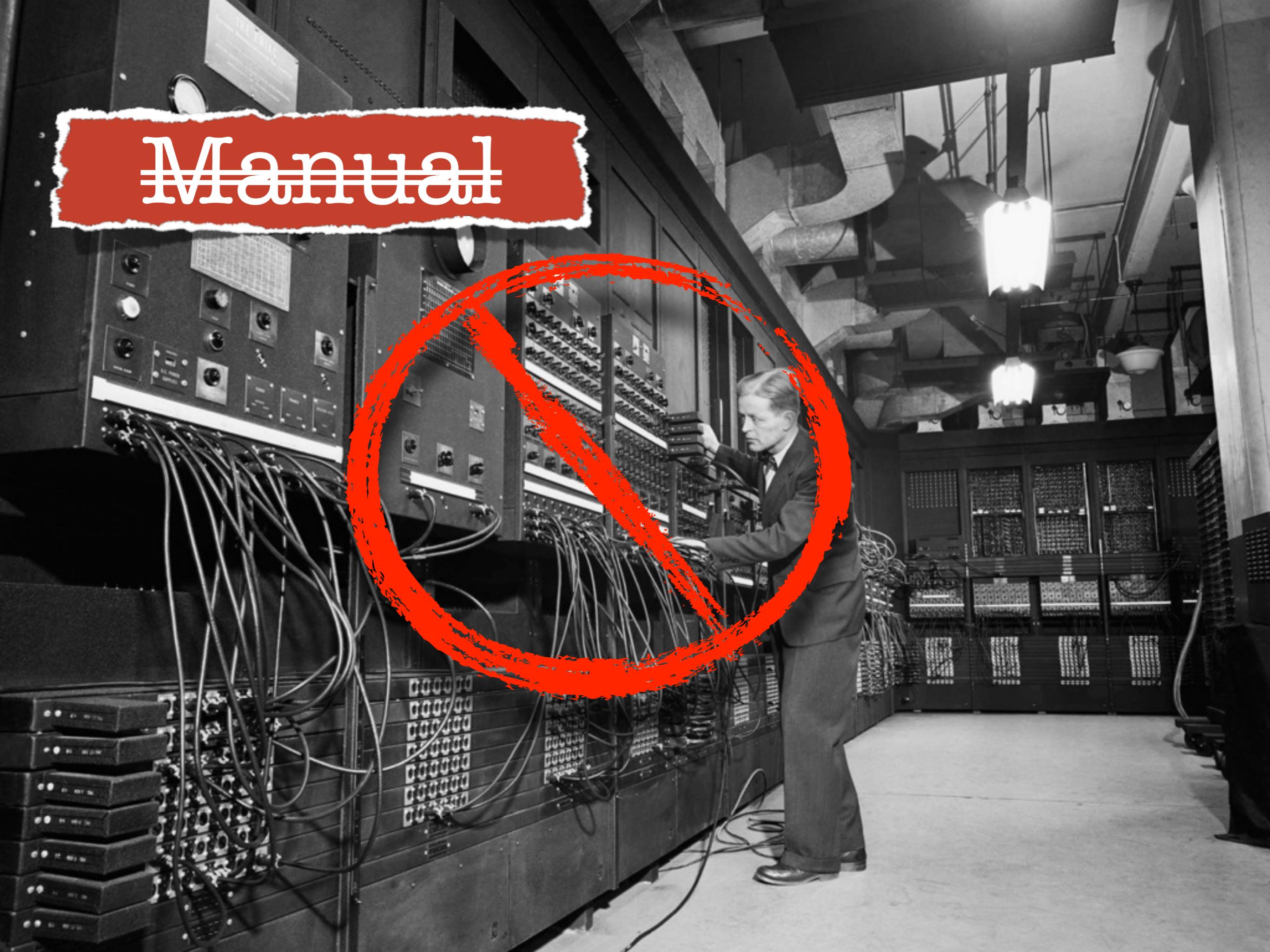
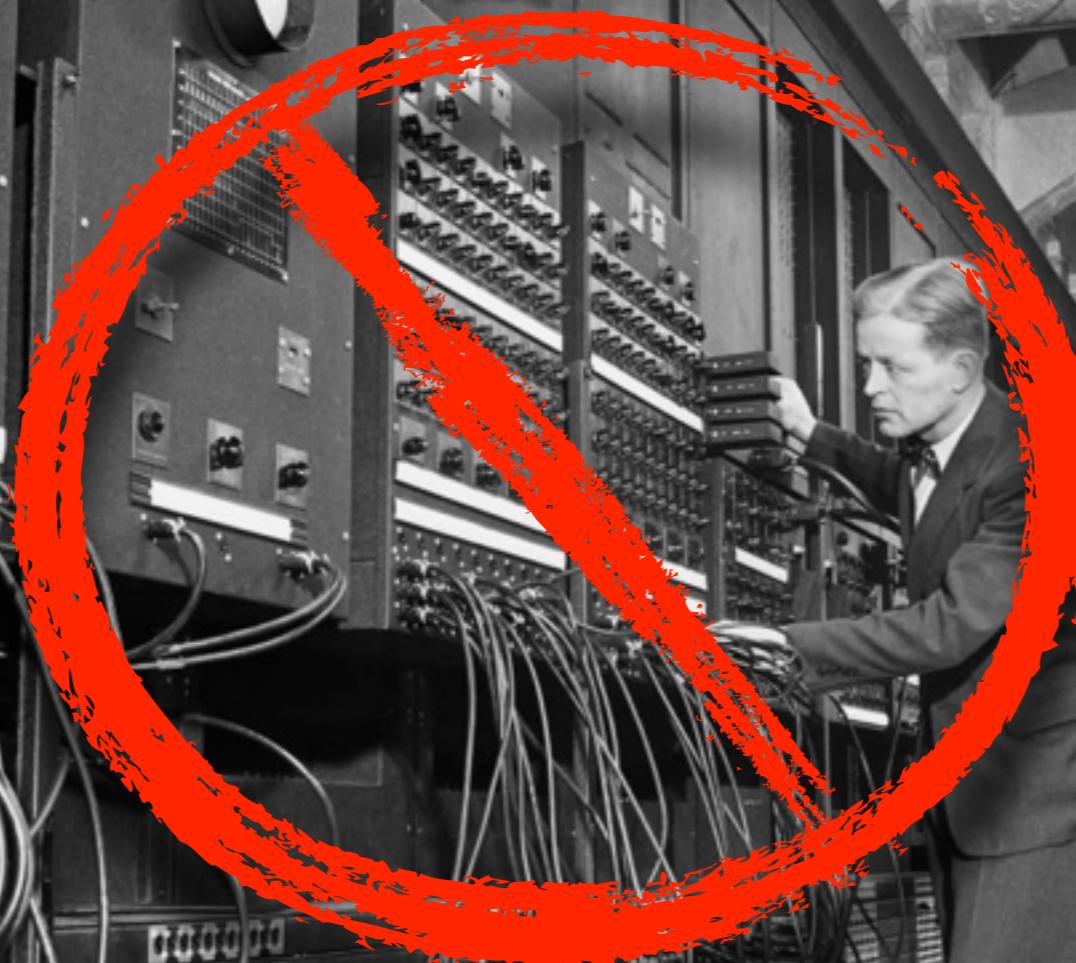
OPERATOR!

Make
Disasters

Western Electric
MITSUBISHI
DENKI

PRODUCED BY
AUDIO PRODUCTIONS, INC.

Manual



线上变更管理

■ 工作流管理

■ 工作流日志

■ 工作流执行日志

■ 配置文件管理

■ 包部署管理

■ WebShell

■ 用户临时授权

运维信息管理

■ 产品单元管理

■ 环境设置

■ 模块设置

■ IDC设置

■ 负载均衡

监控报警管理

■ 性能监控

■ 报警管理

■ 基础监控模板设置

资源资产管理

■ 节点管理

■ DB资产管理

数据库应用

■ 端口管理

■ 域名管理

■ 备份管理

实验特性专区

■ Cron管理

系统管理专区

■ 管理日志

■ 账户设置

■ 修改资料

■ 注销登录

和最新线上版本的配置差异

```
--- /var/tmp/httpd-vhost.conf_dpool2.old.tmp      2012-05-24 18:00:02.000000000 +0800
+++ /var/tmp/httpd-vhost.conf_dpool2.new.tmp      2012-05-24 18:00:02.000000000 +0800
@@ -205,6 +205,7 @@
          SetEnv SINASRV_NDATA_CACHE_URL "http://cache.@@mars@@.weibo.com/nd/photoweibo/"

          SetEnv SINASRV_MEMCACHED_ALBUMS_SERVERS "10.75.31.39:7122 10.75.31.38:7122 10.75.31.37:7122 10.75.31.36:7122 10.77.7.85:7122 10.77.7.86:7122
+
          SetEnv SINASRV_MEMCACHED_NEW_ALBUMS_SERVERS "10.75.31.40:7122 10.75.31.39:7122 10.75.31.38:7122 10.75.31.37:7122 10.75.31.36:7122 10.77.7.85:7122 10.77.7.86:7122
```

线上变更管理

- 工作流管理
- 工作流日志
- 工作流执行日志
- 配置文件管理
- 包部署管理
- WebShell
- 用户临时授权

运维信息管理

- 产品单元管理
- 环境设置
- 模块设置
- IDC设置
- 负载均衡

监控报警管理

- 性能监控
- 报警管理
- 基础监控模板设置

资源资产管理

- 节点管理
- DB资产管理

数据库应用

- 端口管理

- 域名管理
- 备份管理

实验特性专区

- Cron管理

系统管理专区

- 管理日志
- 账户设置
- 修改资料
- 注销登录

» 节点管理(默认只显示30条数据, 请善用"查找"功能)

环境 : 所有环境 模块 : 数据库从库 IDC : 北京土城 关键字 : IP 查找 共166条结果

状态及操作	内网IP	外网IP	环境	初始化状态	模块			
					模块操作	模块名	端口	产品单元
					[初始化]	Fusion_io(fusion_io)		-----
					[初始化]	硬件RAID(hw RAID)		-----
<input type="checkbox"/> 操作	10.73.11.119	10.72.11.119	线上	✓	查看端口 待监测 扩容 [初始化]	MytriggerQ MySQL 从库(mysql-innodb)	3618	数据库平台 /微博客/ 基础服务/ myfeed
					查看端口 待监测 扩容 [初始化]	数据库从库(mysql-innodb)	3618	数据库平台 /微博客/ 基础服务/ myfeed
					查看端口 待监测 扩容 [撤销初始化状态]	mytriggerMySQL从库(mysql)	3618	数据库平台 /微博客/ 基础服务/ myfeed
					查看端口 待监测 扩容 [初始化]	主模块:数据库从库(mysql-innodb)	4624	数据库平台 /微博客/ 基础服务/ 关注
<input type="checkbox"/> 操作	10.73.25.86	10.72.25.86	线上	✓	查看端口 待监测 扩容 [初始化]	SSD群集(ssd)		-----
					查看端口 待监测 扩容 [初始化]	硬件RAID(hw RAID)		-----
					查看端口 待监测	主模块:数据库从库(mysql-innodb)	462	数据库平台 /微博客/

隐藏左栏

足够自动化了吗？

- 运维工程师与开发工程师思路的区别
- 运维是一个体系
- 由许多相关联的部分组成
- 系统扩展性不好
- 无法灵活的联系各个点

Framework



我们发现问题在出发点

- 自动化运维不是“另一个”系统
- 它是对你架构可运维性的更高要求
- 需要将可运维性作为软件和架构设计的主要考量因素之一
- 运维工具也是软件和架构的一部分

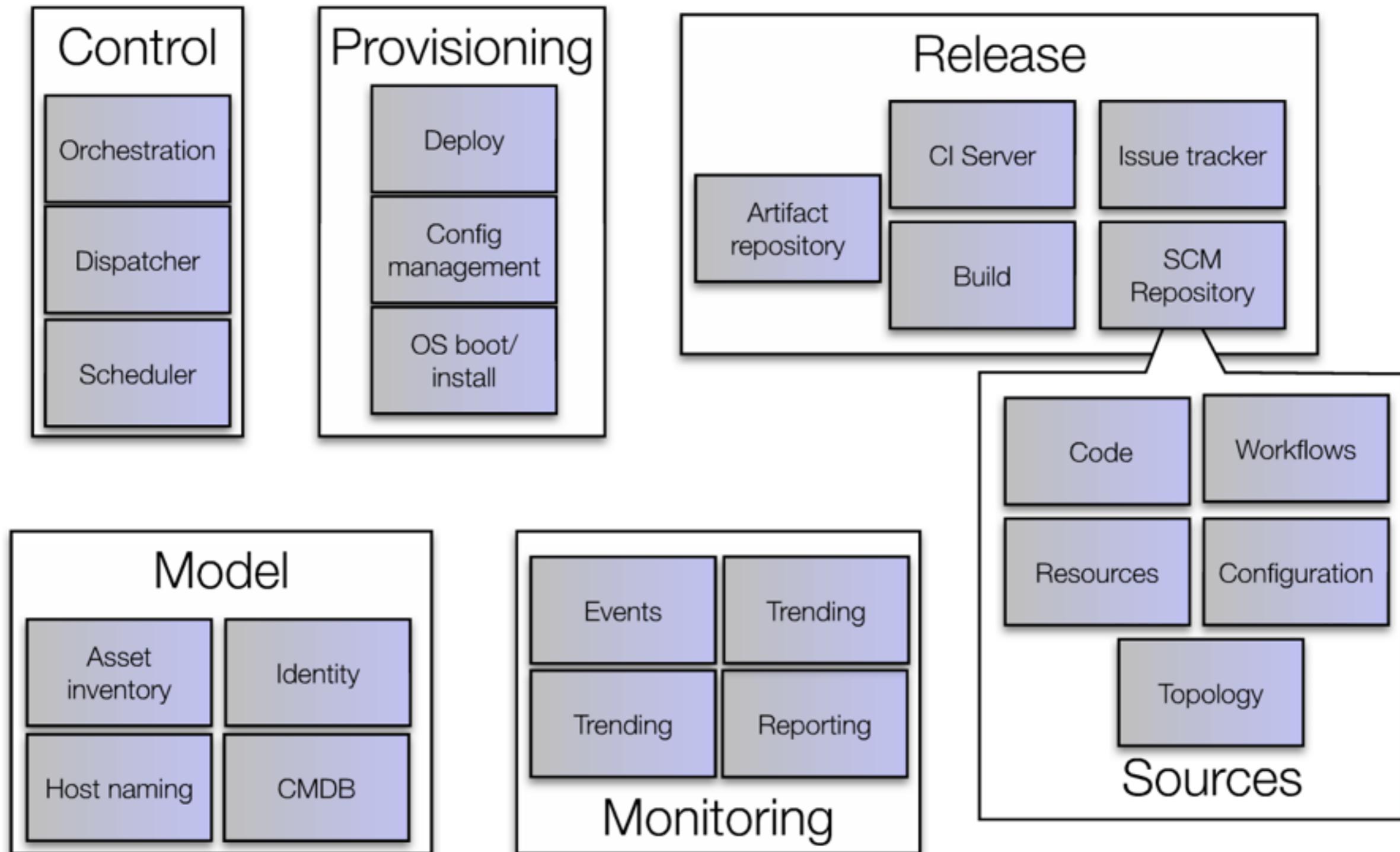
it's

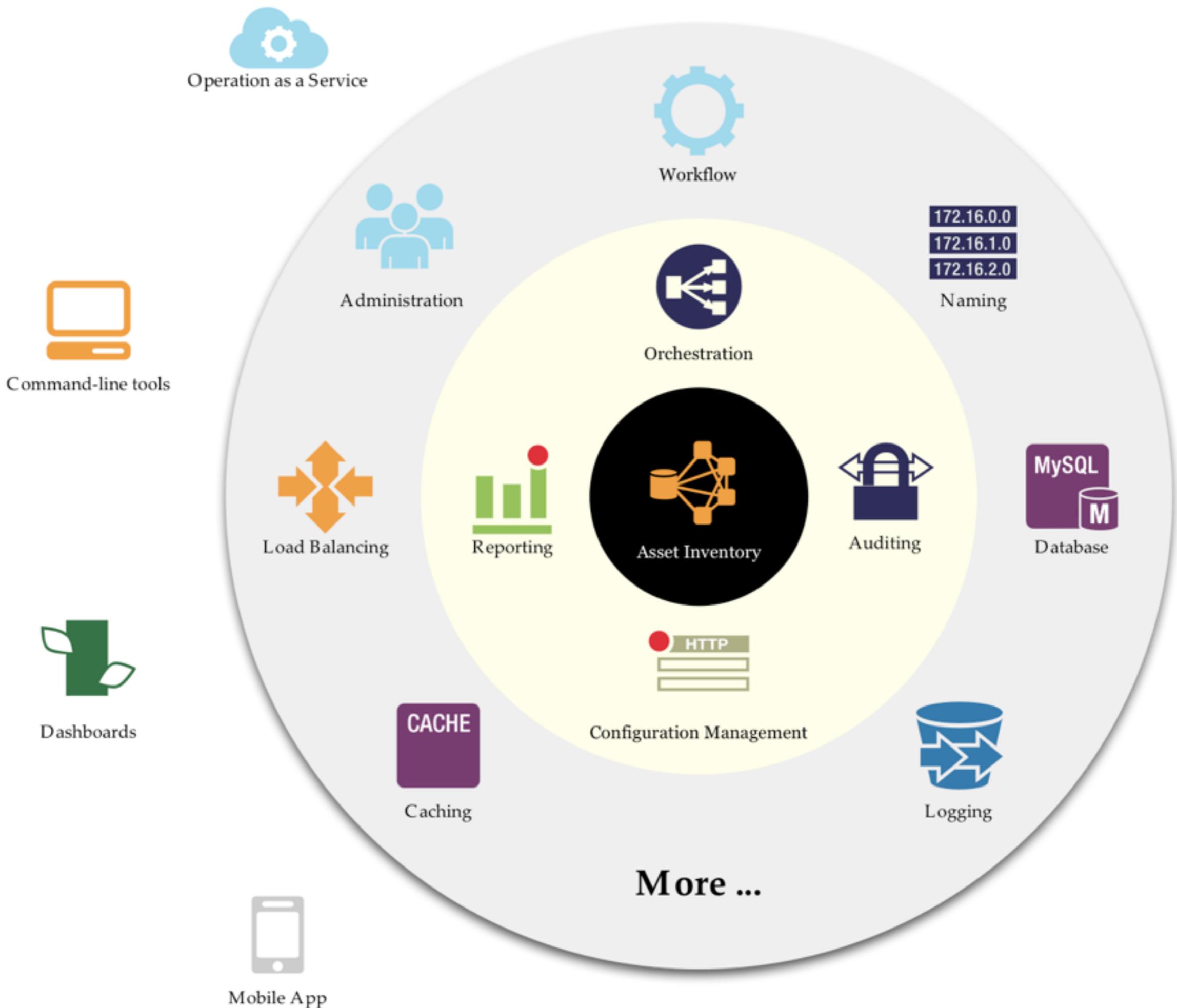
Revolution **O**f **M**artyred **E**lites



ROME

Generalized architecture







Web UI



Mobile App



OaaS

Services



172.16.0.0
172.16.1.0
172.16.2.0

Naming



Monitoring



Database



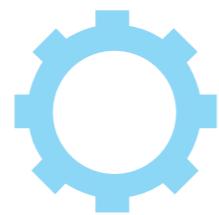
Logging

....

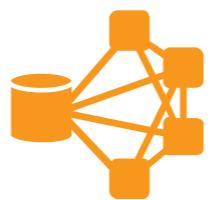
Components



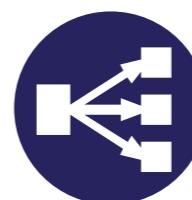
Configuration Management



Workflow



Asset Inventory



Orchestration



Data Storage API

no chocolate without cacao the web is... scalability integration simplicity without coupling through uniformity performance through you might not need that!

303 see other

Tale of the ice-cream maker

The image shows a dark book cover with the title "REST in Practice" in white, sans-serif capital letters. Below the title, the subtitle "with RESTful Web Services" is visible in a smaller font. The background of the cover features a blurred, colorful photograph of what appears to be a tropical beach or a coral reef.

 @getsome restbook Sergey Shishkin
on twitter <http://shishkin.github.com/presentations/f-er.html>
by Steve Klabnik @shishkin

hypermedia representations resources

UI is already restful

REST

APIs

hypermedia is design api
links & forms with resources
GET, POST, PUT, PATCH, DELETE

is it restful? Level 3: hypermedia
POST /REST.svc

```
<PlaceOrder>
  <Product> 123 </Product>
  <Price> 19.90 </Price>
  <Qty> 2 </Qty>
</PlaceOrder>
```

POST /REST.svc

```
{
  PlaceOrder: {
    Product: 123,
    Price: 19.90,
    Qty: 2
  }
}
```

products/books/can-haz-rest

products?title=can-haz-rest

Y2FuLWhhei1yZXN0%3D

rest.svc?op=Order&product=123&qty=2

Level 2: http

Level 1: uri

Level 0

Richardson REST maturity model

is it restful? uri's are opaque what about uri templates?

/user/{id}

RPC & URI make them tunnelingdiscoverable

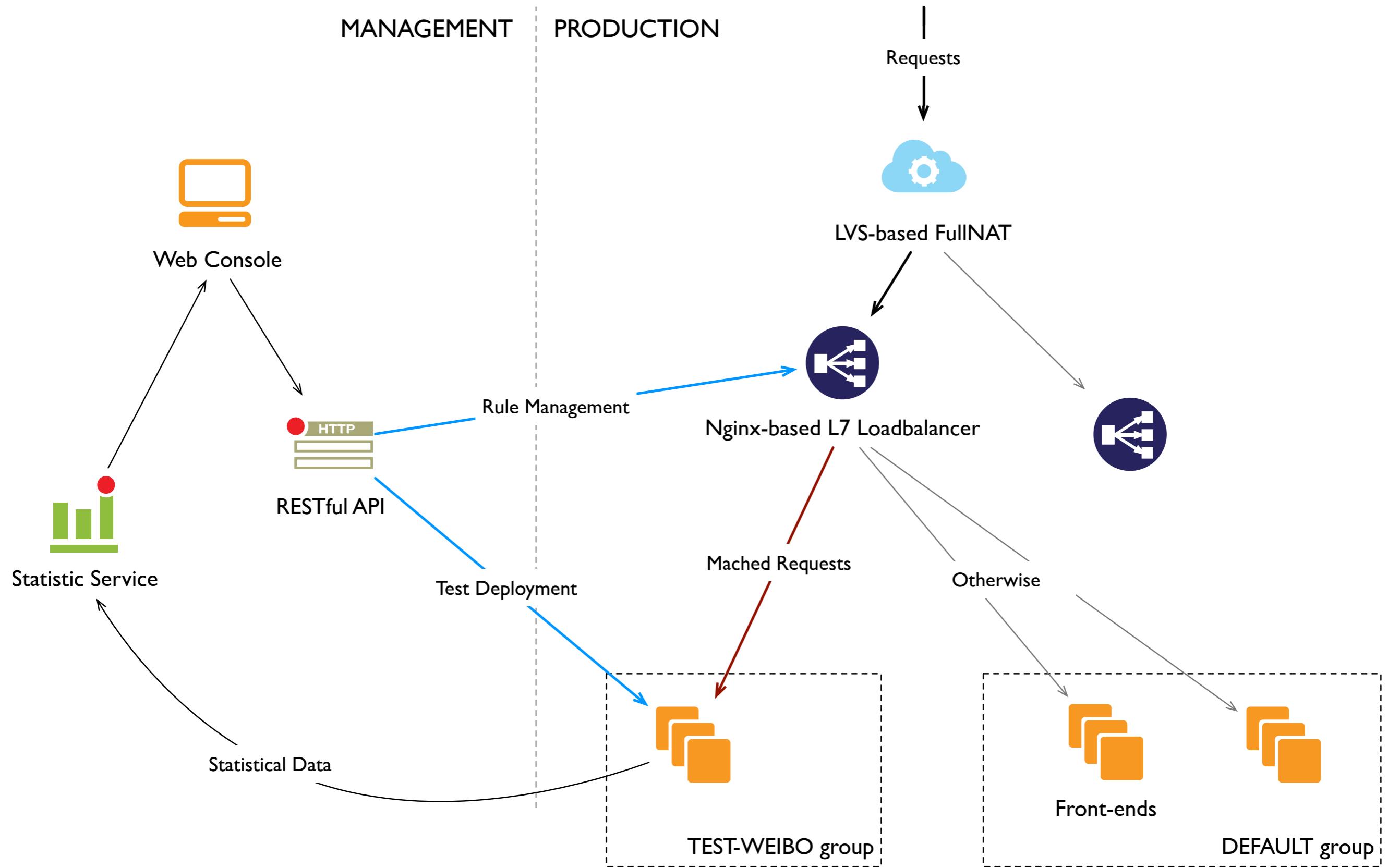
URI identifies
resource, not
representation

http

IT'S SHOWTIME

灰度发布

IT'S SHOWTIME



基于 *Ansible*

- 每个模块只完成较为单一的功能
- 模块之间通过API进行通讯
- 为什么不用Puppet?
- 为什么需要WebUI?
- 有了WebUI为什么还要API?

基于Ansible

- 通过模版引擎解决配置文件管理
- 在Role中定于各种基础操作： 装包、 执行命令、 分发配置文件等
- 可使用Group来进行继承
- 曾经尝试过SaltStack
- 目前为止， Ansible具有让我们满意的扩展性

我们用了什么技术？

- HTTP-based RESTful API with JSON payload
- A variety of MongoDB-like CRUD
- Python
- MongoDB

谢谢

Q&A
@陈尔冬