

# 国内某酒店管理公司kubernetes实践

胡忠良

新老平台介绍

痛点

解决痛点

经验分享

主办：





# 老平台

全球超过8000家酒店80万间客房  
亚洲最大的酒店管理公司

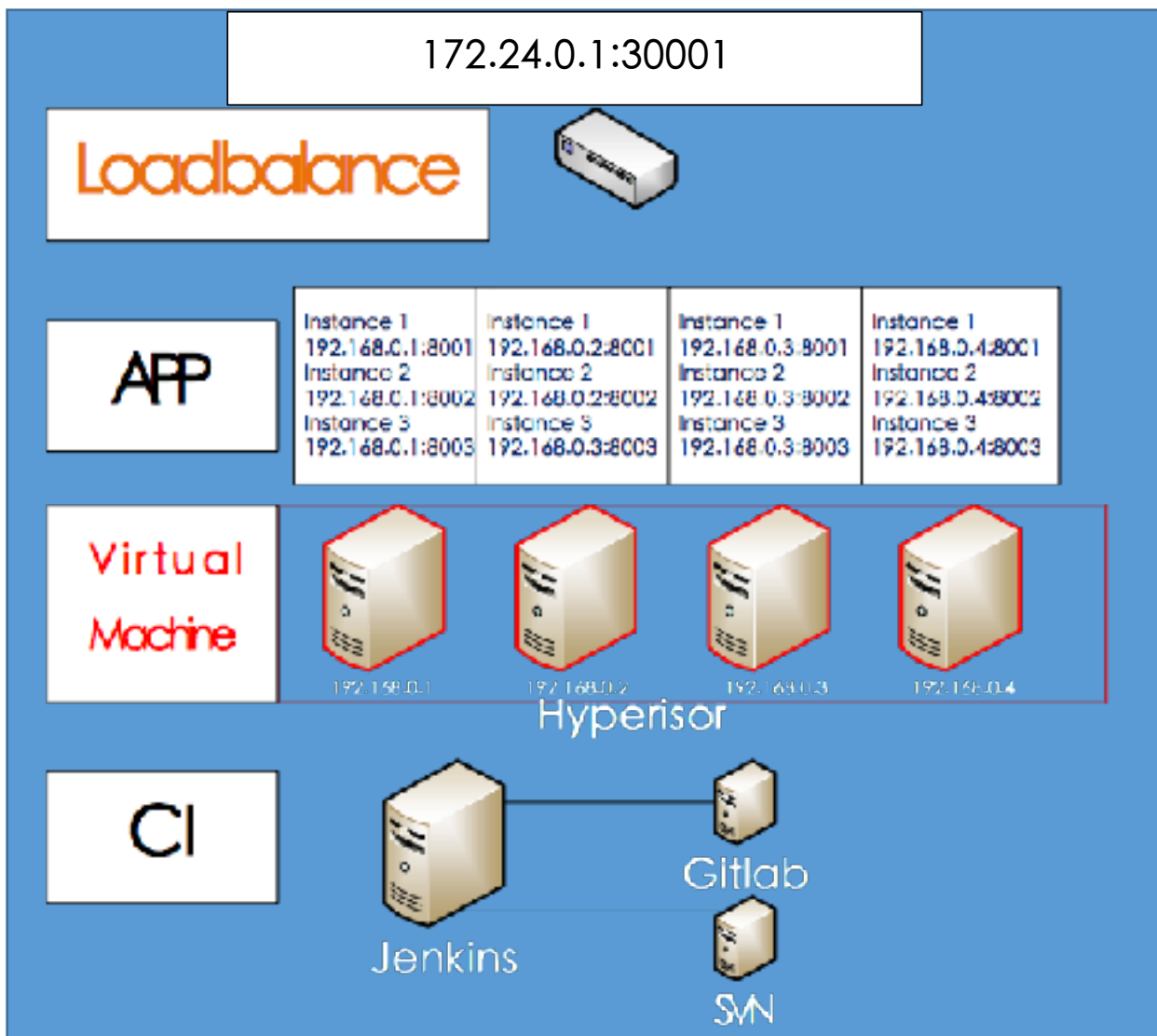
超过600应用  
超过3000台虚拟机  
超过400名开发人员



主办：



before



before

蓝环境--网站平台：war——实例——虚拟机——VIP对照表(172.24.)					
应用名称	实例	虚拟机IP	端口	实例所在console	load balance地址
flamingo	wls_server1	4.2	5066	http://172.24.4.100:7001/console/	172.24.88.12:6065
	wls_server2	4.3	5066		
	wls_server3	4.1	5066		
	wls_server292	4.39	5067	http://172.24.4.120:7001/console	
	wls_server293	4.39	5068		
	wls_server401	4.40	5066		
mgw/tgw/pgw	wls_server21	4.3	5067	http://172.24.4.100:7001/console/	172.24.88.29:6067
	wls_server31	4.4	5067		
	wls_server341	4.34	5066	http://172.24.4.120:7001/console	
	wls_server342	4.34	5067		
admin/wechatAdmin	wls_server32	4.9	5068	http://172.24.4.100:7001/console/	172.24.88.16:6065
	wls_server92	4.10	5068		
	wls_server281	4.28	5066	http://172.24.4.105:7001/console/	
	wls_server1063	4.106	5068		
avatar	wls_server9	4.10	5066	http://172.24.4.100:7001/console/	172.24.88.18:6067
	wls_server10	4.11	5066		
	wls_server292	4.29	5067	http://172.24.4.105:7001/console/	
	wls_server293	4.29	5068		
avataradmin	wls_server9	4.10	5066	http://172.24.4.100:7001/console/	172.24.88.38:6065
	wls_server10	4.11	5066		
	wls_server292	4.29	5067	http://172.24.4.105:7001/console/	
	wls_server293	4.29	5068		
majoraccount	wls_server12	4.2	5068	http://172.24.4.100:7001/console/	172.24.88.24:6069
	wls_server22	4.3	5068		
	wls_server102	4.10	5067	http://172.24.4.120:7001/console	
	wls_server403	4.40	5068		

主办：



after

Hotelservice.bestwehotel.net

Loadbalance



kubernetes

Container

Container

Container

Container

Virtual  
Machine



192.168.0.1



192.168.0.2



192.168.0.3



192.168.0.4

Hyperisor

CI



Jenkins

Gitlab



SVN

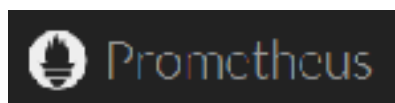
Docker  
registry

主办:





# 使用的开源产品



主办：



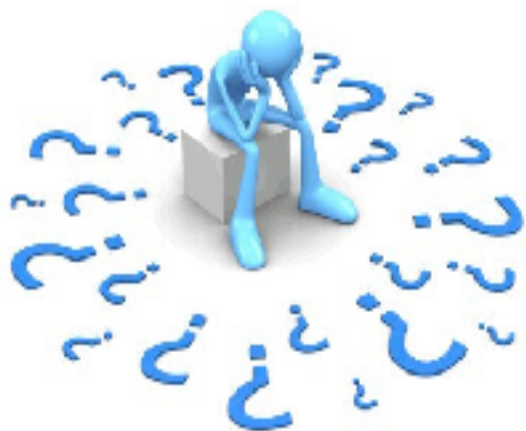
# 我们的痛点

系统架构缺乏弹性

工具或者流程阻碍了测试与版本的上线速度

发布后的系统无法快速查错

服务稳定需要大量人工保障



主办：





## 痛点：系统架构缺乏弹性

基于虚拟机的创建系统环境，占用资源多。

系统部署搭建时间长，系统配置管理困难，扩容牵扯太多，牵一发动全身，CI/CD 均受影响

脏环境问题

统一的weblogic集群的应用池，使开发语言受限制，JRE版本限制



主办：



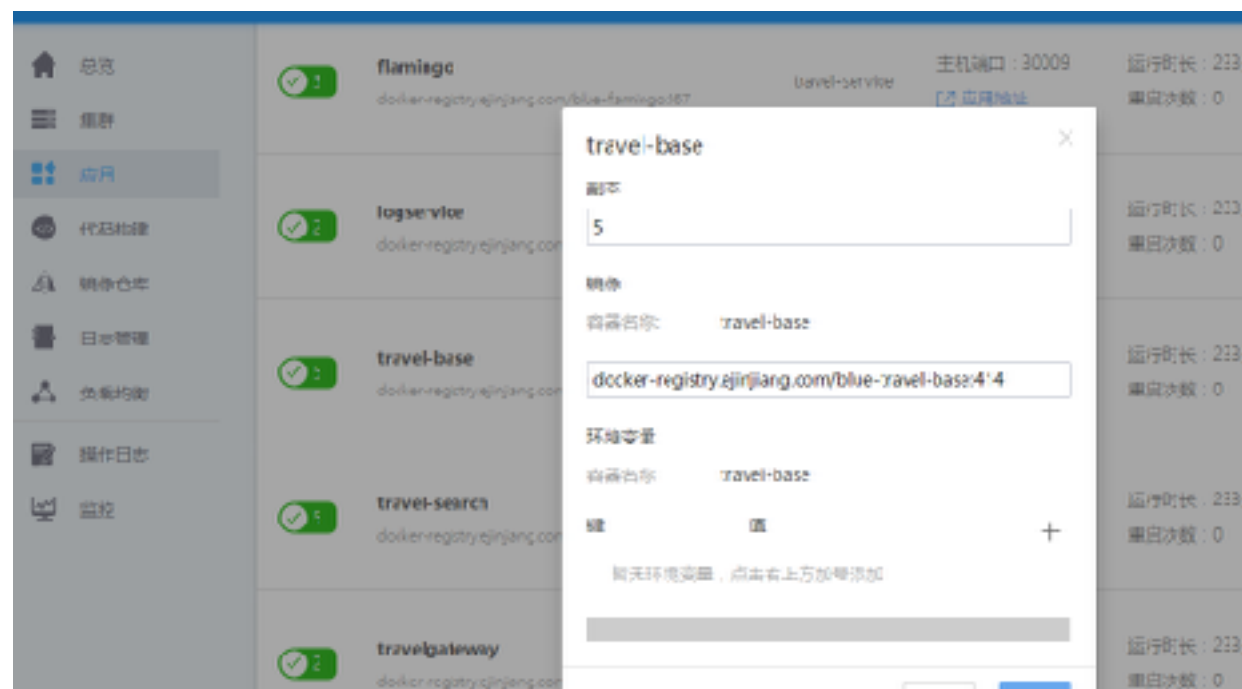
# 解决系统架构缺乏弹性

## Docker & Kubernetes

- 基于容器的集群环境，每个应用环境以进程的方式运行，节省资源，容器可按需进行快速弹性扩展，更合理利用资源
- 使用Dockerfile build镜像，可快速部署环境，并可快速扩展集群。CI/CD 调整，集群扩容ui可实现，无需调整CI/CD
- 集群副本都基于统一镜像，可快速销毁重建
- Kubernetes 管理docker集群，docker集群上可部署不同应用服务器。

主办：





# 痛点：工具或者流程阻碍了测试与版本的上线速度

持续集成，敏捷交付越来越困难

团队无法自主发布，版本上线依赖发布团队，测试团队，发布过程需专人协调，沟通成本高

环境有依赖，配置容易复杂

环境占用问题



主办：





# 解决工具或者流程阻碍了测试与版本的上线速度

## CI/CD优化

集群内环境配置一致，外部调用地址域名化，减少配置量，增加可读性

各组自主发布

可快速创建环境



主办：



[登录](#) | [注册](#)

[Jenkins](#)

[用户](#)  
[任务历史](#)  
[项目关系](#)  
[配置管理](#)  
[Credentials](#)

构建队列

队列中所有构建任务

构建执行状态

空闲

占用

All

pl-all-admin-v1

pl-all-bdp-v1

pl-all-cms-v1

pl-all-collections-v1

pl-all-core-v1

pl-all-coupon-service-v1

pl-all-eagle-v1

pl-all-leg-argint-v1

pl-all-lake-v1

pl-all-may-render-v1

pl-all-nice-v1

pl-all-north-v1

pl-all-north-v1

pl-all-payment-v1

pl-all-qg-v1

pl-all-score-v1

pl-all-vb-v1

pl-all-webota-score-v1

pl-qg-ac-rin-v1

pl-qg-bdp-v1

pl-qg-cms-v1

pl-qg-collections-v1

pl-qg-core-v1

pl-qg-coupon-service-v1

pl-qg-logir-reg-ship-v1

pl-qg-lake-v1

pl-qg-may-render-v1

pl-qg-nice-v1

pl-qg-north-v1

pl-qg-north-v1

pl-qg-payment-v1

pl-qg-qg-v1

pl-qg-score-v1

pl-qg-vb-v1

pl-qg-webota-score-v1

S	W	Name	上次成功	上次失败	上次构建时间
		<a href="#">pl-all-collections-v1</a>	1月20 days - <a href="#">78</a>	无	19 秒
		<a href="#">pl-all-cms-v1</a>	1月17 days - <a href="#">23</a>	无	16 秒
		<a href="#">pl-all-collections-v1</a>	没有	无	无
		<a href="#">pl-all-cms-v1</a>	1月17 days - <a href="#">21</a>	无	27 秒
		<a href="#">pl-all-coupon-service-v1</a>	1月2 days - <a href="#">432</a>	无	18 秒
		<a href="#">pl-all-eagle-v1</a>	1 day 27 小时 - <a href="#">462</a>	无	60 秒
		<a href="#">pl-all-logir-reg-ship-v1</a>	1月21 days - <a href="#">25</a>	无	17 秒
		<a href="#">pl-all-lake-v1</a>	1月21 days - <a href="#">25</a>	无	17 秒

主办:



痛点：发布后的系统无法快速查错

环境问题

查看日志非常痛苦



主办：



## 发布后的系统无法快速查错

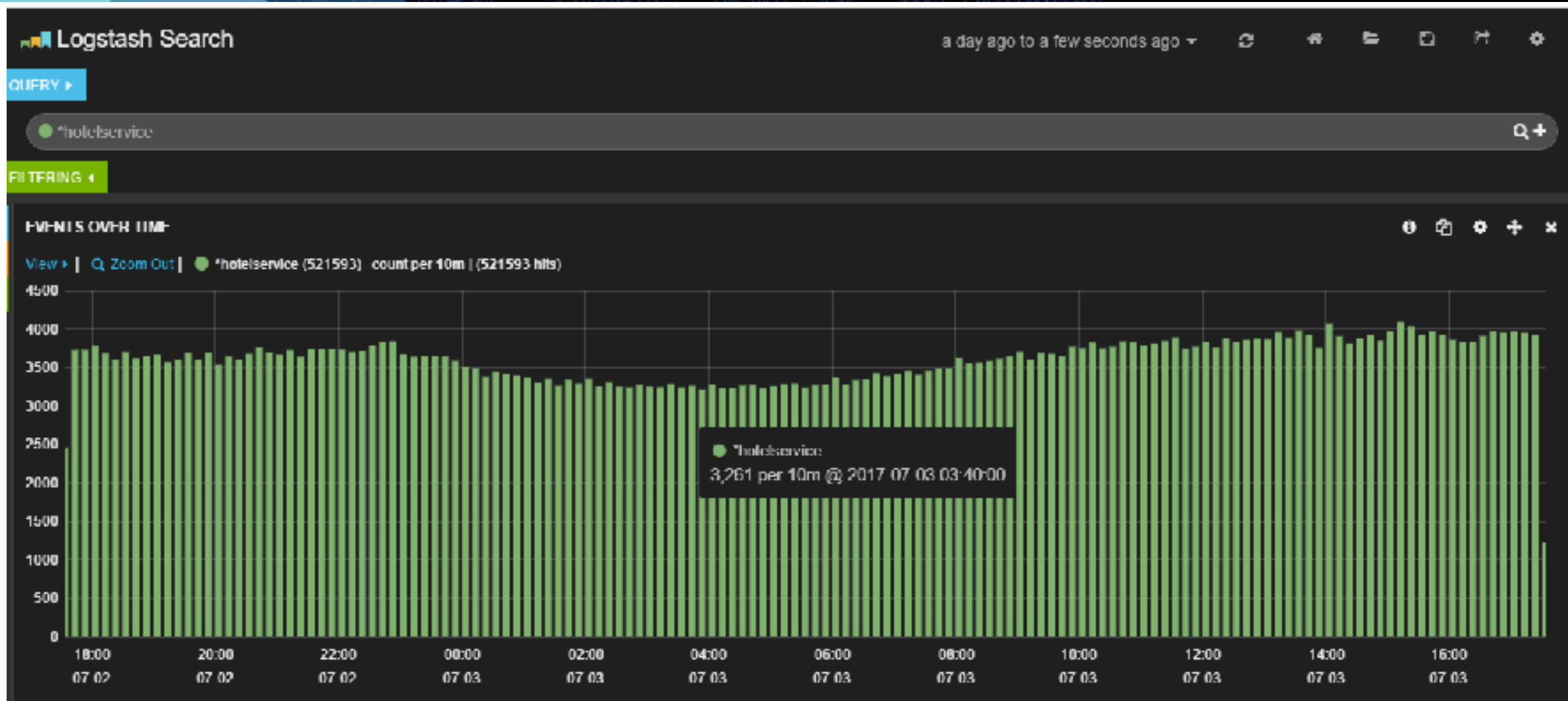
Elk平台，log4j里将日志打到标注输出上，在elk里可查看任何一个应用的日志。

可根据快速调整副本数，以便定位错误（视场景）

主办：







主办:





# 服务稳定需要大量人工保障

服务稳定需要大量的人工保障



主办：



# 服务稳定需要大量人工保障

服务监控加服务自动恢复

Liveness & readiness

<http://kubernetes.io/docs/user-guide/liveness/>



```
livenessProbe:
  httpGet:
    path: /gatewayDC/test.jsp
    port: 8080
    initialDelaySeconds: 120
    timeoutSeconds: 10
  readinessProbe:
    httpGet:
      path: /gatewayDC/test.jsp
      port: 8080
      initialDelaySeconds: 60
      timeoutSeconds: 3
```

主办:





## 服务稳定需要大量人工保障

- <http://172.24.61.1:30211/comorder/metrics/ping>



pong

主办：



mobile-service	mbp-3361790278-g7l51	1/1	Running	38	26d
travel-service	abp-963792191-cwiwz	1/1	Running	0	30d
travel-service	bwhbaseservice-2753269368-o9enz	1/1	Running	5	13d
travel-service	cms-4286596856-0pequ	1/1	Running	0	40d
travel-service	fitpack-377189124-8onk8	1/1	Running	0	11d
travel-service	flamingo-065004540-p8hd1	0/1	ImagePullBackOff	0	12d

主办：



## 总结一下

开源产品

开放性和兼容性

服务的自动修复

可快速，可弹性部署

降低运维成本

主办：



# 其他

开发建议

无状态和持久化数据的处理

Session保持的方法

写一个正确的健康检查url

主办：





# Q&A

Thank you for your time

主办：

