

ArchSummit全球架构师峰会 北京站2015

APM在云服务选型过程中的应用分析

听云研发总监 – 杨金全

Geekbang>

极客邦科技

整合全球最优质学习资源, 帮助技术人 and 企业成长
Growing Technicians, Growing Companies

InfoQ
ueue

专注中高端技术人员的
技术媒体



EGO EXTRA GEEKS' ORGANIZATION
NETWORKS

高端技术人员
学习型社交网络



StuQ
ueue

实践驱动的
IT 职业学习和服务平台



GiT GEEKBANG
INTERNATIONAL
TRAINING
极客邦培训

一线专家驱动的
企业培训服务



旧金山 伦敦 北京 圣保罗 东京 纽约 上海
San Francisco London Beijing Sao Paulo Tokyo New York Shanghai

QCon

全球软件开发大会

2016年4月21-23日 | 北京·国际会议中心

主办方 **Geekbang**  **InfoQ**
极客邦科技

7折 优惠 (截至12月27日)
现在报名, 节省2040元/张, 团购享受更多优惠

www.qconbeijing.com



扫描获取更多大会信息

Magic Quadrant

Figure 1. Magic Quadrant for Application Performance Monitoring Suites



一段代码、几个妹纸...



客户投诉蜂拥而至...



我的应用怎么了...

500

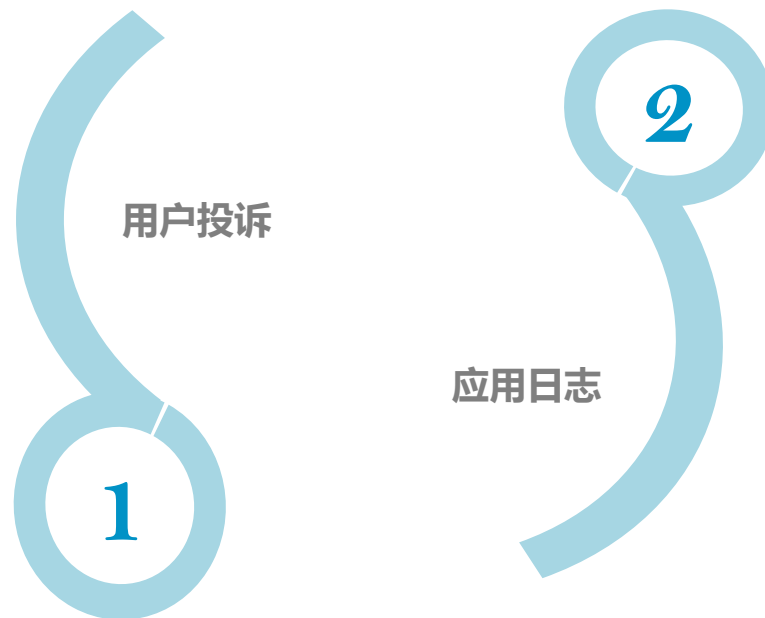
We're sorry, but something went wrong.

We've been notified about this issue and we'll take a look at it shortly.

我的应用怎么了...



一般情况下，应用状态来源有.....



但你随时会面对.....



偶发性故障？



摸不到头脑？



无法定位？



定位不准？

然后.....

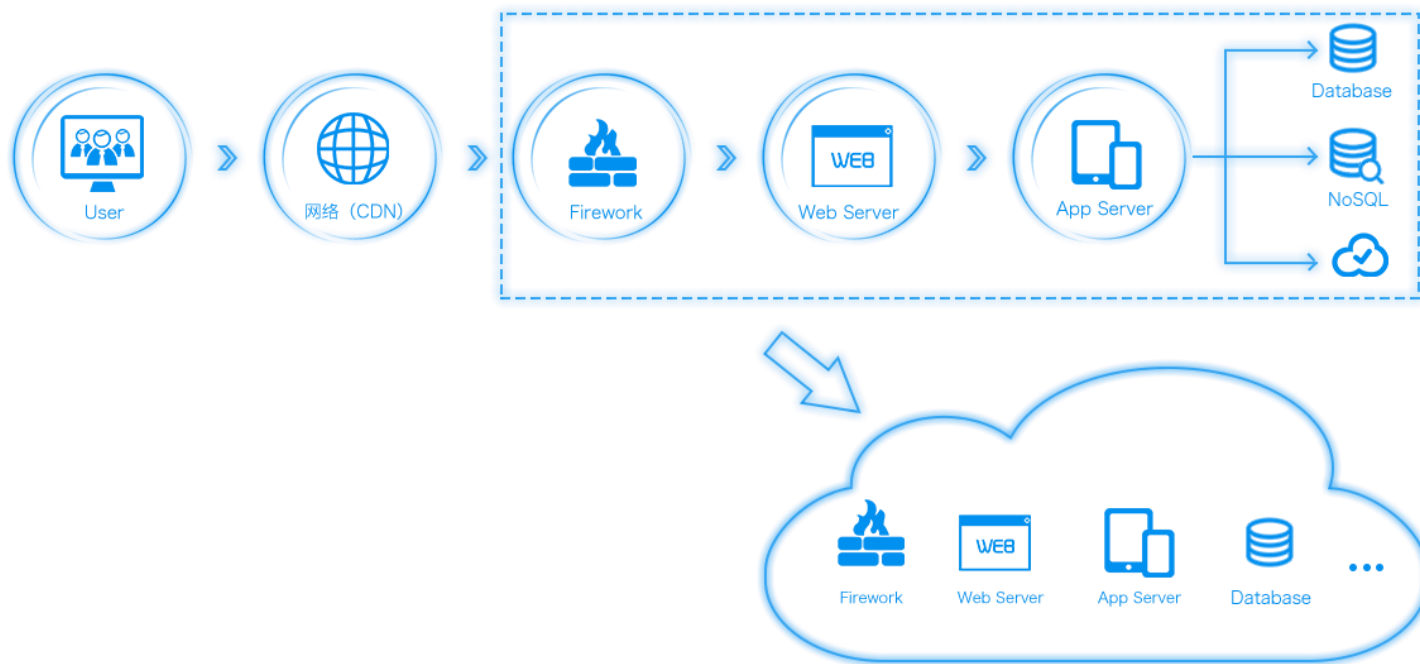


议题

- 云时代变革
- 应用性能管理挑战
- 衡量指标
- 系统监控
- APM – 端到端的应用性能管理
- 助力云选型
- 评测云服务
- 案例实践

云时代变革

云时代变革



云时代变革

降低成本



资源化



快速部署

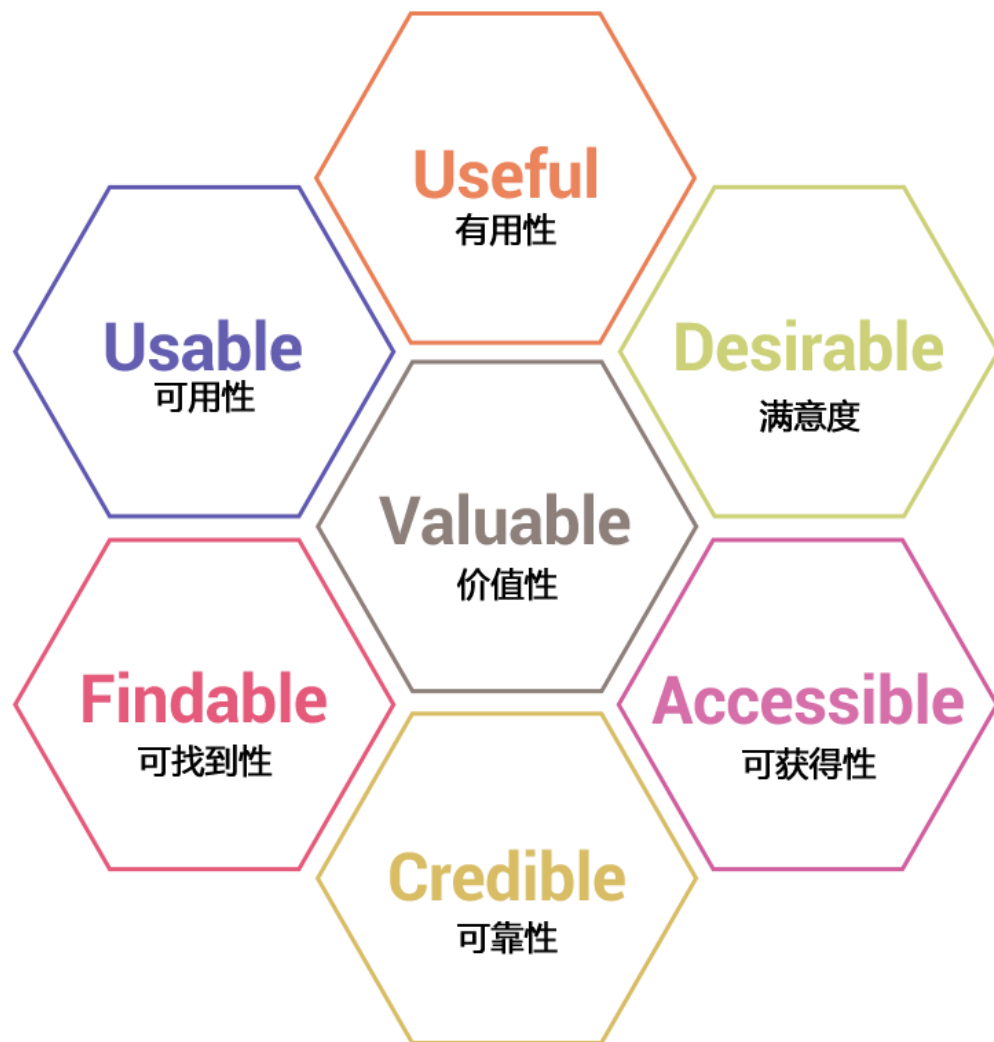


关心业务

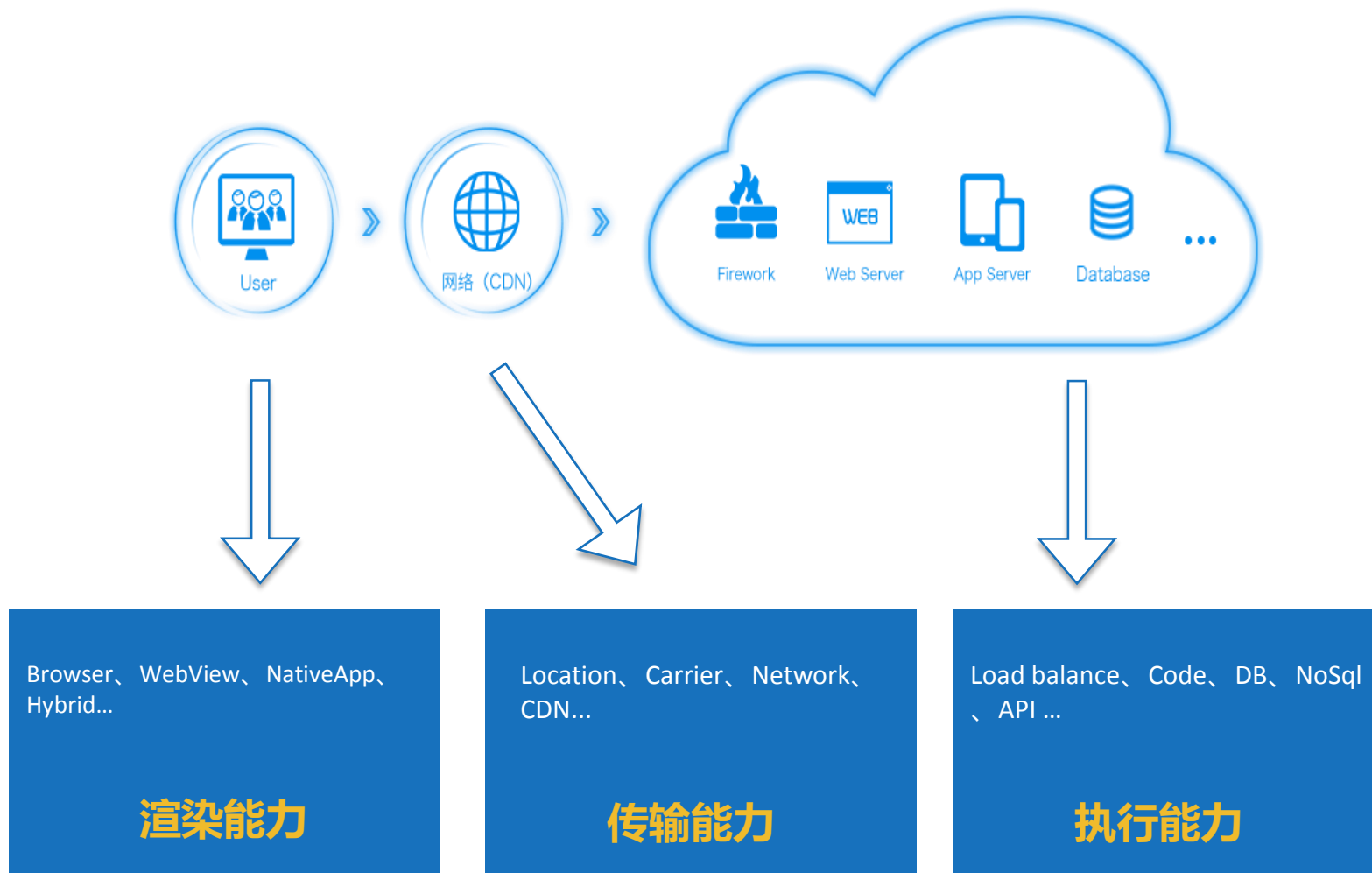


用户体验至上

用户体验之要素



用户体验之影响因素



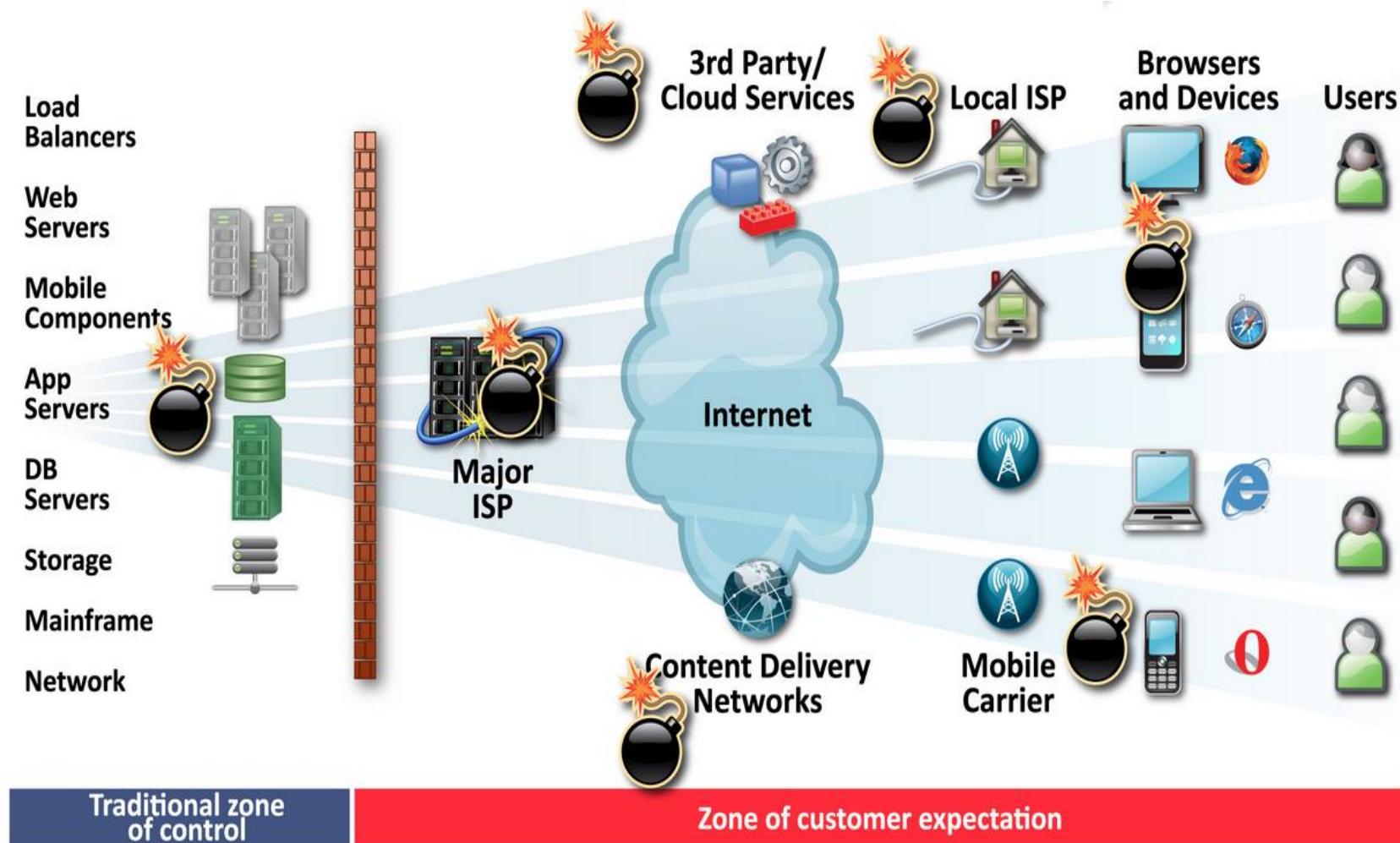
应用性能管理挑战

应用架构变革

- 系统复杂
- 分布式
- 异构
- 微服务
- 云 + 端



传统IT运维挑战



衡量指标

衡量指标

- 交互时间（用户感知）
 - 渲染时间
 - 首屏时间
- 响应时间（独占时间）
 - 网络层时间
 - 阻塞时间
 - 应用层时间
 - 数据库时间
 - NoSQL时间
 - API时间
- 吞吐率
 - rpm -- requests per minute
 - cpm-- calls per minute
- 错误率（可用性）
- 崩溃率
- CPU利用率
- 内存使用率
- 线程数量



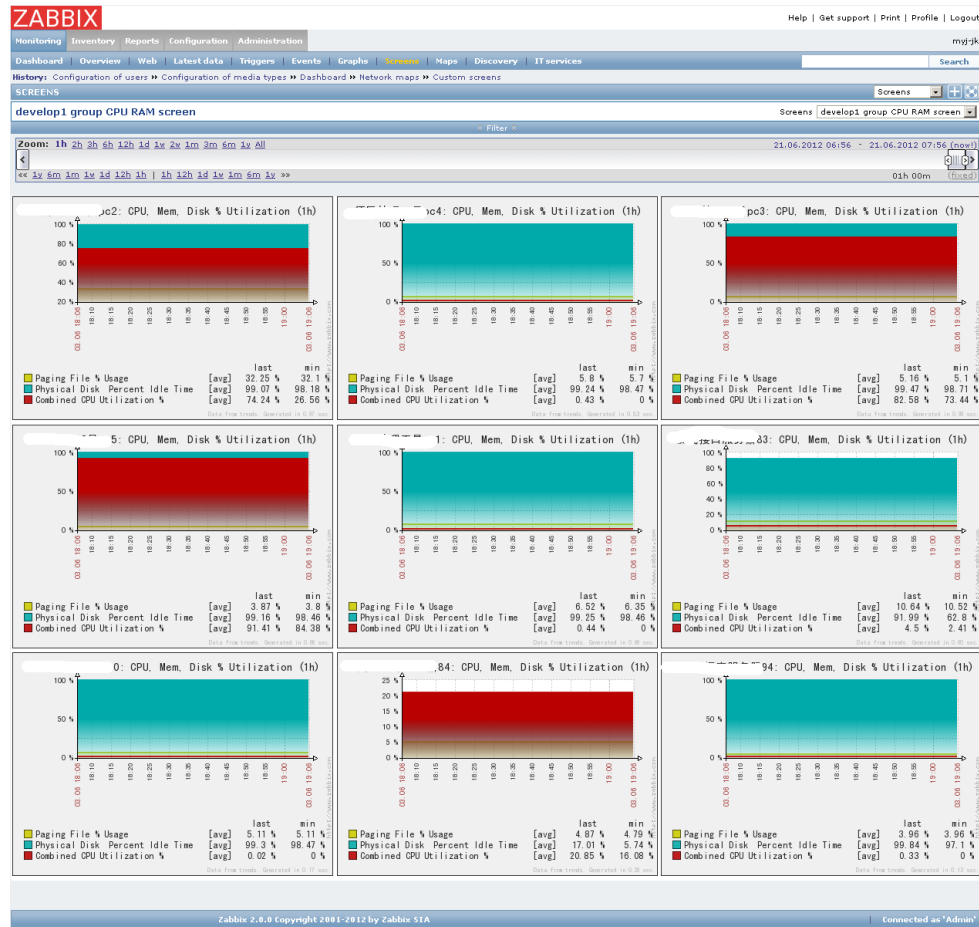
监控(Monitor)

应用性能监控的必要性

- 网络、存储、系统负载、软件 Bug，任何一个点出现问题都有可能影响到整个系统的稳定运行
- 一个完善的系统监控方案要从两个方面帮助我们：
 - 不断检查各项服务的稳定性，出现问题第一时间通知相关人员
 - 记录系统、应用运行的各项指标，帮助运维人员全面掌握系统、应用运行状况，从而做到防患于未然

集成监控工具

- Zabbix
- Nagios
- Ganglia
- ...



应用日志分析



现有监控弊端

- 信息孤立
- 配置复杂
- 非代码级
-

APM – 端到端的应用性能管理

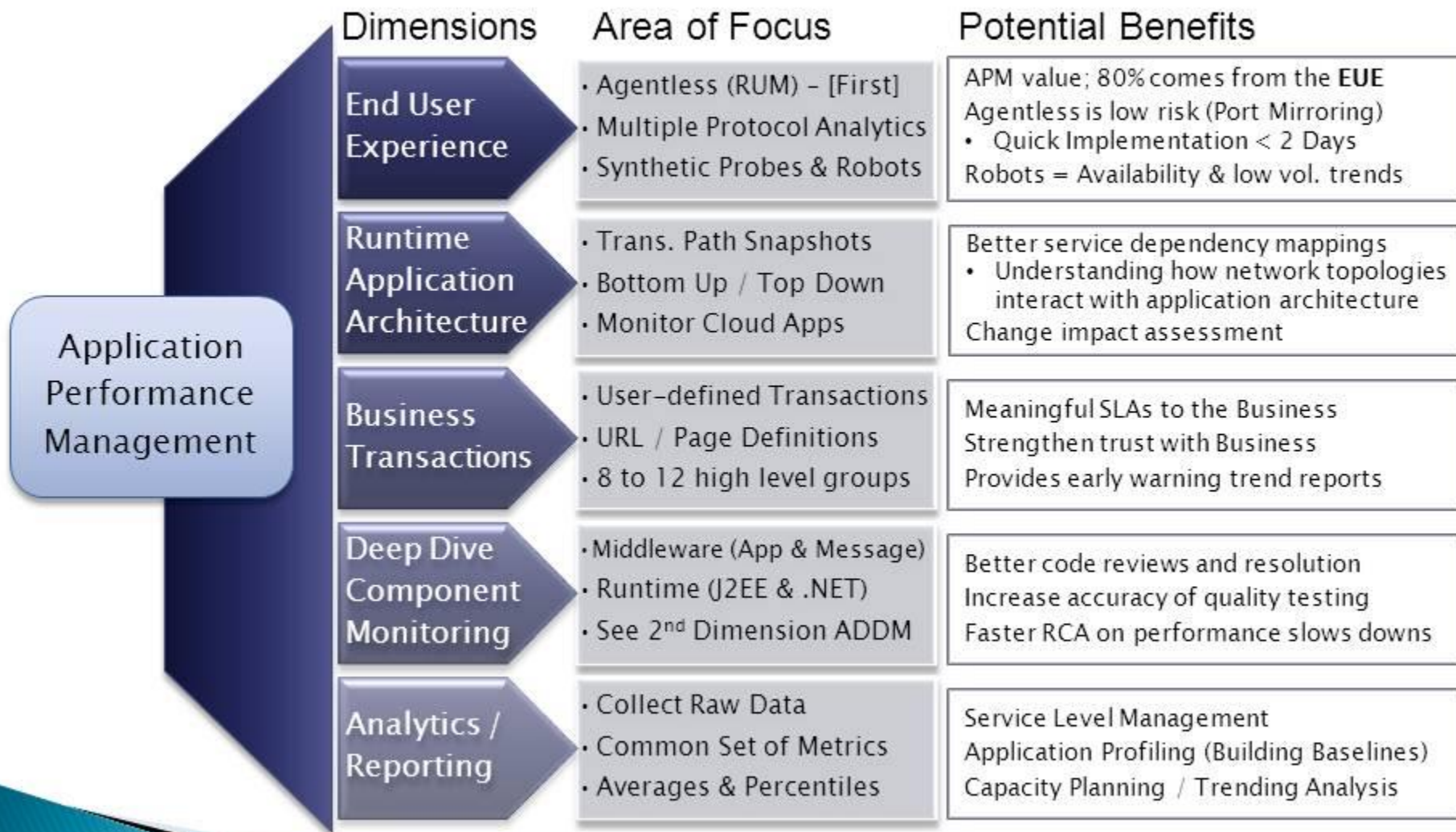
APM

Application Performance Management

对软件应用的性能和可用性进行监控和管理，
致力于发现和定位性能瓶颈和故障，
以保证应用达到预期的服务水平（SLA）

APM Conceptual Framework

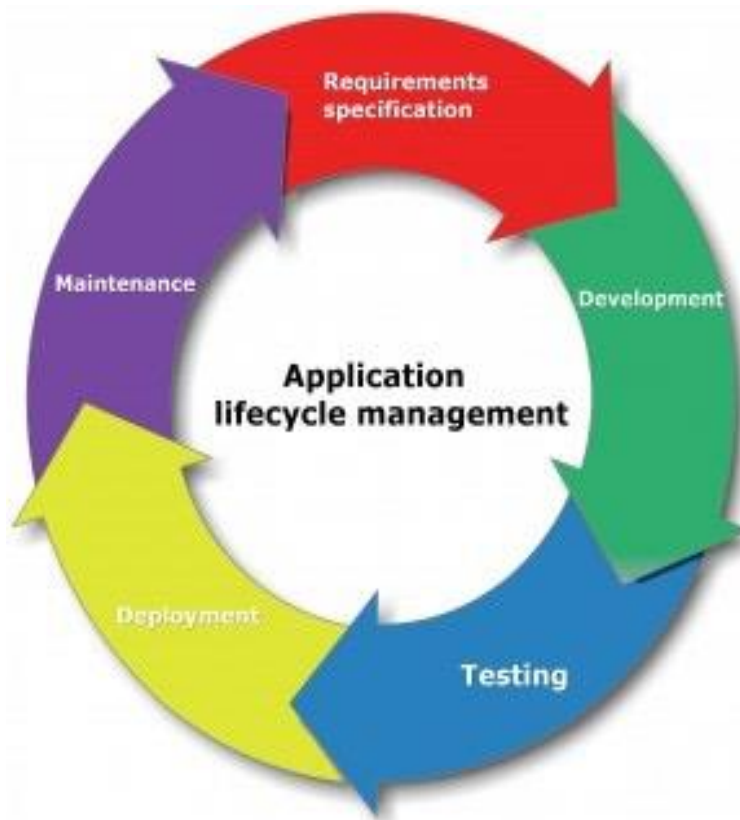
Prioritizing Gartner's APM Model



为什么需要APM

🔗 应用性能**监测**

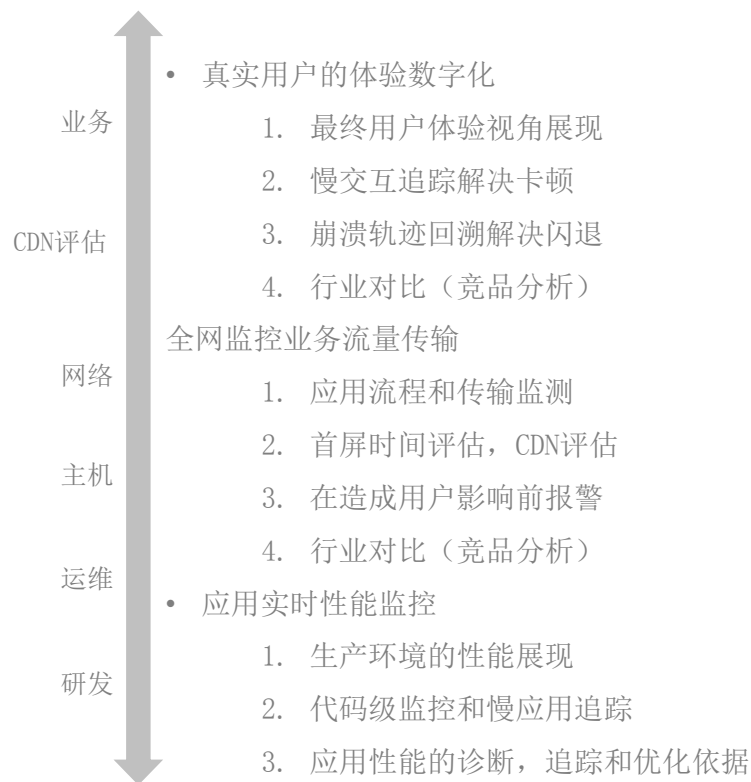
- 应用运营阶段
- 复杂的生产环境
- 发布后



🔗 应用性能**测试**

- 开发测试阶段
- 实验室测试环境
- 发布前

APM助力业务运维



APM 实现方式

位置	方式	技术	侵入式	竞品对标	网络问题定位	全样本	代码级定位	服务监控
客户端	主动	基于自动化测试的拨测	--	○	○	--	--	--
	被动	浏览器嵌码	○	--	--	○	--	--
		Agent自动嵌码	○	--	○	○	○	--
服务端	被动	旁路监听	--	--	○	○	--	○
		Agent自动嵌码	○	--	--	○	○	○

Agent 自动嵌码技术

Java

Bytecode/Instrumentation/ClassLoader

PHP

Opcache/Zend/Extensions/Xhprof

iOS

Hook/Swizzle

Android

Dalvik/Class Rewriting

.Net、Python、NodeJS、JavaScript...

...

如何实现APM

```
public void xxoo() {  
    long startTime = System.currentTimeMillis();  
  
    try {  
        doXX();  
        doOO();  
  
        long endTime = System.currentTimeMillis();  
        long callTime = endTime - startTime;  
  
        APM.reportMetric("xxoo", callTime);  
    } catch (Exception ex) {  
        APM.reportError("xxoo",  
                        ex.getMessage(),  
                        ex.getStackTrace());  
  
        throw ex;  
    }  
}
```

1. 获取方法开始时间

2. 获取方法完成时间，并计算执行时间

3. 上报指标名及性能

4. 上报异常

助力云选型

评测说明

评测目标

- 同一应用在不同云上的用户体验对比

评测手段

- 听云Network主动测试+听云Server&听云Sys产品被动监测

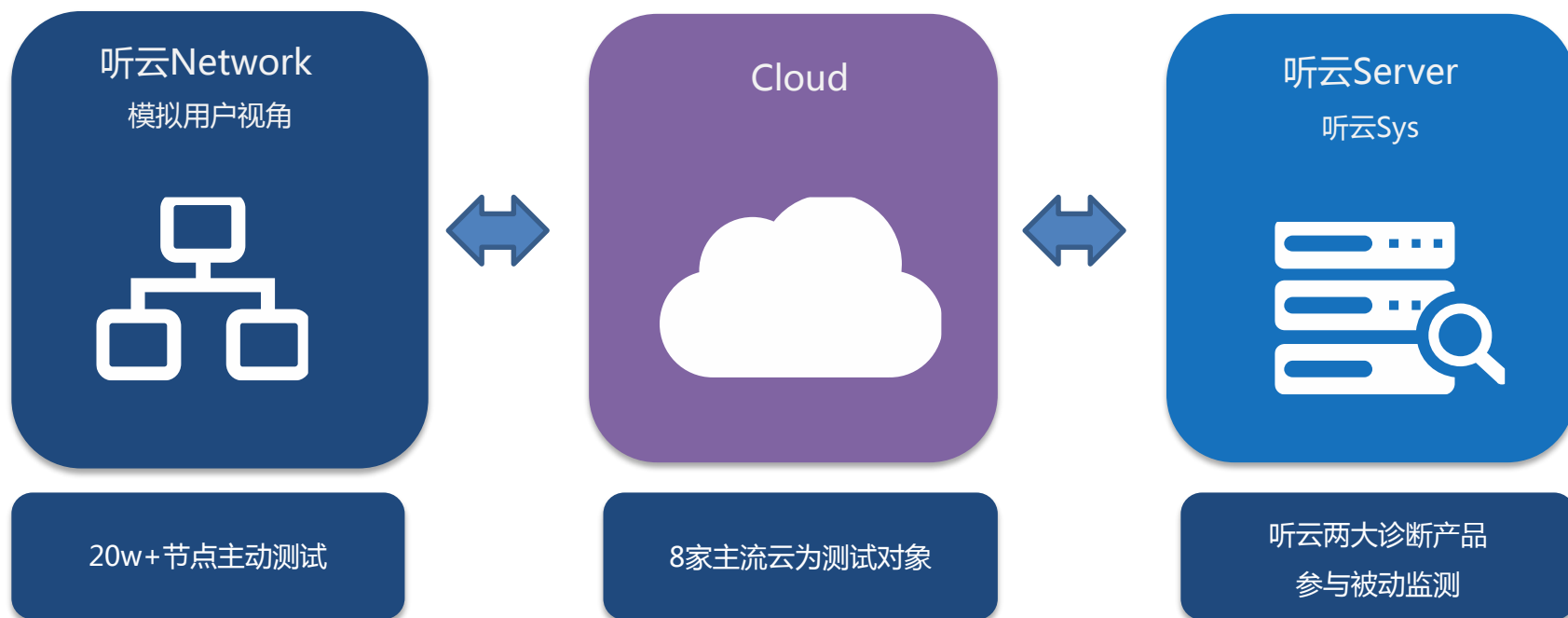
评测环境

- 8家云满足网站应用的基本测试单元

评测周期

- 2015.08.22~2015.09.05

评测逻辑图



备注：听云Network模拟用户（所有客户端都是模拟实际用户部署在各地，不在公有云上。）

对应用首页和内容页进行访问，访问频次如下：

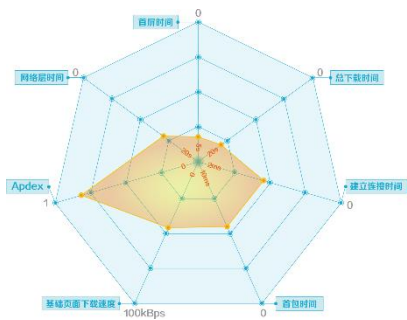
第一测试阶段：20150822日00:00~20150827日00:00 UV/天：3240

第二测试阶段：20140827日00:00~20150905日00:00 UV/天：6480

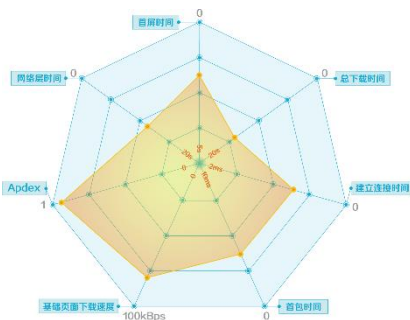
测试结果汇总

- 总结了各个公有云在不同运营商和不同区域的特点之后，下面我们就把每家公有云自身指标进行汇总，以此次测试中每家云的主要指标绘制成雷达图，让我们可以清晰的看到每家云每项指标情况。
- 用户体验类：首屏时间、总下载时间；
应用端指标：首包时间、建连时间、网络层时间、基础页面下载速度
后台综合指标： $\text{Apdex指数} = (1 \times \text{满意数量} + 0.5 \times \text{可容忍数量}) / \text{总样本数}$ ， $T=500$)
- 所有雷达图以测试指标实际值标示，面积越大代表整体测试结果越优秀

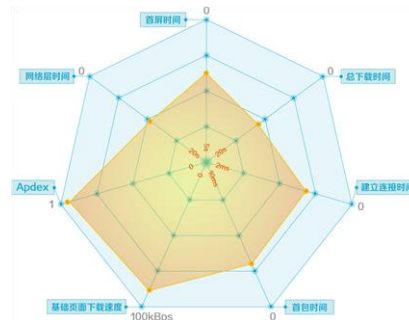
测试结果



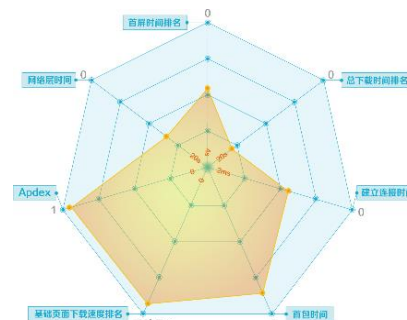
华为云



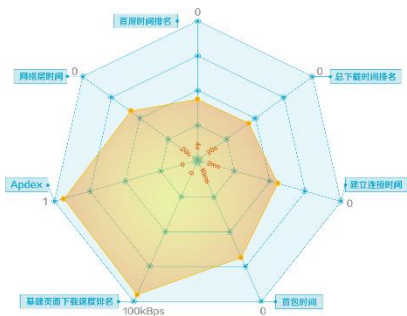
百度开放云



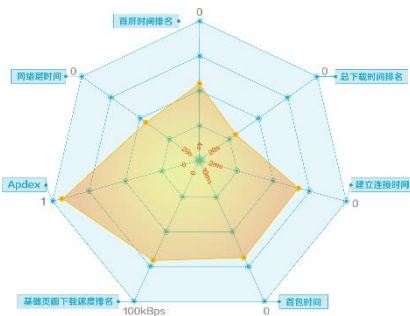
UnitedStack



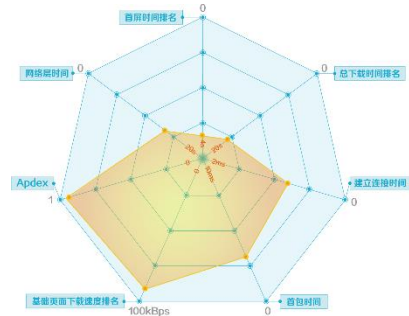
腾讯云



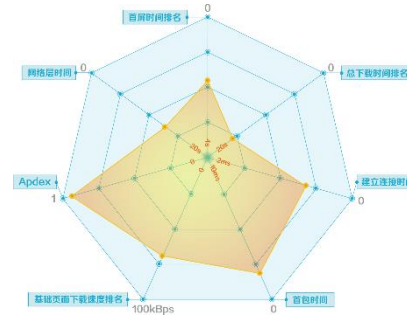
AWS



阿里云



UCloud



青云

评测云服务

评测云服务 - 性能

性能一览

全部

自身服务

第三方

▶ api2.sharesdk.cn:5566 1.335秒

来源: Mob社会化分享

▶ dn-mdpic.qbox.me 0.672秒

来源: 七牛云存储

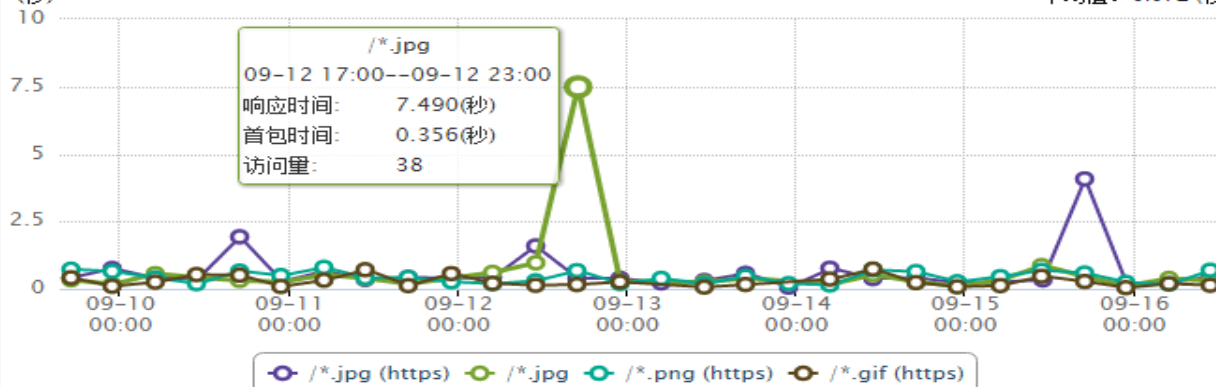
← 返回

TOP5 响应时间最慢URL

响应时间

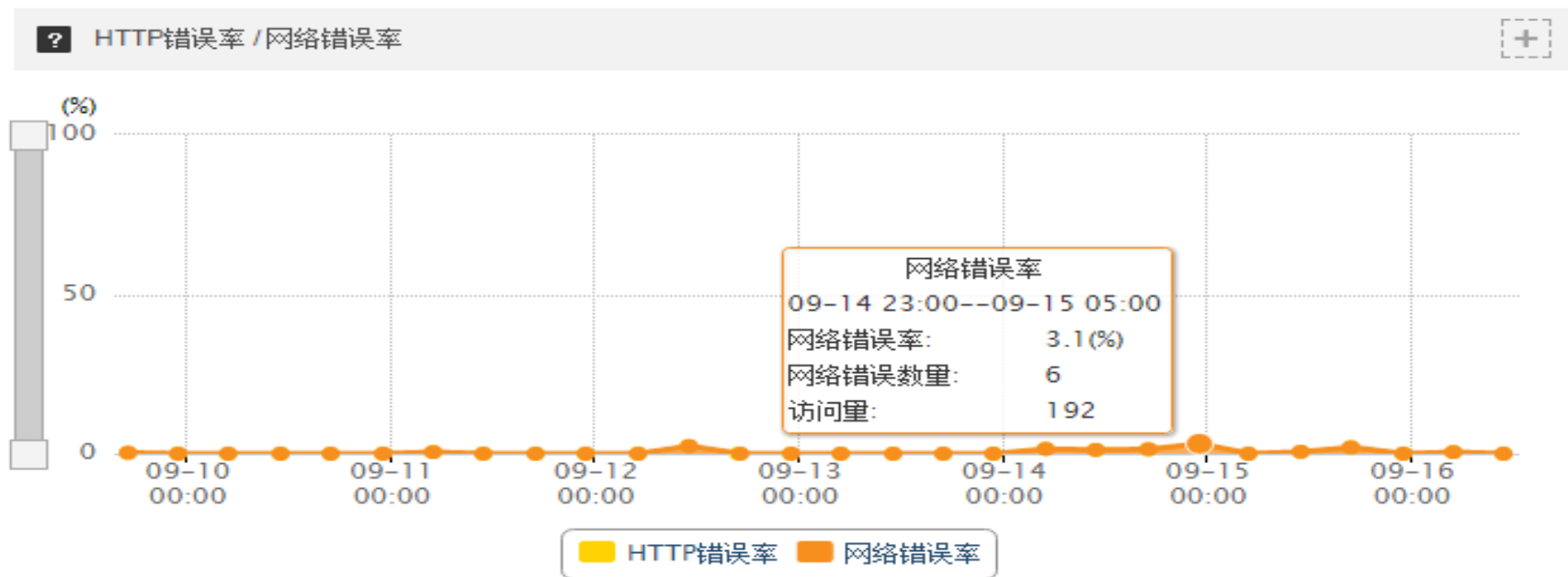
(秒)

平均值: 0.672 (秒)



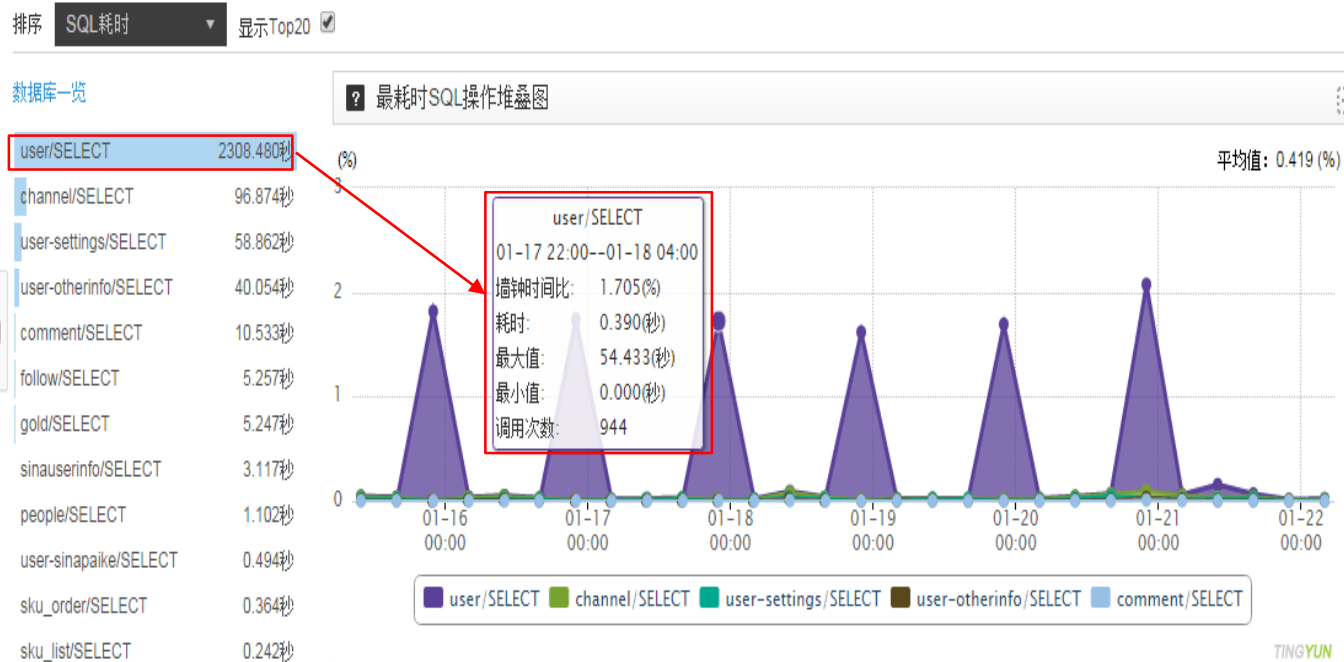
TINGYUN

评测云服务 – 错误



案例分享

SQL性能问题分析 – 发现问题



SQL性能问题分析 – 追溯问题

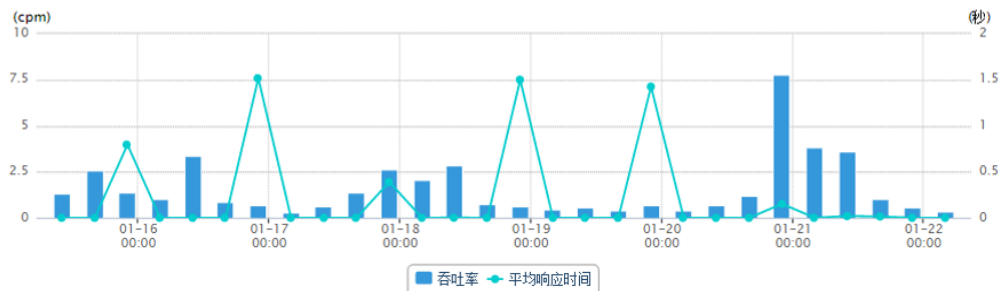
排序 平均响应时间 显示Top20

数据库一览

people/SELECT	0.157秒
user/SELECT	0.146秒
follow/SELECT	0.013秒
news/SELECT	0.005秒
weibojob/SELECT	0.003秒
comment/INSERT	0.002秒
sku_order/UPDATE	0.002秒
channel_extra/INSERT	0.002秒
user/UPDATE	0.002秒
comment/UPDATE	0.002秒
user-settings/SELECT	0.002秒
user-sinapaik/SELECT	0.002秒
sinauserinfo/SELECT	0.002秒
op_stat/SELECT	0.002秒
comment_warehouse/SELECT	0.002秒
news/INSERT	0.001秒

返回

数据库响应时间和吞吐率曲线图



TINGYUN

调用者耗时图表



Legend for the call duration chart:

- CI/[redacted]/index_api
- CI/integral/cashList
- Script/var/www/[redacted]/www/htdocs/index.php
- CI/robcomment/add
- CI/[redacted]/topic
- CI/snsforrob/be_followed
- CI/integral/exchangeList
- CI/snsforrob/batch_mark
- CI/show/htm
- Script/var/www/[redacted]/www/htdocs/index.php
- CI/u/feed
- CI/m/rpt_ios_devid
- CI/integral/exchangeInfo
- CI/user/loginhtm
- CI/integral/cashInfo
- CI/robcomment/com
- CI/robuser/index

SQL性能问题分析 – 定位问题

SQL分析

慢SQL追踪

执行计划分析:

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	user	range	Index_status	Index_status	4	null	3389003	Using where; Using filesort

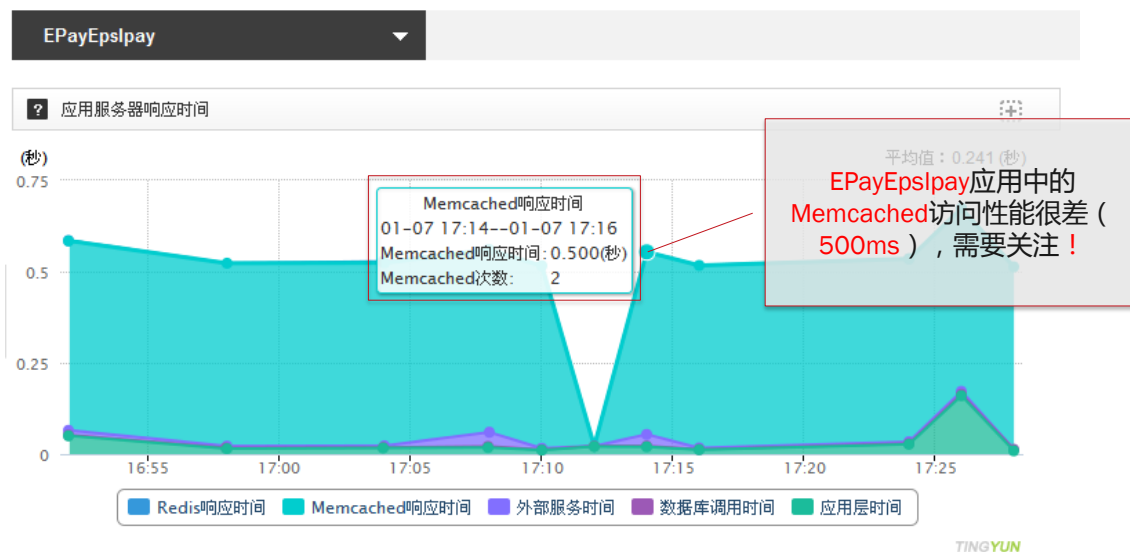
Trace详情

分类

mysql_query	(/var/www/.../htdocs/system/database/drivers/mysql/mysql_driver.php:194)
CI_DB_mysql_driver_execute	(/var/www/.../htdocs/system/database/DB_driver.php:453)
CI_DB_driver.simple_query	(/var/www/.../htdocs/system/database/DB_driver.php:299)
CI_DB_driver.query	(/var/www/.../htdocs/application/controllers/yxxsn/robuser.php:68)
call_user_func_array	(/var/www/.../htdocs/system/core/CodeIgniter.php:359)
require_once	(/var/www/.../htdocs/index.php:205)

select id from user where user.status in (?, ?, -?) order by user.createTime desc limit ?, ?

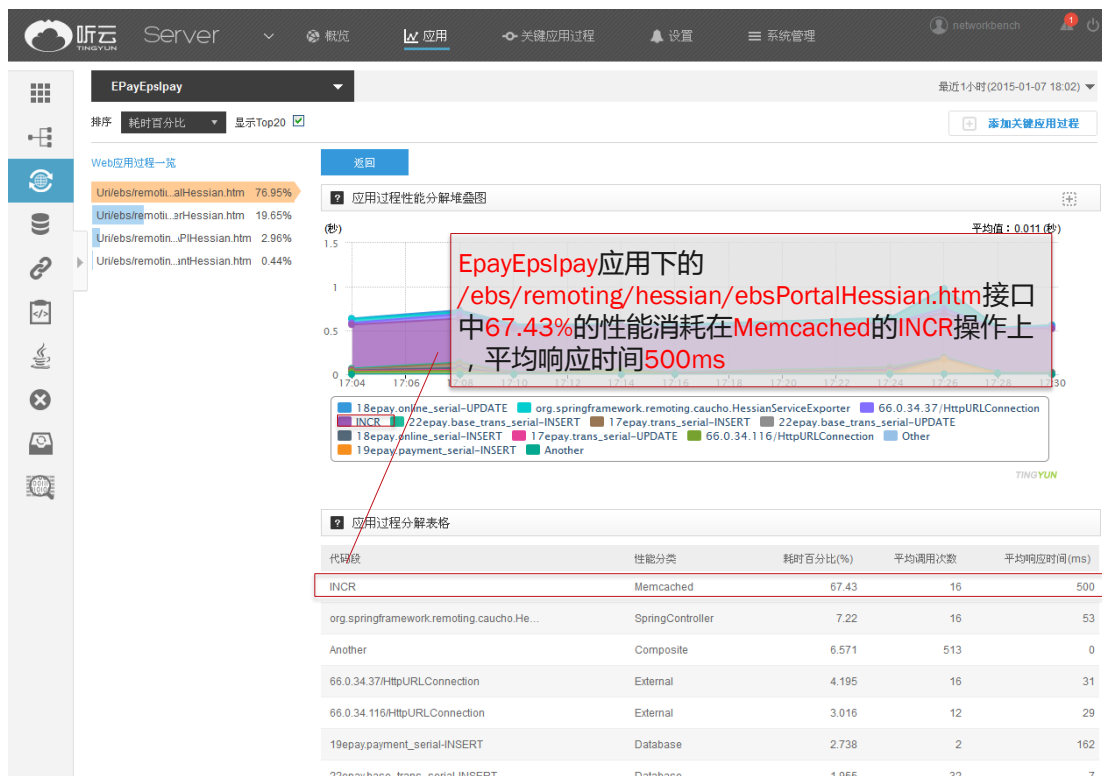
应用性能问题 – 发现问题



分层展示应用响应性能：

- **应用层时间**：应用代码排除其他服务的执行性能
- **数据库调用时间**：SQL数据库访问性能（支持：Oracle, MySQL, SQL Server, DB2...）
- **外部服务时间**：Web Service调用性能
- **NoSQL响应时间**：Memcache, Redis, MongoDB等 NoSQL服务的访问性能

应用性能问题 – 追溯问题 – 应用过程分解



应用性能问题 – 追溯问题 – 慢应用过程追踪

慢应用过程追踪列表		
Web应用过程: <input type="text"/> <input type="button" value="查询"/>		
时间	Web应用过程	服务器响应时间(ms)
2015-01-07 17:26	Uri/ebs/remoting/hessian/ebsPortalHessi...	2084
2015-01-07 17:10	Uri/ebs/remoting/hessian/ebsPortalHessi...	917
2015-01-07 17:26	Uri/ebs/remoting/hessian/ebsPortalHessi...	747
2015-01-07 17:26	Uri/ebs/remoting/hessian/ebsPortalHessi...	707
2015-01-07 17:07	Uri/ebs/remoting/hessian/ebsPortalHessi...	679
2015-01-07 17:11	Uri/ebs/remoting/hessian/ebsPortalHessi...	642
2015-01-07 17:16	Uri/ebs/remoting/hessian/ebsPortalHessi...	634
2015-01-07 17:26	Uri/ebs/remoting/hessian/ebsPortalHessi...	628
2015-01-07 17:30	Uri/ebs/remoting/hessian/ebsPortalHessi...	624
2015-01-07 17:15	Uri/ebs/remoting/hessian/ebsPortalHessi...	624
<input type="button" value="加载更多"/>		

1月7日17:26, EPayEpsIpay应用下的
/ebs/remoting/hessian/ebsPortalHessian.htm接口的有一次响应时间达到了2.084秒, 超过预先设定的阈值500ms, 系统记录了这次访问性能追踪数据

听云Server可以设置慢应用过程追踪阈值, 当某一个应用过程的响应时间超过阈值时, 系统会自动记录详细的性能追踪记录

应用性能问题 – 追溯问题 – 慢应用过程分解

应用: EPayEpsIpay

追踪时间: 2015-01-07 17:26:54

服务器响应时间: 2.084 (s)

实例信息: JAVA:ronge4

URL: [/ebs/remoting/hessian/ebsPortalHessian.htm](#)

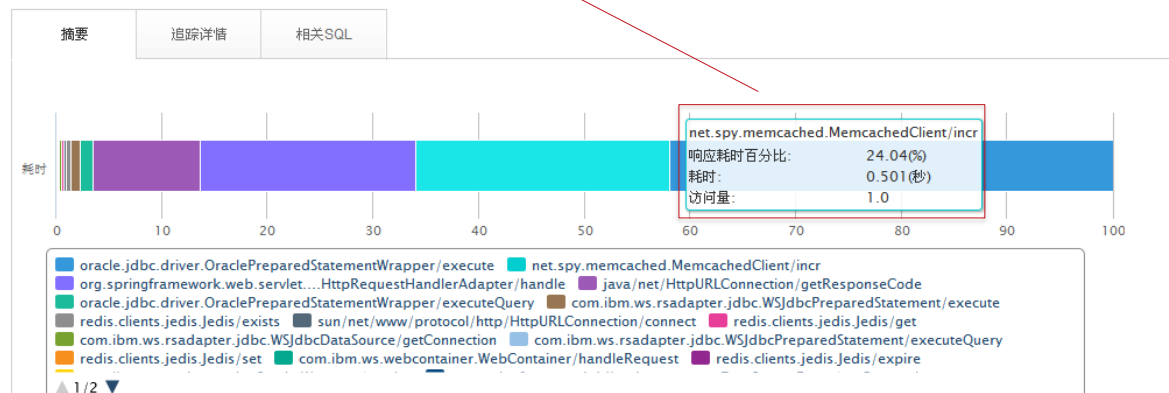
HttpStatus: 200

threadName: WebContainer : 2

referer:

共有25个

性能追踪摘要里展示本次访问过程中各代码模块的耗时占比，其中可见 `net.spy.memcached.MemcachedClient.incr()` 方法的调用耗时0.5秒，占比超过24%



应用性能问题 - 定位问题

慢应用

应用: EPay

追踪时间: 2023-08-15 14:00:00

服务器响应: 200

实例信息: 192.168.1.100

URL: /ebs/

httpStatus: 200

threadName: http-nio-8080-exec-1

referer:

摘要

展开所有

分类

WebCt

Web

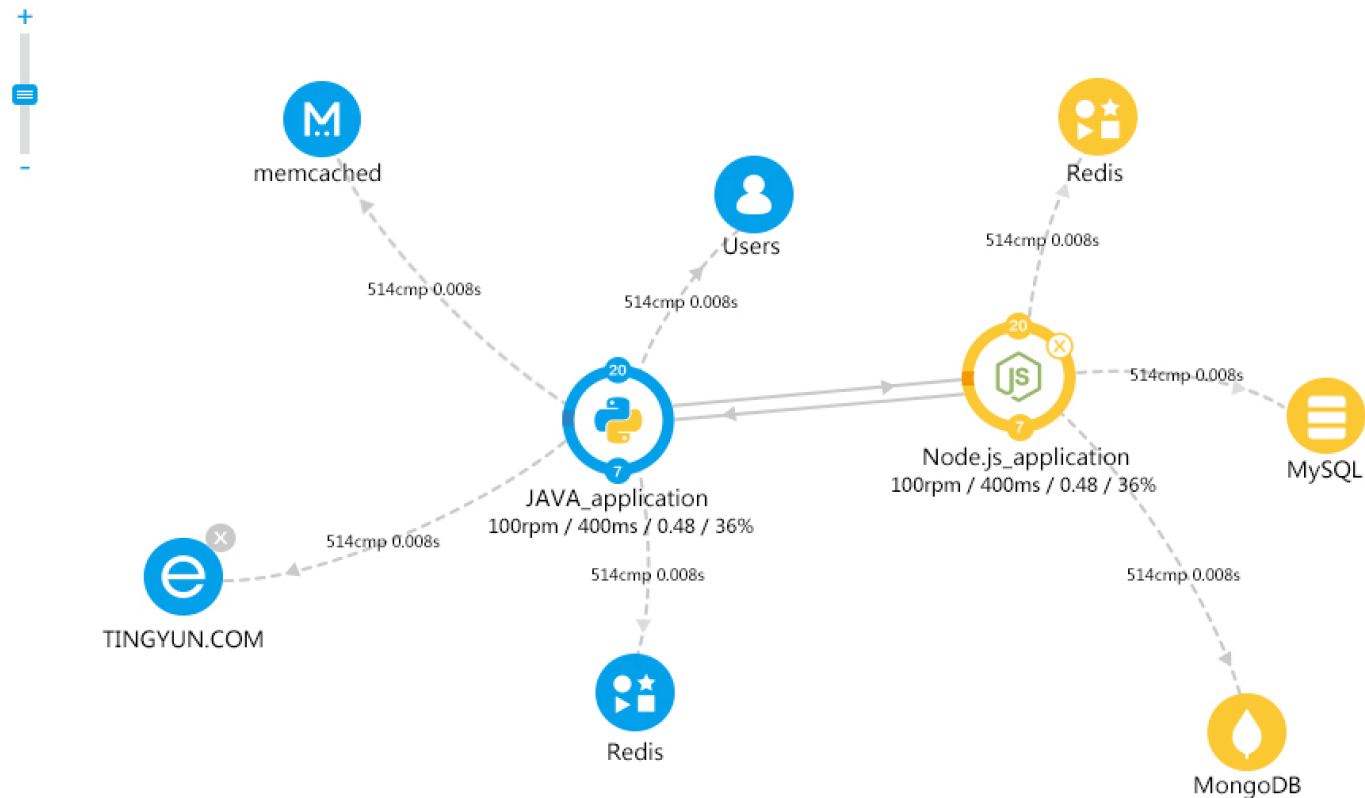
StackTrace

Class	Method	Line
net.spy.memcached.MemcachedClient	incr	1788
com.hundsun.monitor.api.MemcachedManager	increment	201
com.hundsun.monitor.api.MonitorAPI	memcached_Counter	112
com.hundsun.monitor.api.MonitorAPI	api_WriteTranInfo	84
com.hundsun.epay.ebs.portal.services.BasePortalService	finallyAction	195
com.hundsun.epay.base.BaseService	excute	31
com.hundsun.epay.ebs.portal.services.PortalHessianServiceImpl	excute	114
sun.reflect.GeneratedMethodAccessor	255 invoke	(Unknown Source)
sun.reflect.DelegatingMethodAccessorImpl	invoke	(DelegatingMethodAccessorImpl.java:37)
java.lang.reflect.Method	invoke	(Method.java:611)
org.springframework.aop.support.AopUtils	invokeJoinpointUsingReflection	309
org.springframework.aop.framework.ReflectiveMethodInvocation	invoke	(ReflectiveMethodInvocation.java:183)
org.springframework.aop.framework.ReflectiveMethodInvocation	proceed	(ReflectiveMethodInvocation.java:150)
org.springframework.remoting.support.RemoteInvocationTraceInterceptor	invoke	(RemoteInvocationTraceInterceptor.java:77)
org.springframework.aop.framework.ReflectiveMethodInvocation	proceed	(ReflectiveMethodInvocation.java:172)
org.springframework.aop.framework.ReflectiveMethodInvocation	proceed	(ReflectiveMethodInvocation.java:172)
com.caucho.hessian.server.HessianSkeleton	invoke	(HessianSkeleton.java:306)
com.caucho.hessian.server.HessianSkeleton	invoke	(HessianSkeleton.java:221)
org.springframework.remoting.caucho.HessianExporter	doInvoke	(HessianExporter.java:198)
org.springframework.remoting.caucho.HessianExporter	invoke	(HessianExporter.java:118)
org.springframework.remoting.caucho.HessianServiceExporter	handle...	(HessianServiceExporter.java:66)

从追踪详情可查看详细代码调用堆栈，通过堆栈分析，可获取调用这条非常慢Memcached的INCR操作的用户代码在PortalHessianServiceImpl.java文件第114行的excute方法中。

分布式集群问题 – 应用多级拓扑 – 发现问题

■ 正常 ■ 警报 ■ 严重警报 ■ 无数据 ■ 图例



分布式集群问题 – 应用多级拓扑 – 追溯及定位

应用过程慢追踪

应用: transaction client222

追踪时间: 2015-07-16 22:48:

服务器响应时间: 0.349 (s)

实例信息: JAVA:fengzhiyin-r

应用过程慢追踪

应用: transaction server

追踪时间: 2015-07-16 22:48:50

服务器响应时间: 0.297 (s)

实例信息: JAVA:fengzhiyin-mac.lan:18080

摘要

追踪详情

展开所有

全部关闭

分类

▼ CoyoteAdapter.service

▼ CoyoteAdapter.service

▼ WsFilter.doFilter

▼ HttpServlet.service

CloseableHttp

Close

外部

摘要

追踪详情

相关SQL

展开所有

全部关闭

分类		持续时间(ms)	时间占比(%)	时间偏移量(ms)
▼ CoyoteAdapter.service	≡	298	100.00	0
▼ CoyoteAdapter.service	≡	298	100.00	0
▼ WsFilter.doFilter	≡	296	99.66	1
▼ HttpServlet.service	≡	296	99.66	1
Jedis.get		2	0.67	1
com.mchange.v2.c3p0.impl.NewProxyStatement.execute	🔍 ≡	10	3.37	3
MemcachedClient.get		2	0.67	13
DBCollection.find		0	0.00	16
▶ sun/net/www/protocol/http/URLConnection.getInputStream	≡ 🌐	13	4.38	16
▼ sun/net/www/protocol/http/URLConnection.getInputStream	≡ 🌐	261	87.88	32
sun/net/www/protocol/http/URLConnection.connect	≡ 🌐	126	42.42	32
java/net/URLConnection.getResponseCode	🌐	0	0.00	292

问题管理及处理 – Alarm – 策略

← 修改Server警报策略

策略名称: Server应用

监控项目: 应用

触发条件: 吞吐率 ≥ 100 rpm (每分钟请求率或访问量, 当选择分组统计时, 按各分组分别计算)

错误率

触发条件: 吞吐率 ≥ 100 rpm ☐ 例外

告警阈值: 持续 1 分钟 大于 10 %

严重阈值: 持续 2 分钟 20 %

Apdex

触发条件: 吞吐率 ≥ 100 rpm ☐ 例外

告警阈值: 持续 1 分钟 小于 0.99

严重阈值: 持续 2 分钟 0.4

应用响应时间

☒ 静态阈值 ☐ 动态阈值

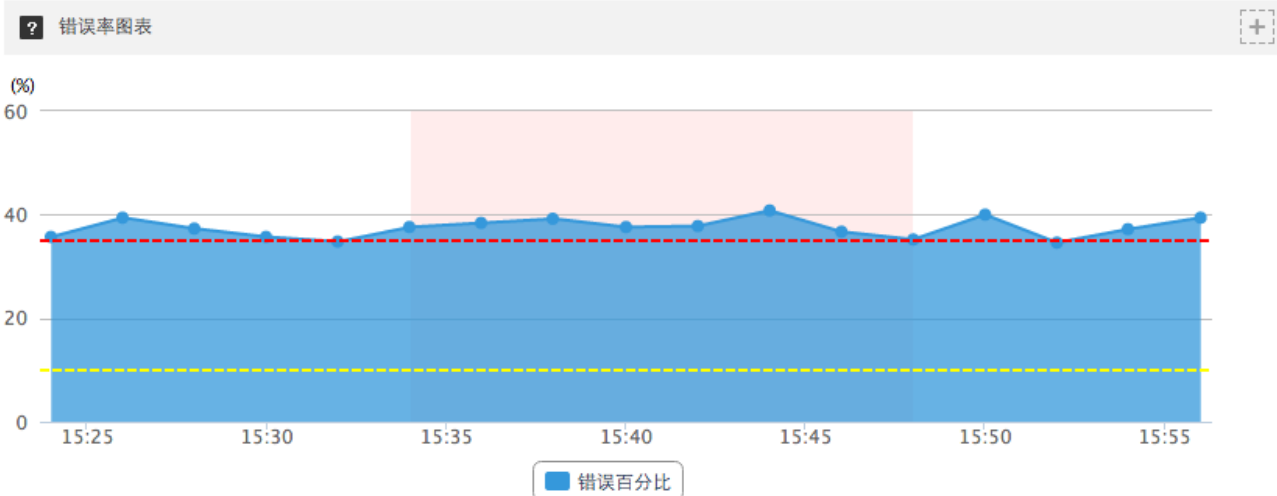
触发条件: 吞吐率 ≥ 100 rpm ☐ 例外

告警阈值: 持续 1 分钟 大于 100 毫秒

严重阈值: 持续 2 分钟 2000 毫秒

问题管理及处理 – Alarm – 事件管理

警报内容	开始时间	结束时间	持续时间	警报状态
lens-register 错误率超过严重阈值35(%)	2015-12-04 15:34:00	2015-12-04 15:48:00	14 分钟	警报解除



警报过程(历史)	
12-04 15:48	事件结束
12-04 15:48	发送解除报警邮件或短信: lens... 解除警报
12-04 15:48	解除警报: len... 错误率超过严重阈值35(%)
12-04 15:39	发送报警邮件或短信: lens-regi... 触发警报
12-04 15:34	警报: lens-reg... 错误率超过严重阈值35(%)
12-04 15:34	事件开始

APM价值

CTO / CIO

- 用户体验可量化
- 建立用户体验为基准的KPI体系
- 行业标准对比
- 提升客户满意度，降低TCO

业务

- 得到真实用户的体验
- 发现卡顿 / 闪退崩溃，并快速解决，留住客户
- 加快迭代和发版速度，增加用户粘度

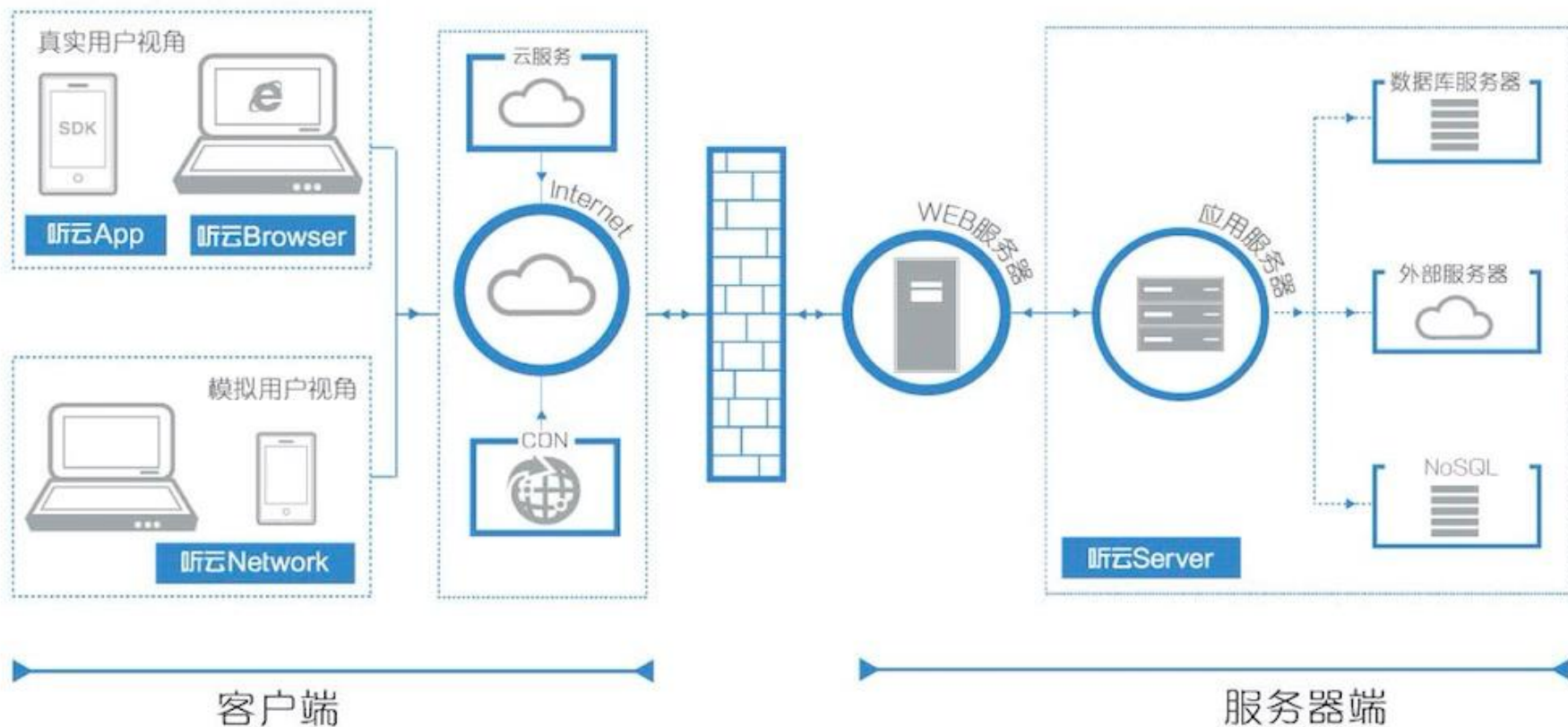
产品 / 研发

- 提高迭代效率
- 生产环境下的代码执行效率和问题定位
- 保留现场，为解决问题提供足够信息

运维

- 业务级运维平台
- 提高运维价值
- 更清晰的责任界定
- 业务 / 研发 / 运维的同平台监控和运维

听云全景图



Thanks!

